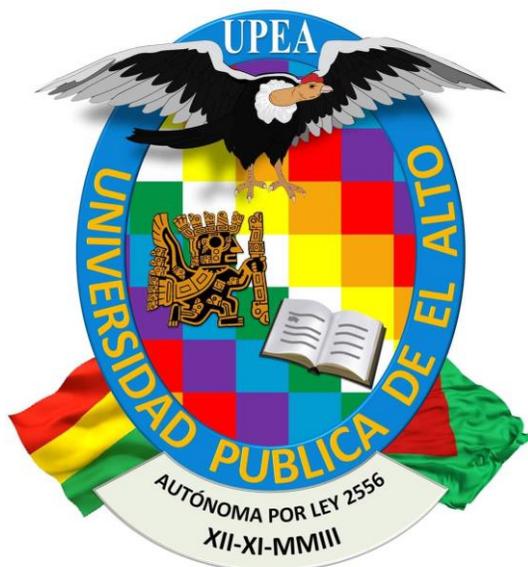


UNIVERSIDAD PÚBLICA DE EL ALTO

CARRERA INGENIERÍA DE SISTEMAS



PROYECTO DE GRADO

**“SISTEMA DE INFORMACIÓN PARA EL CONTROL Y SEGUIMIENTO
DE ENTREGA DE CANASTAS ESTUDIANTILES EN LA CIUDAD DE
EL ALTO”**

CASO: DIRECCIÓN DE EDUCACIÓN – GAMEA

Para Optar al Título de Licenciatura en Ingeniería de Sistemas

MENCIÓN: INFORMÁTICA Y COMUNICACIONES

Postulante: Jose Jossue Hilari Vega

Tutor Metodológico: Ing. Maricel Yarari Mamani

Tutor Especialista: Ing. Enrique Flores Baltazar

Tutor Revisor: Lic. Freddy Salgueiro Trujillo

EL ALTO – BOLIVIA

2020

DEDICATORIA

A mis padres, Andres Hilari y Graciela Vega por el apoyo incondicional que siempre me dieron y la educación que ellos me inculcaron, ya que todo aquello provocó en mí la responsabilidad y los deseos de superación, que fueron fundamentales para la construcción de mi vida profesional.

AGRADECIMIENTO

A Dios, por darme fuerza y salud que me da para seguir adelante.

A mi familia, por el apoyo que me brindo a lo largo de toda mi vida, e impulsarme a seguir y no rendirme.

A mis tutores; metodológico, especialista y revisor: Ing. Maricel Yarari Mamani, Ing. Enrique Flores Baltazar y Lic. Freddy Salgueiro Trujillo, quienes me guiaron paso a paso durante el proceso con sus observaciones y sugerencias para el desarrollo y conclusión del presente proyecto de grado.

A mis compañeros de carrera por su amistad y apoyo moral durante la vida universitaria, gracias a todos

RESUMEN

El presente proyecto titulado “Sistema de información para el control y seguimiento de entrega de canastas estudiantiles en la ciudad de El Alto Caso: Dirección de Educación – GAMEA”, surge en necesidad de la institución, para realizar la entrega de canastas que se determinó por decreto municipal, en pedido de los padres de familia, para que puedan cubrir el equivalente al desayuno escolar, que corresponde a la gestión escolar de cada año y que por razones de bioseguridad se determine la clausura del año escolar, quedando pendiente la repartición del mismo.

En respuesta a la solicitud de contar con un sistema que ayude en el procesamiento de los datos que se recolectaron sobre los beneficiarios, se realizó una serie de procesos que permitió realizar el desarrollo e implementación de dicho sistema, para que ayude a la institución en la repartición y entrega de canastas y así también se pueda contemplar resultados de carácter informativo y que respalden la labor realizada una vez sistematizada y contabilizada en el término de la labor.

Para el desarrollo del sistema se aplicó el uso de herramientas que fueron utilizadas en distintas fases, como el uso de la metodología UWE para el desarrollo de sistema basado en la web, también se requirió de un gestor de base de datos, como mariadb, que es una herramienta útil en el proceso simplificado de tratamiento de datos, además del lenguaje de PHP y framework de desarrollo como es Laravel, además cabe mencionar el uso de Ajax jquery para el manejo de información en la parte del frontEnd.

Así mismo el sistema está basado en estándares y parámetros ideales, como las normas de calidad ISO 25000, parámetros de seguridad basados en la ISO 27000, y la evaluación de costos COCOMO II, todo esto permitirá brindar al sistema un mejor producto para la institución

ÍNDICE DE CONTENIDO

1. MARCO PRELIMINAR	1
1.1. INTRODUCCIÓN.....	1
1.2. ANTECEDENTES	2
1.2.1. Antecedentes de la institución.....	2
1.2.2. Antecedentes académicos.....	2
1.3. PLANTEAMIENTO DEL PROBLEMA.....	3
1.3.1. Problema principal.....	3
1.3.2. Problemas secundarios.....	4
1.4. OBJETIVOS	4
1.4.1. Objetivo general.....	4
1.4.2. Objetivos específicos	5
1.5. JUSTIFICACIONES	5
1.5.1. Justificación Técnica.....	5
1.5.2. Justificación Económica	6
1.5.3. Justificación Social.....	6
1.6. METODOLOGÍA.....	6
1.6.1. UWE	6
1.7. MÉTODOS DE INGENIERÍA.....	7
1.8. MÉTODOS.....	8
1.8.1. Modelo de costos.....	8
1.8.2. Gestión de calidad.....	8
1.8.3. Seguridad.....	9
1.9. HERRAMIENTAS	10
1.9.1. Lenguaje	10
1.9.2. Base de Datos.....	10
1.9.3. Servidor	10
1.9.4. Arquitectura	11
1.9.5. Framework.....	11
1.9.6. Ajax jquery	11
1.9.7. Modelado	12
1.10. LIMITES Y ALCANCES.....	12

1.10.1.	Limites	12
1.10.2.	Alcances.....	12
1.11.	APORTES.....	13
2.	MARCO TEÓRICO.....	14
2.1.	INTRODUCCIÓN.....	14
2.2.	SISTEMA	14
2.2.1.	El termino sistema.....	14
2.2.2.	Sistema	15
2.3.	DATOS.....	15
2.4.	INFORMACIÓN	16
2.4.1.	Tipos de información.....	16
2.5.	CONTROL Y SEGUIMIENTO	17
2.6.	CONTROL DE ENTREGAS.....	18
2.7.	OPTIMIZACIÓN.....	18
2.8.	GESTIÓN DE REPORTEES	20
2.9.	METODOLOGÍA DE DESARROLLO	20
2.10.	UWE	21
2.10.1.	Visión general.....	21
2.10.2.	Modelos de UWE.....	22
2.10.3.	Fases o etapas de UWE.....	26
2.11.	INGENIERÍA DE SOFTWARE	27
2.11.1.	Modelos de proceso de software	28
2.12.	HERRAMIENTAS	28
2.12.1.	HTML.....	28
2.12.2.	PHP.....	29
2.12.3.	Base de datos MariaDB.....	30
2.12.4.	Servidor HTTP Apache.....	31
2.12.5.	Arquitectura MVC	32
2.12.6.	Framework Laravel.....	33
2.12.7.	Ajax JQuery	34
2.12.8.	Modelado Navicat.....	35
2.13.	PRUEBAS DE SOFTWARE	36

2.13.1.	Tipos de pruebas	36
2.14.	MÉTRICAS DE CALIDAD	37
2.14.1.	ISO/IEC 25000	37
2.15.	SEGURIDAD	39
2.15.1.	Norma ISO 27000.....	39
2.16.	MODELO DE COSTOS.....	42
2.16.1.	COCOMO II	42
3.	MARCO APLICATIVO	44
3.1.	INTRODUCCIÓN.....	44
3.2.	ESQUEMA DEL SISTEMA	44
3.3.	DESARROLLO DE MODELADO MEDIANTE UWE.....	45
3.3.1.	Obtención de requisitos.....	45
3.3.2.	Análisis de requerimientos	49
3.3.3.	Diseño conceptual	55
3.3.4.	Diseño de navegación	56
3.3.5.	Diseño de presentación.....	58
3.3.6.	Implementación del sistema	62
4.	CONTROL DE CALIDAD Y ESTIMACIÓN DE COSTOS.....	79
4.1.	MÉTRICAS DE CALIDAD DE SOFTWARE.....	79
4.1.1.	La Norma ISO 25000	79
4.1.2.	Resultados de la aplicación de la matriz de calidad de uso	82
4.2.	EVALUACIÓN DE COSTOS	92
4.2.1.	Modelo COCOMO II.....	92
4.2.2.	Costos del sistema	93
5.	PRUEBAS Y SEGURIDAD DE SOFTWARE.....	97
5.1.	PRUEBAS DE SOFTWARE	97
5.1.1.	Pruebas de caja blanca	97
5.1.2.	Pruebas de caja negra.....	98
5.1.3.	Pruebas de estrés	100
5.2.	SEGURIDAD	101
5.2.1.	Norma ISO 27000.....	101
6.	CONCLUSIONES Y RECOMENDACIONES.....	103

6.1. CONCLUSIONES.....	103
6.2. RECOMENDACIONES.....	103
BIBLIOGRAFÍA.....	105
ANEXOS.....	109

ÍNDICE DE FIGURAS

Figura 1 Grafica general de sistema	15
Figura 2 Uso de la información.....	16
Figura 3 Descripción de modelos UWE	22
Figura 4 Casos de uso.....	23
Figura 5 Modelo de contenido	24
Figura 6 Diagrama de modelo de navegación	24
Figura 7 Modelo de presentación.....	25
Figura 8 Fases de la metodología UWE	26
Figura 9 Estructura básica de HTML.....	29
Figura 10 Descripción de uso de PHP	29
Figura 11 Composición MVC	32
Figura 12 Funcionamiento con Ajax	34
Figura 13 Divisiones para la calidad ISO 25000	37
Figura 14 Familia norma ISO 27000	39
Figura 15 Gestión de seguridad de la información.....	40
Figura 16 Triada de la información	41
Figura 17 Esquema del sistema.....	44
Figura 18 Diagrama de caso de uso (General).....	50
Figura 19 Diagrama de caso de uso: Administración del sistema.....	51
Figura 20 Diagrama de caso de uso: Control y reporte de entrega	52
Figura 21 Diagrama de caso de uso: Registro de entregas.	53
Figura 22 Diagrama de caso de uso: Consulta y validación de datos	54
Figura 23 Diagrama de clases.....	55
Figura 24 Modelo de navegación: Administrador	56
Figura 25 Modelo de navegación: Supervisor	56
Figura 26 Modelo de navegación: Responsable de entrega	57
Figura 27 Modelo de navegación: Estudiante.....	57
Figura 28 Modelo de presentación: Portada y consultas	58
Figura 29 Modelo de presentación: Inicio de sesión.....	58
Figura 30 Modelo de presentación: Información general.....	59
Figura 31 Modelo de presentación: Perfil de usuario	59

Figura 32 Modelo de presentación: Administración de unidad	60
Figura 33 Modelo de presentación: Administración de estudiantes	60
Figura 34 Modelo de presentación: Administración de tutores	61
Figura 35 Modelo de presentación: Administración de entregas	61
Figura 36 Modelo de presentación: Administración de reportes	62
Figura 37 Grafo del sistema	97
Figura 38 Prueba de autenticación de usuario	99
Figura 39 Pruebas de validación de formulario	99
Figura 40 Encriptado	101

ÍNDICE DE TABLAS

Tabla 1 Metodologías de desarrollo de software	21
Tabla 2 Familia ISO IEC 25000	38
Tabla 3 Tarea para obtener requisitos.....	45
Tabla 4 Listado de actores	45
Tabla 5 Categoría de funciones.....	46
Tabla 6 Requisitos funcionales	47
Tabla 7 Caso de uso: Administración del sistema.....	51
Tabla 8 Caso de uso: Control y reporte de entrega	53
Tabla 9 Caso de uso: Registro de entrega	54
Tabla 10 Caso de uso: Consulta y validación de datos	55
Tabla 11 Especificación formal de las métricas de calidad en uso.....	81
Tabla 12 Rangos de puntuación para las métricas	81
Tabla 13 Ponderación de características de calidad en uso.....	82
Tabla 14 Factores de ponderación.....	83
Tabla 15 Ajuste de complejidad.....	84
Tabla 16 Datos de la métrica Efectividad de la tarea.....	87
Tabla 17 Preguntas sobre el uso del sistema	88
Tabla 18 Información requerida por el IMS.....	90
Tabla 19 Calidad global del sistema	92
Tabla 20 Constantes a b c d COCOMO II.....	93
Tabla 21 Muestra de relación para convertir el valor de PF a KLDC	94
Tabla 22 Costos de recursos empleados para la elaboración del sistema	95
Tabla 23 Costos total del sistema	96
Tabla 24 Pruebas de caja negra.....	100

CAPITULO I
MARCO PRELIMINAR

1. MARCO PRELIMINAR

1.1. INTRODUCCIÓN

En la actualidad gracias al avance de la tecnología encontramos que la mayoría de las instituciones, trabajan con sistemas web, debido a que éstos facilitan el trabajo brindando una información rápida y segura para un mejor control del trabajo que se realiza, haciendo que las instituciones sigan avanzando en el ámbito sistemático y estructurado sobre el manejo de la información relevante dentro de la institución y la tecnología que implementa. (Mashiel, 2017)

Los sistemas pueden tomar protagonismo cuando se trata de controlar la información, como un respaldo o descargo, a la hora de justificar los proyectos al cual este dirigido el proyecto. Esto también habla de respaldar con datos la ejecución de planes institucionales y más cuando se trata de un plan de magnitud como la de un municipio, ya que la información se considerará como un respaldo estadístico que contribuya a la mejora de trabajo que será sistematizado y estructurado en forma automática y sin mucha intervención de externos. (Patricio, 2017)

Con el sistema se pretende mejorar las deficiencias que existan, como el de no contar con una base de datos sobre los registros de los estudiantes para que se proceda a realizar la entrega bajo un módulo de asignación, también sobre el seguimiento del cual solo se tiene en registros las unidades educativas pero no así los promedios de estudiantes tomados en cuenta, así mismo la estimación en tiempos adecuados para las entregas por unidad educativa en la Ciudad de El Alto. Por lo cual este sistema tendrá características basados en un seguimiento de entregas que permitan optimizar el trabajo bajo una planificación estructurada según los registros, para así poder controlar y respaldar los resultados obtenidos en todo el proceso de la entrega total.

Se pretende usar la tecnología web para dar una solución que permita registrar, controlar, reportar y confirmar entregas en las distintas fases que atravesase el sistema y este a su vez realizara un seguimiento de forma dinámica, con la metodología UWE se lograra el desarrollo del sistema de una manera más fluida con la arquitectura cliente servidor, pero que a su vez sea fácil de acceder, para los registros de las

funciones que se realizaran, así mismo se utilizara un gestor de base de datos adecuado a la cantidad de registros y como sugerencia principal se considera Mariadb por la facilidad en su uso y además de un servidor web Apache, todo esto basado en la influencia de la tecnología Open Source que es un código diseñado de manera que sea accesible al público.

1.2. ANTECEDENTES

1.2.1. Antecedentes de la institución

El Gobierno Autónomo Municipal de El Alto (G.A.M.E.A.) es una entidad vanguardista, moderna y competitiva que ejerce en la prestación de servicios públicos y mejorar la calidad de vida al promover el desarrollo integral en sus habitantes y entorno, reconociendo y respetando su diversidad intercultural con el talento humano que se encuentra en la autonomía de la Ciudad de El Alto. (gamea, 2020)

La Dirección de Educación dependiente de la Secretaria Municipal de desarrollo humano del G.A.M.E.A. tiene como objetivo de atender a las Unidades Educativas y brindar las condiciones necesarias para el buen desempeño de la educación en sus diferentes niveles, en la Ciudad de El Alto.

Tiene como misión mejorar la calidad de educación al atender las necesidades de las unidades educativas, prestando todos los servicios posibles para el desarrollo normal de la educación en la ciudad de El Alto. (Ramiro, 2018)

1.2.2. Antecedentes académicos

Internacional

- [Leticia Isabel Davalos Valle, 2017]” EFECTO DE UN SISTEMA WEB PARA EL CONTROL Y SEGUIMIENTO DE PROYECTOS DE TESIS EN LA ESCUELA ACADÉMICA PROFESIONAL DE INGENIERÍA EN INFORMÁTICA” tiene como objetivo, determinar el efecto del sistema web para el control y seguimiento de proyectos. de Ingeniería en Informática y Sistemas de la UNJBG. realizado en la Universidad Nacional Jorge Basadre Grohmann de Tacna, Perú el año 2017.

- [Raúl Alexander Gómez Sandoval, 2016]” SISTEMA DE INVENTARIOS PARA EL CONTROL DE MATERIALES, EQUIPOS Y HERRAMIENTAS DENTRO DE LA EMPRESA DE CONSTRUCCIÓN INGENIERÍA SÓLIDA LTDA” tiene como objetivo desarrollar un sistema de inventarios en la empresa Ingeniería Sólida Ltda. Para la gestión eficiente de los materiales, equipos y herramientas para su operación, como principal control de sus materias primas, realizado en la Universidad Libre de Bogotá el año 2016.

Nacional

- [Franklin Patricio Rivero Gonzales, 2017] ” SISTEMA WEB DE CONTROL Y SEGUIMIENTO DE OBRAS MUNICIPALES PARA LA SUBALCALDÍA DE OVEJUYO D-I MUNICIPIO DE PALCA”, tiene por objetivo Implementar un Sistema Web de Control y Seguimiento de Obras Municipales para la Sub alcaldía de Ovejuyo D-I Municipio de Palca, que provea a las autoridades instancias técnicas responsables operacionales de base de información de manera eficiente y verídica para la toma de decisiones. Utilizando metodología RUP y OMG, realizado en la Universidad Mayor de San Andrés el año 2017
- [Maddizon Mashiel Camacho Lugones, 2017] ” CONTROL DE PRODUCCIÓN, PEDIDOS, ENTREGAS E INVENTARIOS CASO: PHOVIDA”, tiene por objetivo desarrollar un sistema web de control de producción, pedidos, entregas e inventarios para la empresa PHOVIDA, mejorando el seguimiento de los anteriores para conseguir un trabajo más óptimo. Utilizando las metodologías OpenUp y UWE, realizado en la Universidad Mayor de San Andrés el año 2017.

1.3. PLANTEAMIENTO DEL PROBLEMA

1.3.1. Problema principal

La Dirección de Educación del G.A.M.E.A tiene como labor la entrega de Canastas Estudiantiles en las unidades educativas de la Ciudad de El Alto, para dicha labor necesita tener registrada una base de datos, a los beneficiarios que recibirán las entregas, que se calcula lleguen a ser alrededor de 300.000 estudiantes, para que después puedan ser sistematizados en la entrega de canastas a las casi 450 unidades

educativas de la ciudad de El Alto, también se requiere de documentación oficial sistematizada que ayude en el seguimiento sobre los resultados obtenidos una vez terminada la labor, el hecho de no contar con dichos resultados generara contratiempos e inconsistencia de datos para justificar los descargos por parte de la dirección de educación hacia el gobierno municipal.

1.3.2. Problemas secundarios

- Burocratización en todo el proceso y recorrido dentro de la dirección de educación, lo cual genera inconsistencia de datos, desorden y falta de coordinación.
- Falta de planificación en entregas, debido a que no se cuenta con una base de datos sobre la estimación total de beneficiarios.
- Búsqueda manual y tardía en hojas impresas para las entregas, lo que provocara contratiempos en búsqueda de registros.
- Difícil acceso a la información por parte de los beneficiarios contemplados en el plan institucional que causa incertidumbre sobre los datos registrados.
- Falta de seguridad en los procesos como parte de una mala planificación.
- Escases de reportes e inventarios que justifiquen el trabajo que se realizó, lo cual generara contratiempos en estimación y validación sobre los resultados obtenidos en todo el proceso de entrega.

1.4. OBJETIVOS

1.4.1. Objetivo general

Desarrollar e implementar un Sistema de Información web que permita hacer el Control y seguimiento en la entrega de canastas estudiantiles a las unidades educativas de la Ciudad de El Alto, que ayuda a optimizar el trabajo bajo un sistema de entregas y a su vez pueda generar todo tipo de reportes posibles con los resultados obtenidos, e información útil que servirá como respaldo del trabajo realizado por parte de la dirección de educación.

1.4.2. Objetivos específicos

- Analizar el manejo de recolección de datos en registros proporcionados por parte de las Unidades Educativas de la Ciudad de El Alto.
- Planificar un manejo sistematizado para la entrega de canastas.
- Diseñar la interfaz Front-End del sistema orientado a la Web con mecanismos de validación de datos en lenguaje PHP.
- Programar una interfaz de consultas que permita revisar datos del estudiante.
- Desarrollar la interfaz Back-End con ajustes a un diseño simplificado.
- Implementar módulos para el registro, validación y búsqueda en un gestor de base de datos MariaDB,
- Implementar módulos de seguridad para los niveles de acceso, que contemplen características especiales en el uso del sistema como por ejemplo; Admin, Supervisor y responsable en entregas.
- Generar reportes dinámicos, especializados en la información del historial y estadística del plan institucional.
- Realizar pruebas de manejo en las fases de diseño e implementación.
- Sistematizar informes que contemplen datos estadísticos.

1.5. JUSTIFICACIONES

1.5.1. Justificación Técnica

Actualmente la dirección de educación cuenta con la infraestructura adecuada para el desarrollo e implementación de un sistema web, tanto en Hardware como en Software, por lo cual no será necesario depender de nuevas tecnologías basadas en sistemas web, solo será necesario implementar un sistema de desarrollo que ayude en la labor.

Para este proyecto se pretende usar software de acceso libre y gratuito, como lenguaje PHP, MariaDB como un gestor de base de datos y Apache servidor web, de tal forma que nos ayude a desarrollar un sistema que no necesitara de tecnología avanzada y de equipos modernos.

1.5.2. Justificación Económica

Cada organización pública o privada tiene la necesidad de abaratar costos administrativos, ejecutar labores con los recursos necesarios, la ejecución del proyecto optimizara el control de entregas en forma automática, logrando con esto reducir la intervención de recurso humano para realizar la elaboración de planillas de la institución. El software a utilizar en el desarrollo del proyecto no necesita licencia, por lo que es más económico al momento de la implementación, además será desarrollado en un lenguaje de programación orientado a objetos basados en Web, con un gestor de base de datos fiable y estarán sobre un servidor con un sistema operativo Linux, los mismos que tienen la característica de ser libre y gratuito, esto garantizara que el proyecto esté basado en la ley de comunicaciones sobre el uso de Software libre.

1.5.3. Justificación Social

En una investigación deductiva, se detectó la necesidad de contar con un sistema interno que ayude con los registros obtenidos, para el control y seguimiento en las entregas de canastas estudiantiles, que nace a partir de un pedido de los padres de familia, que por normas del estado dictaminaran clausura del año escolar debido a una pandemia que afecto a nivel nacional, esto implica dar soluciones al pedido que se le hizo al gobierno municipal mediante su derivado la Dirección de educación de la secretaria municipal, que necesitara dicho sistema no solo para realizar entregas sino para respaldar con toda la información disponible mediante reportes e inventarios del desarrollo normal del plan institucional que haya sido encargada.

Este sistema web beneficiara no solo a los administrativos, secretarios, personal de trabajo sino será de gran ayuda a las Unidades Educativas parte de Gobierno Municipal de El Alto, del cual servirá con funciones de consultas y revisión de datos relevantes por parte de los estudiantes, mediante el uso de un sitio web.

1.6. METODOLOGÍA

1.6.1. UWE

Según Ludwig Maximilians de la Universidad de München cito “UWE es un proceso del desarrollo para aplicaciones Web enfocado sobre el diseño sistemático, la personalización y la generación semiautomática de escenarios que guíen el proceso

de desarrollo de una aplicación Web. UWE describe una metodología de diseño sistemática, basada en las 11 técnicas de UML, la notación de UML y los mecanismos de extensión de UML.

En el marco de UWE es necesario la definición de un perfil UML (extensión) basado en estereotipos con este perfil se logra la asociación de una semántica distinta a los diagramas del UML puro, con el propósito de acoplar el UML a un dominio específico, en este caso, las aplicaciones Web.

1.6.1.1. Modelos UWE

- Modelo de Requerimientos.
- Modelo Conceptual para el contenido (modelo de contenido).
- Modelo de Navegación que comprende la estructura de la navegación.
- Modelo de Presentación que abarca modelos estáticos y dinámicos.
- Modelo de proceso. (WWW02)

1.6.1.2. Fases UWE

- Captura, análisis y especificación de requisitos
- Diseño del sistema
- Codificación del software
- Pruebas
- La Instalación o Fase de Implementación
- El Mantenimiento. (WWW03)

1.7. MÉTODOS DE INGENIERÍA

Observación

La observación consiste en saber seleccionar aquello que queremos analizar. Se suele decir que "Saber observar es saber seleccionar". La observación científica "tiene la capacidad de describir y explicar el comportamiento, al haber obtenido datos adecuados y fiables correspondientes al análisis de requerimientos, conductas, eventos y /o situaciones perfectamente identificadas e insertas en el marco del análisis del sistema. (WWW04)

Entrevistas

La entrevista, es la comunicación interpersonal establecida entre el investigador y el sujeto de estudio a fin de obtener respuestas verbales a los interrogantes planteados sobre el problema propuesto. Se considera que este método es más eficaz que el cuestionario, ya que permite obtener una información más completa. A través de ella se pretende investigar la factibilidad del proyecto y especificar claramente, si el diseño es adecuado o podría variar en la metodología. (WWW05).

1.8. MÉTODOS

1.8.1. Modelo de costos

Modelo COCOMO II, es un modelo de estimación que se encuentra en la jerarquía de modelos de estimación de software. Es una herramienta basada en las líneas de código la cual la hace muy poderosa para la estimación de costos y no como otros que solamente miden el esfuerzo en base al tamaño. (WWW07)

1.8.2. Gestión de calidad

ISO/IEC 25000, es una familia de normas que tiene por objetivo la creación de un marco de trabajo común para evaluar la calidad del producto software, esta norma es el resultado de la evolución de otras normas anteriores, especialmente de las normas ISO/IEC 9126, que describe las particularidades de un modelo de calidad del producto software, e ISO/IEC 14598, que abordaba el proceso de evaluación de productos software. Esta familia de normas ISO/IEC 25000 se encuentra compuesta por cinco divisiones. (ISO25000, 2020)

ISO/IEC 2500n – División de Gestión de Calidad

Definen todos los modelos, términos y definiciones comunes referenciados por todas las otras normas de la familia 25000

ISO/IEC 2501n – División de Modelo de Calidad

Presentan modelos de calidad detallados incluyendo características para calidad interna, externa y en uso del producto software

ISO/IEC 2502n – División de Medición de Calidad

Incluyen un modelo de referencia de la medición de la calidad del producto, definiciones de medidas de calidad (interna, externa y en uso) y guías prácticas para su aplicación

ISO/IEC 2503n – División de Requisitos de Calidad

Ayudan a especificar requisitos de calidad que pueden ser utilizados en el proceso de elicitación de requisitos de calidad del producto software a desarrollar o como entrada del proceso de evaluación

ISO/IEC 2504n – División de Evaluación de Calidad

Incluye normas que proporcionan requisitos, recomendaciones y guías para llevar a cabo el proceso de evaluación del producto software.

1.8.3. Seguridad

La norma ISO 27000 es un conjunto de estándares internacionales sobre la Seguridad de la Información. La familia ISO 27000 contiene un conjunto de buenas prácticas para el establecimiento, implementación, mantenimiento y mejora de Sistemas de Gestión de la Seguridad de la Información. (Intedy, s.f.)

ISO 27001

La ISO 27001 sustituye a la BS 7799-2 estableciendo unas condiciones de adaptación para aquellas empresas que se encuentren certificadas bajo esta última.

ISO 27002

En ella podemos encontrar 39 objetivos de control y 133 controles agrupados en 11 dominios diferentes.

ISO 27003

Es un manual para implementar un Sistema de Gestión de Seguridad de la Información y además, nos da la información necesaria para la utilización del ciclo PHVA

ISO 27004

En este estándar se especifican las técnicas de medida y las métricas que son aplicables a la determinación de la eficacia de un Sistema de Gestión de Seguridad de la Información.

1.9. HERRAMIENTAS

1.9.1. Lenguaje

El lenguaje de programación determinado para este proyecto es PHP 7.1.3 ya que tiene gran aceptación en los diferentes frameworks y aun es gratuito para su diferente uso.

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 muchos sistemas operativos y plataformas sin ningún costo. (PHP, 2020)

1.9.2. Base de Datos

La base de datos que se empleara es Mysql o en su derivada MariaDB 10.3, ya que termina siendo libre y de costo gratuito, además de su facilidad de uso en sistemas web. MariaDB es un sistema de gestión de bases de datos derivado de MySQL con licencia GPL (General Public License). Introduce dos motores de almacenamiento nuevos, uno llamado Aria, que reemplaza a MyISAM y otro llamado XtraDB, en sustitución de InnoDB. Tiene una alta compatibilidad con MySQL ya que posee las mismas órdenes, interfaces, API y bibliotecas, siendo su objetivo poder cambiar un servidor por otro directamente. (Urutiaga, 2018)

1.9.3. Servidor

El servidor HTTP que se pretende usar es Apache 2.4. Debido a su familiaridad con php, que son especializados en sistemas web y además que es de uso libre y gratuito. El servidor HTTP Apache es un servidor web HTTP de código abierto, para plataformas Unix , Microsoft Windows, Macintosh y otras, que implementa el protocolo HTTP/1.1 y la noción de sitio virtual según la normativa RFC 2616.

Apache tiene amplia aceptación en la red desde 1996, Apache es el servidor HTTP más usado. Jugó un papel fundamental en el desarrollo de la World Wide Web y alcanzó su máxima cuota de mercado en 2005, siendo el servidor empleado en el 70% de los sitios web en el mundo. (B., 2018)

1.9.4. Arquitectura

El patrón de arquitectura MVC es un patrón que define la organización independiente del modelo, la vista y el controlador. El patrón de arquitectura "MVC", es una filosofía de diseño de aplicaciones, compuesta por modelo de datos, vista de interfaz de usuario y controlador que comunica al modelo con las vistas, además facilitara una interacción más fluida en el proceso compuesto de cliente – servidor para que las tareas se reparten entre los proveedores de recursos o servicios, llamados servidores, y los demandantes, llamados clientes. (Pavon M., 2008)

1.9.5. Framework

Laravel es uno de los frameworks de código abierto más fáciles de asimilar para PHP, fue creado en 2011 y tiene una gran influencia de frameworks como Ruby on Rails, Sinatra y ASP.NET MVC, el objetivo de Laravel es el de ser un framework que permita el uso de una sintaxis refinada y expresiva para crear código de forma sencilla, evitando el “código espagueti” y permitiendo multitud de funcionalidades. Aprovecha todo lo bueno de otros frameworks y utiliza las características de las últimas versiones de PHP.

La mayor parte de su estructura está formada por dependencias, especialmente de Symfony, lo que implica que el desarrollo de Laravel dependa también del desarrollo de sus dependencias. (Ciceri, 2018)

1.9.6. Ajax jquery

AJAX significa JavaScript asíncrono y XML (Asynchronous JavaScript and XML). Es un conjunto de técnicas de desarrollo web que permiten que las aplicaciones web funcionen de forma asíncrona, procesando cualquier solicitud al servidor en segundo plano.

JavaScript es un lenguaje de programación muy conocido. Entre otras funciones, gestiona el contenido dinámico de un sitio web y permite la interacción dinámica del usuario. XML es otra variante de un lenguaje de marcado como HTML, como lo sugiere su nombre: eXtensible Markup Language. Mientras HTML está diseñado para mostrar datos, XML está diseñado para contener y transportar datos (Gustavo, 2019)

1.9.7. Modelado

Navicat es un administrador gráfico de base de datos y un software de desarrollo producido por PremiumSoft CyberTech Ltd. para MySQL, MariaDB, Oracle, SQLite, PostgreSQL y Microsoft SQL Server. Cuenta con un Explorador como interfaz gráfica de usuario soportando múltiples conexiones para bases de datos locales y remotas. Su diseño está pensado para satisfacer las diferentes necesidades de un amplio sector del público; desde administradores y programadores de bases de datos a diferentes empresas que dan soporte y o comparten información con clientes o socios. (CyberTech, 2019)

1.10. LIMITES Y ALCANCES

1.10.1. Limites

Dentro de los límites del presente proyecto se toman los siguientes puntos, lo cuales deben ser tomados en cuenta al momento de la implementación del sistema de control y seguimiento, dentro de la institución y sus derivados.

- El sistema no se responsabiliza por una mala aplicación del mismo.
- El sistema solo se basa en los datos del estudiante, que se recolectaron en la planificación previa para las entregas a realizar.
- Los módulos implementados son solo de carácter institucional así como los niveles de acceso.
- El sistema no contempla módulos de contabilidad.

1.10.2. Alcances

El sistema podría ser implementado dentro la institución y estará disponible para sus implicados directos, las cuales estarán involucradas en el plan institucional a realizarse. Dentro de las necesidades y requerimientos de la institución se realizara la creación de los siguientes módulos para el diseño implementado

- Módulo de registros: que permita realizar registros de usuarios, estudiantes, tutores, unidades educativas.

- Módulo de entregas: que permitirá realizar la validación de datos antes de registrar la entrega, además de poder guardar los respaldos de las actas de entregas correspondientes.
- Módulo de consultas: que permita ver la situación y estado del estudiante.
- Módulo de búsquedas y filtrados: con datos específicos para los registros de los estudiantes.
- Módulo de reportes: el cual mostrara informes sobre los resultados de las entregas del cual sea de interés para la parte administrativa.
- Módulo de actualizaciones: para que la parte administrativa pueda corregir datos en caso de que se hayan obtenido en las observaciones en las entregas.
- Modulo administrativo: para el respaldo de la base de datos.
- Módulos de seguridad: para la validación de usuarios y seguimiento de actividades en el uso del sistema de los usuarios.

1.11. APORTES

El aporte que dará el presente proyecto, será en beneficio directo de la Dirección de Educación, así también con su implementación del mismo desde el enfoque de la Ingeniería de Sistemas, será un aporte técnico metodológico ya que se detallara y describirá tanto, la arquitectura, funciones, método y manera del uso del sistema, para que la institución logre satisfacer los requerimientos en sus funciones.

CAPITULO II

MARCO TEÓRICO

2. MARCO TEÓRICO

2.1. INTRODUCCIÓN

Realizar un Sistema de información, requiere de conocimiento en metodologías de desarrollo que utilicen modelos y estructuras formales de diseño e implementación de software, que ayuden a la programación de tareas antes de la construcción del sistema, obtener los requerimientos del usuario y organizar cada una de las etapas de entrega de prototipos y pruebas del sistema, esto con el fin de mantener un cuidadoso manejo de la información de los funcionarios y cumplir con las expectativas del usuario final que hará uso de los módulos del sistema más concretamente los módulos de administración, control y seguimiento en labores de gran magnitud.

Un proceso de desarrollo de software es una definición del conjunto complejo de actividades necesarias para convertir los requisitos de usuario en un conjunto de programas que resuelvan estas necesidades y conformen al final un producto que en este caso es El Sistema de información para el control y seguimiento en entregas.

En este capítulo se describirá los conceptos referentes a la metodología que se usara como línea referencial en el proceso de desarrollo de software, por otro lado también se detallará los conceptos básicos del patrón modelo vista controlador, descripción del mapeo objeto/relacional, es decir como a partir de un modelo de datos orientado a objetos se consigue su almacenamiento en una base de datos relacional, finalmente se describirá brevemente los parámetros y conceptos relacionados con la calidad de software, además se definirán criterios de calidad con los que contará el sistema.

2.2. SISTEMA

2.2.1. El termino sistema

Implícitamente el término “sistema” fue conocido por Aristóteles con su famoso enunciado “*El todo es más que la suma de sus partes*”, a lo largo de la historia el movimiento de los sistemas tuvo contribuciones importantes hasta concebir toda una teoría de sistemas (1951).

“Hoy en día el término de sistema es utilizado con mucha frecuencia en diferentes ámbitos tanto técnicos, económicos, políticos o sociales. Sin embargo, debido a esta generalidad y múltiple aplicación este concepto abstracto es usado indiscriminadamente sin considerar la base objetiva para su determinación, ni sus principios y propiedades que lo gobiernan”. (Mariaca, 2016)

2.2.2. Sistema

Un sistema es un módulo ordenado de elementos que se encuentran interrelacionados y que interactúan entre sí, se utiliza tanto para definir a un conjunto de procesos como también para relacionar los objetos reales dotados en la organización, Los sistemas reales comprenden intercambios de energía, información o materia con su entorno. Las células y la biosfera son ejemplos de sistemas naturales.

Los sistemas también pueden servir como guía de trabajo más práctico y simplificado como el que se muestra en la figura 1., también dependerá de la manera en que necesites emplear un sistema y sus diferentes fases, para la elaboración de un trabajo que conlleve diferentes metas pero con un mismo propósito.

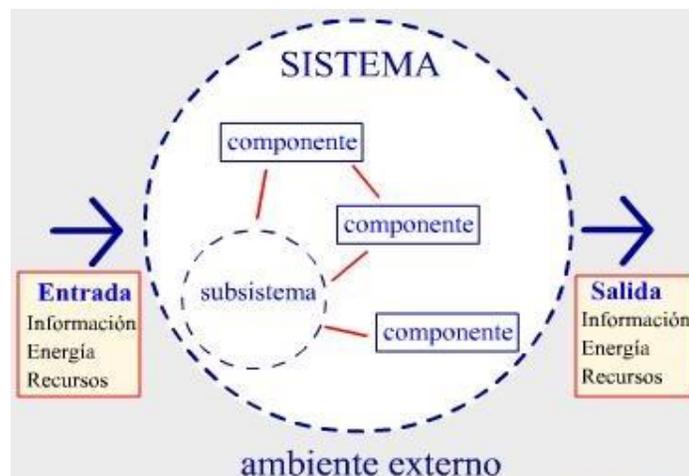


Figura 1 Grafica general de sistema

Fuente: Recuperado de “Teoría General de Sistemas”, (S.N., 2016).

2.3. DATOS

Del latín datum (“lo que se da”), un dato es un documento, una información o un testimonio que permite llegar al conocimiento de algo o deducir las consecuencias legítimas de un hecho.

Es importa tener en cuenta que el dato no tiene sentido en sí mismo, sino que se utiliza en la toma de decisiones o en la realización de cálculos a partir de un procesamiento adecuado y teniendo en cuenta su contexto. Por lo general, el dato es una representación simbólica o un atributo de una entidad (Julian, 2009)

Los datos recolectados, servirá no solo como una base del proyecto sino como la esencia misma del proyecto a realizar, ya que depende de los datos obtenidos, la información que se determina a utilizar en el recorrido del proyecto y de la labor institucional.

2.4. INFORMACIÓN

En la Informática, como información se denomina el conjunto de datos organizados y procesados que funcionan como mensajes, instrucciones y operaciones, o cualquier otro tipo de actividad que tenga lugar en una computadora. (Significados, 2017)

Por tal motivo la información es un conjunto organizado de datos procesados aplicado en él envío y recepción dentro de un sistema como se aprecia en la Figura 2.



Figura 2 Uso de la información

Fuente: Recuperado de "Información", (Wikipedia, 2020)

2.4.1. Tipos de información

- **Información privilegiada:** Se refiere a la información que solo conoce un grupo de personas y que todavía no se ha hecho pública, lo que se puede deber a que su contenido es de acceso restringido y por eso no se debe compartir o divulgar.
- **Información pública:** Es un tipo de información que se comparte al público en general y a la que todos tienen acceso, por tanto se basa en la libertad de expresión. Sin embargo, su publicación dura un tiempo determinado

- **Información privada:** Es el tipo de información que no se divulga de manera pública según lo estipulado en leyes o normas, ya que puede afectar la seguridad personal, empresarial, nacional, entre otros
- **Información externa:** Es el tipo de información que entra a una organización o empresa por diferentes vías externas, y que se emplea en aquellos casos en los que sea necesario resolver alguna situación en particular. Por ejemplo, datos que una institución necesite para realizar una labor encargada.
- **Información interna:** Se trata de aquella información que conoce un selectivo grupo de personas en una empresa u organización, sin que esta tenga necesariamente una mayor relevancia sobre un tema público. (Morales, 2016)

2.5. CONTROL Y SEGUIMIENTO

El seguimiento consiste básicamente en el análisis de la información, mediante proyecciones relacionadas con los tiempos que son generadas en un proyecto, para la identificación temprana de riesgos y desviaciones respecto al plan. Por su parte el control comprende el desarrollo de las actuaciones para conseguir que lo planificado y esperado ocurra. Por lo tanto, controlar no significa sólo identificar, sino que la esencia del control supone indagar en las causas que puedan conllevar a definir las acciones e implementarlas de manera que sus efectos lleven a minimizar riesgos o peligros dentro de un proyecto establecido. (Sanz, 2016)

Los procesos que ayudan en gestionar labores de una magnitud amplia, por lo general siempre coinciden con los pasos que se establecen antes de dar forma al proyecto, pero este se basa en lo que se conoce como control y seguimiento (Junta de andalucia, 2015) *“El procedimiento de Seguimiento y Control, establece el conjunto de acciones que se llevarán a cabo para la comprobación de la correcta ejecución de las actividades del proyecto establecidas en la planificación del mismo”*. Su propósito es proporcionar un entendimiento del progreso del proyecto de forma que se puedan tomar las acciones correctivas apropiadas cuando la ejecución del proyecto se desvíe significativamente de su planificación.

2.6. CONTROL DE ENTREGAS

Está más que probado que la tecnología aplicada a los diferentes segmentos de la actividad logística suma mucho valor y permite conseguir importantes optimizaciones en el circuito integral, según (Chutro, 2017) *“En el control o la gestión de entregas de mercaderías, disponer de herramientas informáticas permite además efectuar muchos ahorros, ganar en productividad y eficiencia, evitar errores y mejorar la calidad del servicio de manera a veces notable”*.

Si nos enfocamos en lo esencial y pensamos en las metas de los sistemas de comprobación de entregas, lo que hacen es ayudar a garantizar que las entregas se realicen de manera correcta, en los tiempos previstos y en el destino definido y además permiten asegurar que las mercaderías estén previamente acordadas y autorizadas.

Gestión de Entregas

Con un software de control de entregas y de confirmación de recibido, el sistema podrá verificar si se han recibido los productos tal como estaba previsto, o si en su defecto en el circuito logístico aparecieron observaciones, pudiendo además conocer los motivos concretos en tiempo real.

Control de Distribución

Los procesos de control de pedidos y entregas se verán notablemente facilitados gracias a la informatización que se obtiene basado en un diseño, con esta solución se puede simplificará mucho en la actividad diaria de control de entrega de herramientas, materiales, mercadería o producto. (Chutro, 2017)

2.7. OPTIMIZACIÓN

La optimización es considerada una fase que parte desde un esquema que pretende llegar a un fin que obtendrá un resultado significativo. El diccionario de la lengua española define la optimización como *“Acción y efecto de optimizar”* está a su vez es la derivación de optimizar que es definida como *“El sentido de ‘obtener el máximo rendimiento o provecho de algo’ y además de, ‘mejorar algo al máximo’”*. El término se utiliza mucho en el ámbito de la informática.

A nivel general, la optimización puede realizarse en diversos ámbitos, pero siempre con el mismo objetivo, *“mejorar el funcionamiento de algo o el desarrollo de un proyecto a través de una gestión perfeccionada de los recursos. La optimización puede realizarse en distintos niveles, aunque lo recomendable es concretarla hacia el final de un proceso.* (Definiciones, 2020)

Dentro de la optimización encontraremos aspectos que se deben considerar importantes para su aplicación correcta con los resultados esperados.

2.7.1.1. Identificar

En primer lugar es importante identificar cual es la optimización a la que queremos llegar y las razones por la cual optimizaremos este procesos, depende de las circunstancias en la cual se encuentran, el sitio de trabajo y los recursos.

2.7.1.2. Repensar

Es el mapeo del proceso, de cómo los pasos se realizaran, cómo fluye el proceso y como parte de la optimización de procesos, para validar que el trabajo realizado no traiga consigo más trabajo del que se necesita.

2.7.1.3. Implementar

Después de conocer el proceso en detalle, se pone en práctica el proceso de una manera nueva, esta es una parte delicada ya que es crucial tanto para los objetivos del proceso como para la optimización de los mismos.

2.7.1.4. Automatizar

Automatizar los procesos que han sido probados y aprobados, para distribuirlos por la empresa y ver los resultados en la reducción de gastos, la prevención de errores, la disminución de desperdicios y una mayor productividad.

2.7.1.5. Monitorear

Durante todo el proceso de optimización, supervisar y controlar, será clave para una mejor automatización, ya que se encontrará nuevos puntos de mejora y obstáculos a ser superados. Para comenzar una vez más con los pasos de optimización, como un proceso de gestión de procesos, es decir cíclico.

2.8. GESTIÓN DE REPORTES

En el ámbito de la informática, los reportes son informes que organizan y exhiben la información contenida en una base de datos. Su función es aplicar un formato determinado a los datos para mostrarlos por medio de un diseño atractivo y que sea fácil de interpretar por los usuarios, de esta forma, confiere una mayor utilidad a los datos. No es lo mismo trabajar con una planilla de cálculos con 10.000 campos que con un dibujo en forma de torta que presenta dichos campos de manera gráfica. (Definiciones.de, 2018)

Los reportes tienen diversos niveles de complejidad, desde una lista o enumeración hasta gráficos mucho más desarrollados, según el programa informático y la base de datos en cuestión, los reportes permiten la creación de etiquetas y la elaboración de facturas, entre otras tareas, de la misma forma, gracias a los reportes cualquier persona puede proceder a realizar un resumen de datos o a clasificar estos en grupos determinados.

Por todo ello, se entiende que estos documentos sean tan importantes en cualquier empresa ya que gracias a ellos cuenta con sus propias bases de datos, realiza recopilaciones de las labores cometidas en un periodo determinado.

2.9. METODOLOGÍA DE DESARROLLO

Una metodología de desarrollo de software se refiere a un sistema de trabajo que es usado para estructurar, planear y controlar el proceso de desarrollo en sistemas de información. A lo largo del tiempo, una gran cantidad de métodos han sido desarrollados diferenciándose por su fortaleza y debilidad. Esta metodología consiste en *"Una filosofía de desarrollo de programas de computación con el enfoque del proceso de desarrollo de software"* (Guarachi, 2014). Estos son a menudo vinculados a algún tipo de organización, que además desarrolla, apoya el uso y promueve la metodología.

Actualmente las metodologías de ingeniería de software pueden considerarse como una base necesaria para la ejecución de cualquier proyecto de desarrollo de software que se considere serio, y que necesite sustentarse en algo más que la experiencia y capacidades de sus programadores y equipo.

Cada metodología de desarrollo de software tiene más o menos su propio enfoque para el desarrollo del mismo y puede variar de acuerdo al tipo de desarrollo que se pretende realizar, ya sea por el tiempo o por la complejidad del proyecto, ver tabla 1.

Tabla 1 Metodologías de desarrollo de software

METODOLOGÍAS TRADICIONALES	METODOLOGÍAS AGILES
Son aquellas metodologías con mayor énfasis en la planificación y control del proyecto, en especificación de requisitos y modelado, reciben el apelativo de metodologías tradicionales o también conocidos como metodologías pesadas.	Están más orientadas a una fuerte planificación durante todo el proceso de desarrollo, llamadas también metodologías clásicas donde se realiza una intensa etapa de análisis y diseño antes de la construcción del sistema

Fuente: Recuperado de “Proyecto socio tecnológico II”, (IUTEB, 2020)

2.10. UWE

Una de las metodologías web, más requeridas para desarrollo de software es sin dudas UWE, debido a la compatibilidad de asimilación en sus diferentes fases de modelación. En su página oficial encontramos que *“UWE es un enfoque de ingeniería de software para el dominio web que tiene como objetivo cubrir todo el ciclo de vida del desarrollo de aplicaciones web”*. El aspecto clave que distingue a UWE es la confianza en los estándares.

Esta metodología está basada en UML de ahí su sigla (UML Web Engineering, en español Ingeniería web basado en UML) nos permite especificar de mejor manera una aplicación web, para el proceso de creación de aplicaciones detalla esta, con una gran cantidad de definiciones, en el proceso de diseño, lista que debe utilizarse, procede de manera iterativa e incremental, coincidiendo con UML, incluyendo flujos de trabajo y puntos de control.

2.10.1. Visión general

El enfoque principal de la perspectiva UWE es proporcionar:

- Lenguaje de modelado específico de dominio basado en UML que también incluye características de seguridad,
- metodología impulsada por modelos,

- soporte de herramientas para el diseño sistemático, y
- soporte de herramientas para la generación (semi) automática de aplicaciones Web.

Además, UWE proporciona soporte de herramientas para el diseño de modelos, verificaciones de coherencia de modelos y generación semiautomática de sistemas Web. ArgoUWE (ArgoUML) y MagicUWE (MagicDraw) son complementos que admiten la notación del perfil UWE y las transformaciones para ayudar al trabajo del diseñador. Para la generación semiautomática de aplicaciones web se implementaron diferentes enfoques. (UWE, 2016)

2.10.2. Modelos de UWE

El enfoque UWE proporciona una notación específica de dominio, un proceso de desarrollo basado en modelos y soporte de herramientas para la ingeniería de aplicaciones web como se aprecia en la Figura 3. La característica de UWE es el hecho de ser un enfoque basado en estándares que no se limita al uso de la "lengua franca" UML, sino que también utiliza XMI como formato de intercambio de modelos, MOF para metamodelado, los principios basados en modelos de enfoque MDA, el lenguaje de transformación de modelos QVT y XML.

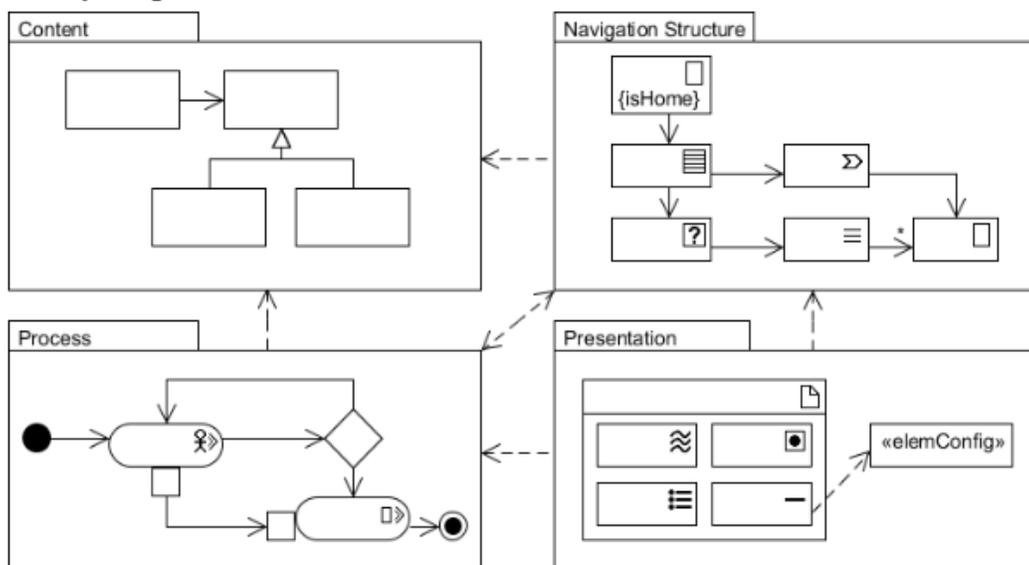


Figura 3 Descripción de modelos UWE

Fuente: Recuperado de "Enfoque basado en estándares, UWE", (UWE, 2016)

2.10.2.1. Modelo de requerimientos

En este modelo se basa, en los requisitos que necesitaremos para los casos de uso, ver figura 4, que serán de aplicación relacional y las actividades descriptivas en detalle sobre los casos de uso. En UWE se distinguen casos de uso estereotipados con «browsing» y con «processing» para ilustrar si los datos persistentes de la aplicación son modificados o no. Los otros casos de uso por el contrario modelan cambios, lo que se especifica con el estereotipo «processing».

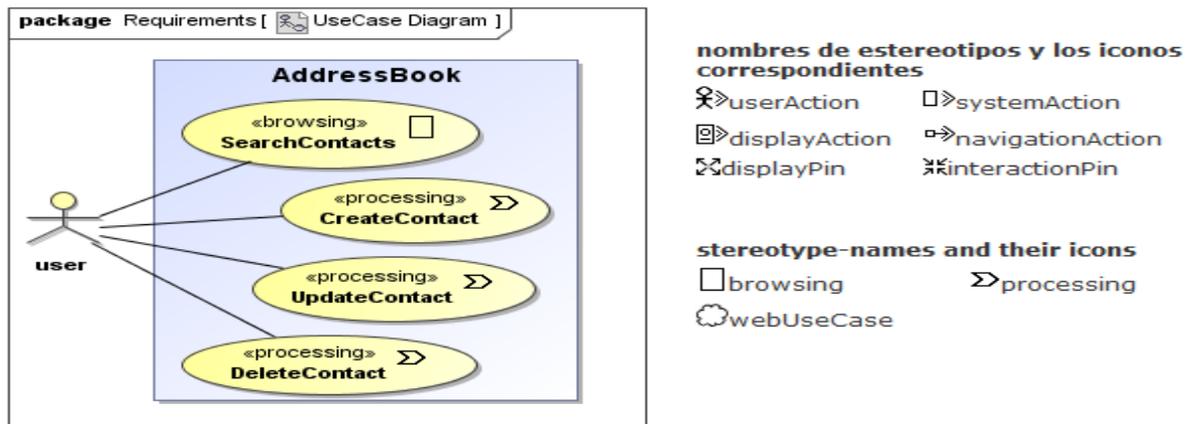


Figura 4 Casos de uso

Fuente: Recuperado de “Caso de uso UWE”, (Ludwin, 2016)

Durante la ingeniería de requisitos es usual determinar qué datos son representados donde y cuando, como estos estereotipos se utilizan para indicar elementos de presentación durante la etapa de ingeniería de requisitos, aspectos que caracterizan a RIAs, pueden ser especificadas mediante valores etiquetados para estos mismos elementos. (UWE, 2016)

2.10.2.2. Modelo de contenido

Este es un diagrama de clases UML normal, por lo tanto, es necesario pensar en las clases que se necesitan para su ejecución. Se requiere de información en cadena, relacionada en aspectos abstractos de tal forma que coadyuven en la integración de una clase total y relacionada, ver figura 5, este modelo pretende ayudar en las especificaciones de los aspectos principales que se tomaran en cuenta para que el contenido principal siempre se base en lo que se quiere del sistema.

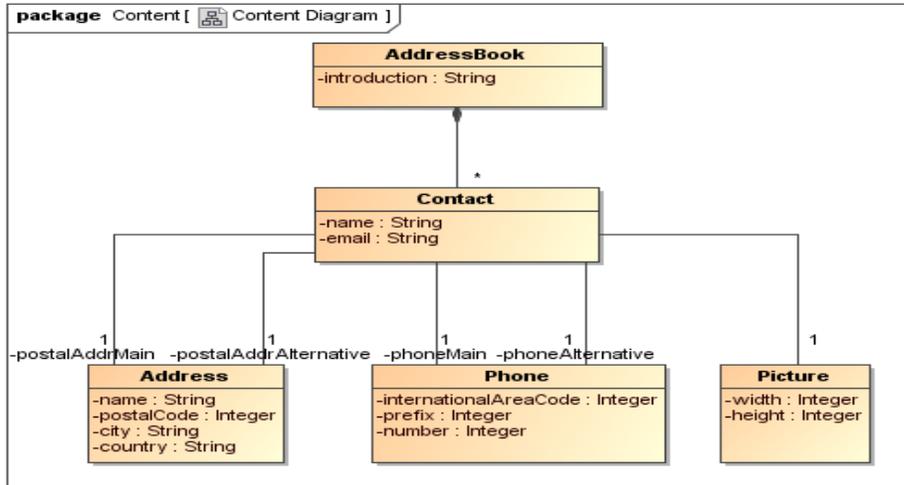


Figura 5 Modelo de contenido

Fuente: Recuperado de “Modelo de contenido, UWE”, (UWE, 2016)

2.10.2.3. Modelo de navegación

En un sistema para la web es útil saber cómo están enlazadas las páginas, esto significa que necesitamos un diagrama conteniendo nodos (nodes) y enlaces (links). Los nodos, son unidades de navegación y están conectados por medio de enlaces, pueden ser presentados en diferentes páginas o en una misma página como se muestra en la Figura 6.

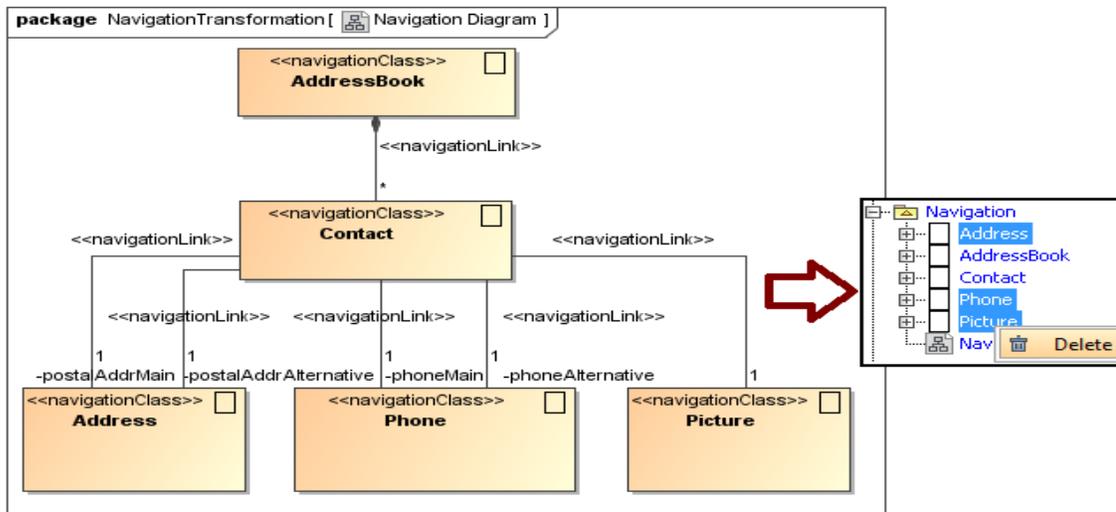


Figura 6 Diagrama de modelo de navegación

Fuente: Recuperado de “Modelo navegacional”, (Ludwin, 2016)

2.10.2.4. Modelo de procesos

Los procesos proveen a una aplicación Web con aspectos dinámicos. Un proceso está compuesto por uno o más subprocessos y/o acciones que el usuario ejecuta para alcanzar una meta, esto representará un cambio en el estado de la aplicación.

Basado en notaciones UML como los diagramas de estados y diagramas de actividades, UWE utiliza diagramas de actividades las cuales proveen un mapa de las características funcionales del sistema por un lado y por otro representan el mapa de interacciones del usuario con el sistema, aplicado a diagramas de actividades y extendiendo su notación para modelar elementos como excepciones, eventos o nodos de actividades estructurados, ver Figura 7. (Ludwin, 2016)

El modelamiento de procesos en UWE está dividido en las siguientes partes:

- Integración de procesos, que permite integrar la ejecución de procesos desde el modelo de navegación.
- Procesamiento, permitirá representar el acceso a datos por procesos.
- Flujos de procesos, representados en flujos de procesos dinámicos, que ejecutan operaciones desde el modelo de contenido. (Chelo S., 2017).

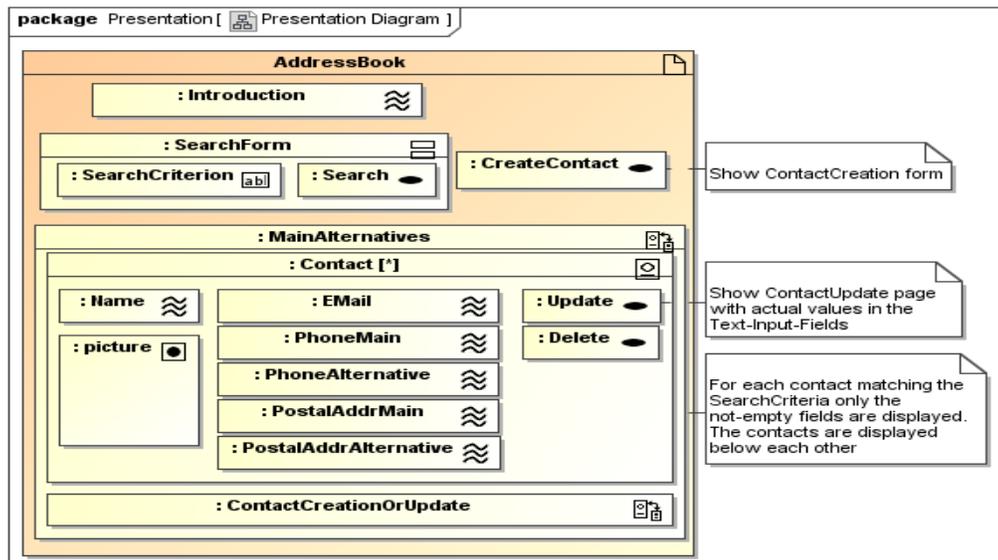


Figura 7 Modelo de presentación

Fuente: Recuperado de "Modelo de presentación, UWE", (UWE, 2016)

2.10.3. Fases o etapas de UWE

UWE es una metodología dirigida y enfocada al modelado de aplicaciones Web, ya que está basada estrictamente en UML, esta metodología nos garantiza que sus modelos sean fáciles de entender para los que manejan UML, también nos proporcionan un esquema de trabajo, ver Figura 8., que garantizara un proceso de desarrollo que nos ayudara de diferentes maneras.

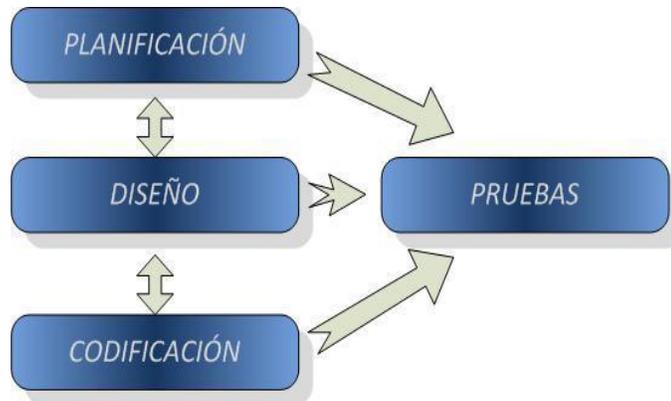


Figura 8 Fases de la metodología UWE

Fuente: Recuperado de "Fases UWE", (s.n. 2019)

2.10.3.1. Análisis de requisitos

Captura, analiza y especifica los requisitos del proyecto, en simple palabras y básicamente, durante esta fase, se adquieren, reúnen y especifican las características funcionales y no funcionales que deberá cumplir la aplicación web, las necesidades de navegación, las necesidades de adaptación y las de interfaz de usuario, así como algunos requisitos adicionales.

2.10.3.2. Diseño del sistema

Se basa en la especificación de requisitos producido por el análisis de los requerimientos, el diseño define cómo estos requisitos se cumplirán, la estructura que debe darse a la aplicación web.

2.10.3.3. Codificación del software

Durante esta etapa se realizan las tareas que comúnmente se conocen como programación, que consiste, esencialmente, en llevar a código fuente, en el lenguaje de programación elegido, todo lo diseñado en la fase anterior.

2.10.3.4. Pruebas

Las pruebas pretenden identificar las fallas u observaciones que no se contemplan en la fase de desarrollo final del sistema, así también como el de complementar algunas fases y así poder asegurar el correcto funcionamiento de secciones de código.

2.10.3.5. Instalación o fase de implementación

Es el proceso por el cual los programas desarrollados son transferidos apropiadamente al computador destino, inicializados y eventualmente, configurados una vez terminada su desarrollo y revisión mediante las pruebas. Todo ello con el propósito de ser ya utilizados por el usuario final.

2.10.3.6. Mantenimiento

Es el proceso de control, mejora y optimización del software ya desarrollado e instalada, este proceso se complementa con el tiempo que se estableció en la fase de análisis y desarrollo, pero también se lo considera con las instancias complementarias a su uso.

2.11. INGENIERÍA DE SOFTWARE

La ingeniería del software es una disciplina de la ingeniería que comprende todos los aspectos de la producción del software desde las etapas iniciales de la especificación del sistema, hasta el mantenimiento de este después que se utiliza. En general, los ingenieros de software adoptan un enfoque sistemático y organizado en su trabajo, ya que es la forma más efectiva de producir software de alta calidad, sin embargo aunque la ingeniería consiste en seleccionar el método más apropiado para un conjunto de circunstancias, un enfoque más informal y creativo de desarrollo podría ser efectivo en algunas circunstancias. El desarrollo informal es apropiado para el desarrollo de sistemas basados en Web, los cuales requieren una mezcla de técnicas de software y de diseño gráfico. (Sommerville, 2005)

La ingeniería de software, también, incorpora el análisis precedente de la situación, el bosquejo del proyecto, el desarrollo del software, el ensayo necesario para comprobar su funcionamiento correcto y poner en funcionamiento el sistema.

2.11.1. Modelos de proceso de software

Los Modelos de Proceso de Software se los puede definir como una descripción simplificada de un proceso del software que presenta una visión de ese proceso. Los modelos de proceso de software, pueden incluir actividades que son parte de los procesos, productos de software y el papel de las personas involucradas en la ingeniería del software.

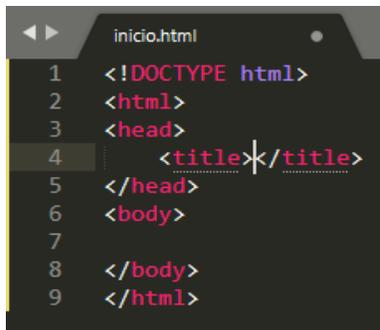
La mayor parte de los modelos de procesos de software se basa en uno de los tres modelos generales o paradigmas de desarrollo de software:

- Enfoque en cascada: Considera las actividades y las representa con fases de procesos separados, tales como la especificación de requerimientos, el diseño de software, la implementación, las pruebas etc.
- Desarrollo iterativo: Este enfoque entrelaza las actividades de especificación, desarrollo y validación, un sistema inicial se desarrolla rápidamente a partir de especificaciones muy abstractas.
- Modelo Prototipado: Un modelo sirve de prototipo para la construcción del sistema final.
- Transformación Formal: Un modelo matemático del sistema se transforma formalmente en la implementación.
- Desarrollo basado en Reutilización: El sistema es ensamblado a partir de componentes existentes. (Sommerville, 2005)

2.12. HERRAMIENTAS

2.12.1. HTML

HTML es un lenguaje de marcado que se utiliza para el desarrollo de páginas de Internet. Se trata de la siglas que corresponden a HyperText Markup Language, es decir, Lenguaje de Marcas de Hipertexto, contiene instrucciones especiales que hacen que el texto este estructurado y pueda enlazar contenido de internet, ver Figura 9, estos se suelen llamar hipervínculo y son muy importantes en la composición de la red. (Prescot, 2015)



```
1 <!DOCTYPE html>
2 <html>
3 <head>
4 <title>*/title>
5 </head>
6 <body>
7
8 </body>
9 </html>
```

Figura 9 Estructura básica de HTML

El consorcio World Wide Web publicó un documento sobre HTML.5 (que es la quinta versión de este lenguaje) en el 2011, en él se detallan algunas de las características de esta versión, “*HTML.5 permite crear aplicaciones web que son capaces de interactuar con los datos locales y servidores de manera sencilla, además de brindar soporte multimedia*”. Al principio surgió como una respuesta de incompatibilidad de versiones anteriores, una diferencia era la no dependencia de APIs ni complemento de terceros, que daban problemas de compatibilidad con todos los navegadores al ofrecer una interfaz.

2.12.2. PHP

PHP (acrónimo recursivo de PHP: Hypertext Preprocessor) es un lenguaje de código abierto muy popular especialmente adecuado para el desarrollo web y que puede ser incrustado en HTML, ver Figura 10. En lugar de usar muchos comandos para mostrar HTML, las páginas de PHP contienen HTML con código incrustado que hace "algo", el código de PHP está encerrado entre las etiquetas especiales de comienzo y final `<?php` y `?>` que permiten entrar y salir del "modo PHP".

```
<!DOCTYPE html>
<html>
  <head>
    <title>Ejemplo</title>
  </head>
  <body>
    <?php
      echo ";Hola, soy un script de PHP!";
    ?>
  </body>
</html>
```

Figura 10 Descripción de uso de PHP

Fuente: Recuperado de “Manual de PHP”, (PHP, 2020)

Lo que distingue a PHP de algo del lado del cliente como Javascript es que el código es ejecutado en el servidor, generando HTML y enviándolo al cliente que recibirá el resultado de ejecutar el script, aunque no se sabrá el código subyacente que era. *“El servidor web puede ser configurado incluso para que procese todos los ficheros HTML con PHP, por lo que no hay manera de que los usuarios puedan saber qué se tiene debajo de la manga”*, (Dimes, 2016). Lo mejor de utilizar PHP es su extrema simplicidad para el principiante, pero a su vez ofrece muchas características avanzadas para los programadores profesionales.

2.12.3. Base de datos MariaDB

MariaDB es un motor de base de datos relacional de código abierto que nace a partir de un fork de MySQL. Esta creado por alguno de creadores originales de MySQL, para asegurar que la comunidad pudiera seguir disfrutando de un “MySQL” de código abierto, mantiene una gran compatibilidad con MySQL eso hace que se pueda utilizar prácticamente en cualquier proyecto en sustitución de MySQL. (Urtiaga, 2018)

Introduce dos motores de almacenamiento nuevos, uno llamado Aria, que reemplaza a MyISAM y otro llamado XtraDB en sustitución de InnoDB, que tiene una alta compatibilidad con MySQL ya que posee las mismas órdenes, interfaces, API y bibliotecas, siendo su objetivo poder cambiar un servidor por otro directamente.

Al ser una base de datos liviana nos garantiza que los procesos que se desarrollen dentro del mismo sean de carácter ligero y fácil de asimilar dentro del trabajo.

2.12.3.1. Características

- MariaDB maneja hasta 32 segmentos clave por clave
- Se agregó `--abort-source-on-error` al cliente mysql
- Precisión de microsegundos en la lista de procesos
- Pool de hilos de ejecución o procesos
- Eliminación de tablas
- Extensiones de prueba `mysqltest`
- Columnas virtuales
- Estadísticas extendidas para el usuario
- Caché de claves segmentadas

- Autenticación a través de plugins
- Especificación de motor de almacenamiento en CREATE TABLE
- Mejoras a la tabla INFORMATION SCHEMA.PLUGINS
- Se agregó –rewrite-db como opción en mysqlbinlog al cambiar de base de datos
- Reporte de Procesos para ALTER TABLE y LOAD DATA INFILE.

2.12.3.2. Motores de almacenamiento

- Aria: Reemplaza MyISAM en cache, mejora consultas Group y Distinc
- XtraDB: es un plugin de InnoDB por compatibilidad que mantiene InnoDB
- PBXT
- FederatedX
- OQGRAPH
- Cassandra (MariaDB 10.0)
- CONNECT (MariaDB 10.0)
- SEQUENCE (MariaDB 10.0)
- Spider (MariaDB 10.0)

2.12.4. Servidor HTTP Apache

Apache HTTP Server es un software de servidor web gratuito y de código abierto para plataformas Unix, Microsoft, Macintosh y otros, con el cual se ejecutan el 46% de los sitios web de todo el mundo. Es mantenido y desarrollado por la Apache Software Foundation, es uno de los servidores web más antiguos y confiables, con la primera versión lanzada hace más de 20 años, en 1995, cuando alguien quiere visitar un sitio web, ingresa un nombre de dominio en la barra de direcciones de su navegador. Luego, el servidor web envía los archivos solicitados actuando como un repartidor virtual. (B., 2018).

Apache tiene amplia aceptación en la red, desde 1996, Apache es el servidor HTTP más usado ya que su papel fue fundamental en el desarrollo de la World Wide Web y alcanzó su máxima cuota de mercado en 2005, siendo el servidor empleado en el 70% de los sitios web en el mundo. Sin embargo, ha sufrido un descenso en su cuota de mercado en los últimos años, en 2009 se convirtió en el primer servidor web que alojó más de 100 millones de sitios web.

2.12.5. Arquitectura MVC

MVC es una propuesta de arquitectura del software utilizada para separar el código por sus distintas responsabilidades, ver Figura 11, manteniendo distintas capas que se encargan de hacer una tarea muy concreta, lo que ofrece beneficios diversos, se usa inicialmente en sistemas donde se requiere el uso de interfaces de usuario, aunque en la práctica el mismo patrón de arquitectura se puede utilizar para distintos tipos de aplicaciones. Surge de la necesidad de crear software más robusto con un ciclo de vida más adecuado, donde se potencie la facilidad de mantenimiento, reutilización del código y la separación de conceptos. (Alvarez, 2019)

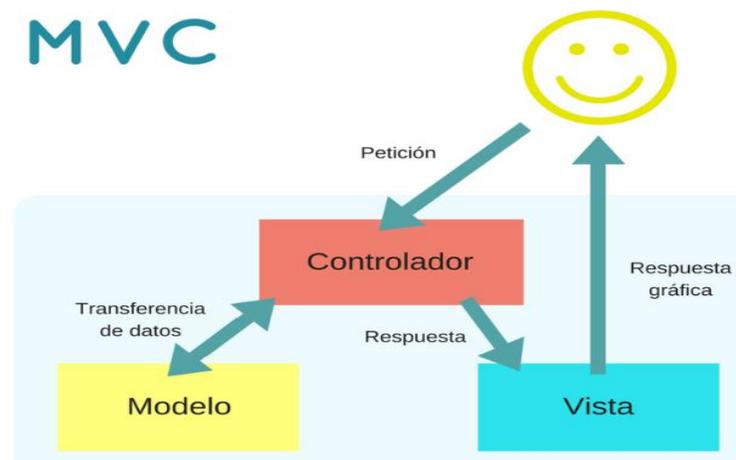


Figura 11 Composición MVC

Fuente: Recuperado de “Que es MVC”, (Alvarez, 2019)

2.12.5.1. Modelo

Es la capa donde se trabaja con los datos y su estructura, por tanto contendrá mecanismos para acceder a la información y también para actualizar su estado. Los datos los tendremos habitualmente en una base de datos, por lo que en los modelos tendremos todas las funciones que accederán a las tablas, no obstante, cabe mencionar que cuando se trabaja con MVC lo habitual también es utilizar otras librerías como PDO o algún ORM como Doctrine, que nos permita trabajar con abstracción de bases de datos y persistencia en objetos. Por ello, en vez de usar directamente sentencias SQL, que suelen depender del motor de base de datos con el que se esté trabajando, se utiliza un dialecto de acceso a datos basado en clases y objetos.

2.12.5.2. Vista

Las vistas, como su nombre nos hace entender, contienen el código de nuestra aplicación que va a producir la visualización de las interfaces de usuario, o sea, el código que nos permitirá renderizar los estados de nuestra aplicación en HTML. En las vistas nada más tenemos los códigos HTML y PHP que nos permite mostrar la salida, generalmente se trabaja con los datos, sin embargo, no se realiza un acceso directo a éstos. Las vistas requerirán los datos a los modelos y ellas se generarán la salida, tal como nuestra aplicación requiera.

2.12.5.3. Controlador

Contiene el código necesario para responder a las acciones que se solicitan en la aplicación, como visualizar un elemento, realizar una compra, una búsqueda de información, etc., es una capa que sirve de enlace entre las vistas y los modelos, respondiendo a los mecanismos que puedan requerirse para implementar las necesidades de nuestra aplicación. Sin embargo, su responsabilidad no es manipular directamente datos, ni mostrar ningún tipo de salida, sino servir de enlace entre los modelos y las vistas para implementar las diversas necesidades del desarrollo.

2.12.6. Framework Laravel

Laravel es un framework de código abierto que según (Ciceri, 2018) *“Sirve para desarrollar aplicaciones y servicios web con PHP 5 y PHP 7, ya que su filosofía es desarrollar código PHP de forma elegante y simple, evitando el código espagueti”*. Fue creado en 2011 y tiene una gran influencia de frameworks como Ruby on Rails, Sinatra y ASP.NET MVC.

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.

2.12.6.1. Herramientas

- Sistema de ruteo, también RESTful
- Blade, Motor de plantillas
- Peticiones Fluent
- Eloquent ORM
- Basado en Composer
- Soporta componentes de estilo
- Soporte para el caché
- Soporte para MVC
- Usa componentes de Symfony

2.12.7. Ajax JQuery

En esencia, AJAX permite que una página web que ya ha sido cargada solicite nueva información al servidor. Dicho así, no supondría en realidad ningún invento novedoso. Una página web que contiene un enlace permite que se solicite al servidor nueva información cada vez que se pincha dicho enlace. Una página web que contiene un formulario envía información al servidor y recibe de él nueva información, normalmente la respuesta ante los datos que se han enviado, ver figura 12. En ambos casos hay una conexión entre el cliente y el servidor

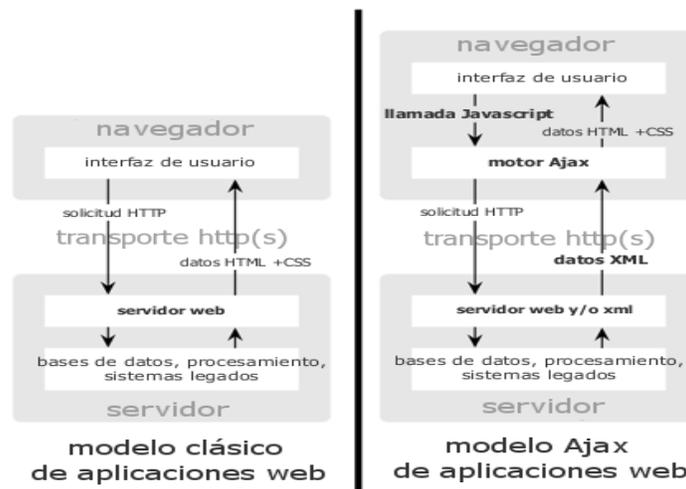


Figura 12 Funcionamiento con Ajax

Fuente: Recuperado de "Que es Ajax", (Digital learning, 2012)

Normalmente, AJAX se define como una técnica para el desarrollo de páginas (sitios) web que implementan aplicaciones interactivas. No obstante, en el análisis encontramos las palabras que definen y forman el sentido de Ajax

JavaScript

Es un lenguaje de programación conocido por ser interpretado por los navegadores de páginas web que ayuda a que se realiza en procesos discretos por parte del cliente en el diseño del front-End.

XML

Es un lenguaje de descripción de datos pensado fundamentalmente para el intercambio de datos entre aplicaciones, más que entre personas, esto ayudara a que la comunicación sea más directa y más fluida.

Asíncrono

En el contexto de las comunicaciones “*y la visualización de una página web no deja de ser un acto de comunicación entre un servidor y un cliente*”, significa que el emisor emite un mensaje y continúa con su trabajo, dado que no sabe *ni necesita saberlo* cuándo le llegará el mensaje al receptor. (Digital learning, 2012)

2.12.8. Modelado Navicat

Navicat es un administrador gráfico de base de datos y un software de desarrollo producido por PremiumSoft CyberTech Ltd. para MySQL, MariaDB, Oracle, SQLite, PostgreSQL y Microsoft SQL Server. Cuenta con un Explorador como interfaz gráfica de usuario soportando múltiples conexiones para bases de datos locales y remotas, está pensado para satisfacer las diferentes necesidades de un amplio sector del público; desde administradores y programadores de bases de datos a diferentes empresas que dan soporte y o comparten información con clientes o socios. (PremiunSoft, 2017).

Las funciones de Navicat son lo suficientemente sofisticadas como para proporcionar a desarrolladores profesionales cumplir todas sus necesidades específicas, pero fáciles de aprender para los usuarios que son nuevos en el área servidor de bases de datos, debido a su interfaz gráfica de usuario bien diseñada.

2.13. PRUEBAS DE SOFTWARE

Las pruebas del software son un elemento crítico para la garantía de calidad del software y representan una revisión final de las especificaciones, del diseño y de la codificación, cabe señalar que *“en cada ciclo de vida del desarrollo del software se plantean un conjunto de pruebas que permiten constatar que el software desarrollado satisface las especificaciones de esa fase”*, (Amo, 2005)

Las pruebas que se realizan en el proceso podrían variar de acuerdo las prioridades que determinen las especificaciones del software, cabe recalcar que existen pruebas para diferentes fases en el desarrollo pero también es considerado de buenas prácticas aplicarlas todas o en sus defectos las más convenientes y efectivas

2.13.1. Tipos de pruebas

2.13.1.1. Pruebas Funcionales

Se basa en las funcionalidades de un sistema que se describen en la especificación de los requisitos, También puede estar no documentado pero se requiere un nivel de experiencia elevado para interpretar estas pruebas. Dentro encontramos lo que son:

- Pruebas unitarias.
- Pruebas de aceptación. (Jimenez, 2019).

2.13.1.2. Pruebas no funcionales

Este tipo de pruebas tienen en cuenta el comportamiento externo del software, es decir cómo funciona el sistema y además se suelen utilizar técnicas de diseño de caja negra. Al igual que las características funcionales, estas tienen que estar definidas en especificaciones del producto, dentro de estas pruebas encontramos:

- Pruebas de estrés.
- Pruebas de portabilidad. (Jimenez, 2019).

2.13.1.3. Pruebas estructurales

Estas pruebas permiten medir la totalidad de las pruebas mediante la evaluación de tipo estructura. En estas pruebas se aplican las técnicas de diseño de caja blanca y el ISTQB “prueba estructural para las pruebas de caja blanca”

2.14. MÉTRICAS DE CALIDAD

2.14.1. ISO/IEC 25000

La familia de normas ISO/IEC 25000 ISO/IEC 25000, conocida como SQuaRE (System and Software Quality Requirements and Evaluation), es una familia de normas que tiene por objetivo la creación de un marco de trabajo común para evaluar la calidad del producto software. (ISO25000, 2020)

La ISO/IEC 25000 es la evolución de otras normas, especialmente de las normas ISO/IEC 9126, que describe las particularidades de un modelo de calidad del producto software, y la ISO/IEC 14598, que abordaba el proceso de evaluación de productos software. Esta familia de normas ISO/IEC 25000 se encuentra compuesta por cinco divisiones, ver figura 13



Figura 13 Divisiones para la calidad ISO 25000

Fuente: Recuperado de “Norma ISO/IEC 25000”, (ISO25000, 2020)

Siendo una mejora de las normas que se solían utilizar con mayor frecuencia, describiremos con facilidad las divisiones sobre los aspectos principales y descripciones concretas de la familia ISO 25000 como se aprecia en la Tabla 2, las cuales proponen considerar aspectos de suma importancia debido a la evolución del uso de la información y los principios de buenas prácticas.

Tabla 2 Familia ISO IEC 25000

DIVISION	ESTANDAR	DESCRIPCION
ISO IEC 2500n Gestión de calidad	ISO IEC 25000	Contiene el modelo de la arquitectura de SQuaRE, la terminología de la familia, un resumen de las partes, los usuarios previstos y las partes asociadas, así como los modelos de referencia.
	ISO IEC 25001	Establece los requisitos y orientaciones para gestionar la evaluación y especificación de los requisitos del producto.
ISO IEC 2501n Modelo de calidad	ISO IEC 25010	Describe el modelo de calidad para el producto software y para la calidad en uso. Esta Norma presenta las características y sub características de calidad frente a las cuales evaluar el producto software.
	ISO IEC 25012	Define un modelo general para la calidad de los datos, aplicable a aquellos datos que se encuentran almacenados de manera estructurada y forman parte de un Sistema de Información.
ISO IEC 2502n Medición de la calidad	ISO IEC 25020	Presenta una explicación sobre un modelo de referencia con los elementos de medición de la calidad, también una guía para que los usuarios seleccionen o desarrollen y apliquen medidas propuestas por normas ISO.
	ISO IEC 25021	Define y especifica un conjunto recomendado de métricas base y derivadas que puedan ser usadas a lo largo de todo el ciclo de vida del desarrollo software.
	ISO IEC 25022	Define específicamente las métricas para realizar la medición de la calidad en uso del producto.
	ISO IEC 25023	Define específicamente las métricas para realizar la medición de la calidad de productos y sistemas software.
ISO IEC 2503n Requisitos de calidad	ISO IEC 25030	Provee de un conjunto de recomendaciones para realizar la especificación de los requisitos de calidad del producto software.
ISO/IEC 2504n División de evaluación de calidad	ISO/IEC 25040	Propone un modelo de referencia general para la evaluación, que considera las entradas al proceso de evaluación, las restricciones y los recursos necesarios para obtener las correspondientes salidas.
	ISO/IEC 25041	Describe los requisitos y recomendaciones para la implementación práctica de la evaluación del producto

ISO/IEC 25042	Define lo que la Norma considera un módulo de evaluación y la documentación, estructura y contenido que se debe utilizar a la hora de definir uno de estos módulos.
----------------------	---

ISO/IEC 25045	Define un módulo para la evaluación de la sub característica y recuperabilidad
----------------------	--

Fuente: Elaboración propia, argumentos de Guía ISO 25000 (ISO25000, 2020)

2.15. SEGURIDAD

2.15.1. Norma ISO 27000

Esta norma proporciona una visión general de las normas que componen la serie 27000, ver Figura 14, indicando para cada una de ellas su alcance de actuación y el propósito de su publicación. Recoge todas las definiciones para la serie de normas 27000 y aporta las bases de por qué es importante la implantación de un SGSI, una introducción a los Sistemas de Gestión de Seguridad de la Información, una breve descripción de los pasos para el establecimiento, monitorización, mantenimiento y mejora de un SGSI.

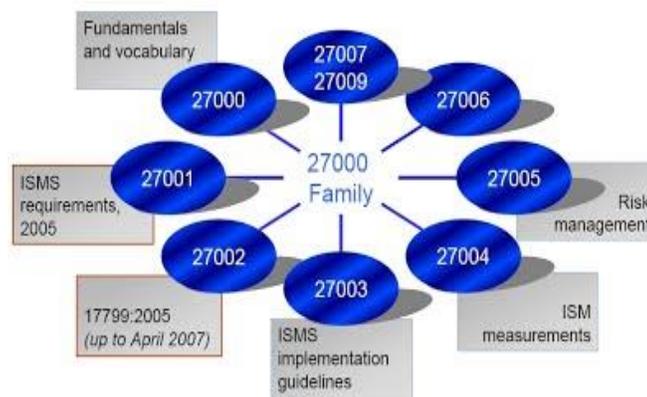


Figura 14 Familia norma ISO 27000

Fuente: Recuperado de "ISO/IEC serie 27000", (Flores, 2018)

2.15.1.1. ISO 27001

Es la norma principal de la serie y contiene los requisitos del sistema de gestión de seguridad de la información. Tiene su origen en la BS 7799-2:2002, y es la norma con arreglo a la cual se certifican por auditores externos los SGSIs de las organizaciones basado en un plan de requisitos y expectativas que conforman la seguridad

gestionada, ver Figura 15. En su Anexo A, enumera en forma de resumen los objetivos de control y controles que desarrolla la ISO 27002:2005, para que sean seleccionados por las organizaciones en el desarrollo de sus SGSI; la organización deberá argumentar sólidamente la no aplicabilidad de los controles no implementados. (ISO27000.ES, 2005)



Figura 15 Gestión de seguridad de la información

Fuente: Recuperado de “Introducción SGSI”, (Craber, 2014)

2.15.1.2. ISO 27002

Es una guía de buenas prácticas que describe los objetivos de control y controles recomendables en cuanto a seguridad de la información. No es certificable. Contiene 39 objetivos de control y 133 controles, agrupados en 11 dominios. Como se ha mencionado en su apartado correspondiente, la norma ISO 27001 contiene un anexo que resume los controles de ISO 27002:2005. (ISO27000.ES, 2005)

2.15.1.3. ISO 27003

Es una guía que se centra en los aspectos críticos necesarios para el diseño e implementación con éxito de un SGSI de acuerdo ISO/IEC 27001. Describe el proceso de especificación y diseño desde la concepción hasta la puesta en marcha de planes de implementación, así como el proceso de obtención de aprobación por la dirección para implementar un SGSI. Tiene su origen en el anexo B de la norma BS 7799-2 y en la serie de documentos publicados por BSI a lo largo de los años con recomendaciones y guías de implantación. (ISO27000.ES, 2005)

2.15.1.4. La triada de la seguridad de la información

Basado en los documentos de la familia ISO 27000, podemos resaltar parámetros de seguridad que son fundamentales en el desarrollo de implementación, o incluso considerar estos parámetros en el desarrollo del proyecto, entre estos parámetros encontramos los tres más principales, ver Figura 16. Que son el pilar fundamental de la seguridad de la información, pero cabe mencionar que existen aspectos que se complementan a estos.



Figura 16 Triada de la información

Fuente: Recuperado de “La triada CIA”, (Cambio Digital, 2020)

- a) **Confidencialidad:** Desde las grandes empresas hasta las personas, tienen información y contenidos que no desean compartir con cualquiera. Es necesario que los sistemas de seguridad rechacen a quienes no deberían ver estos contenidos. Es la clave de la confidencialidad, así que solo el personal autorizado debería poder acceder a los registros de información.
- b) **Integridad:** Es la garantía de que la información a la que se accede no se ha alterado y que lo que allí se lee es exactamente lo que se pretende. Lo que la integridad nos sugiere es que la información a la que accedemos es verdaderamente confiable, que podemos aceptarla como un hecho, pero sabemos que la información puede perder su integridad con la intervención maliciosa. Sin embargo, la integridad también se puede perder de manera involuntaria. Un ejemplo de esto es cuando alguien autorizado por error borra o cambia algún archivo de la información.

c) Disponibilidad: Refiere a la posibilidad de que alguien autorizado pueda acceder sin problema a la información y si es necesario, modificarla. Esto, durante un plazo de tiempo adecuado, que se define de acuerdo al tipo de trabajo. En términos de disponibilidad, el gigante Amazon es un excelente ejemplo, ya que debe estar disponible los 365 días del año sin fallas. Algunos sitios web de empresas, sin embargo, pueden permitirse entrar en mantenimiento por unos días o algo por el estilo.

2.16. MODELO DE COSTOS

Existen tres principales parámetros que se deben usar al calcular los costos de un proyecto de desarrollo de software:

- Costos de esfuerzo (los costos de pagar a los ingenieros y administradores de software).
- Costos de hardware y software, incluido el mantenimiento.
- Costos de viajes y capacitación.

Para la mayoría de los proyectos, el mayor costo es el primer rubro. Debe estimarse el esfuerzo total (en meses-hombre) que es probable se requiera para completar el trabajo de un proyecto. Desde luego, se cuenta con datos limitados para realizar tal valoración, de manera que habrá que hacer la mejor evaluación posible y a continuación agregar contingencia significativa (tiempo y esfuerzo adicionales) en caso de que la estimación inicial sea optimista.

2.16.1. COCOMO II

El modelo COCOMO original se convirtió en uno de los modelos de estimación de costo más ampliamente utilizados y estudiados en la industria. Evolucionó hacia un modelo de estimación más exhaustivo, llamado COCOMO II. Como su predecesor, COCOMO II en realidad es una jerarquía de modelos de estimación que aborda las siguientes áreas:

- Modelo de composición de aplicación. Se usa durante las primeras etapas de la ingeniería de software, cuando son primordiales la elaboración de prototipos de las interfaces de usuario, la consideración de la interacción del software y el

sistema, la valoración del rendimiento y la evaluación de la madurez de la tecnología.

- Modelo de etapa temprana de diseño. Se usa una vez estabilizados los requisitos y establecida la arquitectura básica del software.

Como todos los modelos de estimación para software, los modelos COCOMO II requieren información sobre dimensionamiento. Como parte de la jerarquía del modelo, están disponibles tres diferentes opciones de dimensionamiento: puntos objeto, puntos de función y líneas de código fuente.

Fórmula para hallar el factor de complejidad TCF

$$\mathbf{TCF = (0.65 + 0.01 \times PF)}$$

El procesamiento de datos del punto función se basa en la formula siguiente:

$$\mathbf{PF = Cuenta Total \times TCF}$$

Factor LDC/PF se calcula con la fórmula:

$$\mathbf{LDC = PF \times Factor \frac{LDC}{PF}}$$

Número estimado de líneas de código distribuidas en miles se calcula con la siguiente fórmula:

$$\mathbf{KLCD = \frac{LDC}{1000}}$$

Las ecuaciones del COCOMO básico tienen la siguiente forma:

$$\mathbf{E = a_b(KLCD)^{bb}}$$

$$\mathbf{D = c_b D^{db}}$$

Donde:

E: Esfuerzo aplicado en personas por mes.

D: Tiempo de desarrollo en meses cronológicos.

KLDC: Número estimado de líneas de código distribuidas (en miles).

El número de programadores (Nº Prog) se obtiene con la siguiente fórmula:

$$\mathbf{N^{\circ} Prog = \frac{E}{D}}$$

Costo del software desarrollado por persona = Numero de programadores * salario de un programador.

CAPITULO III
MARCO APLICATIVO

3. MARCO APLICATIVO

3.1. INTRODUCCIÓN

Luego de detallar y definir muchos conceptos y términos en el capítulo anterior se deben poner en marcha para y aplicarlos en el proyecto, como la aplicación de la metodología UWE desde la obtención de requisitos el modelado de los requisitos tales como el modelo de navegación y modelos de presentación del sistema, como también la captura de la implementación del sistema que es parte de la metodología UWE.

3.2. ESQUEMA DEL SISTEMA

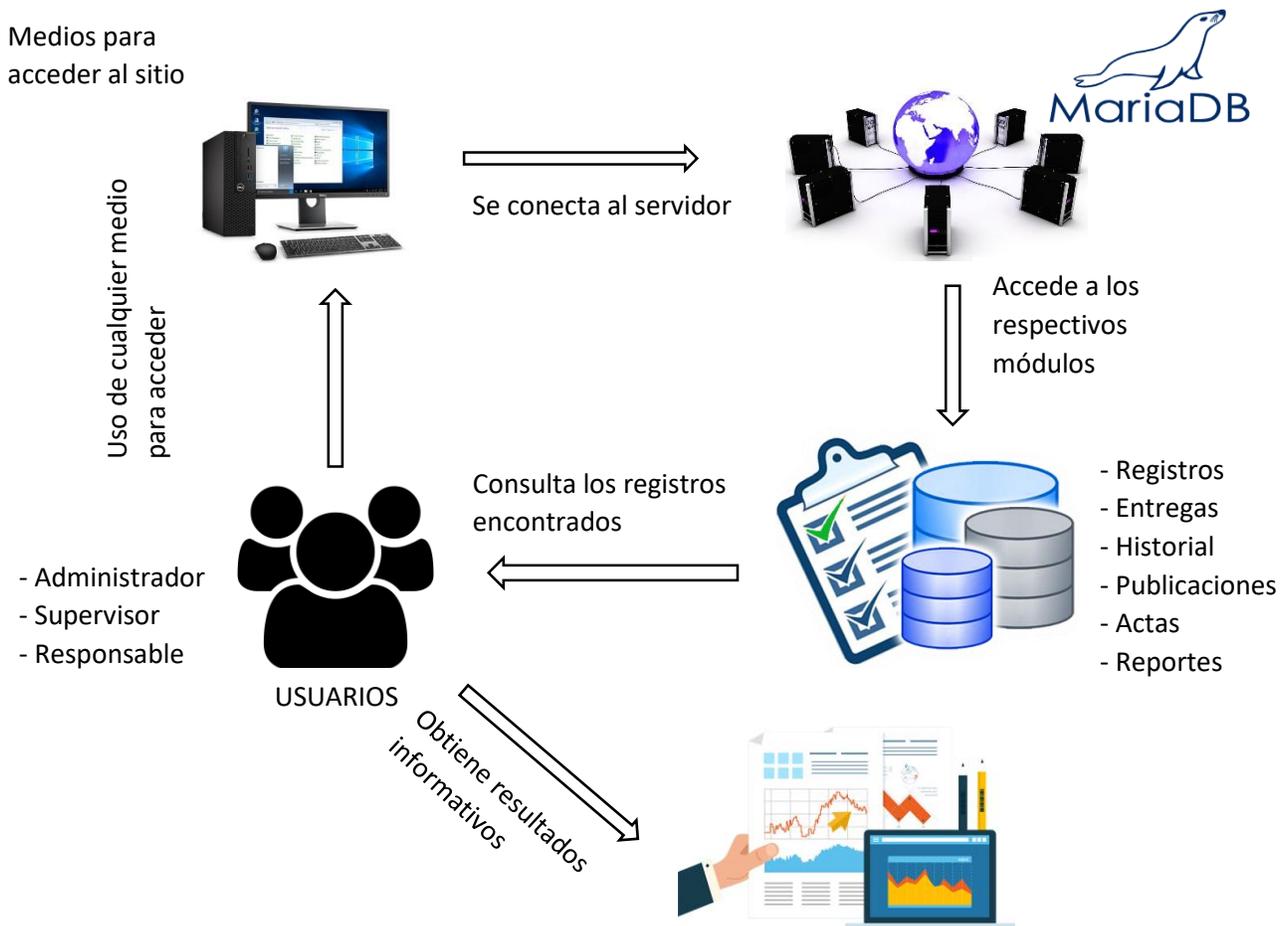


Figura 17 Esquema del sistema

3.3. DESARROLLO DE MODELADO MEDIANTE UWE

3.3.1. Obtención de requisitos

La tarea de obtención de requisitos es muy importante para que un sistema sea exitoso, en este sentido para el presente proyecto se realizaron las siguientes actividades que se detallan en la Tabla 3

Tabla 3 Tarea para obtener requisitos

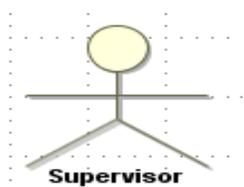
TAREA	CARACTERÍSTICAS
Entrevista	Se entrevistaron a: <ul style="list-style-type: none">- Responsable y miembros del área de TIC'S- Miembros del área de coordinación- Directores de las unidades educativas- Coordinadores del gobierno municipal
Observación	En la Dirección de Educación del GAMEA se pudo observar dificultades a la hora de procesar los registros que se obtuvieron una vez iniciado el proyecto de entrega de canastas, lo cual generaba mucha inconsistencia y descoordinación con las unidades educativas.

3.3.1.1. Definición de actores

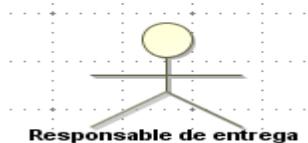
La identificación de actores nos permitió conocer a las personas involucradas en el proceso de entrega, revisión y reporte de la información en cuanto a las actividades programadas dentro de la institución, para formar los casos de uso. En la siguiente tabla se muestran detalladamente la lista de actores y su descripción de actividades relacionados con el sistema.

Tabla 4 Listado de actores

ACTORES	DESCRIPCIÓN
	El administrador es máxima autoridad dentro el sistema y por lo tanto uno de los máximos responsables de la institución, que goza de varios privilegios para administrar y gestionar los registros que se emitan dentro del sistema.



El supervisor brinda apoyo al administrador, cumpliendo roles que ayuden a reducir las actividades administrativas y que de esa manera el sistema tenga variación en las actividades cuya revisión sean directas por un supervisor con tareas que no necesiten directamente la incursión del administrador.



Los responsables son los directos encargados de las entregas en las unidades educativas, su función más detallada es la de brindar y reportar toda la información, relacionado sobre las actividades en las entregas y observaciones que puedan ocurrir en el proceso.



Los estudiantes son los involucrados directos del sistema pero sus funciones se limitan a la revisión y validación de datos para que pueda existir certeza en la información que se requiere.

3.3.1.2. Listado de requerimientos del sistema

Los requerimientos funcionales del sistema se enfocan principalmente en lo que el sistema hará. Una forma de identificar que algún elemento es un requerimiento funcional del sistema, es por medio de la frase que es lo que “el sistema debería hacer”. Los requerimientos funcionales pueden clasificarse por medio de tres categorías:

Tabla 5 Categoría de funciones

CATEGORÍA	SIGNIFICADO
Evidentes	Debe realizarse y el usuario debe estar consciente de que se realice.
Ocultas	Debe realizarse pero no ser visible para los usuarios.
Opcionales	Añadir las no implica que va a incrementar costo o afectar a otros requerimientos funcionales.

Fuente: Liz, L. 2014

Características que necesita el sistema a partir de la información obtenida para el sistema que se muestra a continuación:

Tabla 6 Requisitos funcionales

REQUISITOS F.	FUNCIÓN	CATEGORÍA
R1	Control de acceso seguro y diferenciado de usuarios	Evidente
R2	Gestión de usuarios	Evidente
R3	Gestión de contenidos administrativo	Evidente
R4	Gestión de registros estudiantiles	Evidente
R5	Gestión de actas dinámicas y digitales por cursos de unidades educativas.	Evidente
R6	Gestión de actas de conformidad personal	Evidente
R7	Búsqueda y validación de registros.	Evidente
R8	Generación de reportes	Evidente
R9	Control y estimación de resultados sobre los estudiantes contemplados	Evidente
R10	Seguridad complementaria que ayude a justificar cambios en los registros	Evidente
R11	Gestión de consultas para validar datos por estudiante registrado	Evidente

3.3.1.3. Definición de procesos

Después de obtener los requisitos de los requerimientos del sistema se detalla los procesos del sistema que cada actor (Persona) espera gestionar a través del sistema.

Administrador:

- a) Gestión de Usuarios: El administrador del sistema realiza registros de los usuarios, y sus respectivos roles de trabajo con los siguientes campos:
 - Nombre Completo
 - Nombre de usuario
 - Correo
 - Contraseña
 - Dependencia
 - Rol: para poder asignar a cada Usuario.
- b) Gestión de las unidades educativas: El administrador realiza la gestión de las unidades educativas estimando los estudiantes habilitados para la entrega.
- c) Gestión de Estudiantes: El administrador registra y asigna el estado, para la entrega a cada estudiante.
- d) Gestión de Tutores: El administrador realiza la gestión de los tutores responsables del estudiante para que puedan acompañar en la entrega y tener el respaldo de una persona adulta.
- e) Gestión de Informes: Generación de actas con resultados, por especificación o general de las entregas hechas en las unidades educativas.
- f) Gestión de actas: El administrador determina las actas que se pudieran realizar de forma digital, en caso de ser necesario.

Supervisor:

- a) Gestión de estudiantes: El supervisor podrá realizar revisión y cambios en los datos de los estudiantes.
- b) Gestión de tutores: El supervisor podrá realizar revisión y cambios los datos de los tutores del estudiante.
- c) Gestión de entregas: Sera la persona que supervise la entrega y podrá realizar cambios de datos de los estudiantes en caso de observaciones.
- d) Gestión de actas: será el encargado de generar actas por estudiante y por curso de acuerdo a la prioridad de la unidad educativa.

Responsable de entregas:

- a) Gestión de estudiantes: el responsable podrá verificar y modificar datos del estudiante antes de realizar la entrega.
- b) Gestión de entregas: El responsable podrá verificar los datos y así poder realizar entregas de manera múltiple o personal.
- c) Gestión de actas digitales: El responsable podrá generar actas de manera digital mediante actas escaneadas o imágenes que puedan convalidar las entregas hechas y las observaciones que pudieran haber.

Estudiante:

- a) Consulta y validación: los estudiantes tendrán un acceso de manera pública de tal forma que logren verificar los datos proporcionados por sus unidades educativas.

3.3.2. Análisis de requerimientos

Aquí se plasma mediante gráficos el análisis de requerimientos del sistema con el Diagrama de Caso de Uso que describe el comportamiento del sistema con las acciones que cada actor hace y las funcionalidades del sistema.

3.3.2.1. Diagramas de casos de uso (General)

Aquí se hace el modelado, donde se muestra la interacción de los actores sobre los casos de uso del sistema.

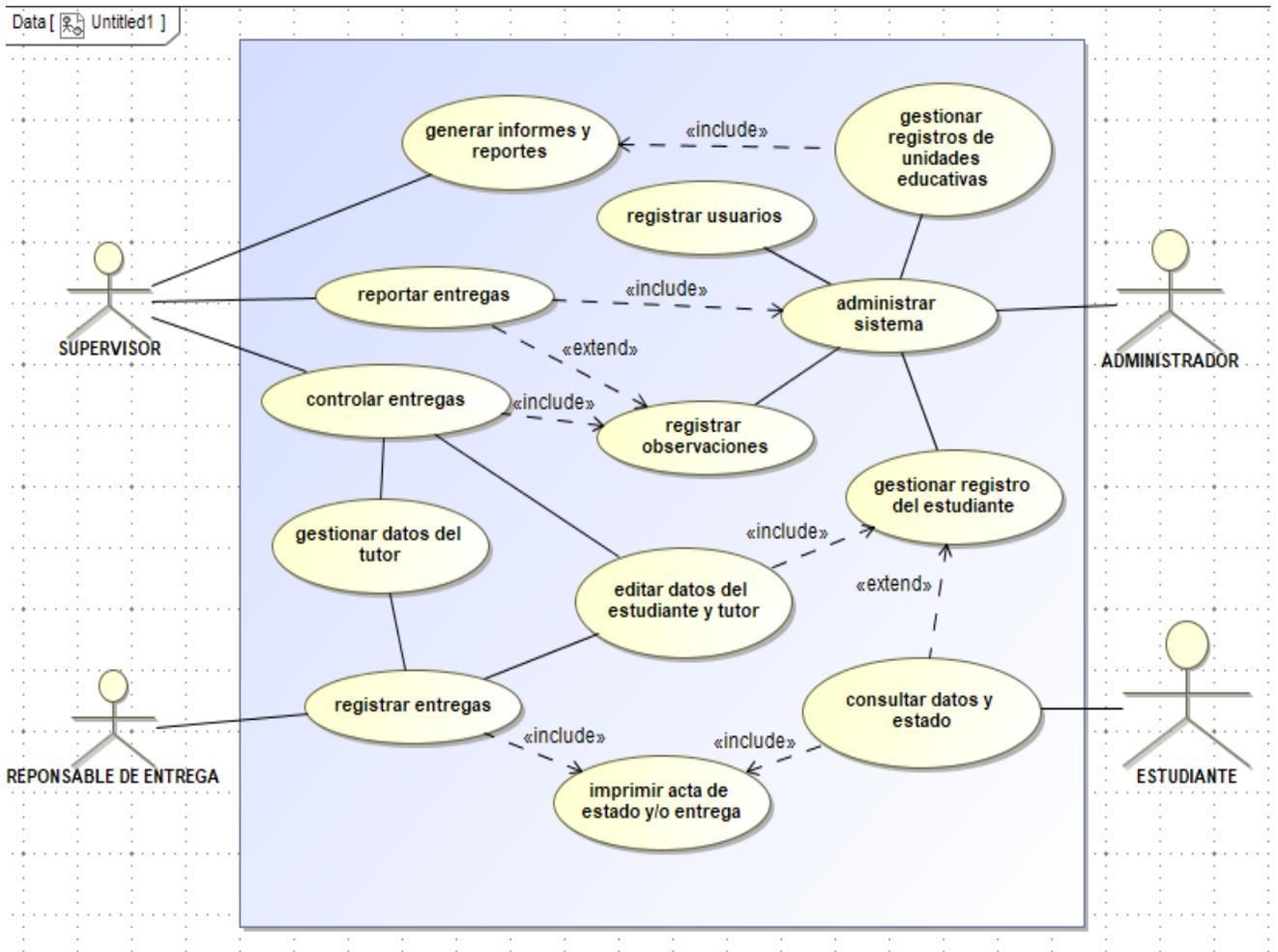


Figura 18 Diagrama de caso de uso (General)

3.3.2.2. Diagramas de casos de uso (Administración del sistema)

Este modelado, ver figura 19, refleja la didáctica general de la administración del sistema y las funciones que se ejecutaran de tal forma que puedan cumplir, de ser posible, todas las funciones que requiere el sistema

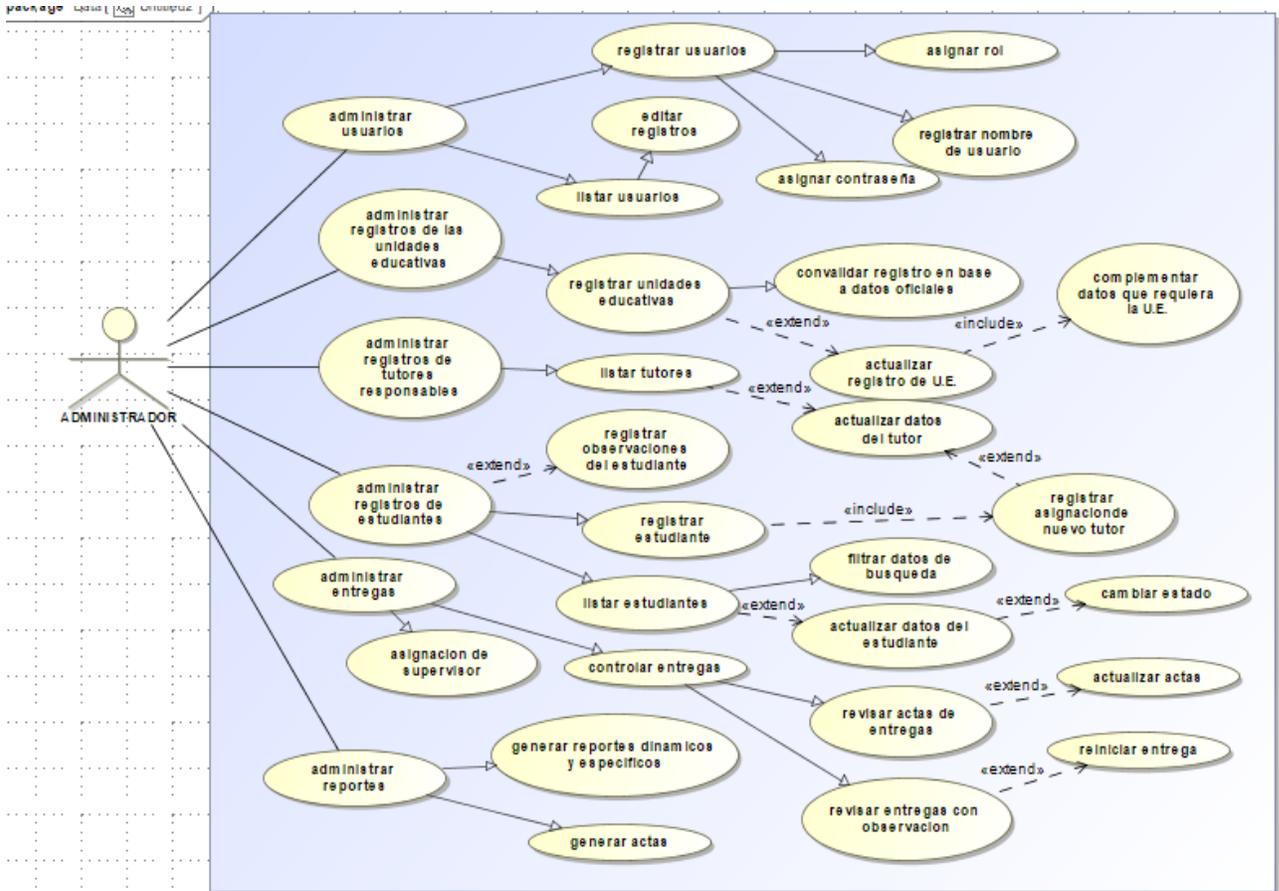


Figura 19 Diagrama de caso de uso: Administración del sistema

Tabla 7 Caso de uso: Administración del sistema

CASO DE USO: ADMINISTRACIÓN DEL SISTEMA	
Objetivos	Describe el proceso de administrador del sistema.
Actores	Administrador

Descripción

El Administrador registra un usuario asignando un nombre de usuario, contraseña, rol y otros atributos del usuario. Por otro lado para gestionar a los usuarios registrados, el administrador puede dar de baja, ver y modificar los datos del Usuario.

El administrador también hace la tarea de registrar a las unidades educativas, con una previa validación de los mismos con datos proporcionados por el ministerio de educación, también la de incluir algunos cursos de unidades educativas que se encuentran en el convenio del gobierno municipal.

También el administrador hace el registro del tutor o tutores (de ser necesario más de uno), para que pueda acompañar en la entrega, bajo el consentimiento de conformidad de una persona mayor de edad.

El administrador de realiza el registro de los estudiantes que no hayan sido tomados en cuenta a su debido momento y su vez realiza la asignación del tutor responsable para que se acredite la entrega.

Otra de las funciones del administrador es la de controlar todas las entregas de todas las formas posibles, como por ejemplo la de asignar al supervisor de unidad educativa que coordinara el trabajo administrativo y de igual forma realizara el control de la entrega mediante revisión de actas para justificar cambios que se realicen en observaciones de las entregas cuya justificación sea necesariamente detallada.

También podrá generar reportes con todo tipo de información que resulten beneficiante al contemplar los resultados finales.

3.3.2.3. Diagramas de casos de uso (Control y reportes de entregas)

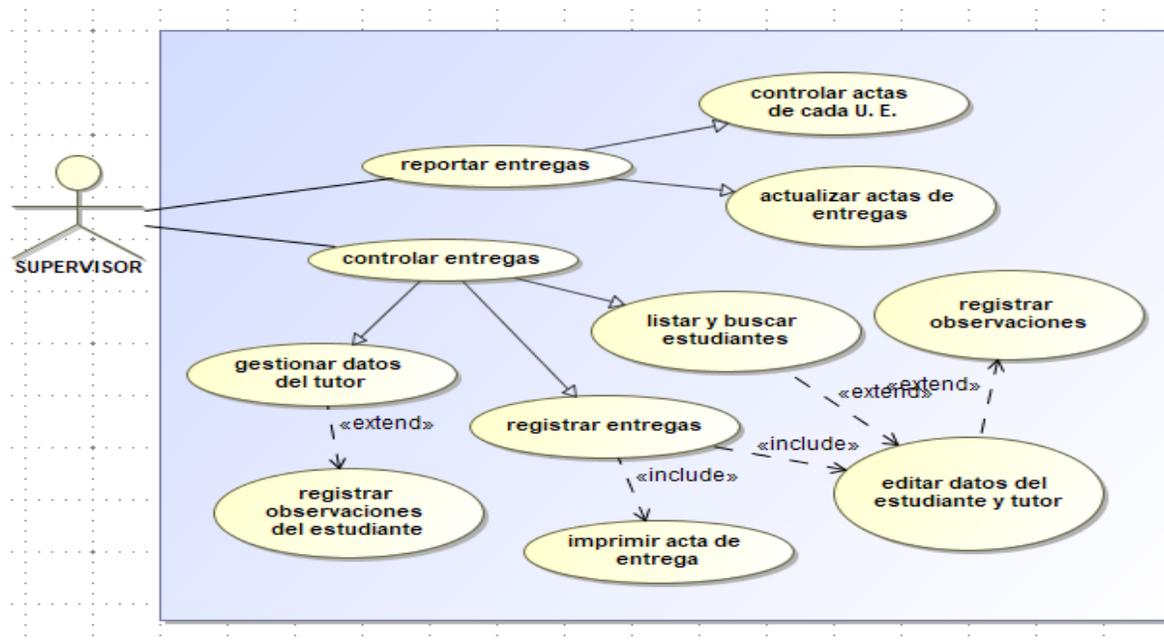


Figura 20 Diagrama de caso de uso: Control y reporte de entrega

Tabla 8 Caso de uso: Control y reporte de entrega

CASO DE USO: CONTROL Y REPORTE DE ENTREGAS	
Objetivos	Describe el proceso grafico del control y reportes en las entregas.
Actores	Supervisor
Descripción	<p>El supervisor una vez asignado por el administrador, para la unidad educativa, tendrá la tarea de controlar los datos observados de los registros del estudiante, eso incluye datos de los tutores asignados También el supervisor debe controlar los reportes que se emitan en las entregas que contengan todo tipo de respaldo que justifique el trabajo o justifiquen inconvenientes que puedan ocurrir.</p> <p>El supervisor será el encargado de actualizar datos del estudiante cuando así se lo requiera, pero con la justificación previa por parte del tutor que acompaña en la entrega.</p> <p>Podrá realizar cambios en los datos del tutor responsable que acompañe en la entrega, así mismo podrá realizar el registro de un nuevo tutor en caso que sea necesario, con la justificación que se acredite por la U.E.</p> <p>El supervisor podrá realizar el registro de la entrega, pero con algunas condicionantes que ayuden a justificar del por qué el supervisor termino realizando la entrega</p>

3.3.2.4. Diagramas de casos de uso (Registro de entregas)

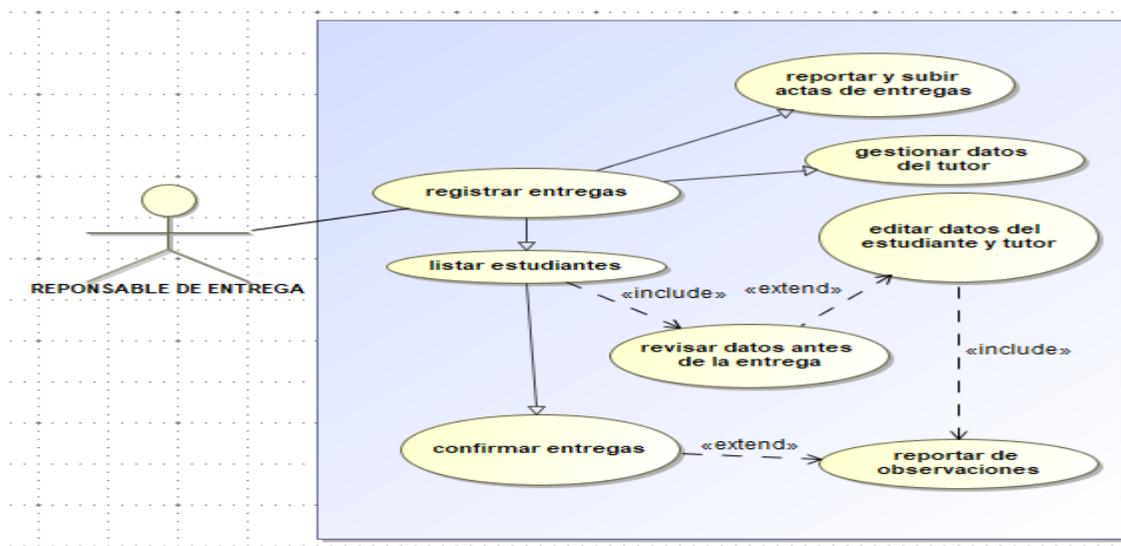


Figura 21 Diagrama de caso de uso: Registro de entregas.

Tabla 9 Caso de uso: Registro de entrega

CASO DE USO: REGISTRO DE ENTREGAS	
Objetivos	Describe el proceso en el de registrar las entregas hechas.
Actores	Responsable de entrega.
Descripción	<p>El responsable será el encargado de registrar las entregas de manera rápida y dinámica, ya sea por curso o por estudiante, de la forma que más crea conveniente.</p> <p>El responsable podrá realizar la validación y actualización de datos de un estudiante, que presente observaciones en su registro, de la misma forma estos cambios deberán ser justificados.</p> <p>El responsable tiene la labor se subir al sistema actas digitales ya sean escaneadas o en fotos, de las entregas realizadas y de las observaciones que pudieron ocurrir.</p>

3.3.2.5. Diagramas de casos de uso (Consulta de validación datos y estado)

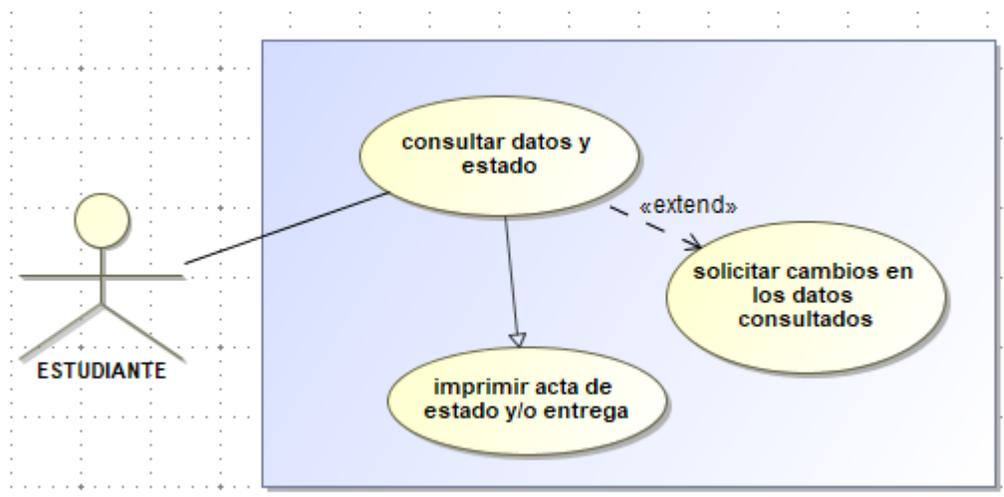


Figura 22 Diagrama de caso de uso: Consulta y validación de datos

Tabla 10 Caso de uso: Consulta y validación de datos

CASO DE USO: CONSULTA DE VALIDACIÓN DE DATOS Y ESTADO	
Objetivos	Describe el proceso público de consultas de datos del estudiante
Actores	Estudiante
Descripción	El estudiante puede realizar consultas sobre el registro de sus datos que ya estén sistematizados en una base de datos. El estudiante puede hacer la descarga del acta de entrega que haya sido realizada.

3.3.3. Diseño conceptual

3.3.3.1. Modelo conceptual

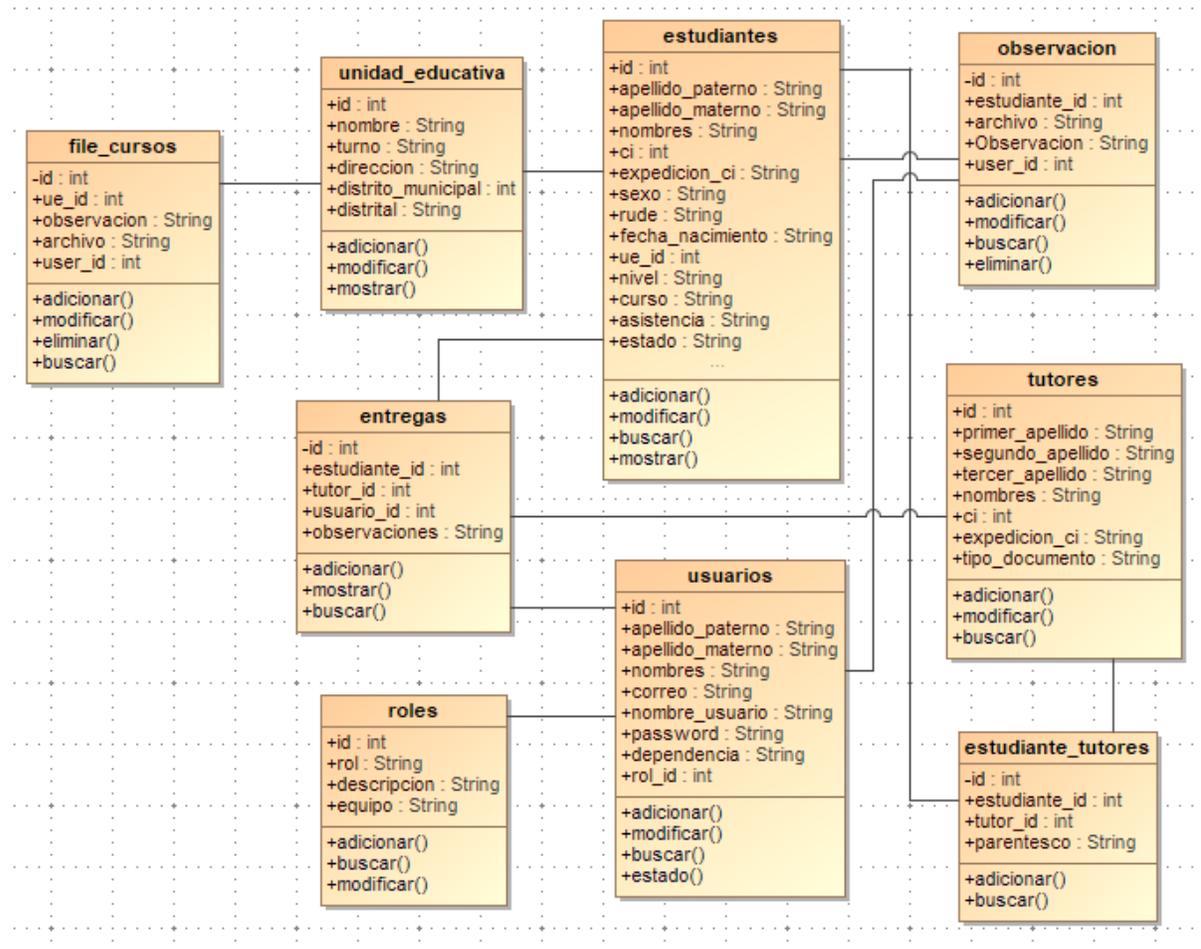


Figura 23 Diagrama de clases

3.3.4. Diseño de navegación

3.3.4.1. Modelo de navegación: Administrador

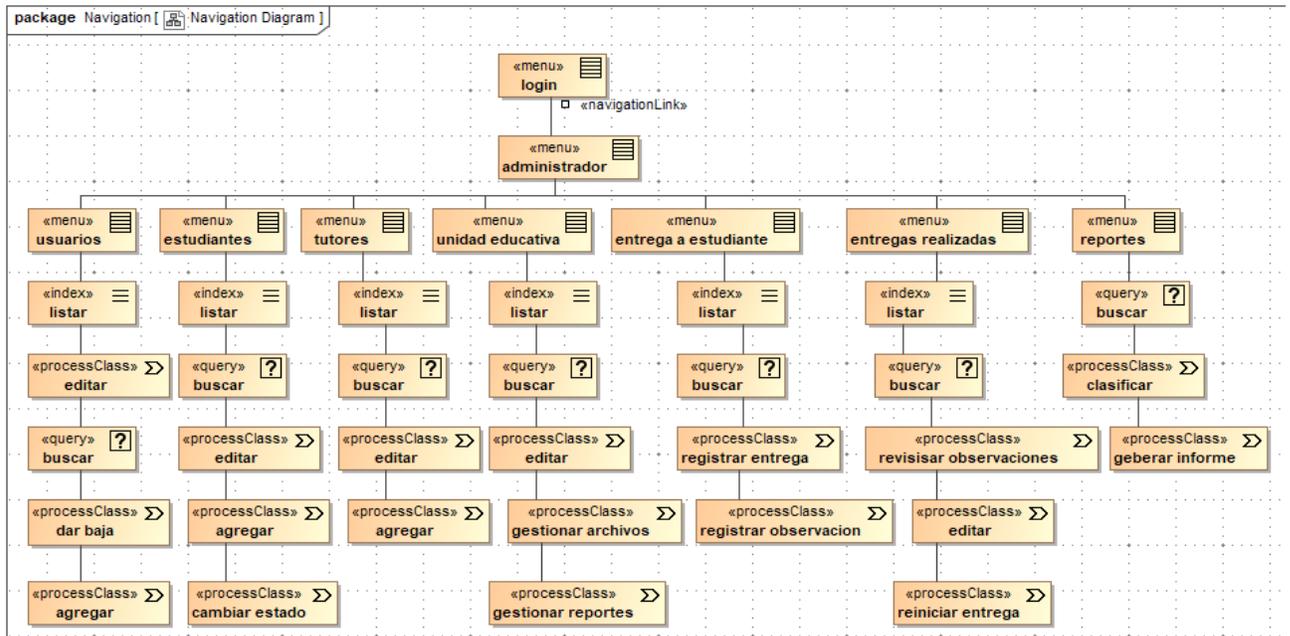


Figura 24 Modelo de navegación: Administrador

3.3.4.2. Modelo de navegación: Supervisor

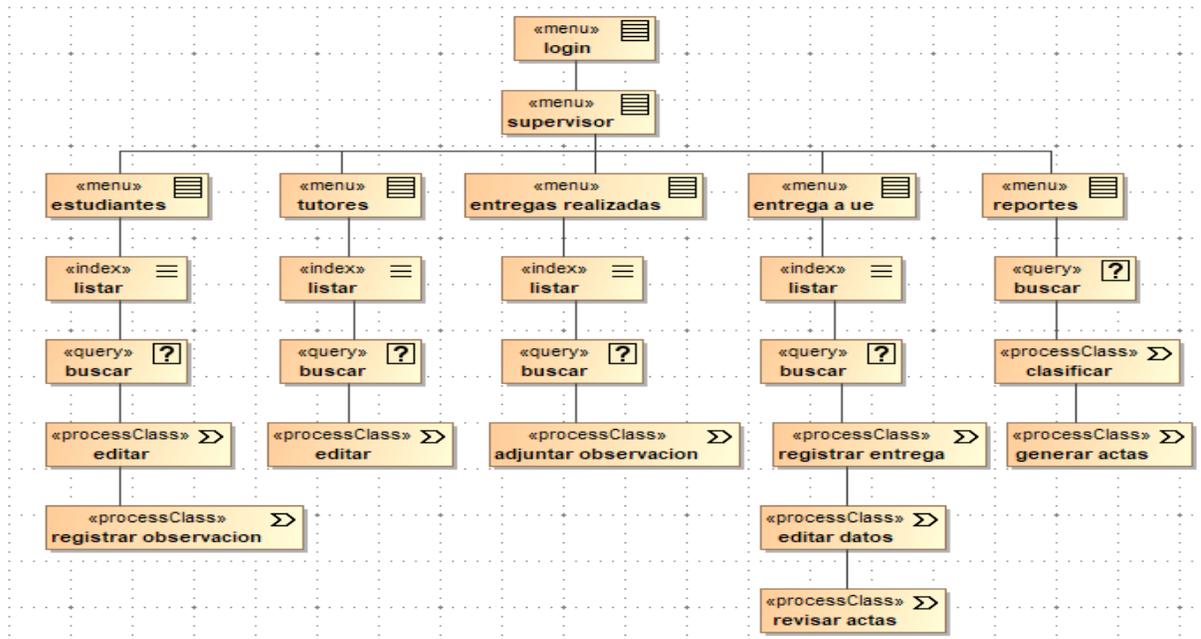


Figura 25 Modelo de navegación: Supervisor

3.3.4.3. Modelo de navegación: Responsable de entrega

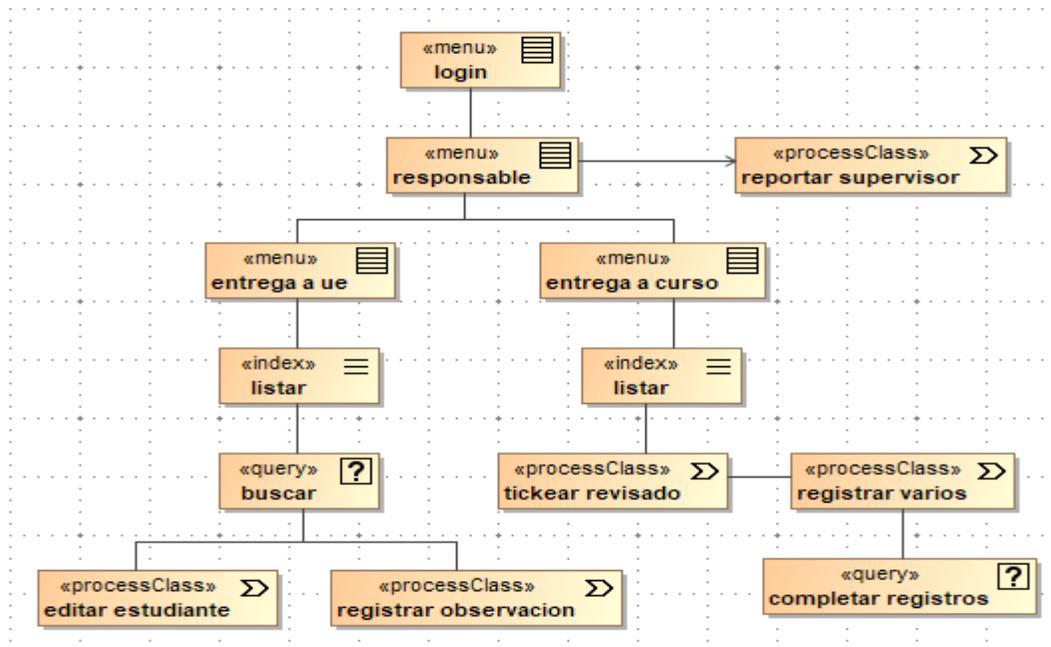


Figura 26 Modelo de navegación: Responsable de entrega

3.3.4.4. Modelo de navegación: Estudiante

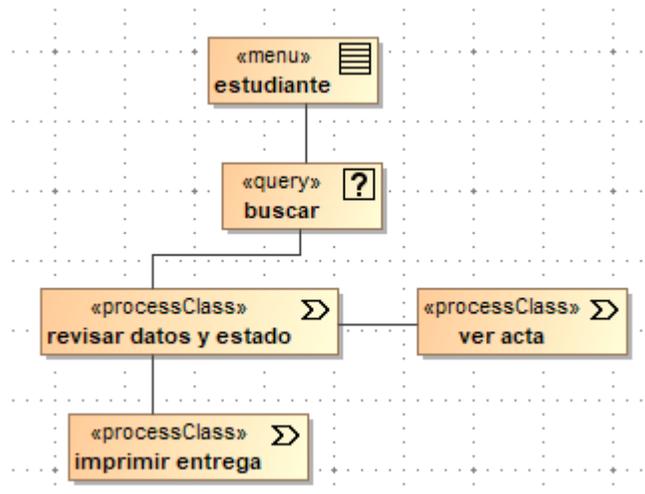


Figura 27 Modelo de navegación: Estudiante

3.3.5. Diseño de presentación

3.3.5.1. Modelo de presentación: Portada y consultas

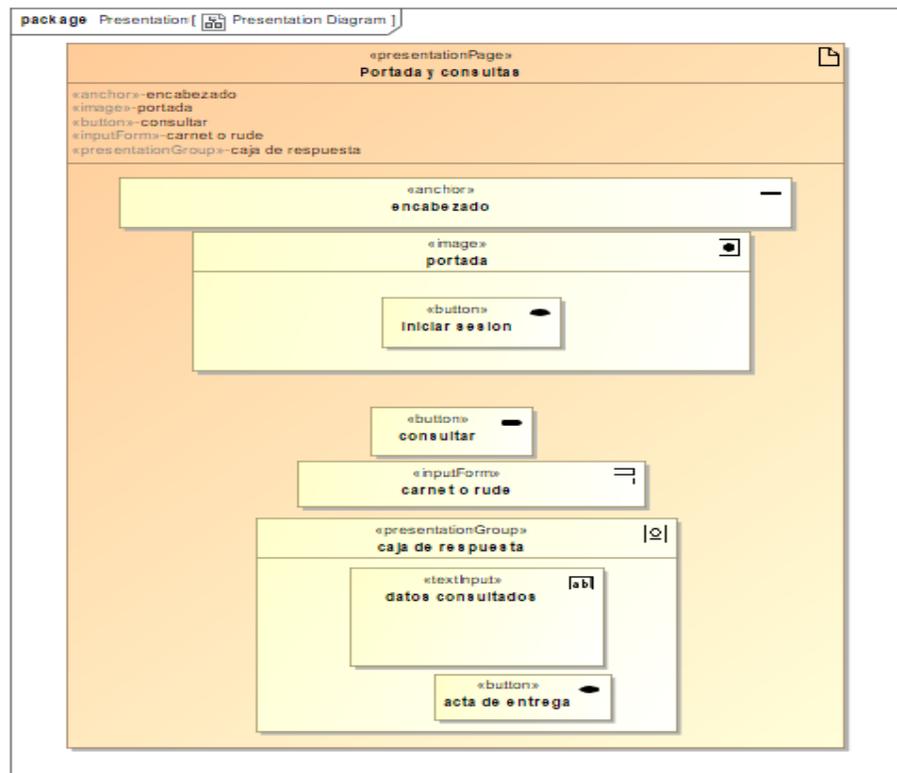


Figura 28 Modelo de presentación: Portada y consultas

3.3.5.2. Modelo de presentación: Inicio de sesión

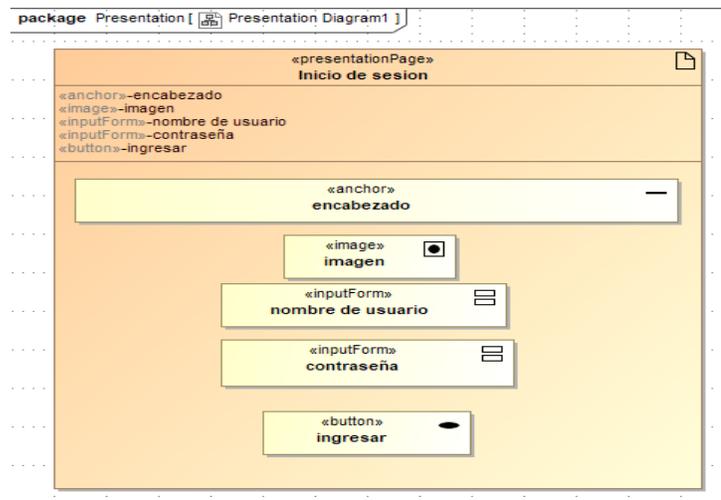


Figura 29 Modelo de presentación: Inicio de sesión

3.3.5.3. Modelo de presentación: Información general

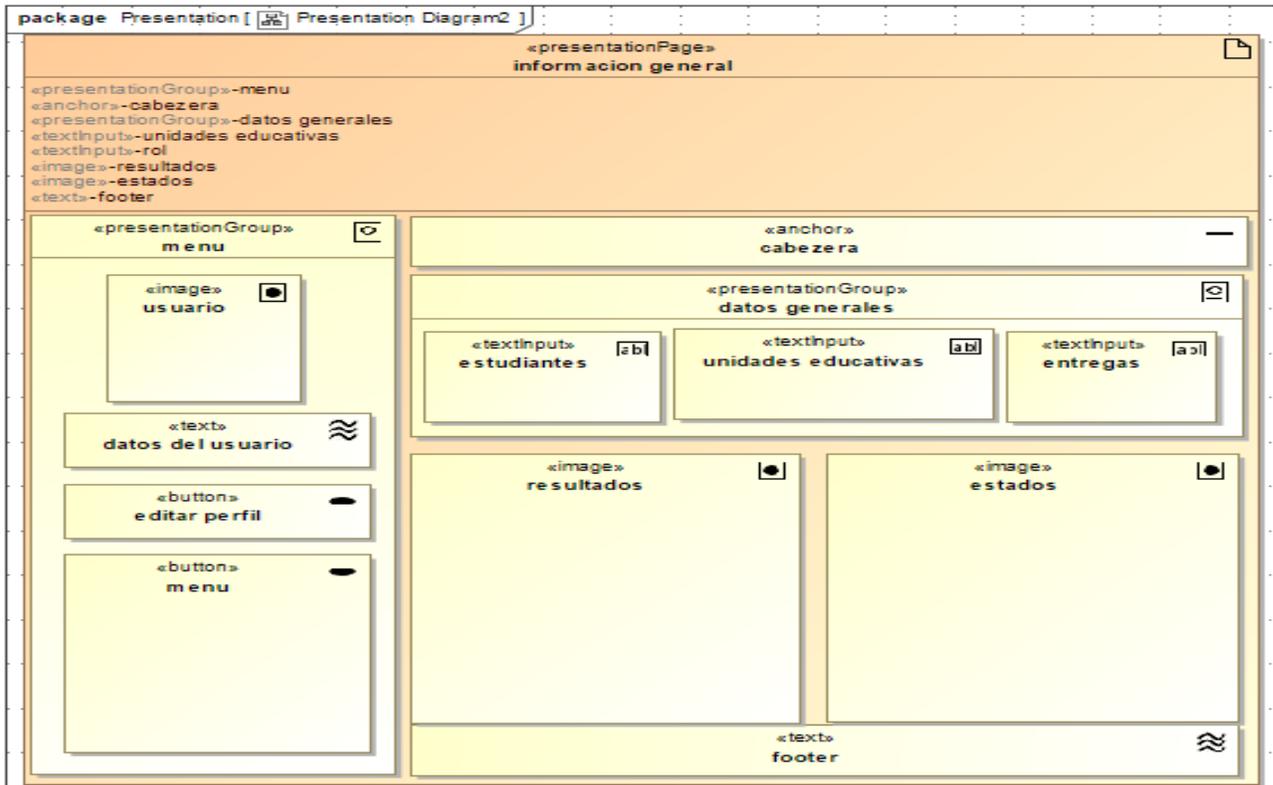


Figura 30 Modelo de presentación: Información general

3.3.5.4. Modelo de presentación: Perfil de usuario

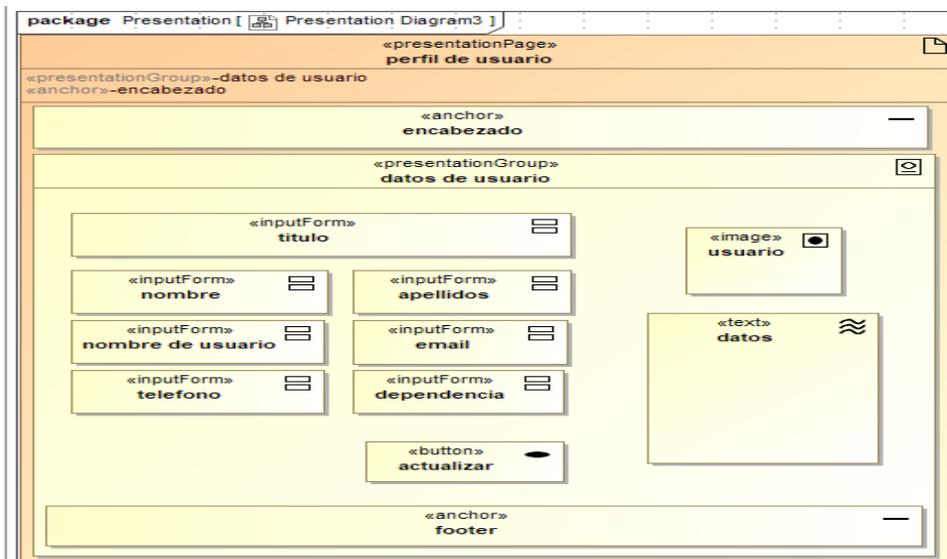


Figura 31 Modelo de presentación: Perfil de usuario

3.3.5.5. Modelo de presentación: Administración de unidad

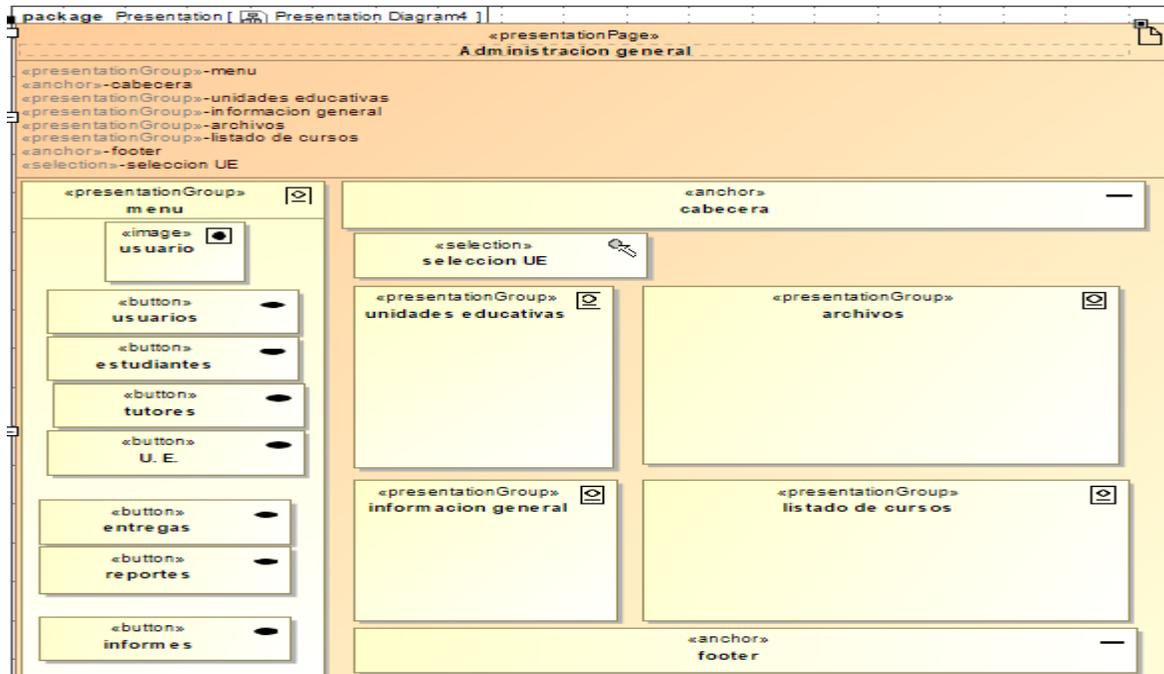


Figura 32 Modelo de presentación: Administración de unidad

3.3.5.6. Modelo de presentación: Administración de estudiantes

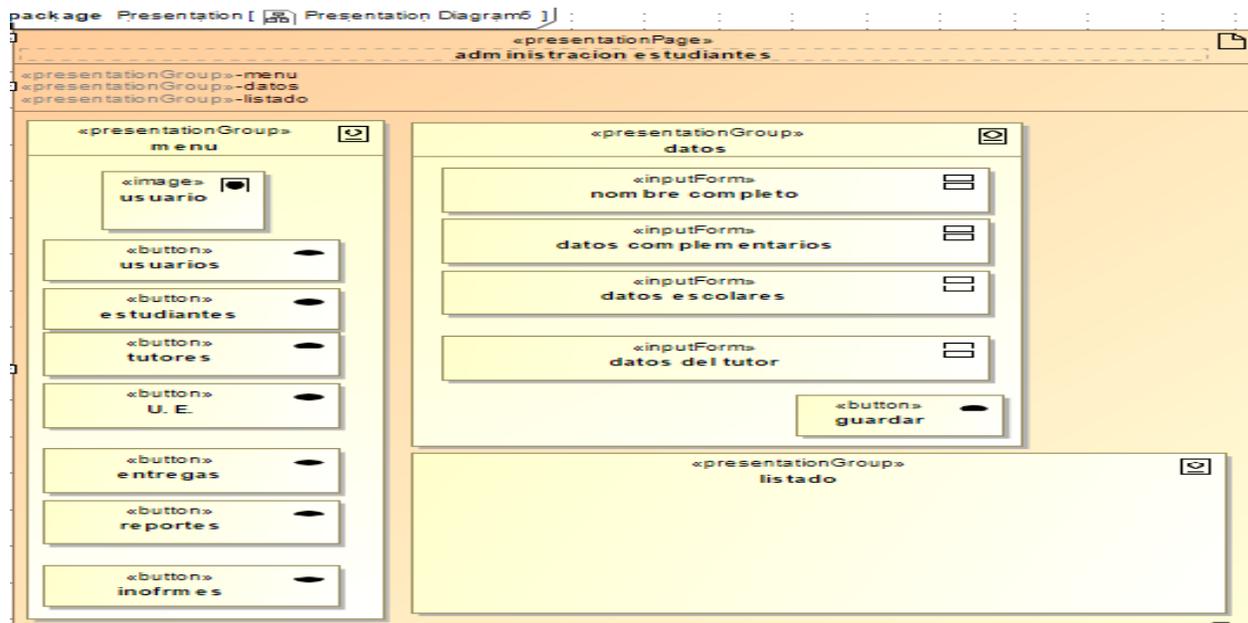


Figura 33 Modelo de presentación: Administración de estudiantes

3.3.5.7. Modelo de presentación: Administración de tutores

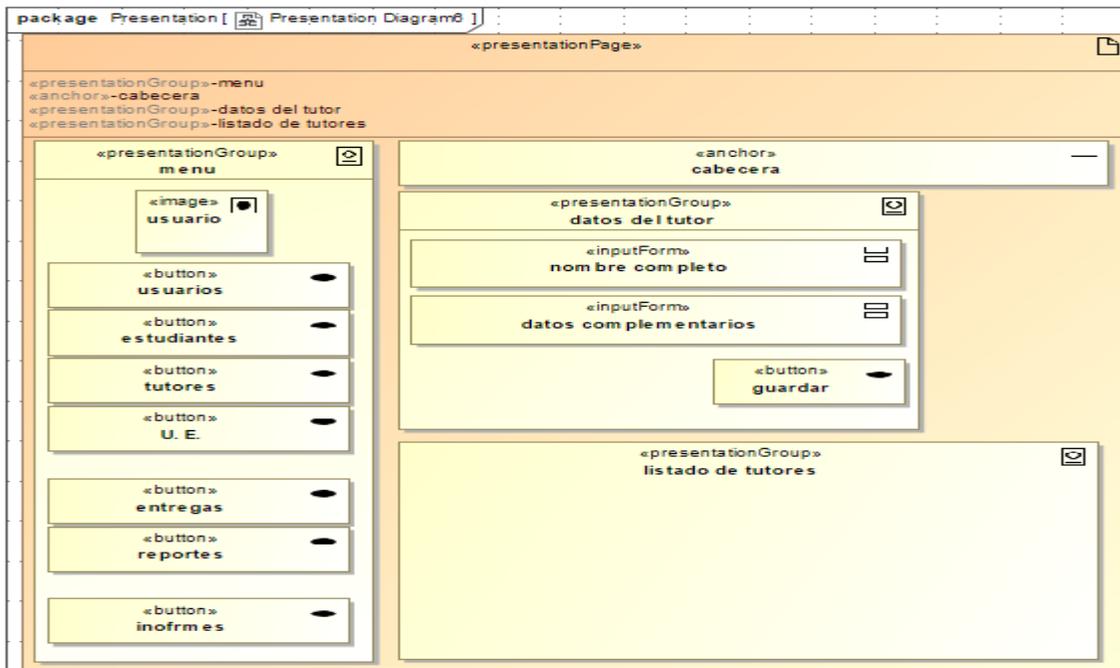


Figura 34 Modelo de presentación: Administración de tutores

3.3.5.8. Modelo de presentación: Administración de entregas

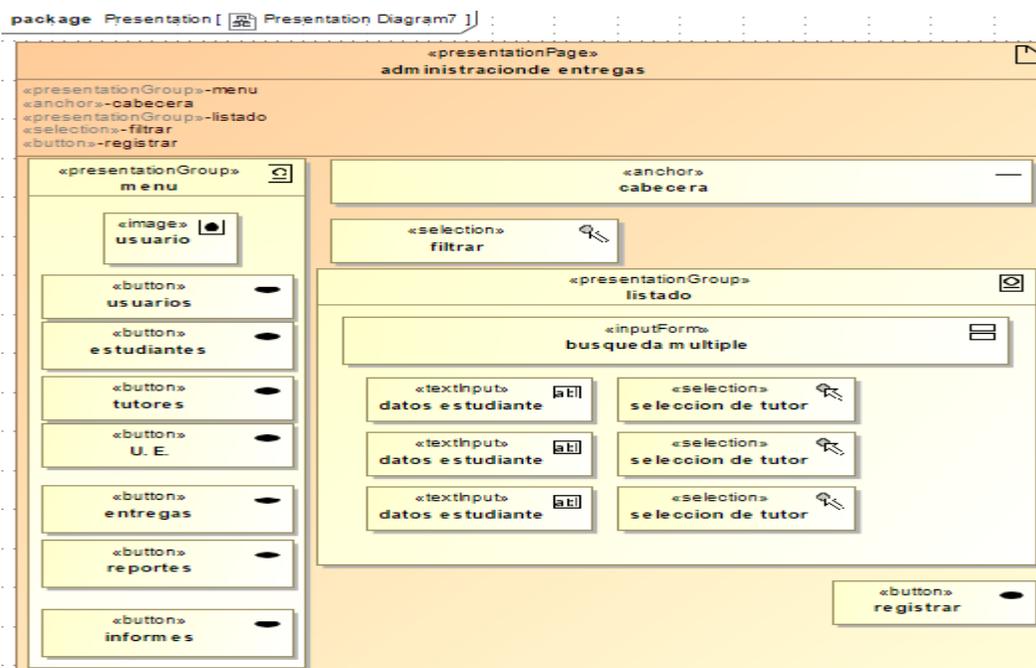


Figura 35 Modelo de presentación: Administración de entregas

3.3.5.9. Modelo de presentación: Administración de reportes

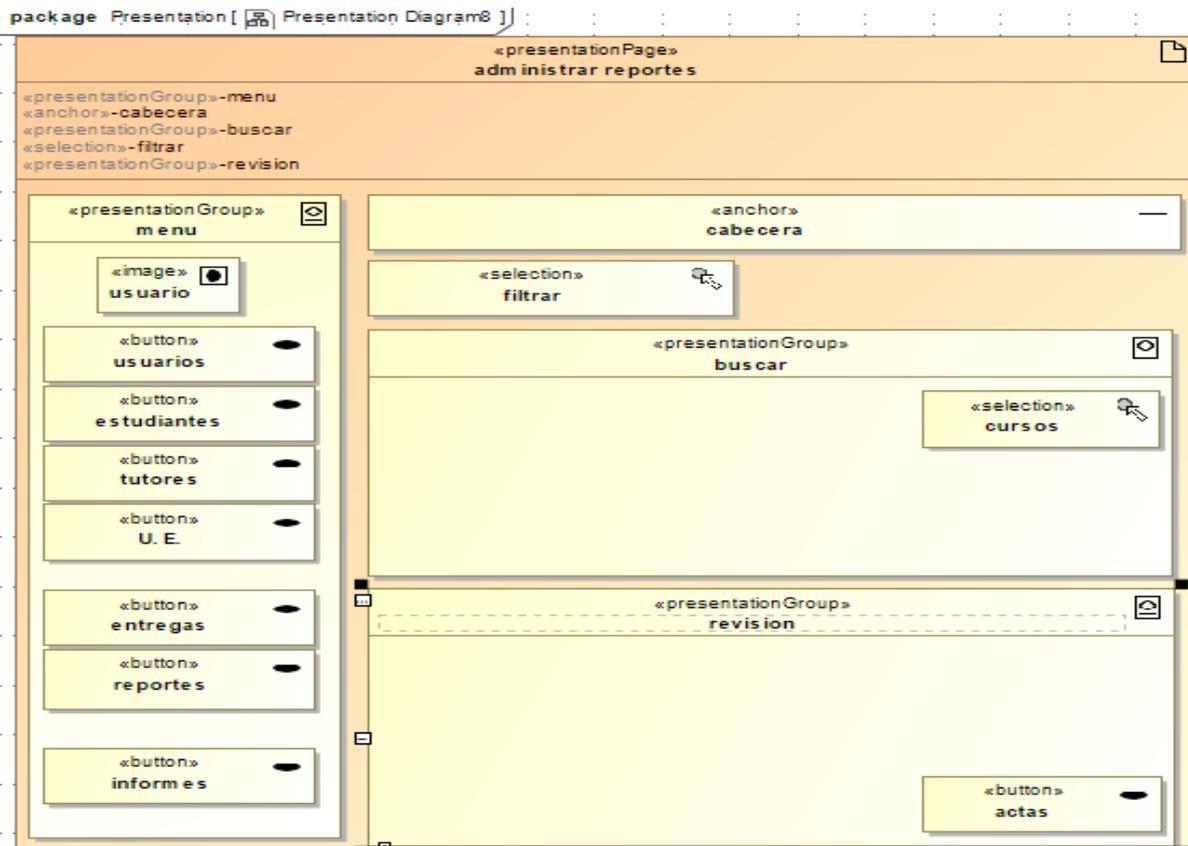


Figura 36 Modelo de presentación: Administración de reportes

3.3.6. Implementación del sistema

3.3.6.1. Interfaz de inicio: portada y consulta de datos

Objetivos	Inicio del sistema y consultas publicas
Descripción	Todos los usuarios e incluso agentes públicos pueden visualizar la portada con información acerca del proyecto e incluso pueden realizar la consulta de datos acerca del estudiante para la verificación y conformidad del mismo además de poder obtener la acta de entrega realizada en caso de que este registrado haya sido hecho.



Entrega de canastas

El gobierno municipal de la ciudad de El Alto mediante la dirección de educación realiza la entrega de canastas estudiantiles a las unidades educativas fiscales y de convenio en la ciudad de El Alto.

DIRECCIÓN DE EDUCACIÓN

GOBIERNO AUTÓNOMO MUNICIPAL DE LA CIUDAD DE EL ALTO

INICIAR SESION

Consultar datos

Realice consultas para la verificación de datos y el estado que se encuentra el estudiante, introduzca el número de Carnet de Identidad del estudiante a buscar o en su defecto el rufe que le proporciona la unidad educativa y por el cual fue registrado para la entrega.

Introduzca C.I. o RUFE.

Datos encontrados

NOMBRE COMPLETO

MAMANI CONDORI LEYDI BELEN

DATOS ADICIONALES

14861454 LP

40730589201685200.

3/9/2010

TUTORES

CONDORI MAYTA ALICIA ELSA 6965984 LP

MAMANI CHOQUEHUANCA JUAN MAMANI 3483781 LP

DATOS ESCOLARES

LATINOAMERICANO

PRIMARIA 4B

ESTADO

HABILITADO

VER ACTA

ACEPTAR

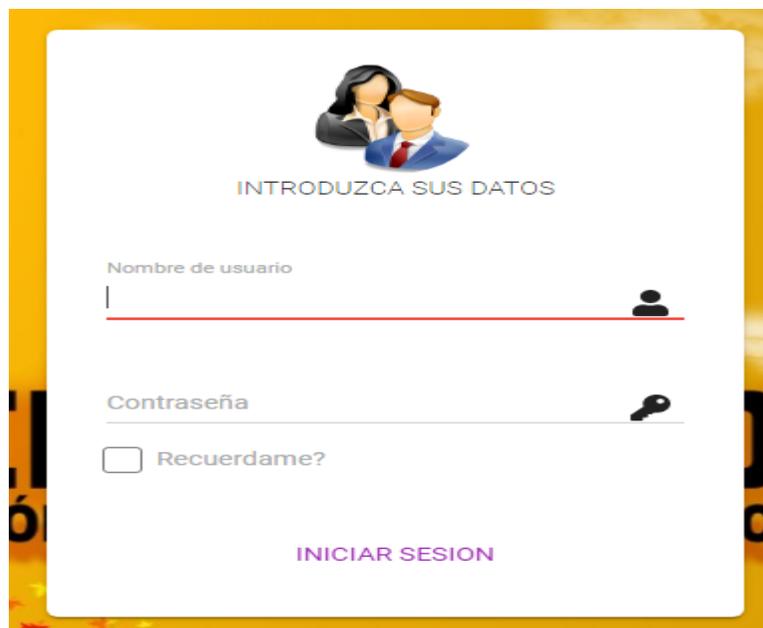
```

</div>
<div class="row" id="Consul">
  <div class="col-md-6 mx-auto" id="busq">
    <div class="title text-center">
      <h2>Consultar datos</h2>
      <p class="">Realice consultas para la verificación de datos y el estado que se encuentra el estudiante, introduzca el número de Carnet de Identidad del estudiante a buscar o en su defecto el rufe que le proporciona la unidad educativa y por el cual fue registrado para la entrega.</p>
    </div>
    <div class="form-group">
      <input type="text" class="form-control" placeholder="Introduzca C.I. o RUDE." id="idc">
      <div class="input-group">
        <span class="bmd-help">Es necesario contar con estos documentos para la consulta</span>
      </div>
    </div>
    <div class="float-right">
      <button type="button" class="btn btn-rose" id="btn-cons"><i class="fas fa-folder-open"></i></button>
    </div>
  </div>
</div>
<div class="row hidden" id="tarjeta">
  <div class="col-md-6 mx-auto">
    <div class="card">
      <div class="card-header card-header-info d-flex">
        <h3 class="card-title"><i class="fas fa-user-graduate"></i> Datos encontrados</h3>
      </div>
      <div class="card-body">
        <div id="d_est"></div>
        <div id="d_tutor"></div>
        <div id="d_ue"></div>
      </div>
      <div class="card-footer ml-auto">
        <button type="button" class="btn btn-default btn-sm"><i class="fas fa-print"></i> Ver acta</button>
        <button type="button" class="btn btn-success btn-sm" id="btn-acepta"><i class="fas fa-check"></i> Aceptar</button>
      </div>
    </div>
  </div>
</div>
</div>
</div>

```

3.3.6.2. Interfaz de inicio de sesión

Objetivos	Autenticación de usuarios
Descripción	Cada usuario registrado en el sistema podrá ingresar, mediante una autenticación de nombre de usuario y contraseña eso realizará una verificación del acceso que tenga.



```

<div class="card-body">
  <div class="form-group">
    <label for="username" class="bmd-label-floating">{{ __('Nombre de usuario') }}</label>

    <div class="">
      <input id="username" type="text" class="form-control @error('username') is-invalid @enderror" name="username" value="{{ old('username') }}" required autocomplete="username" autofocus>
      <span class="form-control-feedback">
        <i class="fas fa-user"></i>
      </span>
      @error('username')
        <span class="invalid-feedback" role="alert">
          <strong>{{ $message }}</strong>
        </span>
      @enderror
    </div>
  </div>

  <div class="form-group">
    <label for="password" class="bmd-label-floating">{{ __('Contraseña') }}</label>

    <div class="">
      <input id="password" type="password" class="form-control @error('password') is-invalid @enderror" name="password" required autocomplete="current-password">
      <span class="form-control-feedback">
        <i class="fas fa-key"></i>
      </span>
      @error('password')
        <span class="invalid-feedback" role="alert">
          <strong>{{ $message }}</strong>
        </span>
      @enderror
    </div>
  </div>

  <div class="form-group">
    <div class="form-check">
      <input class="form-check-input" type="checkbox" name="remember" id="remember" {{ old('remember') ? 'checked' : '' }}>
      Recuerdame?
      <span class="form-check-sign">
        <span class="check"></span>
      </span>
    </div>
  </div>
</div>
</div>
</div>

```

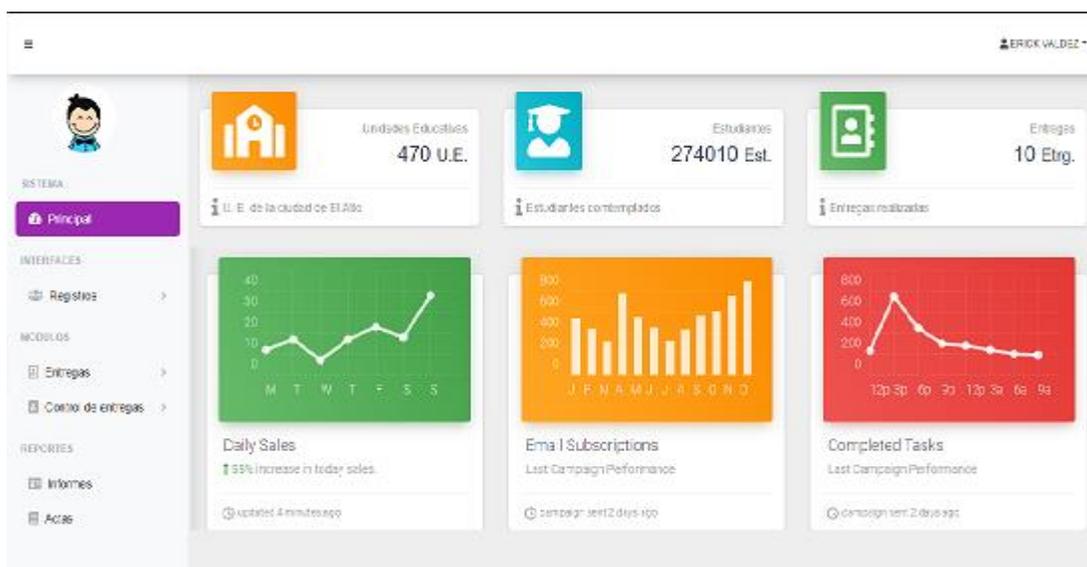
3.3.6.3. Funcionalidades del administrador

Objetivos

Funciones que los administradores pueden realizar

Descripción

El administrador podrá tener el acceso total al sistema, por el cual tendrá varios módulos a su disposición.

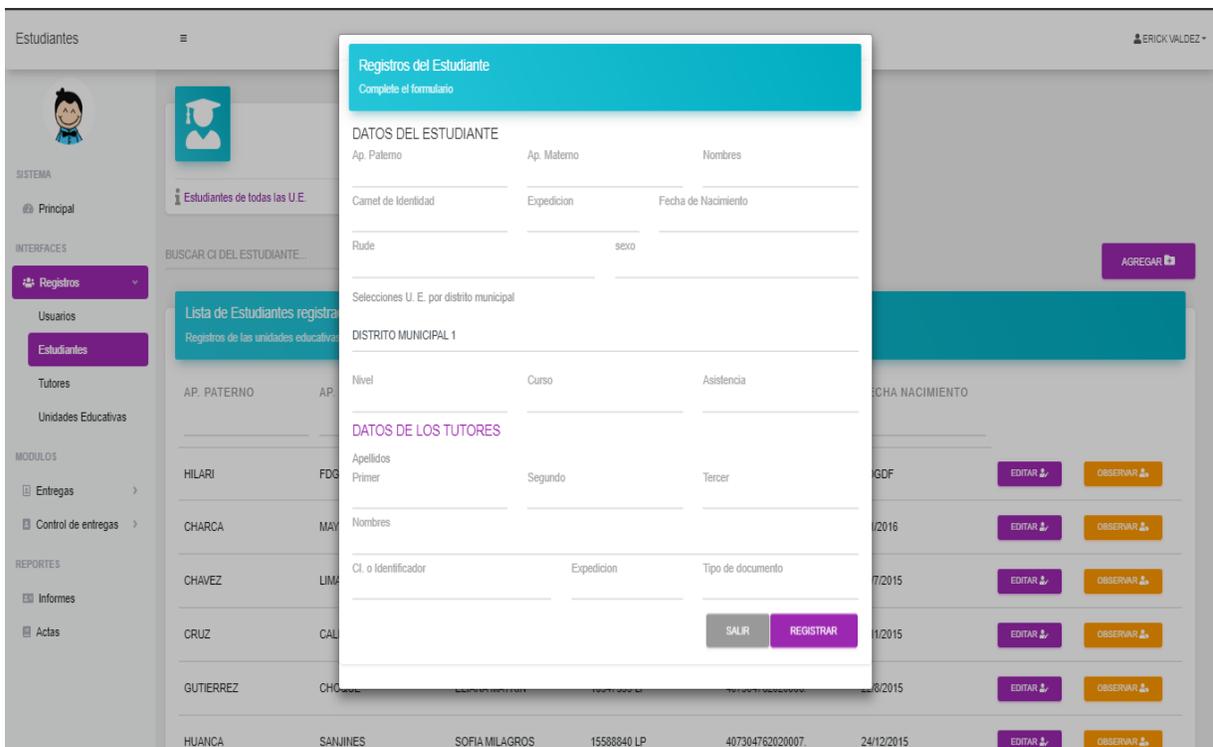


```

@section('content')
<div class="content mt-4">
<div class="container-fluid">
<div class="row">
<div class="col-md-4">
<div class="card card-stats">
<div class="card-header card-header-warning card-header-icon">
<div class="card-icon">
<i class="fas fa-school"></i>
</div>
<p class="card-category">Unidades Educativas</p>
<h3 class="card-title" id="todosue"></h3>
</div>
<div class="card-footer">
<div class="stats">
<i class="fas fa-info" style="font-size: 20px; margin-right: 5px;"></i> U. E. de la ciudad de El Alto
</div>
</div>
</div>
</div>
<div class="col-md-4">
<div class="card card-stats">
<div class="card-header card-header-info card-header-icon">
<div class="card-icon">
<i class="fas fa-user-graduate"></i>
</div>
<p class="card-category">Estudiantes</p>
<h3 class="card-title" id="todos"></h3>
</div>
<div class="card-footer">
<div class="stats">
<i class="fas fa-info" style="font-size: 20px; margin-right: 5px;"></i> Estudiantes contemplados
</div>
</div>
</div>
</div>
<div class="col-md-4">
<div class="card card-stats">
<div class="card-header card-header-success card-header-icon">
<div class="card-icon">
<i class="far fa-address-book"></i>
</div>
<p class="card-category">Entregas</p>
<h3 class="card-title" id="todos-ent"></h3>
</div>
<div class="card-footer">
<div class="stats">
<i class="fas fa-info" style="font-size: 20px; margin-right: 5px;"></i> Entregas realizadas
</div>
</div>
</div>
</div>

```

3.3.6.4. Funcionalidades de registros




```

@section('content')
<div class="content mt-4">
  <div class="container-fluid">
    <div class="row">
      <div class="col-md-6 col-lg-4">
        <div class="card card-stats">
          <div class="card-header card-header-rose card-header-icon">
            <div class="card-icon">
              <i class="fas fa-user"></i>
            </div>
            <p class="card-category">Usuarios registrados</p>
            <h3 class="card-title" id="todos"></h3>
          </div>
          <div class="card-footer">
            <div class="stats">
              <i class="fas fa-info" style="font-size: 20px; margin-right: 5px;"></i>
              <a href="javascript:">Usuarios del sistema</a>
            </div>
          </div>
        </div>
      </div>
      <div class="col-md-12 col-lg-8">
        <a class="btn btn-primary btn-nr text-white float-right" id="vermodal">Agregar <i class="fas fa-folder-plus"></i></a>
      </div>
    </div>
    <div id="respuesta"></div>
    <div class="row">
      <div class="col-md-12">
        <div class="card">
          <div class="card-header card-header-primary">
            <h4 class="card-title">Lista de Unidades Educativas</h4>
            <p class="card-category">Registro de las unidades educativas de la ciudad de El Alto</p>
          </div>
          <div class="card-body">
            <div class="table-responsive">
              <table class="table" id="dataTable">
            </table>
          </div>
        </div>
      </div>
    </div>
  </div>
</div>
@endsection

```

Registro de usuarios

Registros de Usuarios

Complete el formulario

Ap.paterno Ap.materno

Nombres

Correo Nombre de usuario

Dependencia

Rol

SALIR
ACEPTAR

```

@section('modales')
<div class="modal fade" id="AddModal" tabindex="-1" role="dialog" aria-labelledby="exampleModalLabel" aria-hidden="true">
  <div class="modal-dialog" role="document">
    <div class="modal-content">
      <div class="card">
        <div class="card-header card-header-primary">
          <h4 class="card-title">Registros de Usuarios</h4>
          <p class="card-category">Complete el formulario</p>
        </div>
        <div class="card-body">
          <form id="form-registro">
            <input type="hidden" name="id_us" id="unico">
            <div class="col-md-6">
              <div class="form-group">
                <label class="bmd-label-floating">Ap.paterno</label>
                <input type="text" name="ap_us" id="ap_us" class="form-control" autofocus>
              </div>
            </div>
            <div class="col-md-6">
              <div class="form-group">
                <label class="bmd-label-floating">Ap.materno</label>
                <input type="text" name="am_us" id="am_us" class="form-control">
              </div>
            </div>
            <div class="col-md-12">
              <div class="form-group">
                <label class="bmd-label-floating">Nombres</label>
                <input type="text" name="no_us" id="no_us" class="form-control" required>
              </div>
            </div>
            <div class="col-md-6">
              <div class="form-group">
                <label class="bmd-label-floating">Correo</label>
                <input type="email" name="em_us" id="em_us" class="form-control" required>
              </div>
            </div>
            <div class="col-md-6">
              <div class="form-group">
                <label class="bmd-label-floating">Nombre de usuario</label>
                <input type="text" name="un_us" id="un_us" class="form-control" required>
              </div>
            </div>
          </form>
        </div>
      </div>
    </div>
  </div>
</div>

```

Listado de estudiantes

Estudiantes ERICK VALDEZ

Estudiantes registrados
274010 Est.

Estudiantes de todas las U.E.

BUSCAR CI DEL ESTUDIANTE... AGREGAR

Lista de Estudiantes registrados
Registros de las unidades educativas de la ciudad de El Alto

AP. PATERNO	AP. MATERNO	NOMBRES	CI	RUDE	FECHA NACIMIENTO		
HILARI	FDGD	JOSSUE	1616161 FDGDF	FDGDF	FDGDF	EDITAR	OBSERVAR
CHARCA	MAYTA	ARTURO VIDAL	16631974 LP	407304762020003	7/1/2016	EDITAR	OBSERVAR
CHAVEZ	LIMACHI	GENESIS BELEN	16197068 LP	407304762020004	11/7/2016	EDITAR	OBSERVAR
CRUZ	CALLAMPA	EYNAR BLASS	16602231 LP	407304762020005	3/11/2016	EDITAR	OBSERVAR
GUTIERREZ	CHOQUE	ELIANA MAYRIN	16347639 LP	407304762020006	22/8/2016	EDITAR	OBSERVAR
HUANCA	SANJINES	SOFIA MILAGROS	16668840 LP	407304762020007	24/12/2016	EDITAR	OBSERVAR
MAMANI	MAMANI	ANDRES GONALO	1641324 LP	407304762020008	1/3/2016	EDITAR	OBSERVAR

```

<div class="row">
  <div class="col-md-12">
    <div class="card">
      <div class="card-header card-header-info">
        <h4 class="card-title">Lista de Estudiantes registrados</h4>
        <p class="card-category">Registros de las unidades educativas de la ciudad de El Alto</p>
      </div>
      <div class="card-body">
        <div class="table-responsive">
          <table class="table">
            <thead class=" text-primary text-uppercase">
              <th>
                <label class="bmd-label-static">Ap. Paterno</label>
                <input type="text" name="ap_bs" id="ap_bs" class="form-control busca">
              </th>
              <th>
                <label class="bmd-label-static">Ap. Materno</label>
                <input type="text" name="am_bs" id="am_bs" class="form-control busca">
              </th>
              <th>
                <label class="bmd-label-static">Nombres</label>
                <input type="text" name="no_bs" id="no_bs" class="form-control busca">
              </th>
              <th>
                <label class="bmd-label-static">ci</label>
                <input type="text" name="ci_bs" id="ci_bs" class="form-control busca">
              </th>
              <th>
                <label class="bmd-label-static">rude</label>
                <input type="text" name="ru_bs" id="ru_bs" class="form-control busca">
              </th>
              <th>
                <label class="bmd-label-static">fecha nacimiento</label>
                <input type="text" name="fn_bs" id="fn_bs" class="form-control busca">
              </th>
            </thead>
            <tbody id="dataTable">
              <tr>
                <td>HILARI</td>
                <td>FDGD</td>
                <td>JOSSUE</td>
                <td>1616161 FDGDF</td>
                <td>FDGDF</td>
                <td>FDGDF</td>
                <td><span style="background-color: #6c757d; color: white; padding: 2px 5px;">EDITAR</span></td>
                <td><span style="background-color: #ffc107; color: white; padding: 2px 5px;">OBSERVAR</span></td>
              </tr>
              <tr>
                <td>CHARCA</td>
                <td>MAYTA</td>
                <td>ARTURO VIDAL</td>
                <td>16631974 LP</td>
                <td>407304762020003</td>
                <td>7/1/2016</td>
                <td><span style="background-color: #6c757d; color: white; padding: 2px 5px;">EDITAR</span></td>
                <td><span style="background-color: #ffc107; color: white; padding: 2px 5px;">OBSERVAR</span></td>
              </tr>
              <tr>
                <td>CHAVEZ</td>
                <td>LIMACHI</td>
                <td>GENESIS BELEN</td>
                <td>16197068 LP</td>
                <td>407304762020004</td>
                <td>11/7/2016</td>
                <td><span style="background-color: #6c757d; color: white; padding: 2px 5px;">EDITAR</span></td>
                <td><span style="background-color: #ffc107; color: white; padding: 2px 5px;">OBSERVAR</span></td>
              </tr>
              <tr>
                <td>CRUZ</td>
                <td>CALLAMPA</td>
                <td>EYNAR BLASS</td>
                <td>16602231 LP</td>
                <td>407304762020005</td>
                <td>3/11/2016</td>
                <td><span style="background-color: #6c757d; color: white; padding: 2px 5px;">EDITAR</span></td>
                <td><span style="background-color: #ffc107; color: white; padding: 2px 5px;">OBSERVAR</span></td>
              </tr>
              <tr>
                <td>GUTIERREZ</td>
                <td>CHOQUE</td>
                <td>ELIANA MAYRIN</td>
                <td>16347639 LP</td>
                <td>407304762020006</td>
                <td>22/8/2016</td>
                <td><span style="background-color: #6c757d; color: white; padding: 2px 5px;">EDITAR</span></td>
                <td><span style="background-color: #ffc107; color: white; padding: 2px 5px;">OBSERVAR</span></td>
              </tr>
              <tr>
                <td>HUANCA</td>
                <td>SANJINES</td>
                <td>SOFIA MILAGROS</td>
                <td>16668840 LP</td>
                <td>407304762020007</td>
                <td>24/12/2016</td>
                <td><span style="background-color: #6c757d; color: white; padding: 2px 5px;">EDITAR</span></td>
                <td><span style="background-color: #ffc107; color: white; padding: 2px 5px;">OBSERVAR</span></td>
              </tr>
              <tr>
                <td>MAMANI</td>
                <td>MAMANI</td>
                <td>ANDRES GONALO</td>
                <td>1641324 LP</td>
                <td>407304762020008</td>
                <td>1/3/2016</td>
                <td><span style="background-color: #6c757d; color: white; padding: 2px 5px;">EDITAR</span></td>
                <td><span style="background-color: #ffc107; color: white; padding: 2px 5px;">OBSERVAR</span></td>
              </tr>
            </tbody>
          </table>
        </div>
      </div>
    </div>
  </div>
</div>

```

Listado de unidades educativas

Unidades Educativas ERICK VALDEZ

SISTEMA

Principal

INTERFACES

- Registros
 - Usuarios
 - Estudiantes
 - Tutores
- Unidades Educativas

MODULOS

- Entregas
- Control de entregas

REPORTES

- Informes
- Actas



Unidades educativas
470 U.E.

DISTRITO MUNICIPAL 1



SIE 40730001

12 DE OCTUBRE MAÑANA

881 Estudiantes



ARCHIVOS

BUSCAR UE POR ID... AGREGAR

Lista de Unidades Educativas

Registro de las unidades educativas de la ciuda de El Alto

N°	ID	NOMBRE	TURNO	DISTRITO	OPCIONES
1	40730001	12 DE OCTUBRE MAÑANA	M	1	REVISAR EDITAR
2	40730002	12 DE OCTUBRE TARDE	T	1	REVISAR EDITAR
3	40730003	8 DE AGOSTO	T	1	REVISAR EDITAR
4	40730004	MODESTO OMISTE	M	1	REVISAR EDITAR
5	40730005	PILOTO BOLIVIA	M	1	REVISAR EDITAR

```

<div class="row">
  <div class="col-md-12 col-lg-4">
    <div class="input-group no-border">
      <input type="text" id="txt-bus" class="form-control" placeholder="Buscar UE por ID...">
    </div>
  </div>
  <div class="col-md-12 col-lg-8">
    <a class="btn btn-primary btn-nr text-white float-right" id="vermodal">Agregar <i class="fas fa-folder-plus"></i></a>
  </div>
</div>
<div id="respuesta"></div>
<div class="row">
  <div class="col-md-12">
    <div class="card">
      <div class="card-header card-header-primary">
        <h4 class="card-title">Lista de Unidades Educativas</h4>
        <p class="card-category">Registro de las unidades educativas de la ciuda de El Alto</p>
      </div>
      <div class="card-body">
        <div class="table-responsive">
          <table class="table" id="dataTable">
            </table>
          </div>
        </div>
      </div>
    </div>
  </div>
</div>
</div>

```

Listado de estudiantes a entregar

Realizar Entregas ERICK VALDEZ

DISTRITO MUNICIPAL 2
REPUBLICA DE FRANCIA TARDE

Estudiantes de la Unidad Educativa
Registro de las estudiantes

Ap. paterno	Ap. materno	Nombre	C.I.	RUDE	Curso	Unidad Educativa	Estado	
ALANCOCA	TAVAREZ	MARCO ANTONIO	14292873 LP	4073007220160840.	1A	REPUBLICA DE FRANCIA TARDE	HABILITADO	ENTREGAR
CALSINA	HUANCA	SARA LIZET	13608800 LP	807000512013380.	1A	REPUBLICA DE FRANCIA TARDE	HABILITADO	ENTREGAR
CAMPOSALADO	FLORES	YAELENO	10081763 LP	4073008020133A.	1A	REPUBLICA DE FRANCIA TARDE	HABILITADO	ENTREGAR
CASTILLO	APAZA	CARLOS ANDRES	13693209 LP	407304092013488.	1A	REPUBLICA DE FRANCIA TARDE	HABILITADO	ENTREGAR
CHUQUIMIA	FERNANDEZ	CAMILA GABRIELA	12831698 LP	40730080201352.	1A	REPUBLICA DE FRANCIA TARDE	HABILITADO	ENTREGAR
COLQUE	RAMOS	JOSE MANUEL	12460062 LP	407302852013690.	1A	REPUBLICA DE FRANCIA TARDE	HABILITADO	ENTREGAR
CONDORI	CHOQUE	NICOL ESTHER	10919012 LP	8073009720153880.	1A	REPUBLICA DE FRANCIA TARDE	HABILITADO	ENTREGAR

```

<label>Buscar por Distrito municipal y la U. E.</label>
<div class="row">
  <div class="col-md-12 col-lg-6">
    <div class="input-group no-border">
      <select class="form-control" id="dm_ue"> ...
    </select>
    </div>
  </div>
  <div class="col-md-12 col-lg-6">
    <div class="input-group no-border" id="Ues-lista">
      ...
    </div>
  </div>
</div>
<div id="respuesta"></div>
<div class="row">
  <div class="col-md-12">
    <div class="card">
      <div class="card-header card-header-primary">
        <h4 class="card-title">Lista de Estudiantes Registrados</h4>
        <p class="card-category">Registro de los estudiantes</p>
      </div>
      <div class="card-body">
        <div class="table-responsive">
          <table class="table">
            <thead> ...
            </thead>
            <tbody id="dataTable">
              ...
            </tbody>
          </table>
        </div>
      </div>
    </div>
  </div>
</div>
</div>
</div>
</div>
@endsection

```


Confirmación de entrega y observación

Realizar Entregas



SISTEMA

- Principal

INTERFACES

- Registros

MODULOS

- Entregas
- Control de entregas

REPORTES

- Informes
- Actas

Formulario de entrega

Complete y Revise Datos

DATOS DEL ESTUDIANTE

Ap. Paterno CALSINA	Ap. Materno HUANCA	Nombre SARA LIZET
No. De carnet 13508800	Expedicion LP	Sexo F
Ruda 807000512013380.	Fecha de Nacimiento 17/8/2008	
Nivel SECUNDARIA	Curso O Paralelo 1A	

DATOS DEL TUTOR

Apellidos Primer Ap. CALSINA	Segundo Ap. MACHACA	Tercer Ap.
Nombres EDWIN		
No. De carnet o Identificador 4807227	Expedicion LP	Tipo de documento CI

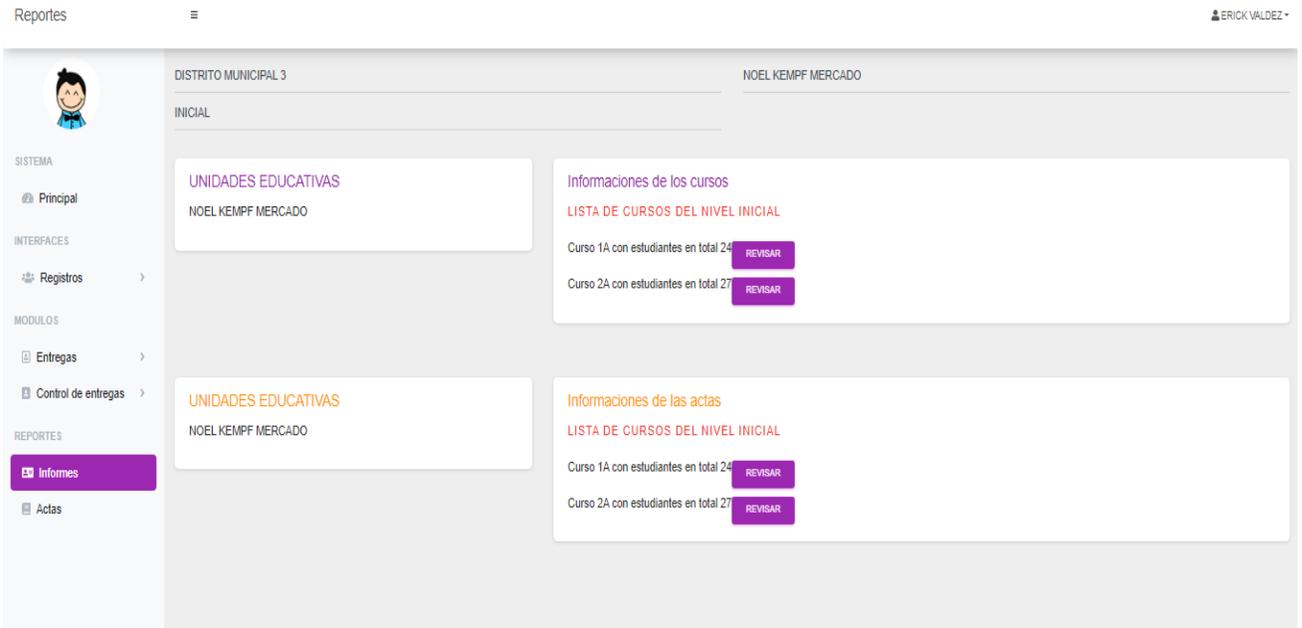
OBSERVACIONES AL ENTREGAR

Observaciones

Ingrese si ocurrio alguna observacion

```
<div class="row hidden" id="vent-form">
  <div class="col-md-8">
    <div class="card">
      <div class="card-header card-header-primary">
        <h4 class="card-title">Formulario de entrega</h4>
        <p class="card-category">Complete y Revise Datos</p>
      </div>
      <div class="card-body">
        <form action="entregas" method="POST" id="form-entr">
          @csrf
          <h4>DATOS DEL ESTUDIANTE</h4>
          <input type="hidden" name="id_es" id="ides">
          <div class="row">
            <div class="row">
            </div>
            <div class="row">
            </div>
            <div class="row">
            </div>
            <div class="row">
            </div>
            <div class="row">
            </div>
          <h4>DATOS DEL TUTOR</h4>
          <input type="hidden" name="id_tu" id="idtu">
          <label>Apellidos</label>
          <div class="row">
            <div class="row">
            </div>
            <div class="row">
            </div>
            <div class="row">
            </div>
          <h4>OBSERVACIONES AL ENTREGAR</h4>
          <div class="row">
            </div>
          <input type="hidden" name="id_us" value="{{Auth:user()->id}}">
          <button type="submit" id="btn-fin" class="btn btn-primary pull-right">confirmar entrega <i class="fas fa-check"></i></button>
          <div id="mensaje"></div>
        </form>
        <a class="btn btn-default text-white" id="btn-sali">SALIR <i class="fas fa-times-circle"></i></a>
      </div>
    </div>
  </div>
</div>
<div id="respuesta"></div>
```

Revisión de reportes



```
<div class="row">
  <div class="col-md-6">
  </div>
  <div class="col-md-6">
  </div>
</div>
<div class="row">
  <div class="col-md-4">
    <div class="card">
      <div class="card-body" style="min-height: 100px">
        <h4 class="text-primary">UNIDADES EDUCATIVAS</h4>
        <p id="nombre-ue">Se mostrara los datos de la unidad educativa tal cual se encuentra en los registros de la direccion de educacion</p>
      </div>
    </div>
  </div>
  <div class="col-md-8">
    <div class="card">
      <div class="card-body" style="min-height: 100px">
        <h4 class="text-primary">Informaciones de los cursos</h4>
        <div id="cursos-enc">Se mostrara los datos de los cursos de la unidad educativa tal como se obtuvieron los registros en la direccion de educacion y sus distritales</div>
      </div>
    </div>
  </div>
</div>
<div class="row">
  <div class="col-md-4">
    <div class="card">
      <div class="card-body" style="min-height: 100px">
        <h4 class="text-warning">UNIDADES EDUCATIVAS</h4>
        <p id="nombre-uea">Se mostrara los actas correspondientes de la unidad educativa tal cual se generaron en los registros de la direccion de educacion</p>
      </div>
    </div>
  </div>
  <div class="col-md-8">
    <div class="card">
      <div class="card-body" style="min-height: 100px">
        <h4 class="text-warning">Informaciones de las actas</h4>
        <div id="cursos-enca">Se mostrara los actas de los cursos de las unidades educativas tal como se generaron a partir de los registros en la direccion de educacion y sus distritales</div>
      </div>
    </div>
  </div>
</div>
</div>
```

- Módulos para usuarios del sistema

Listado de para entregas por UE y curso

Entregas por curso ERICK VALDEZ

SISTEMA

- Principal

INTERFACES

- Registros

MODULOS

- Entregas**
- Control de entregas

REPORTES

- Informes
- Actas



Unidad Educativa
12 DE OCTUBRE TARDE

Unidades educativas registradas



Estudiantes de la U.E.
730 Est.

Estudiantes registrados



Curso de U.E.
4A (25 Estudiantes)

SECUNDARIA

DISTRITO MUNICIPAL 1

12 DE OCTUBRE TARDE

SECUNDARIA

4A (25 ESTUDIANTES)

Estudiantes de la Unidad Educativa
Registro de las estudiantes por curso

N°	RUDE	NOMBRE COMPLETO	TUTORES	ESTADO	OPCIONES
1	407300012010370.	ABIRARI CONDORI AMAYA INDIRA	ALI ABERARI MILENKHA CI: 4949115 LP	HABILITADO	<input type="checkbox"/> EDITAR
2	407300012010622.	ALCON REAL ELVIN GABRIEL	REAL VILA NATALY VIVIANA CI: 6141705 LP	HABILITADO	<input type="checkbox"/> EDITAR
3	4073005920101172.	APAZA SALINAS CESAR KEVIN	SALINAS CHAMBI ROSA CI: 6195464 LP	HABILITADO	<input type="checkbox"/> EDITAR
4	3068000720101453.	CALLAPA MAMANI ALEXANDRE LUIS	MAMANI CRUZ LIZ JAQUELINE CI: 4940074 LP	HABILITADO	<input type="checkbox"/> EDITAR

```

<div class="row">
</div>
<div class="row">
</div>
<div class="row">
  <div class="col-md-12">
    <div class="card">
      <div class="card-header card-header-primary">
        <h4 class="card-title">Estudiantes de la Unidad Educativa</h4>
        <p class="card-category">Registro de las estudiantes por curso</p>
      </div>
      <div class="card-body">
        <div id="report-curso"></div>
        <form id="form-grupal">
          <div class="table-responsive" id="listas-curso">
            </div>
        </form>
      </div>
    </div>
    <div id="salida"></div>
  </div>
</div>
<div class="row hidden" id="a-subir">
  <div class="col-md-12">
    <div class="card">
      <div class="card-header card-header-primary">
        <h4 class="card-title">Completar entrega</h4>
      </div>
      <div class="card-body">
        <form action="/subir_pdf" method="POST" class="dropzone" id="my-dropzone" files="true">
          <input type="hidden" name="token" value="{{ csrf_token() }}">
          <input type="hidden" name="curso" id="curso">
          <input type="hidden" name="nivel" id="nivel">
          <input type="hidden" name="uesss" id="uesss">
          <div class="dz-message" style="height:100px;">
            Arrastre el archivo o haga click para subir al sistema
          </div>
          <div class="dropzone-previews"></div>
          <button type="submit" class="btn btn-success btn-sm float-right" id="submit">Subir Archivos</button>
        </form>
      </div>
    </div>
  </div>
</div>
</div>

```

Completar entregas múltiples por curso

Entregas por curso

ID	Nombre	Dirección	Estado	Acciones
22	QUISBERT COPA JIMMY ANDREZ	COPA MAMANI MARTHA CI: 5470843 LP	HABILITADO	<input checked="" type="checkbox"/> EDITAR
23	SARMIENTO QUINO EDILSON	SARMIENTO CANAZA FRANZ CI: 6007718 LP	HABILITADO	<input checked="" type="checkbox"/> EDITAR
24	TERRAZAS MAMANI REYCHEL ROMINA	TERRAZAS AYCA RENE CI: 2916272 CB	HABILITADO	<input checked="" type="checkbox"/> EDITAR
25	VELASQUEZ LO...	ENA CI: 4760092 LP	HABILITADO	<input checked="" type="checkbox"/> EDITAR

Completar entrega

Arrastre el archivo o haga click para subir al sistema

[SUBIR ARCHIVOS](#)

```

<div class="modal fade" id="myModal" tabindex="-1" role="dialog">
  <div class="modal-dialog modal-dialog-centered" role="document">
    <div class="modal-content">
      <div class="modal-header">
        <h5 class="modal-title">Confirmar entregas</h5>
        <button type="button" class="close" data-dismiss="modal" aria-label="Close">
          <i class="fas fa-times"></i>
        </button>
      </div>
      <div class="modal-body">
        <div class="form-group">
          <label class="label-control">Seleccione la fecha de entrega</label>
          <input type="text" class="form-control datetimepicker" value="10/05/2020">
        </div>
      </div>
      <div class="modal-footer">
        <button type="button" class="btn btn-primary btn-sm text-white" id="btn-grupalgg">Aceptar</button>
        <button type="button" class="btn btn-default btn-sm text-white" data-dismiss="modal">Cerrar</button>
      </div>
    </div>
  </div>
</div>
  
```

Completar entregas con subida de acta

Completar entrega

30.7 KB

file.pdf

Remove file

23.6 KB

file.docx

Remove file

[SUBIR ARCHIVOS](#)

Control de entregas realizadas

Revisar Entregas ERICK VALDEZ

Principal

Registros

Entregas

Control de entregas

Entrega a U.E.

Entrega a estudiantes

Informes

Actas

BUSCAR POR CI O NUM DE ENTREGA.....

Entregas Realizadas

Registro de las entregas hechas a los estudiantes

N° ENT.	NOMBRE DEL ESTUDIANTE	ENTREGADO POR	FECHA	ESTADO	DETALLE	EDITAR
112	QUISPE VALDERRAMA CADMIEL EDGAR	JUANITO	2020-11-15 07:00:36	ENTREGADO	DETALLE	EDITAR
113	TARQUI TAMBO ALAN PEÑAFORD	JUANITO	2020-11-15 07:00:36	ENTREGADO	DETALLE	EDITAR
114	TINCUTA CEREZO NICOL ANAHI	JUANITO	2020-11-15 07:00:36	ENTREGADO	DETALLE	EDITAR
115	QUISPE PALMA OLIVIA ADELET	JUANITO	2020-11-15 07:07:27	ENTREGADO	DETALLE	EDITAR
116	VARGAS RAMIREZ ROY DANNY	JUANITO	2020-11-22 04:07:50	ENTREGADO	DETALLE	EDITAR
117	YANARICO MAMANI CRISTAL SONIA	JUANITO	2020-11-22 04:07:50	ENTREGADO	DETALLE	EDITAR
118	ZULETA MAYTA YENY EMELY	JUANITO	2020-11-22 04:14:27	ENTREGADO	DETALLE	EDITAR
119	ZULETA MAYTA YENY BELEN	JUANITO	2020-11-22 04:14:27	ENTREGADO	DETALLE	EDITAR

```
<div id="respuesta"></div>
<div class="row">
  <div class="col-md-12">
    <div class="card">
      <div class="card-header card-header-primary">
        <h4 class="card-title">Entregas Realizadas</h4>
        <p class="card-category">Registro de las entregas hechas a los estudiantes</p>
      </div>
      <div class="card-body">
        <div class="table-responsive">
          <table class="table">
            <thead>
              <tr class="text-primary">
                <th>N° ENT.</th>
                <th>NOMBRE DEL ESTUDIANTE</th>
                <th>ENTREGADO POR</th>
                <th>FECHA</th>
                <th>ESTADO</th>
              </tr>
            </thead>
            <tbody id="lis-est-ue">
              <tr>
                <td>112</td>
                <td>QUISPE VALDERRAMA CADMIEL EDGAR</td>
                <td>JUANITO</td>
                <td>2020-11-15 07:00:36</td>
                <td>ENTREGADO</td>
              </tr>
              <tr>
                <td>113</td>
                <td>TARQUI TAMBO ALAN PEÑAFORD</td>
                <td>JUANITO</td>
                <td>2020-11-15 07:00:36</td>
                <td>ENTREGADO</td>
              </tr>
              <tr>
                <td>114</td>
                <td>TINCUTA CEREZO NICOL ANAHI</td>
                <td>JUANITO</td>
                <td>2020-11-15 07:00:36</td>
                <td>ENTREGADO</td>
              </tr>
              <tr>
                <td>115</td>
                <td>QUISPE PALMA OLIVIA ADELET</td>
                <td>JUANITO</td>
                <td>2020-11-15 07:07:27</td>
                <td>ENTREGADO</td>
              </tr>
              <tr>
                <td>116</td>
                <td>VARGAS RAMIREZ ROY DANNY</td>
                <td>JUANITO</td>
                <td>2020-11-22 04:07:50</td>
                <td>ENTREGADO</td>
              </tr>
              <tr>
                <td>117</td>
                <td>YANARICO MAMANI CRISTAL SONIA</td>
                <td>JUANITO</td>
                <td>2020-11-22 04:07:50</td>
                <td>ENTREGADO</td>
              </tr>
              <tr>
                <td>118</td>
                <td>ZULETA MAYTA YENY EMELY</td>
                <td>JUANITO</td>
                <td>2020-11-22 04:14:27</td>
                <td>ENTREGADO</td>
              </tr>
              <tr>
                <td>119</td>
                <td>ZULETA MAYTA YENY BELEN</td>
                <td>JUANITO</td>
                <td>2020-11-22 04:14:27</td>
                <td>ENTREGADO</td>
              </tr>
            </tbody>
          </table>
        </div>
      </div>
    </div>
  </div>
</div>
</div>
```

Revisión de entrega por estudiante

Revisar Entregas

ERICK VALDEZ

BUSCAR POR CI O NUM DE ENTREGA.....

Entregas Realizadas
Registro de las entregas hechas a los estudiantes

N° ENT.	NOMBRE DEL ESTUDIANTE	ENTREGADO POR	FECHA	ESTADO	DETALLE	EDITAR
112	QUISPE VALDERRAMA CADMIEL EDGAR	JUANITO	2020-11-15 07:00:36	ENTREGADO	DETALLE	EDITAR
113	TARQUI TAMBO ALAN PEÑAFOR	JUANITO	2020-11-15 07:00:36	ENTREGADO	DETALLE	EDITAR
114	TINCUTA CEREZO NICOLANAH	JUANITO	2020-11-15 07:00:36	ENTREGADO	DETALLE	EDITAR
115	QUISPE PALMA OLIVIA ADELET	JUANITO	2020-11-15 07:07:27	ENTREGADO	DETALLE	EDITAR
116	VARGAS RAMIREZ ROY DANNY	JUANITO	2020-11-22 04:07:50	ENTREGADO	DETALLE	EDITAR

Registro de la entrega
Datos acerca de la entrega

REVISION DE ENTREGA

AL ESTUDIANTE
QUISPE VALDERRAMA CADMIEL EDGAR
EN COMPAÑIA DEL TUTOR
VALDERRAMA PAREDES GRETTY IRMA
ENTREGADO POR
JUANITO
EN FECHA: 2020-11-15T07:00:36.000000Z

SALIR

VERACTA

ESTUDIANTE TUTORES UNIDAD EDUCATIVA

DATOS DEL ESTUDIANTE

Nombre CADMIEL EDGAR	Ap. Paterno QUISPE	Ap. Materno VALDERRAMA
No. De carnet 15349673	Expedicion LP	Sexo MASCULINO
Rude 407300842020022.	Fecha de Nacimiento 15/9/2015	

ACTUALIZAR

BUSCAR POR CI O NUM DE ENTREGA.....

Entregas Realizadas
Registro de las entregas hechas a los estudiantes

N° ENT.	NOMBRE DEL ESTUDIANTE	ENTREGADO POR	FECHA	ESTADO	DETALLE	EDITAR
112	QUISPE VALDERRAMA CADMIEL EDGAR	JUANITO	2020-11-15 07:00:36	ENTREGADO	DETALLE	EDITAR

```

<div class="card card-nav-tabs">
<div class="card-header card-header-primary">
<!-- colors: "header-primary", "header-info", "header-success", "header-warning", "header-danger" -->
<div class="nav-tabs-navigation">
<div class="nav-tabs-wrapper d-flex">
<ul class="nav nav-tabs" data-tabs="tabs">
<li class="nav-item">
<a class="nav-link active" href="#profile" data-toggle="tab">
<i class="fas fa-user-graduate"></i>
Estudiante
</a>
</li>
<li class="nav-item">
<a class="nav-link" href="#messages" data-toggle="tab">
<i class="fas fa-user-tie"></i>
tutores
</a>
</li>
<li class="nav-item">
<a class="nav-link" href="#settings" data-toggle="tab">
<i class="fas fa-school"></i>
Unidad educativa
</a>
</li>
</ul>
<ul class="nav nav-tabs ml-auto">
<li class="nav-item">
<button type="button" class="btn nav-link" id="btn-cer"><i class="fas fa-times"></i></button>
</li>
</ul>
</div>
</div>
</div>
<div class="card-body">
<div class="tab-content">
<div class="tab-pane active" id="profile">
</div>
<div class="tab-pane" id="messages">
</div>
<div class="tab-pane" id="settings">
<div id="con-ues">

```

CAPITULO IV
CONTROL DE CALIDAD
Y ESTIMACIÓN DE COSTOS

4. CONTROL DE CALIDAD Y ESTIMACIÓN DE COSTOS

4.1. MÉTRICAS DE CALIDAD DE SOFTWARE

La calidad del software es el desarrollo de software basado en estándares con la funcionalidad y el rendimiento total que satisfaga los requerimientos del cliente, todo desarrollador pone el máximo esfuerzo y dedicación, pero no siempre llega a la perfección en el producto terminado.

4.1.1. La Norma ISO 25000

La familia ISO/IEC 25000 propone un conjunto de normas que tiene por objetivo la creación de un marco de trabajo común para evaluar la calidad del producto software. La familia se presenta como una evolución de las normas ISO/IEC 9126 e ISO/IEC 14598, con la actualización del modelo de calidad de producto, las métricas correspondientes y su proceso de medición basados en los siguientes principios.

4.1.1.1. Descripción de la aplicación

La evaluación se efectuó a un sistema de información que controla las entregas realizadas, de canastas a estudiantes de las unidades educativas, para cumplir un seguimiento estructurado y así contemplar resultados una vez terminado el trabajo de registros de entregas y poder realizar actas basadas en los registros del sistema.

Para el desarrollo del sistema se utilizó HTML, PHP como lenguaje de programación, Mariadb como gestor de base de datos, Apache Server como framework de la aplicación web, Laravel como framework de desarrollo, que proporciona herramientas de dependencia múltiples, como por ejemplo el uso de DOMPDF para la creación de reportes, además conviene mencionar el uso de Ajax, jquery para el desarrollo front-end, y Navicat para el modelado de la base de datos. El sistema está modelado con la arquitectura Cliente-Servidor que permite desarrollar la aplicación en n capas, garantizando el tratamiento por separado de las capas sin afectar al resto de partes. La aplicación en el lado del servidor emplea dos arquitecturas, N capas para la aplicación web y el estilo de arquitectura REST junto con el patrón de diseño modelo, vista controlador, para el intercambio de información con el cliente.

4.1.1.2. Características de calidad en uso a evaluar

El control y seguimiento de entregas, requiere agilidad y veracidad de la información a través de un sistema, que proporcione fácil manejo a los usuarios, lo que implica un registro rápido y oportuno de los datos. En este sentido, al evaluar la calidad del software es necesario tener en cuenta que para los sistemas web se utilizan los mismos modelos del software tradicional, no obstante, hay aspectos más relevantes a medir para este tipo de software como son la Efectividad, Eficiencia y Utilidad del sistema. Conforme a lo mencionado, se estableció la importancia de evaluar la calidad en uso, en el sistema de control de registros para la entrega y seguimiento de entregas realizadas.

4.1.1.3. Especificación de las características, subcaracterísticas y métricas

Para la aplicación del estándar en el sistema, se utilizó el modelo de calidad ISO 25010, donde se detallan las particularidades de la calidad en uso, seleccionando las características Efectividad, Eficiencia y Satisfacción. Asimismo, las subcaracterísticas a medir fueron Efectividad, Eficiencia y Utilidad, respectivamente. Además, se utilizó las métricas especificadas en la ISO 25022 y, finalmente, para la evaluación del software se utilizaron las especificaciones presentadas en la ISO 25040. Dentro de las subcaracterísticas, antes mencionadas, se encuentran particularizadas las métricas: Efectividad de la tarea, Frecuencia de error, Tiempo de la tarea, Eficiencia de la tarea, Productividad económica, Nivel de satisfacción, Uso discrecional de las funciones y Porcentaje de quejas de los usuarios.

Siguiendo el formato de especificación de las métricas detalladas en la ISO/IEC 25022, se presenta, en la Tabla 11, un ejemplo de la métrica Uso discrecional de las funciones, perteneciente a la subcaracterística Utilidad. Referenciando para cada medición el formato presentado, con sus propias variables

Tabla 11 Especificación formal de las métricas de calidad en uso

Sub-característica	Nombre de la métrica	Fase del ciclo de vida de calidad de producto	Propósito	Método de aplicación	Formula	Valor deseado	Tipo de medida	Recursos empleados
Utilidad	Uso discrecional de las funciones	En uso	¿Qué porcentaje de los usuarios optan por utilizar las funciones del sistema?	Observación de manejo	$X=A/B$ Donde $B>0$	$0 \leq x \leq 1$ Cuanto más cercano a 1, mejor	A=contable B=contable	Usuario

Además, en la Tabla 11, se visualiza la fórmula cuya condición $B > 0$, indica que el número de funciones implementadas debe ser mayor a 0. A su vez, que la variable A establece el número de funciones específicas del software que se usa. Luego de efectuar el cálculo A/B y determinando el valor resultante, se establece como el mejor al más cercano a 1. El tipo de medida es cuantitativo para obtener el cálculo correcto, estableciendo a X, A y B recursos contables. Estos datos son posibles a través del manejo de los usuarios en el sistema.

4.1.1.4. Descripción de niveles de puntuación

Las prácticas tomadas como referencia del modelo de evaluación de la ISO/IEC 25040 permitieron instanciar rangos de medición, como se indica en la Tabla 12, adaptados al criterio de las necesidades de solución del sistema. Los valores están comprendidos entre 0 y 10, consolidados en cuatro niveles de puntuación y tres grados de satisfacción.

Tabla 12 Rangos de puntuación para las métricas

Valor de medición	Nivel de puntuación	Grado de satisfacción
7.91 – 10	Cumple con los requisitos	Muy satisfactorio
4.91 – 7.90	Aceptable	Satisfactorio
1.91 – 4.90	Mínimamente aceptable	No satisfactorio
0 – 91	inaceptable	No satisfactorio

Con el valor cuantitativo obtenido de las mediciones se puede establecer la relación cualitativa, como se observa en la Tabla 12, siendo, el rango inaceptable, los valores que no cumplen con el requerimiento mínimo establecido, y el rango objetivo, los valores deseados, otorgando un equilibrio funcional.

4.1.1.5. Ponderación de las características

En la Tabla 13 se presenta la ponderación determinada, constituyendo de orden fundamental a la Satisfacción, siendo la Utilidad primordial al momento de manejar intuitivamente y utilizar las funciones del sistema sin complicaciones, con un 50%. Siguiendo esta línea, se estableció a la Efectividad con un 30%, debido a que se requería que el sistema cumpla con el propósito para el que fue creado, es decir, completando los objetivos de las tareas sin fallas de funcionalidad. Y, finalmente, la Eficiencia, fue de orden opcional con un 20% pero necesaria de evaluar, para configurar que tan eficientes son los usuarios, basado en el tiempo de concluir una tarea.

Tabla 13 Ponderación de características de calidad en uso

Características	Sub-característica	Métrica	Importancia	Ponderación
Efectividad	Adecuación Funcional	Efectividad de la tarea Frecuencia de error	Fundamental	30%
Eficiencia	Eficiencia	Tiempo de la tarea Eficiencia de la tarea Productividad	Opcional	20%
Satisfacción	Usabilidad Mantenibilidad Portabilidad	Nivel de satisfacción Uso discrecional de las funciones Porcentaje de queja de los usuarios Nivel de adaptación para el uso e importación del sistema	Fundamental	50%

4.1.2. Resultados de la aplicación de la matriz de calidad de uso

Para la evaluación de las métricas, se empleó el registro de valoración, Tabla 11. Teniendo en cuenta que, el valor deseado para todas las métricas, corresponde a un valor mayor o igual a cero y menor o igual a 1. Considerando que las mediciones se

culminaron, se pudo corroborar los resultados de Efectividad, Eficiencia y Utilidad, siguiendo el proceso a continuación para cada una de las subcaracterísticas.

4.1.2.1. Adecuación Funcional

En este grupo se conjunta una serie de atributos que permiten calificar si un producto de software maneja en forma adecuada, el conjunto de funciones que satisfagan las necesidades para las cuales fue diseñado.

Para el cálculo funcional se debe determinar cinco característica de dominios de información los valores de información se definen de la siguiente manera.

- **Número de entradas de usuario.** Se cuenta cada entrada de usuario que proporciona diferentes datos orientados a la aplicación. Las entradas se deberían diferenciar de las peticiones, las cuales se cuentan de forma separada.
- **Número de salidas de usuario.** Se cuenta cada salida que proporciona al usuario información orientada a la aplicación. En este contexto la salida se refiere a informes, pantallas, mensajes de error, etc.
- **Número de peticiones de usuario.** Una petición se define como una entrada interactiva que produce la generación de alguna respuesta del software inmediata en forma de salida interactiva. Se cuenta cada petición por separado.
- **Número de archivos.** Se cuenta cada archivo maestro lógico (esto es, un grupo lógico de datos que puede ser una parte de una gran base de datos o un archivo independiente).
- **Número de interfaces externas.** Se cuentan todas las interfaces legibles por la máquina (por ejemplo: archivos de datos de cinta o disco) que se utilizan para transmitir información a otro sistema.

Tabla 14 Factores de ponderación

Parámetros de medición	Cuenta	Factores de ponderación			Valor obtenido
		Simple	Medio	Complejo	
Nro. de entradas de usuario	5	-	4	-	20
Nro. de salidas de usuario	8	-	5	-	40

Nro. de peticiones de usuario	2	-	4	-	8
Nro. de archivos	18	-	10	-	72
Nro. de interfaces externas	0	-	7	-	0
Cuenta total					140

En la siguiente tabla se muestra el factor de ajuste de complejidad basado en las respuestas de las siguientes preguntas evaluadas.

Tabla 15 Ajuste de complejidad

Factores de complejidad	Sin influencia	Inicial	Moderado	Medio	Significativo	Esencial	Fi
	0	1	2	3	4	5	
¿Requiere el sistema copias de seguridad y de recuperación fiables?				X			3
¿Se requiere comunicación de datos?					X		4
¿Existen funciones de procesamiento distribuido?						X	5
¿Es crítico el rendimiento?				X			3
¿Se ejecutará el sistema con un entorno operativo existente y fuertemente utilizado?					X		4
¿Requiere el sistema entrada de datos interactiva?				X			3
Facilidad Operativa					X		4
¿Se actualiza los archivos maestros de forma interactiva?						X	5

¿Son complejas las entradas, las salidas, los archivos o las peticiones?	X	3
Procesamiento interno complejo	X	3
Diseño de código reutilizable	X	4
Facilidad de Instalación	X	3
Facilidad de cambios	X	4
¿Soporta múltiples instalaciones en diferentes sitios?	X	4
Factor ajuste de complejidad		52

La adecuación funcional es la medida a través del punto función (PF), que proporciona una medida objetiva del tamaño de la aplicación web que está basada en la visión del usuario de la aplicación web.

Para calcular el punto función se utiliza la siguiente fórmula:

$$PF = \text{Cuenta Total} * (X + \text{Min} (Y) * \sum Fi)$$

Dónde:

PF: Medida funcional

Cuenta Total: Es la suma de los siguientes datos (Nro. de entradas, Nro. de salidas, Nro. de peticiones, Nro. de archivos, Nro. de interfaces externas).

X: Confiabilidad del proyecto, varía entre 1 a 100%.

Min (Y): Error mínimo aceptable al de la complejidad.

$\sum Fi$: Son los valores de ajuste de complejidad, donde $(1 \leq i \leq 14)$.

Para calcular el PF se usa la siguiente ecuación:

$$PF = \text{cuenta total} * (X + \text{Min} (Y) * \sum Fi)$$

$$PF = \text{Cuenta total} * [0.65 + (0.01 * \sum Fi)]$$

Reemplazando los valores obtenidos en las tablas 11 y 12 se obtiene el siguiente resultado:

$$PF = 140 * [0.65 + (0.01 * 52)]$$

$$PF = 163.8$$

A continuación, calculamos el PF ideal:

$$PF_{ideal} = 140 * [0.65 + (0.01 * 62)]$$

$$PF_{ideal} = 177.8$$

Entonces la adecuación funcional del sistema es:

$$\text{Adecuación Funcional} = (PF/PF_{ideal}) * 100$$

$$\text{Adecuación Funcional} = (163.8/177.8) * 100$$

$$\text{Adecuación Funcional} = 92.13\%$$

Con el resultado obtenido de los cálculos realizados se puede interpretar de qué 92.13% esta funcionalidad esto determinan que el sistema responde de manera óptima a las funcionalidades requeridas por la institución.

4.1.2.2. Eficiencia

Se seleccionó cinco requerimientos funcionales en conjunto con el cliente, ya que generan mayor interacción con el usuario: gestión de registro estudiantiles, gestión en registros por unidades educativas, registro de entregas, obtener reporte con resultados de entregas realizadas bajo características específicas y obtener reportes en actas de entregas en general, los cuales se aplicó las métricas de evaluación. En la Tabla 16 se presenta los valores obtenidos de la métrica Eficacia de la tarea, para cada uno de los requerimientos funcionales. En este sentido, se empleó una escala de medición de cero a diez y se designó al “más cercano a cero”, como el mejor, certificando un valor de 10 al requerimiento.

Tabla 16 Datos de la métrica Efectividad de la tarea

Requerimiento	Objetivos completados A	Objetivos planeados B	Efectividad de la tarea	Valor de medición
Gestión de registros estudiantiles	5	5	10	10
Gestión de registros por unidad educativa	3	3	1	8
Registro de entregas	4	4	1	9.50
Reportes de resultados de entregas hechas	3	5	0.6	8.50
Reportes y actas entregas	3	3	1	9.50
Promedio				9.1 = 91%

Para determinar los demás resultados, se realizó una regla de tres inversa, ya que, al ser el valor deseado más cercano a cero, si el valor de Eficiencia de la tarea aumenta, entonces el valor de medición disminuye. Una vez obtenidos los resultados, se verificó mediante la Tabla 12, que se cumplió con los requisitos, además de ser muy satisfactorio. El valor promedio se obtuvo de la sumatoria de los valores de medición *10 en porcentaje que da 91%, correspondientes a una subcaracterística, dividido para el número de métricas que posee.

Para la gestión de registros estudiantiles, gestión de registros por unidad educativa y registro de entregas se completaron las siguientes tareas: registró y modificación de estudiantes con tutores, registros de observaciones encontradas antes de la entrega, registró con verificación de cursos por unidad educativa

Para obtener el reporte de resultados en entregas hechas, se completaron las siguientes tareas: buscar los registros de las entregas realizadas y comparar con el total

de registros obtenidos, especificar resultados por filtros en orden de jerarquía para las unidades educativas

Para obtener reporte y actas de entregas, se completaron las siguientes tareas: buscar los registros de las entregas de acuerdo a las especificaciones de búsqueda y de acuerdo a la unidad educativa basado en los registrados de los cursos de dicha unidad educativa.

4.1.2.3. Usabilidad

La usabilidad consiste de un conjunto de atributos que permiten evaluar el esfuerzo necesario que deberá invertir el usuario para utilizar el sistema, dado también que se puede referir a la usabilidad como la capacidad del software de ser entendido, aprendido, y usado de forma fácil y atractiva.

Para determinar el porcentaje del uso del sistema se contabiliza los resultados que se muestran en la tabla 17, las preguntas realizadas para el uso del sistema.

Tabla 17 Preguntas sobre el uso del sistema

N°	Preguntas	Respuesta		% de si
		Si	No	
1	¿No es complicado usar el sistema?	8	1	88.8
2	¿Las pantallas que vista fueron de su agrado	7	2	77.7
3	¿Los datos obtenidos que vio fueron fáciles de comprender?	8	1	88.8
4	¿El sistema responde rápido a sus solicitudes?	6	3	66.6
5	¿El sistema facilita el trabajo para generar datos?	9	0	100
6	¿El sistema reduce el tiempo de trabajo en generar informes?	9	0	100
7	¿Es fácil navegar por las distintas opciones?	8	1	88.8
8	¿La generación de datos que se realizan no es complicada?	7	2	77.7
9	¿El sistema no presento errores?	8	1	88.8
Promedio				86.3%

En consecuencia, se pudo demostrar la calidad en uso del sistema sumando el valor ponderado de las subcaracterísticas, presentando un total de 86.3% de calidad en uso.

De lo antes mencionado, se puede definir la deficiencia de calidad en uso del sistema, tomando en cuenta los valores faltantes de calidad en uso de algunas subcaracterísticas y ponderación de respuestas de la tabla 17 además en el orden de requerimientos como se refleja en la tabla 16.

4.1.2.4. Confiabilidad

La confiabilidad del sistema se define como la probabilidad de operación libre de fallos de un programa de computadora en un entorno determinado y durante un tiempo específico. Para calcular confiabilidad del sistema se toma en cuenta el periodo de tiempo en el cual se ejecuta y se obtiene muestras

La sencilla medida de fiabilidad es el tiempo medio entre fallos, y está dado por:

$$\mathbf{TMEF = TMDF + TMDR}$$

Donde:

TMEF: es el tiempo medio entre fallos.

TMDF: es el tiempo medio de fallos.

TMDR: es el tiempo medio de reparación.

Reemplazando datos tenemos:

$$\mathbf{TMEF = 8 \text{ horas de trabajo} + 0.60 \text{ horas de reparación}}$$

$$\mathbf{TMEF = 8.60 \text{ horas.}}$$

Además de una medida de fiabilidad, debemos calcular la medida de disponibilidad del software, que será de la probabilidad de que un programa funcione de acuerdo a los requisitos en un momento dado, y se define como:

$$\mathbf{\text{Confiabilidad} = [TMDF / TMEF] \times 100\%}$$

Reemplazando datos tenemos:

$$\text{Confiabilidad} = [8 / 8.60] \times 100\%$$

$$\text{Confiabilidad} = [0.93] \times 100\%$$

$$\text{Confiabilidad} = 93\%$$

La confiabilidad del sistema es del 93% en un periodo de prueba con un tiempo de 8 horas de trabajo normal y funcionamiento del sistema

4.1.2.5. Mantenibilidad

Es el esfuerzo necesario para diagnosticar las deficiencias o causas de fallas o para identificar las partes que deberán ser modificadas, la calidad que tiene el software para ser modificado incluyendo correcciones o mejoras del software a cambios en el entorno y especificaciones de requerimientos funcionales, para poder medir la calidad del mantenimiento del sistema utilizaremos el índice de madurez del software (IMS), que indica la estabilidad de un producto de software.

El índice de madurez del software se calcula con la siguiente formula:

$$IMS = [Mt - (Fa + Fb + Fc)]/t$$

Donde:

Mt: Número de módulos en la versión actual

Fa: Número de módulos en la versión actual que se han cambiado

Fb: Número de módulos en la versión actual que se han añadido

Fc: Número de módulos en la versión anterior que se han borrado en la versión actual.

Recopilando la información requerida por la formula se obtuvo la información que se muestra en la siguiente tabla:

Tabla 18 Información requerida por el IMS

Información	Valor
Mt	11
Fa	0
Fb	0
Fc	0

Ahora calculamos el IMS, usando los valores obtenidos:

$$IMS = [11 - (0 + 0 + 0)]/11$$

$$IMS = 11/11 = 1 * 100 = 100\%$$

Con la interpretación a este resultado establece un 100% de madures, lo que indica que no requiere de mantenimientos inmediatamente.

4.1.2.6. Portabilidad

Se evalúa la oportunidad para adaptar el software a diferentes ambientes sin necesidad de aplicarle modificaciones, es decir que el software se puede trasladar de un entorno a otro.

Para poder medir la portabilidad del sistema usaremos la siguiente formula que indica el grado de portabilidad que tiene un software.

$$GP = 1 - (ET/ER)$$

Donde:

ET: Es la medida de los recursos necesarios para llevar el sistema a otro entorno.

ER: Es la medida de los recursos necesarios para crear el sistema en el entorno residente.

Si **GP** > 0, la portabilidad es más rentable que el re-desarrollo

Si **GP** = 1, la portabilidad es perfecta

Si **GP** < 0, el re-desarrollo es más rentable que la portabilidad.

Entonces la factibilidad estimada para trasportar el sistema es de 1/3 días a otro entorno y la implementación del mismo en otro entorno es de 5 días.

$$GP = 1 - ((1/3)/ 5) = 1 - 0.06 = 0.94 * 100 = 94\%$$

Por lo que se concluye que el sistema tiene un grado de portabilidad del 94%.

4.1.2.7. Calidad global

La calidad global del sistema es el total, de todos los cálculos realizados según lo considerado basados en la propuesta del estándar de calidad ISO/IEC 25010, los cálculos realizados se visualizan a continuación.

Tabla 19 Calidad global del sistema

Atributos	Valor (%)
Adecuación funcional	92.13
Eficiencia	91
Usabilidad	86.3
Confiabilidad	93
Mantenibilidad	100
Portabilidad	94
Calidad global	92.7

Con el resultado obtenido se llega a la conclusión de que el 92.7% del sistema es de calidad.

4.2. EVALUACIÓN DE COSTOS

La evaluación del costo determina la cantidad de los recursos necesarios en términos de dinero, esfuerzo, capacidad, conocimientos y tiempo. Se evaluara los costos y beneficios para el sistema en base al modelo de costos COCOMO II.

4.2.1. Modelo COCOMO II

El Modelo COCOMO II, es un modelo de estimación que se encuentra en la jerarquía de modelos de estimación de software, permite realizar estimaciones en función del tamaño del software, y de un conjunto de factores de costo y escala, COCOMO II consta con tres modelos de estimación, los mismos se representan en 3 ecuaciones:

$$E = a(KLDC)^b ; \text{Personas} - \text{mes}$$

$$D = c(E)^d ; \text{Meses}$$

$$P = \frac{E}{D} ; \text{Personas}$$

Dónde:

E: Esfuerzo requerido por el proyecto expresado en persona-mes.

D: Tiempo requerido por el proyecto expresado en meses.

P: Número de personas requeridas para el proyecto.

A, B, C y D: Constantes con valores definidos según cada sub-modelo.

KLDC: Cantidad de líneas de código distribuidas en miles

A la vez cada modelo se subdivide en tres modos:

Modo orgánico: Es un pequeño grupo de programadores experimentados desarrollando proyectos de software en un entorno familiar.

Modo semilibre: Corresponde a un esquema intermedio entre el modo orgánico y el rígido, el grupo de desarrollo puede incluir una mezcla de personas experimentadas y no experimentadas.

Modo rígido: El proyecto tiene fuertes restricciones, que pueden estar relacionadas con la funcionalidad y/o pueden ser técnicas.

A continuación, se describe las constantes de acuerdo a los modos mencionados anteriormente.

Tabla 20 Constantes a b c d COCOMO II

Modo	A	B	C	D
Órgano	2.4	1.05	2.5	0.38
Semilibre	3.0	1.12	2.5	0.35
Rígido	3.6	2.20	2.5	0.32

4.2.2. Costos del sistema

El costo del sistema se lo plantea en dos partes:

4.2.2.1. Costos de desarrollo

Para la estimación de costos del sistema ha sido desarrollado bajo las KLDC (Kilo-Líneas de Código) las que detallamos lo siguiente:

Para este proyecto basado en la funcionalidad obtenemos como base 163.8 de PF que nos servirá para calcular el tamaño de la programación PHP.

En la tabla se muestra la relación para convertir el valor de PF a KLDC

Tabla 21 Muestra de relación para convertir el valor de PF a KLDC

Lector	Nivel	Factor LDC / PF
C	2.5	128
Java	6	53
Visual Basic	7	46
ASP	9	36
PHP	11	29
Visual C++	9.5	34
SQL	12	12

Entonces, realizando los cálculos y escogiendo el valor del lenguaje de programación PHP de la tabla 4.9, tenemos:

$$\text{LCD} = \text{PF} * \text{Factor LCD}$$

$$\text{LCD} = 163.8 * 29$$

$$\text{LCD} = 4750.2$$

$$\text{KLCD} = 4750.2 / 1000$$

$$\text{KLCD} = 4.75$$

Ahora, para hallar el esfuerzo reemplazamos los valores obtenidos hasta ahora:

$$E = 2.4 * (4.75)^{1.05} = 12.32$$

$$D = 2.4 * (4.75)^{0.38} = 4.34$$

Para el cálculo del número de programadores para el desarrollo de software:

$$E = \frac{12.32}{4.34} = 3.83 \approx 4$$

Estimando que el salario medio de un programador es de Bs. 4000, esta cifra será tomada en cuenta para la siguiente estimación:

Costo del software desarrollado = Numero de programadores * Salario de un programador.

Costo del Software Desarrollado por Persona = 4 * 4000 = 16000 Bs.

Costo total del Software Desarrollado = 16000 * 2 = 32000 Bs.

Lo que significa que el costo del sistema desarrollado por los 2 meses es 32000, equivalente en dólares a 4604.31\$.

4.2.2.2. Costos de elaboración del proyecto

Se calcula los costos de inversión de los recursos que se usaron para la elaboración del sistema.

Tabla 22 Costos de recursos empleados para la elaboración del sistema

Recursos	Costo Bs
Materia de escritorio	300
Investigación del proyecto	250
Internet	300
Otros	180
Total	1030

Fuente: elaboración propia

Por tanto, el costo de la elaboración del proyecto es de Bs. 1030

4.2.2.3. Costos total del sistema

El costo total del sistema se obtiene de la sumatoria del costo de desarrollo y el costo de elaboración del proyecto, se puede observar los resultados en la siguiente tabla, todos los costos están expresados en moneda nacional bs.

Tabla 23 Costos total del sistema

Detalle	Costo Bs
Costo de desarrollo	32000
Costo de elaboración del proyecto	1030
Costo Total	33030

Entonces, el costo total del sistema es de Bs. 33030 equivalente en dólares 4752.51\$

CAPITULO V
PRUEBAS Y SEGURIDAD
DE SOFTWARE

5. PRUEBAS Y SEGURIDAD DE SOFTWARE

5.1. PRUEBAS DE SOFTWARE

Para el presente proyecto nos basaremos en las dinámicas que proponen las pruebas de caja blanca y pruebas de caja negra, así mismo se contemplan pruebas de estrés que ayudaran a verificar la saturación del sistema.

5.1.1. Pruebas de caja blanca

Permite obtener una medida de complejidad de un diseño procedimental, y utilizar esta medida para definir una serie de caminos básicos de ejecución, garantizando que cada camino se ejecute al menos una vez, ver figura 37

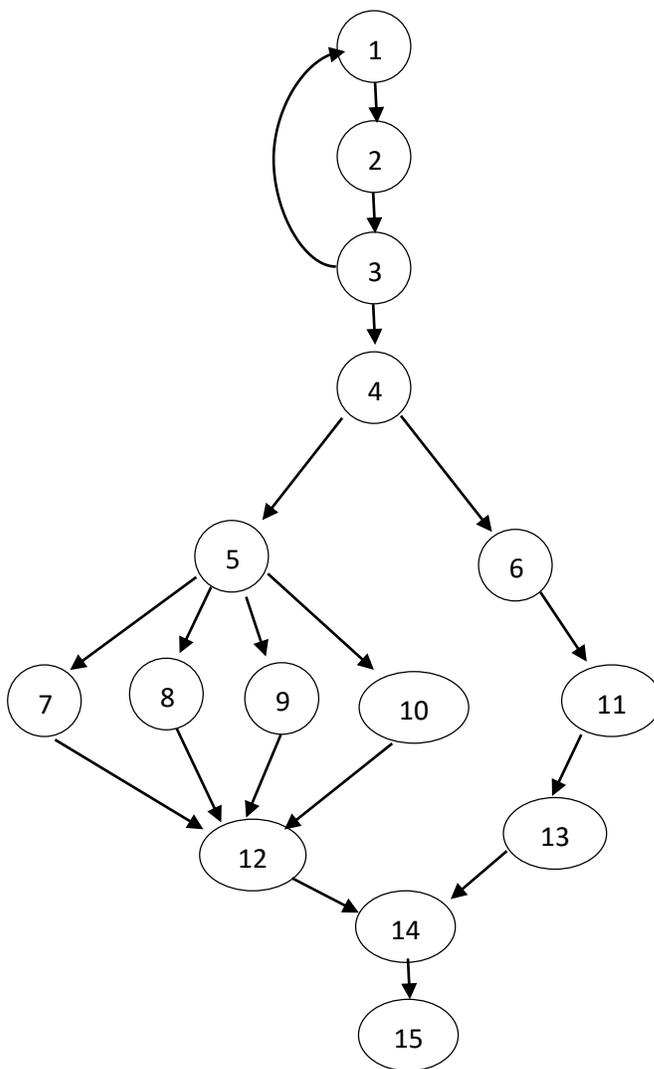


Figura 37 Grafo del sistema

Donde:

1. Inicio del sistema
2. Usuario y contraseña
3. Validar usuario y contraseña
4. Menú principal
5. Menú administrador
6. Realizar entregas
7. Gestión de registro
8. Control de entregas
9. Gestión de Actas
10. Gestión de Reportes
11. Gestión de documentación
12. Fin ciclo administración
13. Fin ciclo de entregas
14. Fin ciclo sistema
15. Fin sistema

Examinamos el grafo creado a partir de las características del sistema, se procede a determinar la complejidad ciclomática del grafo mediante:

$$(G) = A - N + 2$$

Donde:

A = número de aristas

N = número de nodos

$$(G) = 19 - 15 + 2 = 6$$

Por tanto, la complejidad ciclomatica es: $V(G) = 6$, esto significa que existe 6 caminos independientes.

5.1.2. Pruebas de caja negra

La prueba de caja negra consiste en probar cada una de las funciones del sistema que fueron descritas anteriormente. Con este tipo de prueba se debe buscar que las funciones sean operativas, además se debe agotar al sistema de tal manera buscar la mayor cantidad de errores estas pruebas son pruebas sobre la interfaz del software. (Peña, 2016)

A continuación se muestra la prueba de autenticación del usuario, si el usuario accede con las credenciales correctas del sistema ingresara a los módulos internos, en otro caso será denegado el acceso

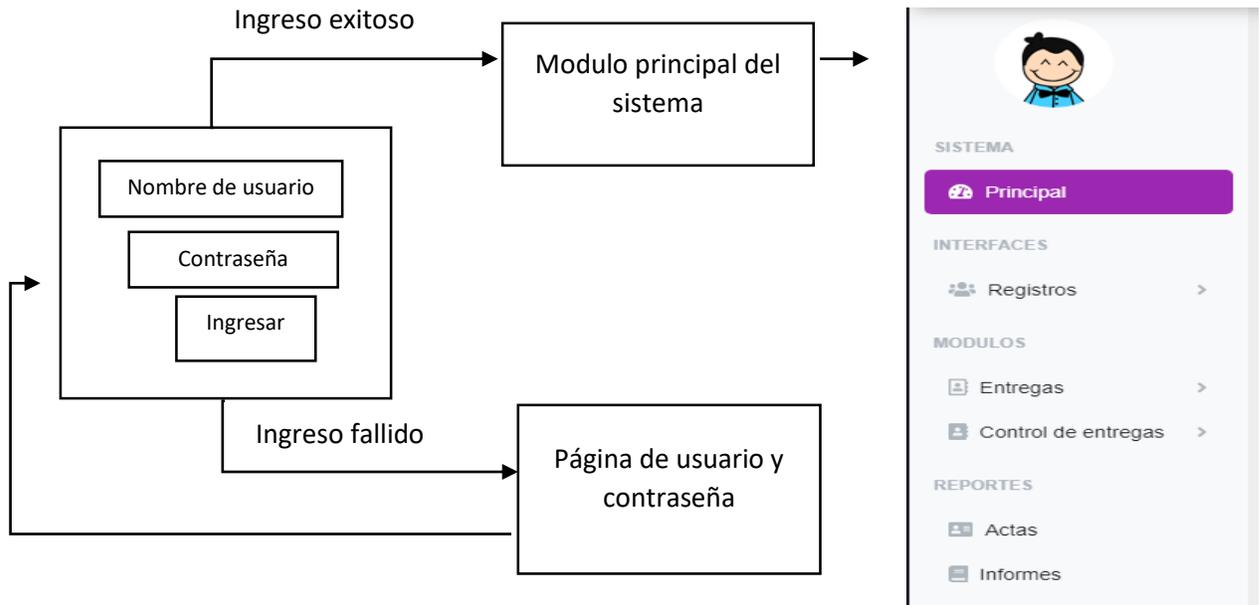


Figura 38 Prueba de autenticación de usuario

Pruebas de validación del formulario para el ingreso de datos, si un formulario no es llenado correctamente muestra una alerta y en caso contrario el ingreso con los datos será exitoso

La imagen muestra una captura de pantalla de un formulario de inicio de sesión con el título 'INTRODUZCA SUS DATOS'. El formulario tiene dos campos de entrada: 'Nombre de usuario' y 'Contraseña'. El campo 'Nombre de usuario' está vacío y tiene una alerta de validación que dice 'Completa este campo'. El campo 'Contraseña' tiene un ícono de ojo para alternar la visibilidad. Debajo de los campos hay un checkbox 'Recuérdame?' y un botón 'INICIAR SESION'.

Figura 39 Pruebas de validación de formulario

Tabla 24 Pruebas de caja negra

Entrada	
Aprobación	Registro de usuarios Registro de estudiantes Registro de tutores Registro de unidades educativas
Seguridad	Una vez autenticado el usuario
Funciones	
Software	Menú de administrador Menú para supervisor Menú para el responsable
Salida	
Resultado	Reportes de entregas Reportes de resultados Reportes específicos Generar actas
Seguridad	Son resultados que verán los usuarios que accedieron al sistema

5.1.3. Pruebas de estrés

El objetivo de las pruebas de estrés es saturar el la aplicación hasta un punto de quiebre donde aparezcan defectos potencialmente peligrosos, no para decir que el sistema no funciona, lo que se intenta es mejorar la aplicación reduciendo riesgos que puedan dar origen a una caída del sistema

Para esta prueba se utilizó un servidor interno dentro la institución con características de media gama respecto a los servidores de alta gama, además cabe mencionar que no se realizaron muchas variaciones en el sistema operativo, ya que la institución así lo dispuso. Dentro de todo esto se obtuvieron resultados que van de acuerdo a los promedios de rendimiento que exigen al sistema

- Usuarios establecidos: 20
- Tiempo de prueba: 5 min
- Solicitudes realizadas: 987
- Fallas HTTP: 0

- Pico de solicitud: 8 (Solicitudes por segundo)
- Tiempo de respuesta media: 87ms.

5.2. SEGURIDAD

5.2.1. Norma ISO 27000

El sistema debe de contar con las medidas de seguridad como la encriptación y uso de variables de sesión. La encriptación nos ayuda a encriptar datos como ser contraseñas, así también las variables de sesión ayudan en el acceso restringido a un sistema, es decir que el usuario debe de tener usuario y contraseña para la verificación existente del usuario se crea una variable de sesión y finaliza las tareas correspondientes al final se destruye las variables de sesión creadas por el sistema.

5.2.1.1. Encriptación

Para la encriptación se utilizará el algoritmo HASH el cual ya está incluido en el framework Laravel, HASH, el cual nos permitirá encriptar la contraseña y la verificación se realiza comparando encriptaciones con la finalidad de proteger las contraseñas con la salida de HASH de 160 bits (20 bytes) como se muestra en la figura 40.

```
protected function create(array $data)
{
    return User::create([
        'name' => $data['name'],
        'username' => $data['username'],
        'email' => $data['email'],
        'password' => Hash::make($data['password']),
    ]);
}
```

Figura 40 Encriptado

5.2.1.2. Autenticación y autorización

La autenticación y autorización van unidas por los accesos de usuario a distintos niveles de información. Este proceso realiza la autenticación de usuario tanto como los encargados o administradores del sistema, además permite comprobar la compatibilidad y el origen ya sea de un programa, una función, una secuencia o una persona.

5.2.1.3. Copias de seguridad

Los respaldos del sistema se basan en almacenar respaldos de las BD de manera lógica y en caso necesario de carácter físico para la parte administrativa esto garantizara que se ejecuten respaldos para la copia de seguridad.

5.2.1.4. Auditoria

La auditoría forma parte de una evaluación misma en la seguridad del sistema y de la base de datos, esto proporcionara una garantía en casos de eventos que se desee indagar para tener un respaldo exacto de información manejada dentro el sistema.

Para una mejor función de auditoria también se realizará registro de logues para los usuarios esto garantiza una ayuda a la indagación del sistema

CAPITULO VI
CONCLUSIONES Y
RECOMENDACIONES

6. CONCLUSIONES Y RECOMENDACIONES

6.1. CONCLUSIONES

Se concluye con los objetivos planteados en el presente proyecto, implementando un Sistema de información para el control y seguimiento de entrega de canastas estudiantiles a las unidades educativas de la ciudad de El Alto, que coadyuvara en el reflejo de resultados tanto informativos, como estadísticos. Aplicando con éxito las normas de calidad y las herramientas de programación para que tenga alta usabilidad, funcionalidad y eficacia

Una vez concluido el presente proyecto, se llegaron a las siguientes conclusiones:

- Se sistematizo en registros de una base de datos, la información recolectada en datos, que brindaron las diferentes unidades educativas en referencia a los estudiantes contemplados para la entrega además del tutor que acompañe.
- Se realizó el seguimiento de entregas con respaldo de planillas y actas que justifican el trabajo realizado.
- Se realizó la implementación de módulos de información estadística según los filtros de datos que se desea consultar
- Se realizó la creación de actas de entregas ya sea de carácter personal o de agrupación por curso

Se Implementó módulos de seguridad para la auditoria de datos que respalden el proceso de uso del sistema.

6.2. RECOMENDACIONES

Hoy en día es recomendable utilizar la metodología orientada a la WEB en los procesos de investigación, las recomendaciones que se deben considerar para el sistema son:

- En primer lugar, capacitar a los usuarios administradores que ingresen al sistema para un buen manejo y uso del mismo.
- Que los usuarios mantengan su contraseña y nombre de usuario como privacidad y que de ser necesario puedan realizar cambio de contraseña cada cierto tiempo.

- Ampliar el sistema si así lo requiere la Institución, ya que este sistema implantado es una versión 1.0.
- Realizar siempre copias de seguridad periódicamente en la base de datos y obtener un respaldo con el proceso que mejor sea conveniente.
- Realizar evaluaciones periódicas del sistema y de la información para determinar las nuevas necesidades apuntarlas en una nueva hoja.

BIBLIOGRAFÍA

- Amo, F. A. (2005). *Introducción a la Ingeniería del software*. Madrid: Delta Publicaciones.
- Andino, M. E. (2001). *Correspondencia y Documentación Comercial*. Tegucigalpa: Guaymuras.
- Barcelo, M. (2008). *Una Historia de la Informática*. UOC.
- Chutro, P. P. (2017). *Logic Tracker*. España, madrid: ediTcc
- Ciceri, M. J. (2018). *Introducción a Laravel*. Buenos Aires: Six Ediciones.
- Dimes, T. (2016). *PHP*. Babelcube.
- gamea*. (Septiembre de 2020). Obtenido de www.elalto.gob.bo
- Giner, C. A. (2004). *Tecnología de sistema de Control*. Barcelona: UPC.
- Gonzales, L. M. (2003). *Cómo implantar un sistema de control de gestión en la práctica*. Barcelona: Gestion 2000 S.A.
- Guarachi, A. R. (2014). SISTEMA WEB DE REGISTRO, SEGUIMIENTO Y CONTROL DE CORRESPONDENCIA BASADO EN BPM. *Proyecto de Grado*. La Paz, Bolivia: Universidad Mayor de San Andres.
- ISO25000. (septiembre de 2020). *ISO 25000 Calidad de software y datos*. Manual de implementación, abstracción y parámetros condicionales.
- ISO27000.ES. (2005). *Seguridad en el seguimiento de software*. Manual de parámetros en seguridad familia 27000
- IUTEB. (Septiembre de 2020). *Slideshare*. Libro digitales basados en metodologías del desarrollo de software y sus modelos
- Ludwin. (2016). *Modelos de la metodología UWE*. Manual de seguimiento en modelados UWE, bajo estándares UML.
- Mariaca, D. L. (2016). *Teoría de sistemas*. La Paz: Ediciones Huara S.A.
- Mashiel, C. L. (2017). Proyecto de Grado. *CONTROL DE PRODUCCIÓN, PEDIDOS, ENTREGAS E INVENTARIOS CASO: PHOVEDA*. La Paz, Bolivia: UMSA.
- Patricio, R. G. (2017). Proyecto de Grado. *SISTEMA WEB DE CONTROL Y SEGUIMIENTO DE OBRAS MUNICIPALES PARA LA SUBALCALDÍA DE OVEJUNO D-I MUNICIPIO DE PALCA*. La Paz, Bolivia: UMSA.
- PHP, G. (2020). *Manual de PHP*. Obtenido de <https://www.php.net/manual/es/intro-what-is.php>

Prescot, P. (2015). *HTML 5*. Babelcube.

Ramiro, A. Q. (2018). *gestion escolar de la ciuda de El Alto*. El Alto.

Sanz, E. (febrero de 2016). *Consultores Documentales*. Obtenido de <https://sorprendemos.com/consultoresdocumentales/?p=507#:~:text=El%20seguimiento%20consiste%20b%C3%A1sicamente%20en,lo%20planificado%20y%20esperado%20ocurra>.

Sommerville, I. (2005). *Ingenieria de Software*. Madrid: Pearson Educacion.

Urtiaga, G. (2018). *Administrar MySQL y MariaDB: Aprende a administrar MySQL y MariaDB fácilmente*. AprendeIT.

WEB GRAFÍA

Alvarez, M. A. (Julio de 2019). *Desarrollo web*. Obtenido de <https://desarrolloweb.com/articulos/que-es-mvc.html>

B., G. (2018). *Hostinger Tutoriales*. Obtenido de <https://www.hostinger.es/tutoriales/que-es-apache/>

Cambio Digital. (Marzo de 2020). Obtenido de <https://cambiodigital-ol.com/2020/03/la-triada-cia-definicion-componentes-y-ejemplos/>

Craber. (Junio de 2014). *Slideshare*. Obtenido de <https://es.slideshare.net/tfluxa/implantacin-de-un-sistema-de-gestin-de-seguridad-de-la-informacin-una-visin-prctica>

Definiciones. (septiembre de 2020). *Definiciones.de*. Obtenido de <https://definicion.de/optimizacion/>

Digital learning. (Marzo de 2012). Obtenido de <https://www.digitallearning.es/blog/que-es-ajax/>

Flores, A. (2018). *Portales*. Obtenido de https://sites.google.com/a/istpargentina.edu.pe/andres_flores/1/t

Gustavo, B. (13 de Mayo de 2019). *Hostinger Tutoriales*. Obtenido de <https://www.hostinger.es/tutoriales/que-es-ajax/>

Informatica. (2020). Obtenido de <https://sites.google.com/site/informaticamcprats/iso-9126>

Interpolados. (Enero de 2017). Obtenido de <https://interpolados.wordpress.com/2017/01/21/norma-isoiec-9126-calidad-del-producto-de-software/>

Jimenez, J. B. (Junio de 2019). *Open Webinars*. Obtenido de <https://openwebinars.net/blog/tipos-de-pruebas-de-software/>

Julian, P. P. (2009). *Definicion.de*. Obtenido de <https://definicion.de/datos/>

Junta de andalucia. (octubre de 2015). *Marco de desarrollo de la Junta de andalucia*. Obtenido de <http://www.juntadeandalucia.es/servicios/madeja/contenido/procedimiento/28>

Morales, A. (2016). *TodaMateria*. Obtenido de <https://www.todamateria.com/informacion/>

PremiunSoft. (2017). *Manual de Navicat*. Obtenido de https://www.navicat.com/manual/online_manual_new/en/navicat/win_manual/

S.N. (2016). Obtenido de <https://sites.google.com/site/teoriageneraldesistemasuco/2-sistemas-y-subsistemas/subsistemas>

Significados. (2017). Obtenido de <https://www.significados.com/informacion/>

UWE. (16 de Agosto de 2016). *Uml-Based Web Engineering*. Obtenido de <https://uwe.pst.ifi.lmu.de/teachingTutorialSpanish.html>

WWW02: Uwe s.f. *UWE-UML Caracteristicas*, recuperado en Mayo del 2019 de: <http://uwe.pst.ifi.lmu.de/aboutUwe.html>

WWW03: Uwe, 2013 *Fases y etapas dela metodología uwe*, recuperado en Mayo del 2019 de: <http://evangellyscarolinacabellorodriguez.blogspot.com/2013/10/fases-o-etapas-de-la-metodologia-uwe.html>

WWW04: Pid-practicum, 2009. *Metodo de observación*, recuperado en Mayo del 2019 de: https://www.ugr.es/~rescate/practicum/el_m_todo_de_observaci_n.html

WWW05: M. Galan, 2009. *Metodo de entrevistas*, recuperado en Mayo del 2019 de: <http://manuelgalan.blogspot.com/2009/05/la-entrevista-en-investigacion.html>

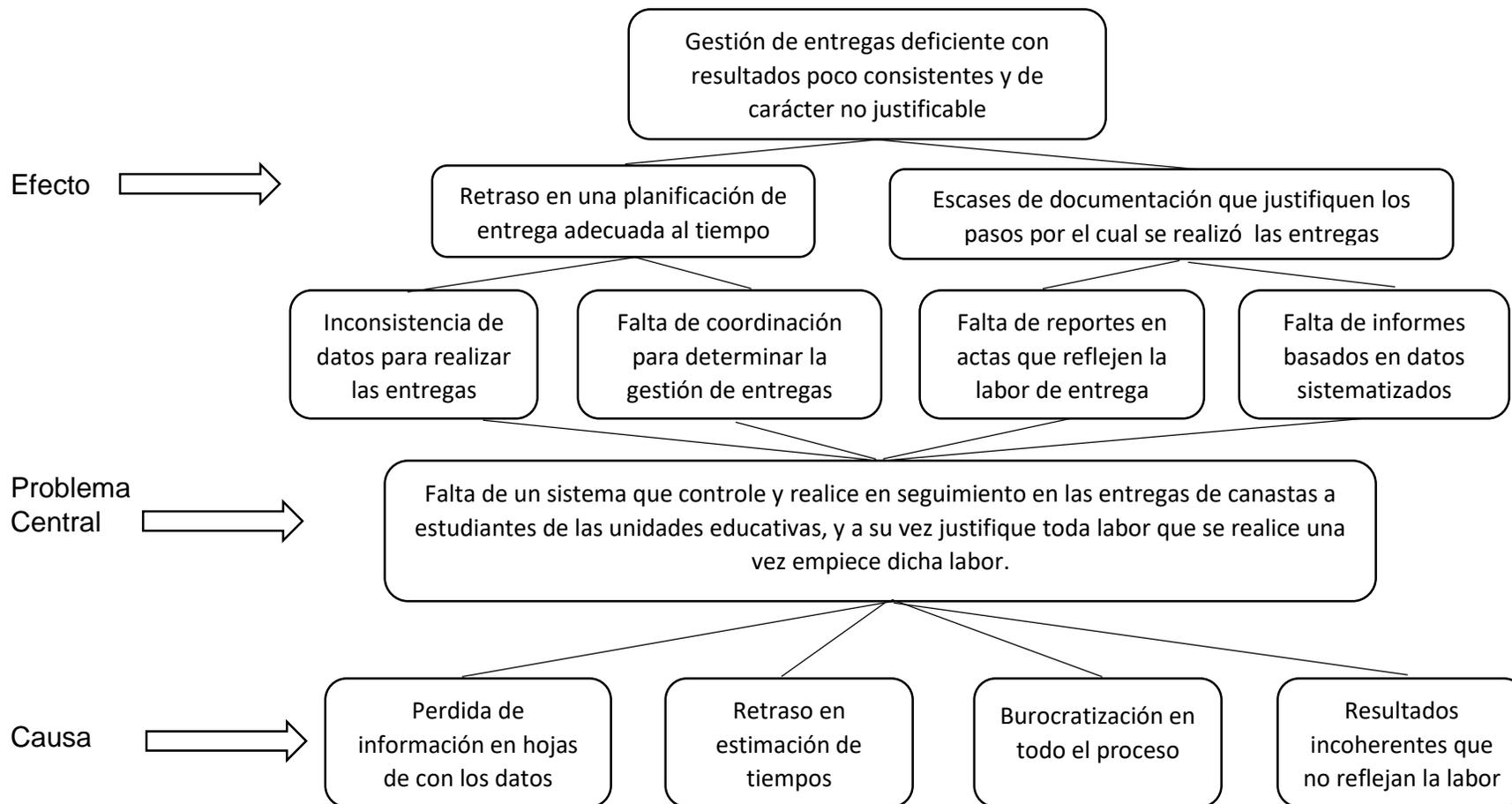
WWW06: Wikioedia, 2019. *Metodo de cuestionario*, recuperado en Mayo del 2019 de: <https://es.wikipedia.org/wiki/Cuestionario>

WWW07: EcuRed, s.f. *Modelo Cocomo II*, recuoerado en Mayo del 2019 de: https://www.ecured.cu/COCOMO_II

WWW08. (10 de septiembre de 2020). *INFORMATICA*. Obtenido de <https://sites.google.com/site/informaticamcprats/iso-9126>

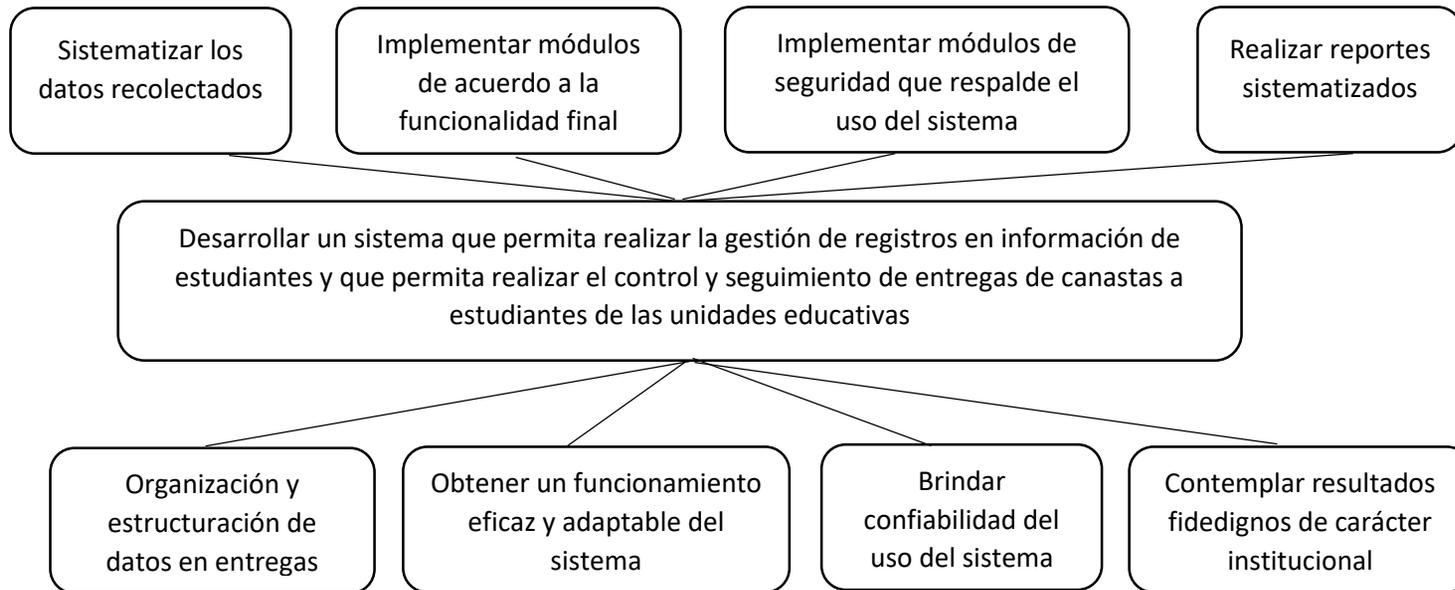
ANEXOS

ÁRBOL DE PROBLEMAS



ANEXO A

ÁRBOL DE OBJETIVOS



Calibri 11
Ajustar texto
Porcentaje
Formato condicional
Dar formato como tabla
Estilos de celda
Insertar Eliminar Formato
Autosuma
Rellenar
Borrar
Ordenar y filtrar
Buscar y seleccionar

AC613 60%

DATOS DE LA MADRE O TUTOR							GESTION	DATOS DEL ESTUDIANTE										DISTRITO	UNIDAD EDUCATIVA Y SIE	% ASISTENCIA
PRIMER APELLIDO	SEGUNDO APELLIDO	... TERCER APELLIDO	NOMBRES	CI DE LA MADRE O TUTOR	EXPEDICION DE LA CEDULA DE IDENTIDAD	TIPO DE DOCUMENTO		APELLIDO PATERNO DEL ESTUDIANTE	APELLIDO MATERNO DEL ESTUDIANTE	NOMBRES DEL ESTUDIANTE	NUMERO CEDULA DE IDENTIDAD DEL ESTUDIANTE BENEFICIARIO	EXPEDICION DE LA CEDULA DE IDENTIDAD	SEXO	NUMERO RUDE DEL ESTUDIANTE BENEFICIARIO	FECHA DE NACIMIENTO DEL BENEFICIARIO	NIVEL	CURSO Y PARALELO			
OUSI	ROJAS		LIDIA	6050074	LAPA2	CARNET DE IDENTIDAD	2020	ADUIRI	OUSI	EZEQUIEL DILAN	15644994	LAPA2	VARON	407200012020028	10/04/2016	INICIAL	1A	EL ALTO-2	12 DE OCTUBRE 40720001	100%
CASTRO	SAAIEDRA		ROXIANA AMANDA	9247306	LAPA2	CARNET DE IDENTIDAD	2020	APAZA	CASTRO	HUBERT LEONEL			VARON	407200012020002	21/12/2015	INICIAL	1A	EL ALTO-2	12 DE OCTUBRE 40720001	100%
CRUZ	TICONA		DORIS	6752236	LAPA2	CARNET DE IDENTIDAD	2020	APAZA	CRUZ	MIROSMAR	15294592	LAPA2	VARON	407200012020004	02/02/2016	INICIAL	1A	EL ALTO-2	12 DE OCTUBRE 40720001	100%
CAZAS	QUISPE		CARLA ALISON	8336166	LAPA2	CARNET DE IDENTIDAD	2020	ASISTIRI	CAZAS	REYNEL VEICKOOB	16301743	LAPA2	VARON	407200012020015	10/09/2015	INICIAL	1A	EL ALTO-2	12 DE OCTUBRE 40720001	100%
CALLA	ALIAGA		MARUJA	6871909	LAPA2	CARNET DE IDENTIDAD	2020	CARCILAZO	CALLA	YIANI SHEYLA			MUJER	407200012020001	08/10/2015	INICIAL	1A	EL ALTO-2	12 DE OCTUBRE 40720001	100%
TICONA	TANCARA		OLGA	6066259	LAPA2	CARNET DE IDENTIDAD	2020	CASAS	TICONA	ELIANA ARACELY	14378873	LAPA2	MUJER	407200012020019	20/11/2015	INICIAL	1A	EL ALTO-2	12 DE OCTUBRE 40720001	100%
QUISPE	BOLIVAR		XIMENA SEGUNDINA	10507979	LAPA2	CARNET DE IDENTIDAD	2020	CASTILLO	QUISPE	NICOLAS MATTHEW	16331776	LAPA2	VARON	407200012020005	20/12/2015	INICIAL	1A	EL ALTO-2	12 DE OCTUBRE 40720001	100%

MODELO PARA RECOLECCIÓN DE DATOS

ANEXO C

ANEXO D
MANUAL DE USUARIO

MANUAL DE USUARIO

Sistema de información para el control y seguimiento de entregas



**DIRECCIÓN DE EDUCACIÓN –
GAMEA**

Autor: Jose Jossue Hilari Vega

La paz – Bolivia

2020

CONTENIDO

1. INTRODUCCIÓN.....	1
2. MODULO ADMINISTRADOR.....	1
3. MODULO DE SUPERVISOR	1
4. MODULO TEC. RESPONSABLE	1
5. FUNCIONAMIENTO DEL SISTEMA	2
5.1. Página principal	2
5.2. Módulo de Inicio de sesión.....	3
5.3. Módulo de Inicial de ingreso al sistema	3
5.4. Menú principal.....	4
5.5. Contenidos principales: Registros de estudiantes	5
5.6. Contenido principal: Registro de entregas	6
5.7. Contenido principal: Informe de resultados.....	7

1. INTRODUCCIÓN

Para facilitar el control, administración en registros, para la entrega de canastas estudiantiles, se desarrolló el Sistema llamado "SISTEMA DE INFORMACIÓN PARA EL CONTROL Y SEGUIMIENTO DE ENTREGA DE CANASTAS ESTUDIANTILES EN LA CIUDAD DE EL ALTO, CASO: DIRECCIÓN DE EDUCACIÓN" el cual procesa automáticamente la información para realizar dichas entregas. El manual del sistema informático, es una herramienta útil para la implementación y automatización de trabajo minucioso para el control y seguimiento de entregas, que permitirá reflejar resultados de todo el proceso y así lograr conseguir las metas y objetivos de la institución para el trabajo que se le ha encargado. En el presente Manual del Usuario, se explican detalladamente los pasos y procedimientos para su manejo, así como los reportes e información que proporciona.

2. MODULO ADMINISTRADOR

El modulo del administrador tienes todos los privilegios para utilizar y acceder el sistema, registrar, editar y eliminar; también puede monitorear los registros de cada usuario que esta interactuando con el sistema, así mismo asumirá responsabilidades cautelosas ya que dependerá de la administración, el reflejo de resultados que, después se mostrara como respaldo hacia el gobierno municipal.

3. MODULO DE SUPERVISOR

Este módulo permite a los supervisores registrar y controlar las entregas, dentro de ese proceso también podrá realizar modificaciones en datos de estudiantes que presenten observaciones en sus registros, por otra parte el supervisor podrá generar nuevas actas de entregas para registro actualizados.

4. MODULO TEC. RESPONSABLE

Este módulo permite a los técnicos responsable de entregas, realizar el registro de entregas agrupados por cursos, además deberán completar las entregas con la subidas de actas ya sea en PDF o imagen fotográficas, para las entregas tendrá la opción de editar datos del estudiante.

5. FUNCIONAMIENTO DEL SISTEMA

El sistema está alojado en un servidor interno de la institución y para acceder a ello deberá recibir una invitación de la parte administrativa para poder realizar el uso del mismo

5.1. Página principal

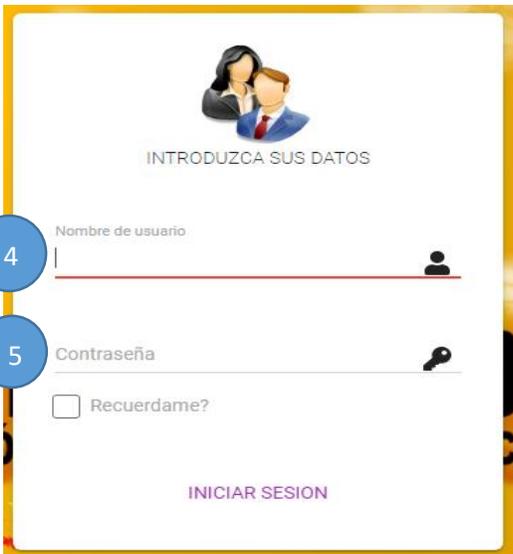
La página principal es aquella donde se encuentra algunas informaciones del proyecto y de labor de la institución, además de que contiene un módulo de consultas para la revisión de datos del estudiante y desde luego el link para poder iniciar sesión



- 1: Link de Inicio de sesión
- 2: Campo para buscar datos del estudiante, por CI. O RUDE.
- 3: Botón de búsqueda

5.2. Módulo de Inicio de sesión

Es la página donde se realiza la autenticación de los usuarios registrados para el ingreso al sistema.



INTRODUZCA SUS DATOS

4 Nombre de usuario

5 Contraseña

Recuérdame?

INICIAR SESION

- 4: Nombre de usuario
- 5: Contraseña

5.3. Módulo de Inicial de ingreso al sistema

Es la página donde, una vez logueados, podremos ver un poco de información del sistema y en caso de ser administración podremos ver algunas actividades que nos servirán de información en los usos del sistema.



6: Menú principal

7: Información general

5.4. Menú principal

En la barra lateral izquierdo encontramos el menú desplegable con diferentes opciones que ayudaran en el funcionamiento del sistema, dentro de los menús encontramos opciones de registros, de información, de reportes y de actas



8: Modulo principal

9: Modulo de registros

10: Modulo de entregas

11: Modulo de actas, reportes e informes.

5.4.1. Módulo de registros

En este menú desplegable encontraremos los registros del sistema, de los usuarios, de los estudiantes, los tutores, y de las Unidades educativas



12: Gestión de registro de usuarios

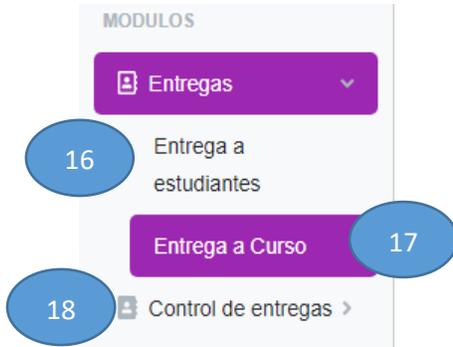
13: Gestión de registro de Estudiantes

14: Gestión de registro de tutores

15: Gestión de registro de Unidades educativas

5.4.2. Módulo de entregas

En este menú desplegable encontraremos los registros de entregas, a los estudiantes, y sus tutores, y las opciones de registro.



- 16: Registro de entregas por estudiante
- 17: Registro de entregas agrupadas por curso
- 18: Gestión y control de entregas

5.4.3. Módulo de reportes en actas e informes

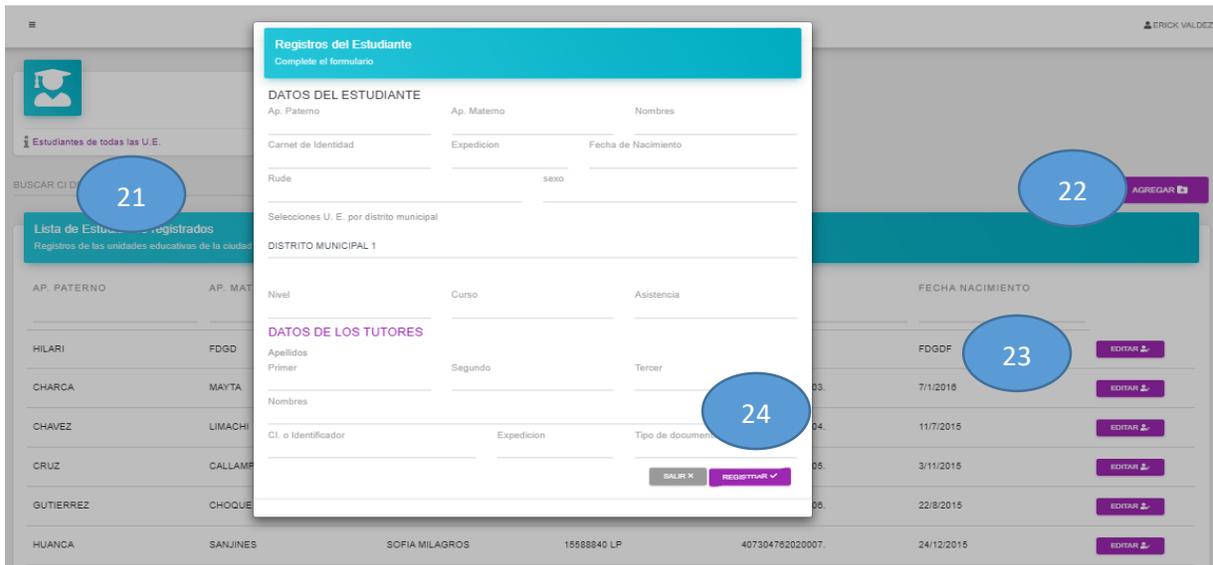
En este menú desplegable encontraremos la información del avance de la labor que se realiza de manera sistematizada.



- 19: Revisión y generación de actas de las U.E.
- 20: Revisión y descarga de informes específicos

5.5. Contenidos principales: Registros de estudiantes

Se muestran los contenidos de las diferentes visualizaciones del sistema para los registros, edición y validación de datos informativos.



- 21: Búsqueda y filtrado de datos
- 22: Botón de nuevo registro
- 23: Listado de datos de los estudiantes
- 24: Formulario de registro de estudiantes.

5.6. Contenido principal: Registro de entregas

Se muestran los contenidos para las entregas a los estudiantes agrupados por cursos, que deben ser especificados para su selección

N°	RUDE	NOMBRE COMPLETO	TUTORES	ESTADO	OPCIONES
1	407301112018001.	ADUVIRI CHAMISSO CAMILA	CHAMISSO TUCCO LOURDES CAMELIA CI: 8068498 LP	HABILITADO	<input type="checkbox"/> EDITAR
2	407300502018010.	AMADOR JIMENEZ SERGIO LEONEL	JIMENEZ SUAREZ ANA GERALDINA CI: 8044007 LP	HABILITADO	<input type="checkbox"/> EDITAR
3	407301112018007.	AMARU QUIROGA	QUISPE CALLISAYA YRMA CI: 8044007 LP	HABILITADO	<input type="checkbox"/> EDITAR
4	407301112018041.	APAZA TAMBO KEVIN	TAMBO APAZA FLORA CI: 8997147 LP	HABILITADO	<input type="checkbox"/> EDITAR
5	407301112019001.	ARRATIA SUMI GUSTAVO BRAYAN	SUMI DE ARRATIA HILDA HILARIA CI: 4363819 LP	HABILITADO	<input type="checkbox"/> EDITAR

- 25: Información de la Unidad seleccionada
- 26: Filtrado de campos para agrupar los cursos
- 27: Listado de los estudiantes
- 28: Listado de los tutores del estudiante.
- 29: Estado y tickeo para adjuntar las entregas

22	807300412018077.	VILLCA FLORES LYA FERNANDA	FLORES JARANDILLA VILLCA GABRIELA VICTORIA CI: 9954018 LP	HABILITADO	<input type="checkbox"/> EDITAR
23	407301332018020.	VINO ESPINOZA ZOE VIOLETA		HABILITADO	<input type="checkbox"/> EDITAR
24	407301112019025.	ZARATE FLORES DORIAN BRYAN		HABILITADO	<input type="checkbox"/> EDITAR

ENVIAR LOS SELECCIONADOS ✓

Completar entrega

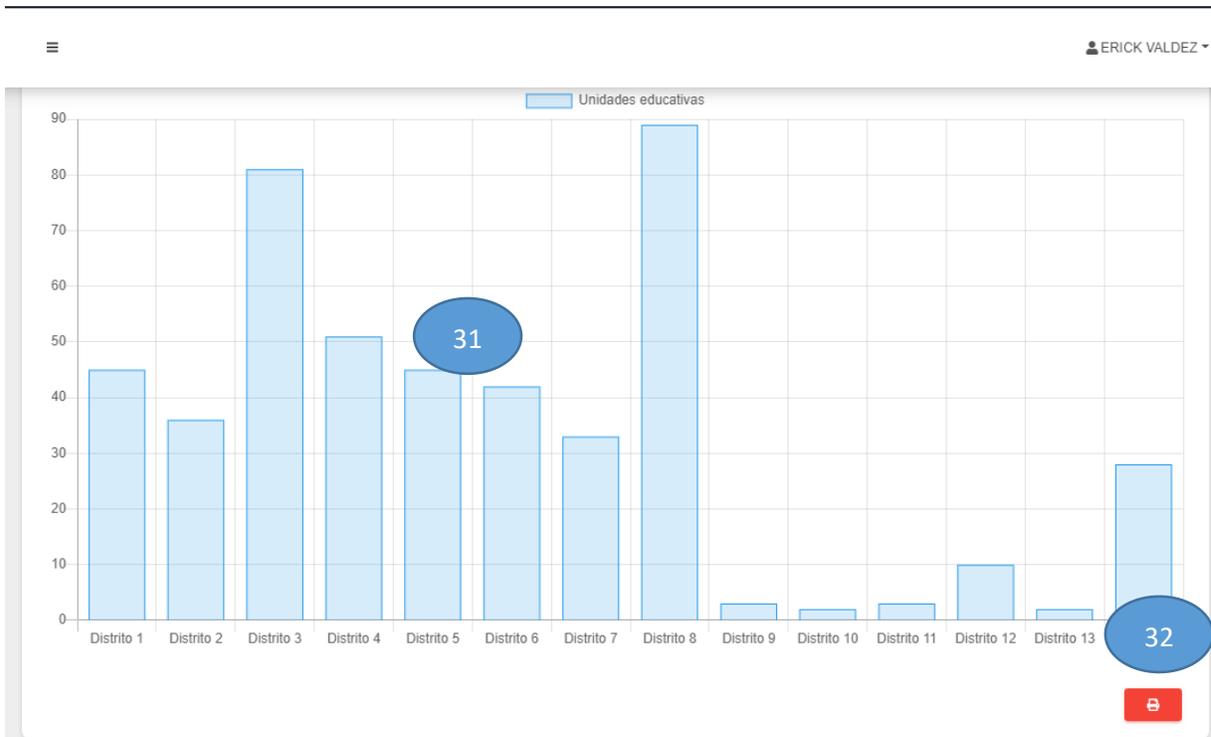
Arrastre el archivo o haga click para subir al sistema

SUBIR ARCHIVOS

- 30: Confirmación de entrega con la fecha según el acta
- 31: Campo para subir actas en formato PDF o imágenes fotográficas.

5.7. Contenido principal: Informe de resultados

Se muestran los contenidos del avance del trabajo en las entregas, con información estadística que resulte útil en la convalidación una vez termine la labor



31: Informes estadísticos sobre resultados obtenidos en el avance del trabajo

32: Impresión de resultados basados en los resultados estadísticos

ANEXO E
MANUAL TÉCNICO

MANUAL TÉCNICO

Sistema de información para el control y seguimiento de entregas



**DIRECCIÓN DE EDUCACIÓN –
GAMEA**

Autor: Jose Jossue Hilari Vega

La paz – Bolivia

2020

1. INTRODUCCIÓN

El presente manual técnico describe los pasos necesarios para cualquier persona que tenga ciertas bases de sistemas pueda realizar la instalación del sistema creado para la dirección de educación del gamea. Es importante tener en cuenta que en el manual se hace mención a las especificaciones mínimas de hardware y software para la correcta instalación del aplicativo.

2. OBJETIVO

Brindar la información necesaria para poder realizar la instalación y configuración del sistema aplicativo basado en un conocimiento general sobre informática y administración de redes.

3. REQUERIMIENTOS DEL SISTEMA

3.1. Requerimientos mínimos de Hardware

- Procesador: Core i3
- Memoria RAM: Mínimo: 2 Gigabytes (GB)
- Disco Duro: 500Gb.

3.2. Requerimientos mínimos de Software

- Sistema Operativo: Linux debían o Windows 7, 8 o 10.
- Conexión internet
- Navegadores (Mozilla Firefox o Google Chrome)

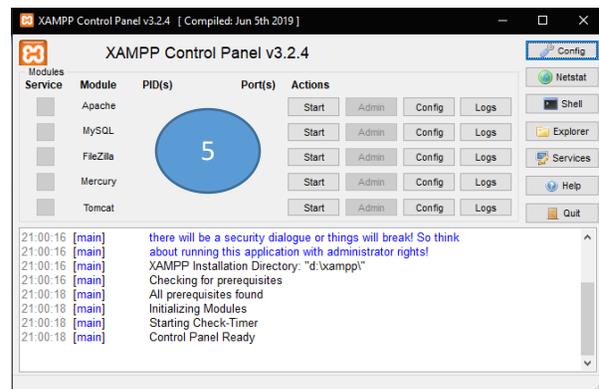
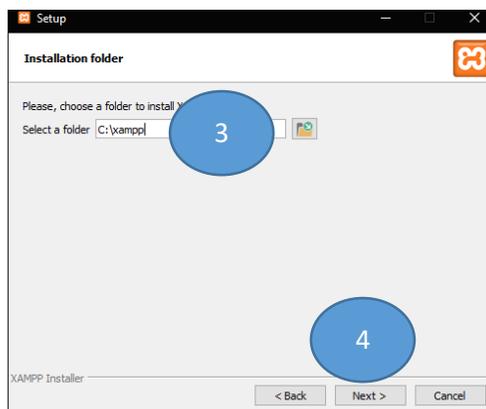
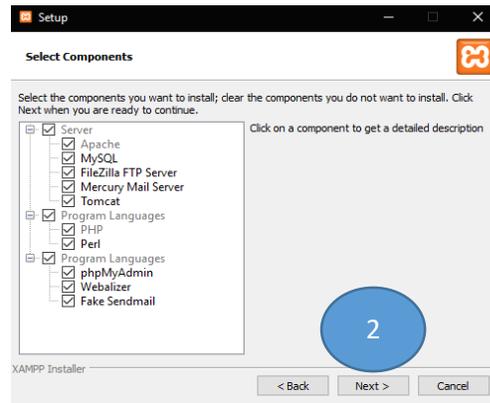
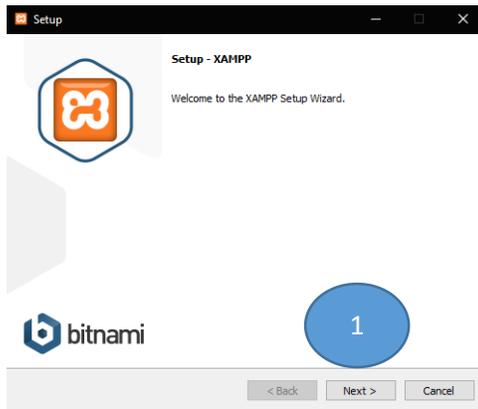
4. INSTALACIÓN DEL SISTEMA (servidor local)

4.1. Instalación de XAMPP

XAMPP es un paquete libre que gestiona herramientas en aplicaciones web como, base de datos Mariadb mysql, servidor local Apache, lenguaje PHP, además de la administración grafica en phpmyadmin y config.php.

A continuación descargamos el paquete XAMPP de su página oficial: <https://www.apachefriends.org/es/index.html>. Lo instalaremos en el disco de preferencia que se decida, revisando la documentación oficial de XAMPP. Una vez instalado iniciamos los servicios como se ve en la siguiente imagen, donde encontraremos el panel de control de XAMPP.

CONFIGURAR INSTALACIÓN



- 1: Configuración de instalación
- 2: Instalación de complementos y herramientas
- 3: Seleccionamos directorio de instalación
- 4: Confirmar y terminar la instalación.
- 5: Panel de control e iniciar servicios en XAMPP

Luego de terminar de configurar la instalación queda, iniciar servicios en los archivos del sistema, que en este caso nos basaremos en el sistema operativo de Windows en sus diferentes versiones.

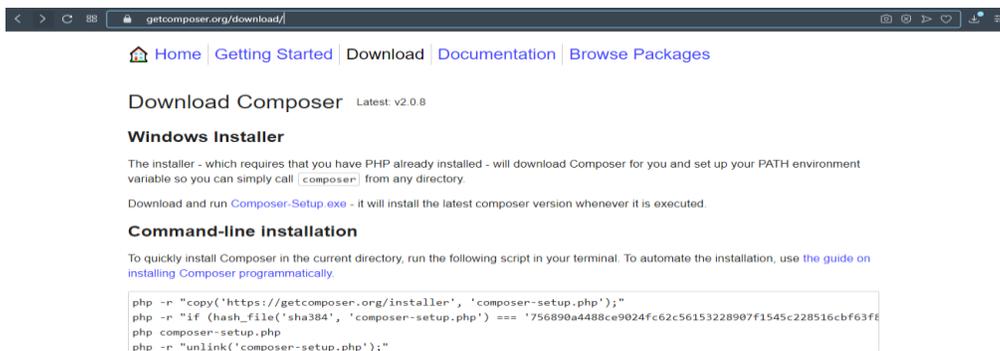
4.2. Instalación Laravel

Para la instalación del framework de desarrollo laravel nos basaremos en la documentación oficial de laravel. Ya que esta herramienta nos servirá para iniciar las dependencias de herramientas que vienen en el paquete de instalación, que nos proporcionan en las fases de seguridad, librerías y extensiones oficiales y de uso

libre para el normal funcionamiento del sistema: para ello nos dirigimos a <https://laravel.com>

COMPOSER

Para la instalación de laravel nos basaremos en la herramienta de configuración composer que la encontramos en: <https://getcomposer.org/download/>, ya que nos servirá de conexión entre el funcionamiento del sistema y los servicios php, a través de las extensiones artisan que proponen laravel.



Comando de instalación de laravel:

```
composer global require laravel/installer
```

Además de composer encontraremos herramientas de servicios en el modelado y desarrollo en la extensión de aplicaciones, como es node.js que lo encontraremos en su página oficial: <https://nodejs.org/es/>, con esto podríamos terminar la fase de instalación en laravel.



4.3. Instalación del proyecto laravel en directorio XAMPP

En la instalación del proyecto laravel que se entregó a la institución, una vez terminada, en su versión 1.0, de la cual encontraremos todos los archivos que son parte del directorio del proyecto. Encontrado el proyecto procedemos a copiar el archivo al directorio de xampp en donde se almacenan los proyectos de servicios php que proporcionan el servidor apache de XAMPP.

Nombre	Fecha de modifica...	Tipo
 proyecto-laravel.rar	9/12/2020 22:51	Archivo WinRAR

Procederemos a descomprimir y encontrando la carpeta, queda copiar al directorio de instalación que configuramos en XAMPP



4.4. Instalación de la base de datos

En esta fase realizamos la creación de la base de datos e importamos los datos obtenidos en las fases de planificación del sistema, que luego se derivó en la fase de recolección de datos.

Creación de la base de datos

Para este apartado realizaremos la creación de las base de datos, y consideraremos las distintas opciones que nos proporcionan mariadb mysql en XAMPP, ya sea por directorio de comandos o en su versión de phpmyadmin.

Bases de datos

Crear base de datos

Nombre de la base de datos: utf8mb4_general_ci

Filtros

Que contengan la palabra:

Base de datos Cotejamiento Acción

Importación de la base de datos

La importación de los datos nos servirá como inicio de funcionamiento del sistema ya que con los datos incorporados procederemos a dar uso a los módulos del sistema que proporcionan la información de suma importancia a la institución.

Archivo a importar:

El archivo puede ser comprimido (gzip, bzip2, zip) o descomprimido.
A compressed file's name must end in `.[format].[compression]`. Example: `.sql.zip`

Buscar en su ordenador: Ningún archivo seleccionado (Máximo: 40MB)

También puede arrastrar un archivo en cualquier página.

Conjunto de caracteres del archivo:

Importación parcial:

Allow the interruption of an import in case the script detects it is close to the PHP timeout limit. (This might be a good way to import large files, however it can break trans

Omitir esta cantidad de consultas (en SQL) desde la primera:

Otras opciones:

Habilite la revisión de las claves foráneas

Formato:

Opciones específicas al formato:

Modalidad SQL compatible:

No utilizar `AUTO_INCREMENT` con el valor 0

La copia de la base de datos lo encontraremos en el archivo comprimido del proyecto laravel, este archivo lleva la extensión sql la cual nos garantiza que sea el archivo de importación de la base de datos mariadb.

Nombre	Fecha de modifica...	Tipo	Tamaño
 2020-12-07-02-08-52.zip	7/12/2020 02:10	Archivo WinRAR Z...	52.916 KB
 mysql-db.sql	7/12/2020 02:08	Archivo SQL	84.310 KB

Una vez terminada la importación solo queda iniciar la aplicación del sistema, en el servidor local para que puedan tener acceso todos los equipos que se encuentren con IP repartida dentro la institución.

Para iniciar el servidor nos basaremos en la documentación de laravel o en su defecto nos enfocaremos a configurar el servicio en apache de XAMPP.

ANEXO F

AVALES



GOBIERNO AUTÓNOMO MUNICIPAL DE EL ALTO



La Paz – El Alto, Noviembre de 2020

Señor:

Ing. David Carlos Mamani Quispe
DIRECTOR DE LA CARRERA INGENIERIA DE SISTEMAS
Presente.-

REF.: AVAL DE CONFORMIDAD INSTITUCIONAL

De nuestra mayor consideración:

Por la presente notificamos a su persona que la DIRECCION DE EDUCACION dependiente de la SECRETARIA MUNICIPAL DE DESARROLLO HUMANO del GOBIERNO AUTONOMO MUNICIPAL DE EL ALTO, avala la ejecución de proyecto: "SISTEMA DE INFORMACION PARA EL CONTROL Y SEGUIMIENTO DE ENTREGA DE CANASTAS ESTUDIANTILES EN LA CIUDAD DE EL ALTO" presentado a la institución mediante el ÁREA DE TECNOLOGÍAS DE LA INFORMACIÓN Y COMUNICACIÓN - TIC'S, por el Universitario **Jose Jossue Hilari Vega**, con **CI. 10077184 LP**.

Teniendo en cuenta los aspectos del proyecto, declaramos conocer y aceptar los términos y condiciones previstas para su ejecución, estando conformes con todas aquellas actividades que se prevean realizar con nuestro aporte y/o recursos establecidos en el proyecto. Las actividades previstas en el proyecto incluyen el desarrollo e implementación del sistema que beneficiara a la institución.

Sin otro particular, saludamos a su persona deseándole éxitos en su labor.

Atentamente.


Erik J. Ramos Valdez
RESPONSABLE DEL AREA DE TECNOLOGIAS
DE LA INFORMACION Y COMUNICACION
EL ALTO DIRECCION DE EDUCACION
GOBIERNO AUTONOMO MUNICIPAL DE EL ALTO





GOBIERNO AUTÓNOMO MUNICIPAL DE EL ALTO



La Paz – El Alto, 26 de noviembre de 2020

Señor(a):

Ing. Maricel Yarari Mamani
DOCENTE TALLER DE GRADO II
Ingeniería de sistemas UPEA.
Presente. -

REF.: AVAL DE CONFORMIDAD

De mi mayor consideración

Por intermedio de la presente le hago llegar un cordial saludo, el motivo es hacerle conocer a su persona, que el universitario Jose Jossue Hilari Vega con CI.: 10077184 L.P. realizo y cumplió con los requerimientos de la institución para su proyecto de grado titulado **“SISTEMA DE INFORMACIÓN PARA EL CONTROL Y SEGUIMIENTO DE ENTREGA DE CANASTAS ESTUDIANTILES EN LA CIUDAD DE EL ALTO”**, CASO: DIRECCIÓN DE EDUCACIÓN – GAMEA, El proyecto será de gran ayuda y aporte para nuestra institución, con el cual damos paso libre para su defensa pública.

Sin otro particular me despido, con las más distinguidas consideraciones

Atentamente:


Erik J. Ramos Valdez
RESPONSABLE DEL ÁREA DE TECNOLOGÍAS
DE LA INFORMACIÓN Y COMUNICACIÓN
DIRECCIÓN DE EDUCACIÓN
GOBIERNO AUTÓNOMO MUNICIPAL DE EL ALTO



El Alto 25 de Noviembre de 2020

Señores:

HONORABLE CONSEJO DE CARRERA
CARRERA INGENIERÍA DE SISTEMAS

Presente. –

Ref.: AVAL DE CONFORMIDAD

Distinguida autoridad

Mediante la presente tengo a bien comunicarle mi conformidad del proyecto de grado "SISTEMA DE INFORMACIÓN PARA EL CONTROL Y SEGUIMIENTO DE ENTREGA DE CANASTAS ESTUDIANTILES EN LA CIUDAD DE EL ALTO", CASO: DIRECCIÓN DE EDUCACIÓN – GAMEA, que propone el postulante Jose Jossue Hilari Vega con C.I.: 10077184 LP. Para su defensa publica, evaluación correspondiente a la materia "Taller de Grado II", de acuerdo a la reglamentación vigente de la carrera de Ingeniería de sistemas de la Universidad Pública de El Alto.

Sin otro particular, me despido con las consideraciones más distinguidas.

Atentamente.



Ing. Maricel Yarari Mamani
DOCENTE TALLER DE GRADO II

El Alto 1 de diciembre de 2020

Señores:

HONORABLE CONSEJO DE CARRERA
CARRERA INGENIERÍA DE SISTEMAS

Presente. –

Ref.: AVAL DE CONFORMIDAD

Distinguida autoridad

Mediante la presente tengo a bien comunicarle mi conformidad del proyecto de grado "SISTEMA DE INFORMACIÓN PARA EL CONTROL Y SEGUIMIENTO DE ENTREGA DE CANASTAS ESTUDIANTILES EN LA CIUDAD DE EL ALTO", CASO: DIRECCIÓN DE EDUCACIÓN – GAMEA, que propone el postulante Jose Jossue Hilari Vega con C.I.: 10077184 LP. Para su defensa publica, evaluación correspondiente a la materia "Taller de Grado II", de acuerdo a la reglamentación vigente de la carrera de Ingeniería de sistemas de la Universidad Pública de El Alto.

Sin otro particular, me despido con las consideraciones más distinguidas.

Atentamente.



Ing. Enrique Flores Baltazar
TUTOR ESPECIALISTA

El Alto 26 de noviembre de 2020

Señores:

HONORABLE CONSEJO DE CARRERA
CARRERA INGENIERÍA DE SISTEMAS

Presente. –

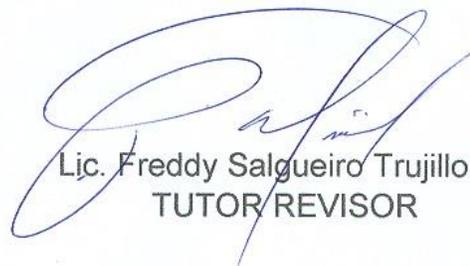
REF.: AVAL DE CONFORMIDAD

Distinguida autoridad

Mediante la presente tengo a bien comunicarle mi conformidad del proyecto de grado "SISTEMA DE INFORMACIÓN PARA EL CONTROL Y SEGUIMIENTO DE ENTREGA DE CANASTAS ESTUDIANTILES EN LA CIUDAD DE EL ALTO", CASO: DIRECCIÓN DE EDUCACIÓN – GAMEA, que propone el postulante Jose Jossue Hilari Vega con C.I.: 10077184 LP. Para su defensa publica, evaluación correspondiente a la materia "Taller de Grado II", de acuerdo a la reglamentación vigente de la carrera de Ingeniería de sistemas de la Universidad Pública de El Alto.

Sin otro particular, me despido con las consideraciones más distinguidas.

Atentamente.



Lic. Freddy Salgueiro Trujillo
TUTOR REVISOR