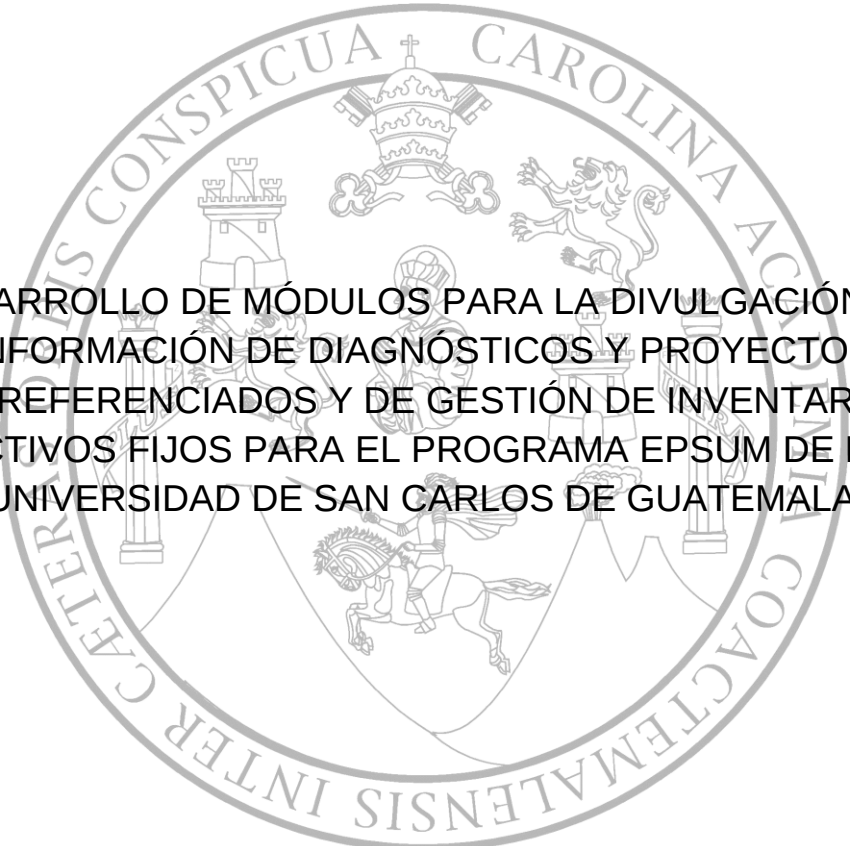


UNIVERSIDAD DE SAN CARLOS DE GUATEMALA
CENTRO UNIVERSITARIO DE ORIENTE
INGENIERÍA EN CIENCIAS Y SISTEMAS

The seal of the University of San Carlos of Guatemala is a circular emblem. It features a central shield with a figure on horseback, a crown above it, and a lion on the right. The shield is flanked by two towers. The outer ring of the seal contains the Latin text "SICUT ERAT IN CONSPICUA CAROLINA ACUTIA COACTEMALENSIS INTER CETERA".

DESARROLLO DE MÓDULOS PARA LA DIVULGACIÓN DE
INFORMACIÓN DE DIAGNOSTICOS Y PROYECTOS
GEORREFERENCIADOS Y DE GESTIÓN DE INVENTARIO DE
ACTIVOS FIJOS PARA EL PROGRAMA EPSUM DE LA
UNIVERSIDAD DE SAN CARLOS DE GUATEMALA

CARLOS ORLANDO PAIZ RAYMUNDO

CHIQUMULA, GUATEMALA, FEBRERO 2019

UNIVERSIDAD DE SAN CARLOS DE GUATEMALA
CENTRO UNIVERSITARIO DE ORIENTE
INGENIERÍA EN CIENCIAS Y SISTEMAS

DESARROLLO DE MÓDULOS PARA LA DIVULGACIÓN DE
INFORMACIÓN DE DIAGNÓSTICOS Y PROYECTOS
GEORREFERENCIADOS Y DE GESTIÓN DE INVENTARIO DE
ACTIVOS FIJOS PARA EL PROGRAMA EPSUM DE LA
UNIVERSIDAD DE SAN CARLOS DE GUATEMALA

TRABAJO DE GRADUACIÓN

Sometido a consideración del Honorable Consejo Directivo

Por

CARLOS ORLANDO PAIZ RAYMUNDO

Al conferírsele el título de

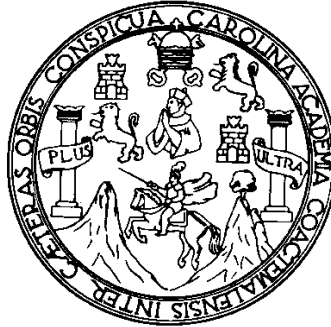
INGENIERO EN CIENCIAS Y SISTEMAS

En el grado académico de

LICENCIADO

CHIQUMULA, GUATEMALA, FEBRERO 2019

**UNIVERSIDAD DE SAN CARLOS DE GUATEMALA
CENTRO UNIVERSITARIO DE ORIENTE
INGENIERÍA EN CIENCIAS Y SISTEMAS**



RECTOR
M.Sc. Ing. MURPHY OLYMPO PAIZ RECINOS

CONSEJO DIRECTIVO

Presidente:	Ing. Agr. Edwin Filiberto Coy Cordón
Representante de Profesores:	M.Sc. Mario Roberto Díaz Moscoso
Representante de Profesores:	M.Sc. Gildardo Guadalupe Arriola Mairén
Representante de Graduados:	Inga. Evelin Dee Dee Sumalé Arenas
Representante de Estudiantes:	P.C. Diana Laura Guzmán Moscoso
Representante de Estudiantes:	M.E.P. José Roberto Martínez Lemus
Secretaria:	Licda. Marjorie Azucena González Cardona

AUTORIDADES ACADÉMICAS

Coordinador Académico:	M.A. Edwin Rolando Rivera Roque
Coordinador de Carrera:	Ing. Carlos Enrique Aguilar Rosales

ORGANISMO COORDINADOR DE TRABAJOS DE GRADUACIÓN

Presidente:	M.Sc. Luis Fernando Quijada Beza
Secretario:	MBA. René Estuardo Alvarado González
Vocal:	Ing. Elder Avildo Rivera López

TERNA EVALUADORA

Presidente:	MBA. René Estuardo Alvarado González
Secretario:	Ing. Carlos David Ardón Muñoz
Vocal:	MBA. Samy Eunice Pinto Castañeda

Chiquimula, marzo de 2019.

Honorable Consejo Directivo:

En cumplimiento con los preceptos que establece la ley de la Universidad de San Carlos de Guatemala y del Centro Universitario de Oriente, presento a su consideración mi trabajo de graduación titulado:

DESARROLLO DE MÓDULOS PARA LA DIVULGACIÓN DE INFORMACIÓN DE
DIAGNÓSTICOS Y PROYECTOS GEORREFERENCIADOS Y DE GESTIÓN DE
INVENTARIO DE ACTIVOS FIJOS PARA EL PROGRAMA EPSUM DE LA
UNIVERSIDAD DE SAN CARLOS DE GUATEMALA.

Tema que me fuera asignado por la coordinación de las carreras de ingeniería, con fecha septiembre de 2018.

Deferentemente,



Carlos Orlando Paiz Raymundo



SEPC.01.18

Chiquimula, 14 de noviembre de 2018.

MBA. René Estuardo Alvarado González
Coordinador del Ejercicio Profesional Supervisado –EPS-
De la Carrera de Ingeniería en Ciencias y Sistemas
CUNORI – USAC

Respetable Ingeniero:

Atentamente me dirijo a usted para hacer de su conocimiento que he revisado y efectuado las correcciones del caso al trabajo de graduación: **“DESARROLLO DE MÓDULOS PARA LA DIVULGACIÓN DE INFORMACIÓN DE DIAGNÓSTICOS Y PROYECTOS GEORREFERENCIADOS Y DE GESTIÓN DE INVENTARIO DE ACTIVOS FIJOS PARA EL PROGRAMA EPSUM DE LA UNIVERSIDAD DE SAN CARLOS DE GUATEMALA”**, presentado por el estudiante **CARLOS ORLANDO PAIZ RAYMUNDO**.

Habiendo llenado dicho trabajo los requisitos, me permito aprobarlo en calidad de Asesor - Revisor del mismo.

“ID Y ENSEÑAD A TODOS”

MBA. Samy Eunice Pinto Castañeda
Asesor - Revisor

C.c. archivo



CEPSICYS.01.2019

Chiquimula, 10 de enero de 2019.

Ing. Carlos Enrique Aguilar Rosales
Coordinador Carreras de Ingeniería
CUNORI - USAC

Respetable Ingeniero Aguilar Rosales:

Atentamente me dirijo a usted para informarle que he revisado el trabajo de graduación titulado: **“DESARROLLO DE MÓDULOS PARA LA DIVULGACIÓN DE INFORMACIÓN DE DIAGNÓSTICOS Y PROYECTOS GEORREFERENCIADOS Y DE GESTIÓN DE INVENTARIO DE ACTIVOS FIJOS PARA EL PROGRAMA EPSUM DE LA UNIVERSIDAD DE SAN CARLOS DE GUATEMALA”**, elaborado por el estudiante Carlos Orlando Paiz Raymundo, quien contó con la asesoría de la MBA. Samy Eunice Pinto Castañeda

Considero que el trabajo desarrollado por el estudiante Carlos Orlando Paiz Raymundo, satisface los requisitos exigidos, por lo cual recomiendo su aprobación.

Agradezco a usted la atención a la presente, atentamente,

“ID Y ENSEÑAD A TODOS”

MBA. René Estuardo Alvarado González
Coordinador del Ejercicio Profesional Supervisado (E.P.S)
Carrera de Ingeniería en Ciencias y Sistemas

C.c. archivo



UNIVERSIDAD DE SAN CARLOS DE GUATEMALA
CENTRO UNIVERSITARIO DE ORIENTE –CUNORI-
CARRERAS DE INGENIERÍA



ALTG.9.18

Chiquimula, 26 de noviembre de 2018.

Ing. Carlos Enrique Aguilar Rosales
Coordinador Carreras de Ingenierías
CUNORI-USAC

Estimado Ing. Aguilar Rosales:

El propósito de la presente, es para informarle que he procedido a revisar la parte lingüística, del trabajo de investigación titulado **“DESARROLLO DE MÓDULOS PARA LA DIVULGACIÓN DE INFORMACIÓN DE DIAGNÓSTICOS Y PROYECTOS GEORREFERENCIADOS Y DE GESTIÓN DE INVENTARIO DE ACTIVOS FIJOS PARA EL PROGRAMA EPSUM DE LA UNIVERSIDAD DE SAN CARLOS DE GUATEMALA”**; elaborado por el estudiante **Carlos Orlando Paiz Raymundo**.

El informe cumple con los requisitos exigidos, por la carrera de Ingeniería en Ciencias y Sistemas, por lo tanto, recomiendo su aprobación para seguir con los trámites correspondientes.

Agradeciendo su atención a la presente,

Deferentemente,

ID Y ENSEÑAD A TODOS


Lic. Julio César Hernández Ortiz
Revisor Área Lingüística

C.c. archivo



UNIVERSIDAD DE SAN CARLOS DE GUATEMALA
CENTRO UNIVERSITARIO DE ORIENTE -CUNORI-
CARRERAS DE INGENIERÍA



TGICYS.01-2019

Chiquimula, 14 de enero de 2019.

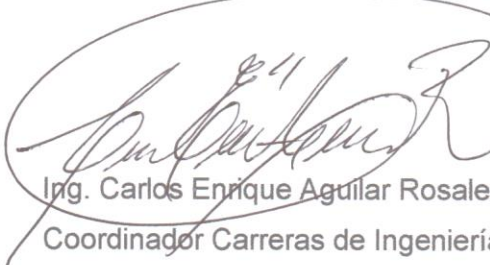
Ing. Agr. Edwin Filiberto Coy Cordón
Director
Centro Universitario de Oriente

CUNORI - USAC

Respetable Ing. Agr. Coy Cordón:

El coordinador de las Carreras de Ingeniería del Centro Universitario de Oriente CUNORI, de la Universidad de San Carlos de Guatemala, después de conocer el dictamen del asesor-revisor MBA. Samy Eunice Pinto Castañeda y del Coordinador del Ejercicio Profesional Supervisado de la Carrera de Ingeniería en Ciencias y Sistemas el MBA. René Estuardo Alvarado González, y luego de la revisión y aprobación del Lic. Julio Cesar Hernández Ortiz, revisor del área de Lingüística, al trabajo de graduación del estudiante Carlos Orlando Paiz Raymundo: **“DESARROLLO DE MÓDULOS PARA LA DIVULGACIÓN DE INFORMACIÓN DE DIAGNÓSTICOS Y PROYECTOS GEORREFERENCIADOS Y DE GESTIÓN DE INVENTARIO DE ACTIVOS FIJOS PARA EL PROGRAMA EPSUM DE LA UNIVERSIDAD DE SAN CARLOS DE GUATEMALA”**, da por este medio su aprobación a dicho trabajo de graduación y recomendando la autorización del mismo.

“ID Y ENSEÑAD A TODOS”


Ing. Carlos Enrique Aguilar Rosales
Coordinador Carreras de Ingeniería
CUNORI - USAC



C.c. archivo

EL INFRASCRITO DIRECTOR DEL CENTRO UNIVERSITARIO DE ORIENTE DE LA UNIVERSIDAD DE SAN CARLOS DE GUATEMALA, POR ESTE MEDIO HACE CONSTAR QUE: Conoció el Trabajo de Graduación que efectuó el estudiante **CARLOS ORLANDO PAIZ RAYMUNDO** titulado “**DESARROLLO DE MÓDULOS PARA LA DIVULGACIÓN DE INFORMACIÓN DE DIAGNÓSTICOS Y PROYECTOS GEORREFERENCIADOS Y DE GESTIÓN DE INVENTARIO DE ACTIVOS FIJOS PARA EL PROGRAMA EPSUM DE LA UNIVERSIDAD DE SAN CARLOS DE GUATEMALA**”, trabajo que cuenta con el aval del Revisor y Coordinador de Trabajos de Graduación, de la carrera de Ingeniería en Sistemas. Por tanto, la Dirección del CUNORI con base a las facultades que le otorga las Normas y Reglamentos de Legislación Universitaria **AUTORIZA** que el documento sea publicado como **Trabajo de Graduación** a Nivel de Licenciatura, previo a obtener el título de **INGENIERO EN CIENCIAS Y SISTEMAS**.

Se extiende la presente en la ciudad de Chiquimula, el veintiséis de febrero de dos mil diecinueve.

“ID Y ENSEÑAD A TODOS”



Ing. Agr. Edwin Fillberto Coy Cordón
DIRECTOR
CUNORI – USAC



AGRADECIMIENTOS A:

Dios	Creador de todo cuanto existe. Por la vida y toda la serie de eventos que me han traído a este momento.
Mi familia	No merecía tanto en el universo.
Mis formadores	Todos y cada uno: desde los que me enseñaron a leer, hasta los que me enseñaron cálculo diferencial. ¿De verdad estaría <i>aquí</i> si no fuera por ellos?
Mis amigos	No hace falta mencionarlos por nombre, ustedes saben quiénes son.
Mi asesora	Samy Eunice Pinto Castañeda, quien además de acompañarme en este proyecto, fue quien me enseñó a programar. Nada del trabajo concretado hubiera sido posible sin su dedicación para colocar las bases del conocimiento de un nuevo mundo.
Programa EPSUM	Por abrirme las puertas y permitirme formar parte de su misión, de convivir y mostrarme cómo cientos de estudiantes están justo en este momento, aplicando los conocimientos adquiridos dentro de las aulas de la Universidad de San Carlos de Guatemala y llevándolos a los rincones más necesitados del país.

ACTO QUE DEDICO A:

**Orlando Paiz y Alicia
Raymundo**

Mis padres. Quizá este acto no sea lo suficiente comparado con el esfuerzo que ustedes han hecho para conmigo y mis hermanos: todas las jornadas de trabajo, día y noche, para que un día sus hijos alcanzaran sus sueños. Ni en un millón de años luz, estoy seguro, encontraría a dos seres con sus mismas cualidades, con su misma paciencia y cariño. Si aún esto no fuera suficiente, y sin ánimos de jactarme, quiero reconocer de manera pública, lo feliz que me han hecho por siempre apoyarme en lo académico, principalmente. Nadie más que ustedes saben las alegrías de los muchos reconocimientos que he obtenido en mi vida, pues siempre estuvieron ahí, comprándome los crayones con los que gané un concurso de dibujo, o dándome los libros que estudié para ganar una olimpiada de ciencias. No pretendo justificar con mis logros, todo lo que me han dado, porque también son sus logros; más bien, con todo esto solo quiero decir, que su esfuerzo no ha sido en vano.

Agradezco a Dios, porque en este momento, están *aquí*. Merecen tanto y, sin embargo, esto es por y para ustedes. Perdón por hacerles esperar.

ÍNDICE GENERAL

ÍNDICE DE ILUSTRACIONES	I
GLOSARIO	V
RESUMEN	IX
INTRODUCCIÓN	XI
OBJETIVOS	XIII
1. FASE DE INVESTIGACIÓN	1
1.1. Antecedentes de la organización	1
1.1.1. Acerca de la organización	1
1.1.2. Estructura organizativa	2
1.1.3. Objetivos	3
1.1.3.1. Objetivo General	3
1.1.3.2. Objetivos Específicos	4
1.1.4. Descripción del trabajo de la organización	4
1.1.5. Productos o servicios que realiza	6
1.1.6. Principales necesidades que plantea la organización	8
1.2. Análisis de los procesos actuales del negocio	8
1.2.1. Información generada por equipos multidisciplinarios	10
1.2.1.1. Informe diagnóstico	10
1.2.1.2. Plan de trabajo	14
1.2.1.3. Perfil de proyecto	15
1.2.1.4. Ficha de resultados de proyectos	15
1.3. Antecedentes del problema	16
1.4. Planteamiento y definición del problema o necesidad	17
1.5. Delimitación del problema	18
1.5.1. Delimitación geográfica	18
1.5.2. Delimitación institucional	18

1.5.3.	Delimitación personal	18
1.5.4.	Delimitación temporal	19
1.5.5.	Delimitación teórica	19
1.6.	Justificación del proyecto	19
2.	FASE DE TRABAJO TÉCNICO PROFESIONAL	21
2.1.	Descripción del proyecto	21
2.2.	Fundamentación teórica	22
2.2.1.	Aplicación Web	22
2.2.2.	Servidor Web	23
2.2.3.	Arquitectura de aplicación	24
2.2.4.	Tecnologías de desarrollo	25
2.3.4.1.	PHP	25
2.3.4.2.	Laravel	26
2.3.4.3.	JavaScript	28
2.3.4.4.	MySQL	28
2.3.4.5.	Google Maps Javascript API	29
2.2.5.	Sistemas de información georreferenciada	30
2.2.6.	Rational Unified Process	31
2.2.6.1.	Fases RUP	32
2.2.6.2.	Flujos de trabajo	35
2.3.	Características del producto final	37
2.4.	Requerimientos funcionales	37
2.5.	Requerimientos no funcionales	40
2.6.	Organización del equipo de trabajo	40
2.6.1.	Participantes en el proyecto	40
2.6.2.	Roles y responsabilidades	41
2.7.	Cronograma de actividades	41
2.8.	Desarrollo del plan de trabajo	44

2.8.1.	Modelado del negocio	45
2.8.1.1.	Modelado de casos de uso	45
2.8.2.	Modelado de objetos de negocio	49
2.9.	Especificaciones de casos de uso	50
2.9.1.	Registro de diagnóstico	50
2.9.2.	Retomar edición de diagnóstico	52
2.9.3.	Actualizar información de diagnóstico	53
2.9.4.	Registrar perfil de proyectos	54
2.9.5.	Actualizar información de proyectos	57
2.9.6.	Registro de plan de trabajo	58
2.9.7.	Actualización de información en el plan de trabajo	63
2.9.8.	Finalización de proyectos	65
2.9.9.	Registro de ficha de resultados	66
2.9.10.	Consulta de información de diagnósticos y proyectos	68
2.9.11.	Generación de reportes de diagnósticos y/o proyectos	70
2.9.12.	Gestión de inventario de activos fijos	70
2.9.12.1.	Registro de proveedores	72
2.9.12.2.	Actualización de información de proveedores	73
2.9.12.3.	Baja de proveedores	74
2.9.12.4.	Registro de activos fijos	75
2.9.12.5.	Actualización de información de bienes	76
2.9.12.6.	Registro de compras	78
2.9.12.7.	Consulta de registros de inventario	79
2.9.12.8.	Exportación de registros de inventario	80
2.9.13.	Especificaciones adicionales	81
2.10.	Análisis y diseño	82
2.10.1.	Modelo de análisis y diseño	82
2.10.1.1.	Diagrama de Clases	82
2.10.1.2.	Diagrama de estados	86

2.10.1.3.	Diagrama de secuencia	90
2.10.1.4.	Modelo de datos	92
2.11.	Implementación	96
2.11.1.	Diagrama de paquetes	97
2.11.2.	Herramientas de desarrollo	99
2.11.2.1.	Software de terceros	99
2.11.3.	Pruebas	100
2.11.4.	Configuración	100
2.11.5.	Despliegue	101
2.12.	Gestión del proyecto	103
2.12.1.	Evaluación de iteraciones	103
2.13.	Beneficios del proyecto	108
2.14.	Valoración económica del proyecto	109
3.	FASE DE ENSEÑANZA-APRENDIZAJE	113
3.1.	Capacitación	113
3.1.1.	Programación de capacitaciones	113
3.1.2.	Certificación de capacitaciones	116
3.2.	Documentación de la solución	116
3.2.1.	Guía de usuario para registro de informes	116
3.2.2.	Guía de usuario de Geoportal del Programa EPSUM	117
3.2.3.	Guía de uso para módulo de gestión de inventario	117
3.2.4.	Manual de especificaciones técnicas y mantenimiento	117
3.3.	Aceptación del proyecto	117
	CONCLUSIONES	119
	RECOMENDACIONES	121
	REFERENCIAS BIBLIOGRÁFICAS	123
	APÉNDICES	125
	ANEXOS	205

ÍNDICE DE ILUSTRACIONES

FIGURAS

1.	Modelo de fortalecimiento de gestión pública, EPSUM	2
2.	Estructura organizacional interna de EPSUM	3
3.	Metodología de intervención, EPSUM	6
4.	Procesos en las fases de metodología de intervención	9
5.	Proceso general de recopilación de informes	10
6.	Planificación con base a resultados	14
7.	Arquitectura de tres niveles	24
8.	Arquitectura de una Aplicación de Laravel	27
9.	Fases de Metodología RUP	32
10.	Caso de uso: gestionar información de diagnóstico y proyectos	45
11.	Diagrama de caso de uso: registrar diagnóstico	46
12.	Diagrama de caso de uso: registrar proyecto y plan de trabajo	47
13.	Caso de uso: gestionar información de inventario	47
14.	Diagrama de caso de uso: registrar ingreso en inventario	48
15.	Caso de uso: consulta de información pública	48
16.	Modelo de dominio de registro de informes	49
17.	Modelo de dominio de registro en inventario	49

18.	Diagrama de clases de módulo para la gestión de informes diagnósticos y de proyectos	83
19.	Diagrama de clases de gestión de inventario de activos fijos	85
20.	Diagrama de estados para el registro de diagnósticos	86
21.	Diagrama de estados para el registro de general de proyectos	88
22.	Diagrama de estados para el registro de general de inventario	89
23.	Diagrama de secuencia	90
24.	Modelo relacional	93
25.	Diagrama de paquetes	98
26.	Diagrama de despliegue	102

TABLAS

1.	Proyectos desarrollados según ámbito de intervención	7
2.	Indicadores de informe diagnóstico	12
3.	Roles y responsabilidades de los participantes del proyecto	41
4.	Cronograma de actividades	42
5.	Especificación de caso de uso: registrar diagnóstico	50
6.	Especificación de caso de uso: continuar editando diagnóstico	52
7.	Especificación de caso de uso: editar diagnóstico	53
8.	Especificación de caso de uso: registrar proyecto multidisciplinario o de convivencia	54

9.	Especificación de caso de uso: registrar proyecto monodisciplinario	56
10.	Especificación de caso de uso: editar información de proyecto	57
11.	Especificación de caso de uso: registrar objetivos específicos, metas e indicadores	58
12.	Especificación de caso de uso: registrar actividades	59
13.	Especificación de caso de uso: Agregar un objetivo específico al plan de trabajo de un proyecto existente	60
14.	Especificación de caso de uso: Agregar actividades al plan de trabajo de un proyecto existente	61
15.	Especificación de caso de uso: editar objetivo específicos, metas e indicadores	63
16.	Especificación de caso de uso: editar actividades	64
17.	Especificación de caso de uso: marcar actividad como hecha	65
18.	Especificación de caso de uso: registrar ficha de resultados de proyectos	66
19.	Especificación de caso de uso: consulta de diagnóstico	68
20.	Especificación de caso de uso: consulta de proyectos	69
21.	Especificación de caso de uso: generar reportes de diagnósticos y/o proyectos	70
22.	Especificación de caso de uso: registrar proveedores	72
23.	Especificación de caso de uso: editar proveedor	73
24.	Especificación de caso de uso: dar de baja a proveedores	74
25.	Especificación de caso de uso: registrar bien mueble	75
26.	Especificación de caso de uso: editar bienes	76
27.	Especificación de caso de uso: registrar compras	78

28.	Especificación de caso de uso: consultar información de inventario	79
29.	Especificación de caso de uso: exportar información de inventario	80
30.	Iteraciones para el módulo de registro y consulta de informes	103
31.	Iteraciones para el módulo de gestión de inventario	104
32.	Riesgos inherentes	106
33.	Criterios de nivel de riesgo	107
34.	Valoración de riesgo	107
35.	Resultados de valoración de riesgo	107
36.	Perfiles de recurso humano	109
37.	Costos de recurso humano	110
38.	Tiempo asignado a las actividades desarrolladas	110
39.	Costo total del proyecto	111
40.	Planificación de presentaciones	114
41.	Planificación de capacitaciones	115

GLOSARIO

ACID	ACID es el acrónimo de <i>Atomicity, Consistency, Isolation and Durability</i> ; Atomicidad, Consistencia, Aislamiento y Durabilidad en español. En bases de datos se denomina ACID a las características de los parámetros que permiten clasificar las transacciones de los sistemas de gestión de bases de datos.
API	Acrónimo de <i>Application Programming Interface</i> , representa una llave de acceso a funciones que podemos utilizar de un servicio web provisto por un tercero, dentro de nuestra propia aplicación web, de manera segura y confiable.
<i>Backend</i>	<i>Backend</i> es el área que se dedica a la parte lógica de un sitio web, es el encargado de que todo funcione como debería. Parte de una aplicación que no es visible para el usuario
BSD	La licencia BSD es una licencia de software otorgada principalmente para los sistemas BSD (<i>Berkeley Software Distribution</i>), un tipo del sistema operativo Unix-like.

Feedback

Anglicismo que se refiera a la retroalimentación: capacidad de un emisor para recoger reacciones de los receptores y modificar su mensaje, de acuerdo con lo recogido.

Frontend

Frontend se refiere a la parte del desarrollo web que se centra en la vista de un sitio web, en pocas palabras del diseño de un sitio web, desde la estructura del sitio hasta los estilos como colores, fondos, tamaños hasta llegar a las animaciones y efectos.

FTP

El Protocolo de transferencia de archivos (en inglés *File Transfer Protocol* o FTP), es un protocolo de red para la transferencia de archivos entre sistemas conectados a una red TCP (*Transmission Control Protocol*), basado en la arquitectura cliente-servidor.

HTTP

Abreviatura de la forma inglesa *Hypertext Transfer Protocol*, 'protocolo de transferencia de hipertextos', que se utiliza en algunas direcciones de Internet.

Script

Archivo de órdenes o archivo de procesamiento por lotes, es un programa usualmente simple, que por lo regular se almacena en un archivo de texto plano.

SIG

Un sistema de información geográfica (también conocido con los acrónimos SIG en español o GIS en inglés) es un conjunto de herramientas que integra y relaciona diversos componentes que permiten la organización, almacenamiento, manipulación, análisis y modelización de grandes cantidades de datos procedentes del mundo real que están vinculados a una referencia espacial.

Stakeholder

Anglicismo que hace referencia a las personas u organizaciones que toman parte en las decisiones de una actividad o proyecto y que se ven afectadas por las mismas.

UML

El lenguaje unificado de modelado (UML, por sus siglas en inglés, *Unified Modeling Language*) es el lenguaje de modelado de sistemas de software más conocido y utilizado en la actualidad; está respaldado por el *Object Management Group* (OMG).

RESUMEN

El Programa EPSUM, con el objetivo de mejorar el tratamiento de la información generada por los equipos multidisciplinarios, proyectó el seguimiento a la extensión de la plataforma MiE, implementada en el primer semestre de 2017. El proyecto permite, además de agilizar el proceso de revisión de informes y de centralizar procesos administrativos, dar a conocer información que el Programa condensa de manera pública.

Derivado de lo anterior, se desarrollaron tres proyectos orientados a contribuir con el registro para la divulgación de la información generada por los equipos multidisciplinarios y que concluyó con la creación de un geoportal. Así también, se apoyó con el registro de información administrativa correspondiente a la tesorería del Programa, a través de un sistema de gestión de inventario de activos fijos.

En general, el objetivo de las soluciones tecnológicas implementadas, ha sido automatizar los procesos de registros públicos y administrativos del Programa, a efecto de contribuir con el ordenamiento, divulgación y control de la información mediante los módulos integrados en la plataforma de EPSUM.

INTRODUCCIÓN

La gestión de la información constituye parte fundamental en cualquier organización. La misma debe añadir valor con el fin de generar conocimiento y proyectar los resultados de la entidad. Por tal razón, es de suma importancia la selección, procesamiento y divulgación de dicha información para dar cuenta de la consecución de objetivos propuestos por la organización.

Para el Programa EPSUM, la necesidad de automatizar el proceso de recopilación de información generada por el creciente número de estudiantes que se asignan al Programa, representó uno de los principales problemas que repercutía en la proyección del trabajo que los equipos multidisciplinarios realizan. Además, la descentralización de información administrativa que manejan las tesorerías del Programa, dilucidaron una necesidad en la automatización de registros de inventario. Por lo anterior, se realizó el estudio pertinente para el desarrollo de soluciones de software orientadas a contribuir con el ordenamiento, divulgación y control de la información pública y administrativa a través de módulos integrados en la plataforma de EPSUM.

En primera instancia, se delimitó el proyecto “Gestión de informes diagnósticos y de proyectos”, integrando a la plataforma existente del Programa, los formularios correspondientes para el registro de diagnóstico, planes de trabajo y perfiles de proyectos realizados por los epesistas. Consecutivamente, se elaboró el proyecto “Desarrollo e implementación de módulo georreferencial para la divulgación de información de equipos multidisciplinarios de EPSUM”, el cual muestra a través de un geoportal, la investigación y el trabajo registrado por los estudiantes en cada una de las sedes, proyectando socialmente al Programa a través de la visualización de información de dominio público. Así también, se

implementó el proyecto “Módulo de Gestión de inventario de activos fijos del Programa EPSUM” el cual permitió centralizar la información de tesorería a fin de llevar un control sobre los bienes del Programa y agilizar el proceso de registro mediante la sistematización del mismo.

En la Fase de Investigación se describe a EPSUM, como la entidad para la cual se desarrollaron las soluciones; se incluye la definición y el planteamiento del problema, así como la justificación del proyecto.

En la Fase Técnico-Profesional se hace una descripción del proyecto y se contextualiza sobre la metodología y tecnologías utilizadas, a través de la fundamentación teórica; se definen los objetivos del proyecto y se listan los involucrados en el mismo. Además, se expone la aplicación de la metodología RUP, definiendo el modelado del negocio y especificaciones de casos de uso, de acuerdo a los requerimientos establecidos por la Coordinación del Programa. Posteriormente en la fase de elaboración, se definen los diagramas correspondientes que logran una visión general de la funcionalidad del proyecto, y que son concretados en la fase de construcción con la codificación de los requerimientos funcionales del sistema. Por último, se explica la implementación y el despliegue de los nuevos módulos integrados a la plataforma de EPSUM.

En la Fase de Enseñanza-Aprendizaje se describe brevemente las actividades de capacitación realizadas.

OBJETIVOS

General

Desarrollar módulos de software para la divulgación de diagnósticos y proyectos georreferenciados y de gestión de inventario de activos fijos para el Programa EPSUM de la Universidad de San Carlos de Guatemala.

Específicos

1. Elaborar un módulo para el registro de información generada por los equipos multidisciplinarios para crear una base de datos de diagnósticos y proyectos del Programa EPSUM.
2. Crear un sistema de consulta de información georreferenciada registrada por los estudiantes que integran los equipos multidisciplinarios, a fin de proyectar el trabajo realizado desde el Programa EPSUM.
3. Implementar un módulo para la gestión y control de inventario de activos fijos, para centralizar la información de las tesorerías del Programa EPSUM.
4. Capacitar a estudiantes, supervisores y personal administrativo del Programa EPSUM para el uso de los módulos de software desarrollados.

1. FASE DE INVESTIGACIÓN

1.1. Antecedentes de la organización

1.1.1. Acerca de la organización

EPSUM, de la Universidad de San Carlos de Guatemala, es un programa que integra equipos multidisciplinarios, quienes conjuntamente con las Unidades Académicas, el apoyo de municipalidades, ministerios, secretarías y otras dependencias del Gobierno y organizaciones internacionales, diseña y ejecuta proyectos de desarrollo integral, con pertinencia local y sostenibilidad para mejorar las condiciones de vida de la población.

El programa EPSUM es parte de la Dirección General de Extensión Universitaria -DIGEU- y tiene como precedente directo las Prácticas Profesionales Multidisciplinarias llamadas PROSAC. Desde 1976, la DIGEU realizó esfuerzos para generar un programa de prácticas supervisadas multidisciplinarias que abarcara la totalidad de disciplinas presentes en la Universidad. En 1983, el Consejo Superior Universitario formó la Comisión EPSUM, tres años después, el 26 de febrero de 1986, se aprobaron los subprogramas, metodología, ejes y fases de desarrollo, organización y divulgación del programa y el 17 de abril de 1991 se dio la aprobación de su estructura administrativa, asignándose al mismo un presupuesto para su funcionamiento. (EPSUM, 2017)

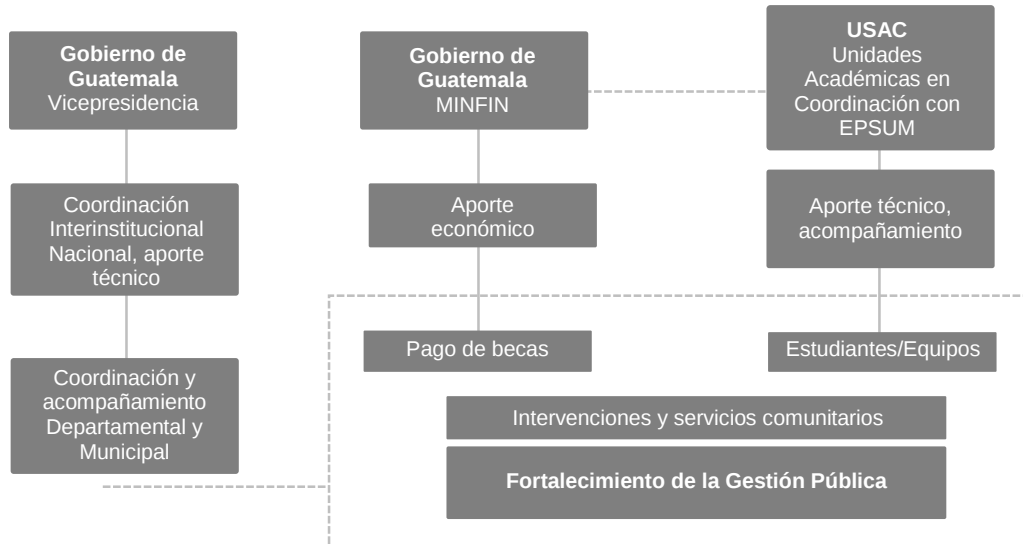
Actualmente, EPSUM trabaja en conjunto con la Vicepresidencia de la República de Guatemala y el Ministerio de Finanzas Públicas, en el marco del

Convenio de Cooperación Interinstitucional de atención a la población más vulnerable, en los 55 municipios con indicadores de pobreza más altos del país.

1.1.2. Estructura organizativa

EPSUM funciona bajo un modelo interinstitucional que fortalece la estructura operacional del programa, por medio del vínculo Universidad-Sociedad-Estado, como muestra la figura 1, se promueve un modelo de desarrollo humano que busca atender a la población vulnerable en todo el país, a través de gobernaciones, municipalidades, puestos de salud y centros de atención permanente (EPSUM, 2017).

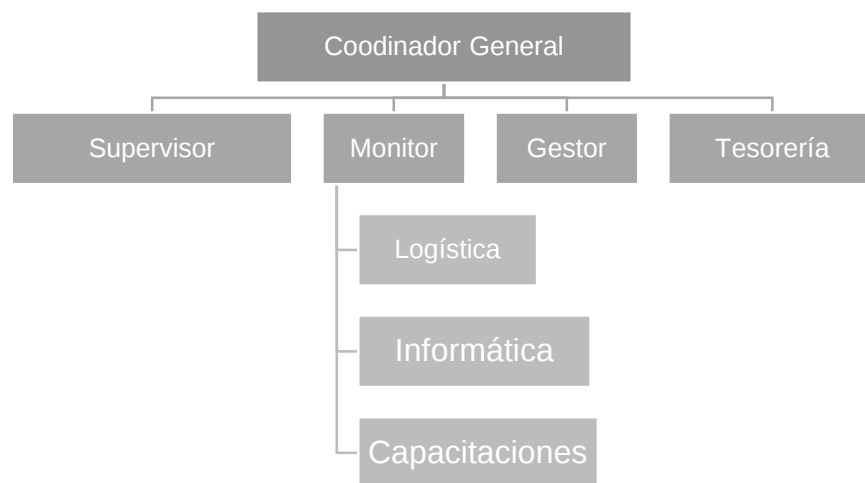
Figura 1. **Modelo de fortalecimiento de gestión pública, EPSUM**



Fuente: Programa EPSUM 2016.

La estructura administrativa de EPSUM, mostrada en la figura 2, está encabezada por la coordinación general del programa, la cual tiene a su cargo potencializar en forma efectiva la realización de los proyectos asignados a las distintas sedes.

Figura 2. **Estructura organizacional interna de EPSUM**



Fuente: elaboración propia

1.1.3. Objetivos

1.1.3.1 Objetivo General

Contribuir al desarrollo humano de la población en los municipios priorizados por pobreza y pobreza extrema del país, por medio de la participación de equipos multidisciplinarios de diversas disciplinas de las unidades académicas de la Universidad de San Carlos de Guatemala. (Programa EPSUM, 2016)

1.1.3.2 Objetivos Específicos

1. Contribuir a la formación integral de estudiantes universitarios, incorporándolos a la realidad nacional.
2. Fortalecer la gestión y autogestión de comunidades, a través de equipos multidisciplinarios de estudiantes.
3. Contribuir al desarrollo rural del país a través de programas, proyectos y acciones que los equipos multiprofesionales intervienen a través de su práctica. (Programa EPSUM, 2016)

1.1.4. Descripción del trabajo de la organización

EPSUM atiende a la población más vulnerable del país a través de los equipos multidisciplinarios que se distribuyen en los 74 municipios cubiertos por el programa (EPSUM, 2017). Puesto que su objetivo es cooperar en la solución integral de la problemática nacional, en relación a las comunidades con índice de pobreza extrema, el programa se centra en los siguientes ámbitos de intervención:

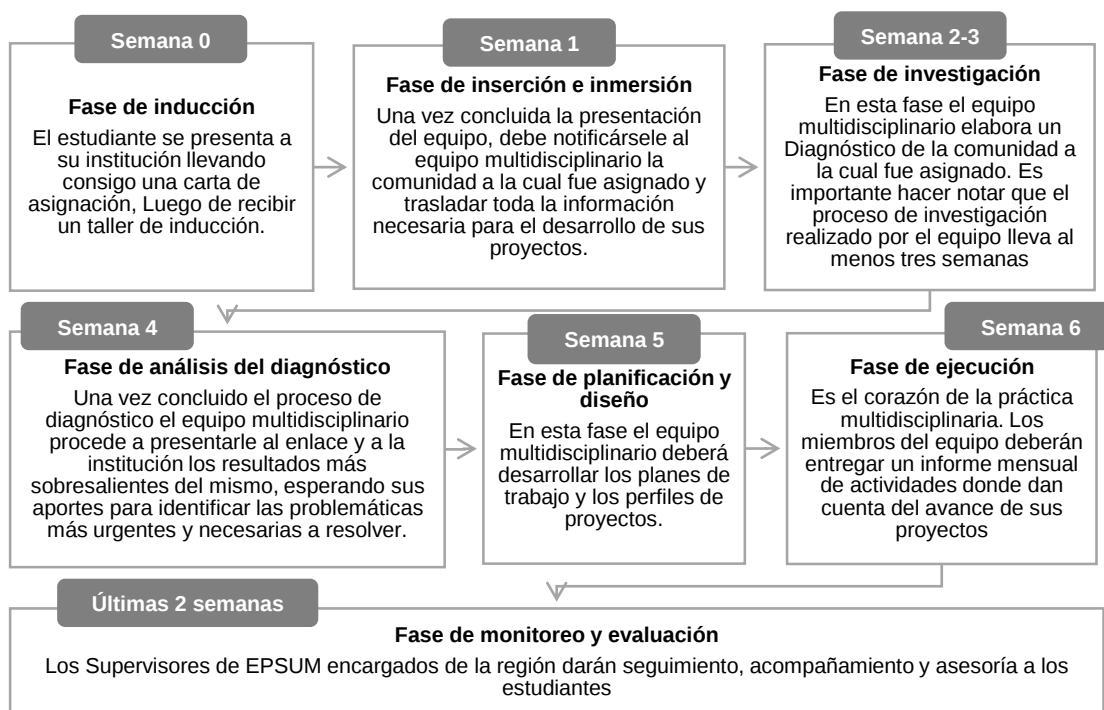
- Economía
- Educación
- Medio ambiente
- Salud

De los ámbitos de intervención del programa, derivan los ejes de intervención interinstitucional, los cuales tienen como fin promover un modelo de desarrollo humano (EPSUM, 2017):

- Desarrollo local, extensión rural, mejoramiento de las economías campesinas
- Apoyo en la atención a salud primaria y salud comunitaria.
- Fortalecimiento y acceso de alimentos locales, incidiendo en la reducción de la vulnerabilidad nutricional.
- Fortalecimiento al sistema educativo, diagnóstico y propuesta técnica de infraestructura local.
- Vulnerabilidad a sequía, adaptación al cambio climático, resiliencia.
- Fortalecimiento de capacidades institucionales en la gestión pública.

Para la consecución de resultados en los distintos ámbitos de intervención, EPSUM sigue una metodología de trabajo dividida en 7 fases, como se muestra en la figura 3, la cual se basa en la lógica de planificación y gestión de proyectos, con la cual todos los integrantes de los equipos multidisciplinarios desarrollan su práctica profesional.

Figura 3. Metodología de Intervención



Fuente: elaboración propia.

Una implementación óptima de la metodología conlleva un monitoreo constante por parte de supervisores del programa y de las unidades académicas. Además, los equipos deben presentar un informe mensual sobre el avance y desarrollo de sus proyectos, los cuales son evaluados. Al finalizar el proceso, los equipos presentan resultados, alcances e impactos de sus proyectos tanto multidisciplinario, como monodisciplinarios a su sede de práctica, comunidad, unidad académica y Programa EPSUM.

1.1.5. Productos o servicios que realiza

Derivado de la aplicación de la metodología presentada en la figura 3, se generan 3 productos principales:

- El desarrollo de un proyecto multidisciplinario que aborde una problemática a nivel local.
- El desarrollo de un proyecto de convivencia comunitaria en equipo, que fortalezca la organización local.
- El desarrollo de proyectos individuales desde un enfoque monodisciplinario.

Estos productos aplicados a cualquier ámbito de intervención que el programa ha definido, coadyuvan a promover un desarrollo humano local a través de los proyectos ejecutados, además de fortalecer las instituciones con las cuales se coordinan los proyectos. La tabla 1 muestra el aporte económico producido por proyectos en cada ámbito de intervención en 2016.

Tabla 1. **Proyectos desarrollados según ámbito de intervención, 2016**

Ámbito	No. de proyectos	Usuarios directos	Usuarios indirectos	Aporte económico (Q)
Agricultura familiar	6	383	1,010	221,600.00
Economía	3	584	4,979	189,550.00
Educación	4	552	5,742	216,599.00
Medio ambiente	11	7,483	54,285	579,895.00
Salud	9	2,805	11,015	441,972.00
Total general	33	11,807	77,031	1 649,616.00

Fuente: Programa EPSUM, 2016.

Para el año 2017 se proyectaron 67 proyectos enfocados en temas como: salud, economía, educación, cultura y seguridad; los cuales involucraron más de 21 políticas públicas vinculadas.

1.1.6. Principales necesidades que plantea la organización

Uno de los principales problemas que enfrenta el programa EPSUM, es la deficiencia en el proceso de recopilación de información en las fases intermedias y finales de los proyectos. Por tal razón y buscando agilizar el tiempo de evaluación de informes generados por los distintos equipos asignados en el programa, se implementó en el primer semestre de 2017 el Sistema de Información de Sedes, Equipos y Estudiantes asignados al Programa de Ejercicio Profesional Supervisado Multidisciplinario.

El sistema de registro abre la puerta para mejorar la administración de los datos generados por la institución. Sin embargo, existen deficiencias en el proceso de recopilación y divulgación de la información de los proyectos y equipos.

Por otra parte, se ha manifestado una necesidad en la gestión de operaciones de inventario, puesto que el proceso actual de registro es demorado y la consulta de información resulta difícil para los interesados en la misma.

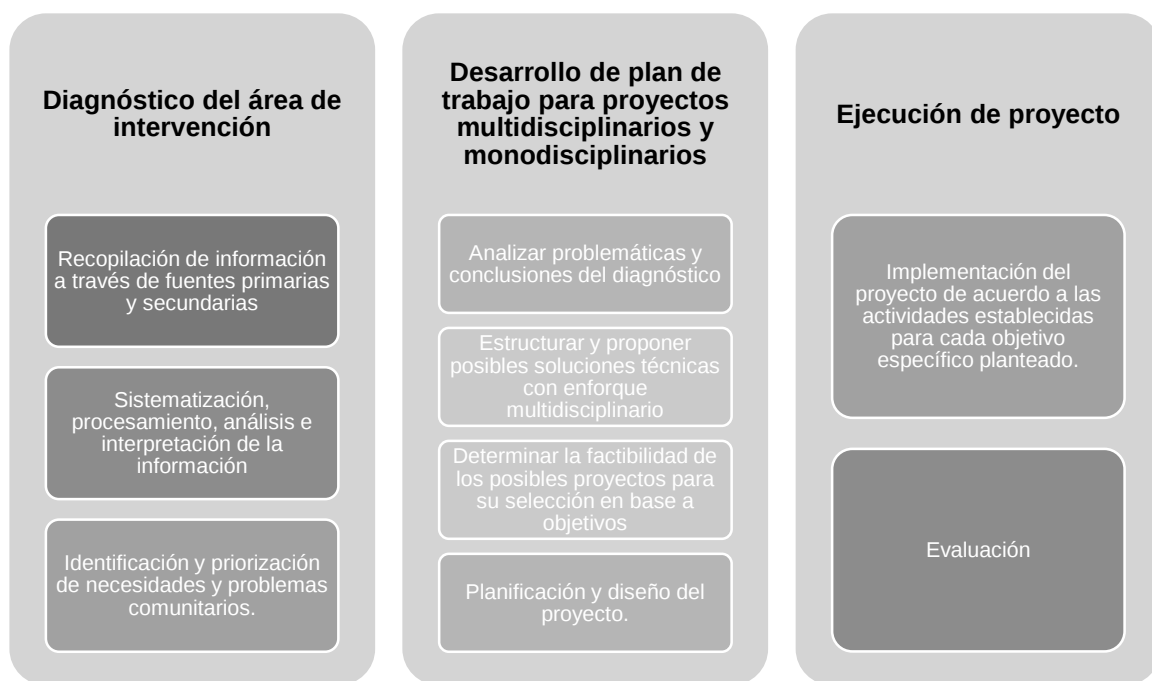
1.2. Análisis de los procesos actuales del negocio

La ejecución de proyectos multidisciplinarios sigue la metodología de intervención planteada en la figura 3. Derivado de la metodología se realizan algunos instrumentos que corresponden al Informe Diagnóstico Comunitario y Plan de Trabajo (como se muestra en la figura 4), los cuales son imprescindibles ya que están orientados a fortalecer de manera integrada con la comunidad y la institución receptora del equipo multidisciplinario, la organización comunitaria y el desarrollo humano local.

Los supervisores de área, se dan a la tarea de recopilar los instrumentos por cada equipo de cada una de las sedes. Una vez recopilada la información, se procede a su evaluación por parte de los mismos supervisores para su aprobación y/o corrección, tal como muestra el diagrama de la figura 5, de manera que se realice el proyecto eficientemente.

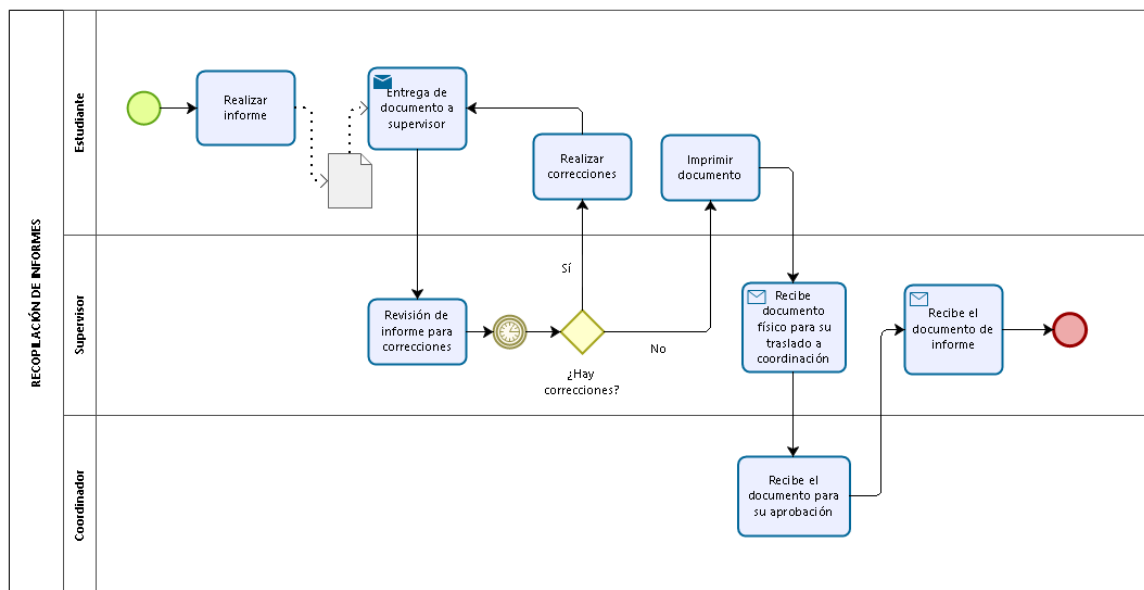
Durante la fase final del proyecto, debe generarse un informe final que muestre los resultados obtenidos a partir de la ejecución del mismo, el cual es recopilado nuevamente por los supervisores de cada región y evaluados para la entrega final (EPSUM, 2017).

Figura 4. Procesos en las fases de metodología de intervención



Fuente: elaboración propia.

Figura 5. Proceso general de recopilación de informes



Fuente: elaboración propia.

Con respecto a la gestión del inventario, su operación se realiza a través del personal de tesorería, quienes registran los bienes activos en una hoja de cálculo y generan la respectiva ficha de responsabilidad, para luego imprimir la información en formatos elaborados y almacenarla en archivo.

1.2.1. Información generada por equipos multidisciplinarios

1.2.1.1. Informe diagnóstico

El informe diagnóstico es el documento iniciado en la fase de investigación que realiza en conjunto el equipo multidisciplinario. El mismo recopila información para conocer, explorar y analizar el territorio asignado. En la Guía

Metodológica para equipos multidisciplinarios se establece la finalidad del diagnóstico:

El diagnóstico diseñado por el Programa EPSUM tiene como finalidad describir tres grandes ámbitos comunitarios:

La historia de la comunidad: en donde se describen los hechos históricos, desde su fundación, desarrollo, fiestas patronales, que se convierte en la identidad de la misma y que sirve para comprender la transformación local.

El estado socioeconómico y cultural: en este apartado se analizan los aspectos de la realidad social de la comunidad, datos de economía, salud, educación y organización entre otros.

El ambiente biofísico: describe el entorno físico-biológico, aborda recursos naturales, datos climáticos, hidrológicos, geológicos, además de amenazas y vulnerabilidades.

La finalidad de la realización del diagnóstico es generar una imagen científica e investigativa del conglomerado de comunidades sujeto de estudio, con un enfoque y análisis integral y complementario de la formación de las disciplinas que integran el equipo multidisciplinario. (EPSUM, 2017, p.50)

La estructura del informe diagnóstico lo componen una serie de indicadores que ayudan a estudiar el caso particular de la comunidad, bajo los ámbitos anteriormente descritos. Los indicadores se muestran en la tabla 2.

Tabla 2. **Indicadores de informe diagnóstico**

Ámbito	Indicadores
Territorio	Ubicación geográfica Coordenadas geográficas Altura sobre el nivel del mar Superficie (km ²) División política
Demografía	Población varones Población mujeres Población rural Población urbana Población entre 0 y 34 años de edad Población entre 15 y 64 años de edad Población de 65 años de edad y más Población indígena Población garífuna Población xinca Población por comunidad lingüística Población monolingüe en su idioma
Vivienda	Total de viviendas en área urbana Total de viviendas en área rural
Niveles de Pobreza	Pobreza en área urbana Pobreza rural Pobreza en población indígena Pobreza en población no indígena Pobreza en mujeres Pobreza en hombres Pobreza extrema Pobreza urbana Pobreza rural
Educación	Tasa de alfabetización Tasa de alfabetización en mujeres Tasa de alfabetización en varones Tasa de analfabetismo Tasa de analfabetismo en mujeres Tasa de analfabetismo en varones Inscripción escolar a nivel primario, secundario y terciario Tasa de repitencia a nivel primario, secundario y terciario Tasa de deserción escolar Sistema de alimentación escolar

Continuación de Tabla 2

Salud	Mortalidad de mujeres en edad fértil Mortalidad materna Mortalidad infantil Distribución de pacientes atendidos según género Distribución de pacientes atendidos según grupo etario Distribución de principales patologías encontradas Número de pacientes en edad fértil que asisten a planificación familiar Número de pacientes embarazadas que asisten a control prenatal Número de pacientes puérperas evaluadas en el puesto de salud Distribución de inmunizaciones administradas en el puesto de salud
Economía	Tasa de población económica activa Ingreso por familia Principales actividades económicas Comercio formal mujeres Comercio formal hombres Comercio informal mujeres Comercio informal hombres Tasa de desempleo en mujeres Tasa de desempleo en hombres
Medio Ambiente	Zona de vida Áreas protegidas Cobertura boscosa Temperatura Precipitación pluvial Suelos Uso de la tierra para tareas agrícolas Capacidad del uso del suelo Intensidad de uso de la tierra
Conflictividad social	Violencia intrafamiliar Violencia sexual Violencia contra la niñez y adolescencia Trata de personas
Organización comunitaria	Organizaciones socio-económicas

Fuente: EPSUM, 2017.

1.2.1.2. Plan de trabajo

La planificación de trabajo define los proyectos a realizar, partiendo del análisis participativo de los resultados del diagnóstico. Este proceso se realiza en la fase de planificación y diseño de proyectos, presentada en la figura 3.

La planificación se realiza con base a objetivos (componentes) y resultados; cada objetivo específico (componente) del proyecto se desglosa en una serie de actividades que producen un resultado. El proyecto de convivencia comunitaria, debe corresponder a un objetivo específico del proyecto multidisciplinario, conteniendo actividades programadas durante el desarrollo de la práctica supervisada multidisciplinaria. (EPSUM, 2017)

El plan de trabajo busca estructurar soluciones técnicas con enfoque multi e interdisciplinario. La planificación en base a resultados para un proyecto se muestra en la figura 6.

Figura 6. **Planificación en base a resultados**

Proyecto		
Proceso de diagnóstico (problemáticas identificadas)	Planificación del proyecto	
	Objetivo General	Objetivos específicos
		Actividades
	Objetivos Específicos (componentes)	Recursos disponibles

Fuente: EPSUM, 2017.

1.2.1.3. Perfil de proyecto

De acuerdo a la Guía Metodológica para Equipos Multidisciplinarios, se define el perfil del proyecto como sigue:

El perfil de proyecto pretende ser un resumen sistemático del proyecto, donde se pueda identificar lo que se ha planificado realizar, se desarrolla un perfil de proyecto que contenga el proyecto multidisciplinario y comunitaria, no siendo necesario la realización de perfiles por proyectos individuales o monodisciplinarios, el cual debe contener lo siguiente:

- Nombre del proyecto
- Ubicación del proyecto
- Descripción del proyecto
- Objetivos
 - General
 - Específicos
- Línea de intervención
- Aporte técnico asociado al proyecto. (EPSUM, 2017)

1.2.1.4. Ficha de resultados de proyectos

La ficha de resultados constituye parte importante en el informe final, pues en él se resumen los objetivos planteados en el plan de trabajo y los resultados obtenidos como parte de la intervención realizada por el equipo o el estudiante que ejecutó el plan.

1.3. Antecedentes del problema

La gestión de grandes volúmenes de información descentralizada constituye un problema que dificulta la organización y el tratamiento de la misma. Desde sus inicios, EPSUM se ha valido de distintos medios para el proceso de recopilación de datos, siendo los informes en físico el primer recurso utilizado para la verificación de resultados de los proyectos ejecutados bajo el programa.

Posteriormente se utilizaron disquetes como medios de almacenamiento. Sin embargo, existía el inconveniente de la limitación en cuanto a la cantidad de memoria que los mismos ofrecían. Por tal razón, a partir de 2008 se optó por la utilización de discos compactos (CD), con los cuales era posible adjuntar distintos documentos referentes al proyecto, así como fotografías de la ejecución del mismo. Al igual que los medios anteriores, los CD eran enviados a la sede central de EPSUM, para su posterior revisión por parte de los supervisores.

Con el fin de facilitar la organización de la información, durante la coordinación para la segunda cohorte de 2015 se recurrió a la utilización de formularios a través de la plataforma Typeform, en la cual se registraban los resultados más relevantes de la ejecución de los proyectos realizados. Derivado de la utilización de dicha plataforma, se obtienen los primeros registros de informes, mediante una base de datos.

La coordinación general, consciente de la necesidad de un sistema para el tratamiento de la información generada por los distintos equipos multidisciplinarios, buscó en 2017 la implementación del Sistema de Información de Sedes y Equipos asignados al programa. El proyecto, que estuvo a cargo de Kevin Hared González, epesista de Ingeniería en Ciencias y Sistemas, del Centro

Universitario de Oriente y asesorado por el Ingeniero René Estuardo Alvarado González, estableció la base de datos inicial para el posterior registro de las principales entidades a través de un portal en Internet. Asimismo, se implementó una solución para la gestión de contratos y pagos de ayuda becaria a los estudiantes participantes en el programa, con lo cual se redujeron los tiempos para la realización de dichos procesos.

A partir del segundo semestre de 2017, se comenzó el proceso para la recopilación de informes generados por los equipos en las distintas fases de la metodología de trabajo implementada por EPSUM. Sin embargo, la información reportada por los estudiantes en los distintos proyectos distribuidos en todo el país, aún es inaccesible al público en general. Esto representa un problema para EPSUM, puesto que dificulta la divulgación de la información y, por consiguiente, la proyección social del programa.

1.4. Planteamiento y definición del problema o necesidad

La evaluación de resultados en los proyectos ejecutados bajo el programa EPSUM constituye una etapa fundamental para medir el impacto de los mismos en la población. Sin embargo, la recopilación y divulgación de información referente a diagnósticos y proyectos representa una necesidad que se debe principalmente a los grandes volúmenes de información generada por los equipos multidisciplinarios y el orden temporal en el cual los estudiantes entregan su trabajo. Sumado a esto, el Sistema de Información General de EPSUM, no contempla el registro de dichos informes que permitan dar cuenta de los proyectos desarrollados a través del programa y los recursos destinados a las partes involucradas para la ejecución de los mismos. Como consecuencia de lo anterior, el acceso a la información está restringida al público, lo que dificulta la

proyección de EPSUM como ente cooperador a las poblaciones más vulnerables del país.

Por otra parte, entre las operaciones internas de la institución, se ha encontrado un proceso demorado en el registro y consulta de activos de inventario, esto como consecuencia de la utilización de herramientas poco adecuadas que impiden agilizar dicho proceso.

1.5. Delimitación del problema

1.5.1. Delimitación geográfica

El proyecto se efectuó en las instalaciones de EPSUM, en el Centro Cultural Universitario (antiguo Paraninfo Universitario), de la Universidad de San Carlos de Guatemala, ubicado en 2a. Avenida, 12-40 Zona 1, Guatemala, Guatemala.

1.5.2. Delimitación institucional

Para efectos del presente trabajo, se limitó la obtención de la información a la suministrada por la coordinación del Programa EPSUM.

1.5.3. Delimitación personal

La fase de investigación y de desarrollo para este proyecto involucró a Ing. Mirna Regina Valiente, Coordinadora del Programa EPSUM; Mónica Saraí Pereira Sical, Tesorera de la institución y a Lic. Max José Castillo, Gestor y Vinculador del programa. Además, se contó con la asesoría de MBA. Samy Eunice Pinto Castañeda.

1.5.4. Delimitación temporal

El proyecto de EPS se realizó en un período de 6 meses, comprendido de enero a julio del año 2018. Se asignaron los tiempos de gestión de proyectos tal como se presenta en la tabla 4 del apartado 2.7 y según establece el artículo 5, inciso c, del Normativo del Ejercicio Profesional Supervisado de Ingeniería.

1.5.5. Delimitación teórica

El proyecto constituyó módulos de aplicación web conectado al sistema de información de EPSUM. El mismo se realizó siguiendo la metodología de desarrollo de software *Rational United Process* (RUP).

1.6. Justificación del proyecto

El programa EPSUM, cumpliendo con lo establecido en el artículo 82 de la Constitución Política de la República de Guatemala para la Universidad de San Carlos de Guatemala: "(...) promoverá por todos los medios a su alcance la investigación en todas las esferas del saber humano y cooperará al estudio y solución de los problemas nacionales", tiene como principal objetivo, contribuir con el desarrollo humano local.

Para lograr tal objetivo, EPSUM ha establecido que la manera más eficaz y factible de conseguirlo es la investigación, planificación, diseño, ejecución y seguimiento de proyectos comunitarios (EPSUM, 2017). En ese sentido, y teniendo en cuenta que debe llevarse un monitoreo constante de la elaboración de los proyectos, se hace uso de informes y/o instrumentos específicos para la

evaluación del cumplimiento de los objetivos propuestos en la planificación de los mismos. Sin embargo, la recopilación de los informes generados por los equipos multidisciplinarios constituye una dificultad para EPSUM, debido a los grandes volúmenes de información obtenida.

Derivado de lo anterior, al no existir un centro de recopilación que almacene los informes de los equipos multidisciplinarios, se dificulta la divulgación de resultados obtenidos en los proyectos realizados, lo que hace poco visible la proyección social del programa y el aporte económico que el mismo genera.

Por otra parte, dentro de las actividades que realiza el personal administrativo de EPSUM, se encuentra el registro de bienes activos que forman el inventario de la institución. Sin embargo, la utilización de herramientas poco adecuadas, ralentizan el proceso, lo que dificulta el aval de registros por parte de la tesorería.

2. FASE DE TRABAJO TÉCNICO PROFESIONAL

2.1. Descripción del proyecto

El proyecto desarrollado constituye un módulo para la recopilación y divulgación de la información referente a los proyectos realizados bajo el Programa EPSUM a nivel nacional, implementado como un sistema de información georreferenciada y de ahora en adelante denominado: Geoportal de EPSUM. La utilización de sistemas de georreferenciación, se presentan como una solución para la organización y consulta de información de manera eficiente para este tipo de problema.

El módulo hace uso de lenguajes como PHP y el *framework* Laravel para el *backend*; HTML y Javascript para el *frontend*. Además, se utiliza la API de Google Maps a razón de la implementación de un mapa interactivo desde el cual puede seleccionarse un marcador para mostrar información detallada de cada uno de los equipos, entre la cual destacan los informes diagnósticos y perfiles de proyectos. El Geoportal de EPSUM permite filtrar la búsqueda de información por cohorte, departamento y equipo para hacer las mismas más eficientes.

En cuanto al módulo de gestión de inventario de activos fijos, se utilizaron las tecnologías disponibles y se realizaron las modificaciones necesarias a la base de datos actual para la implementación de dicho módulo, que contempló el registro de proveedores y compras, además del ingreso de activos fijos.

Para ambos proyectos, se siguieron las especificaciones de la metodología RUP para el desarrollo de software y se utilizaron las tecnologías disponibles con las que cuenta el Programa.

2.2. Fundamentación teórica

2.2.1. Aplicación Web

Desde sus inicios, la aparición de Internet abrió un sinfín de posibilidades para el acceso a la información desde cualquier sitio, de tal manera que la web fue pensada como un medio para desplegar información estática, contenida en servidores.

Aunque en un principio, el acceso remoto de la información permitió una facilidad en la comunicación, la necesidad de una interacción dinámica entre el usuario y el servidor seguía latente. Naturalmente, el concepto de la web fue evolucionando hasta adoptar una característica dinámica, con la cual la información puede actualizarse como resultado de la ejecución en el servidor de alguna lógica de programación (García-Caro, 2006).

En ese sentido, una aplicación web puede definirse como aquella que es accesible desde un servidor web a través de una red, generalmente Internet, mediante un navegador web que soporta el lenguaje o tecnología con el cual dicha aplicación fue desarrollada (Mateu, 2004). La utilización de aplicaciones web cargadas desde Internet resulta una opción ventajosa al facilitar el mantenimiento y accesibilidad de dichas aplicaciones sin la necesidad de distribuir e instalar un software adicional en la máquina del usuario, además, dan la posibilidad de ser ejecutadas en múltiples plataformas.

2.2.2. Servidor Web

Un servidor web resulta fundamental en el desarrollo de aplicaciones del lado del servidor. Mateu (2004) explica: “Un servidor web es un programa que atiende y responde a las diversas peticiones de los navegadores, proporcionándoles los recursos que solicitan mediante el protocolo HTTP o el protocolo HTTPS (la versión segura, cifrada y autenticada de HTTP)” (p. 23).

Dependiendo del tipo de petición, el servidor web buscará una página web o bien ejecutará un programa en el servidor. De cualquier modo, siempre devolverá algún tipo de resultado HTML (HyperText Mark-up Language) al cliente que realizó la petición (García-Caro, 2006).

Un servidor web básico tiene un esquema de funcionamiento muy sencillo, ejecutando de forma infinita el bucle siguiente:

1. Espera peticiones en el puerto TCP asignado (el estándar para HTTP es el puerto 80).
2. Recibe una petición.
3. Busca el recurso en la cadena de petición.
4. Envía el recurso por la misma conexión por donde ha recibido la petición
5. Vuelve al punto 2 (Mateu, 2004).

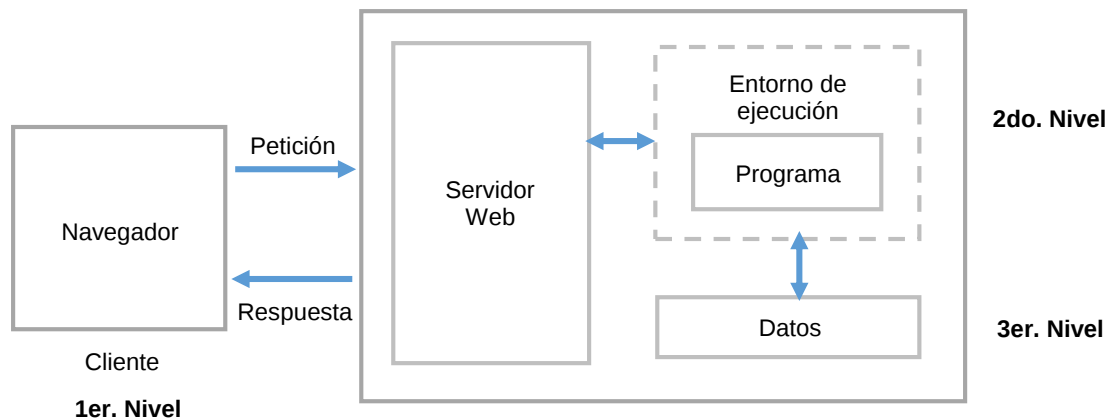
Aunque muy básico, el esquema anterior es la base a partir de la cual se han diseñado y construido todos los programas servidores, según el tipo de petición que atienden, como lo son Servlets, Java Server Pages, XML, entre otros.

2.2.3 Arquitectura de aplicación

Las aplicaciones basadas en una arquitectura cliente-servidor, pertenecen a la categoría de aplicaciones de dos niveles. En este tipo de aplicaciones, el peso del cálculo recae en el cliente, mientras que el servidor realiza la parte menos pesada (García-Caro, 2006). Por tal razón, las aplicaciones web conviene presentarlas dentro de una arquitectura multinivel, aunque de manera simplificada, como arquitectura de tres niveles. Dicha arquitectura distingue tres capas: Interfaz de presentación, lógica de la aplicación y de datos.

La capa de datos se encarga de gestionar los registros de la aplicación, que comúnmente se encuentran en un servidor de base de datos o como documentos XML (eXtensible Stylesheet Language). La capa intermedia representa el código que contiene la lógica de negocio para recuperar los datos deseados. Por último, la capa de presentación, recibe los datos para mostrarlos adecuadamente al usuario. La separación entre la capa de presentación y de lógica, permite gran flexibilidad debido a que pueden realizarse distintas interfaces sin necesidad de cambiar la lógica de la aplicación.

Figura 7. **Arquitectura de tres niveles**



Fuente: elaboración propia.

La figura 7 muestra un esquema de arquitectura de 3 niveles, en donde el cliente, representa el primer nivel, el cual interactúa con el servidor web, que también forma parte del mismo nivel y que es responsable de devolver los datos en un formato adecuado. El segundo nivel se refiere a un programa o script soportado por el entorno de ejecución y el tercer nivel proporciona datos al segundo para su ejecución.

2.2.4. Tecnologías de desarrollo

Se refiere a los lenguajes de programación, tecnologías y software utilizado para la realización del proyecto.

2.2.4.1. PHP

PHP (acrónimo recursivo de PHP: *Hypertext Preprocessor*) es un lenguaje de scripting de uso general de código abierto ampliamente utilizado que es especialmente adecuado para el desarrollo web y puede integrarse en HTML (PHP, 2017).

Para este lenguaje, el código es interpretado por un servidor web con un módulo de procesador de PHP que genera la página web resultante y tiene la ventaja de contar con diversos módulos y gran cantidad de librerías que son de código abierto.

PHP ha evolucionado por lo que ahora incluye también una interfaz de línea de comandos que puede ser usada en aplicaciones gráficas independientes. Puede ser usado en la mayoría de los servidores web al igual que en casi todos los sistemas operativos y plataformas sin ningún costo. Mateu (2004) afirma:

PHP suele ser utilizado conjuntamente con Perl, Apache, MySQL o PostgreSQL en sistemas Linux, formando una combinación de bajo costo, potente y versátil. Tal ha sido la expansión de esta combinación que incluso ha merecido conocerse con un nombre propio LAMP, formado por las iniciales de los diversos productos. (p. 187)

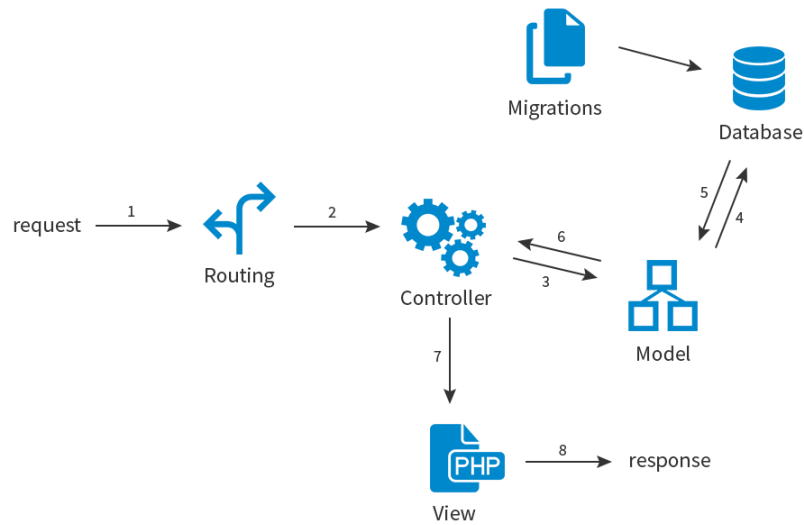
2.2.4.2. Laravel

Laravel es un *framework* de código abierto para el desarrollo de aplicaciones web en PHP, creado en 2011 por Taylor Otwell. Esta tecnología facilita el desarrollo, simplificando el trabajo con tareas comunes como la autenticación, el enrutamiento, la gestión de sesiones y de base de datos (Stauffer, 2016). Laravel está inspirado en Ruby on Rails y Symfony, de los cuales ha adoptado sus principales ventajas:

- Diseñado para trabajar bajo el patrón MVC, permitiendo una correcta separación y modularización del código.
- Integración de un sistema ORM de mapeado de datos relacional, llamado Eloquent. Además, permite la construcción de consultas directas a base de datos mediante su *Query Builder*.
- Permite la gestión de base de datos y la manipulación de tablas desde código, manteniendo un control de versiones de las mismas mediante su sistema de migraciones.
- Utiliza un sistema de plantillas para las vistas llamado Blade, el cual hace uso de la cache para darle mayor velocidad, facilitando así la creación de vistas mediante el uso de *layouts*, herencia y secciones.
- Facilita la extensión de funcionalidad mediante paquetes o librerías externas.

- Incorpora un intérprete de línea de comandos llamado Artisan que facilita la creación de distintos componentes de código, gestión de rutas, migraciones, tareas programadas, entre otras (Gallego Sánchez, 2015).

Figura 8. **Arquitectura de una aplicación de laravel**



Fuente: Stauffer, 2016

Como se observa en la figura 8, la estructura de Laravel está basada en una arquitectura Modelo-Vista-Controlador clásica. En el diagrama se muestra cómo la petición es recogida por un módulo de Routing (1), el cual la redirigirá a un controlador específico (2) que recaba la información necesaria para procesarla y producir una salida esperada. El controlador realiza una llamada a un proveedor de servicios que corresponde a un repositorio de entidades (3), el cual accederá a un modelo según la información solicitada, cargando todos los datos del mismo en memoria (4, 5) y generando una respuesta que será devuelta al controlador (6). Con la información obtenida del modelo, el controlador manejará los datos como se establece en el método y posteriormente, el resultado será mostrado en la vista (7), finalizando de esta manera con la petición.

2.2.4.3. JavaScript

JavaScript (JS) es un lenguaje ligero e interpretado, orientado a objetos con funciones de primera clase, más conocido como el lenguaje de *script* para páginas web, pero también usado en muchos entornos sin navegador, tales como node.js o Apache CouchDB. Es un lenguaje *script* multi-paradigma, basado en prototipos, dinámico, soporta estilos de programación funcional, orientada a objetos e imperativa. (MDN Web Docs, 2017)

La sintaxis básica es similar a Java y C++ con la intención de reducir el número de nuevos conceptos necesarios para aprender el lenguaje. JavaScript puede funcionar como lenguaje procedimental y como lenguaje orientado a objetos. Los objetos se crean programáticamente añadiendo métodos y propiedades a lo que de otra forma serían objetos vacíos en tiempo de ejecución, en contraposición a las definiciones sintácticas de clases comunes en los lenguajes compilados como C++ y Java. Una vez se ha construido un objeto, puede usarse como modelo (o prototipo) para crear objetos similares.

Se utiliza principalmente en su forma del lado del cliente, implementado como parte de un navegador web permitiendo mejoras en la interfaz de usuario y páginas web dinámicas, aunque existe una forma de JavaScript del lado del servidor. Su uso en aplicaciones externas a la web, por ejemplo, en documentos PDF, aplicaciones de escritorio es también significativo.

2.2.4.4. MySQL

MySQL es un sistema de gestión de bases de datos relacional muy rápido, multiproceso, multiusuario y robusto, desarrollado bajo licencia dual

GPL/Licencia comercial por Oracle Corporation. Está diseñado para sistemas de producción de carga crítica y de misión crítica, así como para la integración en software implementado en masa (MySQL, 2017).

MySQL es muy utilizado en aplicaciones web. Su popularidad se debe a su rendimiento probado, a su fiabilidad y a su facilidad de uso. La utilización de MySQL en aplicaciones web está muy ligada a PHP, permitiendo realizar aplicaciones que requieran el uso de una base de datos rápida, segura y potente.

Entre las principales características de MySQL pueden mencionarse:

- Escalabilidad, siendo capaz de manejar bases de datos empotradas ocupando sólo 1MB.
- Flexibilidad, puesto que soporta distintas versiones de Linux, UNIX y Windows.
- Alta disponibilidad, permitiendo la replicación a servidores de clúster especializados.
- Robusto soporte transaccional, incluyendo un soporte completo de ACID, bloqueo a nivel de filas, posibilidad de transacciones distribuidas y soporte de transacciones con múltiples versiones (MySQL, 2017).

2.2.4.5. Google Maps Javascript API

La API de Google Maps para Javascript ofrece la posibilidad de visualizar los mapas de Google y acceder a distintas funciones que permiten personalizarlos según se requiera a través de los servicios de Google que proporcionan datos geográficos como: geocodificación, indicaciones, elevación, sitio, información de zonas horarias, entre otros. La utilización de los servicios

de Google Maps atienden a una tarifa que se aplica según el lucro que se genere con la aplicación.

2.2.5. Sistemas de información georreferenciada

Según ArcGIS Resources (s.f.), un sistema de información georreferenciada o geográfica se define de la siguiente manera:

SIG es un sistema empleado para describir y categorizar la Tierra y otras geografías con el objetivo de mostrar y analizar la información a la que se hace referencia espacialmente. Este trabajo se realiza fundamentalmente con los mapas.

El objetivo de SIG consiste en crear, compartir y aplicar útiles productos de información basada en mapas que respaldan el trabajo de las organizaciones, así como crear y administrar la información geográfica pertinente.

Un sistema de información geográfica, de manera general, es cualquier sistema de información capaz de integrar, almacenar, editar, analizar, compartir y mostrar la información geográficamente referenciada. Los SIG también son herramientas que permiten a los usuarios crear consultas interactivas, analizar la información espacial, editar datos y presentar los resultados de todas estas operaciones.

2.2.6. Rational Unified Process

RUP (*Rational Unified Process*) es un proceso de ingeniería de software. Proporciona un enfoque disciplinado para asignar tareas y responsabilidades dentro de una organización de desarrollo. Su objetivo es garantizar la producción de software de alta calidad que satisfaga las necesidades de sus usuarios finales, dentro de un calendario y un presupuesto predecibles. (IBM, s.f.)

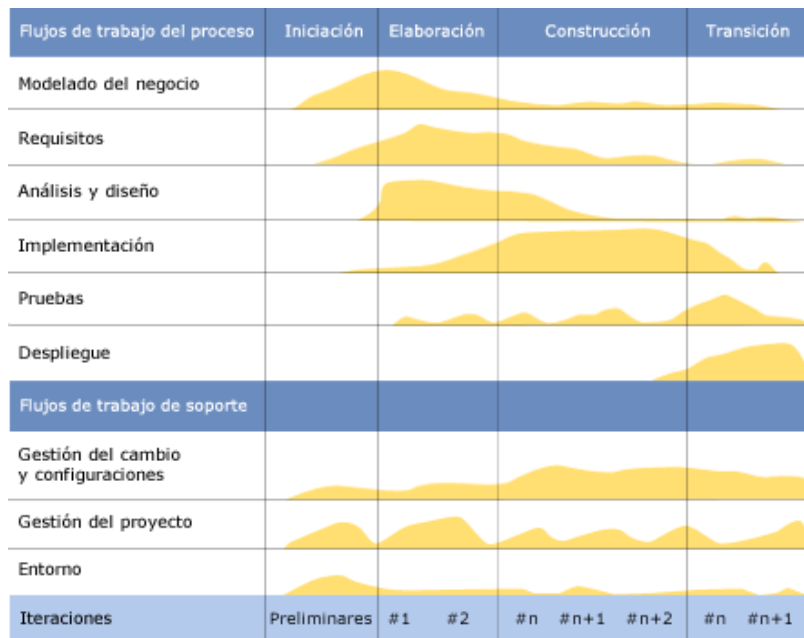
RUP fue desarrollado por la empresa Rational Software, actualmente propiedad de IBM. Junto con el Lenguaje Unificado de Modelado (UML), constituye la metodología estándar más utilizada para el análisis, diseño, implementación y documentación de sistemas orientados a objetos.

Esta metodología mejora la productividad al proporcionar a todos los miembros del equipo un acceso fácil a una base de conocimientos con directrices, plantillas y diversas herramientas para todas las actividades de desarrollo fundamentales, haciendo uso de un lenguaje común, que permite a cada integrante visualizar el avance y desarrollo del proyecto.

El proceso se puede describir en dos ejes o dimensiones, tal como se muestra en la figura 9:

- El eje horizontal representa el tiempo y muestra el aspecto dinámico del proceso a medida que trasciende; se expresa en términos de: ciclos, fases, iteraciones e hitos.
- El eje vertical representa el aspecto estático del proceso: cómo se describe en términos de actividades, artefactos y flujos de trabajo.

Figura 9. Fases de metodología RUP



Fuente: Wikimedia Commons, 2008

2.2.6.1. Fases RUP

El ciclo de vida del software se divide en ciclos, cada ciclo trabaja en una nueva generación del producto. El proceso unificado de *Rational* divide un ciclo de desarrollo en cuatro fases consecutivas (IBM, s.f.).

- Fase de inicio: esta fase tiene como propósito definir y acordar el alcance del proyecto con los *stakeholders* del proyecto. En ella se identifican los riesgos asociados al proyecto y se propone una visión general de la arquitectura de software. Como producto de esta fase, se generan los siguientes artefactos:

- Un documento de visión: una visión general de los requisitos del proyecto central, las características clave y las principales limitaciones.
 - Un modelo de caso de uso inicial (10% -20% completado).
 - Un glosario inicial del proyecto (opcionalmente se puede expresar parcialmente como un modelo de dominio).
 - Una evaluación de riesgo inicial.
 - Especificación de requisitos.
 - Un modelo de negocio, si es necesario.
 - Uno o varios prototipos.
-
- Fase de elaboración: el propósito de la fase de elaboración es analizar el dominio del problema, establecer una base arquitectónica sólida, desarrollar el plan del proyecto y eliminar los elementos de mayor riesgo del proyecto. Las decisiones arquitectónicas deben tomarse teniendo en cuenta todo el sistema: su alcance, funcionalidad principal y requisitos no funcionales, como los requisitos de rendimiento. Los artefactos producidos durante esta fase son:
 - Modelo de casos de uso (al menos 80% completado).
 - Requisitos adicionales que capturan los requisitos no funcionales y cualquier requisito que no esté asociado con un caso de uso específico.
 - Una descripción de arquitectura de software.
 - Diagramas de Clase.
 - Modelo E-R.
 - Diagrama de secuencia.
 - Diagrama de estados.

- Diagrama de colaboración
 - Un plan de desarrollo para el proyecto en general.
-
- Fase de Construcción: durante la fase de construcción, todos los componentes restantes y las características de la aplicación se desarrollan e integran en el producto y todas las características se prueban exhaustivamente. La fase de construcción es, en cierto sentido, un proceso de fabricación en el que se hace hincapié en la gestión de los recursos y el control de las operaciones para optimizar los costes, los horarios y la calidad. En este sentido, la mentalidad de la gerencia pasa por una transición desde el desarrollo de la propiedad intelectual durante el inicio y la elaboración, hasta el desarrollo de productos desplegables durante la construcción y la transición. Como producto de esta fase se obtiene:
 - Pruebas de los casos de uso desarrollados, y pruebas de regresión según sea el caso.
 - Diseño y desarrollo de casos de uso y/o flujos de acuerdo con la planeación iterativa.
 - Especificación de requisitos faltantes.
 - El producto de software integrado en las plataformas adecuadas.
-
- Fase de Transición: el propósito de la fase de transición es hacer la transición del producto de software a los usuarios finales. Una vez que el producto ha sido entregado al usuario final, generalmente surgen problemas que requieren que desarrolle nuevas versiones, corrija algunos problemas o finalice las funciones que se pospusieron. La

fase de transición se centra en las actividades necesarias para poner el software en manos de los usuarios.

Por lo general, esta fase incluye varias iteraciones, incluidas versiones beta, versiones de disponibilidad general, así como versiones de reparación de errores y mejoras. De esta fase resulta:

- Puesta en producción.
- Pruebas de aceptación.
- Capacitación de usuarios.

Cada fase en el proceso unificado de *Rational* se puede dividir en iteraciones. Una iteración constituye un ciclo de desarrollo completo que da como resultado un lanzamiento de un producto ejecutable, un subconjunto del producto final en desarrollo, que crece de manera gradual en cada iteración para convertirse en el sistema final.

2.2.6.2. Flujos de trabajo

Un flujo de trabajo es una secuencia de actividades que produce un resultado de valor observable (IBM, s.f.). Para la consecución del proyecto, RUP hace uso de seis flujos de trabajo principales:

- Modelado de negocio: en este flujo de trabajo se documentan los procesos de negocio a través de casos de uso. Esto asegura un entendimiento común entre todas las partes interesadas sobre qué proceso de negocio necesita ser respaldado en la organización.

- **Requerimientos:** el objetivo del flujo de trabajo de requerimientos es describir qué debe hacer el sistema y permitir a los desarrolladores y al cliente ponerse de acuerdo sobre esa descripción. Para lograr esto, se debe obtener, organizar y documentar las funcionalidades y limitaciones requeridas para el proyecto.
- **Análisis y diseño:** el objetivo de este flujo es mostrar cómo se verá el sistema en la fase de implementación, realizando una abstracción del código.
- **Implementación:** el propósito del flujo de trabajo de implementación es definir la organización del código, implementar clases y objetos en términos de componentes (archivos fuente, binarios, ejecutables y otros), probar los componentes desarrollados como unidades, e integrar los resultados producidos por los equipos en un sistema ejecutable.
- **Pruebas:** en este flujo se debe verificar la interacción entre los objetos definidos, una apropiada integración de todos los componentes del software y que todos los requisitos han sido correctamente implementados.
- **Despliegue:** el objetivo del flujo de trabajo de implementación es producir lanzamientos exitosos de productos y entregar el software a sus usuarios finales.

2.3. Características del producto final

El proyecto ha sido desarrollado en el lenguaje de programación PHP versión 7.0 y utilizando el framework Laravel, versión 5.1. La base de datos la constituye MariaDB y MySQL como DBMS, tecnologías disponibles para el desarrollo del proyecto.

El módulo de divulgación de información de proyectos de EPSUM, cuenta con una interfaz interactiva que permite al usuario consultar los proyectos a través de un mapa en donde se muestra información georreferenciada concerniente a los equipos multidisciplinarios inscritos en el Programa EPSUM. Para su implementación se han agregado al sistema los formularios de registros de instrumentos como informe diagnóstico, perfiles de proyectos y planes de trabajo que constituyen parte reservada para los usuarios registrados en la plataforma. La información en el geoportal puede ser filtrada a través de componentes de fácil uso, como listas desplegables. Se da la opción de generar documentos como reportes acerca de la consulta realizada.

Con respecto al módulo de gestión de inventario, se muestra la información consultada a través de tablas que representan el detalle de cada uno de los registros de activos realizadas por EPSUM. Este módulo también da la opción de generar informes en base a la información consultada.

2.4. Requerimientos funcionales

Para el módulo de divulgación de información de proyectos de EPSUM, se efectuaron los siguientes requerimientos:

- El sistema permite el registro y edición de los documentos y proyectos realizados por los equipos a los usuarios registrados como estudiantes en la plataforma.
- Consideración del registro colaborativo por equipo en cuanto al informe diagnóstico, proyectos multidisciplinario y de convivencia. Se almacena el registro de los cambios realizados por cada integrante del equipo al editar los documentos y/o información.
- Para el módulo de consulta, se muestra la información referente a proyectos de manera georreferenciada a través de un mapa interactivo.
- El mapa interactivo muestra puntos o marcadores que representan los equipos de EPSUM distribuidos en todo el país.
- En la vista general del mapa indican el número de estudiantes asignados al Programa, número de equipos registrados hasta ese momento y el número total de sedes activas en todo el país.
- Se permite filtrar los proyectos según cohorte y año.
- Los proyectos georreferenciados son enlaces a un cuadro que da información detallada del proyecto, en donde se muestra la sede, equipo (integrantes) y perfiles de proyectos realizados por los estudiantes.
- La interfaz da la opción de elegir un departamento específico para ampliar la vista del mapa.

- La vista ampliada por departamento muestra los puntos geográficos para los equipos en el lugar seleccionado.
- Se permite ampliar la información de diagnóstico y perfiles de proyectos para el equipo seleccionado.
- El módulo permite la generación de reportes para la consulta de informes registrados. Los reportes que deberá contemplar el sistema son:
 - Informe de diagnóstico
 - Informe de perfil de proyecto
 - Informe final del proyecto

En el caso del módulo de gestión de inventario, se ha cumplido con los requisitos siguientes:

- Presentación de registros a través de tablas de manera que la información esté organizada para su consulta inmediata por parte de los interesados.
- Generación automática del número correlativo de inventario según el estándar manejado por el Programa EPSUM.
- Consideración del registro de compras/ingresos y proveedores.
- El módulo permite la generación de informes generales, los cuales pueden exportarse como archivos PDF o XLS (Excel).

2.5. Requerimientos no funcionales

- La aplicación está disponible en cualquier momento para un usuario que requiera consultar información en el sistema.
- El módulo es accesible desde cualquier dispositivo con acceso a Internet.
- La interfaz de usuario cumple características que la hacen simple e intuitiva, de manera que cualquier usuario que necesite información del sistema de EPSUM, puede consultarla sin ningún problema.
- Debido a que se trata de un sistema de consulta de información, proporciona toda la facilidad para la exportación de información, la cual atiende a los formatos PDF y XLS (Excel).

2.6. Organización del equipo de trabajo

2.6.1. Participantes en el proyecto

Para la realización de este proyecto, se involucró la Inga. Agr. Mirna Regina Valiente y Licda. Jenifer Pinzón como supervisores del Programa EPSUM; Mónica Saraí Pereira, tesorera y Lic. Max José Castillo, Vinculador del Programa, quienes conformaron los *stakeholders* y proporcionaron información útil para el desarrollo del proyecto. Se integró en el equipo de trabajo a MBA. Samy Eunice Pinto, como asesora del proyecto. Como desarrollador, el titular de este trabajo, Carlos Orlando Paiz Raymundo.

2.6.2. Roles y responsabilidades

Se refiere a los cargos que representan a los involucrados en el proyecto. En la tabla 3 se muestran las responsabilidades de cada uno de los roles en el equipo de desarrollo durante la realización del proyecto.

Tabla 3. Roles y responsabilidades de los participantes del proyecto

Sujeto	Noción	Impacto
Stakeholder	• Es quien da inicio al desarrollo del sistema el cual se construye de acuerdo a sus necesidades. • Tiene participación activa en el desarrollo del sistema.	• Participa en la toma de los requerimientos, acepta y controla su cumplimiento a través de las pruebas de aceptación. • Da su opinión en el Feedback.
Asesor	• Persona con un conocimiento específico en algún tema necesario para el proyecto. • Guía al equipo de desarrollo en la resolución de problemas específicos aportando conocimiento específico sobre un tema en particular.	• Ayuda al programador en la resolución de un problema o inquietud.
Supervisor	• Verifica el avance en el proceso global del proyecto. • Se encarga de coordinar al equipo de desarrollo.	• Participa en la fase de planificación. • En algunas ocasiones participa en la producción.
Desarrollador	• Responsable de desarrollar el sistema	• Generan en cada iteración los artefactos correspondientes a la ejecución de cada una de las fases de la metodología aplicada para el desarrollo del proyecto.

Fuente: elaboración propia.

2.7. Cronograma de actividades

La duración del proyecto fue de 6 meses, que corresponde al período establecido para la realización del Ejercicio Profesional Supervisado en el Normativo de EPS. La distribución del tiempo entre los proyectos se presenta en la tabla 4.

Tabla 4. Cronograma de actividades

Actividad	ENE			FEB				MAR				ABR				MAY					JUN				JUL		
	1	2	3	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	5	1	2	3	4	1	2	3
Proyecto: Módulo de registro y consulta geográfica de proyectos																											
Modelado de negocio y Requerimientos																											
Requerimientos, modelado casos de uso																											
Especificaciones de casos de uso																											
Análisis y diseño																											
Análisis y modelado lógico																											
Codificación																											
Conexión																											
Estructura y componentes del proyecto con Laravel framework																											
Interfaz																											
Registro de diagnóstico																											
Registro de plan de trabajo																											
Registro de informe final																											
Implementación de API de Google Maps																											
Mejor de la interfaz de usuario																											
Primera revisión de funcionalidades con la coordinación																											
Reportes																											
Implementación, Despliegue y pruebas																											

Continuación de Tabla 4

Actividad		ENE			FEB				MAR				ABR				MAY					JUN				JUL		
		1	2	3	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	5	1	2	3	4	1	2	3
	Implementación en el sistema																											
	Segunda revisión de funcionalidades con la coordinación																											
	Evaluación y pruebas																											
	Mejoras y modificaciones																											
	Manual técnico y guía de usuario																											
Proyecto: Módulo de gestión y control de inventario de activos fijos																												
Análisis y Diseño																												
	Análisis y modelado de negocio																											
Codificación																												
	Conexión a base de datos																											
	Estructura del proyecto																											
	Interfaz																											
	Registro de activos																											
	Reportes y fichas de responsabilidad																											
Despliegue y pruebas																												
	Implementación y pruebas																											
	Mejoras y modificaciones																											
	Guía del usuario																											

Fuente: elaboración propia.

De la tabla 4, se observa cómo se ha repartido el tiempo para los proyectos definidos. La tabla se encuentra dividida en dos partes: la primera, agrupa las actividades para módulos de registro de informes y módulo de consulta de información georreferenciada, mostrando en azul los flujos de trabajo correspondientes a la metodología aplicada y en color amarillo las actividades. La segunda parte, muestra las actividades para el módulo de gestión de inventario, en donde se distinguen los flujos de trabajo en color naranja y nuevamente se presentan las actividades en color amarillo.

Es importante destacar que el cronograma representó una planificación inicial y que las casillas marcadas no reflejan un número de horas semanales establecido, sino que limitan una distribución del tiempo, asignando el 65% y 35% respectivamente, entre los módulos indicados. En ese sentido y de acuerdo a la metodología, ésta se aplicó para cada proyecto por separado, con la salvedad de que la fase de inicio que incluye la toma de requerimientos y modelado de negocio para el módulo de gestión de inventario, fue realizada en conjunto con su homóloga para el módulo de registro y consulta de proyectos, a efecto de realizar la planificación del proyecto en general.

2.8. Desarrollo del plan de trabajo

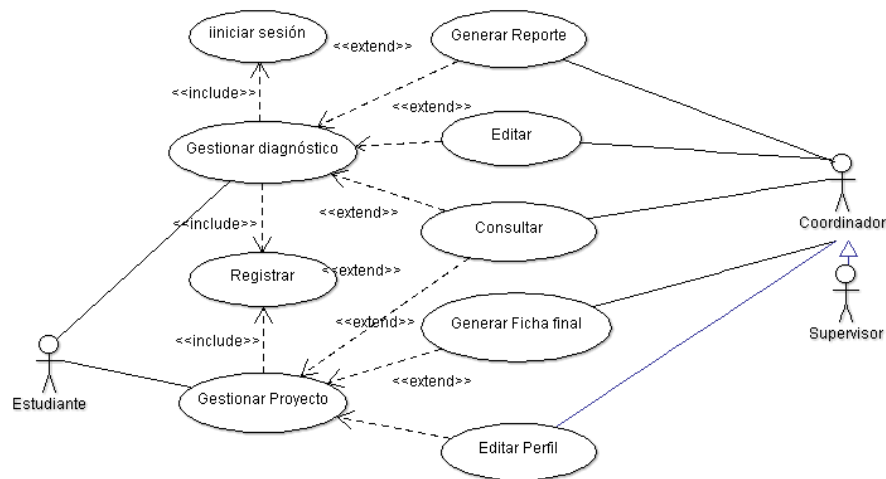
El análisis y diseño de un proyecto de software es fundamental para la realización del mismo. Como producto de la implementación de la metodología RUP para el desarrollo de software, se presentan los artefactos generados en cada uno de los flujos de trabajo.

2.8.1. Modelado del negocio

2.8.1.1. Modelado de casos de uso

El modelado de negocio constituye el primer flujo de trabajo de la metodología RUP. Derivado de este flujo se elaboraron los diagramas de casos de uso, los cuales se enfocan en la representación de las acciones que pueden ejecutar los usuarios dentro del sistema e ilustran procesos de manera generalizada. Según los requerimientos planteados por la Coordinación del Programa EPSUM, para este trabajo se distinguen cinco actores principales: Coordinador, Supervisor, Estudiante, Tesorero y Usuario no registrado (Invitado).

Figura 10. **Caso de uso: gestionar información de diagnósticos y proyectos**

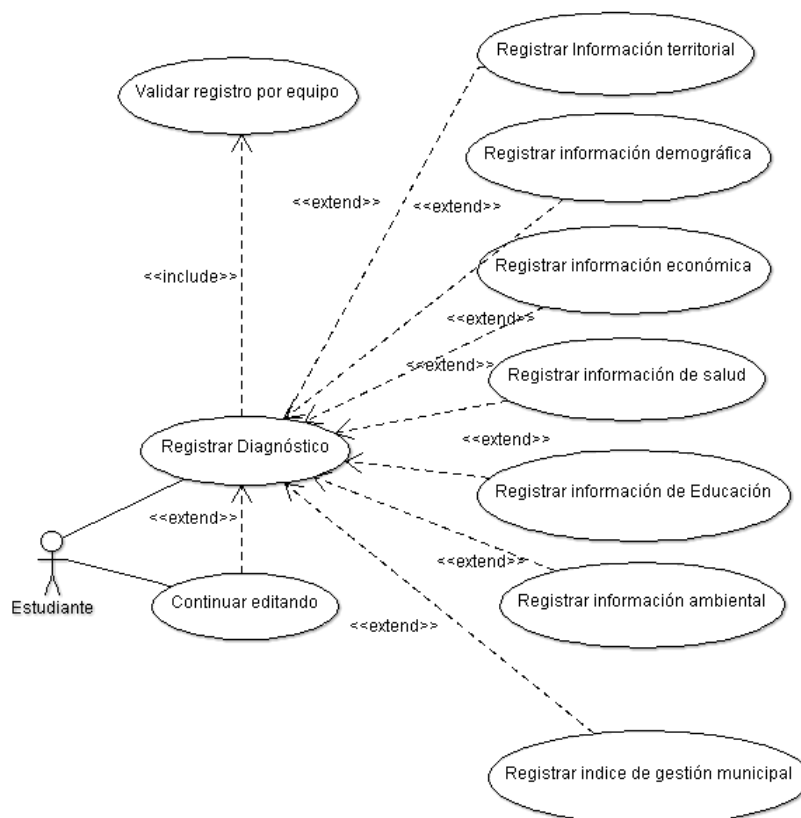


Fuente: elaboración propia.

La figura 10 muestra el marco general dentro del cual se desarrollará el módulo correspondiente a la Gestión de Diagnósticos y Proyectos de EPSUM y la interacción de los principales actores en el mismo.

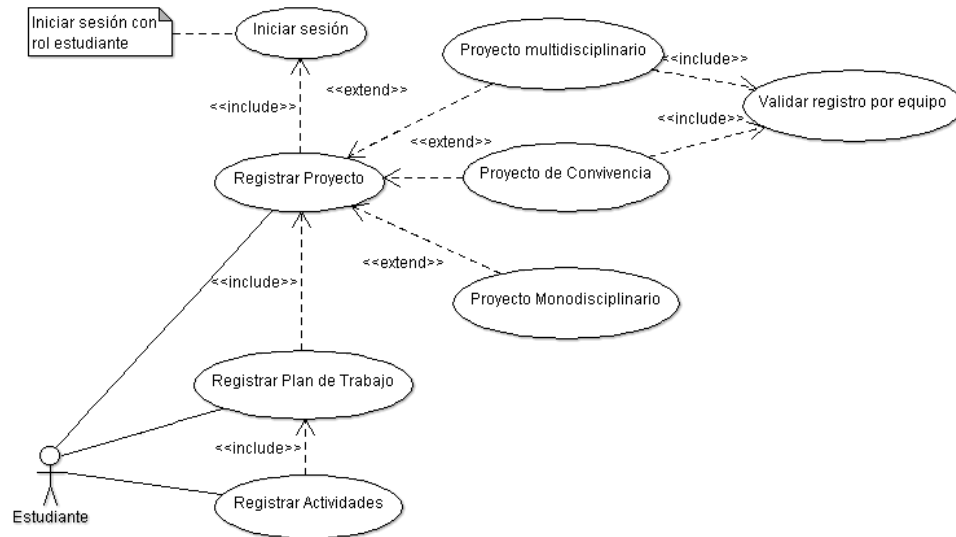
En la figura 11 se ilustran los requerimientos funcionales para el registro diagnóstico y en la figura 12 se muestran las acciones requeridas para el registro de proyectos, tales acciones son realizadas por el usuario estudiante.

Figura 11. **Diagrama de caso de uso: registrar diagnóstico**



Fuente: elaboración propia.

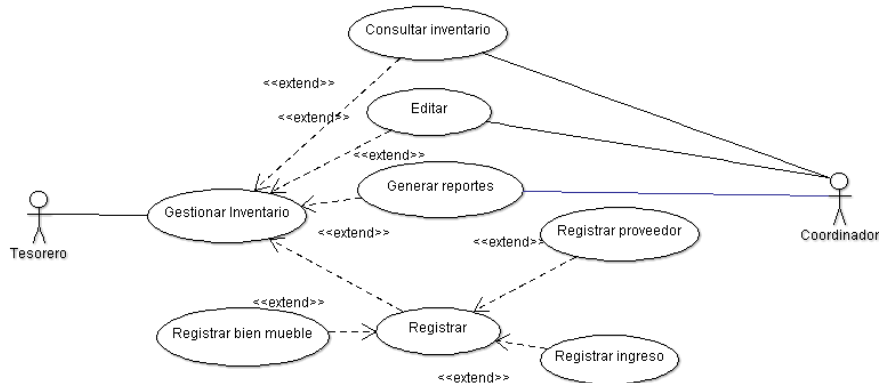
Figura 12. **Diagrama de caso de uso: registrar proyecto y plan de trabajo**



Fuente: elaboración propia.

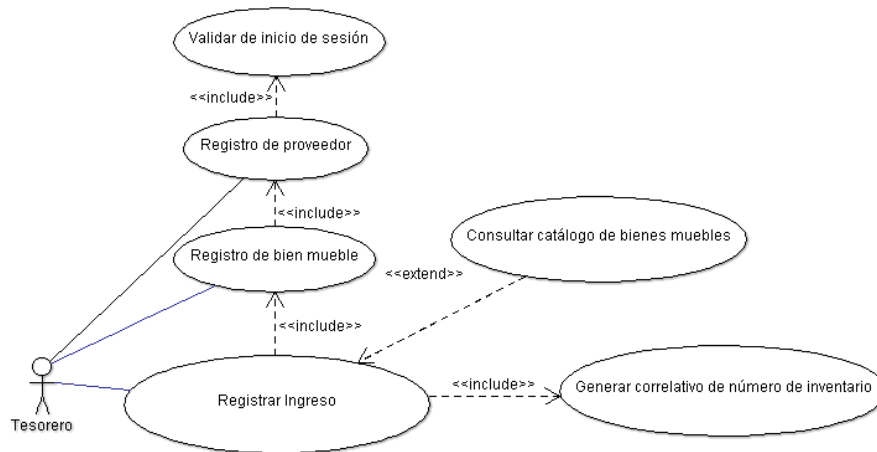
La figura 13 muestra al usuario Tesorero, quien será el encargado de registrar información administrativa del Programa, específicamente acerca del inventario de activos fijos, así también se muestra la interacción del coordinador con dicha información. En la figura 14 se ilustran los requerimientos para el registro de ingresos en el inventario.

Figura 13. **Caso de uso: gestionar información de inventario**



Fuente: elaboración propia.

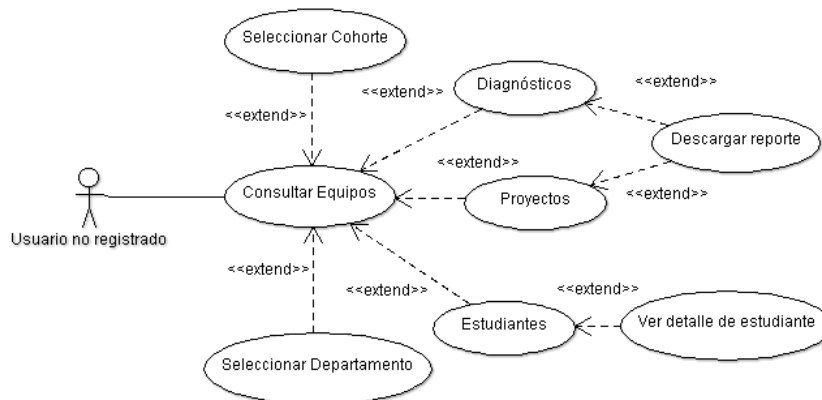
Figura 14. Diagrama de caso de uso: registrar ingreso en inventario



Fuente: elaboración propia.

Un usuario no registrado se define como la persona que accede al sistema para la consulta de información pública. En la figura 15 se presentan algunas acciones que el usuario invitado puede realizar.

Figura 15. Caso de uso: consulta de información pública

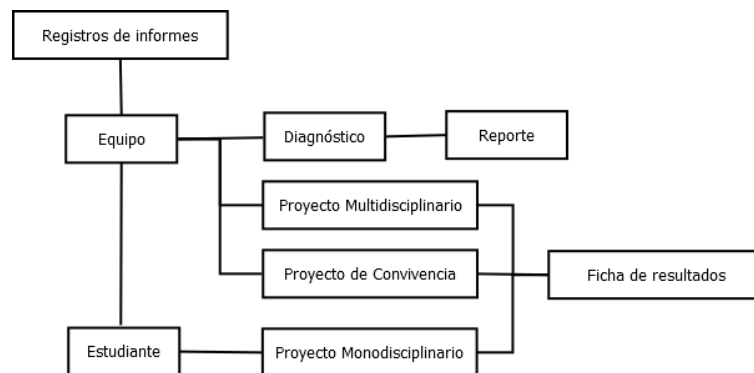


Fuente: elaboración propia.

2.8.2 Modelado de objetos de negocio

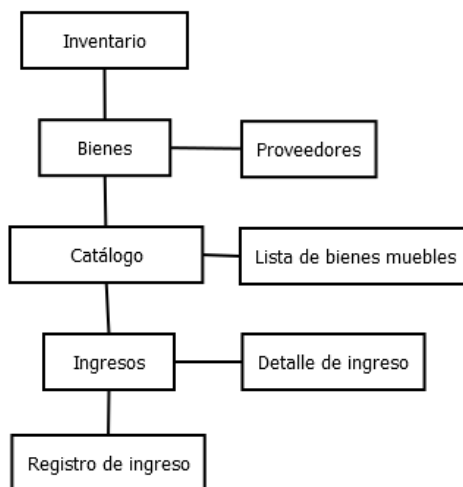
Este modelo está representado con un diagrama de modelo de dominio, asociados a los casos de uso del negocio. En la figura 16 se ilustra el modelo de dominio para el registro de informes y en la figura 17, el que corresponde al registro de inventario.

Figura 16. **Modelo de dominio de registro de informes**



Fuente: elaboración propia.

Figura 17. **Modelo de dominio de registro en inventario**



Fuente: elaboración propia.

2.9. Especificaciones de casos de uso

Siguiendo con el flujo de trabajo de la metodología de desarrollo, se realizaron las especificaciones de casos de uso como producto de la toma de requerimientos, las cuales son fundamentales para la fase de elaboración.

2.9.1. Registro de diagnóstico

Tabla 5. Especificación de caso de uso: registrar diagnóstico

Caso de Uso	Registrar diagnóstico.
Propósito	Ingresar la información del diagnóstico en el sistema para la consulta de la misma en el geoportal.
Actor	Estudiante
Resumen	El estudiante debe iniciar sesión en el sistema y se dirigirá al menú correspondiente para registrar la información de diagnóstico del proyecto para el equipo multidisciplinario. Este informe es por equipo, por tal razón, otro estudiante puede contribuir con la edición del mismo diagnóstico, pero no puede registrar un diagnóstico nuevo.
Pre-condición	El usuario debe de estar registrado como estudiante en el sistema y tener sesión iniciada.
Flujo Básico	1. El usuario con rol estudiante inicia sesión en la plataforma del programa desde un navegador. 2. Se validan los datos de inicio de sesión y se muestra la plataforma de estudiantes al usuario. 3. El estudiante se dirige al menú correspondiente y selecciona la opción de diagnóstico. 4. El sistema muestra la vista principal de diagnóstico y da la opción para registrar dicho informe. 5. El usuario procede a ingresar la información solicitada para completar el registro. 6. El usuario confirma los cambios y hace clic en la opción específica para guardar el registro. 7. El sistema muestra una pantalla con el resumen de la información ingresada.

Continuación de Tabla 5

Flujos Alternos	1. Como curso alternativo al inciso 4, el sistema verifica si existe un registro previo del diagnóstico para el equipo del usuario en la sesión, si lo hay, se muestra la pantalla del resumen del diagnóstico hasta el punto en que haya sido editado. 2. Para el inciso 5, como flujo alternativo, el usuario decide no completar el registro y continuar editando más tarde. En tal caso, se deberá dar clic en una opción específica para guardar los cambios hasta donde se haya ingresado la información. El usuario no cuenta con la información solicitada o no la tiene a la mano. Se deberá hacer clic en la opción correspondiente para seguir editando más tarde. El usuario da la oportunidad a otro compañero para seguir editando el informe. En tal escenario, se procederá también por medio de la opción correspondiente para seguir editando en otro momento y guardar la información ya ingresada.
Post-condición	Informe de diagnóstico registrado total o parcialmente en el sistema.
Requerimientos trazados	Registro de diagnóstico. Registro colaborativo entre el equipo. Almacenamiento parcial de la información a efecto del trabajo en equipo.
Notas	El sistema ya administra la sesión, por lo que su mención en el curso normal del caso de uso, es para indicar que el usuario debe iniciar sesión para registrar la información.

Fuente: elaboración propia.

2.9.2. Retomar edición de diagnóstico

Tabla 6. Especificación de caso de uso: continuar editando diagnóstico

Caso de Uso	Continuar editando diagnóstico.
Propósito	Retomar la edición por parte de un integrante del equipo para seguir editando, luego haber seleccionado una opción específica para guardar parcialmente la información.
Actor	Estudiante
Resumen	Un estudiante que ha procedido con el registro del diagnóstico, decide por algún motivo continuar editando más tarde o, en su defecto, darle el turno a otro integrante del equipo para seguir completando la información solicitada. Por tal razón, selecciona una opción específica para continuar editando más tarde, acto seguido, el sistema mostrará en pantalla el punto donde se dejó editando el informe.
Pre-condición	El usuario ha ingresado como estudiante al sistema y el mismo usuario u otro integrante del mismo equipo ya comenzó el registro del diagnóstico y guardó cambios parciales.
Flujo Básico	1. El estudiante ingresa a la plataforma y selecciona la opción para ir al diagnóstico. 2. El sistema muestra el resumen del diagnóstico hasta el último punto donde se guardaron cambios. 3. El usuario con rol de estudiante selecciona una acción específica para seguir editando el informe. 4. El sistema muestra la pantalla de registro y lo lleva hasta el último punto donde se registraron cambios. 5. El usuario continúa completando los campos con la información solicitada. 6. El usuario guarda los cambios, vuelve a seleccionar la opción específica para seguir editando en otro momento o cancela el registro de cambios. 7. El sistema muestra la pantalla de resumen del diagnóstico.

Continuación de Tabla 6

Flujos Alternos	1. Como curso alternativo al inciso 2, el sistema verifica si se ha completado el diagnóstico. Si ya se ha completado, entonces no se muestra la opción para continuar editando. 2. Para el inciso 6 del flujo normal, puede darse el caso de seleccionar por error la opción de seguir editando más tarde, sin haber guardado la información, en tal caso, se mantendrán los cambios hasta el último punto en que se seleccionó la opción específica para guardar los registros.
Post-condición	Informe de diagnóstico registrado total o parcialmente en el sistema.
Requerimientos trazados	Registro de diagnóstico. Registro colaborativo entre el equipo. Almacenamiento parcial de la información a efecto del trabajo en equipo.
Notas	La opción seguir editando en otro momento podrá utilizarse las veces que el equipo necesite para un auténtico trabajo en equipo.

Fuente: elaboración propia.

2.9.3. Actualizar información de diagnóstico

Tabla 7. Especificación de caso de uso: editar información de diagnóstico

Caso de Uso	Editar diagnóstico.
Propósito	Actualizar la información de un diagnóstico cuyo registro ha sido completado.
Actor	Estudiante
Resumen	El usuario con rol de estudiante requiere hacer modificaciones al informe diagnóstico previamente completado. Para ello se dirige a la opción para editar el mismo y procede a realizar la actualización de los registros.

Continuación de Tabla 7

Pre-condición	El informe diagnóstico a editar se ha registrado por completo.
Flujo Básico	1. El estudiante ingresa a la plataforma y selecciona la opción para ir al diagnóstico. 2. El sistema muestra el resumen del diagnóstico completado. 3. El usuario con rol de estudiante selecciona la opción para editar el informe. 4. El sistema muestra la vista respectiva para editar la información, cargando los registros ingresados previamente. 5. El usuario realiza los cambios deseados y selecciona la opción específica para actualizar los registros. 6. El sistema muestra la pantalla de resumen de diagnóstico.
Flujos Alternos	1. Como curso alternativo al flujo normal en el inciso 5, el usuario ya no desea realizar cambios, en tal caso, selecciona la opción correspondiente para cancelar el proceso.
Post-condición	Informe de diagnóstico modificado.
Requerimientos trazados	Actualización de diagnóstico.
Notas	La opción para editar el diagnóstico se mostrará únicamente al estar completo el registro de dicho informe.

Fuente: elaboración propia.

2.9.4. Registrar perfil de proyectos

Tabla 8. **Especificación de caso de uso: registrar proyecto multidisciplinario o de convivencia**

Caso de Uso	Registrar proyecto multidisciplinario o de convivencia.
Propósito	Almacenar el perfil de un proyecto en el sistema.

Continuación de Tabla 8

Actor	Estudiante
Resumen	El usuario con rol estudiante ingresa al sistema y se dispone a registrar un proyecto. Elige para el registro de proyecto multidisciplinario o de convivencia y se muestra en pantalla el formulario de registro respectivo.
Pre-condición	El usuario ha iniciado en sistema con rol de estudiante.
Flujo Básico	1. El usuario con rol estudiante inicia sesión en la plataforma del programa desde un navegador. 2. Se validan los datos de inicio de sesión y se muestra la plataforma de estudiantes al usuario. 3. El estudiante se dirige al menú correspondiente y selecciona la opción de proyecto. 4. El sistema muestra la vista principal del proyecto y da la opción para registrar el perfil. 5. El usuario procede a ingresar la información solicitada para completar el registro. 6. El usuario confirma los cambios y hace clic en la opción específica para guardar el registro. 7. El sistema muestra una pantalla con el resumen de la información ingresada.
Flujos Alternos	1. Como curso alternativo al inciso 4, el sistema verifica si existe un registro previo de proyecto para el equipo del usuario en la sesión, si lo hay, se muestra la pantalla del resumen del proyecto.
Post-condición	Perfil de proyecto registrado.
Requerimientos trazados	Registro de perfil de proyecto multidisciplinario y de convivencia de acuerdo a la metodología del Programa.
Notas	Los perfiles de proyecto de tipo multidisciplinario y de convivencia corresponde a un registro de perfil por equipo.

Fuente: elaboración propia.

Tabla 9. **Especificación de caso de uso: registrar proyecto monodisciplinario**

Caso de Uso	Registrar proyecto monodisciplinario.
Propósito	Almacenar el perfil de un proyecto monodisciplinario en el sistema.
Actor	Estudiante
Resumen	El usuario con rol estudiante ingresa al sistema y se dispone a registrar un proyecto. Elige para el registro de proyecto monodisciplinario y se muestra en pantalla el formulario de registro respectivo.
Pre-condición	El usuario ha iniciado en el sistema con rol de estudiante.
Flujo Básico	1. El usuario con rol estudiante inicia sesión en la plataforma del programa desde un navegador. 2. Se validan los datos de inicio de sesión y se muestra la plataforma de estudiantes al usuario. 3. El estudiante se dirige al menú correspondiente y selecciona la opción de proyecto. 4. El sistema muestra la vista principal del proyecto y da la opción para registrar el perfil. 5. El usuario procede a ingresar la información solicitada para completar el registro. 6. El usuario confirma los cambios y hace clic en la opción específica para guardar el registro. 7. El sistema muestra una pantalla con el resumen de la información ingresada.
Flujos Alternos	1. Como curso alternativo al inciso 4, el sistema verifica si existe un registro previo de proyecto, si lo hay, se muestra la pantalla del resumen del proyecto.
Post-condición	Perfil de proyecto monodisciplinario registrado.
Requerimientos trazados	Registro de perfil de proyecto monodisciplinario de acuerdo a la metodología del Programa.
Notas	El registro de perfil de proyecto monodisciplinario es individual y corresponde a un registro por cada integrante del equipo.

Fuente: elaboración propia.

2.9.5. Actualizar información de proyectos

Tabla 10. Especificación de caso de uso: editar información de proyecto

Caso de Uso	Editar información de proyecto.
Propósito	Actualizar la información de un proyecto de cualquier tipo registrado previamente en el sistema.
Actor	Estudiante
Resumen	El estudiante requiere editar la información de un proyecto registrado previamente. Para ello accede al sistema y luego elige el proyecto. En la vista del proyecto selecciona la opción para editar y podrá actualizar la información.
Pre-condición	Existe un registro previo del proyecto que se desea editar.
Flujo Básico	1. El estudiante ingresa a la plataforma y selecciona la opción para ir al tipo de proyecto deseado. 2. El sistema muestra el resumen del proyecto completado. 3. El usuario con rol de estudiante selecciona la opción para editar el proyecto. 4. El sistema muestra la vista respectiva para editar la información, cargando los registros ingresados previamente. 5. El usuario realiza los cambios deseados y selecciona la opción específica para actualizar los registros. 6. El sistema muestra la pantalla de resumen de diagnóstico.
Flujos Alternos	1. Como curso alternativo al flujo normal en el inciso 5, el usuario ya no desea realizar cambios, en tal caso, selecciona la opción correspondiente para cancelar el proceso.
Post-condición	Perfil de proyecto actualizado.
Requerimientos trazados	Actualización de información del perfil del proyecto.
Notas	La opción para editar el proyecto se mostrará únicamente al estar completo el registro de dicho informe.

Fuente: elaboración propia.

2.9.6. Registro de plan de trabajo

Tabla 11. **Especificación de caso de uso: registrar objetivos específicos, metas e indicadores**

Caso de Uso	Registrar objetivos específicos, metas e indicadores de un proyecto.
Propósito	Registrar los objetivos específicos, metas e indicadores de un proyecto en el sistema.
Actor	Estudiante
Resumen	El usuario con rol de estudiante ha registrado un proyecto y se dispone a registrar el plan de trabajo para el mismo. Para ello, ingresa uno de los objetivos específicos del proyecto, la meta para el objetivo propuesto y su respectivo indicador, posteriormente selecciona la opción para guardar la información.
Pre-condición	Existe un registro previo del proyecto para el cual se registrará el plan de trabajo.
Flujo Básico	1. Se ha registrado un proyecto y se da la opción al estudiante para registrar el plan de trabajo. El estudiante selecciona la opción para crear el plan de trabajo y el sistema muestra el formulario respectivo. 2. El estudiante ingresa uno de los objetivos específicos para el proyecto. 3. El estudiante ingresa la meta para el objetivo propuesto. 4. Por último, se ingresa el indicador para el mismo objetivo, diferenciando una cantidad para el indicador. 5. El usuario da clic en la opción para guardar la información. 6. El sistema valida la información ingresada y muestra un aviso al estudiante como notificación de registro exitoso.
Flujos Alternos	1. Como curso alternativo al flujo normal en el inciso 6, el sistema encuentra alguna inconsistencia en los datos o falta información, en tal caso, no se permite continuar con el proceso.
Post-condición	Objetivos, metas e indicadores para un proyecto, registrados en el sistema.

Continuación de Tabla 11

Requerimientos trazados	Registro de Objetivos, metas e indicadores de un proyecto.
Notas	Esta opción puede ser continua al registro del perfil del proyecto.

Fuente: elaboración propia.

Tabla 12. **Especificación de caso de uso: registrar actividades del proyecto**

Caso de Uso	Registrar actividades para un proyecto.
Propósito	Almacenar las actividades para los proyectos en el sistema con el fin de llevar un mejor control.
Actor	Estudiante
Resumen	El usuario con rol de estudiante ha registrado un proyecto y los objetivos, metas e indicadores para el mismo. Se dispone a registrar las actividades que están asociadas al objetivo ingresado y completa con la descripción de la actividad y las fechas específicas, posteriormente selecciona la opción para guardar la información.
Pre-condición	Se ha registrado un objetivo específico, meta e indicador con anterioridad.
Flujo Básico	1. Se ha registrado un objetivo específico y su respectiva meta e indicador, por lo que se da la opción al estudiante para registrar las actividades. 2. El sistema muestra la vista respectiva para el ingreso de la información. 3. El estudiante ingresa la descripción de la actividad y definiendo sus fechas. 4. El usuario da clic en la opción para guardar la información en el sistema. 5. El sistema muestra el resumen del proyecto y el avance porcentual del mismo de acuerdo a las actividades registradas.

Continuación de Tabla 12

Flujos Alternos	1. De manera alterna al flujo normal en el inciso 4, el sistema encuentra alguna inconsistencia en las fechas indicadas o falta información, en tal caso, no se permite continuar con el proceso y se notifica al usuario.
Post-condición	Registro de actividades para un plan de trabajo en un proyecto.
Requerimientos trazados	Registro de actividades de un proyecto en el sistema.
Notas	Pueden registrarse varias actividades para un objetivo y por consiguiente, para el proyecto.

Fuente: elaboración propia.

Tabla 13. **Especificación de caso de uso: Agregar un objetivo específico al plan de trabajo de un proyecto existente**

Caso de Uso	Agregar un objetivo específico al plan de trabajo de un proyecto existente.
Propósito	Registrar un objetivo nuevo al plan de trabajo de un proyecto.
Actor	Estudiante
Resumen	Un estudiante requiere agregar un objetivo no contemplado al inicio del proyecto, por lo que desea registrarlo en el sistema. Ingresa a la plataforma, selecciona el proyecto y luego elige la opción específica para agregar un objetivo nuevo. Se dispone a completar la información solicitada para el objetivo y guarda los cambios.
Pre-condición	Existe un plan de trabajo creado para el proyecto seleccionado.
Flujo Básico	1. El estudiante ingresa al sistema y se dirige al apartado de proyectos en donde selecciona el proyecto para el cual agregará un nuevo objetivo. 2. El sistema carga el resumen del proyecto. 3. El estudiante selecciona la opción específica para agregar un nuevo objetivo.

Continuación de Tabla 13

	4. El sistema muestra la vista correspondiente para agregar el objetivo. 5. El usuario ingresa la información solicitada y selecciona la opción correspondiente para registrar los cambios. 6. El sistema procesa la solicitud y notifica al usuario del estado del registro.
Flujos Alternos	1. Como curso alternativo al flujo normal en el inciso 6, el sistema encuentra alguna inconsistencia en los datos o falta información, en tal caso, no se permite continuar con el proceso y se notifica al usuario.
Post-condición	Agregación de objetivos específicos al plan de trabajo de un proyecto en el sistema.
Requerimientos trazados	Agregar objetivos nuevos a un plan de trabajo existente.

Fuente: elaboración propia.

Tabla 14. **Especificación de caso de uso: Agregar actividades al plan de trabajo de un proyecto existente**

Caso de Uso	Agregar actividades al plan de trabajo de un proyecto existente.
Propósito	Registrar actividades no contempladas al momento de registrar el plan de trabajo.
Actor	Estudiante
Resumen	Un estudiante requiere agregar actividades no contempladas al inicio del proyecto, por lo que desea registrarlas en el sistema. Accede a la plataforma, selecciona el proyecto y luego elige la opción específica para agregar actividades nuevas. Se dispone a completar la información solicitada para la actividad y guarda los cambios.
Pre-condición	Existe un plan de trabajo creado para el proyecto seleccionado.

Continuación de Tabla 14

Flujo Básico	1. El estudiante ingresa al sistema y se dirige al apartado de proyectos en donde selecciona el proyecto para el cual agregará nueva actividad. 2. El sistema carga el resumen del proyecto. 3. El estudiante selecciona la opción específica para agregar una actividad. 4. El sistema muestra la vista correspondiente para agregar el objetivo. 5. El usuario selecciona el objetivo para el cual registrará la actividad. 6. El usuario ingresa la información solicitada y selecciona la opción correspondiente para registrar los cambios. 7. El sistema procesa la solicitud, notifica al usuario del estado del registro y actualiza el porcentaje de avance del proyecto.
Flujos Alternos	1. Como curso alternativo al flujo normal en el inciso 7, el sistema encuentra alguna inconsistencia en los datos o falta información, en tal caso, no se permite continuar con el proceso y se notifica al usuario.
Post-condición	Agregación de actividades al plan de trabajo de un proyecto en el sistema.
Requerimientos trazados	Agregar actividades nuevas a un plan de trabajo existente.

Fuente: elaboración propia.

2.9.7. Actualización de información en el plan de trabajo

Tabla 15. **Especificación de caso de uso: editar objetivo específicos, metas e indicadores**

Caso de Uso	Editar objetivos específicos, metas e indicadores de un proyecto.
Propósito	Actualizar información de objetivos específicos, metas e indicadores de un proyecto en el sistema.
Actor	Estudiante
Resumen	Un estudiante requiere editar alguno de los objetivos, metas o indicadores. Para ello, se dirige al proyecto del cual desea modificar los datos y se dirige a la opción específica para editar los objetivos específicos. Realiza los cambios necesarios y guarda el registro mediante la opción correspondiente.
Pre-condición	Existe un registro previo de la información a modificar.
Flujo Básico	1. El usuario ingresa al sistema como estudiante y se dispone a actualizar la información de objetivos o metas o indicadores. 2. El usuario se dirige al apartado de proyectos y selecciona el informe deseado. 3. El sistema carga el resumen del proyecto y toda la información registrada para el mismo. 4. El usuario se dirige al apartado de objetivos y selecciona la opción editar. 5. El usuario realiza los cambios requeridos y los guarda mediante una opción específica. 6. El sistema realiza la actualización y notifica al usuario del estado del proceso.
Flujos Alternos	1. Como curso alternativo al flujo normal en el inciso 6, el sistema encuentra alguna inconsistencia en los datos o falta información, en tal caso, no se permite continuar con el proceso y se notifica al usuario.
Post-condición	Objetivos, metas e indicadores para un proyecto, actualizados en el sistema.

Continuación de Tabla 15

Requerimientos trazados	Actualización de registros de objetivos, metas e indicadores de un proyecto.
Notas	Debe existir un registro de cambios para garantizar la responsabilidad de los estudiantes en los objetivos propuestos en cada uno de los proyectos, de tal manera que los supervisores y el mismo equipo pueda verificar el responsable de algún cambio no justificado.

Fuente: elaboración propia.

Tabla 16. **Especificación de caso de uso: editar actividades**

Caso de Uso	Editar actividades de un proyecto.
Propósito	Actualizar información de actividades de un proyecto en el sistema.
Actor	Estudiante
Resumen	Un estudiante requiere editar alguna de las actividades. Para ello, se dirige al proyecto del cual desea modificar los datos y se dirige a la opción específica para editar actividades. Realiza los cambios necesarios y guarda el registro mediante la opción correspondiente.
Pre-condición	Existe un registro previo de la información a modificar.
Flujo Básico	1. El usuario ingresa al sistema como estudiante y se dispone a actualizar la información de actividades. 2. El usuario se dirige al apartado de proyectos y selecciona el informe deseado. 3. El sistema carga el resumen del proyecto y toda la información registrada para el mismo. 4. El usuario se dirige al apartado de actividades y selecciona la opción editar. 5. El usuario realiza los cambios requeridos y los guarda mediante una opción específica. 6. El sistema realiza la actualización y notifica al usuario del estado del proceso.

Continuación de Tabla 16

Flujos Alternos	1. Como curso alternativo al flujo normal en el inciso 6, el sistema encuentra alguna inconsistencia en los datos o falta información, en tal caso, no se permite continuar con el proceso y se notifica al usuario.
Post-condición	Actividades asociadas a objetivos de un proyecto, actualizados en el sistema.
Requerimientos trazados	Actualización de registros de actividades asociadas a objetivos de un proyecto.
Notas	Debe existir un registro de cambios para garantizar la responsabilidad de los estudiantes en las actividades propuestas en cada uno de los proyectos, de tal manera que los supervisores y el mismo equipo pueda verificar el responsable de algún cambio no justificado.

Fuente: elaboración propia.

2.9.8. Finalización de proyectos

Tabla 17. **Especificación de caso de uso: Marcar una actividad como hecha**

Caso de Uso	Marcar actividades como hechas.
Propósito	Marcar las actividades de proyecto como hechas para indicar que han sido realizadas y actualizar el progreso del proyecto.
Actor	Estudiante
Resumen	Un estudiante requiere indicar que una actividad ha sido realizada, por lo que ingresa al sistema y se dirige al apartado de proyectos. Selecciona un proyecto y desde ahí se dirige a la vista de actividades. Selecciona la opción específica para marcar como hecha la actividad y el sistema procesa la acción para luego actualizar el porcentaje de avance del proyecto.

Continuación de Tabla 17

Pre-condición	Se han registrado actividades para el proyecto
Flujo Básico	1. El usuario ingresa al sistema como estudiante y se dispone a marcar actividades como hechas. 2. El usuario se dirige al apartado de proyectos y selecciona el informe deseado. 3. El sistema carga el resumen del proyecto y toda la información registrada para el mismo. 4. El usuario se dirige al apartado de actividades y selecciona la opción específica para marcar las actividades como hechas. 5. El sistema realiza la acción solicitada y actualiza el porcentaje de avance del proyecto.
Flujos Alternos	-
Post-condición	Actividades marcadas como terminadas.
Requerimientos trazados	Marcado de actividades como hechas para indicar avance del proyecto.
Notas	Debe existir un registro de cambios para garantizar la responsabilidad de los estudiantes en las actividades propuestas en cada uno de los proyectos, de tal manera que los supervisores y el mismo equipo pueda verificar el responsable de algún cambio no justificado.

Fuente: elaboración propia.

2.9.9. Registro de ficha de resultados

Tabla 18. **Especificación de caso de uso: registrar ficha de resultados de proyectos**

Caso de Uso	Registrar ficha de resultados de proyectos.
Propósito	Registrar la ficha que resume la información del proyecto para su consulta.
Actor	Estudiante

Continuación de Tabla 18

Resumen	Un estudiante se dispone a registrar la ficha de resultados de un proyecto. Para ello, ingresa al sistema y se dirige al apartado para ficha de resultados que se genera por proyecto. El sistema verifica que el proyecto se ha finalizado y el usuario procede a completar la información necesaria para dicho informe.
Pre-condición	El proyecto ha sido finalizado.
Flujo Básico	1. El usuario ingresa al sistema como estudiante y se dispone a registrar la ficha final del proyecto. 2. El usuario se dirige al apartado de ficha de resultados y selecciona la opción para la ficha del proyecto deseado. 3. El sistema verifica si el proyecto ya fue finalizado y muestra el formulario correspondiente para completar información en la ficha. 4. El estudiante completa la información solicitada y guarda los cambios. 5. El sistema almacena la información y muestra un resumen del proyecto a modo de ficha final del proyecto.
Flujos Alternos	1. Como curso alternativo al flujo normal en el inciso 3, el sistema detecta que el proyecto no ha sido finalizado, por lo que notifica al usuario y lo redirige al detalle del proyecto. 2. Para el inciso 4, el sistema encuentra alguna inconsistencia en los datos o falta información, en tal caso, no se permite continuar con el proceso y se notifica al usuario.
Post-condición	Registro de ficha final en el sistema como resumen del proyecto.
Requerimientos trazados	Registro de fichas finales de proyecto como resumen de proyectos de acuerdo a las fichas solicitadas en el informe final.
Notas	Las fichas de resultados se registran por proyecto y equipo, a excepción de la del proyecto monodisciplinario, que es individual.

Fuente: elaboración propia.

2.9.10. Consulta de información de diagnósticos y proyectos

Tabla 19. Especificación de caso de uso: consulta de diagnóstico

Caso de Uso	Consultar diagnósticos.
Propósito	Obtener los registros de diagnósticos almacenados en el sistema.
Actor	Estudiante, Coordinador, Supervisor, Usuario General.
Resumen	Un usuario se dispone a consultar un diagnóstico de un determinado equipo. La consulta puede ser realizada desde una vista pública o como usuario registrado en el sistema. En el caso de ser un usuario no registrado, podrá acceder a la información a través de un módulo tipo geoportal desde el cual podrá seleccionar entre la cohorte deseada, el equipo y el territorio requerido donde se realizó el diagnóstico. Si es el caso de un usuario registrado, como un estudiante o supervisor, podrá consultarlo desde el apartado correspondiente para diagnósticos.
Pre-condición	Existen registros de diagnósticos en el sistema.
Flujo Básico	1. Un usuario necesita consultar un diagnóstico en específico. 2. El usuario ingresa al apartado para consulta de información que estará definido como un geoportal. 3. El sistema cargará la vista con la información correspondiente. 4. El usuario selecciona entre la cohorte, el lugar y/o equipo del cual desea consultar el diagnóstico. 5. Se carga la información para el territorio seleccionado y se elige la opción de diagnóstico. 6. El sistema carga la información del diagnóstico y la muestra al usuario.
Flujos Alternos	1. Para el inciso 2, como flujo alternativo, puede ocurrir que el usuario está registrado en el sistema, por lo que se dirige al inicio de sesión y luego a la opción correspondiente para consulta de diagnóstico. 2. De manera alternativa para el inciso 6, el sistema no encuentra coincidencias para el diagnóstico solicitado, por lo que notifica al usuario a través de un mensaje.

Continuación de Tabla 19

Post-condición	Resultados para la consulta de diagnóstico.
Requerimientos trazados	Consulta pública y privada de informes diagnósticos según la información registrada por los estudiantes.
Notas	La consulta de diagnósticos para supervisor mostrará solo los registros disponibles de los equipos con los que trabaja.

Fuente: elaboración propia.

Tabla 20. **Especificación de caso de uso: consulta de proyectos**

Caso de Uso	Consultar proyectos.
Propósito	Obtener los registros de proyectos almacenados en el sistema.
Actor	Estudiante, Coordinador, Supervisor, Usuario General.
Resumen	Un usuario necesita consultar un proyecto de un determinado equipo. La consulta puede ser realizada desde una vista pública o como usuario registrado en el sistema. En el caso de ser un usuario no registrado, podrá acceder a la información a través de un geoportal desde el cual podrá seleccionar entre la cohorte deseada, el equipo y el territorio requerido donde se realizó el proyecto. Si se trata de un usuario registrado, como un estudiante o supervisor, podrá consultarlo desde el apartado correspondiente para proyectos.
Pre-condición	Existen registros de proyectos en el sistema.
Flujo Básico	1. El usuario ingresa al apartado para consulta de información que estará definido como un geoportal. 2. El sistema cargará la vista con la información correspondiente. 3. El usuario selecciona entre la cohorte, el lugar y/o equipo del cual desea consultar un proyecto. 4. Se carga la información para el territorio seleccionado y se elige la opción de proyectos. 5. El sistema carga la información del proyecto y la muestra al usuario.

Continuación de Tabla 20

Flujos Alternos	1. Para el inciso 1, como flujo alternativo, puede ocurrir que el usuario esté registrado en el sistema, por lo que se dirige al inicio de sesión y luego a la opción correspondiente para consulta de proyecto. 2. De manera alterna para el inciso 5, el sistema no encuentra coincidencias para el proyecto solicitado, por lo que notifica al usuario a través de un mensaje.
Post-condición	Resultados para la consulta de proyectos.
Requerimientos trazados	Consulta pública y privada de proyectos, según la información registrada por los estudiantes.
Notas	La consulta de proyectos para supervisor mostrará solo los registros disponibles de los equipos con los que trabaja.

Fuente: elaboración propia.

2.9.11. Generación de reportes de diagnósticos y/o proyectos

Tabla 21. **Especificación de caso de uso: generar reportes de diagnósticos y/o proyectos**

Caso de Uso	Generar reporte de informe diagnóstico y/o proyecto.
Propósito	Generar un documento en formato PDF que detalle el informe diagnóstico y/o proyectos.
Actor	Estudiante, Coordinador, Supervisor, Usuario General.
Resumen	Un usuario requiere un reporte de un diagnóstico o proyecto de un determinado equipo. Para ello, realiza la consulta correspondiente desde una vista pública o como usuario registrado en el sistema. En el caso de ser un usuario no registrado, podrá acceder a la información a través de un geoportal desde el cual podrá descargar el informe luego de hacer una consulta. Si se trata de un usuario registrado, como un estudiante o supervisor, podrá generarlo desde el apartado correspondiente y luego de realizar la búsqueda.

Continuación de Tabla 21

Pre-condición	El diagnóstico o el proyecto han sido registrados completamente en el sistema.
Flujo Básico	1. Un usuario requiere un reporte de un diagnóstico o proyecto en específico. 2. El usuario ingresa al apartado para consulta de información que estará definido como un geoportal. 3. El sistema cargará la vista con la información correspondiente. 4. El usuario selecciona entre la cohorte, el lugar y/o equipo del cual necesita el reporte de informe diagnóstico o proyecto. 5. Se carga la información para el territorio seleccionado y se elige la opción para descargar el reporte. 6. El sistema atiende la petición, prepara el informe y lo muestra al usuario.
Flujos Alternos	1. Para el inciso 2, como flujo alternativo, puede ocurrir que el usuario esté registrado en el sistema, por lo que se dirige al inicio de sesión y luego a la opción correspondiente para consultar diagnósticos o proyectos, donde encontrará la opción de descarga de dichos informes.
Post-condición	Reporte generado y puesto a disposición del usuario.
Requerimientos trazados	Generación de reportes para los informes de diagnóstico y proyectos registrados en el sistema.
Notas	Puesto que la información registrada en el diagnóstico representa una investigación realizada por los estudiantes y considerando que dicha información será pública al momento del registro, se deberá utilizar una licencia que atribuya el trabajo realizado por el equipo en el territorio asignado, estableciendo los permisos suficientes para el uso de la información contenida en tal informe.

Fuente: elaboración propia.

2.9.12. Gestión de inventario de activos fijos

2.9.12.1. Registro de proveedores

Tabla 22. Especificación de caso de uso: registrar proveedores

Caso de Uso	Registrar proveedor para inventario.
Propósito	Registro de proveedores para la gestión de información de inventario de activos fijos.
Actor	Tesorero, Coordinador.
Resumen	El usuario con rol de tesorería ingresa al sistema y requiere registrar un nuevo proveedor. Se dirige al módulo de inventario y luego selecciona el apartado de proveedores. Desde ahí selecciona la opción para registrar un nuevo proveedor y el sistema muestra el formulario respectivo. El usuario completa la información y selecciona la opción para registrar los cambios.
Pre-condición	Sesión iniciada con usuario de rol tesorero o coordinador.
Flujo Básico	1. El usuario ingresa al sistema con rol de tesorero o coordinador. 2. Se dispone a registrar un nuevo proveedor y se dirige al apartado para el efecto. 3. El sistema le muestra la vista general de proveedores y da la opción al usuario para registrar un proveedor. 4. El usuario selecciona la opción para registrar. 5. Se muestra el formulario respectivo para la creación de un nuevo proveedor. 6. El usuario completa los campos con la información solicitada y selecciona la opción para registrar los cambios. 7. El sistema procesa la solicitud y notifica al usuario del estado del registro. 8. El sistema limpia los campos para dar la opción al usuario de registrar otro proveedor y muestra otra opción para terminar con el registro de proveedores.

Continuación de Tabla 22

Flujos Alternos	1. Como flujo alternativo al inciso 7, puede ocurrir que el usuario no haya completado todos los campos o existan inconsistencias en la información ingresada, si es el caso, el sistema notificará al usuario y no procederá con el registro.
Post-condición	Proveedor registrado en el sistema.
Requerimientos trazados	Registro de proveedores para la gestión de información de inventario de activos fijos.
Notas	-

Fuente: elaboración propia.

2.9.12.2. Actualización de información de proveedores

Tabla 23. **Especificación de caso de uso: editar proveedor**

Caso de Uso	Editar proveedor.
Propósito	Actualización de registro de proveedores para la gestión de información de inventario de activos fijos.
Actor	Tesorero, Coordinador.
Resumen	El usuario con rol de tesorería ingresa al sistema y requiere editar la información de un proveedor. Se dirige al módulo de inventario y luego selecciona el apartado de proveedores. Desde ahí selecciona la opción para editar un proveedor y el sistema carga la información del proveedor indicado. El usuario realiza los cambios deseados y selecciona la opción para registrarlos.
Pre-condición	Sesión iniciada con usuario de rol tesorero o coordinador. Existe un registro previo de un proveedor.

Continuación de Tabla 23

Flujo Básico	1. El usuario ingresa al sistema con rol de tesorero o coordinador y se dispone a editar la información de un proveedor, por lo que se dirige al apartado de proveedores. 2. El sistema le muestra la vista general de proveedores y da la opción al usuario para editar. 3. El usuario selecciona la opción para editar. 4. Se muestra el formulario respectivo para la edición de proveedor y se cargan los datos del mismo. 5. El usuario realiza los cambios necesarios y selecciona la opción para guardarlos. 6. El sistema procesa la solicitud y notifica al usuario del estado del registro.
Flujos Alternos	1. Como flujo alternativo al inciso 6, puede ocurrir que el usuario no haya completado todos los campos o existan inconsistencias en la información ingresada, si es el caso, el sistema notificará al usuario y no procederá con el registro.
Post-condición	Actualización de información de un proveedor en el sistema.
Requerimientos	Actualización de información de proveedores.

Fuente: elaboración propia.

2.9.12.3. Baja de proveedores

Tabla 24. Especificación de caso de uso: dar de baja a proveedores

Caso de Uso	Baja de proveedor.
Propósito	Dar de baja a un proveedor registrado en el sistema.
Actor	Tesorero, Coordinador.
Resumen	El usuario con rol de tesorería necesita dar de baja a un proveedor. Para ellos, se dirige al módulo de inventario y luego selecciona el apartado de proveedores. Desde ahí selecciona la opción para dar de baja a un proveedor y el sistema procesa la solicitud y notifica al usuario.

Continuación de Tabla 24

Pre-condición	Sesión iniciada con usuario de rol tesorero o coordinador. Existe un registro previo de un proveedor.
Flujo Básico	1. El usuario ingresa al sistema con rol de tesorero o coordinador y se dispone a dar de baja a un proveedor, por lo que se dirige al apartado respectivo. 2. El sistema le muestra la vista general de proveedores y le da la opción al usuario para dar de baja a los proveedores listados. 3. El usuario selecciona la opción para dar de baja a un proveedor en específico. 4. El sistema procesa la solicitud y notifica al usuario del estado del registro.
Post-condición	Baja de un proveedor en el sistema.
Requerimientos trazados	Baja de proveedores registrados.

Fuente: elaboración propia.

2.9.12.4. Registro de activos fijos

Tabla 25. **Especificación de caso de uso: registrar bien mueble**

Caso de Uso	Registrar bienes muebles.
Propósito	Registro de bienes muebles para la gestión de información de inventario en el sistema.
Actor	Tesorero, Coordinador.
Resumen	El usuario con rol de tesorería ingresa al sistema y requiere registrar un nuevo activo. Se dirige al módulo de inventario y luego selecciona el apartado de activos/bienes muebles. Desde ahí selecciona la opción para registrar un nuevo bien y el sistema muestra el formulario respectivo. El usuario completa la información y selecciona la opción para registrar los cambios.
Pre-condición	Sesión iniciada con usuario de rol tesorero o coordinador.

Continuación de Tabla 25

Flujo Básico	1. El usuario ingresa al sistema con rol de tesorero o coordinador. 2. Se dispone a registrar un nuevo bien y se dirige al apartado para el efecto. 3. El sistema le muestra la vista general de activos y da la opción al usuario para registrar un bien. 4. El usuario selecciona la opción para registrar. 5. Se muestra el formulario respectivo para la creación de un nuevo activo. 6. El usuario completa los campos con la información solicitada y selecciona la opción para registrar los cambios. 7. El sistema procesa la solicitud y notifica al usuario del estado del registro. 8. El sistema limpia los campos para dar la opción al usuario de registrar otro activo y muestra otra opción para terminar con el registro de bienes.
Flujos Alternos	1. Como flujo alternativo al inciso 7, puede ocurrir que el usuario no haya completado todos los campos o existan inconsistencias en la información ingresada, si es el caso, el sistema notificará al usuario y no procederá con el registro.
Post-condición	Bien mueble registrado en el sistema.
Requerimientos trazados	Registro de bienes muebles para la gestión de información de inventario de activos fijos.

Fuente: elaboración propia.

2.9.12.5. Actualización de información de bienes

Tabla 26. **Especificación de caso de uso: editar bienes**

Caso de Uso	Editar bienes.
Propósito	Actualización de registro de bienes para la gestión de información de inventario de activos fijos.
Actor	Tesorero, Coordinador.

Continuación de Tabla 26

Resumen	El usuario con rol de tesorería ingresa al sistema y requiere editar la información de un bien. Se dirige al módulo de inventario y luego selecciona el apartado de bienes muebles. Desde ahí selecciona la opción para editar un bien y el sistema carga la información del bien seleccionado. El usuario realiza los cambios deseados y selecciona la opción para registrarlos.
Pre-condición	Sesión iniciada con usuario de rol tesorero o coordinador. Existe un registro previo de un bien mueble.
Flujo Básico	1. El usuario ingresa al sistema con rol de tesorero o coordinador y se dispone a editar la información de un bien, por lo que se dirige al apartado de bienes. 2. El sistema le muestra la vista general de bienes y da la opción al usuario para editar. 3. El usuario selecciona la opción para editar. 4. Se muestra el formulario respectivo para la edición del bien seleccionado y se cargan los datos del mismo. 5. El usuario realiza los cambios necesarios y selecciona la opción para guardarlos. 6. El sistema procesa la solicitud y notifica al usuario del estado del registro.
Flujos Alternos	2. Como flujo alternativo al inciso 6, puede ocurrir que el usuario no haya completado todos los campos o existan inconsistencias en la información ingresada, si es el caso, el sistema notificará al usuario y no procederá con el registro.
Post-condición	Información actualizada de un bien registrado en el sistema.
Requerimientos trazados	Actualización de información de bienes.
Notas	-

Fuente: elaboración propia.

2.9.12.6. Registro de compras

Tabla 27. **Especificación de caso de uso: registrar compras**

Caso de Uso	Registrar compra.
Propósito	Registro compras para la gestión de información de inventario en el sistema.
Actor	Tesorero, Coordinador.
Resumen	El usuario con rol de tesorería ingresa al sistema y se dispone a registrar una compra. Se dirige al módulo de inventario y luego selecciona el apartado de compras. Desde ahí selecciona la opción correspondiente y el sistema muestra el formulario para el registro de compras. El usuario completa la información y selecciona la opción para registrar los cambios.
Pre-condición	Sesión iniciada con usuario de rol tesorero o coordinador.
Flujo Básico	1. El usuario ingresa al sistema con rol de tesorero o coordinador. 2. Se dispone a registrar una compra y se dirige al apartado para el efecto. 3. El sistema le muestra la vista general de compras y da la opción al usuario para registrar una nueva. 4. El usuario selecciona la opción para registrar. 5. Se muestra el formulario respectivo para el registro de compras. 6. El usuario completa los campos con la información solicitada y selecciona la opción para guardar los cambios. 7. El sistema procesa la solicitud y notifica al usuario del estado del registro. 8. El sistema limpia los campos para dar la opción al usuario de registrar otra compra y muestra otra opción para terminar con el registro de compras.
Flujos Alternos	1. Como flujo alternativo al inciso 7, puede ocurrir que el usuario no haya completado todos los campos o existan inconsistencias en la información ingresada, si es el caso, el sistema notificará al usuario y no procederá con el registro.

Continuación de Tabla 27

Post-condición	Compras registradas en el sistema.
Requerimientos trazados	Registro de compras para la gestión de información de inventario de activos fijos.
Notas	Se debe generar un correlativo para el número de inventario del bien adquirido de acuerdo a las especificaciones de tesorería.

Fuente: elaboración propia.

2.9.12.7. Consulta de registros de inventario

Tabla 28. **Especificación de caso de uso: consultar información de inventario**

Caso de Uso	Consultar información de inventario .
Propósito	Consulta de información referente a inventario.
Actor	Tesorero, Coordinador.
Resumen	El usuario con rol de tesorería o coordinador ingresa al sistema y se dispone a consultar información referente al inventario. Se dirige al módulo de inventario y luego selecciona el apartado que necesite entre compras, bienes muebles y/o proveedores. El sistema muestra los resultados encontrados y los presenta al usuario.
Pre-condición	Sesión iniciada con usuario de rol tesorero o coordinador.
Flujo Básico	1. El usuario ingresa al sistema con rol de tesorero o coordinador. 2. Se dispone a consultar información de inventario y se dirige al módulo destinado para el mismo. 3. El usuario selecciona el apartado que desea consultar (proveedores, bienes y/o compras). 4. El sistema carga la vista principal para el apartado seleccionado.

Continuación de Tabla 28

Flujos Alternos	2. Como flujo alternativo al inciso 4, existe la posibilidad de que no haya información almacenada, por lo que no se deberá notificar al usuario.
Post-condición	Información consultada acerca de inventario
Requerimientos trazados	Consulta de proveedores, bienes y compras registradas en el sistema, así como del inventario de activos fijos.
Notas	-

Fuente: elaboración propia.

2.9.12.8. Exportación de registros de inventario

Tabla 29. Especificación de caso de uso: exportar información de inventario

Caso de Uso	Exportar información de inventario.
Propósito	Exportación de información de inventario para disposición fuera del sistema.
Actor	Tesorero, Coordinador.
Resumen	El usuario con rol de tesorería o coordinador ingresa al sistema y se dispone a exportar información de inventario. Para ello, se dirige al módulo de inventario y luego selecciona el apartado que necesite, entre compras, bienes muebles y/o proveedores. Desde ahí selecciona la opción para exportar en formato XLS (Excel) o generar reporte como PDF. El sistema atiende la petición y muestra los resultados.
Pre-condición	Sesión iniciada con usuario de rol tesorero o coordinador.

Continuación de Tabla 29

Flujo Básico	1. El usuario ingresa al sistema con rol de tesorero o coordinador. 2. Se dispone a exportar información de inventario y se dirige al módulo destinado para el mismo. 3. El usuario selecciona el apartado para el cual desea exportar registros (proveedores, bienes y/o compras). 4. El sistema carga la vista principal para el apartado seleccionado. 5. El usuario elige exportar como archivo XLS o reporte en PDF. 6. El sistema procesa la solicitud y muestra los resultados.
Flujos Alternos	3. Como flujo alternativo al inciso 4, existe la posibilidad de que no haya información almacenada, por lo que no se deberá notificar al usuario.
Post-condición	Información exportada acerca de inventario.
Requerimientos trazados	Exportación de información referente a proveedores, bienes y compras registradas en el sistema, así como del inventario de activos fijos.
Notas	-

Fuente: elaboración propia.

2.9.13. Especificaciones adicionales

Puesto que el propósito principal de los proyectos desarrollados es la divulgación de información registrada por estudiantes que conforman los equipos multidisciplinarios, se estableció un registro de log para cada una de las acciones realizadas por el usuario, de tal forma que se establece un compromiso entre el usuario y la información que se ingresa.

El módulo de consulta pública constituyó un geoportal al cual un usuario, sin necesidad de estar registrado en el sistema, puede acceder. El mismo sigue las

guías de diseño del Programa establecidas en el Manual de Normas Gráficas según aplique y se consideró el diseño responsivo para su adaptabilidad en cualquier dispositivo desde el cual se acceda.

2.10. Análisis y diseño

A partir de este flujo de trabajo, dentro de la fase de elaboración, se definieron las clases y atributos como producto de la fase de inicio de la metodología. Posteriormente se representaron los principales estados y secuencia del sistema, y se concluyó con la creación del modelo de datos.

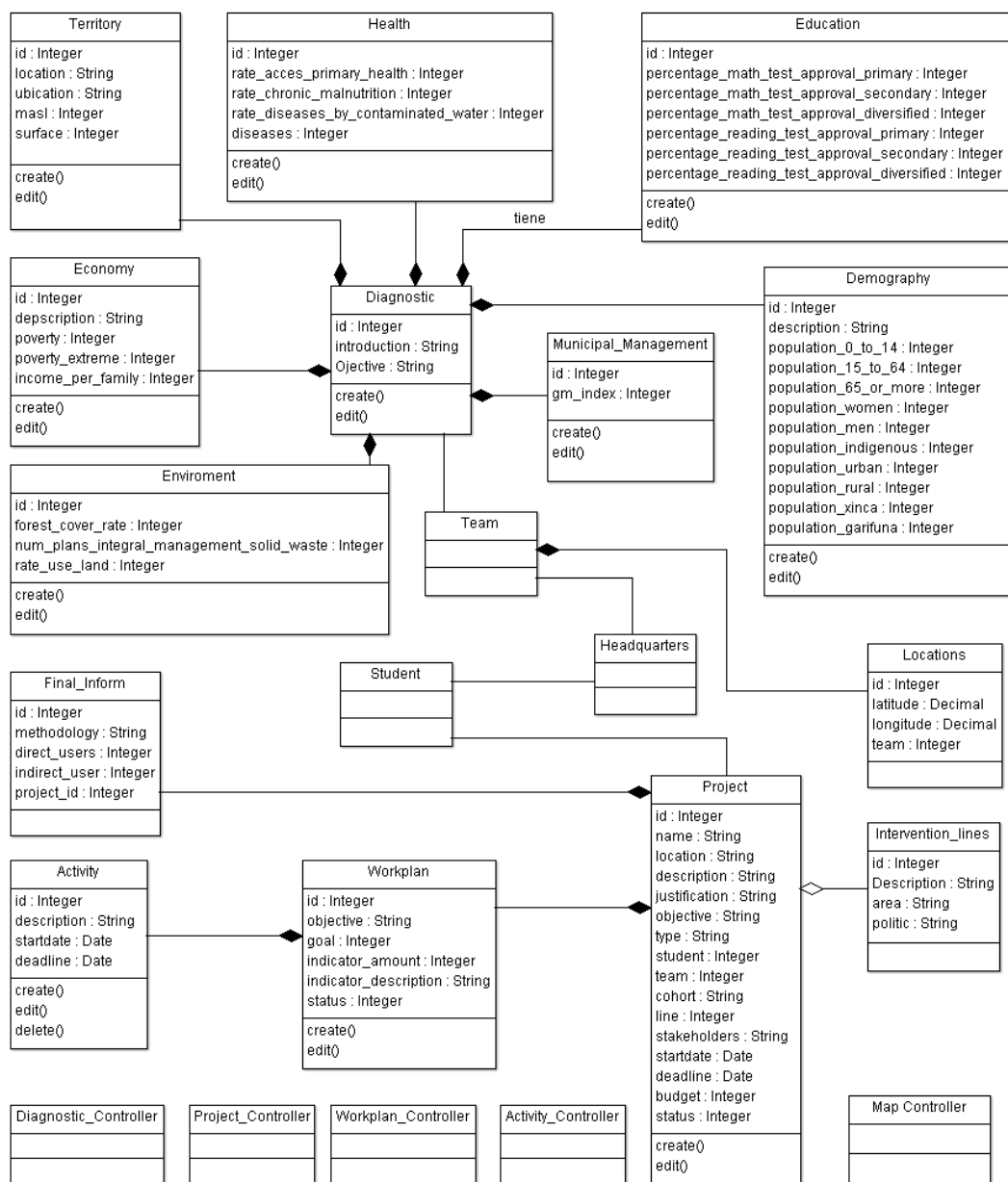
2.10.1. Modelo de análisis y diseño

2.10.1.1. Diagrama de Clases

El diagrama de clases representa parte fundamental en el análisis y diseño del sistema, pues en él se fundamenta la codificación del nuevo software (Shelly y Rosenblatt, 2012). En la figura 18 se ilustra el diagrama de clases para la gestión de diagnósticos y proyectos de EPSUM. Tomando en consideración el uso del *framework* Laravel y su arquitectura MVC, en el diagrama se han incluido las clases que corresponden a modelos y controladores, puesto que las clases que definen las vistas, no tienen ningún tratamiento directo con los datos.

Además, es importante mencionar que todos los modelos de la aplicación heredan de la clase Model de Laravel, al igual que los controladores heredan de la clase Controller, también de Laravel y que, por simplificación, no se representa en el diagrama.

Figura 18. Diagrama de clases de módulo para la gestión de informes diagnósticos y de proyectos



Fuente: elaboración propia.

Como puede observarse en el diagrama de la figura 18, destacan las clases Diagnostic y Project, puesto que el módulo está orientado al tratamiento de ese tipo de registros en el sistema. Dichas clases tienen algunas relaciones importantes que las complementan, entre las que vale la pena resaltar:

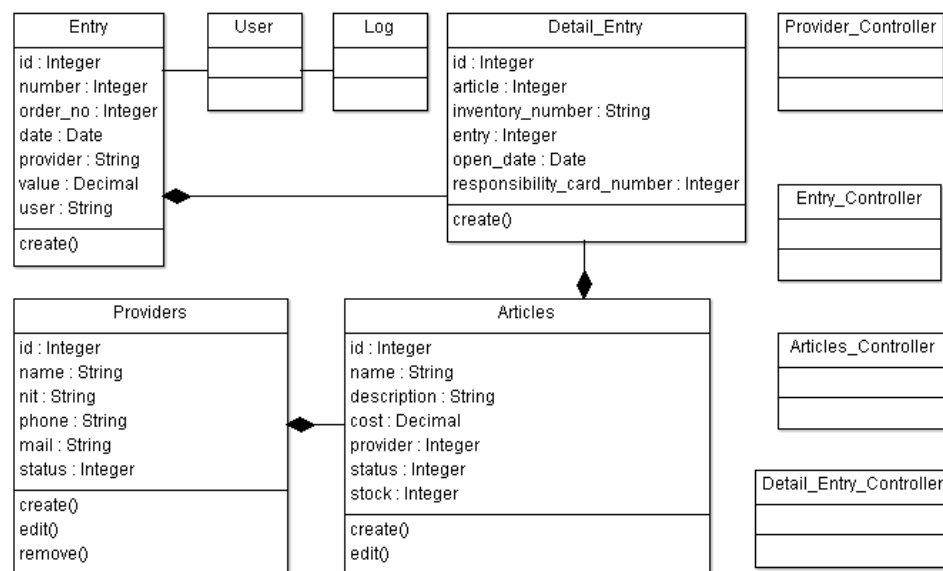
- Relación de composición entre las clases Territory, Demography, Economy, Health, Education, Environment o Municipal Management y la clase Diagnostic. Este tipo de relación incluyente indica que no puede existir una instancia de cualquiera de las clases anteriores sin un registro previo de un diagnóstico. Por tal razón, no es posible registrar información territorial, demográfica o de salud, sin definir un diagnóstico en el sistema.
- Relación de composición entre las clases Activity, Workplan y la clase Project. A diferencia de la clase Diagnostic, la clase Project solo tiene una relación de composición con la clase Workplan, la cual a su vez tiene el mismo tipo de relación incluyente con la clase Activity. De esta manera, se deduce que no puede existir un registro de un plan de trabajo, sin antes haber un registro de proyecto y de la misma forma sucede con instancias de la clase Activity.
- Relación de agregación de la clase Intervention Lines y la clase Project. Esta relación nos muestra que no existe dependencia entre ambas clases y que pueden ser referenciadas por separado. Sin embargo, cabe resaltar que la clase Intervention Lines, únicamente trata registros preestablecidos en la Metodología para Equipos Multidisciplinarios del Programa EPSUM (líneas de intervención), por lo que el usuario no tendrá opción para interactuar directamente con tal entidad.

A efecto de normalizar la base de datos se introduce la clase Locations, en la cual se guardan las coordenadas de los equipos para la georreferenciación de la información.

El diagrama de la figura 19 muestra la representación de la gestión de inventario de activos fijos. En él destaca la clase Detail Entry, que tiene una relación de composición con la clase Entry y la clase Article. Esto se debe a que la relación entre Entry y Article tiene una cardinalidad de muchos a muchos (ver Modelo Entidad-Relación), y ello obliga a la creación de una entidad intermedia, Detail Entry; en ese sentido, no puede existir una instancia de ésta última, sin antes haber un registro previo de un artículo y un identificador de ingreso.

Por otro lado, se muestran las clases que definen los controladores que tratan con los datos referentes al módulo de gestión de inventario. Además, se expresan las clases User y Log, que ya están definidas en el sistema, pero se agregan con el fin de ilustrar la interacción para el registro de log del usuario.

Figura 19. Diagrama de clases de gestión de inventario de activos fijos

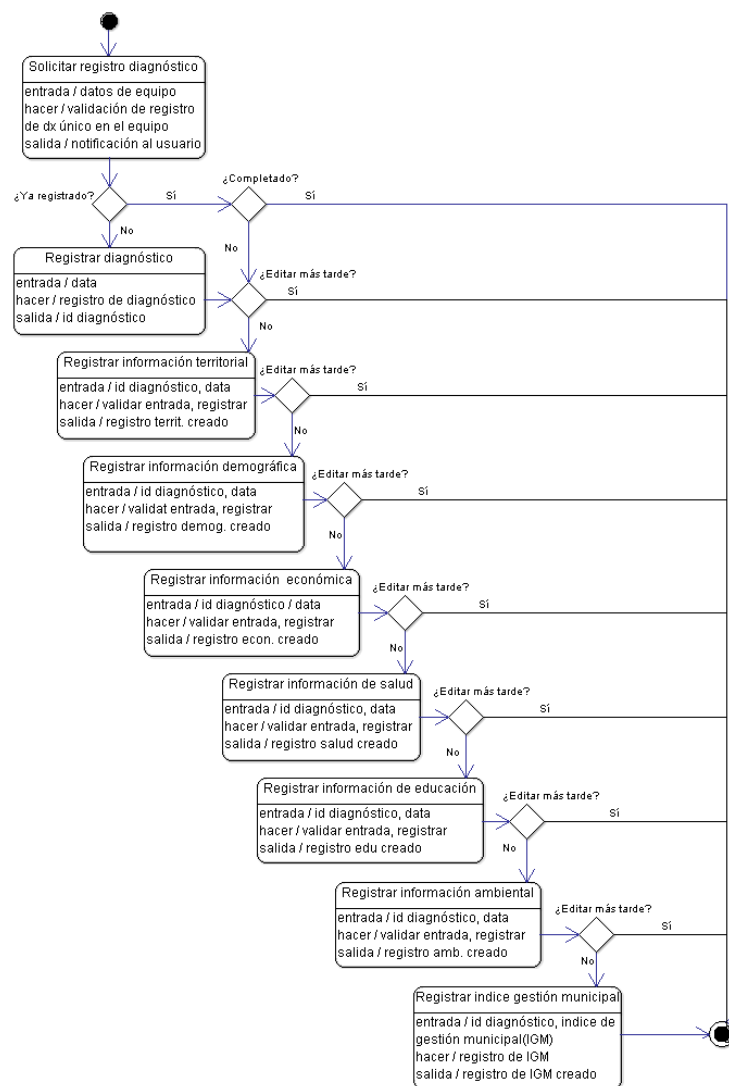


Fuente: elaboración propia.

2.10.1.2. Diagrama de estados

Un diagrama de estados nos permite analizar la dinámica de los objetos en el sistema y ver cómo estos evolucionan en el transcurso del proceso.

Figura 20. Diagrama de estados para el registro de diagnósticos



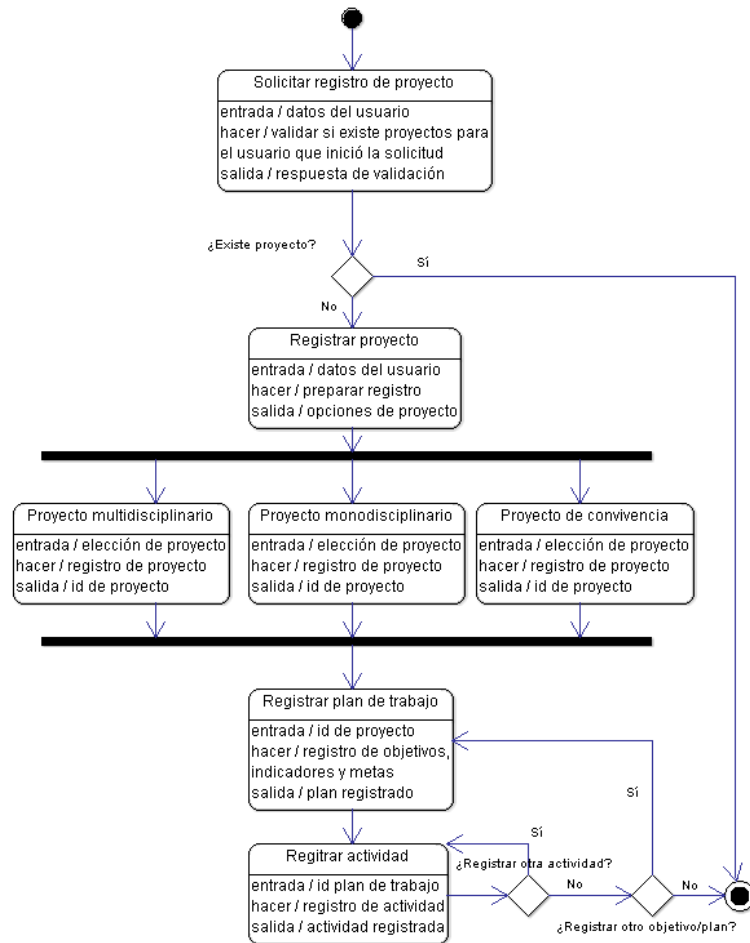
Fuente: elaboración propia.

De acuerdo al análisis de los requerimientos y tomando en cuenta el trabajo colaborativo para el informe diagnóstico de los Equipos Multidisciplinarios de EPSUM, se determinó una lógica para su registro como se muestra en el diagrama de la figura 20. La idea central representada en dicho diagrama, modela la opción de seguir editando más tarde para guardar de manera parcial el informe diagnóstico. Esta opción, además de permitir seguir editando en un momento posterior, da la oportunidad a que otro integrante del equipo pueda continuar con el registro hasta el punto en el cual el último usuario haya guardado cambios, cumpliendo así el objetivo del trabajo en equipo.

En el caso del registro de proyectos, se ha mencionado la clasificación de los mismos en tres tipos: Multidisciplinarios, Monodisciplinarios y de Convivencia Comunitaria; con lo cual se deduce que deberán existir registros de por lo menos 3 proyectos por equipo, uno de cada tipo. Esto se debe a que los proyectos Multidisciplinario y el de Convivencia Comunitaria pueden ser editados por cualquiera de los integrantes del equipo y el proyecto Monodisciplinario constituye un registro individual por cada integrante, como se muestra en la figura 21.

En el mismo diagrama, se ilustra de manera general el proceso de registro de un proyecto en el sistema, el cual comprende 3 fases principales: registro del perfil del proyecto, registro del plan de trabajo y registro de actividades. Siguiendo el planteamiento del párrafo anterior, también se destaca en el diagrama, la concurrencia en el registro de los tres tipos de proyectos para un mismo equipo. Una vez registrado un perfil de cualquier tipo de proyecto, se procede al registro del plan de trabajo, cuya definición central es un objetivo específico; seguidamente, se registran las actividades para la consecución de tal objetivo.

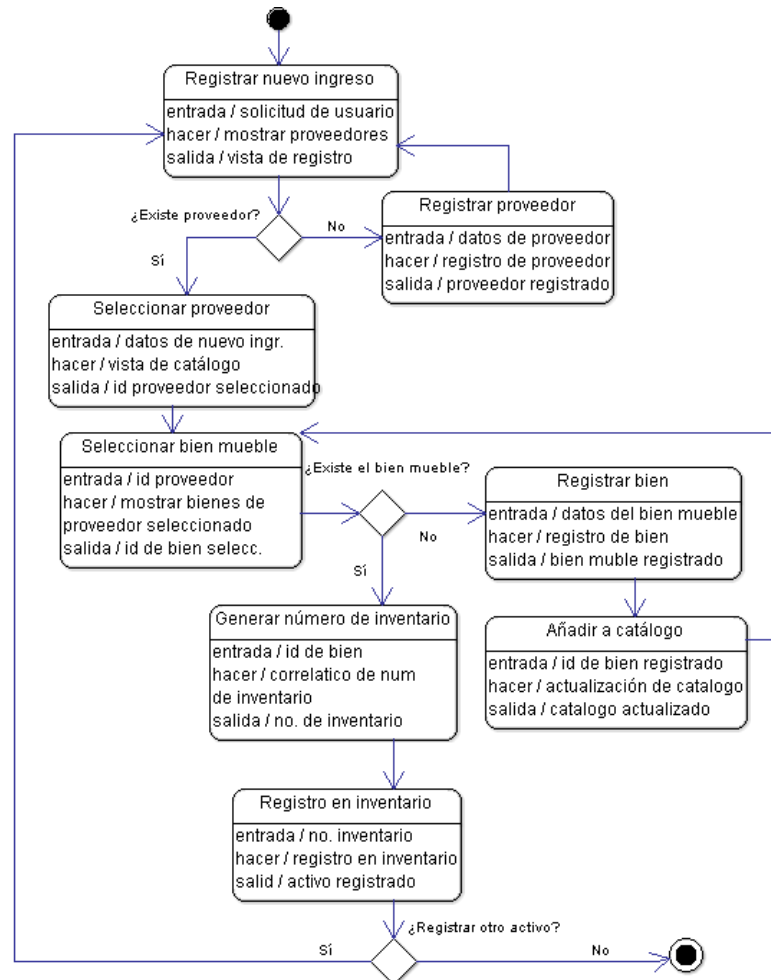
Figura 21. Diagrama de estados para el registro general de proyectos



Fuente: elaboración propia.

Otro caso importante que vale la pena representar, es el registro de un nuevo ingreso en el sistema de gestión de inventario. Este proceso se ilustra en la figura 22, en donde un usuario hace la solicitud inicial para el registro de un nuevo ingreso de inventario.

Figura 22. Diagrama de estados para el registro general de inventario



Fuente: elaboración propia.

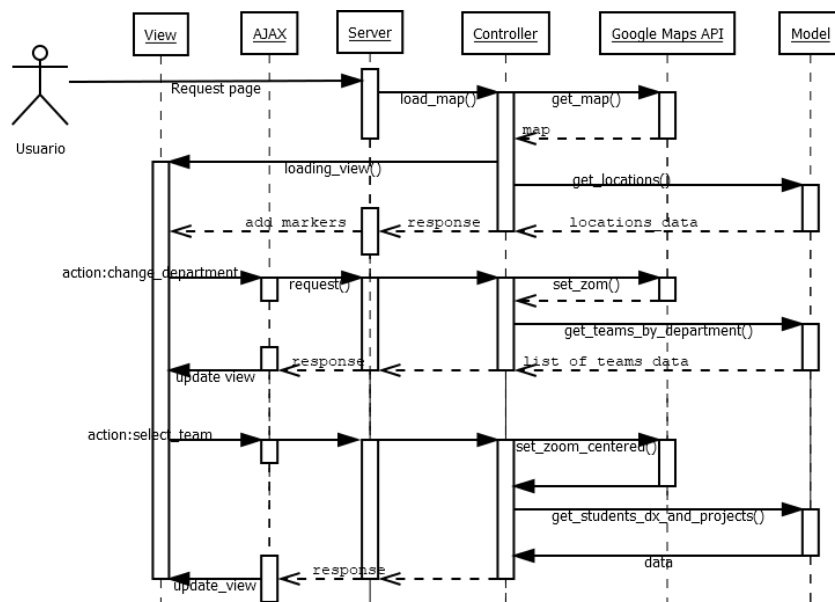
Como puede observarse en el diagrama, el flujo normal de eventos lleva a un estado en donde, antes del registro final del nuevo ingreso, se genera un correlativo para el bien mueble, que corresponde al número de inventario, tal como establece el Reglamento para el registro y control de bienes muebles y otros activos fijos de la Universidad de San Carlos de Guatemala.

2.10.1.3. Diagrama de secuencia

Una manera óptima para representar la interacción entre componentes lo constituyen los diagramas de secuencia. Un diagrama de secuencia es un modelo dinámico que muestra el intercambio entre clases, objetos y/o procesos en un período de tiempo determinado y que permite cumplir los objetivos de los casos de uso. La interacción entre componentes se realiza a través de mensajes que se envían entre las líneas de vida establecidas.

Puesto que el propósito del registro de informes de diagnóstico y de proyectos de los Equipos Multidisciplinarios de EPSUM es la divulgación, se presenta en la figura 23, el proceso de consulta a través del geoportal.

Figura 23. Diagrama de secuencia



Fuente: elaboración propia.

De manera general, puede apreciarse en el diagrama de secuencia cómo el usuario inicia una petición a través de las rutas especificadas en el sistema, se redirige al controlador específico y este a su vez gestiona a través de las funciones definidas para obtener el modelo de datos. Posteriormente retorna una respuesta al usuario quien verá reflejados los resultados en la vista.

Las acciones mostradas en el diagrama de la figura 23 muestran la interacción descrita anteriormente entre las distintas capas del sistema. Cuando un usuario procede a elegir una opción (como una cohorte o un equipo) a través de un componente específico en la vista, se activa un evento que realiza una petición a través de AJAX, que explícitamente posee una ruta que lleva hasta el controlador para luego obtener los datos y mostrar los resultados al usuario. En el mismo diagrama se ve la interacción con la interfaz de programación de aplicaciones (API) de Google Maps para Javascript, la cual provee algunas funciones para interactuar con el mapa y realizar ciertos eventos como el *paneo* y *zoom* del mapa.

Es importante resaltar que, al hacer uso de AJAX, la vista se mantiene sin ser refrescada tras ejecutar las acciones. Esto además de mejorar la experiencia del usuario, se planeó con el fin de facilitar la navegación para dispositivos móviles, haciendo que el módulo del geoportal se asemeje a una aplicación nativa para el usuario. Si bien el uso de esta tecnología resultó un tanto complicado para el desarrollo de la aplicación web, otorgó flexibilidad para cumplir con los requisitos definidos en la toma de requerimientos.

2.10.1.4. Modelo de datos

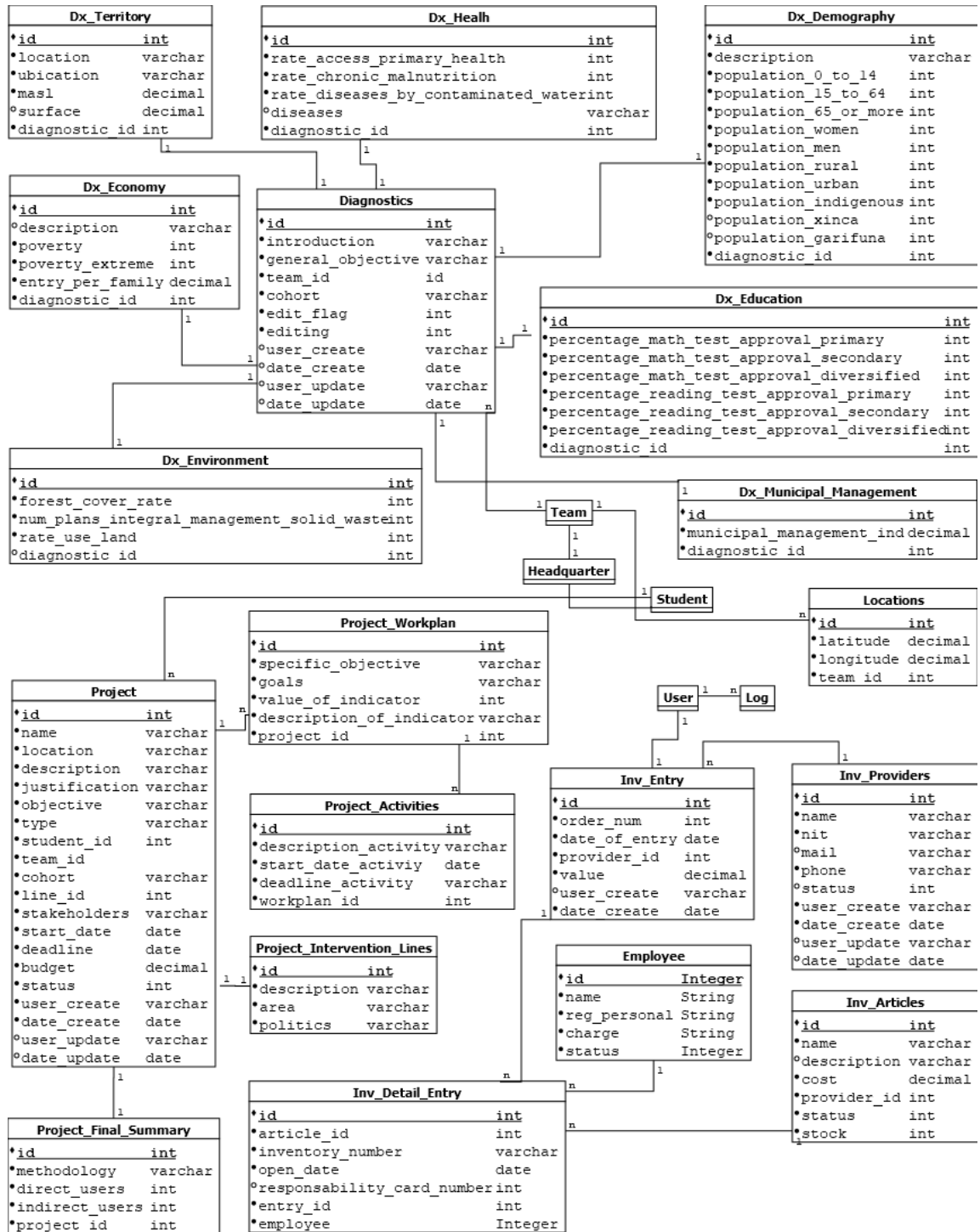
El modelo de datos está enfocado a la representación lógica de la estructura de datos y condiciones que guardan la consistencia de los registros. El modelo Entidad-Relación es el ejemplo más claro para la representación de los elementos en un sistema de información. El mismo proporciona una visión completa del conjunto de datos que intervienen en el sistema, está compuesto por tablas o entidades, relaciones y restricciones que cumplen los requerimientos establecidos en la fase de análisis.

La figura 24 muestra la representación del modelo de datos definidos a través de la toma de requerimientos para los módulos desarrollados. En él destacan las entidades Diagnostic y Project, que corresponden a los informes generados por los estudiantes. También vale la pena mencionar la representación del modelo lógico de datos para el sistema de gestión de inventario, que puede distinguirse en la parte inferior derecha.

Se presentan a continuación, algunos elementos del diagrama que desempeñan funciones importantes o que necesitan aclaración.

- Diagnostic. Entidad importante dentro del módulo de registro de informes, pues define la estructura de un informe de diagnóstico comunitario realizado por los equipos multidisciplinarios. Entre sus atributos destacan:
 - Id. Se refiere a la clave primaria que identifica a un diagnóstico entre todos los registros.
 - Introduction. Almacena un resumen introductorio al informe, el registro tiene un número limitado de caracteres.

Figura 24. Modelo relacional



Fuente: elaboración propia.

- Team_Id. Representa una clave foránea que guarda la correspondencia con el equipo multidisciplinario que realiza el registro.
 - Cohort. Define la cohorte del equipo multidisciplinario. Este atributo es importante, pues establece la diferencia entre un diagnóstico de una cohorte y otra, para un mismo equipo.
 - Edit_Flag. Guarda el avance de registro del diagnóstico, por lo cual se actualiza cada vez que el usuario registra alguna otra información de las clases que componen al diagnóstico.
 - Editing. Determina si algún integrante del equipo está editando, por lo que se actualiza cada vez que un usuario decide continuar el registro del diagnóstico en otro momento.
- Project. Representa la entidad que define un proyecto dentro del sistema. De manera general, establece los perfiles de proyectos a registrar por los estudiantes. Se describen algunos de sus atributos:
 - Id. Clave primaria que identifica un proyecto entre todos los registros.
 - Type. Puesto que se diferencian tres tipos de proyectos, este atributo almacena dicha clasificación tomando valores como “Multidisciplinario”, “Monodisciplinario” y “Convivencia Comunitaria”, según la metodología del Programa EPSUM para equipos multidisciplinarios.
 - Team_Id. Clave foránea que mantiene la relación entre un proyecto y un equipo. Este atributo aplica para los proyectos multidisciplinarios y de convivencia.
 - Student_Id. Guarda la relación entre un proyecto y un estudiante. Su inclusión es importante debido a que los proyectos

monodisciplinarios son de carácter individual y no constituyen un registro en equipo como sucede con los proyectos multidisciplinarios y los de convivencia comunitaria.

- Cohort. Define la cohorte para la cual se hizo el registro del proyecto, importante para los registros de equipos multidisciplinarios y de convivencia, pues están enlazados a un equipo y no a un registro único como lo serían los estudiantes asignados a dicho equipo.
 - Line_Id. Se refiere a las líneas de intervención que, según la metodología del Programa EPSUM, enfocan los proyectos a alguna problemática en el territorio. Este atributo es una clave foránea a la entidad Intervention Lines, en la cual se definen las líneas de intervención y sus principales propiedades.
 - Status. Atributo que guarda el estado del proyecto, el cual toma únicamente los valores 0 o 1, que representan el proyecto en estado de ejecución o finalizado, respectivamente.
-
- Inv_Detail_Entry. Esta tabla es de importancia pues almacena el detalle de un activo en el sistema de gestión de inventario. Su existencia permite hacer el enlace entre los datos del registro de entrada, bienes/artículos, y empleados a través de claves foráneas. Entre sus atributos destacan:
 - Article_Id. Clave foránea que mantiene la correspondencia con un bien o artículo registrado previamente.
 - Inventory_Number. Almacena un número de inventario generado automáticamente por el sistema. El mismo sigue un estándar definido por EPSUM y el número corresponde a un correlativo.

- Responsibility_Card_Number. Número de tarjeta de responsabilidad generado a partir del número de inventario. Su fin es únicamente para control en el sistema.
- Entry_Id. Clave foránea que guarda la relación con el ingreso registrado (que registra la información de la orden/compra) previamente.
- Employee_Id. Se refiere a la persona a quien se carga un activo en inventario. Constituye una clave foránea que hace el enlace con el personal registrado en la tabla Employees.

Como puede observarse en el diagrama, la tabla Diagnostic tiene una relación de uno a uno con entidades como Territory, Demography o Education, por tal razón, se define que, para un diagnóstico, únicamente puede hacerse una instancia de alguna de las entidades mencionadas. En el caso de la tabla Workplans, se observa que existe una relación de uno a muchos con la entidad Activities, deduciendo que un plan de trabajo puede tener muchas actividades asociadas. Por otra parte, en las tablas correspondientes al inventario de activos fijos, destaca la relación de muchos a muchos entre las entidades Entry y Articles, puesto que un nuevo ingreso puede tener tantos artículos como sea el caso, y un artículo puede estar asociado a muchos ingresos; derivado de lo anterior se creó la tabla intermediaria Detail_Entry que guarda la relación entre las entidades mencionadas anteriormente, a través de sus claves foráneas.

2.11. Implementación

La etapa de implementación constituye la construcción del software según los modelos elaborados en la fase de análisis y diseño. Como se ha mencionado anteriormente, para los módulos desarrollados se utilizó el *framework* Laravel, de

acuerdo a las tecnologías aplicadas en el sistema previamente implantado que corresponde a la Plataforma de EPSUM.

El desarrollo del proyecto inició con la agregación de las migraciones, las cuales definen las tablas en la base de datos y cuyo propósito es registrar un histórico para el control de versiones de la base de datos. Estas clases heredan de Migration, de Laravel y se encuentran en el paquete *migrations*, que a su vez se encuentra en *database*.

Posteriormente, se crearon los modelos que interactúan con los datos, estos se encuentran en el paquete *app*. Dentro del mismo paquete, en el directorio Http, se definieron los controladores considerados según los modelos en la etapa de análisis y diseño. Los controladores heredan de la clase Controller de Laravel y los mismos implementan métodos que retornan data en formato json o una vista específica. Las vistas hacen uso de la librería *Blade*, la cual brinda funcionalidades útiles, tal como la herencia de una vista principal, mostrar datos de variables y utilizar estructuras de control. Algunas dependencias utilizadas en las vistas las constituyen archivos css y de javascript, los cuales fueron agregados en las carpetas respectivas dentro del paquete *public*.

Para una explicación más amplia sobre las clases y métodos implementados, referirse al Apéndice 6, Manual de documentación técnica y de mantenimiento.

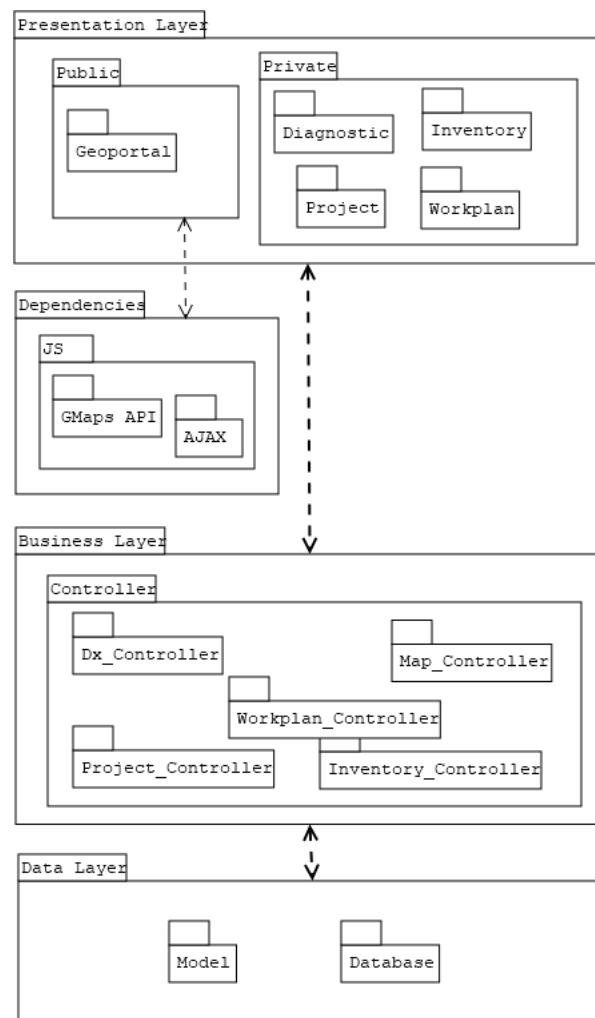
2.11.1. Diagrama de paquetes

Un diagrama de paquetes permite representar estructuras lógicas entre las dependencias de paquetes. Tal como se explicó en el apartado 2.2.4 sobre las tecnologías de desarrollo, Laravel sigue el patrón de diseño MVC (Modelo-Vista-

Controlador). Por tal razón, la estructura de paquetes por defecto está distribuida a modo de facilitar la implementación de la capa de presentación, de negocio y de datos.

En la figura 25 se ilustra un diagrama de paquetes simplificado del proyecto desarrollado.

Figura 25. Diagrama de paquetes



Fuente: elaboración propia.

2.11.2. Herramientas de desarrollo

La implementación del software requirió el uso las siguientes herramientas:

- XAMPP 5.7.23, paquete de software libre que permite ejecutar aplicaciones web basadas en PHP. Las tecnologías que integra esta herramienta son: Apache 2.4.18, PHP 7.0.30, MySQL 5.7.23 y phpMyAdmin en su versión 4.8.3.
- Atom v1.25.0, editor de código, multiplataforma y de código abierto. Está escrito en CoffeeScript y Less; su soporte para plug-ins lo transforma en un IDE adaptado al proyecto a desarrollar.
- Composer, un manejador de dependencias que descarga todas las dependencias directas e indirectas, permitiendo instalar las versiones de librerías que el proyecto requiera.
- Bitbucket, servicio de alojamiento basado en web para proyectos que integran un sistema de control de versiones, como Git.
- Navegador web (Firefox y Google Chrome), utilizado para test de la aplicación y depuración de errores mediante la consola.

2.11.2.1. Software de terceros

Para incorporar todas las funcionalidades especificadas en los requerimientos se ha hecho uso de algunas dependencias que facilitaron la implementación del proyecto.

- Maps Javascript API
- jQuery 3.1.1
- Dompdf 0.8.0
- PhpExcel 2.0.0

2.11.3. Pruebas

Durante la fase de construcción, se realizaron algunas pruebas unitarias para garantizar el correcto funcionamiento de funciones de la aplicación. Para tal efecto, Laravel cuenta con su propio componente para pruebas dentro de la aplicación, el cual es ejecutado a través de la interfaz de líneas de comandos e indicando el nombre de la prueba a realizar. Además, Laravel dispone de soporte para PHPUnit, un framework para la realización de pruebas unitarias de PHP. Las especificaciones de las pruebas realizadas pueden comprobarse a detalle en el Apéndice 6, Manual de especificaciones técnicas y de mantenimiento.

Entre otras técnicas de depuración, se utilizó la consola de los navegadores web Firefox y Google Chrome, consultando específicamente el monitor de red y los elementos contenidos en los paquetes de respuesta. Además, al término de un hito importante, y a modo de presentación a la coordinación del Programa EPSUM, se socializaron algunos avances del proyecto, realizando pruebas específicas para comprobar el funcionamiento deseado en la toma de requerimientos. Las reuniones concretadas se detallan en los informes mensuales entregados al supervisor de EPSUM, mismos que pueden ser consultados en el Apéndice 7, del contenido del CD.

2.11.4. Configuración

Para un correcto funcionamiento de la aplicación, se dispone de una carpeta llamada *config*, en donde se recomienda agregar todos los archivos de configuración nuevos.

La configuración del entorno para la aplicación se realiza a través de la biblioteca Dotenv de PHP. Este módulo permite gestionar la configuración de la aplicación mediante el archivo de configuración de entorno con extensión *.env*. Dicho archivo contiene las variables de entorno necesarias para la gestión de la aplicación y la conexión a la base de datos. Este archivo deberá ser modificado, si fuera necesario, de acuerdo al entorno en donde se implemente la aplicación.

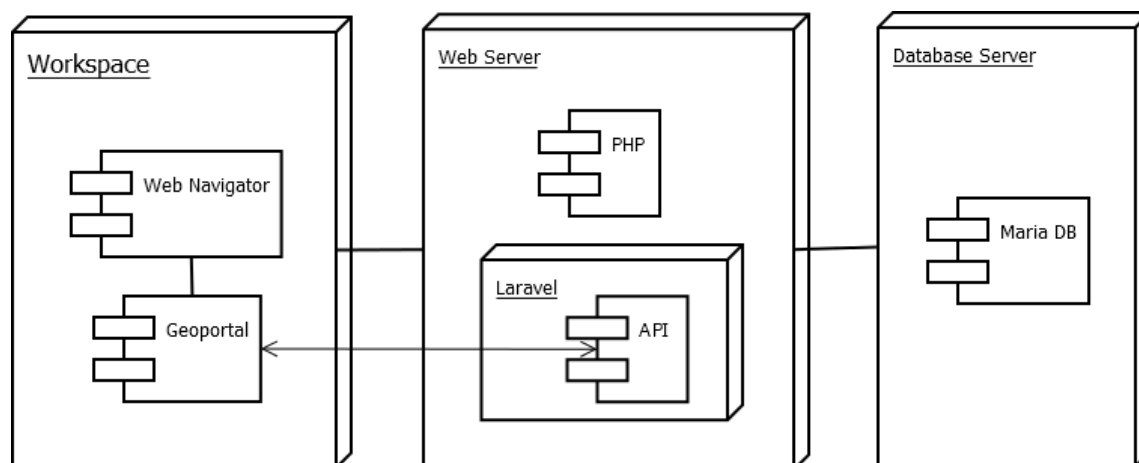
2.11.5. Despliegue

Configurada la aplicación, se ha procedido al despliegue correspondiente al último flujo de trabajo de la metodología aplicada. Para ello, se inició con la ejecución de las migraciones para la creación de las tablas respectivas en la base de datos, la cual se encuentra alojada en el servidor c3 del departamento de Procesamiento de Datos de la Universidad de San Carlos. La realización de las migraciones fue a través de *artisan* como interfaz de línea de comandos, por lo cual no fue necesario el acceso específico a la base de datos en el servidor. Una vez creadas las nuevas entidades se ha procedido con la inserción de registros de tablas estáticas, como lo son las de locaciones y líneas de intervención.

Los archivos creados para los proyectos realizados, fueron cargados y actualizados en el servidor c3, haciendo el *pull* desde el repositorio de bitbucket en donde se encuentra alojado el código actualizado de la aplicación.

En la figura 26, se muestran los distintos nodos que agrupan los distintos componentes de software que se ejecutan en ellos y que se enlazan con otros nodos para realizar la funcionalidad deseada en el sistema.

Figura 26. **Diagrama de despliegue**



Fuente: elaboración propia.

Como se ha indicado con anterioridad, el servidor web corresponde al servidor c3 del departamento de Procesamiento de Datos, el cual es accedido vía SSH con una llave ppk proporcionada por el mismo departamento. En el mismo servidor se encuentra alojada la base de datos cuyas credenciales de acceso se encuentran indicadas en el archivo de configuración de entorno que se mencionó en el apartado de configuración.

Como se observa en el diagrama de la figura 26, el procesador de PHP es el subsistema que ejecuta todas las instrucciones del software y que permite ejecutar la aplicación, por lo que el servidor web actúa únicamente como intermediario para ejecutar las conexiones entre el cliente y servidor.

2.12. Gestión del proyecto

Puesto que la metodología aplicada está basada en un proceso iterativo e incremental, se presenta un resumen de las iteraciones realizadas en las fases de inicio, elaboración, construcción y transición en los proyectos tal cual fueron presentados en la tabla 4.

2.12.1. Evaluación de iteraciones

En la tabla 30, se presentan las iteraciones realizadas para el módulo de registro y consulta de informes de diagnóstico y de proyectos. Para la fase de inicio se observa que sólo se realizó una iteración a partir de la tercera semana de enero, continuando con la fase de elaboración, a partir de la cuarta semana del mismo mes y culminando la línea base de la fase de inicio en la misma semana. La segunda iteración para la fase de elaboración se realizó a partir de la primera semana de febrero y culminó con el análisis y diseño de la aplicación.

Tabla 30. Iteraciones para el módulo de registro y consulta de informes

Fase	Enero			Febrero				Marzo				Abril				Mayo			
	3	4	5	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
Inicio																			
Elaboración																			
Construcción																			
Transición																			

Fuente: elaboración propia.

Es importante destacar que, durante estas fases, se realizó parte del anteproyecto entregado a la unidad académica, el cual incluyó información obtenida como producto de estas iteraciones. Durante la fase de construcción se realizaron 3 iteraciones: la primera definió la estructura, modelos y migraciones para el proyecto. La segunda iteración concluyó con el registro de informes diagnósticos y de proyectos y la tercera con la implementación del geoportal. En las últimas dos iteraciones se hicieron modificaciones a los resultados de la anterior y se debe mencionar que existió una demora en la finalización de productos a partir de la segunda iteración (presentada en color naranja), esto debido a una reestructuración del informe diagnóstico. Por último, se hicieron dos iteraciones en la fase de transición, concluyendo con la implementación, despliegue y realización de manuales.

De la misma manera, la tabla 31 muestra la ejecución de las iteraciones en cada una de las fases para el módulo de gestión de inventario, realizando una iteración para la fase de inicio y para la de elaboración y dos iteraciones para las fases de construcción y transición.

Tabla 31. Iteraciones para el módulo de gestión de inventario

Fase	Mayo			Junio				Julio				Agosto			
	3	4	5	1	2	3	4	1	2	3	4	1	2	3	4
Inicio															
Elaboración															
Construcción															
Transición															

Fuente: elaboración propia.

Las actividades realizadas se encuentran descritas en los informes mensuales rendidos al programa y las mismas describen los aciertos y desaciertos obtenidos. Dichos informes pueden consultarse en el Apéndice 7, contenido del CD, en el apartado de Informes de EPSUM.

Es importante mencionar que, en la planificación, no se consideraron actividades propias de EPSUM, como el taller de inducción realizado durante la primera semana de febrero, mas no así los descansos oficiales por parte de la Universidad de San Carlos de Guatemala.

2.12.2. Matriz de riesgos

En la gestión de proyectos, cualquier actividad está expuesta a riesgos de diversa índole que influyen de distintas formas en los resultados esperados. La capacidad de identificar estas posibles eventualidades, origen e impacto, constituyen parte importante para la consecución de los objetivos trazados.

La tabla 32 plantea los riesgos en los cuales la aplicación podría verse comprometida, así como su descripción y las acciones de mitigación para la posible reducción de su impacto.

Tabla 32. **Riesgos inherentes**

No.	Riesgo	Descripción	Medida de mitigación
1	Ataques DDoS	Puesto que se trata de una aplicación web, no está exenta a cualquier tipo de situaciones que comprometan la información del sitio. Los ataques distribuidos de denegación de servicios sería uno de los riesgos más comunes que enfrentaría la aplicación.	Monitoreo constante de la actividad que genere el uso de la aplicación. Contratación de servicios especializados para la prevención de ataques DDoS.
2	Mantenimiento del servidor	Problemas de acceso a la información a causa de mantenimientos imprevistos en el servidor por parte del Departamento de Procesamiento de datos, debido a que la aplicación se encuentra alojada en el servidor c3 que pertenece a la Universidad de San Carlos de Guatemala.	Notificación de mantenimiento del servidor de base de datos para la realización de backup de la información.

Fuente: elaboración propia.

La tabla 33 presenta los criterios establecidos para el nivel de riesgo con la respectiva medida porcentual. En la tabla 34 se muestra la valoración del impacto contra la frecuencia o probabilidad de que ocurra el riesgo, cuya calificación representa la clasificación del nivel de riesgo. El resumen de los resultados como producto del impacto y frecuencia de los riesgos planteados se observan en la tabla 35.

Tabla 33. **Criterios de nivel de riesgo**

Calificación	Código de color	Criterio	Porcentaje de nivel de riesgo
1		Insignificante	10%
2		Bajo	20%
3		Medio	50%
4		Moderado	80%
5		Alto	100%

Fuente: elaboración propia.

Tabla 34. **Valoración de riesgo**

IMPACTO	Alto	4	5	5
	Medio	3	3	5
	Bajo	1	2	4
		Bajo	Medio	Alto
PROBABILIDAD DE OCURRENCIA				

Fuente: elaboración propia.

Tabla 35. **Resultados de valoración de riesgos**

No. Riesgo	Probabilidad	Impacto	Calificación	Responsable
1	Media	Alto	5	Administrador del sistema y Depto. Procesamiento de datos.
2	Baja	Medio	3	Depto. Procesamiento de datos

Fuente: elaboración propia.

2.13. Beneficios del proyecto

El geoportal de EPSUM está orientado a la divulgación del trabajo que realizan los estudiantes en todas las sedes que actualmente atiende el Programa. El mismo permite la consulta de información por departamento, cohorte y equipo, referente a diagnósticos, proyectos y estudiantes inscritos en el Programa, mediante un mapa interactivo que facilita a cualquier usuario la búsqueda de información generada por los equipos de EPSUM. Con la implementación de un diseño responsivo, se proyecta que la información que se almacena en el sistema pueda tener mayor alcance, al poder consultarse la misma desde distintos dispositivos.

Para apoyar con las tareas administrativas del Programa EPSUM, el módulo de gestión de inventario permite ingresar de una manera ágil, los bienes activos que se adquieran por parte de la institución. Debido que el mismo trata información sensible, se hicieron las consideraciones necesarias para el acceso restringido de acuerdo a los roles definidos en el Sistema de Información.

En ambos casos se contempló la generación de reportes según la información solicitada, por tal razón, tanto los estudiantes, como la coordinación y supervisores, pueden descargar los reportes como informes de diagnósticos y de proyectos. Además, como extra en la gestión de información administrativa, pueden generarse reportes en formato XLS (Excel) para la realización de cálculos según se requiera.

2.14. Valoración económica del proyecto

En esta sección se presenta una estimación del costo técnico asociado a los proyectos desarrollados, según los roles presentados en la tabla 36 y su estipendio de acuerdo a las competencias laborales de cada perfil, resumido en la tabla 37. En la tabla 38 se muestra el tiempo asignado a las actividades desarrolladas y en la tabla 39 se presenta la estimación del costo total del proyecto, tomando en cuenta los costos indirectos asociados.

Tabla 36. **Perfiles de recurso humano**

Rol	Descripción
Analista de sistemas	Recopila la información para satisfacer las especificaciones requeridas por una empresa para el desarrollo del software. Trabaja en estrecha colaboración con los <i>stakeholders</i> .
Desarrollador	Codifica la solución de software diseñada mediante el análisis previo y utilizando la plataforma de desarrollo y lenguajes de programación de la arquitectura establecida.
Documentador de software	Realiza la documentación técnica del software en cada una de las etapas del proyecto y facilita la información generada a las instancias correspondientes.
Gestor/Director de proyecto	Encargado de la planificación y control de cada una de las fases del proyecto para la consecución de los objetivos propuestos para el mismo.

Fuente: elaboración propia.

Tabla 37. **Costos de recurso humano**

Rol	Costo/hora	Horas	Total
Desarrollador master	Q 75.00	240	Q 18,000.00
Documentador	Q 25.00	40	Q 1,000.00
Analista de sistemas	Q 80.00	120	Q 9,600.00
Gerente de proyecto	Q 100.00	400	Q 40,000.00
Total			Q 68,600.00

Fuente: elaboración propia.

Tabla 38. **Tiempo asignado a las actividades desarrolladas**

Fase	Actividad	Duración (horas)
Inicio	Especificación de requerimientos	8
	Conceptualización del sistema	20
	Modelo de casos de uso	8
Elaboración	Requerimientos adicionales	8
	Estructura y arquitectura de software	12
	Modelado UML	20
	Modelo Entidad-Relación	4
	Plan de trabajo	16
Construcción	Codificación	240
Transición	Implementación y despliegue	8
	Pruebas del sistema	16
	Documentación	40
Total		400

Fuente: elaboración propia.

Tabla 39. Costo total del proyecto

Concepto	Costos fijos	Costos variables			Total
		Costo unitario	Cant.	Unidad	
Costos directos					
Mano de obra	Q 68,600.00		1	Global	Q 68,600.00
Capacitación		Q 250.00	4	Global	Q 1,000.00
Costos indirectos					
Energía eléctrica		Q 500.00	5.5	Mes	Q 2,750.00
Internet		Q 355.00	5.5	Mes	Q 1,952.50
Total					Q 74,302.50
Impuesto					Q 8,916.30
Total con impuestos					Q 83,218.80

Fuente: elaboración propia.

En la tabla 39, los costos indirectos hacen referencia al concepto de gastos realizados durante la realización del proyecto en cada una de las fases anteriormente descritas. Dichos costos no reflejan necesariamente los servicios y/o facilidades proveídos por el Programa EPSUM durante la realización de la práctica en sus instalaciones.

3. FASE DE ENSEÑANZA-APRENDIZAJE

3.1. Capacitación

Derivado de los productos realizados, es necesaria la capacitación con el objetivo de instruir a los usuarios sobre el uso del software desarrollado. Las capacitaciones han sido impartidas de acuerdo a los roles definidos en la aplicación a modo de tratar las funcionalidades específicas de cada módulo.

3.1.1. Programación de capacitaciones

De acuerdo a la metodología aplicada y a la planificación para los proyectos desarrollados, en la tabla 40 se presenta un calendario de las presentaciones realizadas para la socialización de resultados. En la tabla 41 se muestra un cronograma de las capacitaciones realizadas las cuales fueron impartidas tanto a estudiantes como al personal de EPSUM.

El material utilizado correspondiente a las presentaciones para el Módulo de Registro de Información de Diagnósticos y Planes de Trabajo se muestra en el Apéndice 7, del Contenido de CD.

Tabla 40. Planificación de presentaciones

Actividad	Dirigida a	Fecha	Objetivos
Revisión general y avalado por parte de supervisora de EPSUM y Coordinación del programa	Licda. Jenifer Pinzón Ing. Agr. Mirna Regina Valiente	26/06/2018	Revisar las funcionalidades del sistema para su evaluación de acuerdo a los requerimientos solicitados.
Presentación y socialización de proyecto de Gestión de Informes Diagnósticos y Planes de Trabajo	Coordinación y supervisores de EPSUM	02/07/2018	Presentar a detalle las nuevas funcionalidades del sistema de EPSUM para la recopilación de datos, según la información generada por los equipos multidisciplinarios. Evaluar los conocimientos adquiridos en la capacitación a través de preguntas y la utilización del sistema.
Presentación y revisión y avalado por asesora de proyecto	MBA. Samy Eunice Pinto	16/07/2018	Evaluar las características y funcionalidad del sistema según lo establecido en el anteproyecto.

Fuente: elaboración propia.

Tabla 41. Planificación de capacitaciones

Actividad	Dirigida a	Fecha	Objetivos
Capacitación 1 Módulo de registro de Diagnósticos y Planes de Trabajo	Estudiantes de 1ra. Cohorte 2018	06/07/2018	Instruir sobre las principales funcionalidades de los módulos para registro de información a los equipos multidisciplinarios.
Capacitación 2 Módulo de registro de Diagnósticos y Planes de Trabajo	Estudiantes de 1ra. Cohorte 2018 (Región Las Verapaces)	12/07/2018	Instruir sobre las principales funcionalidades de los módulos para registro de información a los equipos multidisciplinarios de Región de las Verapaces.
Capacitación 3 Módulo de Gestión de Inventario de Activos Fijos	Coordinación y personal de Tesorería	31/07/2018	Exponer el módulo de gestión y control de inventario para el registro de activos dentro del sistema. Evaluar lo aprendido en la capacitación a través de preguntas y la utilización del sistema.
Capacitación 4 Módulo de registro de Diagnósticos y Planes de Trabajo	Estudiantes de 2da. Cohorte 2018	09/08/2018	Presentar e instruir sobre el uso de la plataforma de registro de información de diagnósticos y proyectos a los equipos multidisciplinarios de la segunda cohorte 2018 de EPSUM.

Fuente: elaboración propia.

La capacitación 1 indicada en la tabla 41, fue desarrollada en el Taller de Informe Final del Programa EPSUM, la misma fue impartida a los estudiantes distribuidos por región de la 1ra. Cohorte 2018. La capacitación 2 corresponde a la instrucción dada a 16 equipos que conforman la región de las Verapaces a solicitud de los supervisores de dicha región. La capacitación 4 se realizó como presentación durante el Taller de Inducción a los estudiantes que forman parte de la 2da. Cohorte 2018.

3.1.2. Certificación de capacitaciones

Puesto que las capacitaciones realizadas coinciden con los talleres desarrollados por el Programa EPSUM, se presentan los registros de asistencia recabados por el personal de logística en el Anexo 3, a efecto de comprobación del cuórum convocado.

3.2. Documentación de la solución

3.2.1. Guía de usuario para registro de diagnósticos y planes de trabajo

El manual de registro de informes muestra paso a paso el proceso de registro para el diagnóstico territorial y los planes de trabajo, especificando las funcionalidades del módulo como la edición y consulta de información; esta guía se adjunta en el Apéndice 3.

3.2.2. Guía de usuario de Geoportal del Programa EPSUM

La guía para uso del geoportal detalla de manera sencilla las funcionalidades que brinda éste para la consulta de información de los diagnósticos y proyectos realizados por los equipos multidisciplinarios por cohorte (registrados desde la 1ra. Cohorte 2018), este manual se presenta en el Apéndice 4.

3.2.3. Guía de uso para módulo de gestión de inventario

El manual para la gestión de inventario muestra al usuario el proceso para registro de activos fijos, presentando de manera gradual el registro de proveedores, bienes e ingresos en el sistema. Además, se detallan las funciones de mantenimiento, tales como la actualización y baja de datos registrados; este manual puede encontrarse en el Apéndice 5.

3.2.4. Manual de especificaciones técnicas y mantenimiento

El manual técnico está dirigido a desarrolladores que necesiten continuar trabajando sobre los módulos implementados. El mismo reúne todas las especificaciones para la correcta configuración y despliegue de la aplicación. La guía técnica se muestra en el Apéndice 6.

3.3. Aceptación del proyecto

La certificación de entrega y aceptación del proyecto se refiere a un documento dirigido al coordinador de EPS de la carrera de Ingeniería en Ciencias y Sistemas, el mismo se encuentra en el Anexo 1.

Asimismo, se presenta en el Anexo 2 el finiquito otorgado por el Programa EPSUM como constancia de la finalización del EPS y el cumplimiento de los objetivos propuestos según la metodología del Programa.

CONCLUSIONES

1. Se elaboró el módulo para el registro de información de diagnósticos y proyectos, de acuerdo a las normas establecidas en la Guía Metodológica para Equipos Multidisciplinarios del Programa EPSUM.
2. Se desarrolló el sistema de consulta de información georreferenciada con los datos de diagnósticos y perfiles de proyectos registrados por los estudiantes que integran los equipos multidisciplinarios, con la finalidad de proyectar el trabajo realizado desde el Programa EPSUM.
3. Se implementó el módulo para la gestión y control de inventario de activos fijos del Programa EPSUM, que permite la centralización de la información de las tesorerías y la generación de reportes según el Reglamento para el registro y control de bienes muebles y otros activos fijos de la Universidad de San Carlos de Guatemala.
4. Mediante talleres, se capacitó al 90% de los supervisores y al 100% de los estudiantes inscritos en la primera cohorte del año 2018 que hicieron uso del módulo de registro de diagnósticos y proyectos; así mismo, se capacitó al 100% del personal administrativo, para el uso del módulo de gestión de inventario de activos fijos del Programa EPSUM.

RECOMENDACIONES

1. Es necesario verificar de manera constante los datos registrados por los equipos multidisciplinarios, a fin de garantizar que la información esté validada por los supervisores del Programa EPSUM y tomando en consideración que la información registrada es de carácter público.
2. Es conveniente tomar en cuenta la retroalimentación de los principales usuarios para generar nuevas especificaciones que mejoren la funcionalidad de los módulos desarrollados, dada la propiedad de escalabilidad de software con la que cuentan los mismos y que permite la agregación de nuevas características.
3. Se sugiere considerar la contratación de digitalizadores de datos para alimentar la información del módulo de gestión de inventario de activos fijos, debido a la extensa información histórica contenida en los libros de inventario con los que cuenta el Programa EPSUM, obteniendo así una integración total de la información.
4. Se propone realizar un estudio basado en el análisis estadístico de los datos, haciendo uso de tecnologías que apliquen algoritmos de inteligencia artificial para generar un módulo de predicciones a partir de la información registrada, puesto que los módulos elaborados representan un banco de datos con información actualizada acerca de los territorios cubiertos por el Programa EPSUM.

REFERENCIAS BIBLIOGRÁFICAS

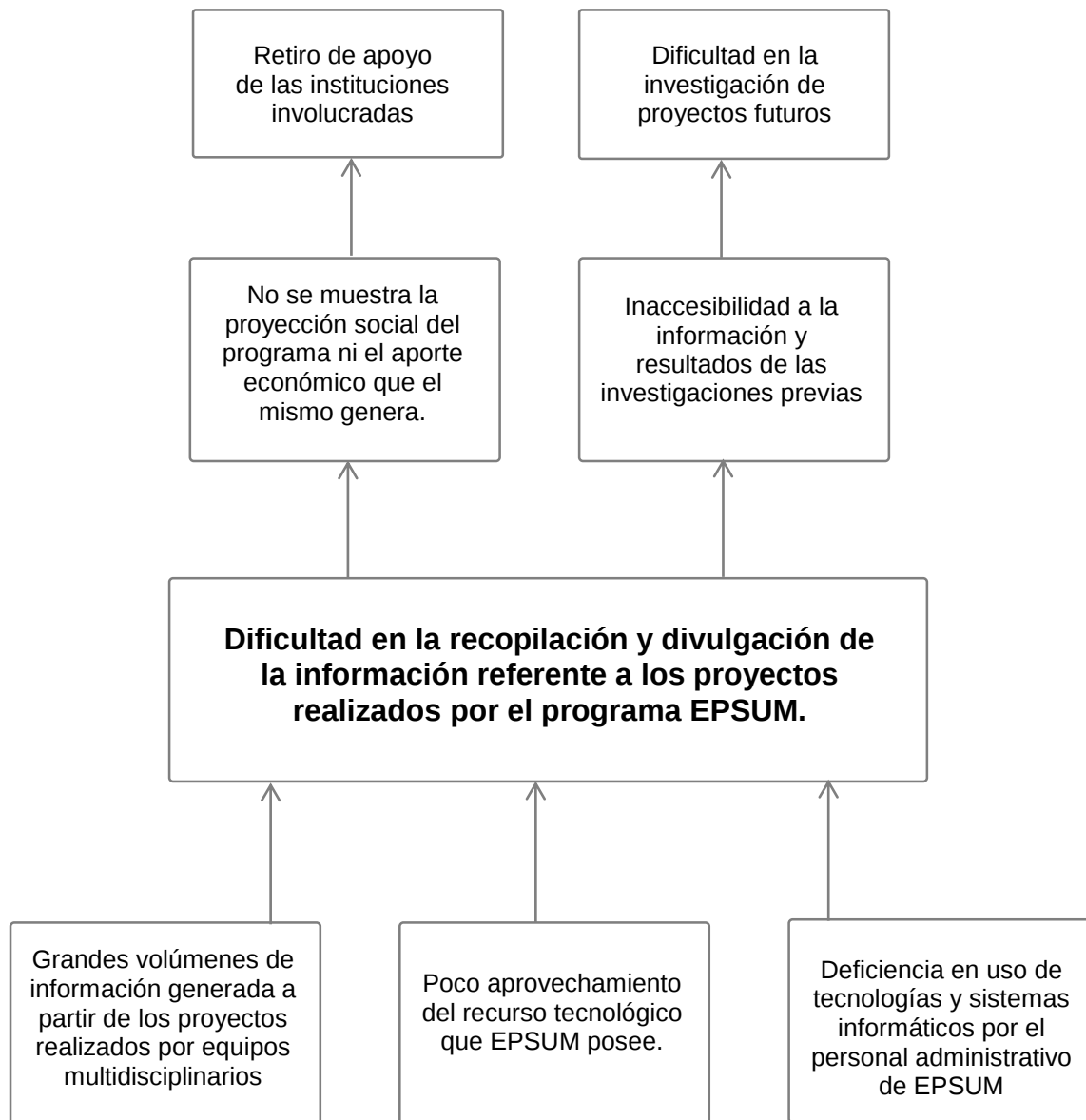
1. ArcGIS Resources (s.f). *Introducción a SIG*. Recuperado el 16 de noviembre de 2017 en <http://resources.arcgis.com/es/help/getting-started/articles/026n0000000t000000.htm>
2. EPSUM (2017). *Guía Metodológica para equipos multidisciplinarios*. Guatemala, Guatemala: Programa EPSUM.
3. Gallego Sánchez, A. (2015). *Laravel 5, the PHP framework for web artisans*. Recuperado el 20 de enero de 2018 en <https://www.gitbook.com/book/ajgallego/laravel-5/details>
4. García-Caro, I. (2006). *Aplicación web para el conocimiento y conversión de unidades*. Recuperado el 20 de enero de 2018 en http://meteo.ieec.uned.es/www_Usumeteo2/Memoria/Capitulo3.pdf
5. IBM (s.f.). *Rational Unified Process: Best Practices for Software Development Teams*. Recuperado el 15 de noviembre de 2017 en https://www.ibm.com/developerworks/rational/library/content/03July/1000/1251/1251_bestpractices_TP026B.pdf
6. Mateu, C. (2004). *Desarrollo de Aplicaciones Web*. Barcelona, España: Eureka Media.
7. MDN Web Docs, Mozilla (2017). *Javascript*. Recuperado el 16 de noviembre de 2017 en <https://developer.mozilla.org/es/docs/Web/JavaScript>

8. MySQL (2017). *What is MySQL*. Recuperado el 15 de noviembre de 2017 en <https://dev.mysql.com/doc/refman/5.7/en/what-is-mysql.html>
9. PHP (2017). What is PHP. Recuperado el 15 de noviembre de 2017 en <http://php.net/manual/en/intro-what-is.php>
10. Programa EPSUM (2016). *Programa EPSUM y Convenio Interinstitucional USAC Gobierno de Guatemala para la Atención de la población vulnerable*. Recuperado el 21 de octubre de 2017 en <http://ingenieria.cunoc.usac.edu.gt/portal/articulos/d5e50b72d0c6ed418afccad77cfc3b37d75865d2.pdf>
11. Reglamento para el registro y control de bienes muebles y otros activos fijos de la Universidad de San Carlos de Guatemala. (Acuerdo de Consejo Superior Universitario, Pto. Séptimo, Inciso 7.1, Acta No. 21-2007 del 24 de octubre de 2007).
12. Shelly, G. y Rosenblatt, H. (2012). *Systems Analysis and Design*. Boston, USA: Cengage Learning.
13. Stauffer, M. (2016). *Laravel Up & Running*. California, USA: O'Reilly Media.

APÉNDICES

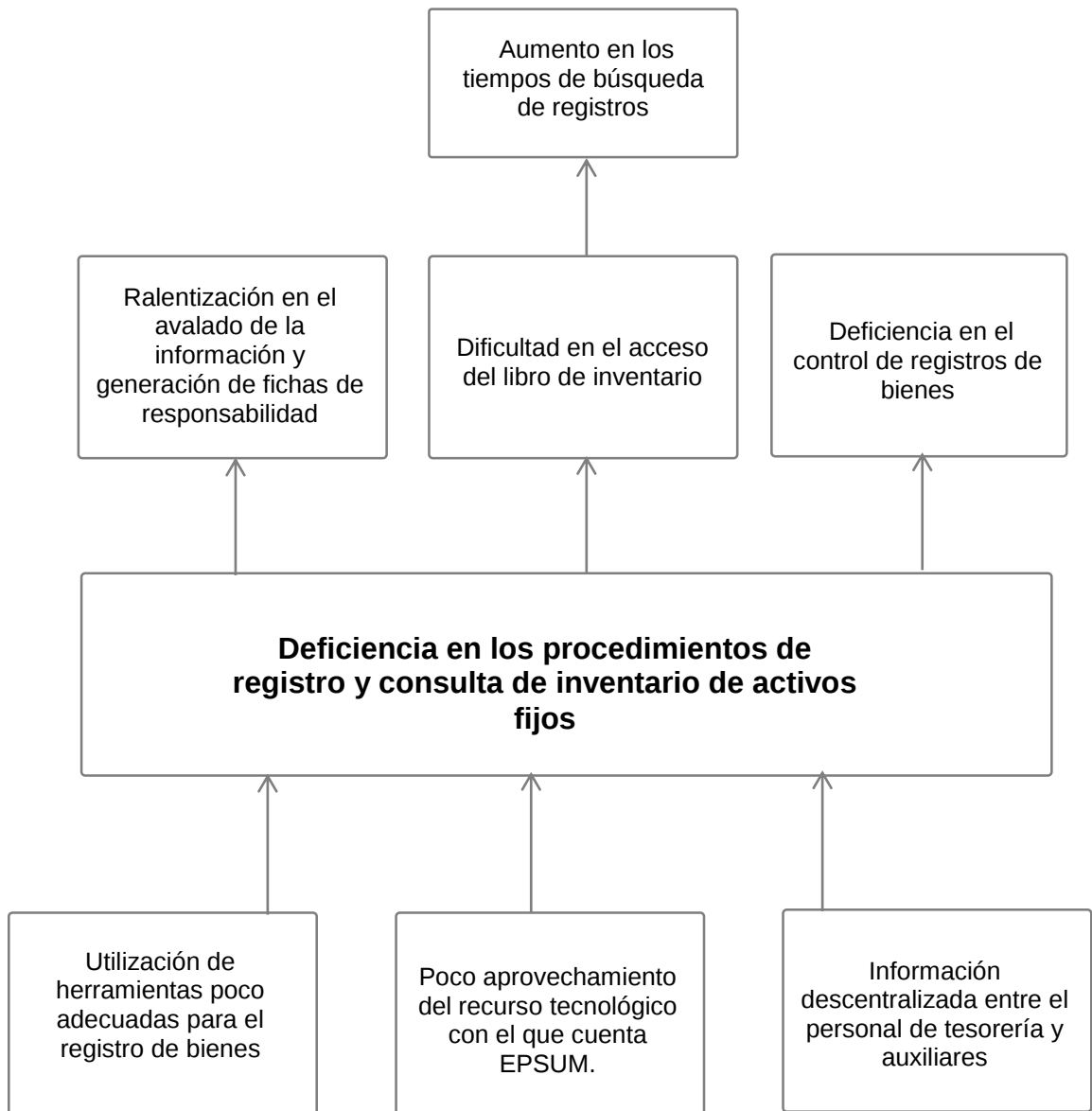
- 1. Árbol del problema: Desarrollo e implementación de un módulo de divulgación y organización geográfica de información de proyectos referentes al programa EPSUM.**
- 2. Árbol del problema: Desarrollo de un módulo para la gestión y control de inventario para la documentación del libro de activos fijos del Programa EPSUM.**
- 3. Guía de usuario de registro de informes diagnósticos y de información referente a proyectos.**
- 4. Guía de usuario de Geoportal de EPSUM**
- 5. Guía de usuario de sistema de gestión de inventario**
- 6. Manual de especificaciones técnicas y de mantenimiento del proyecto**
- 7. Contenido del CD**

Apéndice 1. Árbol del Problema: Desarrollo e implementación de un módulo de divulgación y organización geográfica de información de proyectos referentes al programa EPSUM.



Fuente: elaboración propia.

Apéndice 2. Árbol del Problema: Desarrollo de un módulo para la gestión y control de inventario para la documentación del libro de activos fijos del Programa EPSUM.



Fuente: elaboración propia.

Apéndice 3. Guía de usuario de registro de informes diagnósticos y de información referente a proyectos



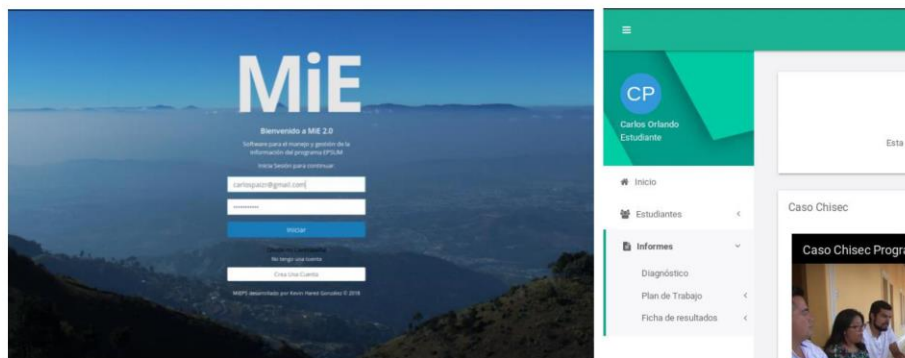
Guía de registro de Informes
diagnósticos, proyectos y planes
de trabajo en la Plataforma
EPSUM.

Estudiantes

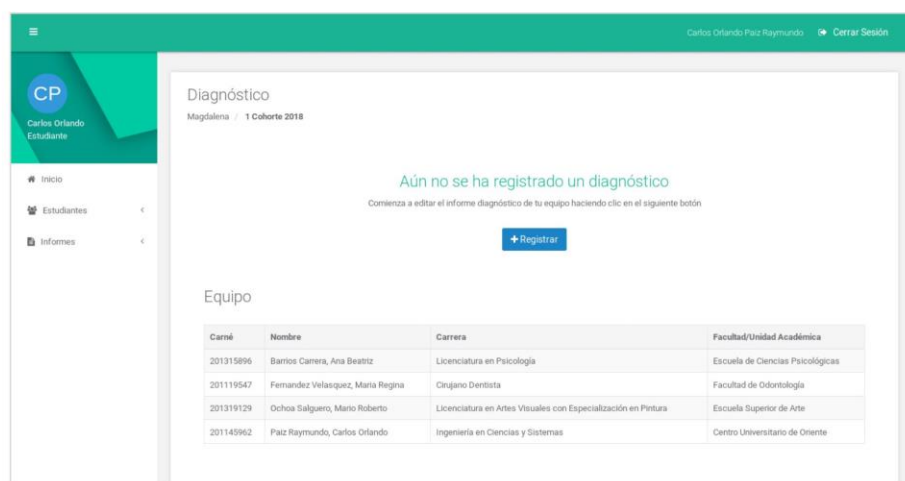
Registro de Diagnóstico

El informe diagnóstico en la plataforma de EPSUM constituye un formulario para el registro y procesamiento de información referente a los distintos territorios sedes en los cuales se enfoca el Programa. A continuación, se muestra una guía para el registro de la información en el sistema.

Para proceder con el registro antes mencionado, primero debes estar registrado como estudiante. Puedes iniciar sesión desde la opción en la página del Programa EPSUM epsum.usac.edu.gt o directamente en mieps.usac.edu.gt/public/login



Una vez iniciada la sesión deberás dirigirte al menú lateral izquierdo y en el apartado de **Informes**, deberás seleccionar la opción **Diagnóstico**. Si ningún integrante del equipo ha comenzado el registro del diagnóstico, verás una pantalla que te indica que puedes proceder a ingresar la información haciendo clic en la opción **Registrar**.



Preliminar

La información solicitada para el registro se divide en 7 apartados divididos de la siguiente manera:

- Información general
- Territorio
- Demografía
- Economía
- Salud
- Educación
- Medio Ambiente
- Gestión Municipal

La información será obtenida del documento de informe diagnóstico solicitado según la metodología del Programa EPSUM.

En cada apartado tendremos 2 opciones: **Guardar** o **Seguir editando más tarde**. La primera nos permitirá guardar la información que hayamos ingresado y la segunda opción servirá únicamente para seguir editando más tarde o darle la oportunidad a otro integrante del equipo para que colabore en el registro del diagnóstico.

¿Quiénes editan el diagnóstico?

Todo el equipo puede editar el informe diagnóstico. Esto es posible seleccionando la opción de **Seguir editando más tarde**. Al hacer clic en dicha opción, se mostrará la pantalla principal con el resumen del diagnóstico indicando el avance del registro del mismo. En la esquina superior derecha se encontrará la opción **Editar** que permitirá seguir editando el diagnóstico.

Registros

Información general

En este apartado se deberá ingresar una breve introducción acerca del diagnóstico y el objetivo del mismo, por lo que no se requiere mayor información.

Diagnóstico Magdalena

Registro

Completado: 0%

Introducción

Ingresa una breve introducción para el diagnóstico...

Objetivo General

Objetivo general del diagnóstico.

Debes llenar este campo

Guardar Seguir editando más tarde

Territorio intervenido

Campo	Definición
Lugar	Este campo se refiere al territorio donde se realiza el diagnóstico. Se completa automáticamente.
Ubicación	Campo descriptivo acerca de la locación donde se realiza el diagnóstico.
Metros sobre el nivel del mar	Campo numérico en donde se indica los MSNM del territorio intervenido. No es necesario colocar la dimensional de metros (m).
Superficie	Campo numérico donde se indica la superficie territorial en km ² . No es necesario colocar la dimensional de kilómetros cuadrados.

Diagnóstico Magdalena

1
Territorio intervenido

Completado: 0%

Lugar
Magdalena Milpas Altas, Sacatepéquez

Ubicación
Magdalena Milpas Altas, es un municipio del departamento Sacatepéquez. Catalogado como Municipalidad de 4ª. Categoría. Nombre geográfico oficial: Magdalena Milpas Altas, se ubica 12 kilómetros de la cabecera departamental y a 38 kilómetros de la ciudad capital

Metros sobre el nivel del mar
2045

Superficie
8

Guardar
Se
Undo
Cut
Copy

Demografía

Campo	Definición
Descripción	Campo descriptivo acerca de la demografía del lugar.
Población 0-14 años	Campo numérico en donde se indica la población comprendida entre 0 y 14 años de edad.
Población 15-64 años	Campo numérico en donde se indica la población comprendida entre 15 y 64 años de edad.
Población 65 años o más	Campo numérico en donde se indica la población de 65 o más años de edad.
Población mujeres	Campo numérico que se refiere a la población de mujeres
Población hombres	Campo numérico referente a la población de hombres en el territorio
Población rural	Campo numérico en donde se indica la cantidad de personas que conforman la población rural del territorio intervenido
Población urbana	Campo numérico en donde se indica la cantidad de personas que conforman la población urbana del territorio intervenido

Población indígena	Campo numérico en donde se indica la cantidad de personas que conforman la población indígena del territorio intervenido
Población garífuna	Campo numérico en donde se indica la cantidad de personas que conforman la población garífuna del territorio intervenido. No requerido.
Población xinca	Campo numérico en donde se indica la cantidad de personas que conforman la población xinca del territorio intervenido. No requerido.

2
Demografía

Completado: 14%

Descripción
Breve descripción acerca de la población del lugar

Población (0-14 años)
2983

Población (15-64 años)
6641

Población (65+ años)
562

Población mujeres
5021

Población hombres
5198

Población rural
6250

Población urbana
10598

Población indígena
7579

Población garífuna
0

Población xinca
0

Guardar
Seguir editando más tarde

Economía

Campo	Definición
Descripción	Campo descriptivo acerca de la economía del lugar.
Pobreza	Campo numérico que se refiere al porcentaje de pobreza en el territorio intervenido. No es necesario ingresar símbolo de porcentaje.

Pobreza extrema	Campo numérico que se refiere al porcentaje de pobreza extrema en el territorio intervenido. No es necesario ingresar símbolo de porcentaje
Ingreso por familia	Campo numérico en donde se indica la cantidad en quetzales de ingreso por familia. No es necesario ingresar símbolo de moneda (Q.)
ENBI	¿Se utilizó Encuesta de Necesidades Básicas Insatisfechas? Seleccionar entre Sí o No según sea el caso. Si la respuesta es sí, ingresar algunas observaciones, de lo contrario, no es necesario hacer observaciones.

Diagnóstico Magdalena

3 Economía

Completado: 25%

Descripción
El municipio se ha desarrollado agricolamente y ha tecnificado la producción de hortalizas, sin embargo existen otras actividades que potencialmente son generadoras de ingresos para la población.

Pobreza
33.56

Pobreza extrema
3.90

Ingreso por familia
0.00

¿Se utilizó Encuesta de Necesidades Básicas Insatisfechas?
☐ Sí
☒ No

Observaciones
Indicar si se usó ENBI para realizar las mediciones

Guardar **Seguir editando más tarde**

Context menu options: Congress, congress, Add to Dictionary, Undo, Cut, Copy, Paste, Delete, Select All, Check Spelling, Languages

Salud

Campo	Definición
Atención primaria	Campo numérico que se refiere a la tasa de acceso de atención primaria en salud en el territorio intervenido.

Enfermedades vinculadas al consumo de agua contaminada	Campo numérico en donde se indica la tasa de enfermedades vinculadas al consumo de agua no apta y mala disposición de excretas.
Desnutrición crónica	Campo numérico en donde se indica la tasa de desnutrición crónica en niños menores de 5 años
Enfermedades frecuentes en niños menores de 5 años	Seleccionar de la lista las enfermedades encontradas en el territorio intervenido. En el caso de que no se encuentre en la lista, ingresarla en el cuadro de texto que se encuentra al final de la lista.

Diagnóstico Magdalena

4 Salud

Completado: 42%

Atención primaria: 40

Enfermedades vinculada al consumo de agua contaminada: 0

Desnutrición crónica: 44.8

Enfermedades frecuentes en niños menores de 5 años
 Seleccionar las enfermedades identificadas en el lugar

- ☒ Infecciones respiratorias
- ☐ Síndrome diarreico agudo
- ☐ Parasitismo intestinal
- ☐ Infección urinaria o ITU
- ☐ Dermatitis
- ☐ Otitis media aguda
- ☒ Bronquitis aguda
- ☐ Enfermedades transmitidas por vectores (Ej. Malaria, Zika, Dengue, etc.)
- ☐ Desnutrición aguda moderada o severa

Educación

Campo	Definición
Aprobación de pruebas de matemática	
Nivel primario	Campo numérico en donde se indica el porcentaje de aprobación de pruebas de matemática en el nivel primario.

Nivel secundario	Campo numérico en donde se indica el porcentaje de aprobación de pruebas de matemática en el nivel secundario.
Nivel diversificado	Campo numérico en donde se indica el porcentaje de aprobación de pruebas de matemática en el diversificado.
Aprobación de pruebas de lectura	
Nivel primario	Campo numérico en donde se indica el porcentaje de aprobación de pruebas de lectura en el nivel primario.
Nivel secundario	Campo numérico en donde se indica el porcentaje de aprobación de pruebas de lectura en el nivel secundario.
Nivel diversificado	Campo numérico en donde se indica el porcentaje de aprobación de pruebas de lectura en el nivel diversificado.

Diagnóstico Magdalena

5 Educación

Completado: 55%

Aprobación de pruebas de matemática

Primaria Porcentaje de aprobación de prueba de matemática para el nivel primario

Secundaria Porcentaje de aprobación de prueba de matemática en el nivel secundario

Diversificado Porcentaje de aprobación de prueba de matemática en nivel diversificado

Aprobación de pruebas de lectura

Primaria Porcentaje de aprobación de pruebas de lectura para el nivel primario

Secundaria Porcentaje de aprobación de pruebas de lectura en el nivel secundario

Diversificado Porcentaje de aprobación de pruebas de lectura en nivel diversificado

Guardar Seguir editando más tarde

Ambiente

Campo	Definición
Cobertura forestal	Campo numérico referente al porcentaje de cobertura forestal del territorio intervenido

Planes de gestión integral	Campo numérico en donde se indica el número de planes de gestión integral en el territorio intervenido.
Uso de la tierra	Campo numérico que se refiere al porcentaje de uso de la tierra para tareas agrícolas.

Diagnóstico Magdalena

6 Ambiente

Completado 70%

Cobertura forestal 35.00

Planes de gestión integral 5

Uso de la tierra 25.00

Guardar Seguir editando más tarde

?

Gestión municipal

Campo	Definición
Índice de gestión municipal	Campo numérico en donde se ingresa el índice de gestión municipal del territorio intervenido.

Diagnóstico Magdalena

7 Gestión Municipal

Completado 84%

Índice de gestión municipal 0.49

0.5

Finalizar

?

Una vez ingresados la información hacer clic en **Finalizar**. Se mostrará la pantalla con el resumen del diagnóstico.

Editar información de diagnóstico

Para editar información del diagnóstico hacer clic en el botón **Editar** que se encuentra en la esquina superior derecha de la vista principal de diagnóstico.

Diagnóstico

Magdalena / 1 Cohorte 2018

Estado **Finalizado**

Equipo:

Magdalena

Cohorte:

1 Cohorte 2018

Municipio:

Magdalena Milpas Altas

Departamento:

Sacatepéquez

Creado por:

Carlos Orlando Paiz

Modificado por:

Carlos Orlando Paiz

Última actualización:

2018-07-20 12:38:44

Participantes:

AB MF MC CP

Completado:

Diagnóstico completado al 100%. Clic en el botón Editar, para modificar la información.

Editar

Subir

Equipo

Carné	Nombre	Carrera	Facultad/Unidad Académica
201315896	Barrios Carrera, Ana Beatriz	Licenciatura en Psicología	Escuela de Ciencias Psicológicas
201119547	Fernandez Velasquez, Maria Regina	Cirujano Dentista	Facultad de Odontología
201319129	Ochoa Salguero, Mario Roberto	Licenciatura en Artes Visuales con Especialización en Pintura	Escuela Superior de Arte
201145962	Paiz Raymundo, Carlos Orlando	Ingeniería en Ciencias y Sistemas	Centro Universitario de Oriente

Al seleccionar dicha opción, se mostrará la vista de edición con la información previamente guardada. En la parte inferior aparecerán tres opciones: **Actualizar**, que permite guardar los cambios que se realicen; **Siguiente**, que pasa al siguiente apartado del diagnóstico sin modificar la información del apartado actual; **Finalizar**, permite salir de la vista de edición.

Editar Diagnóstico Magdalena

General

Introducción

Magdalena Milpas Altas un pueblo con riqueza cultural y agrícola fue fundado en el año 1585 y se constituyó como municipio en 1825, año en el que se celebraron las primeras elecciones populares. El municipio cuenta con la mayoría de los servicios públicos, un mercado municipal, servicio de transporte para la cabecera departamental y Ciudad Capital, oficinas de correo y canchas deportivas, no obstante, cuenta únicamente con un centro de salud para cubrir las necesidades básicas de la población en general.

Objetivo General

Conocer las principales necesidades dentro del municipio para aportar al desarrollo del mismo a través del diseño y ejecución del proyecto multidisciplinario en Magdalena Milpas Altas

Actualizar

Siguiente

Finalizar

Registro de Proyectos

El sistema permitirá el registro de los proyectos Multidisciplinario, Monodisciplinario y de Convivencia según la metodología del Programa EPSUM. La siguiente guía muestra el registro de un proyecto multidisciplinario, el mismo procedimiento es aplicable para los proyectos monodisciplinario y de convivencia.

Preliminar

El registro de proyectos está dividido en dos partes: Información general del proyecto y Plan de trabajo. En el siguiente diagrama se muestra el orden en que se ingresa la información de proyectos. El paso 0 se refiere a la información general para el registro del proyecto y los pasos del 1 al 3 corresponden al plan de trabajo. Los pasos 1, 2 y 3 son repetitivos y se realizarán por cada uno de los objetivos específicos planteados para el proyecto.



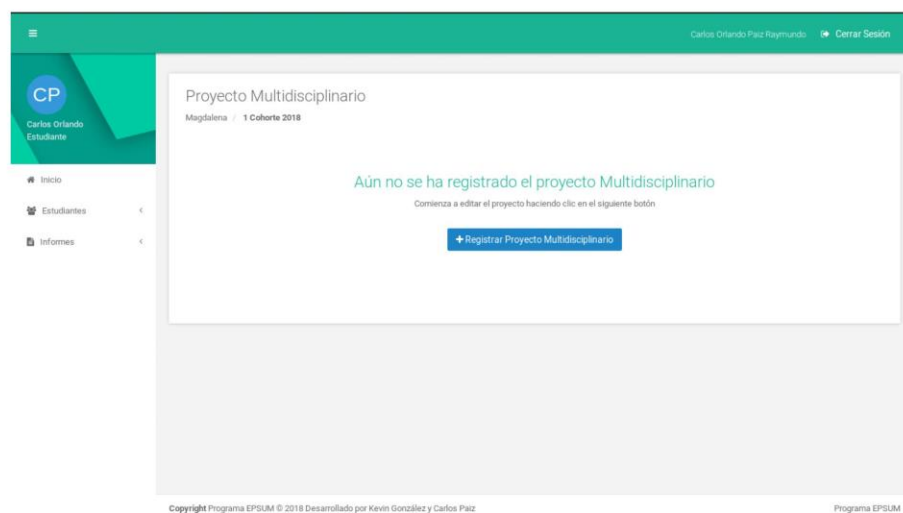
La información requerida puede ser extraída del documento de perfil de proyecto y plan de trabajo realizados con anterioridad.

¿Quiénes editan los proyectos?

A excepción del proyecto monodisciplinario, los proyectos pueden ser registrados por cualquier miembro del equipo.

Registro de información general de proyecto

Para iniciar con el registro de proyectos, dirigirse al menú lateral izquierdo y seleccionar la opción **Informes**. Posteriormente, seleccionar entre las opciones **Proyecto Multidisciplinario**, **Proyecto Monodisciplinario** o **Proyecto de Convivencia**, según se requiera.



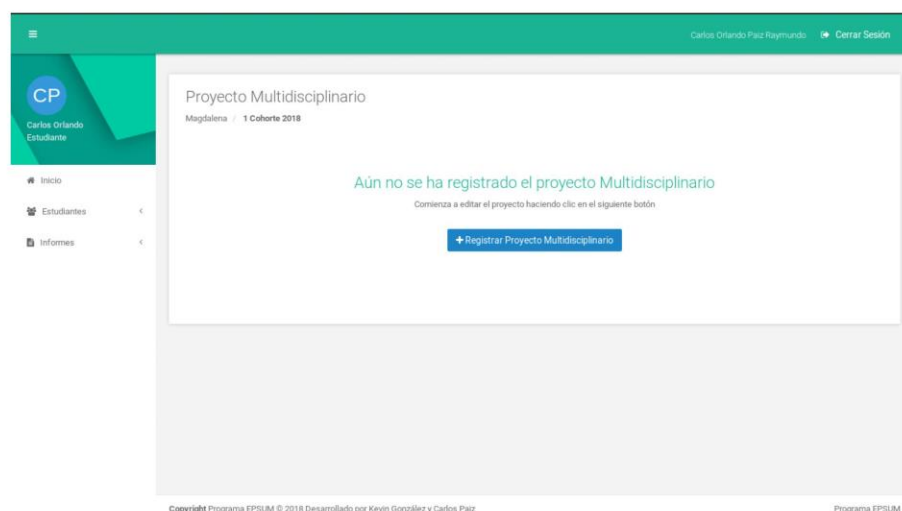
Hecho esto, aparecerá una vista indicando que aún no se ha registrado el proyecto seleccionado. Hacer clic en el botón **Registrar** para iniciar.

Perfil del proyecto

Campo	Definición
Nombre	Campo descriptivo para el nombre del proyecto.
Ubicación	Campo descriptivo que se refiere a la ubicación específica donde se realiza el proyecto
Descripción	Campo descriptivo acerca de algunas generalidades del proyecto
Objetivo General	Campo que guarda el objetivo general del proyecto
Justificación	Campo descriptivo que requiere la justificación del proyecto.

Registro de información general de proyecto

Para iniciar con el registro de proyectos, dirigirse al menú lateral izquierdo y seleccionar la opción **Informes**. Posteriormente, seleccionar entre las opciones **Proyecto Multidisciplinario**, **Proyecto Monodisciplinario** o **Proyecto de Convivencia**, según se requiera.



Hecho esto, aparecerá una vista indicando que aún no se ha registrado el proyecto seleccionado. Hacer clic en el botón **Registrar** para iniciar.

Perfil del proyecto

Campo	Definición
Nombre	Campo descriptivo para el nombre del proyecto.
Ubicación	Campo descriptivo que se refiere a la ubicación específica donde se realiza el proyecto
Descripción	Campo descriptivo acerca de algunas generalidades del proyecto
Objetivo General	Campo que guarda el objetivo general del proyecto
Justificación	Campo descriptivo que requiere la justificación del proyecto.

Línea de intervención	Selección de Línea de intervención de acuerdo al proyecto a realizar.
Descripción de línea de intervención	Campo que muestra la descripción de la línea de intervención seleccionada. Este campo se completa automáticamente
Área	Campo que muestra el área de la línea de intervención seleccionada. Este campo se completa automáticamente
Políticas asociadas	Campo que muestra las políticas asociadas a la línea de intervención seleccionada. Este campo se completa automáticamente.
Fecha de inicio de ejecución del proyecto	Campo de fecha que requiere la fecha de inicio del proyecto en formato dd/mm/YYYY
Fecha de finalización del proyecto	Campo de fecha que requiere la fecha de finalización del proyecto en formato dd/mm/YYYY
Actores involucrados	En este campo se ingresan los nombre de personas o instituciones involucradas en el proyecto, separadas por comas.
Valor económico	Campo numérico en donde se ingresa el valor económico del proyecto. No es necesario ingresar el símbolo de moneda.

The screenshot shows a web application interface for a project registration system. The header is green with a user profile 'CP Carlos Orlando Estudiante' and a 'Cerrar Sesión' button. The left sidebar contains navigation links: 'Inicio', 'Estudiantes', and 'Informes'. The main content area is titled 'Proyecto / Magdalena' and 'Registro de Proyecto Multidisciplinario'. It contains a form with the following fields:

- Nombre:** Promoción de Excelencia Educativa en la Escuela HUNAPU de Magdalena Milpas Altas.
- Ubicación:** Escuela Oficial Urbana Mixta HUNAPU, J.M. de Magdalena Milpas Altas, Sacatepequez
- Descripción:** El proyecto busca aumentar el nivel de aprobación en las pruebas de matemáticas y lectura de los niños de la Escuela Oficial Urbana Mixta HUNAPU J.M de Magdalena M.A. a través de reforzamiento en el aprendizaje de dichas áreas.
- Objetivo General:** Aumentar el nivel de aprobación en las pruebas de matemáticas y lectura de los niños de la Escuela Oficial Urbana Mixta HUNAPU J.M de Magdalena M.A.
- Justificación:** Justificación del proyecto
- Línea de Intervención:** — Seleccionar línea de intervención —

Al finalizar, hacer clic en el botón **Guardar** para registrar la información.

Plan de trabajo

Una vez ingresado el perfil del proyecto se mostrará una pantalla que solicitará el plan de trabajo. Esta vista está dividida en tres partes:

1. **Objetivo específico**
2. **Indicador**
3. **Actividades**

Estos pasos deberán repetirse con cada uno de los objetivos específicos planteados para el proyecto, al cual corresponde un indicador y una serie de actividades que permitan la consecución del objetivo y se realiza de la siguiente manera:

Ingresamos el primer objetivo específico, la meta propuesta para el objetivo **(1)** y posteriormente el indicador **(2)** para medir el grado de avance del objetivo.

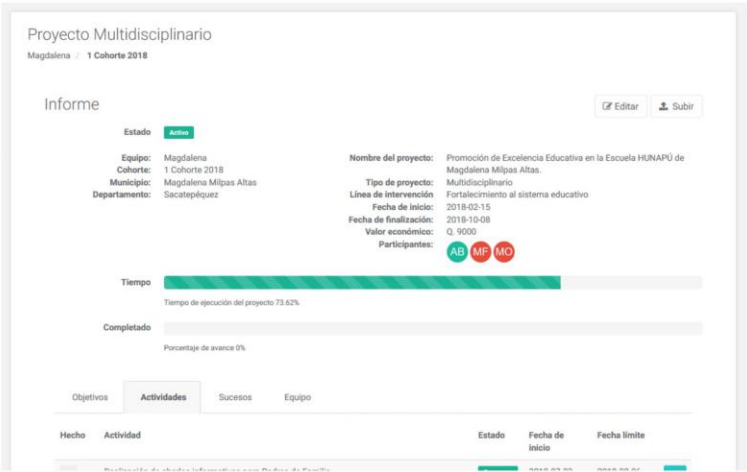
Completada la información solicitada, hacemos clic en el botón **Agregar actividades**. Nos aparecerá un formulario para ingresar las actividades **(3)** asociadas al objetivo, en la cual completaremos con una descripción la actividad y las fechas de inicio y finalización para la misma. Una vez hemos llenado estos campos hacemos clic en el botón **Agregar actividad** para registrarla. Nos aparecerá el detalle en la tabla y procedemos a agregar la siguiente actividad. Se podrán registrar tantas actividades como se consideren necesarias para poder conseguir el objetivo.

Al terminar de ingresar las actividades, tenemos dos opciones: **Registrar otro objetivo** o **Finalizar**. La primera opción nos permitirá ingresar el siguiente objetivo específico y realizar el procedimiento previamente descrito, tantas veces como objetivos específicos se hayan definido. La segunda opción permite finalizar el proceso y nos lleva al resumen del proyecto registrado.

Vista de detalle del proyecto

En la vista de resumen del proyecto nos aparecerán algunos datos generales del proyecto y veremos 2 barras de progreso. La barra superior (**Tiempo**) nos indicará el porcentaje de avance

del tiempo a la fecha, según el periodo de tiempo designado al proyecto. La barra inferior (**Completado**) nos indicará el porcentaje de avance del proyecto, según se vayan realizando las actividades (marcándolas como hechas).



En esta vista tendremos 4 pestañas con información relevante del proyecto:

Pestaña	Definición
Objetivos	En esta pestaña se mostrarán los objetivos que se hayan registrado, sus metas e indicadores definidos.
Actividades	Esta pestaña es la principal y en ella encontraremos todas las actividades registradas para el proyecto, las mismas aparecerán por orden de prioridad según las fechas definidas para las mismas.
Sucesos	Aquí se mostrará cada uno de los sucesos realizados por el usuario, tales como creación de actividades y cambio de estado de las mismas (marcar o desmarcar como hechas)
Equipo	Esta pestaña aparece únicamente en el detalle del proyecto multidisciplinario y de convivencia.

Agregar más objetivos

Para agregar más objetivos luego de haber terminado con la pantalla de registro de proyecto, ir a la derecha de las pestañas y seleccionar la opción agregar un objetivo. Nos aparecerá una ventana donde se mostrará un formulario solicitando la información del objetivo como sigue:



Una vez completados los campos, hacemos clic en el botón **Agregar objetivo**. Se limpiarán los campos y nos permitirá registrar otro objetivo si fuera necesario, de lo contrario hacemos clic en el botón **Terminar**.

Agregar más actividades

Si fuera necesario agregar más actividades, seleccionar la opción **Agregar actividad** (al lado derecho de las pestañas). Hecho esto aparecerá una ventana con un formulario en el cual deberemos completar con la información solicitada.

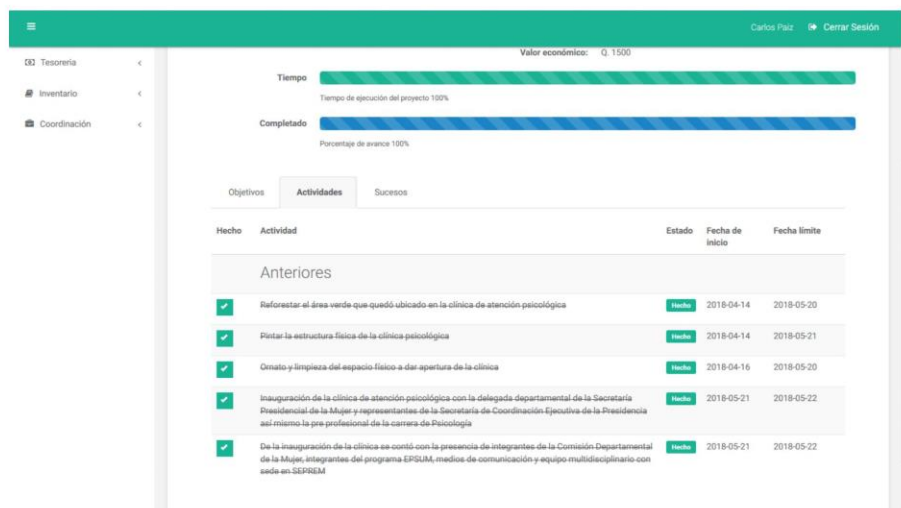


Recordar que las actividades están asociadas a un objetivo, por lo que primero que debemos hacer es seleccionar uno de la lista de objetivos. Si se tuviera que agregar un nuevo objetivo, ver el título anterior. Una vez seleccionado el objetivo, completar los campos indicando una descripción y seleccionando las fechas correctas. Hecho esto hacemos clic en el botón **Agregar actividad**. Se cargará la actividad a la tabla y se limpiarán los campos para agregar una nueva actividad si fuera necesario, de lo contrario hacemos clic en **Terminar**.

Registrar el avance del proyecto

Para poder registrar el avance del proyecto nos dirigimos a la pestaña de actividades y para el efecto simplemente seleccionamos las casillas de verificación que se encuentran al lado de la descripción de las actividades. Veremos como la barra de progreso de ejecución del proyecto (Completado) marcará un porcentaje de avance. El avance se mide según el número de actividades que se vayan marcando como hechas entre el total de actividades registradas.





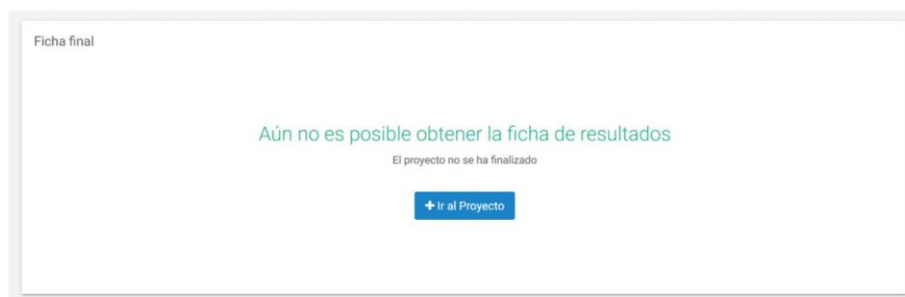
Tomar en cuenta que al momento de marcar una actividad como hecha o desmarcarla, se guarda un registro de quien realizó tal acción. Esto puede verse en la pestaña de sucesos.

Objetivos | Actividades | **Sucesos**

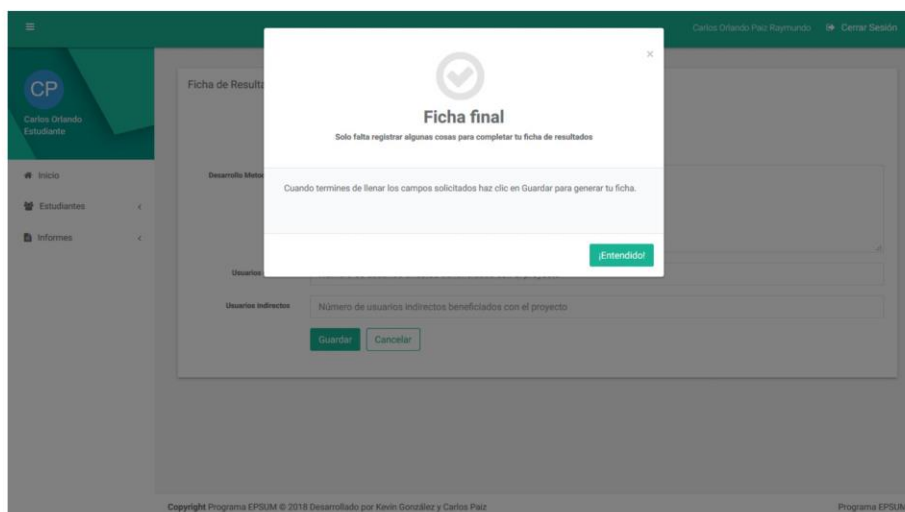
FA	Frederick Augusto Waldeimar marcó como hecha la actividad: Taller sobre el manejo y apoyo del plan de manejo para el cultivo de papa y entrega del mismo. RF:61168	2018-08-01 10:32:23
FA	Frederick Augusto Waldeimar marcó como hecha la actividad: Entrega de agroinsumos para el manejo fitosanitario del cultivo de papa RF:61168	2018-08-01 10:32:22
FA	Frederick Augusto Waldeimar marcó como hecha la actividad: Elaboración del plan de manejo y fitosanitario para cultivo de papa RF:61168	2018-08-01 10:32:21
FA	Frederick Augusto Waldeimar marcó como hecha la actividad: Establecimiento del cultivo de papa RF:61168	2018-08-01 10:32:19
FA	Frederick Augusto Waldeimar marcó como hecha la actividad: Entrega de diferentes materiales didácticos para el manejo de cultivo de papa y manejo de animales pecuarios. RF:61168	2018-08-01 10:32:18
FA	Frederick Augusto Waldeimar marcó como hecha la actividad: Entrega de bulbos de gallinaza en diferentes comunidades de la micro región 13. RF:61168	2018-08-01 10:32:14
FA	Frederick Augusto Waldeimar marcó como hecha la actividad: Taller para capacitación del cultivo de papa RF:61168	2018-08-01 10:32:14
FA	Frederick Augusto Waldeimar marcó como hecha la actividad: Gestión para compra de semilla de papa en Páquis, Chiantla, Huehuetenango. RF:61168	2018-08-01 10:32:11
FA	Frederick Augusto Waldeimar marcó como hecha la actividad: Entrega de semilla de papa de diferentes variedades del ICTA. RF:61168	2018-08-01 10:32:11

Generación de ficha de resultados

La ficha de resultados constituye un informe final que reúne la información previamente registrada por el usuario en el sistema. En ella se muestra un resumen del proyecto y un estimado de usuarios beneficiados. Para poder generarla tendremos que haber finalizado el proyecto del cual necesitemos generar ficha de resultados, esto significa que todas las actividades deberán haberse realizado (marcarse como hechas).



Para poder generar la ficha de resultados nos dirigimos al menú lateral derecho y en la opción **Informes** seleccionamos **Ficha de resultados**, luego procedemos a elegir el tipo de proyecto. Nos aparecerá una ventana indicando que solo debemos llenar unos cuantos campos más para generar la ficha.



Se mostrará un formulario donde rellenaremos con la siguiente información:

Campo	Definición
Desarrollo Metodológico	Campo descriptivo acerca de la metodología aplicada al proyecto
Usuarios directos	Campo numérico que se refiere al total de usuarios directos beneficiados con el proyecto.
Usuarios indirectos	Campo numérico que se refiere al total de usuarios indirectos beneficiados con el proyecto.

Una vez completados los campos hacemos clic en **Guardar** y se generará la ficha de resultados. Aparecerá una nueva vista y en la parte inferior encontraremos la opción **Generar Reporte**, al seleccionarla se procesará el documento y aparecerá una nueva vista con el documento PDF el cual podremos imprimir o guardar en el equipo.



Proyecto Multidisciplinario

Inicio: 2018-03-01, Finalización: 2018-07-15

Nombre	Fortalecimiento a la seguridad alimentaria con Invernadero comunitario de hortalizas
Ubicación	Aldea Las Cuevas de la microregión III del municipio de San Bartolomé Jocotenango, departamento de Quiché
Objetivo General	Propiciar la producción y diversidad de cultivos, con el fin de reducir la desnutrición en la comunidad de Las Cuevas, San Bartolomé Jocotenango.
Objetivos Específicos	• Desarrollar capacidades técnicas en los beneficiarios para la producción de hortalizas, así con el fin de reducir la desnutrición en la comunidad. • Organizar a los comunitarios que sean beneficiados con la producción de pilones, en la comunidad de Las Cuevas, San Bartolomé Jocotenango, Quiché.
Desarrollo Metodológico	El proceso metodológico abordado para la implementación del proyecto partió de la priorización de las comunidades a intervenir por la cohorte anterior del equipo EPSUM, seguidamente se aplicó la encuesta de necesidades básicas insatisfechas, posteriormente al análisis de la situación general y la dispición de recursos para la implementación del proyecto. Dicho proyecto se estableció en base al árbol de problemas encontrado en el municipio y en base al diagnóstico municipal elaborado. Una vez establecidas la línea de intervención a trabajar, se procedió a elaborar el perfil de proyecto y el plan de trabajo en grupo, posteriormente se resento a las autoridades municipales y comunitarias para su aprobación. Una vez aprobado se procedió a la gestión de recursos económicos, humano y didáctico para la implementación del proyecto. También se establecieron alianzas de trabajo con instituciones del área, seguidamente se empezó la etapa de elaboración de estructura del invernadero, capacitación de grupos y la siem
Línea de intervención	Agricultura familiar y mejoramiento de las economías campesinas
Área de intervención	Economía
Políticas asociadas	Política Nacional de Desarrollo Rural. Política Nacional para el Desarrollo de la micro, pequeña y mediana Empresa
Metas	• Capacitar a 42 familias de la comunidad • 35 mujeres capacitadas
Usuarios directos	35
Usuarios indirectos	299
Aporte económico técnico al proyecto	Q.39227.83

Generar Reporte

Volver

?

PROYECTO MULTIDISCIPLINARIO	
EQUIPO: SAN BARTOLOMÉ JOCOTENANGO, 1 COHORTE 2018 SAN BARTOLOMÉ JOCOTENANGO, QUICHÉ	
Nombre	Fortalecimiento a la seguridad alimentaria con Invernadero comunitario de hortalizas
Ubicación	Aldea Las Cuevas de la microregión III del municipio de San Bartolomé Jocotenango, departamento de Quiché

¡Enhorabuena!

Esto es todo lo que necesitas saber para poder utilizar el sistema sin problemas. Siéntete libre de contactar a los desarrolladores en caso de que se presente algún inconveniente con el uso de esta plataforma.

El equipo de desarrolladores de EPSUM.

Apéndice 4. **Guía de usuario de Geoportal de EPSUM**



Guía de uso del Geoportal del
Programa EPSUM.

mieps.usac.edu.gt

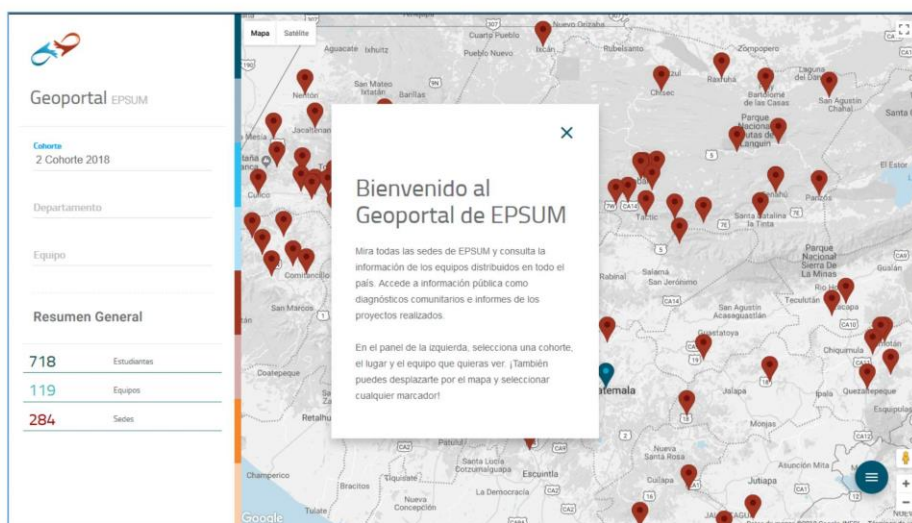
Geoportal de EPSUM

El Geoportal del Programa EPSUM constituye una herramienta para consulta de información georreferenciada referente a los equipos multidisciplinarios distribuidos en todo el país. El sistema está enfocado en la divulgación y ordenamiento de la información generada por los estudiantes de las distintas disciplinas, con el fin de apoyar la proyección social del programa y el aporte económico que el mismo genera.

Vista General

Escritorio, Laptop o iPad

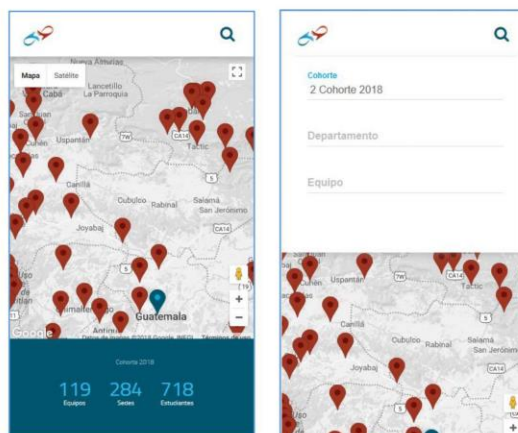
Al momento de iniciar la aplicación web, se distinguen un panel lateral a la izquierda con opciones de menú y a la derecha se muestra la vista del mapa. Adicional a esto, existe un panel de resultados que se muestra únicamente cuando se necesita.



Móvil

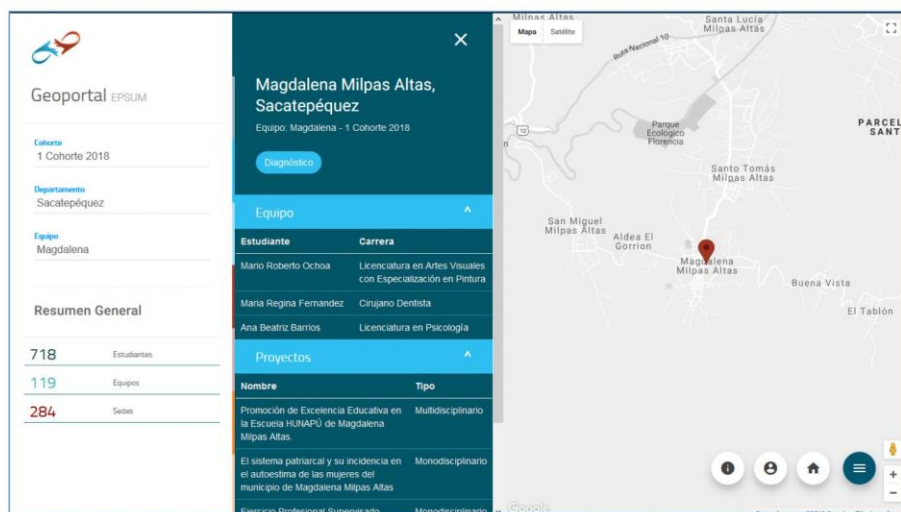
La aplicación móvil del geoportal ha sido adaptada de manera que la misma brinde una mejor experiencia al usuario desde cualquier dispositivo. Ésta cuenta con una barra en la parte superior

desde la cual se accede al menú y el resto lo constituye la vista del mapa. El panel de resultados, al igual que la vista de escritorio, se muestra únicamente cuando se necesita.



Realizar una búsqueda

Para comenzar con una búsqueda desde un equipo de escritorio, laptop o iPad, nos dirigimos al panel lateral izquierdo y seleccionamos la cohorte para la cual se realizará la búsqueda. Posteriormente elegimos el departamento que necesitamos consultar y por último seleccionamos uno de los equipos filtrados para ese departamento. Una vez hemos seleccionado el equipo, se desplegará el panel de resultados (color azul) en donde se mostrará la información del equipo y los productos que realizan en el lugar.



Para buscar un equipo desde la vista móvil se procede de la misma forma, con la diferencia que en la vista móvil hacemos clic en la lupa ubicada en la esquina superior derecha para mostrar el menú. Para ambos casos (vista de escritorio y móvil) se puede desplazar por el mapa y hacer clic en uno de los marcadores rojos para desplegar el panel de resultados con la información de ese marcador, esto como alternativa a la consulta desde el menú.

Consultar información de estudiantes

Una vez mostrados los resultados, podremos consultar la información general de los estudiantes que integran el equipo seleccionado. Basta con hacer clic y se mostrará el siguiente panel:

The screenshot displays the Geoportal EPSUM interface. On the left, a sidebar contains filters for 'Cohorte' (1 Cohorte 2018), 'Departamento' (Sacatepéquez), and 'Equipo' (Magdalena). Below these is a 'Resumen General' table:

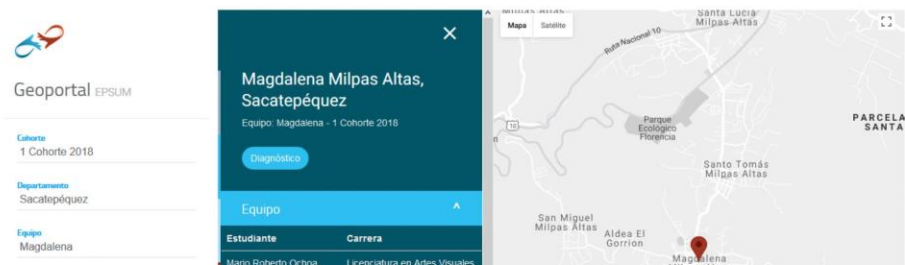
718	Estudiantes
119	Equipos
284	Sedes

The main content area is divided into two panels. The left panel, titled 'Estudiante', shows details for 'Fernandez Velasquez' with fields for Apellidos, Nombre, and Lugar de origen (Guatemala, Guatemala). Below this is the 'Información Académica' section, including 'Registro académico' (201119547), 'Carrera' (Cirujano Dentista), 'Unidad Académica' (Facultad de Odontología), and 'Datos de EPS' (Ejercicio Profesional Supervisado). The right panel is a map showing the location of Magdalena Milpas Altas, with a red pin marker. The map includes labels for nearby locations like Santa Lucía Milpas Altas, Santo Tomás Milpas Altas, and Buena Vista. The bottom of the map shows Google Maps controls and a copyright notice for 2018.

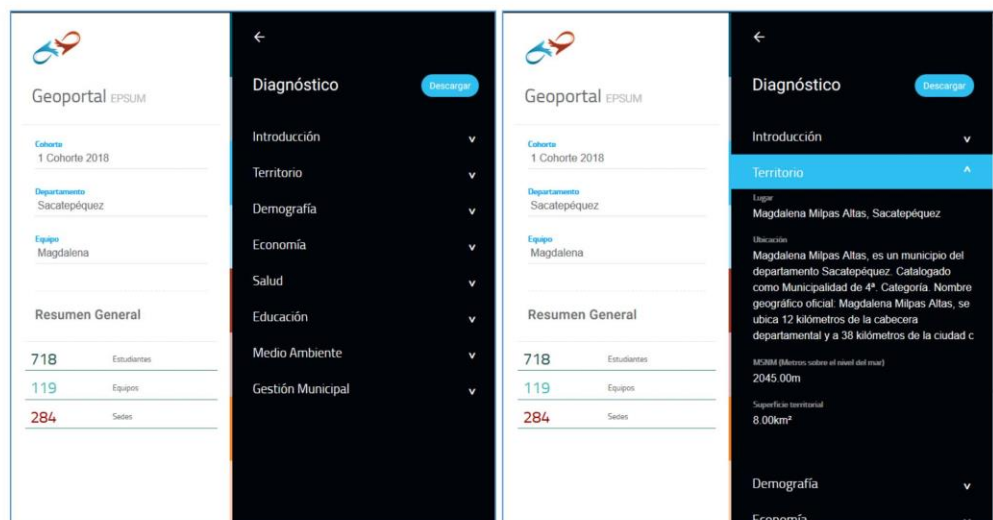
De la misma forma se procede para consultar desde la vista móvil. Para volver al panel de resultados, hacemos clic en la flecha ubicada en la esquina superior izquierda dentro del panel actual.

Consulta de diagnósticos

Para consultar un diagnóstico registrado por un equipo multidisciplinario, hacemos clic en el botón **Diagnóstico** ubicado en la parte inferior del nombre del equipo en el panel de resultados.



Al hacer clic en el botón se desplegará un nuevo panel donde se mostrará la información del diagnóstico separado por secciones definidas. Podremos consultar la sección que necesitemos haciendo clic en cada una de ellas.



La opción descargar nos permite generar un documento en PDF como reporte de la información consultada. La misma forma aplica a la vista móvil de la aplicación.

Consulta de Proyectos

La consulta de productos realizados por los equipos la realizamos desde el panel de resultados. Los mismos los encontramos en la sección proyectos, debajo del listado de integrantes del equipo. Para ver el detalle del proyecto, hacemos clic en el mismo y se desplegará un panel con la información requerida.



Geoportal EPSUM

Cohorte:
1 Cohorte 2018

Departamento:
Sacatepéquez

Equipo:
Magdalena

Resumen General

718	Estudiantes
119	Equipos
284	Sedes

Proyecto

Datos Generales

Nombre del proyecto:
Promoción de Excelencia Educativa en la Escuela HUNAPU de Magdalena Milpas Altas.

Ubicación:
Escuela Oficial Urbana Mixta HUNAPU, J.M. de Magdalena Milpas Altas, Sacatepéquez

Objetivo General:
Aumentar el nivel de aprobación en las pruebas de matemáticas y lectura de los niños de la Escuela Oficial Urbana Mixta HUNAPU J.M. de Magdalena M.A.

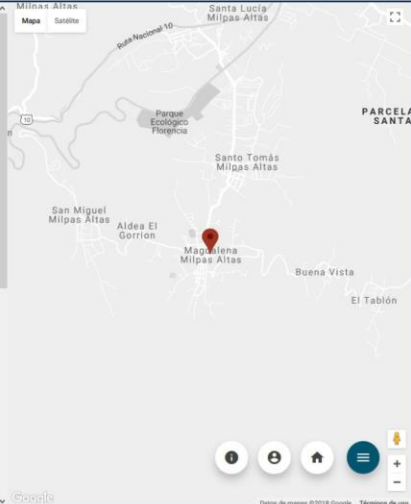
Tipo de proyecto:
Multidisciplinario

Valor económico estimado:
Q.9000

Ámbito de intervención:

Línea:
Fortalecimiento al sistema educativo

Actores:



Mapa de Magdalena Milpas Altas, Sacatepéquez. Se muestra la ubicación de la escuela HUNAPU J.M. de Magdalena M.A. en el centro del pueblo, rodeada por zonas como San Miguel Milpas Altas, Aldea El Gorrión, Santo Tomás Milpas Altas, Buena Vista, El Tablón, y Santa Lucía Milpas Altas. El mapa incluye la Ruta Nacional 10 y el Parque Ecológico Florencia.



Geoportal EPSUM

Cohorte:
1 Cohorte 2018

Departamento:
Sacatepéquez

Equipo:
Magdalena

Resumen General

718	Estudiantes
119	Equipos
284	Sedes

Valor económico estimado:
Q.9000

Ámbito de intervención:

Línea:
Fortalecimiento al sistema educativo

Área:
Educación

Políticas asociadas:
Políticas Educativas, Políticas de Educación Ambiental

Tiempo y ejecución:

Fecha de inicio:
2018-02-15

Fecha de finalización:
2018-10-08

Estado:
En ejecución

Actores:

Partners involucrados:
COMUSAN, COCODE, MINEDUC,

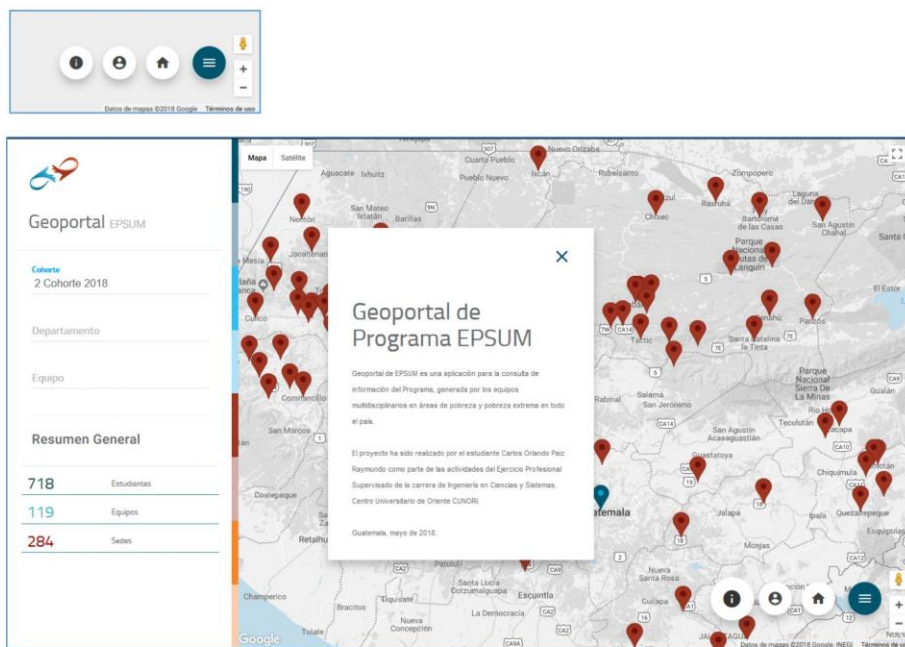


Mapa de Magdalena Milpas Altas, Sacatepéquez. Se muestra la ubicación de la escuela HUNAPU J.M. de Magdalena M.A. en el centro del pueblo, rodeada por zonas como San Miguel Milpas Altas, Aldea El Gorrión, Santo Tomás Milpas Altas, Buena Vista, El Tablón, y Santa Lucía Milpas Altas. El mapa incluye la Ruta Nacional 10 y el Parque Ecológico Florencia.

Para consultar proyectos desde un dispositivo móvil, procedemos de la misma manera desde el panel de resultados.

Opciones adicionales

En la esquina inferior derecha, encontraremos un botón de menú de color azul. Al hacer clic en él se desplegarán opciones que nos permitirá visualizar distinta información. El botón con el ícono de casa nos llevará a la página principal del programa EPSUM [epsum.usac.edu.gt]. El botón con el ícono de usuario corresponde a un enlace que llevará al inicio de sesión de usuarios en la plataforma de EPSUM y el botón de información muestra información acerca del geoportal.



¡Enhorabuena!

Esto es todo lo que necesitas saber para poder utilizar el sistema sin problemas. Siéntete libre de contactar a los desarrolladores en caso de que se presente algún inconveniente con el uso del geoportal.

El equipo de desarrolladores de EPSUM.

Apéndice 5. **Guía de usuario de sistema de gestión de inventario**



Guía de uso del Sistema de Gestión de Inventario de Activos fijos del Programa EPSUM.

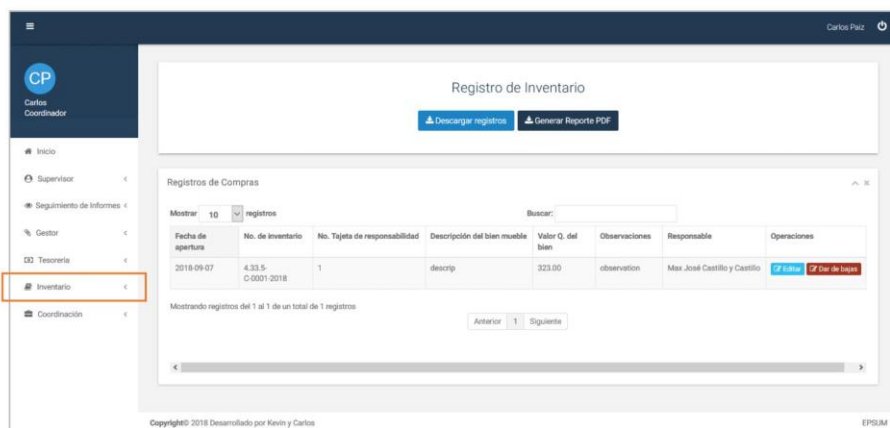
Sistema de Gestión de Inventario de Activos Fijos del Programa EPSUM

El sistema de gestión de inventario integrado a la Plataforma MiE del Programa EPSUM, tiene como objetivo centralizar la información de tesorería a fin de llevar registro del control de bienes muebles del Programa, así como de agilizar el proceso de registro mediante la sistematización del mismo.

Para tal efecto, el módulo requiere de la información necesaria para el registro de inventario, la cual es solicitada en un orden específico y de manera gradual por parte del usuario. Esta información se refiere a

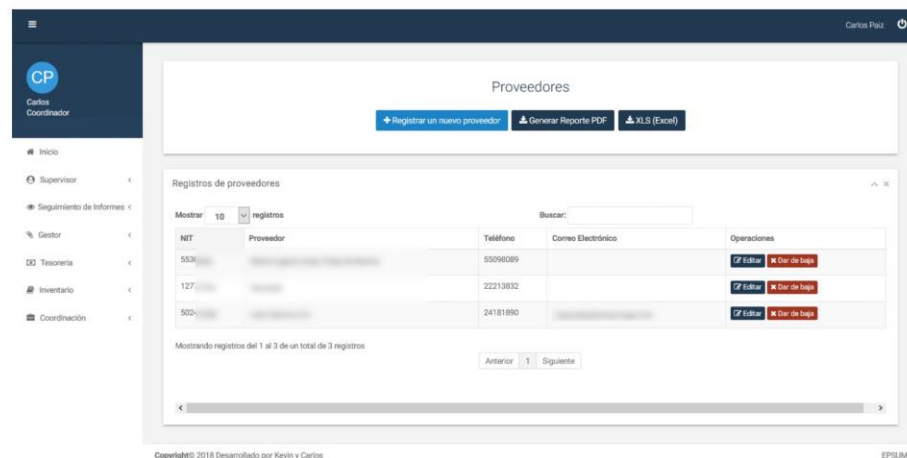
- Proveedores
- Bienes muebles
- Ingresos (como compras)

Como primer paso, el usuario deberá iniciar sesión en la plataforma con rol de Tesorero o Coordinador para poder tener acceso al apartado de Inventario que se encontrará en el menú lateral de la página de Inicio.

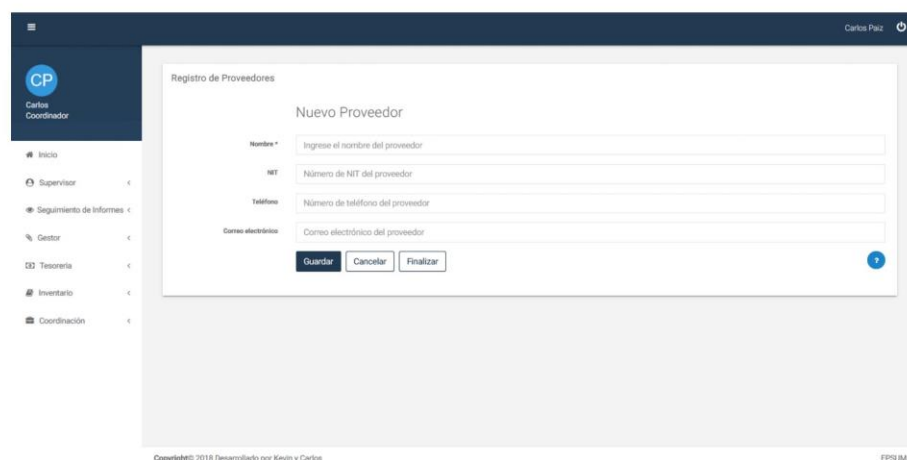


Registro de Proveedores

Para poder registrar nuevos ingresos en inventario, primero deberá contar con una lista de proveedores de los cuales se adquieran los bienes. Para comenzar con el registro de proveedores, dirigirse al menú de inventario y seleccionar la opción proveedores.



Aparecerá la vista de Proveedores con una tabla que lista los proveedores registrados (si los hubiera). El registro iniciará al hacer clic en la opción **Registrar Nuevo Proveedor**, en la parte superior de la tabla. Se mostrará un formulario para iniciar con el registro de proveedores.



El formulario solicita la información básica de un proveedor, con 2 campos únicos obligatorios: Nombre y número de NIT.

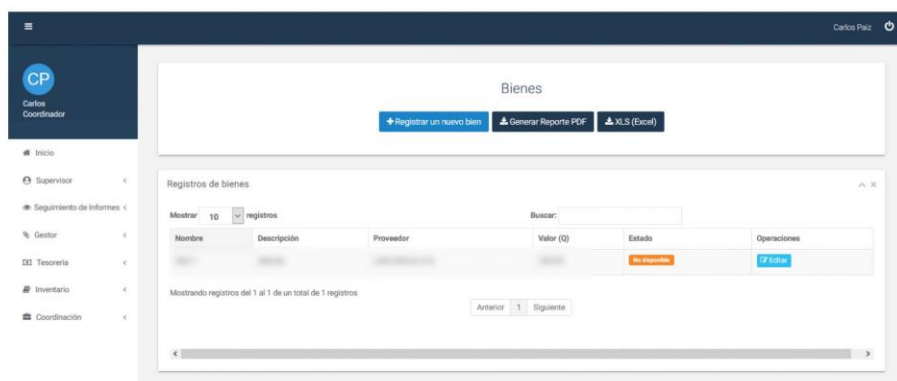
Registro de proveedor

Campo	Descripción
Nombre	Se refiere al nombre del proveedor bajo el cual se representa la empresa
NIT	Corresponde al número de NIT del proveedor registrado en la SAT
Teléfono	Número de teléfono para contacto con el proveedor. Este campo no es obligatorio
Correo electrónico	Email del proveedor para contacto vía virtual.

Una vez completados los campos necesarios podrá seleccionarse la opción **Guardar** para terminar el registro. El sistema limpiará los campos automáticamente para proceder con el registro de otro proveedor si se necesitara, caso contrario, seleccionar la opción **Finalizar**.

Registro de bienes muebles

Para continuar alimentando al sistema de inventario, se deberá ingresar los bienes a registrar, definiendo sus principales características. Para ello, desde el menú de inventario seleccionar la opción **Bienes**. Se mostrará en pantalla un listado de los bienes registrados (si los hubiera). Seleccionar la opción Registrar nuevo bien en la parte superior de la tabla para iniciar con el registro.



Se mostrará en pantalla un formulario con los siguientes campos a completar:

Registro de Bienes

Nombre: Nombre del bien mueble

Descripción: Ingrese una descripción para el bien mueble

Proveedor: [Dropdown menu]

Valor (\$): Ingrese el valor del bien

Guardar Cancelar Finalizar

Copyright© 2018 Desarrollado por Kevin y Carlos EPOLIM

Registro de bien

Campo	Descripción
Nombre	Se refiere al nombre del bien mueble a completar, sin describirlo detalladamente
Descripción	Solicita la descripción a detalle del bien a registrar, ingresando información como sus características principales
Proveedor	Seleccionar a través de la lista desplegable el proveedor que provee el bien
Valor	Se refiere al precio del bien al cual fue adquirido

Completados todos los campos, se podrá seleccionar la opción **Guardar** para terminar el registro. El sistema limpiará todos los campos y dará la opción para seguir registrando bienes. Al terminar con el registro de bienes, se podrá seleccionar la opción **Finalizar** para ir a la vista principal de bienes.

Registro de ingresos

El registro de ingresos representa la parte principal del sistema de gestión de inventario, pues es en este apartado en el cual se registrarán las compras o adquisiciones como tal.

Para comenzar con el registro de ingresos, seleccionar la opción **Compras** desde el menú de **Inventario**. Se mostrará en pantalla la vista principal de Ingresos en la cual se listarán los registros si los hubiera.

Copyright© 2018 Desarrollado por Kevin y Carlos EPSUM

Seleccionar la opción Registrar un nuevo ingreso desde el menú ubicado en la parte superior de la tabla.

Copyright© 2018 Desarrollado por Kevin y Carlos EPSUM

Aparecerá un nuevo formulario con la información a llenar para el registro de la compra. Los datos a considerar se muestran en la siguiente tabla.

Registro de nuevo ingreso

Campo	Descripción
Proveedor	Selección del proveedor de la lista de proveedores registrados.
Factura No.	Número del documento de factura asociado a la compra
Valor	Monto total especificado en la factura
Orden de solicitud de compra No.	Número de orden a través del cual se giró la compra
Fecha	Fecha de realización de la compra.

Al llenar todos los campos, hacer clic en la opción **Iniciar con el registro de inventario** para continuar con el registro de ingresos.

El proceso de registro de activos es simple, pues la mayoría de campos se completan automáticamente. La lista de bienes se llenará de acuerdo al proveedor seleccionado en el formulario anterior. Para comenzar, seleccionar un bien de la lista precargada. El sistema completará los campos de descripción y valor del bien (registrado previamente).

El número de inventario se llena automáticamente de acuerdo al estándar utilizado, de la misma forma el número de tarjeta de responsabilidad y la fecha.

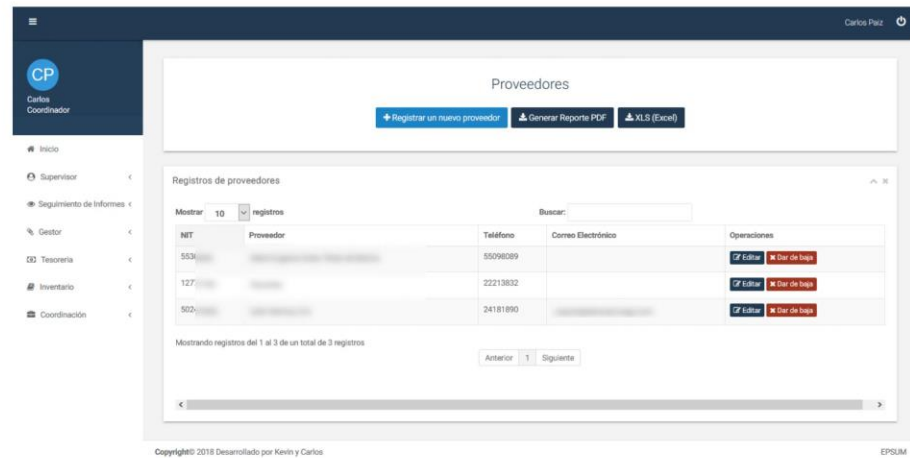
Registro de activos en inventario	
Campo	Descripción
Bien	Se refiere al nuevo activo que se registrará en inventario. Seleccionarlo de la lista de bienes del proveedor indicado en la compra.
Número de inventario	Es el número correlativo de registro en inventario, el cual sigue un estándar definido. Éste es generado automáticamente
Descripción	Campo llenado automáticamente al seleccionar un bien, el cual se refiere a la descripción del mismo.
Valor	Campo llenado automáticamente al seleccionar un bien, el cual se refiere al valor del mismo
No. Tarjeta de responsabilidad	Número de tarjeta asignado al bien, el cual corresponde al correlativo de inventario.
Fecha de apertura	La fecha de ingreso, completada automáticamente
Encargado	Persona responsable del nuevo activo ingresado. Seleccionar de la lista de personal.
Observaciones	Campo no obligatorio para ingresar observaciones, si las hubiera.

Una vez completados los campos, seleccionar la opción de Registro para completar. Se mostrará el resumen del nuevo ingreso en la tabla que aparece en la parte inferior de la misma pantalla y se limpiarán los campos en el caso de que sea necesario registrar nuevos ingresos. Para finalizar, hacer clic en la opción **Finalizar** ubicada debajo de la tabla de detalle.

Edición de proveedores

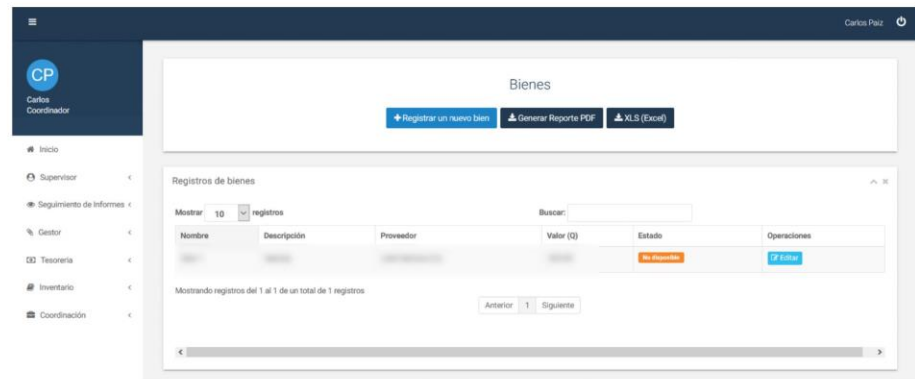
Para modificar información de proveedores, seleccionar la opción **Editar** ubicada en la tabla de proveedores. Se mostrará un formulario con la información cargada del proveedor indicado.

Para dar de baja a un proveedor, seleccionar la opción **Dar de baja**, el sistema solicitará la confirmación de dicha acción y si cumple, se procederá con la baja del proveedor seleccionado.



Editar información de bienes

Para modificar la información de bienes, seleccionar la opción **Editar** en la tabla de consulta de la vista principal de bienes. Se mostrará el formulario con la información cargada del bien seleccionado.



Generar reportes en PDF y XLS (Excel)

El sistema permite la generación de reportes en formato PDF a partir de la información almacenada en el sistema. Para ello, dirigirse a la vista principal de Proveedores, Bienes o Ingresos para seleccionar la opción Generar Reporte PDF. De la misma manera, podrá elegirse la opción **XLS (Excel)** para exportar los registros como archivo XLS.

Bienes

[+ Registrar un nuevo bien](#) [Generar Reporte PDF](#) [XLS \(Excel\)](#)



PROGRAMA EPSUM
CENTRO CULTURAL UNIVERSITARIO, SEGUNDO NIVEL, ALA
SUR. 2DA. AVENIDA 12-40 ZONA 1, GUATEMALA

REPORTE DE PROVEEDORES

NIT	PROVEEDOR	TELÉFONO	CORREO
127	Proveedores	55717000	
502	Labio Dental, S.A.	55717000	labio@labiodental.com.gt
553	Centro Cultural Universitario	55717000	

¡Enhorabuena!

Esto es todo lo que necesitas saber para poder utilizar el sistema sin problemas. Siéntete libre de contactar a los desarrolladores en caso de que se presente algún inconveniente con el uso del sistema de inventario.

El equipo de desarrolladores de EPSUM.

EPSUM (miepsum@gmail.com)

Apéndice 6. **Manual de especificaciones técnicas y de mantenimiento del proyecto**

Manual de Especificaciones Técnicas

/ *

- * Módulo de Gestión de Informes de Diagnóstico y de Proyectos
- * Sistema de Gestión de inventario de Activos Fijos

* /

Introducción

El software desarrollado constituye un módulo para registro de informes diagnósticos, planes de trabajo y perfiles de proyectos, así como el módulo de consulta de información georreferenciada. Además, se incluye la configuración para el mantenimiento del sistema de gestión de inventario de la entidad. Los proyectos, fueron diseñados utilizando PHP y el framework Laravel, así también otras tecnologías como JavaScript y AJAX.

Se inicia con la contextualización del proyecto desarrollado, seguido de la clonación del repositorio para posteriormente definir todas las especificaciones del proyecto como clases y métodos principales.

Importante

El presente manual técnico constituye una guía para la instalación y mantenimiento de los módulos de Registro de informes de EPSUM, Geoportal y Sistema de Inventario. Puesto que éstos corresponden a una extensión del Proyecto de Sistema de Gestión de Estudiantes, Sedes y Pagos, referirse al manual técnico de dicho proyecto para la configuración global del sistema.

Este manual no detalla o describe funcionalidades del framework Laravel, por lo que debe referirse a la documentación oficial de dicho framework en su versión 5.1.

Requerimientos

A continuación, se lista un resumen de las herramientas utilizadas y sus respectivas versiones.

- | | |
|-----------------|--------------------|
| ▪ XAMP 5.7.23 | ▪ MySQL 5.7.23 |
| ▪ Apache 2.4.18 | ▪ phpMyAdmin 4.8.3 |
| ▪ PHP 7.0.30 | ▪ Composer |

Contextualización del proyecto

Producto de la metodología aplicada para el desarrollo del proyecto, se generaron una serie de artefactos que se presentan a continuación. Para introducir al análisis y diseño del sistema, se utilizará el modelo de vistas 4+1 de Kruchten, el cual permitirá describir de una forma clara, la arquitectura del software.

Vista Lógica

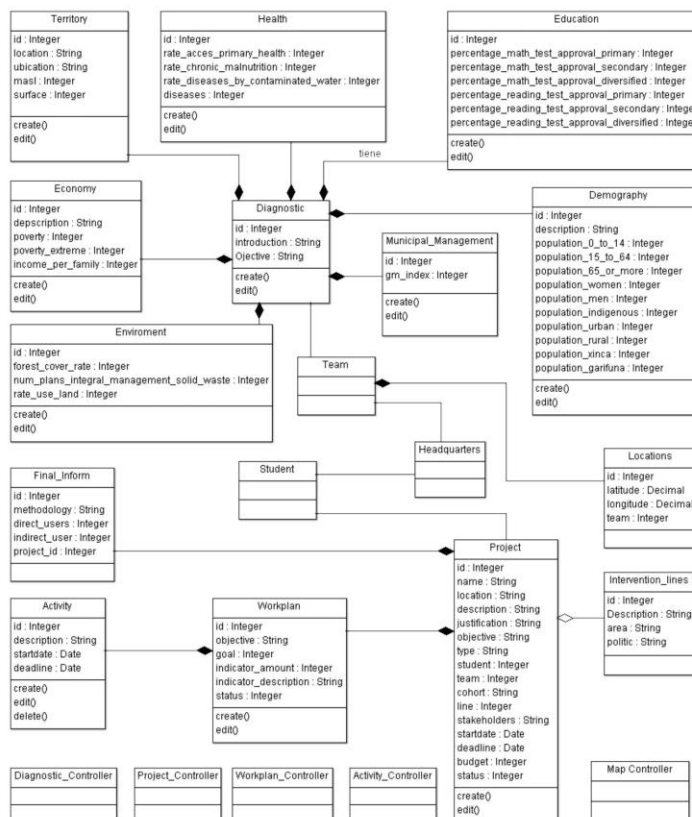
La vista lógica describe la estructura y funcionalidad del sistema. Se representa esta vista mediante un diagrama de clases. En este caso, Laravel sigue una arquitectura pura MVC (Modelo-Vista-Controlador), de manera que se detallarán los modelos y controladores en la vista lógica de Kruchten, mas no las vistas, pues no determinan la lógica de la aplicación.

Las principales clases correspondientes al proyecto de registro de informes diagnósticos y de proyectos, son las siguientes:

Clase	Definición
Diagnostic	Almacena la información referente a informes diagnósticos
Project	Define el perfil de un proyecto
Workplan	Almacena la información referente a los planes de trabajo asociado a un proyecto
Activities	Almacena las actividades asociadas a los planes de trabajo
Final_Inform	Define la ficha de resumen de proyectos registrados
Diagnostic_Controller	Clase controlador con métodos para la gestión de diagnósticos
Project_Controller	Clase controlador con métodos para la gestión de proyectos
Workplan_Controller	Clase controlador con métodos para la gestión de planes de trabajo
Activity_Controller	Clase controlador con métodos para la gestión de actividades

En el siguiente diagrama se muestra el diagrama de clases para el registro de diagnósticos y proyectos.

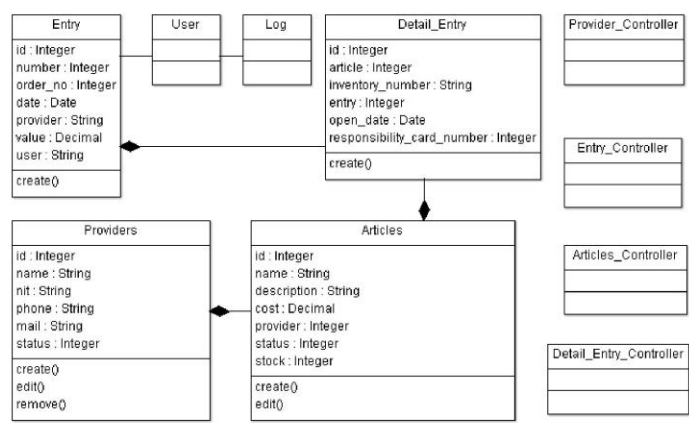
Diagrama de Clases de Registro de Informes Diagnóstico y de Proyectos



Notar las clases dependientes de Diagnostic y de Project, con lo cual que establece que dichas clases dependientes no pueden ser instanciadas sin existir un registro previo de las clases que dependen. Se supone que los nombres de los atributos en las clases describen la razón de los mismos, por lo que no se detallará en su explicación. Si se necesitara conocer a detalle los atributos presentados, referirse a la Guía metodológica para Equipos Multidisciplinarios del Programa EPSUM (2017) en la cual se describen los indicadores plasmados en las clases del diagrama.

Para el sistema de gestión de inventario se definen las clases como se muestra en el siguiente diagrama.

Diagrama de clases de sistema de gestión de inventario



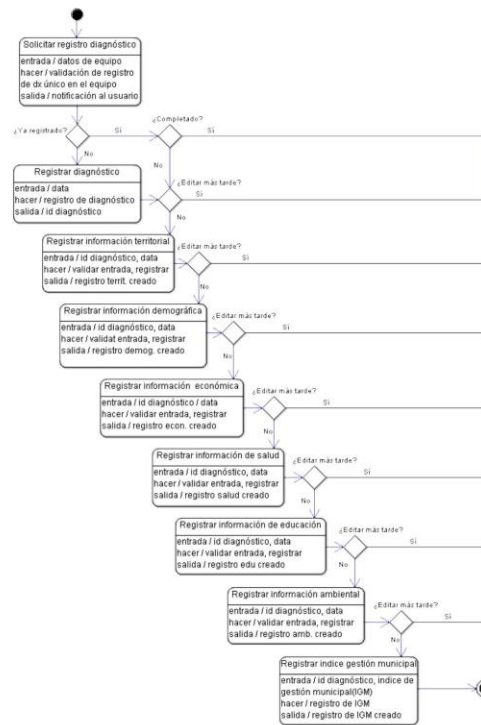
Nuevamente, notar las relaciones de dependencia entre las clases, con lo cual se establece que no pueden ser instanciadas sin los registros previos de las clases de las cuales dependen. En la siguiente tabla se definen las clases más importantes representadas en el diagrama.

Clase	Definición
Providers	Almacena la información referente a los proveedores
Articles	Guarda la información de los bienes adquiridos
Entry	Almacena la información de la compra realizada
Detail_Entry	Se refiere al detalle del bien adquirido a través de la compra
Provider_Controller	Clase controladora para la gestión de información de proveedores
Article_Controller	Clase controladora para la gestión de bienes/artículos
Entry_Controller	Clase controladora para la gestión de ingresos

Vista de proceso

La vista de proceso define aspectos del diseño del software que están relacionados con la concurrencia y sincronización de la ejecución. Se representará esta vista mediante diagramas de estados. Se inicia con el diagrama de estados para el proceso de registro de diagnósticos.

Diagrama de estados de registro de diagnósticos



Como se aprecia en el diagrama, el proceso atiende a un registro colaborativo por parte del equipo multidisciplinario, por lo cual se condiciona la edición de cada uno de los apartados que componen el diagnóstico.

Diagrama de estados de registro de proyectos

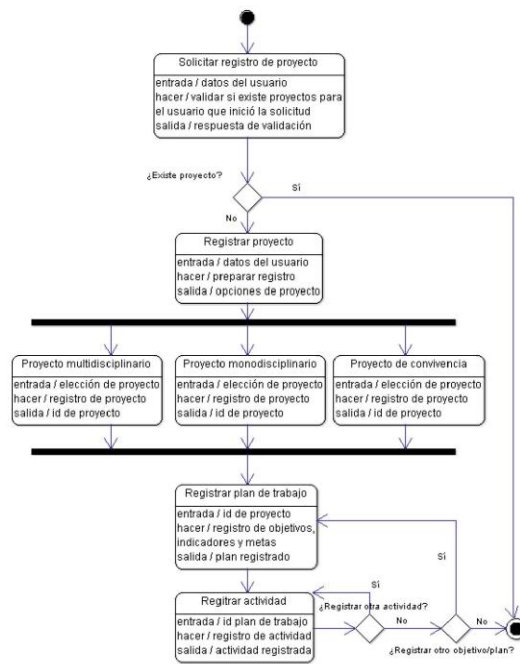


Diagrama de estados de registro de planes de trabajo

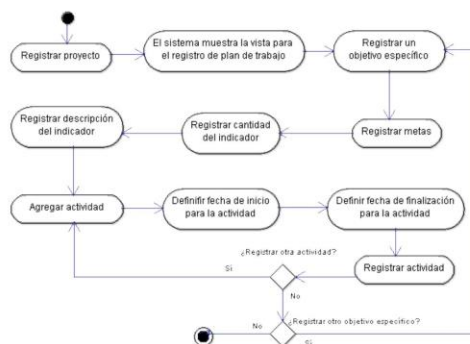


Diagrama de estados de registro de actividades

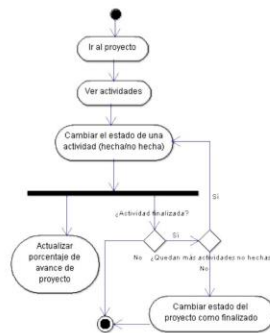
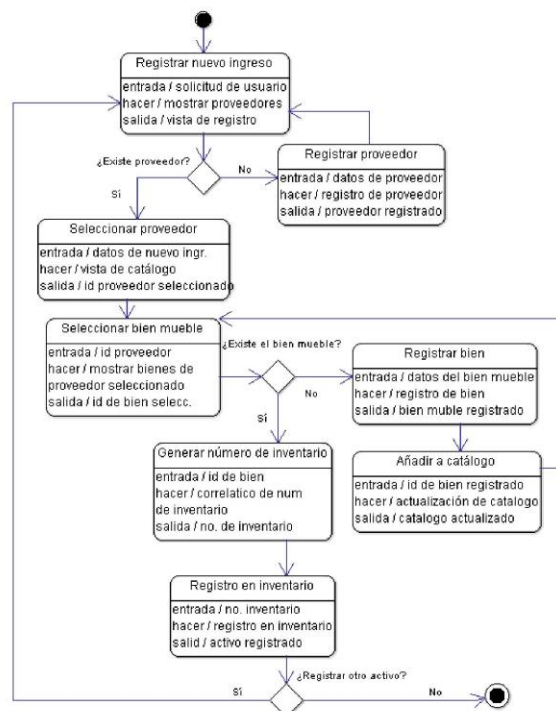


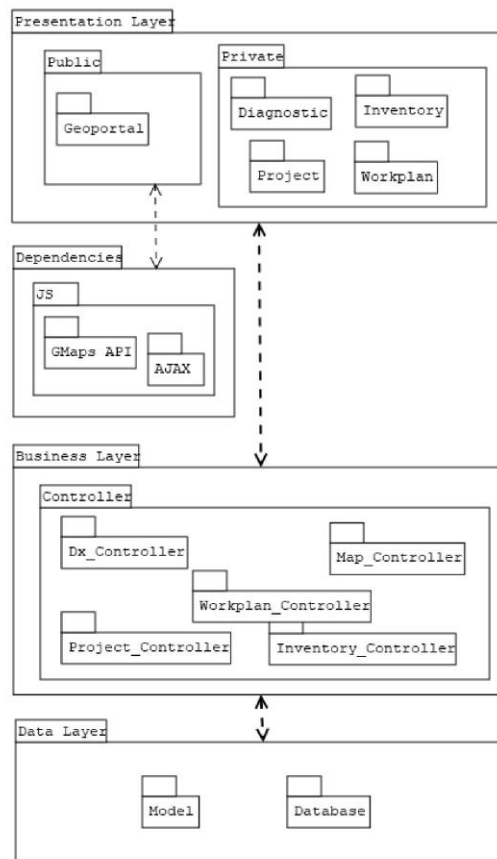
Diagrama de estados de gestión de inventario



Vista de desarrollo

La vista de desarrollo muestra la organización estática del software en su entorno de desarrollo. Se incluyen referencias a la modularidad del sistema, gestión del software y se muestra una descomposición de subsistemas. Se representa esta vista mediante un diagrama de paquetes.

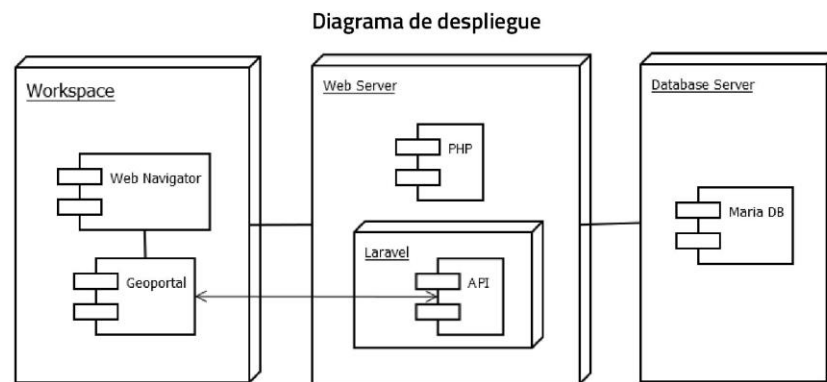
Diagrama de paquetes



Como se muestra en el diagrama anterior, se encuentran los paquetes principales que se implementan el sistema, separados por las capas de datos, de negocio y de presentación. Además, se muestra de manera simplificada los controladores y las vistas que utiliza la aplicación.

Vista física o de despliegue

La vista física realiza un mapeo del software al hardware teniendo en cuenta los requisitos no funcionales del sistema, tales como la disponibilidad, rendimiento y escalabilidad. Se presenta esta vista mediante un diagrama de despliegue.



En el diagrama, se observan los distintos nodos hardware que agrupan los componentes software que se ejecutan en ellos y se enlazarán con otros nodos de proceso necesarios en el sistema.

Vista de escenarios

La quinta vista se representa con los casos de uso o escenarios definidos en la toma de requerimientos. Los escenarios conforman una abstracción de los requisitos más importantes del sistema y su diseño se expresa a través de los diagramas de casos de uso y sus interacciones.

Diagrama de caso de uso de alto nivel

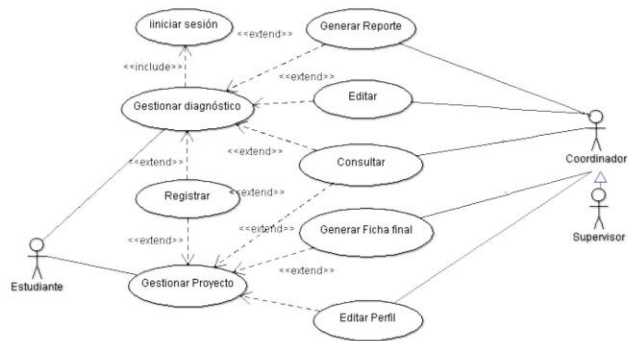


Diagrama de caso de uso de registro de diagnóstico

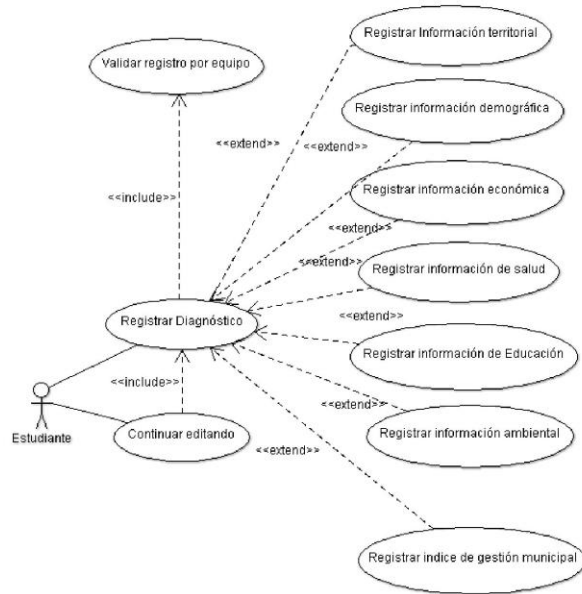


Diagrama de caso de uso de registro de proyectos

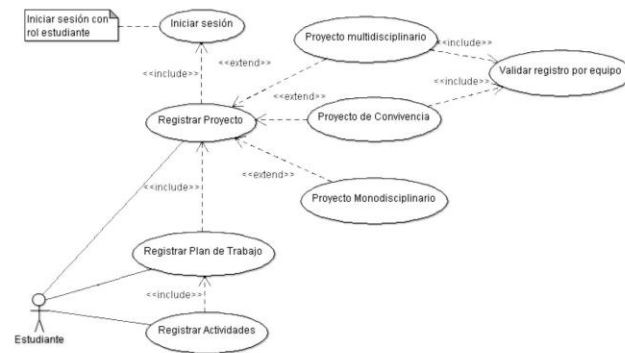


Diagrama de caso de uso de gestión de inventario de alto nivel

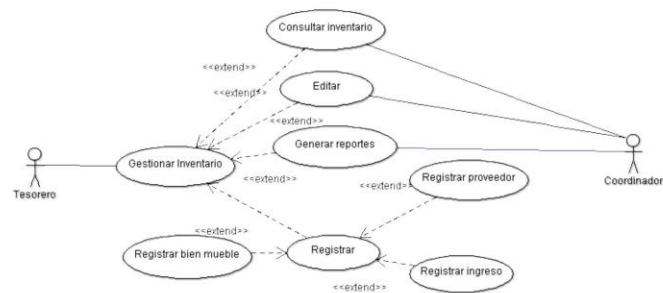


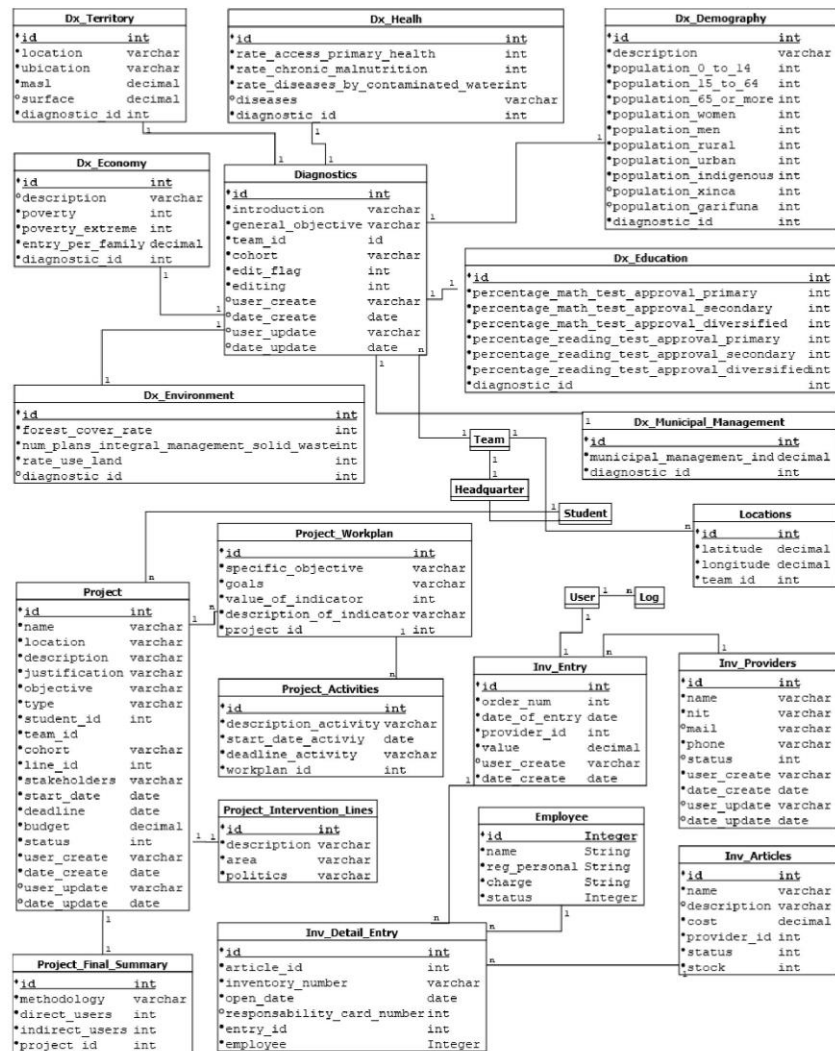
Diagrama de caso de uso de registro en inventario



Diagrama de caso de uso de consulta en geoportal



Modelo relacional de datos



Clonación de repositorio

El proyecto se encuentra alojado en un repositorio de bitbucket, al cual se le dará acceso en el caso que el monitor de Informática solicite la modificación o agregación de funcionalidades al sistema. Para obtener el código, clonarlo desde la consola con GIT, utilizando el siguiente comando:

```
git clone https://[user]@bitbucket.org/mieps.git
```

Estructura del proyecto

El proyecto se compone con la siguiente estructura de paquetes, de los cuales se detallan aquellos que son necesarios para el proyecto correspondiente.

/	Descripción
app	En este directorio se encuentran todos los archivos referentes a la aplicación desarrollada. Este paquete contiene otros directorios como los controladores y los modelos del sistema.
Http	Subdirectorio que resguarda archivos referentes al ruteo de la aplicación.
Controllers	Directorio que contiene los controladores de la aplicación
routes.php	Archivo de configuración de rutas establecidas para la aplicación
Models	Modelos de la aplicación, extienden de la clase Model de Laravel
bootstrap	Archivos de configuración de Laravel (no modificados)
config	Archivos de configuración de la aplicación, modificar los archivos correspondientes sólo si es necesario o según lo establecido por los manuales de uso de aplicaciones de terceros.
database	Contiene directorios con información de archivos para la base de datos
migrations	En este directorio se almacenan los archivos de tipo migration creados utilizando la interfaz de línea de comandos de artisan.
seeds	Directorio que almacena los archivos seeds, creados para la inserción inicial de registros en la base de datos.
public	En este directorio se almacenan archivos varios como recursos utilizados por la aplicación.

css	Directorio que almacena los archivos css creados para la aplicación.
img	En este paquete se encuentran las imágenes creadas para la aplicación.
js	Carpeta que contiene scripts utilizados por la aplicación
resources	Este directorio almacena entre otras cosas, las vistas utilizadas por la aplicación
views	Contiene todas las vistas creadas para la aplicación, las cuales se encuentran separadas por paquetes
storage	Directorio que contiene registros de la cache de la app.
tests	En este paquete se encuentran los tests realizados para la aplicación
vendor	Directorio que contiene los archivos utilizados por aplicaciones de terceros dentro de la aplicación
.env	Archivo de configuración de la aplicación

Configuración inicial

Bajo el supuesto que se cuenta con las tecnologías anteriormente mencionadas, y se ha obtenido el código fuente del proyecto, se puede proceder con la configuración del entorno de desarrollo. Todos los archivos de configuración de Laravel están ubicados por defecto en el directorio “config”, y se recomienda que, en caso de añadirse nuevos, se incluyan en el mismo. Estos archivos permitirán configurar el comportamiento del framework y en última instancia, el de la aplicación soportada.

La configuración del entorno se realiza haciendo uso de la biblioteca DotEnv de PHP. Este módulo permite toda la configuración del entorno en un fichero de extensión *.env. El archivo .env se encuentra en la raíz del paquete principal y está configurado de la siguiente manera:

```
APP_ENV=local
APP_DEBUG=true
APP_KEY=[token]

DB_CONNECTION=mysql
DB_HOST=127.0.0.1
DB_DATABASE=epsum
DB_USERNAME=[database_name]
DB_PASSWORD=[database_password]
```

```
CACHE_DRIVER=file
SESSION_DRIVER=file
QUEUE_DRIVER=sync

REDIS_HOST=127.0.0.1
REDIS_PASSWORD=null
REDIS_PORT=6379

MAIL_DRIVER=smtp
MAIL_HOST=smtp.gmail.com
MAIL_PORT=587
MAIL_USERNAME=niepsun@gmail.com
MAIL_PASSWORD=[mail_password]
MAIL_ENCRYPTION=tls
```

La configuración mostrada almacena el nombre de usuario y contraseña de la aplicación, que serán proporcionados por el encargado de informática en el caso de que se necesite. Así también, se indican el tipo de base de datos, el host y puerto en el cual se encuentra alojada la aplicación. Una vez cargada la configuración desde el fichero .env esta queda almacenada en la caché de la aplicación y no se modificará la misma a menos que se realice el cambio de manera explícita.

Dentro de la aplicación y en consola, ejecutamos el siguiente comando para iniciar con la carga de registros en la base de datos:

```
php artisan migrate:refresh --seed
```

Hecho esto, podremos proceder a trabajar con la aplicación. Para iniciarla, tendremos que tener los servicios de XAMPP, MySQL activos y desde la raíz del proyecto procedemos con el siguiente comando:

```
php artisan serve
```

Migraciones

Las migraciones representan las entidades definidas en el modelo relacional de datos. Las mismas fueron creadas utilizando la interfaz de línea de comandos de Artisan, a través del siguiente comando:

```
php artisan make:migration create_[name_of_the_table]_table
```

Las migraciones tomadas en cuenta para la creación de tablas se listan en la siguiente tabla:

No.	Migration	Clase
1	_create_locations_table.php	CreateLocationsTable
2	_create_diagnostics_table.php	CreateDiagnosticsTable
3	_create_dx_territory_table.php	CreateDxDemographyTable
4	_create_dx_demography_table.php	CreateDxDemographyTable
5	_create_dx_economy_table.php	CreateDxEconomyTable
6	_create_dx_health_table.php	CreateDxHealthTable
7	_create_dx_education_table.php	CreateDxEducationTable
8	_create_dx_environment_table.php	CreateDxEnvironmentTable
9	_create_dx_municipal_management_table.php	CreateDxMunicipalTable
10	_create_interv_lines_table.php	CreateInterLinesTable
11	_create_projects_table.php	CreateProjectsTable
12	_create_workplans_table.php	CreateWorkplansTable
13	_create_activities_table.php	CreateActivitiesTable
14	_create_final_inform_table.php	CreateFinalInformTable
15	_create_inv_providers_table.php	CreateInvProvidersTable
16	_create_inv_articles_table.php	CreateInvArticlesTable
17	_create_inv_purchases_table.php	CreateInvPurchasesTable
18	_create_inv_details_table.php	CreateInvDetailTable
19	_create_employees_table.php	CreateEmployeesTable

Tras ejecutar el comando para crear una migración se crea un archivo de clase en el directorio migrations, cuyo path se definió en la estructura del proyecto. La clase creada extiende de Migration, de Laravel. Consta de dos funciones específicas para la gestión de la nueva tabla: **up()** y **down()**. En la primera se definieron los atributos (también presentados en el diagrama de clases) para crear la tabla en la base de datos y la segunda para eliminar la tabla, esto se realiza haciendo uso de la clase Schema, de Laravel.

Para crear las tablas en la base de datos se utilizó el siguiente comando utilizando el CLI de artisan

```
php artisan migrate
```

Realizadas las migraciones, se podrán ejecutar los seeds o, en su defecto, utilizar un script de sql para proceder a realizar las inserciones en la base de datos. Se recomienda la segunda opción a fin de contar con la información actualizada. Solicitar el script al encargado de informática.

Modelos

Los modelos posibilitan el uso de características propias de la orientación a objetos, con lo cual se puede acceder directamente a los campos de un objeto para leer los datos de una base de datos o realizar otras acciones. Laravel dispone de su propio sistema ORM llamado Eloquent, el cual proporcionó una manera fácil para interactuar con la base de datos. Para crear un nuevo modelo se utilizó el siguiente comando de artisan:

```
php artisan make:model NameOfTheModel
```

Las clases creadas tras la ejecución del comando para cada uno de los modelos se detalla en la siguiente tabla. Las clases se crearon en la carpeta app, especificada en la estructura del proyecto.

No.	Clase
1	Diagnostic
2	DxTerritory
3	DxDemography
4	DxEconomy
5	DxHealth
6	DxEducation
7	DxEnvironment
8	DxMunicipalManagement
9	Project
10	Workplan
11	Activity
12	FinalInform
13	Provider
14	Article
15	Purchase

Los atributos agregados en los modelos corresponden a los definidos en el diagrama de clases presentado con anterioridad. Para la utilización de los modelos, es importante indicar al inicio de la clase el espacio de nombres del modelo. Esto se realiza haciendo uso de la siguiente línea de código:

```
Use App\NameOfTheModel
```

Controladores

Los controladores constituyen el punto de entrada de las peticiones de los usuarios y contienen toda la lógica asociada a un recurso, encargándose de realizar las consultas necesarias a la base de datos, preparar los datos y de llamar a la vista correspondiente para mostrar los datos. Para crear los controladores de esta aplicación se utilizó el siguiente comando de artisan:

```
php artisan make:controller [Name]Controller
```

Los controladores creados se almacenan en la carpeta Http que encuentra en el paquete app. Los mismos se listan con sus respectivos métodos en la siguiente tabla:

Controlador	Métodos	Definición
MapController	index()	Carga la vista inicial con la data necesaria.
	filterTeams(\$department)	Devuelve los equipos según el id de departamento
	getInfo(\$team, \$cohort)	Devuelve la información de estudiantes por equipo y cohorte
	getTeamProjects(\$team, \$cohort)	Retorna los registros de proyectos por equipo y cohorte
	getDetail(\$id)	Devuelve la información de un estudiante mediante su id
	getDetailProject(\$id)	Retorna la información de un proyecto por su id
	getLocation(\$team)	Devuelve las coordenadas de un equipo específico
	getName(\$team)	Devuelve el nombre de un equipo específico

	getDxTeam(\$team, \$cohort)	Devuelve un diagnóstico único por equipo y cohorte
DxController	index()	Carga la vista inicial de diagnóstico
	create()	Carga la vista de un diagnóstico si existe para el usuario iniciado o redirige al registro de diagnóstico
	store(Request \$request)	Almacena un nuevo registro de diagnóstico
	goToEdit(\$edit_flag)	Redirige al apartado de diagnóstico según edit_flag para continuar editando
	closeEdit()	Cambia el estado de editando a 0 para indicar que ya no se está editando
	editDx()	Retorna la vista de edición
	getDxs()	Devuelve todos los registros de diagnósticos según rol de usuario
	getDxByCohort(\$current_cohort)	Devuelve los registros de diagnóstico según la cohorte seleccionada
	getDxDeatil(\$team,\$cohort)	Devuelve el detalle de un diagnóstico específico
	getReport(\$team,\$cohort)	Crea y retorna un archivo de reporte en formato PDF
	changeEditDxStatus()	Cambia el estado de diagnóstico a 0 (finalizado)
	update(Request \$request)	Actualiza la información de un diagnóstico
DxTerritoryController	index()	Carga la vista inicial del apartado territorio de dx
	store(Request \$request)	Guarda el registro de territorio del diagnóstico
	editTerritory()	Retorna la vista de edición de territorio
	update(Request \$request)	Actualiza la información de territorio del dx
DxDemographyController	index()	Carga la vista inicial del apartado Demografía de dx
	store(Request \$request)	Guarda el registro de demografía del diagnóstico
	editDemography()	Retorna la vista de edición de demografía

	update(Request \$request)	Actualiza la información de demografía del dx
DxEconomyController	index()	Carga la vista inicial del apartado economía de dx
	store(Request \$request)	Guarda el registro de economía del diagnóstico
	editEconomy()	Retorna la vista de edición de economía
	update(Request \$request)	Actualiza la información de economía del dx
DxHealthController	index()	Carga la vista inicial del apartado salud del dx
	store(Request \$request)	Guarda el registro de salud del diagnóstico
	editHealth()	Retorna la vista de edición de salud
	update(Request \$request)	Actualiza la información de salud del dx
DxEducationController	index()	Carga la vista inicial del apartado educación de dx
	store(Request \$request)	Guarda el registro de educación del diagnóstico
	editEducation()	Retorna la vista de edición de educación
	update(Request \$request)	Actualiza la información de educación del dx
DxEnvironmentController	index()	Carga la vista inicial del apartado de medio ambiente del dx
	store(Request \$request)	Guarda el registro de medio ambiente del diagnóstico
	editEnvironment()	Retorna la vista de edición de medio ambiente del dx
	update(Request \$request)	Actualiza la información de medio ambiente del dx
DxMunicipalManagementController	index()	Carga la vista inicial del apartado gestión municipal de dx
	store(Request \$request)	Guarda el registro de gestión municipal del diagnóstico
	editMunicipalManagement()	Retorna la vista de edición de gestión municipal del dx
	update(Request \$request)	Actualiza la información de gestión municipal del dx

ProjectController	create(\$type)	Carga la vista de creación de proyecto de acuerdo al tipo de proyecto
	getIntervLines(\$id)	Retorna una línea de intervención especificada por id
	store(Request \$request)	Almacena un nuevo registro de proyecto
	getProject(\$type)	Devuelve un proyecto para un usuario iniciado según el tipo especificado
	checkActivityDone(\$id)	Cambia el estado de una actividad del proyecto como hecha
	checkActivityUndone(\$id)	Cambia el estado de una actividad del proyecto como no hecha
	getAllProjects()	Devuelve todos los registros de proyectos según el rol de usuario iniciado
	showReport(\$id)	Devuelve toda la data asociada al proyecto, como plan de trabajo y actividades
	getProjectDetail(\$id, \$team, \$cohort, \$type)	Devuelve toda la data de un proyecto específico
	update(Request \$request)	Actualiza la información de un proyecto específico
WorkplanController	create()	Carga la vista inicial del plan de trabajo
	store(Request \$request)	Almacena un nuevo registro de plan de trabajo
	addWorkplan(Request \$request)	Guarda un nuevo registro de plan de trabajo con validaciones para AJAX
	editObjective(Request \$request)	Actualiza la información de un objetivo de plan de trabajo utilizando validaciones para AJAX
ActivityController	addActivity(Request \$request)	Guarda el registro de una actividad utilizando validaciones para AJAX
	addActivityFromProject(Request \$request)	Guarda una nueva actividad agregada desde la vista inicial del proyecto, AJAX
	delete(Request \$request)	Elimina el registro para una actividad específica

	<code>editItem(Request \$request)</code>	Actualiza la información de una actividad
FinalInformController	<code>toInform(\$type)</code>	Carga la vista de creación de informes con la data del proyecto
	<code>store(Request \$request)</code>	Guarda un nuevo registro de informe final (ficha de resultados)
	<code>getReport()</code>	Crea y retorna un archivo PDF del informe final (ficha de resultados)
InventoryProviderController	<code>create()</code>	Carga la vista inicial con todos los registros de proveedores
	<code>allProviders()</code>	Retorna todos los proveedores con estado activo
	<code>store(Request \$request)</code>	Guarda el registro de un nuevo proveedor
	<code>edit(\$id)</code>	Carga la vista de edición con el proveedor especificado
	<code>update(Request \$request)</code>	Actualiza la información de un proveedor especificado
	<code>dismiss(\$id)</code>	Da de baja al proveedor especificado
	<code>getReport()</code>	Genera un reporte en PDF de los proveedores registrados
InventoryArticleController	<code>create()</code>	Carga la vista inicial con todos los registros de bienes/artículos registrados
	<code>allArticles()</code>	Retorna todos los artículos registrados
	<code>store(Request \$request)</code>	Guarda el registro de un nuevo bien/artículo
	<code>edit(\$id)</code>	Carga la vista de edición con el artículo especificado
	<code>update(Request \$request)</code>	Actualiza la información de un artículo especificado
	<code>getReport()</code>	Genera un reporte en PDF de los bienes/artículos registrados
InventoryController	<code>index()</code>	Carga la vista general de inventario con todos los registros
	<code>allPurchases()</code>	Carga la vista de registros de compras registradas

	<code>create()</code>	Retorna la vista de creación de un nuevo ingreso
	<code>store(Request \$request)</code>	Almacena el registro de un nuevo ingreso
	<code>register(\$id)</code>	Carga la vista de registro de adquisiciones asociadas a la compra
	<code>fillArticlesFields(\$id)</code>	Retorna los datos de un artículo especificado
	<code>regArticle(Request \$request)</code>	Guarda el registro de un activo en inventario asociado a una compra. AJAX
	<code>getPurchaseReport()</code>	Genera un reporte en PDF de las compras registradas
	<code>getReport()</code>	Genera un reporte PDF de los registros generales de inventario

Vistas

Las vistas son almacenadas en el directorio `resources/views` del proyecto. Las mismas utilizan Blade (que es una librería utilizada por Laravel para gestionar de una manera más fácil las vistas) por lo cual los archivos tendrán una extensión `.blade.php`. Las vistas creadas para el proyecto se listan en la siguiente tabla

Paquete	Vista	Definición
<code>/</code>	<code>map</code>	Vista general del geoportal
<code>dx</code>	<code>create.blade.php</code>	Vista de creación de diagnóstico
	<code>demography-edit.blade.php</code>	Vista de edición de información de demografía
	<code>diagnostics.blade.php</code>	Vista general de diagnóstico
	<code>dx-detail.blade.php</code>	Vista de detalle de diagnóstico por usuario iniciado

	dx-edit.blade.php	Vista de edición de diagnóstico
	economy-edit.blade.php	Vista de edición de información de economía
	economy.blade.php	Vista de registro de inf. de economía del dx
	education-edit.blade.php	Vista de edición de info. de educación del dx
	education.blade.php	Vista de registro de info. de educación del dx
	environment-edit.blade.php	Vista de edición de información de ambiente del dx
	environment.blade.php	Vista de registro de información de ambiente del dx
	health-edit.blade.php	Vista de edición de información de salud del dx
	health.blade.php	Vista de registro de información de salud del dx
	index.blade.php	Vista de inicio de diagnóstico, si no se ha creado
	municipalmanagement-edit.blade.php	Vista de edición de información de gestión municipal
	municipalmangement.blade.php	Vista de registro de información de gestión municipal del dx
	terrotory-edit.blade.php	Vista de edición de información del territorio de dx
	territory.blade.php	Vista de registro de información del territorio del dx
informs	final_inform.blade.php	Vista de registro de ficha de resultados
	inform.blade.php	Vista de informe de resultados

	Inform_from_detail.blade.php	Vista de informe de diagnóstico cargada desde el detalle del proyecto
	no_inform.blade.php	Vista de notificación para informes que no pueden ser creados
inventory_article	create.blade.php	Vista de registro de artículos
	edit.blade.php	Vista de edición de artículos
	index.blade.php	Vista de consulta general de artículos
inventory_input	create.blade.php	Vista de creación de nuevo registro de compra/ingreso
	index.blade.php	Vista general de inventario
	index_purchases.blade.php	Vista general de compras/ingresos
	regis.blade.php	Vista de creación de registro de inventario
inventory_provider	create.blade.php	Vista de creación de registro de proveedor
	edit.blade.php	Vista de edición de información de proveedor
	index.blade.php	Vista general de consulta de proveedores
projects	allprojects.blade.php	Vista general de consulta de proyectos
	create.blade.php	Vista de registro de proyectos
	edit.blade.php	Vista de edición de proyectos
	index.blade.php	Vista de creación inicial de proyectos, si no existe
	project-detail.blade.php	Vista de detalle de un proyecto
	workplan.blade.php	Vista de creación de plan de trabajo
report	summary-articles.blade.php	Plantilla de reporte de artículos
	summary-dx.blade.php	Plantilla de reporte de diagnósticos

	<code>summary-inventory.blade.php</code>	Plantilla de reporte general de inventario
	<code>summary-providers.blade.php</code>	Plantilla de reporte de proveedores
	<code>summary-purchases.blade.php</code>	Plantilla de reporte de compras
	<code>summary.blade.php</code>	Plantilla de reporte de proyectos

Es importante mencionar que algunos mensajes emergentes o actualizaciones de la vista son realizados a través de JQuery y JavaScript. La librería Blade permite la integración de scripts en la sección

```
@section('script')
```

```
...
```

```
@endsection
```

Por tal motivo, se debe prestar atención a las funciones que no se realizan explícitamente a través del controlador de la vista, si no que directamente a través de un script. Tal es el caso de las funciones AJAX para agregar actividades o planes de trabajo. De la misma forma, la vista del mapa contiene varios scripts para realizar las distintas funcionalidades que ofrece el geoportal, sin necesidad de refrescar la página y así mejorar la experiencia del usuario.

Rutas

Las rutas devuelven un valor o generan una llamada a un controlador. Las mismas se definieron el archivo **routes.php** ubicado en la carpeta **app/Http**. A continuación se listan las rutas creadas para el proyecto

Tipo	Ruta	Acción
GET	<code>/diagnostics/cohort/{current_cohort}</code>	<code>DxController@getDxByCohort</code>
GET	<code>/map</code>	<code>MapController@index</code>

GET	/teams/info/{team}/{cohort}	MapController@getInfo
GET	/teams/projects/{team}/{cohort}	MapController@getTeamProjects
GET	/teams/pj/{id}	MapController@getDetailProject
GET	/teams/student/{id}	MapController@getDetail
GET	/teams/location/{team}	MapController@getLocation
GET	/teams/name/{team}	MapController@getName
GET	/teams/dx/{team}/{cohort}	MapController@getDxTeam
GET	/teams/download/dx/{team}/{cohort}	DxController@getReport
GET	/dx/index	DxController
RESOURCE	/dx/territory	DxTerritoryController
RESOURCE	/dx/demography	DxDemographyController
RESOURCE	/dx/economy	DxEconomyController
RESOURCE	/dx/health	DxHealthController
RESOURCE	/dx/education	DxEducationController
RESOURCE	/dx/environment	DxEnvironmentController
RESOURCE	/dx/municipalmanagement	DxMunicipalManagementController
GET	/dx/all	DxController@getDxs
GET	/dx/detail/{team}/{cohort}	DxController@getDxDetail
GET	/dx/edit/{edit_flag}	DxController@goToEdit
GET	/dx/close	DxController@closeEdit
GET	/dx/edit	DxController@editDx
GET	/dx/1/edit	DxTerritoryController@editTerritory
GET	/dx/2/edit	DxDemographyController @editDemography
GET	/dx/3/edit	DxEconomyController@editEconomy
GET	/dx/4/edit	DxHealthController@editHealth
GET	/dx/5/edit	DxEducationController@editEducation
GET	/dx/6/edit	DxEnvironmentController @editEnvironment
GET	/dx/7/edit	DxMunicipalManagementController @editMunicipalManagement
GET	/project/edit	ProjectController@editProject
RESOURCE	/project	ProjectController
RESOURCE	/workplan	WorkplanController
GET	/project/create	ProjectController@create
GET	/project/{id}/workplan	WorkplanController@create
POST	/add/wp	WorkplanController@addWorkplan
POST	/add/activity	ActivityController@addActivity
POST	/add/activity-from-project	ActivityController @addActivityFromProject
POST	/editact	ActivityController@editItem

GET	/project/line/{id}	ProjectController@getInterLines
GET	/check/activity/{id}	ProjectController@ActivityDone
GET	/uncheck/activity/{id}	ProjectController@ActivityUndone
GET	/projects/all	ProjectController@getAllProjects
GET	/showpj/{id}	ProjectController@showReport
GET	/gotoproject/{id}/{team}/{cohort}	ProjectController@getProjectDetail
POST	/rmvact	ActivityController@delete
POST	/edtobj	WorkplanController@editObjective
POST	/changeeditdxstatus	DxController@changeEditDxStatus
RESOURCE	/inform	FinalInformController
GET	/inform/project	FinalInformController@toInform
GET	/results/report	FinalInformController@getReport
GET	/inventory/providers	InventoryProviderController @allProviders
GET	/inventory/articles	InventoryArticleController @allArticles
GET	/inventory/regs	InventoryController@index
GET	/inventory/purchases	InventoryController/allPurchases
GET	/inventory/provider/edit/{id}	InventoryProviderController@edit
GET	/inventory/article/edit/{id}	InventoryArticleController@edit
GET	/inventory/provider/dismiss/{id}	InventoryProviderController@dismiss
GET	/inventory/register/{id}	InventoryController@register
GET	/regis/fill/{id}	InventoryController@fillArticleFields
POST	/reg/article	InventoryController@regArticle
RESOURCE	/inventory/provider	InventoryProviderController
RESOURCE	/inventory/article	InventoryArticleController
RESOURCE	/inventory	InventoryController
GET	/rep/providers	InventoryProviderController @getReport
GET	/rep/arts	InventoryArticleControllers @getReport
GET	/rep/purs	InventoryController @getPurchasesReport
GET	/repgeneral	InventoryController@getReport

La definición de la ruta puede deducirse a partir de la acción ejecutada, descrita anteriormente en la definición de los controladores.

Tests

Laravel provee un módulo de pruebas de código que utiliza la librería TestCase. Los archivos de pruebas unitarias se encuentran en la carpeta **tests**, del paquete principal. Además de las pruebas básicas como del Web Service y de uso de librerías, se encuentran las pruebas para comprobación de retorno de data.

Para ejecutar los test, utilizar el siguiente comando de artisan:

```
php artisan make:test Test_Name
```

Las pruebas se definen en la siguiente tabla

No.	Test
1	FuncAPITest()
2	GetMapDataTest()
3	GetDxTest()
4	DataProjectTest()
5	DataInvTest()

/ *

* Esto es todo lo que necesitas saber para poder trabajar
* en el proyecto sin problemas. Siéntete libre de contactar
* a los desarrolladores en caso de que se presente
* algún inconveniente.

* El equipo de desarrolladores de EPSUM

* EPSUM (miepsum@gmail.com)

* 2018

* /

Apéndice 7. **Contenido del CD.**

El CD adjunto contiene las siguientes carpetas:

1. Informe Final
2. Anteproyecto
3. Plan de trabajo
4. Proyecto
5. Presentación de resultados finales
6. Fotografías.
7. Informes de EPSUM
 - a. Informe Final de EPSUM
 - b. Informes Mensuales EPSUM
8. Anexos y Apéndices

ANEXOS

- 1. Carta de Aceptación del Software emitida por la Coordinación del Programa EPSUM**
- 2. Finiquito Programa EPSUM**
- 3. Lista de capacitación a estudiantes: Listado de estudiantes capacitados para el uso del software**
- 4. Resumen de diagnóstico territorial EPSUM**
- 5. Resumen de proyecto – Ficha de Resultados EPSUM**
- 6. Manual de Mantenimiento de Sistema Gestión de Estudiantes, Sedes y Equipos del Programa EPSUM**

Anexo 1. Carta de Aceptación del Software emitida por la Coordinación del Programa EPSUM



Ref. EPSUM-783-2018

Guatemala, 13 de agosto de 2018

Ing. René Estuardo Alvarado González
Coordinador de EPS y Trabajos de Graduación
Carrera de Ingeniería y Ciencias y Sistemas
Centro Universitario de Oriente
Universidad San Carlos de Guatemala

Respetable Ingeniero:

Por medio de la presente se hace constar la entrega-recepción del software desarrollado derivado del Ejercicio Profesional Supervisado del estudiante Carlos Orlando Paiz Raymundo, que se identifica el registro académico No. 201145962 y código único de identificación 2399588532001, de la carrera de Ingeniería en Ciencias y Sistemas del Centro Universitario de Oriente. Se recibió a entera satisfacción por la Coordinación del Programa EPSUM la entrega de lo siguiente:

- Software de gestión de informes diagnósticos, plan de trabajo y perfiles de proyectos.
- Software de consulta de información georreferenciada referente a los equipos multidisciplinarios (Geoportal)
- Software de gestión de inventario de activos fijos del Programa EPSUM.

No habiendo otro asunto que tratar, se concluye la presente. Firmo a continuación como responsable de la verificación de la solución, por parte del Programa del Ejercicio Profesional Supervisado Multiprofesional – EPSUM-.




Ing. Mirna Regina Valiente
Coordinadora de Programa EPSUM

cc. archivo

2a. Avenida 12-40 zona 1.
2do. Edificio Antiguo
Paraninfo Universitario

2232-9374 / 2251-2764 /
2253-7893

Anexo 2. Finiquito Programa EPSUM



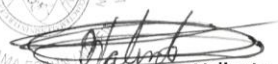
FINIQUITO EJERCICIO PROFESIONAL SUPERVISADO

MULTIPROFESIONAL

No. 168-2018

El Programa de Ejercicio Profesional Supervisado Multiprofesional, extiende el presente **FINIQUITO** al estudiante: Carlos Orlando Paiz Raymundo carné 201145962, de la carrera de **Ingeniería en Ciencias y Sistemas** por haber culminado satisfactoriamente su Ejercicio Profesional Supervisado –EPS-, durante el período comprendido del quince de enero del dos mil dieciocho al quince de julio del dos mil dieciocho, de acuerdo a la normativa del Programa. -----

Se extiende, firma y sella el presente, quince de julio del dos mil dieciocho.-----

“DID Y ENSEÑAD A TODOS”
UNIVERSIDAD DE SAN CARLOS DE GUATEMALA
DIRECCIÓN GENERAL DE EXTENSIÓN
PROGRAMA

Ing. Agr. Mirna Regina Valiente
Coordinadora EPSUM

cc. Archivo

2a. Avenida 12-40 zona 1.
2do. Edificio Antiguo
Parque Universitario
2232-9374 / 2251-2764 /
2253-7893

Anexo 3. Lista de capacitación a estudiantes: Listado de estudiantes capacitados para el uso del software



USAC
TRICENTENARIA
Universidad de San Carlos de Guatemala
Dirección General de Extensión Universitaria



EPSUM
Programa del Empleo Profesional Sumariado Multidisciplinario

DIRECCION GENERAL DE EXTENSION UNIVERSITARIA
UNIDAD: PROGRAMA EPSUM
ACTIVIDAD PROGRAMADA: Taller de Inducción 202018
LUGAR: 75 km
FECHA: 8 agosto 2018

No.	NOMBRE	UNIDAD DONDE LABORA	FIRMA
1	Cristel Gabriela Mumbaza Arzuac	SEPROM, Alta Verapaz	[Firma]
2	Eduardo Ngj	Chisee - Alto Verapaz	[Firma]
3	Karla Maudie Casador	Tactic, Tatumulon	[Firma]
4	Karla Viviana Sosa M	Yuxilha Tamahu	[Firma]
5	Evelyn Maldonado	Chiacal, Tactic	[Firma]
6	Diana Arevalo	El Rancho, San Cristobal	[Firma]
7	Jenifer Del cid	CAP, Tactic	[Firma]
8	Ingrid Rodriguez	San Cristobal V.	[Firma]
9	Evelyn Garcia Xitumul	Carcha	[Firma]
10	Maria Andrea Cagua Carbon	Municipalidad Chisee	[Firma]
11	Rosa Maria Reyes Bach	Mojitob, Santa Cruz	[Firma]
12	Andrea Alejandra Ramirez Morales	Chiquito, Santa Cruz	[Firma]
13	Rosa Maria Ferrnanda Ponce Cortes	Santa San Chamelco, Municipalidad	[Firma]
14	Jose Arnaldo Cu Des Paz	Municipalidad de Santa Catalina La Tron	[Firma]
15	Mildred Emileer Perez Santiago	Municipalidad de San Cristobal V.	[Firma]
16	Estrella de Maria Ruiz Goday	Municipalidad de Tactic, A.V.	[Firma]
17	Gabriela Teresa Gualm Sanchez	Puesto de Salud Chical Coban, A.V.	[Firma]
18	Wilfred Figueroa Rothman	Aldea Seceer Coban, A.V.	[Firma]
19	Rony Danilo Gonzalez Torres	Municipalidad San Pedro Carcha	[Firma]
20	Jorge Roberto Pucal	Puesto de Salud Yalpench Coxruha, A.V.	[Firma]

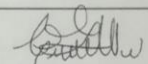
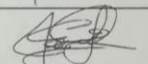
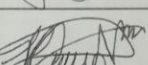
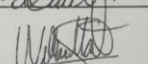
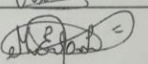
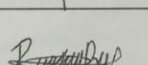
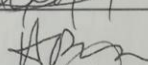
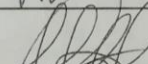
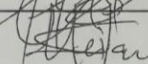
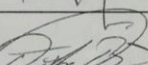
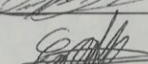
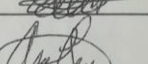
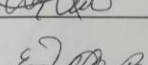
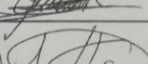
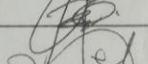
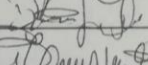
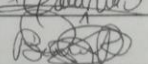
DIRECCION GENERAL DE EXTENSION UNIVERSITARIA

UNIDAD: PROGRAMA EPSUM

ACTIVIDAD PROGRAMADA: _____

LUGAR: _____

FECHA: _____

No.	NOMBRE	UNIDAD DONDE LABORA	FIRMA
1	Cindy Meliza González	Arquitectura	
2	FERNANDO CHAN COANERO	ARQUITECTURA	
3	Hansy Almán Najera Palomo	CUNORI	
4	Williams Manrique Sarcoño Martínez	CUNORI	
5	Marvin Edgardo Calderón Linares	CUNORI	
6	Rony José Collat Flores	CUNORI	
7	Oscar Eduardo Paz Portillo	CUNORI	
8	Rodrigo Enrique Zetina	CUDEF	
9	William de León	Arquitectura	
10	Mei-Ling Lau	CUNOC Arquitectura	
11	Diego Armando Rian Curi	Ingeniería	
12	Edwin Haroldo Cojón Chay	Ingeniería	
13	Andrea Cadena H.	Ingeniería Agronomía	
14	Enril Roberto Ortiz Pérez	CUSAM	
15	Pérez Urizar, Denny Víctor	CUSAM	
16	Jelinton Fernando Contreras López	CUNOC	
17	Amrafi Escobar	CUSAM	
18	Delfino Gabriel López	CUSAM	
19	Carlos Viricio Fuentes	CUSAM	
20	Byron Eduvís Pérez Puc	CUSAM	

DIRECCION GENERAL DE EXTENSION UNIVERSITARIA

UNIDAD: PROGRAMA EPSUM

ACTIVIDAD PROGRAMADA: _____

LUGAR: _____

FECHA: _____

No.	NOMBRE	UNIDAD DONDE LABORA	FIRMA
1	Verónica Johanna Coto Córdova	CUM	[Firma]
2	Gladis Mariela Hernández Lainez	CUM	[Firma]
3	Ana Gabriela Pardo Contreras Castro	CUM	[Firma]
4	Josman Vicente Esteban	CUM	[Firma]
5	Rosa María Barrios	CUM	[Firma]
6	Suzuki Pamela Ruiz	Cum	[Firma]
7	Eduardo Soto y Pardo	Cum	[Firma]
8	maeum Ajup.	Cum	[Firma]
9	Roberto Mancilla	Cum	[Firma]
10	Evelyn Maldonado	Cum	[Firma]
11	Diana Arevalo	Cum	[Firma]
12	Luis Fernando Valdez	Cum	[Firma]
13	Dinora J. Hernández Hernández	Cum	[Firma]
14	Hugo Leonel Boc Boch	Cum	[Firma]
15	Valeria Lirabel Caño Gr.	CUDER-USAC	[Firma]
16	Paula Malán	Cum	[Firma]
17	Juan Canavieja	Cum	[Firma]
18	Rosa María Reyes	Cum	[Firma]
19	Sergio Josue García	CUM	[Firma]
20	Daniel Rodas Cruz	FAUSAC	[Firma]

DIRECCION GENERAL DE EXTENSION UNIVERSITARIA

UNIDAD: PROGRAMA EPSUM

ACTIVIDAD PROGRAMADA: _____

LUGAR: _____

FECHA: _____

No.	NOMBRE	UNIDAD DONDE LABORA	FIRMA
1	José Avelino Cú de Paz	CUNOR	<i>[Signature]</i>
2	Olga María Fernanda Donce Cabel	CUNOR	<i>[Signature]</i>
3	Mildred Emilia Pérez Santiago	CUNOR	<i>[Signature]</i>
4	Estrella de María Ruiz Goady	CUNOR	<i>[Signature]</i>
5	Milsi Yesenia López	CUNSORI	<i>[Signature]</i>
6	Karla Adriana Ortiz Cerado	CUNSORI	<i>[Signature]</i>
7	Lucia Albertina Jimenez Agustín	CUNSORI	<i>[Signature]</i>
8	Lenny Andrés Lemus Giron	CUNORI	<i>[Signature]</i>
9	Lucía de la Cruz Cerna	CUNORI	<i>[Signature]</i>
10	Stephany Lisette Cerna M.	CUNORI	<i>[Signature]</i>
11	Andrea Miranda M	CUNORI	<i>[Signature]</i>
12	Celeste Marescaín	CUNORI	<i>[Signature]</i>
13	Christian Loyo	CUNORI	<i>[Signature]</i>
14	María Andree Castilla	CUNORI	<i>[Signature]</i>
15	Brittney Judith Villeda A.	Cunori	<i>[Signature]</i>
16	Caral Paola Hengaly Martínez	Cunori	<i>[Signature]</i>
17	Adri Schwertner Cortáez	CUNORI	<i>[Signature]</i>
18	Reyna Esther Montero Velásquez	CUNORI	<i>[Signature]</i>
19	José David Obaco Paz	CUDEP	<i>[Signature]</i>
20	Julio Gerardo Morales Palma	CUDEP	<i>[Signature]</i>

DIRECCION GENERAL DE EXTENSION UNIVERSITARIA

UNIDAD: PROGRAMA EPSUM

ACTIVIDAD PROGRAMADA: _____

LUGAR: _____

FECHA: _____

No.	NOMBRE	UNIDAD DONDE LABORA	FIRMA
1	Melany Abigail Archila Marrasquin	Trabajo Social CUNSURORI	
2	Yori Marisela Pío Rosales	Trabajo Social CUNSURORI	
3	Dulce Maria Celeste Sabido Cruz	Trabajo Social CUNSURORI	
4	Mynor Odilcer González Ramos	Trabajo Social CUNSURORI	
5	Brayan Lidany Sarcos Zepeda	Medicina CUNORI	
6	Elias Rodolfo Cardona Sagastume	Zootecnia CUNSURORI	
7	Nestor Rene Martinez Jimenez	Medicina CUNORI	
8	Sandra L. Avalos Acevedo	Medicina CUNORI	
9	Karla Daniela Samayoa Veliz	Medicina CUNORI	
10	Julio Rafael Estrada Gonzalez	Psicología USAC - CUM	
11	Jorge Eduardo Marrasquin Ybarbo	Medicina CUNORI	
12	Luisa Fernanda Ramirez S	Zootecnia CUNORI	
13	Andrea Nineth Andón Valdez	Zootecnia Cunori	
14	Juan Alberto Roche Lemus	Zootecnia Cunori	
15	William Edward Galindo	Zootecnia CUNORI	
16	Alejandro Pérez González	Psicología USAC - CUM	
17	José Alejandro Camacho Varela	Licenciatura en Admin. CUNARO	
18	Briana Sosa L.	Te. P. F CUNPROC - USAC	
19	Doris Carolina Morales Tapia	Medicina - CUM -	
20	Josselyne Dora Cardona Veliz	Medicina - CUM -	
21	Nely Nolasco Chaves Lopez	Medicina CUM	

Anexo 4. Resumen de diagnóstico territorial EPSUM



INFORME DIAGNÓSTICO

JOCOTÁN, CHIQUIMULA

EQUIPO: JOCOTÁN, 1 COHORTE 2018

Introducción

Introducción

El municipio de Jocotán, del departamento de Chiquimula, forma parte de la denominada región Ch'orti' en el oriente de Guatemala. En este municipio la cultura. Es uno de los municipios que presentan mayores cantidades de personas pobres y altos niveles de desnutrición. El Ejercicio profesional supervisado se realiza con el objetivo de crear un vínculo entre la Universidad de San Carlos y la sociedad mediante convenios entre las municipalidad e instituciones que trabajan para el desarrollo de la población de Guatemala, para que los estudiantes fortalezcan los conocimientos adquiridos en las aulas universitarias aplicándolos a la realidad nacional y contribuir con el desarrollo integral de la población, mediante el diagnóstico, la propuesta y ejecución del Plan de servicios y por medio de estas acciones fomentar y mejorar el desarrollo de la comunidad teniendo un impacto positivo en ella.

Territorio de intervención

Lugar	Jocotán, Chiquimula
Ubicación	Aldea la Mina, Jocotán, Chiquimula
MSNM (metros sobre el nivel del mar)	1472.00 m
Superficie territorial	7.00 km²

Demografía

Población de 0 a 14 años de edad	1679 personas
----------------------------------	---------------

Población de 15 a 64 años de edad	1328 personas
Población de 65 años o más	248 personas
Población mujeres	1617 mujeres
Población hombres	1638 hombres
Población Rural	3255 habitantes
Población Urbana	0 habitantes
Población Indígena	0 habitantes
Población Garífuna	0 habitantes
Población Xinca	0 habitantes

Economía

Porcentaje de pobreza	30.00 %
Porcentaje de pobreza extrema	70.00 %
Total de ingreso por familia	Q. 30.00

Observaciones

Salud

Tasa de acceso a atención primaria en salud	10.00 %
Tasa de enfermedades vinculadas al consumo de agua no apta y mala disposición de excretas	20.00 %
Tasa de desnutrición crónica en niños menores de 5 años	65.00 %
Enfermedades frecuentes en niños menores de 5 años	Parasitismo intestinal, Infecciones respiratorias, Síndrome diarreico agudo, Enfermedades transmitidas por vectores, Desnutrición aguda moderada o severa,

Educación

Porcentaje de aprobación de pruebas de matemática	
Nivel Primario	154.00 %
Nivel Secundario	34.00 %
Nivel Diversificado	13.00 %
Porcentaje de aprobación de pruebas de lectura	
Nivel Primario	154.00 %
Nivel Secundario	34.00 %
Nivel Diversificado	13.00 %

Medio ambiente

Tasa de cobertura forestal	12.00 %
Número de planes de gestión integral de desechos sólidos	0
Porcentaje de uso de la tierra para tareas agrícolas	62.00 %

Gestión municipal

Índice de gestión municipal	65
------------------------------------	----



This work is licensed under a Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International License.

Este trabajo está protegido bajo una licencia internacional Creative Commons: Atribución - Uso No Comercial - Compartir Igual.

Equipo Multidisciplinario

Alvarez Guerra, Ronald Gabriel

Ayala Guerra, Greysy Esmeralda

Castañeda Sánchez, Karla Alejandra

Gonzalez Martinez, Marvin Anibal

Klee Sosa, Jorge Eduardo

Melgar Urquía, Karla María

Oliva Vargas, José Carlos

Anexo 5. Resumen de proyecto – Ficha de Resultados EPSUM



PROYECTO MONODISCIPLINARIO

EQUIPO: MAZATENANGO, 1 COHORTE 2018
MAZATENANGO, SUCHITEPÉQUEZ

Nombre	Caracterización morfométrica de la subcuenca del río Chitá, Mazatenango, Suchitepéquez.
Ubicación	El proyecto se realizó en el Cantón Chitá, está ubicado a cuatro kilómetros al sur oeste de la ciudad de Mazatenango, tiene una extensión territorial de 19.8752 hectáreas, el nombre del Cantón Chitá se originó porque la comunidad se formó en la Finca Chitá cerca de la ribera del río Chita, el Cantón fue legalizado el 7 de enero de 1986, cuenta con una población de 915 habitantes, 141 casas habitadas, se encuentra ubicada, a 14°31'36.78" latitud norte y 91°31'46.65" longitud oeste, y a 334 msnm, limitado al
Objetivo General	Caracterizar morfométricamente la subcuenca del río Chitá.
Objetivos Específicos	• Determinar los aspectos lineales, superficiales y de relieve. • Elaborar un plan de actividades para disminuir los daños ocasionados por los desbordamientos e inundaciones de la subcuenca del río Chitá • Describir las condiciones biofísicas del Cantón Chitá.
Desarrollo Metodológico	Los aspectos lineales, de superficie y de relieve se determinaron utilizando hojas cartográficas, a escala 1:50,000, los coeficientes, se calcularon de acuerdo a Horton 1945, modificado por Strahler 1964, Schum, 1956 y Miller, 1953 e Ibáñez, 1995. Se ubicó el punto de aforo en las coordenadas 91°33'00" longitud oeste y 14°25'27" latitud norte, parte más baja de la subcuenca río Chitá, antes que se una al río Ican en el municipio de San Lorenzo, Suchitepéquez. Se trazó el parteaguas, a partir del punto de aforo, utilizando el comportamiento de las curvas a nivel y las partes altas de la cuenca hidrográfica. A partir del punto de aforo se trazó el cauce principal, utilizando un rapidógrafo. Seguidamente se midió la longitud del cauce principal con el método del hilo mojado, sobreponiendo el hilo, en el cauce desde el punto de aforo, hasta la parte más alta de la subcuenca, luego se calculó la longitud del hilo, con una regla, el resultado en centímetros se multiplicó por los kilómetros según la esc
Línea de Intervención	Vulnerabilidad, adaptabilidad al cambio climático y gestión de riesgo
Área de intervención	Ambiente
Políticas asociadas	Política de Conservación, protección y mejoramiento del ambiente y de los recursos naturales. Política Nacional del cambio Climático. Política Forestal de Guatemala Política Macro de Gestión Ambiental. Política Nacional y Estrategia para el Desarrollo del Sistema Nacional de Áreas Protegidas

- Metas**
- Calcular los aspectos lineales, superficiales y de relieve.
 - Realizar un plan con 3 actividades
 - Determinar el estado actual del Cantón Chitá.

Usuarios directos	13
Usuarios indirectos	915
Aporte económico técnico al proyecto	Q.18000

Anexo 6. Manual de Mantenimiento de Sistema Gestión de Estudiantes, Sedes y Equipos del Programa EPSUM

Manual de mantenimiento

DESCRIPCIÓN DEL SISTEMA

El sistema de gestión de estudiantes, sedes y equipos del Programa de Ejercicio Profesional Supervisado Multidisciplinario está codificado en el lenguaje de programación PHP, utilizando como base el *framework Laravel* en su versión LTS 5.1. *Laravel* es un *framework* de código abierto para desarrollar aplicaciones y servicios web con *PHP 5* y *PHP 7*.

Laravel tiene como objetivo ser un *framework* que permita el uso de una sintaxis elegante y expresiva para crear código de forma sencilla y permitiendo multitud de funcionalidades. Intenta aprovechar lo mejor de otros *frameworks* y aprovechar las características de las últimas versiones de *PHP*.

Gran parte de *Laravel* está formado por dependencias, especialmente de *Symfony*, esto implica que el desarrollo de *Laravel* dependa también del desarrollo de sus dependencias.

CARACTERÍSTICAS DE LARAVEL

- Sistema de ruteo, también *RESTful*
- *Blade*, Motor de plantillas
- Peticiones *Fluent*
- *Eloquent ORM*
- Basado en *Composer*
- Soporte para el *caché*
- Soporte para *MVC10*
- Usa componentes de *Symfony*
- Adopta las especificaciones *PSR-2* y *PSR-4*

REQUISITOS DEL SISTEMA

El sistema necesita un servidor PHP con los requerimientos siguientes:

- *PHP >= 5.5.9*
- *OpenSSL PHP Extension*
- *PDO PHP Extension*
- *Mbstring PHP Extension*
- *Tokenizer PHP Extension*
- *Git*
- *PHP 7.0*
- *HHVM*
- *Nginx*
- *MySQL*
- *MariaDB*
- *Composer*

INSTALACIÓN

Para instalar el software desde cero es necesario tener un proyecto *Laravel* base. En la carpeta pública del servidor creamos un proyecto de la manera siguiente:

```
composer create-project laravel/laravel _NAME_ "5.1.*"
```

Donde *_NAME_* será el nombre de la carpeta del proyecto. El nombre nos ayudará a conocer si *Laravel* funciona correctamente en el servidor, de lo contrario resolver los mensajes de error que aparecen.

REPOSITORIO GIT

Luego de tener un proyecto inicial insertamos el proyecto desarrollado, descomprimos el *rar* obtenido desde *BitBucket*, el repositorio GIT seleccionado para almacenar el software con el correo miepsum@gmail.com. Con los accesos concedidos al

personal del Programa EPSUM puede descargar el proyecto. Creará una carpeta llamada “*epsum*” donde se encuentra el proyecto *Laravel*.

CONFIGURACION .ENV

Luego realizamos las configuraciones dentro del archivo .env ubicado al interior de la carpeta *epsum* con los valores que corresponden al proyecto. Por seguridad de la aplicación se deja un ejemplo.

```
APP_ENV=local
APP_DEBUG=false
APP_KEY=SomeRandomString
```

```
DB_CONNECTION=mysql
DB_HOST=127.0.0.1
DB_DATABASE=homestead
DB_USERNAME=homestead
DB_PASSWORD=secret
```

```
CACHE_DRIVER=file
SESSION_DRIVER=file
QUEUE_DRIVER=sync
```

```
REDIS_HOST=127.0.0.1
REDIS_PASSWORD=null
REDIS_PORT=6379
```

```
MAIL_DRIVER=smtp
MAIL_HOST=mailtrap.io
MAIL_PORT=2525
MAIL_USERNAME=null
MAIL_PASSWORD=null
```

```
MAIL_ENCRYPTION=null
```

El real con las contraseñas se encuentra en el proyecto solo realizar los cambios en *APP_ENV* con la dirección IP del servidor, *DB_HOST* con la IP del servidor de base de datos, *DB_DATABASE* con el nombre de la base de datos *DB_USERNAME* con el usuario de la base de datos y *DB_PASSWORD* el *password* de la base de datos.

MIGRACIONES

Para poder generar las migraciones necesitamos una base de datos en *MySQL* con codificación *utf8_general_ci* con el nombre establecido en el archivo *env*. Un usuario y una contraseña para la aplicación.

Para crear migraciones dentro del sistema utilizamos el comando:

```
php artisan make:migration create__NAMETABLE__table
```

Donde *_NAMETABLE_* es el nombre de la tabla para realizar la migración, esto creará una estructura de tabla para la base de datos.

En el caso de los insert iniciales se utilizan los Seeding. Se creará un seed por tabla a insertar con la estructura predefinida, para crear el seed de una tabla utilizamos el comando:

```
php artisan make:seeder _SEEDERNAME_
```

Donde *_SEEDERNAME_* es un nombre que hace referencia a la tabla seleccionada. Para más información sobre migraciones y sembrado de datos utilizando artisan en <https://laravel.com/docs/5.1/seeding>.

La base de datos se encuentra configurada dentro del sistema con sus migraciones y también el sembrado de la información inicial, por lo que la estructura inicial y los datos iniciales se encuentran dentro de la aplicación. Independientemente de la base de datos a utilizar Laravel almacena las migraciones para efectuar cambios en base de datos únicamente con el cambio en el archivo .env y los siguientes comandos:

Iniciando las migraciones:

```
php artisan migrate
```

Algunas migraciones son destructivas para la base de datos, de ser necesario un reinicio de la aplicación utilizamos:

```
php artisan migrate:rollback
```

```
php artisan migrate -force
```

Para incluir las inserciones iniciales utilizamos el comando:

```
php artisan migrate:refresh --seed
```

Con esto tenemos configurado el entorno de datos correctamente. Para más información visite el sitio <https://laravel.com/docs/5.1/database>.

ACTUALIZACIÓN DEL SISTEMA

Cualquier cambio dentro del sistema debe de hacerse un commit en el repositorio destinado para el proyecto. Para clonar y tener el repositorio en una nueva máquina utilizamos:

```
git clone https://hxrxd@bitbucket.org/hxrxd/mieps.git
```

Hacemos los cambios necesarios en el código y ejecutamos las siguientes sentencias:

```
git add .  
git commit -m "Referencia del Commit"  
git push -u origin master
```

Con esto mantendremos la rama principal del proyecto y conservamos los nuevos cambios. Luego ingresamos al servidor definido para la aplicación por medio de la llave publica a la siguiente dirección:

```
c3.usac.edu.gt
```

Con la conexión establecida eliminamos la instancia actual de la aplicación, el usuario en el servidor no tiene permisos de escribir el repositorio solo de clonar:

```
rm -r epsum/ --force
```

Esto eliminará todas las carpetas del proyecto, luego clonamos el repositorio nuevamente:

```
git clone https://hxrxd@bitbucket.org/hxrxd/mieps.git
```

Generamos los permisos necesarios para las carpetas del proyecto y salimos del servidor:

```
chmod 777 -R epsum/  
exit
```

Con esto ya tenemos actualizada la aplicación.

ESTRUCTURA DEL PROYECTO

La estructura predeterminada de la aplicación *Laravel* está destinada a proporcionar un Excelente punto de partida para aplicaciones tanto grandes como pequeñas. Por supuesto, se puede organizar la aplicación como se dese. *Laravel* casi no impone restricciones sobre dónde se encuentra una clase determinada, siempre que *Composer* pueda auto cargar la clase.

El directorio raíz de una nueva instalación de *Laravel* contiene una variedad de carpetas:

app, contiene el código central de su aplicación.

bootstrap contiene algunos archivos que inician el framework y configuran la carga automática, así como también una cachecarpeta que contiene algunos archivos generados por el framework para la optimización del rendimiento del bootstrap.

config, contiene todos los archivos de configuración de su aplicación.

database, contiene su migración de base de datos y semillas. Si lo desea, también puede usar esta carpeta para contener una base de datos SQLite.

public, contiene el controlador frontal y sus activos (imágenes, JavaScript, CSS, etc.).

resources, contiene sus vistas, activos sin procesar (LESS, SASS, CoffeeScript) y archivos de localización.

storage, contiene plantillas Blade compiladas, sesiones basadas en archivos, cachés de archivos y otros archivos generados por el marco. Esta carpeta se segrega en **app**, **frameworky** **logsdirectorios**. **app** puede usarse para almacenar cualquier archivo utilizado

por su aplicación. framework se usa para almacenar archivos y cachés generados por el marco. Finalmente, el log contiene los archivos de registro de su aplicación.

test, contiene tus pruebas automáticas. Un ejemplo PHPUnit se proporciona de fábrica.

vendor, contiene tus dependencias de Composer.

RUTAS

Laravel propone en el desarrollo usar *“Routes with Closures”*, en lugar de un MVC tradicional con el objetivo de hacer el código más claro. Aun así, permite el uso de MVC tradicional. Las rutas contenidas dentro del software son las siguientes:

```
Route::get('/', 'HomeController@index');
Route::get('home', 'HomeController@index');
// Authentication routes...
Route::get('login', 'Auth\AuthController@login');
Route::post('login', 'Auth\AuthController@postLogin');

Route::get('logout', 'Auth\AuthController@logout');
// Registration routes...
Route::get('register', 'Auth\AuthController@register');
Route::post('register', 'Auth\AuthController@postRegister');
// Password reset link request routes...
Route::get('password/email', 'Auth\PasswordController@email');
Route::post('password/email', 'Auth\PasswordController@postEmail');
// Password reset routes...
Route::get('password/reset/{token}', 'Auth\PasswordController@getReset');
Route::post('password/reset', 'Auth\PasswordController@postReset');
Route::resource('municipality', 'MunicipalityController');
Route::resource('supervisor', 'SupervisorE');
Route::resource('users', 'UserController');
Route::resource('teams', 'TeamController');
Route::resource('headquarters', 'HeadquartersController');
Route::resource('student', 'StudentController');
Route::resource('cohortes', 'CohorteController');
Route::resource('referencias', 'ReferenciasController');
Route::resource('studentg', 'StudentGestorController');
Route::resource('agreement', 'AgreementController');
Route::resource('log', 'LogController');
Route::resource('contract', 'ContratosController');
Route::resource('gestor', 'GestorController');
Route::resource('payments', 'PaymentsController');
Route::resource('supervisorstudent', 'SupervisorStudentController');
Route::resource('statuses', 'StatusesController');
```

```

Route::get('genficha','StudentController@getFicha');
Route::get('gencarta','StudentController@getCarta');
Route::get('gencontrato','StudentController@getContrato');
Route::get('gcontract/{id}/download', 'GestorController@genContract')->name('gcontract.download');
Route::get('gpayment/{id}/download', 'PaymentsController@genPayment')->name('gpayment.download');
Route::get('forms05/{id}/download', 'GestorController@genForm05')->name('forms05.download');
Route::get('forms02/{id}/download', 'GestorController@genForm02')->name('forms02.download');
Route::get('ficha/{id}/download', 'StudentGestorController@getFicha')->name('ficha.download');
Route::get('settlement/{id}/download', 'SettlementController@createSettlement')->name('settlement.download');
//contratos
Route::get('gcontrato1/{id}/download', 'GestorController@genContract130')->name('gcontrato1.download');
Route::get('gcontrato15/{id}/download', 'GestorController@genContract1515')->name('gcontrato15.download');
Route::get('gcontrato1530/{id}/download', 'GestorController@genContract1530')->name('gcontrato1530.download');
Route::get('gform31/{id}/download', 'GestorController@genForm3130')->name('gform31.download');
Route::get('gform315/{id}/download', 'GestorController@genForm31515')->name('gform315.download');
Route::get('gform3131/{id}/download', 'GestorController@genForm3130')->name('gform3131.download');
Route::get('gform31531/{id}/download', 'GestorController@genForm31530')->name('gform31531.download');
//contratos estudiante
Route::get('contrato1','ContratosController@getContrato1');
Route::get('contrato15','ContratosController@getContrato15');
Route::get('contrato131','ContratosController@getContrato131');
Route::get('contrato1531','ContratosController@getContrato1531');
Route::get('contrato115','ContratosController@getContrato115');
Route::get('contrato15e','ContratosController@getContrato15e');
Route::get('contrato31e','ContratosController@getContrato31e');
//Pagos
Route::get('pay/{id}/index', 'PaymentsController@inicio')->name('pay.index');
Route::get('pay/{id}/createp', 'PaymentsController@createP')->name('pay.createp');
Route::get('forms06/{id}/download', 'PaymentsController@genForm06')->name('forms06.download');
Route::get('forms04/{id}/download', 'PaymentsController@genForm04')->name('forms04.download');
//Excel
Route::get('supervisorExcel','ExcelController@getStudentBySupervisor');
Route::get('gestorExcel','ExcelController@getStudent');
Route::get('getacuerdo','ExcelController@getAcuerdo');
Route::get('logExcel','ExcelController@getSucesos');
Route::get('studentcreate',function(){
    # code...
    return view('student.welcome');
});

```

```

Route::get('studentficha',function(){
    # code...

    return view('student.ficha');
});

Route::get('studentcarta',function(){
    # code...

    return view('student.carta');
});
Route::get('studentws','StudentController@studentFromServer');
Route::get('contractstudent','ContratosController@findStudent');
Route::get('createcontract','ContratosController@createContract');
// Función Ajax
Route::get('/information/create/ajax-state',function()
{
    $id_departament = Input::get('id_departament');
    $subcategories = Municipality::where('id_departament','=',$id_departament)->get();
    return $subcategories;
});

```

Las rutas get y post responden a transacciones RESTful de peticiones según se requiere en la ruta, mientras que las resource generan un entorno predefinido de implementado dentro del controlador con los métodos generados.

Para más información acceda al sitio: <https://laravel.com/docs/5.1/routing>

VISTAS

Las vistas contienen el HTML servido por su aplicación y separa la lógica de su controlador / aplicación de su lógica de presentación. Las vistas se almacenan en el directorio.

```
resources/views
```

Una vista simple podría verse de la siguiente forma:

```

<html>
  <body>
    <h1>Hello, <?php echo $name; ?></h1>
  </body>
</html>

```

En el caso de la aplicación se encuentra una carpeta por cada controlador por datos para crear, editar y modificar información. La carpeta *template* tiene la plantilla de *bootstrap inspira*, que es la base para cualquier *view*:

```

<!DOCTYPE html>
<html>
<head>
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width, initial-
scale=1.0">
  <title>MiE </title>

  {!!Html::style('css/bootstrap.min.css')!!}
  {!!Html::style('font-awesome/css/font-awesome.css')!!}

  {!!Html::style('css/animate.css')!!}
  {!!Html::style('css/style.css')!!}

  {!!Html::style('css/plugins/fullcalendar/fullcalendar.css')!!}
  {!!Html::style('css/plugins/fullcalendar/fullcalendar.print.css',
['media'=>'print'])!!}

  {!!Html::style('css/plugins/chosen/bootstrap-chosen.css') !!}
</head>
<body>

  <!-- Wrapper-->
  <div id="wrapper">

    <!-- Navigation -->
    @include('template/sidebar')
    <!-- Page wrapper -->
    <div id="page-wrapper" class="gray-bg">

      <!-- Page wrapper -->
      @include('template/header')

      <!-- Main view -->
      @yield('place')

      <!-- Footer -->
      @include('template/footer')

    </div>
  <!-- End page wrapper-->

```

```

        @include('template/scripts')

        @section('script')
        @show
    </body>
</html>

```

En el *yield* place se coloca el código de *HTML* que cambia de la plantilla principal, además incluye la barra de menú, la cabecera y el pie de la página. Además, incluye una sección *script* donde se colocan los *script* que usa esa vista en particular.

Se puede pasar datos fácilmente entre vistas así:

```
return view('greetings', ['name' => 'Victoria']);
```

Más información en <https://laravel.com/docs/5.1/views>

MODELOS

Para comenzar, creemos un modelo *Eloquent*. Los modelos generalmente viven en el directorio *app*, pero puede colocarlos en cualquier lugar que pueda cargarse automáticamente de acuerdo con su archivo. Todos los modelos *Eloquent* extienden la clase:

```
composer.json\Illuminate\Database\Eloquent\Model
```

La forma más fácil de crear una instancia de modelo es utilizando el comando *Artisan*, esto incluye la migración del modelo:

```
php artisan make:model _Model_ --migration
```

Donde *_MODEL_* es el modelo a crear

A continuación se muestra un ejemplo de la clase `Flight` como modelo, que se utilizará para recuperar y almacenar información de nuestra tabla `flights` de base de datos:

```
<?php
namespace App;

use Illuminate\Database\Eloquent\Model;

class Flight extends Model
{
    //
}
```

Tener en cuenta que no le indicamos a Eloquent qué tabla usar para nuestro `Flight` modelo. El "nake case", nombre plural de la clase se utilizará como el nombre de la tabla a menos que se especifique explícitamente otro nombre. Entonces, en este caso, Eloquent asumirá que el `Flight` modelo almacena registros en la `flightst` tabla. Puede especificar una tabla personalizada definiendo una `table` propiedad en su modelo:

```
<?php
namespace App;

use Illuminate\Database\Eloquent\Model;

class Flight extends Model
{
    /**
     * The table associated with the model.
     *
     * @var string
     */
    protected $table = 'my_flights';
}
```

Eloquent también supondrá que cada tabla tiene una columna de clave principal nombrada `id`. Puede definir una `$primaryKey` para anular esta convención. Por defecto, Eloquent crea el `created_at` y `updated_at` como columnas, de existir en sus tablas. Si no desea que Eloquent administre estas columnas automáticamente, configure la propiedad `$timestamps` en su modelo para falso:

```
<?php
namespace App;
```

```

use Illuminate\Database\Eloquent\Model;

class Flight extends Model
{
    /**
     * Indicates if the model should be timestamped.
     *
     * @var bool
     */
    public $timestamps = false;
}

```

Si necesita personalizar el formato de sus marcas de tiempo, configure la propiedad `$dateFormat` en su modelo. Esta propiedad determina cómo se almacenan los atributos de fecha en la base de datos, así como su formato cuando el modelo se serializa en una matriz o *JSON*:

```

<?php

namespace App;

use Illuminate\Database\Eloquent\Model;

class Flight extends Model
{
    /**
     * The storage format of the model's date columns.
     *
     * @var string
     */
    protected $dateFormat = 'U';
}

```

Para interactuar con la base de datos *Laravel* utiliza *Eloquent* como lenguaje de interacción.

Para más información, ubicar el sitio Web de documentación de *Eloquent* en:
<https://laravel.com/docs/5.1/eloquent>

CONTROLADORES

En lugar de definir toda su lógica de manejo de solicitudes en un solo archivo, es posible organizar este comportamiento utilizando las clases de *Controller*. Los controladores pueden agrupar la lógica de manejo de solicitudes *HTTP* relacionada en una clase. Los controladores generalmente se almacenan en el directorio *routes.php* *Http/Controllers*

A continuación, se muestra un ejemplo de una clase de controlador básico. Todos los controladores de *Laravel* deben ampliar la clase de controlador base incluida con la instalación predeterminada de *Laravel*:

```
<?php

namespace App\Http\Controllers;

use App\User;
use App\Http\Controllers\Controller;

class UserController extends Controller
{
    /**
     * Show the profile for the given user.
     *
     * @param int $id
     * @return Response
     */
    public function showProfile($id)
    {
        return view('user.profile', ['user' => User::findOrFail($id)]);
    }
}
```

Podemos dirigir la acción del controlador de la siguiente manera:

```
Route::get('user/{id}', 'UserController@showProfile');
```

Es muy importante tener en cuenta que no necesitamos especificar el espacio de nombres completo del controlador al definir la ruta del controlador. Solo definimos la porción del nombre de clase que viene después del espacio de nombres "raíz". De forma

predeterminada, cargará el archivo dentro de un grupo de ruta que contenga el espacio de nombres del controlador raíz.

```
app\Http\ControllersRouteServiceProviderroutes.php
```

Los controladores de recursos facilitan la creación de controladores *RESTful* alrededor de los recursos. Por ejemplo, es posible que desee crear un controlador que maneje las solicitudes *HTTP* relacionadas con "Photo" almacenadas por su aplicación. Usando el comando *Artisan*, podemos crear rápidamente un controlador de este tipo:

```
php artisan make:controller PhotoController
```

El comando *Artisan* generará un archivo de controlador. El controlador contendrá un método para cada una de las operaciones de recursos disponibles.

```
app/Http/Controllers/PhotoController.php
```

Esta declaración de ruta única crea múltiples rutas para manejar una variedad de acciones *RESTful* en el recurso de la foto. Del mismo modo, el controlador generado ya tendrá métodos restringidos para cada una de estas acciones, incluidas notas que le informan qué URI y verbos manejan.

Metodo	Ruta	Función	Nombre de la ruta
GET	/photo	index	photo.index
GET	/photo/create	create	photo.create
POST	/photo	store	photo.store
GET	/photo/{photo}	show	photo.show
GET	/photo/{photo}/edit	edit	photo.edit
PUT/PATCH	/photo/{photo}	update	photo.update
DELETE	/photo/{photo}	destroy	photo.destroy

Al declarar una ruta de recursos, puede especificar un subconjunto de acciones para manejar en la ruta:

```
Route::resource('photo', 'PhotoController', ['only' => ['index',  
'show']]);
```

```
Route::resource('photo', 'PhotoController', ['except' => ['create',  
'store', 'update', 'destroy']]);
```