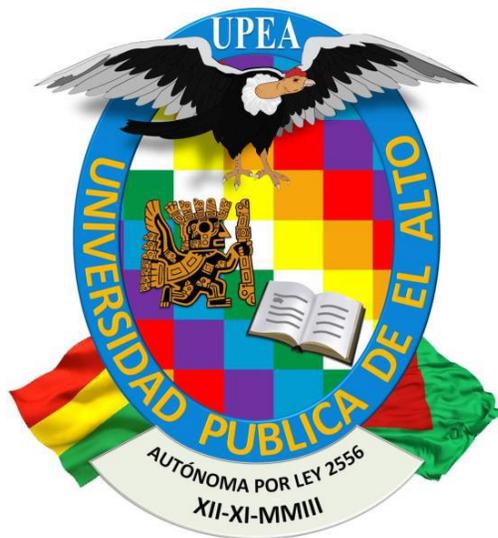


UNIVERSIDAD PÚBLICA DE EL ALTO

CARRERA INGENIERÍA DE SISTEMAS



PROYECTO DE GRADO

SISTEMA DE ADMISIÓN ESTUDIANTIL PARA LA GESTIÓN ACADÉMICA DE LAS CARRERAS

CASO: (Universidad Pública de El Alto)

**Para optar al título de Licenciatura en Ingeniería de Sistemas
MENCIÓN: Informática y Comunicaciones**

Postulante: Manfred Alejandro Maidana Burgos

Tutor Metodológico: Ing. Marisol Arguedas Balladares

Tutor Revisor: Ing. Fanny Helen Perez Mamani

Tutor Especialista: Ing. Juan Regis Muñoz Sirpa

EL ALTO – BOLIVIA

2020

Dedicatoria

El presente trabajo va dedicado a mis padres; Susana Burgos y Felix Maidana, quienes constantemente me brindaron su apoyo incondicional, amor y paciencia, que sin ellos este proyecto no sería posible. A Danitza Mamani Choque, que siempre estuvo acompañándome y brindando su apoyo incondicional, así como la fuerza moral para seguir adelante. A mis hermanos y hermanas que sin su apoyo y motivación no llegaría a ser quien soy

Agradecimientos

A todos aquellos que me brindaron su apoyo durante la realización y conclusión de este proyecto de grado.

A mis padres por ser el pilar fundamental de mi educación y por su constante apoyo y motivación.

A mi tutora metodológica Ing. Marisol Arguedas Balladares por su conocimiento, paciencia, tiempo y sus enseñanzas.

Un profundo agradecimiento a mi tutor especialista Ing. Juan Regis Muñes Sirpa, por todas sus enseñanzas, por el conocimiento compartido y la paciencia prestadas para la finalización de este proyecto.

Un profundo agradecimiento a mi tutora revisora Ing. Fanny Helen Perez Mamani por todo su apoyo y motivación, por el tiempo dedicado, por el conocimiento y sugerencias brindadas en el desarrollo de este proyecto.

A la unidad de SIE (Sistemas de Información y Estadística) por el apoyo y las sugerencias que me brindaron.

A la carrera de Ingeniería de Sistemas por todo el conocimiento que me brindó.

A la Universidad Pública de El Alto por acogerme todos estos años entre sus instalaciones y aulas.

RESUMEN

El sistema de admisión estudiantil para la gestión académica de las carreras en la Universidad Pública de El Alto se desarrolla como una respuesta a la necesidad de mejorar administración y gestión de la información del proceso de admisión estudiantil en sus diferentes modalidades, organizando y centralizando la información de los postulantes y responder a necesidades como la impresión de reportes, la generación de actas de notas y certificados de habilitación, el control de notas. De esta manera se logró proporcionar información confiable, segura y oportuna a los coordinadores, docentes y a los postulantes del proceso de admisión estudiantil.

Para la realización del proyecto se utilizó la metodología web UWE, que proporciona las herramientas necesarias para el desarrollo del sistema orientado a la web.

Para el desarrollo del sistema se usaron las herramientas PHP para la programación de lado de servidor, el framework de JavaScript jQuery para la programación de lado del cliente y para un funcionamiento más dinámico del sistema. Para la base de datos se empleó el gestor de base de datos MariaDB. Se tiene como servidor el uso de Apache2 para los servidores.

Para la construcción del sistema se empleó el framework CodeIgniter, que utiliza el modelo de desarrollo MVC que facilita la organización de los archivos del sistema.

Para el maquetado del sistema se empleó el uso de Bootstrap4 y jQuery.

Después de terminar el desarrollo del sistema de Admisión Estudiantil para la gestión académica de las carreras se procedió a realizar las pruebas de funcionamiento, calidad y seguridad en base a las normas ISO/IEC 9126 e ISO 27000, con las que se pudo constatar que el sistema responde a los requerimientos de la Universidad Pública de El Alto.

Finalmente se concluye que se lograron cumplir con los objetivos que fueron planteados y que el sistema fue implementado dentro de la Universidad Pública de El Alto.

Palabras clave: Admisión Estudiantil, Gestión Académica, Sistema de Admisión, UWE

ABSTRACT

The student admission system for the academic management of careers at the Public University of El Alto is developed as a response to the need to improve administration and information management of the student admission process in its different modalities, organizing and centralizing the information of the applicants and respond to needs such as the printing of reports, the generation of transcripts and qualification certificates, the control of marks. In this way, it was possible to provide reliable, safe and timely information to the coordinators, teachers and applicants of the student admission process.

To carry out the project, the UWE web methodology was used, which provided the necessary tools for the development of the web-oriented system.

For the development of the system, the PHP tools for server-side programming, the JavaScript jQuery framework for client-side programming and for a more dynamic operation of the system were used. For the database, the MariaBD database manager was used. The server is the use of Apache2 for the servers.

To build the system, the CodeIgniter framework was used, which uses the MVC development model that facilitates the organization of system files. For the layout of the system, the use of Bootstrap4 and jQuery was used.

After finishing the development of the Student Admission system for the academic management of the careers, the performance, quality and safety tests were carried out based on the ISO / IEC 9126 and ISO 27000 standards, with which it was found that the system responds to the requirements of the Public University of El Alto. Finally, it is concluded that the objectives that were set were met and that the system was implemented within the Public University of El Alto.

Keywords: Student Admission, Academic Management, Admission System, UWE

CONTENIDO

1.	MARCO PRELIMINAR.....	1
1.1.	INTRODUCCIÓN.....	1
1.2.	ANTECEDENTES.....	2
1.3.	PLANTEAMIENTO DEL PROBLEMA.....	3
1.3.1.	Problema principal.....	3
1.3.2.	Problemas secundarios.....	3
1.4.	OBJETIVOS.....	4
1.4.1.	General.....	4
1.4.2.	Específicos.....	4
1.5.	JUSTIFICACIÓN.....	4
1.5.1.	Técnica.....	4
1.5.2.	Económica.....	5
1.5.3.	Social.....	5
1.6.	METODOLOGÍA.....	5
1.6.1.	Ingeniería de Software.....	5
1.6.2.	Ingeniería de Requerimientos.....	6
1.6.3.	Ingeniería Web basada en UML (UWE).....	6
1.7.	HERRAMIENTAS.....	7
1.7.1.	Lenguaje de programación: PHP.....	7
1.7.2.	Gestor de Base de Datos: MariaDB.....	7
1.7.3.	Herramientas de desarrollo web.....	8
1.8.	LÍMITES Y ALCANCES.....	9
1.8.1.	Limites.....	9
1.8.2.	Alcances.....	9
1.9.	APORTES.....	9
2.	MARCO TEORICO.....	11
2.1.	SISTEMAS DE INFORMACIÓN.....	11
2.1.1.	Web.....	12
2.2.	ADMISIÓN ESTUDIANTIL.....	12
2.3.	GESTIÓN ACADÉMICA.....	13

2.4.	INTERPOLACIÓN	13
2.5.	INGENIERIA DE SOFTWARE	14
2.6.	METODOLOGÍA.....	14
2.6.1.	Metodología UWE	14
2.7.	HERRAMIENTAS.....	21
2.7.1.	Gestor de Base de Datos MariaDB	21
2.7.2.	Lenguaje de Programación PHP	22
2.7.3.	Framework CodeIgniter.....	23
2.7.4.	Estructura de CodeIgniter	24
2.7.5.	Lenguaje de Hipertexto HTML.....	25
2.7.6.	Bootstrap.....	26
2.7.7.	JavaScript.....	26
2.7.8.	jQuery	27
2.8.	METRICAS DE CALIDAD DE SOFTWARE	28
2.8.1.	Calidad de Software	28
2.8.2.	Control de calidad y pruebas.....	29
2.9.	CALIDAD DEL SISTEMA	30
2.9.1.	Norma ISO/IEC 9126.....	30
2.10.	COSTOS.....	41
2.10.1.	El Modelo COCOMO II	41
2.11.	SEGURIDAD DE LA INFORMACIÓN	43
2.11.1.	SGSI (Sistema de Gestión de la Seguridad de la Información)	43
2.11.2.	Aspectos de seguridad para los sistemas tradicionales y los basados en web	45
2.12.	PRUEBAS DE SOFTWARE	46
2.12.1.	Pruebas de caja negra	46
2.12.2.	Pruebas de caja Blanca.....	48
3.	MARCO APLICATIVO.....	49
3.1.	ANÁLISIS DE REQUERIMIENTOS	49
3.1.1.	Actores	50
3.1.2.	Casos de Uso	51
3.2.	MODELO DE CONTENIDO.....	63
3.2.1.	Diagrama de clases.....	64

3.3.	MODELOS DE NAVEGACIÓN.....	66
3.4.	MODELO DE PRESENTACIÓN.....	70
3.5.	DESARROLLO DE SISTEMA.....	74
3.5.1.	Interfaz de Inicio de sesión y registro de estudiantes.....	74
3.5.2.	Interfaz de Administrador	77
3.5.3.	Interfaz de Coordinador	77
3.5.4.	Interfaz de Docente.....	78
3.5.5.	Interfaz de postulante	79
3.6.	PRUEBAS DE SOFTWARE	79
3.6.1.	Pruebas de caja negra	79
4.	MÉTRICAS DE CALIDAD, ESTIMACIÓN DE COSTOS DE SOFTWARE	82
4.1.	MÉTRICAS DE CALIDAD	82
4.1.1.	Funcionalidad	82
4.1.2.	Confiabilidad	86
4.1.3.	Usabilidad.....	87
4.1.4.	Eficiencia.....	88
4.1.5.	Mantenibilidad	90
4.1.6.	Portabilidad	91
4.2.	ESTIMACIÓN DE COSTOS DE SOFTWARE	92
4.2.1.	Método de estimación COCOMO II.....	92
4.3.	SEGURIDAD.....	96
4.3.1.	Seguridad Lógica.....	96
4.3.2.	Seguridad Física	97
5.	CONCLUSIONES Y RECOMENDACIONES.....	98
5.1.	CONCLUSIONES.....	98
5.2.	RECOMENDACIONES	98
	BIBLIOGRAFÍA.....	100
	ANEXOS	102

INDICE DE FIGURAS

Figura 2.1 Actividades realizadas en un sistema de información	11
Figura 2.2 Nombres estereotipados y sus iconos	17
Figura 2.3 Modelo de contenido	18
Figura 2.4 Modelo de navegación	19
Figura 2.5 Modelo de presentación	20
Figura 2.6 Interacción de los componentes modelo, vista y controlador de la arquitectura MVC	25
Figura 2.7 Calidad interna y externa de la norma ISO 9126.....	31
Figura 2.8 Funcionalidad ISO/IEC 9126	32
Figura 2.9 Evaluación Interna, externa y Calidad de Uso ISO/IEC 9126.....	32
Figura 2.10 Característica de funcionalidad ISO/IEC 9126	34
Figura 2.11 Característica de Confiabilidad ISO/IEC 9126.....	34
Figura 2.12 Característica de Usabilidad ISO/IEC 9126.....	36
Figura 2.13 Característica de Eficiencia	37
Figura 2.14 Característica de Mantenimiento	38
Figura 2.15 Característica de portabilidad	39
Figura 2.16 Característica Calidad de uso	40
Figura 2.17 Modelos de estimación COCOMO	42
Figura 2.18 Ciclo de vida de un SGSI	44
Figura 3.1 Casos de uso general.....	52
Figura 3.2 Caso de uso para la admisión de postulantes.....	53
Figura 3.3 Diagrama de caso de uso Administración de usuarios.....	54
Figura 3.4 Caso de uso administración de postulantes	56
Figura 3.5 Gestión de la modalidad de curso preuniversitario.....	57
Figura 3.6 Caso de uso de la gestión de la modalidad de prueba de suficiencia académica.....	59
Figura 3.7 Caso de uso gestión de la modalidad especial	60
Figura 3.8 Caso de uso Habilitación de fechas	61
Figura 3.9 Caso de uso Gestión de reportes	62
Figura 3.10 Diagrama de clases sistema de admisión estudiantil	64

Figura 3.11	Diagrama de clases Base_upea.....	65
Figura 3.12	Modelo de navegación para administrador.....	66
Figura 3.13	Modelo de navegación para el módulo de coordinador	67
Figura 3.14	Modelo de navegación para el módulo docente	68
Figura 3.15	Modelo de navegación para el modulo postulante	69
Figura 3.16	Diagrama de presentación Página principal	70
Figura 3.17	Diagrama de presentación Login.....	71
Figura 3.18	Diagrama de presentación Administrador	71
Figura 3.19	Diagrama de presentación Coordinador.....	72
Figura 3.20	Diagrama de presentación Docente	73
Figura 3.21	Diagrama de presentación Estudiante.....	74
Figura 3.22	Inicio de sesión y registro de postulantes.....	75
Figura 3.23	Verificación para el registro de postulantes.....	75
Figura 3.24	Formulario de inscripción de postulante	76
Figura 3.25	Administración de usuarios	77
Figura 3.26	Lista de postulantes.....	77
Figura 3.27	Listado de materias	78
Figura 3.28	Listado de inscripciones	79

INDICE DE TABLAS

Tabla 3.1 Herramientas usadas para la ingeniería de requerimientos.....	49
Tabla 3.2 Actores.....	50
Tabla 3.3 Descripción del caso de uso admisión de postulantes.....	53
Tabla 3.4 Descripción de los casos de uso Administración de usuarios.....	55
Tabla 3.5 Descripción del caso de uso administración de postulantes.....	56
Tabla 3.6 Descripción del caso de uso gestión de modalidad de curso preuniversitario.....	57
Tabla 3.7 Descripción del caso de uso de la modalidad de prueba de suficiencia académica.....	59
Tabla 3.8 Descripción del caso de uso gestión de la modalidad especial.....	60
Tabla 3.9 Descripción del caso de uso Habilitación de fechas.....	61
Tabla 3.10 Descripción del caso de uso Gestión de reportes.....	63
Tabla 3.11 Caso de prueba de Ingreso al sistema.....	80
Tabla 3.12 Caso de prueba de registro de usuario.....	80
Tabla 3.13 Caso de prueba de registro de estudiante.....	81
Tabla 4.1 Parámetros de medición de punto función.....	83
Tabla 4.2 Punto función sin ajustar.....	83
Tabla 4.3 Factores de complejidad.....	84
Tabla 4.4 Parámetros de medición de usabilidad.....	87
Tabla 4.5 Tabla de usabilidad del sistema.....	88
Tabla 4.6 Escala de valores de eficiencia.....	89
Tabla 4.7 Valoración de la eficiencia del sistema.....	89
Tabla 4.8 Valores para la mantenibilidad.....	90
Tabla 4.9 Coeficientes de modelo COCOMO II.....	93
Tabla 4.10 Ecuaciones COCOMO II.....	93
Tabla 4.11 Cálculo de los atributos FAE.....	94

1. MARCO PRELIMINAR

1.1. INTRODUCCIÓN

El uso de las tecnologías de información para centralizar la información y automatizar procesos ha ido en constante crecimiento y de manera exponencial con el paso de los años, desde la creación de las primeras aplicaciones y con la evolución del internet su uso ha sido casi indispensable al día de hoy, y es ahora que nos encontramos en la era de la tecnología y con la evolución de la web, las aplicaciones y plataformas web han tomado un amplio terreno dentro de estas tecnologías de la información como propuesta efectiva para la solución de estos problemas.

La Universidad Pública de El Alto no queda exento de estos problemas, y por lo tanto demanda soluciones para el manejo de la información, así como la centralización de este y la automatización de los procesos administrativos que se llevan dentro de la universidad. Y es de esta manera que el proceso para la admisión estudiantil necesita de centralizar la información que maneja, así como la automatización de dicho proceso.

Los sistemas de administración web poseen una amplia gama de herramientas y características que se adecuan para la solución efectiva para la centralización y automatización del proceso, además de brindar otros beneficios para la Universidad Pública de El Alto, así como la presencia digital al mundo exterior y el poder brindar información a la comunidad de bachilleres que buscan una oferta académica.

El desarrollo del sistema de admisión estudiantil se realizará utilizando la metodología de desarrollo UWE, que emplea las herramientas UML que es usará la captura de requisitos y el modelado del proceso de admisión estudiantil de la UPEA. Se empleará el lenguaje PHP y el framework codeignite para facilitar el desarrollo del sistema de administración web, también se utilizará las herramientas de la WEB 2.0 como javascript, css, Ajax y json para un manejo más dinámico del sistema.

1.2. ANTECEDENTES

(Pemán B. Nieves, 2009), el DESARROLLO DE UN PORTAL WEB PARA LA GESTIÓN DE INFORMACIÓN DE INVESTIGACIÓN tiene por objetivo principal el desarrollo de un portal web para la gestión de contenidos dentro de una universidad y que pueda adaptarse a cualquier otra universidad, utilizando la tecnología Struts que permite desarrollar aplicaciones en Java, hace uso de la arquitectura MVC. Realizado en la Universidad Carlos III de Madrid.

(Ponce H, 2016), DESARROLLO E IMPLEMENTACIÓN DE UN SISTEMA WEB DE ADMISIÓN Y MATRICULACIÓN PARA EL INSTITUTO DE POSTGRADO DE LA UNIVERSIDAD ESTATAL PENÍNSULA DE SANTA ELENA tiene por objetivo implementar un sistema web para el instituto de postgrado UPSE que permita automatizar el proceso de admisión y matriculación, utiliza el framework ZK y el lenguaje java para la realización del sistema. Realizado en la Universidad Estatal Península de Santa Elena.

(Roque C., Marizol, 2014) el SISTEMA WEB DE INFORMACION Y SEGUIMIENTO ACADÉMICO Caso: Programa de Admisión Prefacultativa carrera de Psicología tiene como objetivo desarrollar un sistema web que permita un adecuado control y registro de los procesos realizados por el programa de admisión prefacultativa. Para lograr dicho objetivo utiliza la metodología AUP y el desarrollo del sistema web es llevado a cabo utilizando el framework Spring y el lenguaje Java. Realizado en la Universidad Mayor de San Andrés

(Tola M. Mirna), el SISTEMA INFORMATICO DE GESTION Y SEGUIMIENTO INFORMATICO tiene como objetivo la implementación de un sistema que coadyuve con el cumplimiento de requerimientos académicos y para la gestión del Instituto Técnico San Pablo. Utiliza la metodología UWE para adaptar el sistema de acuerdo a las necesidades de los usuarios. Realizó el desarrollo del sistema en el lenguaje PHP y usando el framework Laravel. Realizado en la Universidad Pública de El Alto.

1.3. PLANTEAMIENTO DEL PROBLEMA

1.3.1. Problema principal

En la Universidad Pública de El Alto la gestión del proceso de admisión estudiantil se realiza de manera manual lo que genera que la información se maneje de manera descentralizada, que el acceso y control de la información sea poco eficiente generando demoras en los tiempos de búsqueda y presentación de documentos.

1.3.2. Problemas secundarios

- La información se encuentra descentralizada lo que genera que no se tenga un control fidedigno sobre los datos de los postulantes.,.
- Los reportes se generan manualmente lo que ocasiona en pérdidas de tiempo al momento de preparar y presentar la información de postulantes.
- Las listas de postulantes se manejan en hojas de cálculo, que causa que la información no sea confiable y cause posibles pérdidas en la información.
- No existe un control riguroso de estudiantes que ya se encuentren inscritos en otras carreras o que sean de casos especiales, lo que genera duplicidad en los datos que causa que los estudiantes tengan que realizar papeleos y regularizaciones administrativas a futuro.
- Los reportes como las actas de calificaciones se realizan de manera manual, lo que conlleva cierta demora de tiempo y posibles fallas en la información de los estudiantes que se maneja.
- Debido a que la información se maneja en medios físicos, el proceso de revisión en registro y admisiones demora tiempo lo que genera que el proceso de matriculación de nuevos estudiantes sea más demoroso y poco ordenada.

Después de analizar los problemas anteriores, nos formulamos la siguiente pregunta:

¿Cómo se puede mejorar y optimizar el proceso de admisión estudiantil con la implementación de un sistema de admisión estudiantil para la gestión de las carreras en la Universidad Pública de El Alto?

1.4. OBJETIVOS

1.4.1. General

Implementar un sistema de admisión estudiantil para la gestión académica de las carreras de la Universidad Pública de El Alto, a fin de lograr una eficiente administración y gestión de la información para la Admisión estudiantil en sus diferentes modalidades.

1.4.2. Específicos

- Diseñar el sistema de manera que se facilite la administración y gestión de la información que se maneja dentro del proceso de admisión estudiantil de la Universidad Pública de El Alto.
- Facilitar y generar los reportes, actas de calificación de notas, información para para la certificación de habilitación.
- Optimizar el control del registro de postulantes a las modalidades, para no incurrir en observaciones según la norma.
- Interpolar el sistema académico de admisión estudiantil de carreras con el sistema de matriculación.
- Aplicar métricas de calidad para asegurar que la información se encuentre respaldada y segura.

1.5. JUSTIFICACIÓN

1.5.1. Técnica

Siendo que el sistema de admisión estudiantil está en línea solo es necesario una conexión a internet y un equipo, ya sea computadora personal, laptop o un celular para poder acceder al sistema de administración web.

La Universidad Pública de El Alto cuenta con servidores y equipos para poder alojar un sistema de administración web que coadyuve con el proceso de inscripción preuniversitaria, en las carreras de la universidad también se cuenta con el equipo necesario para poder acceder al sistema.

1.5.2. Económica

La implementación de un sistema de admisión estudiantil permite reducir costos en material de escritorio, así como una reducción de tiempo al tener la información centralizada y más organizada.

El sistema de admisión estudiantil está alojado en los servidores web con los que cuenta la Universidad, lo que permite que se pueda acceder a este sistema en cualquier equipo que tenga acceso a internet y que cuento con un navegador. Con esto no se requiere de la compra de equipos extra.

1.5.3. Social

La implementación del sistema de admisión estudiantil facilita el trabajo del personal involucrado dentro del proceso como los coordinadores, docentes de cursos preuniversitarios, facilita el control de la información que se tiene, logrando dar una mejor atención a los nuevos estudiantes y poder orientarlos mejor.

La población de nuevos estudiantes se ve beneficiada ya que reduce los tiempos de inscripción y podrán tener un mejor control de la información que entregan a la universidad.

1.6. METODOLOGÍA

1.6.1. Ingeniería de Software

La ingeniería de software es una disciplina de ingeniería que se interesa por todos los aspectos de la producción de software, desde las primeras etapas de la especificación del sistema hasta el mantenimiento del sistema después de que se pone en operación.

La ingeniería busca obtener resultados de la calidad requerida dentro de la fecha y del presupuesto.

El enfoque sistemático que se usa en la ingeniería de software se conoce en ocasiones como proceso de software. Un proceso de software es una secuencia de actividades que conducen a la elaboración de un producto de software. (Somerville, 2011)

1.6.2. Ingeniería de Requerimientos

Los requerimientos para un sistema son descripciones de lo que el sistema debe hacer: el servicio que ofrece y las restricciones en su operación. Tales requerimientos reflejan las necesidades de los clientes por un sistema que atienda cierto propósito, como sería controlar un dispositivo, colocar un pedido o buscar información. Al proceso de descubrir, analizar, documentar y verificar estos servicios y restricciones se le llama ingeniería de requerimientos (IR).

El término “requerimiento” no se usa de manera continua en la industria del software. En algunos casos, un requerimiento es simplemente un enunciado abstracto de alto nivel en un servicio que debe proporcionar un sistema, o bien, una restricción sobre un sistema. En el otro extremo, consiste en una definición detallada y formal de una función del sistema.

1.6.3. Ingeniería Web basada en UML (UWE)

La Ingeniería web basada en UML surgió a finales de los noventa (Baumeister et al., 1999; Wirsing et al., 1999) con la idea de encontrar una forma estándar para construir modelos de análisis y diseño de sistemas web basados en los métodos actuales OOHDM (Schwabe y Rossi, 1995), RMM (Isakowitz et al., 1995) y WSDM (de Troyer y Leune, 1998). El objetivo, que aún se persigue, fue utilizar un lenguaje común o al menos definir asignaciones basadas en metamodelos entre los enfoques existentes (Koch y Kraus, 2003; Escalona y Koch, 2006).

En ese momento el Lenguaje Unificado de Modelado (UML) que evolucionó de la integración de los tres enfoques de modelado diferentes de Booch, OOSE y OMT parecía ser un enfoque prometedor para el modelado de sistemas. Desde el principio de los esfuerzos de integración, el UML se convirtió en la "lengua franca" de (orientado a objetos) ingeniería de software (Object Management Group, 2005). Una característica destacada de UML es que proporciona un conjunto de ayudas para la definición de dominios específicos para los lenguajes de modelado (DSL) - también llamados mecanismos de extensión. Además, los DSL recién definidos siguen siendo

compatibles con UML, lo que permite el uso de todas las funciones complementadas, por ejemplo, con extensiones específicas de la Web.

Tanto la aceptación del UML como estándar en el desarrollo de software y sistemas, y la flexibilidad proporcionada por los mecanismos de extensión son las razones para la elección del lenguaje de modelado unificado en lugar del uso de técnicas de modelado patentadas. La idea seguida por UWE de adherirse a los estándares no se limita a UML. UWE también utiliza XML como formato de intercambio de modelos (con la esperanza de una futura interoperabilidad de herramientas habilitada por un XMI verdaderamente portátil), MOF para metamodelado, los principios impulsados por modelos dados por el modelo de OMG El enfoque de arquitectura impulsada (MDA), el lenguaje de transformación QVT y XML. (Koch, Knapp, Zhang, & Baumeister, 2008)

1.7. HERRAMIENTAS

1.7.1. Lenguaje de programación: PHP

PHP (PHP Hypertext Preprocesor) es uno de los lenguajes de programación más utilizados en la actualidad (se utiliza mayormente para el desarrollo de sitios web, pero para muchos es ya un lenguaje de propósito general). Esto se debe a diferentes factores entre los cuales podemos citar los siguientes: (Minera, 2010)

- Es libre y gratuito.
- Disponibilidad.
- Soporte para múltiples bases de datos
- Facilidad de aprendizaje.

1.7.2. Gestor de Base de Datos: MariaDB

MariaDB Server es uno de los servidores de bases de datos más populares del mundo. Está hecho por los desarrolladores originales de MySQL y está garantizado para ser de código abierto. Los usuarios notables incluyen Wikipedia, WordPress.com y Google.

MariaDB convierte los datos en información estructurada en una amplia gama de aplicaciones, desde bancos hasta sitios web. Es un reemplazo mejorado y directo para MySQL. MariaDB se usa porque es rápido, escalable y robusto, con un rico ecosistema de motores de almacenamiento, complementos y muchas otras herramientas que lo hacen muy versátil para una amplia variedad de casos de uso. (MariaDB, 2020)

1.7.3. Herramientas de desarrollo web

1.7.3.1. HTML 5

HTML5 (HyperText Markup Language) es la quinta revisión del lenguaje de marcado estándar que se emplea para la web. Es uno de los lenguajes de marcado más usados en todo el mundo y la razón es bastante obvia: gracias a HTML5 podemos crear la estructura de una página web. Texto, imágenes y material multimedia pueden mostrarse correctamente gracias a HTML5.

1.7.3.2. CSS

El nombre hojas de estilo en cascada viene del inglés Cascading Style Sheets, del que toma sus siglas. CSS es un lenguaje usado para definir la presentación de un documento estructurado escrito en HTML o XML (y por extensión en XHTML). El W3C (World Wide Web Consortium) es el encargado de formular la especificación de las hojas de estilo que servirán de estándar para los agentes de usuario o navegadores.

La idea que se encuentra detrás del desarrollo de CSS es separar la estructura de un documento de su presentación. (EcuRed c. , 2019)

1.7.3.3. JavaScript

JavaScript a menudo abreviado como JS, es un lenguaje de programación interpretado de alto nivel. Es un lenguaje que también se caracteriza como dinámico, débilmente tipado, basado en prototipos y multi-paradigma.

Junto con HTML y CSS, JavaScript es una de las tres tecnologías principales de la World Wide Web. JavaScript permite páginas web interactivas y, por lo tanto, es una parte esencial de las aplicaciones web. La gran mayoría de los sitios web lo usa, y

todos los principales navegadores web tienen un motor JavaScript dedicado para ejecutarlo. Disponible en: Wikipedia.org; 05/18; 17:30 Hrs].

1.8. LÍMITES Y ALCANCES

1.8.1. Limites

- El sistema de admisión se desarrolló específicamente dentro del marco del proceso de admisión estudiantil en sus diferentes modalidades en las carreras de la Universidad Pública de El Alto.
- El sistema de admisión estudiantil solo almacena la información necesaria dentro del proceso para la admisión estudiantil.
- El sistema de admisión estudiantil cuenta con niveles de administración, por lo que solo cierto tipo de usuario tendrá acceso a los diferentes módulos de información del sistema.

1.8.2. Alcances

El desarrollo del sistema de admisión estudiantil para poder organizar la información y alcanzar los objetivos planteados dispondrá de los siguientes módulos:

- Módulo de Admisión Estudiantil
- Módulo Administrativo
- Módulo Coordinador.
- Módulo Docente
- Módulo Estudiante
- Módulo de seguridad
- Módulo de difusión de información

1.9. APORTES

La realización de este proyecto aporta a la Universidad Pública de El Alto de un sistema que permite una mejor gestión de proceso de admisión estudiantil, mejora los tiempos de presentación de documentos y permite centralizar la información manejada dentro del proceso de admisión estudiantil.

Aporta una manera más ordenada y más eficiente de manejar la información a los coordinadores asignados en el proceso de admisión estudiantil.

Aporta a los postulantes con la opción de poder revisar las inscripciones de las carreras a las que postulan.

En este proyecto se plasmó el conocimiento adquirido en la carrera de Ingeniería de Sistemas de la Universidad Pública de El Alto.

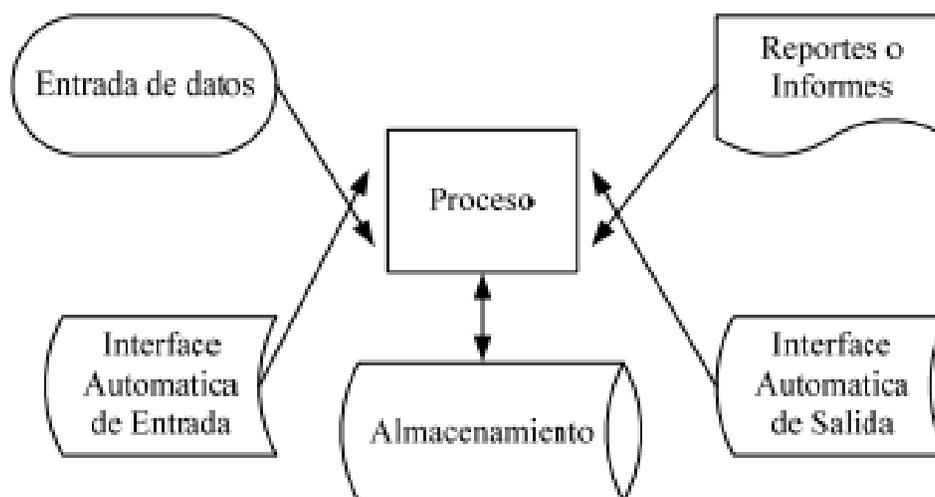
2. MARCO TEORICO

En este capítulo se explorarán los conceptos básicos que sirvieron como base para poder desarrollar el presente proyecto. Es necesario para empezar el desarrollo del sistema de admisión estudiantil profundizar el conocimiento de la metodología y las herramientas que se aplicaran dentro del proyecto.

2.1. SISTEMAS DE INFORMACIÓN

El sistema informático consiste en la compleja interconexión de numerosos componentes de hardware y software, los cuales son básicamente sistemas deterministas y formales, de tal forma que con un input determinado siempre se obtiene un mismo output. Los sistemas de información son sistemas sociales cuyo comportamiento se ve en gran medida influido por los objetivos, valores y creencias de individuos y grupos, así como por el desempeño de la tecnología. Así pues, el comportamiento del sistema de información no es determinista y no se ajusta a la representación de ningún modelo algorítmico formal. (Lapiedra, Devece, & Guiral, 2011)

Figura 2.1 Actividades realizadas en un sistema de información



Fuente: Francisco M. Gonzales-Longatt

Para el desarrollo de este proyecto se empleó el uso de tecnologías web, para poder llegar fácilmente a la población de bachilleres.

2.1.1. Web

La web es una interconexión de sistemas, hosts (anfitrión) y usuarios en una red. Tim Berners-Lee lo inventó en 1989. Tuvo la amabilidad de permitir que otros lo usaran libremente y no lo patentó. La web 1.0 es lo que tenemos cuando llegó a existencia.

En ese momento, era una maravilla tecnológica (aún lo es), pero con el tiempo la tecnología ha evolucionado.

El desarrollo de la World Wide Web tuvo un profundo efecto en todas nuestras vidas. En un inicio, la Web fue básicamente un almacén de información universal accesible que tuvo escaso efecto sobre los sistemas de software. Dichos sistemas corrían en computadoras locales y eran sólo accesibles desde el interior de una organización. Alrededor del año 2000, la Web comenzó a evolucionar, y a los navegadores se les agregaron cada vez más funcionalidades. Esto significó que los sistemas basados en la Web podían desarrollarse donde se tuviera acceso a dichos sistemas usando un navegador Web, en lugar de una interfaz de usuario de propósito específico. Esta situación condujo al desarrollo de una gran variedad de nuevos productos de sistemas que entregaban servicios innovadores, a los cuales se ingresaba desde la Web. A menudo los financiaban los anuncios publicitarios que se desplegaban en la pantalla del usuario y no requerían del pago directo de los usuarios. (EcuRed c. , 2019) (Somerville, 2011)

2.2. ADMISIÓN ESTUDIANTIL

Es el proceso al cual se somete el postulante para poder ingresar a la Universidad Pública de El Alto a cualquiera de las modalidades que existen dentro de la institución.

Este proceso se encuentra regido por normas e instructivos que los postulantes deberán cumplir para ingresar a esta casa de estudios.

Pasos a seguir

- Adquisición de formulario02
- Depósito bancario a la cuenta 10000004713083
- Presentación de los documentos que avalen el bachillerato.

- Presentación de documentos personales
- Evaluación de los postulantes mediante prueba de suficiencia académica o curso preuniversitario

2.3. GESTIÓN ACADÉMICA

La Gestión Académica es aquel ámbito de acción orientada a facilitar y mejorar los procesos formativos que imparten las instituciones de educación superior. Esto incluye la definición, implementación y seguimiento de procesos de mejoramiento continuo que les permitan responder y anteponerse a las necesidades formativas de sus estudiantes, a las exigencias del entorno, del mercado laboral y la política pública.

Destaca que mejorar la calidad de los procesos formativos en la educación superior "requiere de una revisión acuciosa de la gestión académica como un área en sí misma que promueva no sólo su profesionalización, sino que su desarrollo como área del conocimiento dentro del ámbito de la educación superior". (cnachile, 2018)

2.4. INTERPOLACIÓN

En el subcampo matemático del análisis numérico, se denomina interpolación a obtención de nuevos puntos partiendo del conocimiento de un conjunto de puntos.

En ingeniería y algunas ciencias es frecuente disponer de un cierto número de puntos obtenidos por muestreo o a partir de un experimento y pretender construir una función que los ajuste.

Otro problema estrechamente ligado con el de la interpolación es la aproximación de una función complicada por una más simple. Si tenemos una función cuyo cálculo resulta costoso, podemos partir de un cierto número de sus valores e interpolar dichos datos construyendo una función más simple. En general, por supuesto, no obtendremos los mismos valores evaluando la función obtenida que si evaluamos la función original, si bien dependiendo de las características del problema y del método de interpolación usado la ganancia en eficiencia puede compensar el error cometido.

Para la aplicación de este proyecto, entendemos como interpolación a los datos de referencia que nos servirán en la admisión estudiantil, como futuro referente para los

datos de la matriculación, donde se interpolarán los datos que se almacenen en el sistema de admisión estudiantil para facilitar el llenado de formularios en el sistema de matriculación.

2.5. INGENIERIA DE SOFTWARE

La ingeniería de software es una disciplina de ingeniería que se interesa por todos los aspectos de la producción de software, desde las primeras etapas de la especificación del sistema hasta el mantenimiento del sistema después de que se pone en operación. En esta definición se presentan dos frases clave:

1. Disciplina de ingeniería Los ingenieros hacen que las cosas funcionen. Aplican teorías, métodos y herramientas donde es adecuado. Sin embargo, los usan de manera selectiva y siempre tratan de encontrar soluciones a problemas, incluso cuando no hay teorías ni métodos aplicables. Los ingenieros también reconocen que deben trabajar ante restricciones organizacionales y financieras, de modo que buscan soluciones dentro de tales limitaciones.

2. Todos los aspectos de la producción del software La ingeniería de software no sólo se interesa por los procesos técnicos del desarrollo de software, sino también incluye actividades como la administración del proyecto de software y el desarrollo de herramientas, así como métodos y teorías para apoyar la producción de software. (Somerville, 2011)

2.6. METODOLOGÍA

2.6.1. Metodología UWE

UWE es un método de ingeniería del software para el desarrollo de aplicaciones web basado en UML. Cualquier tipo de diagrama UML puede ser usado, porque UWE es una extensión de UML.

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. La característica de UWE es el hecho de ser un enfoque basado en estándares que no se limita al uso de la UML "lingua franca", sino que también utiliza

XML como formato de intercambio de modelos, MOF¹ para el metamodelado, los principios basados en modelos del enfoque MDA, el lenguaje de transformación de modelos QVT² y XML.

Las principales razones para utilizar los mecanismos de extensión de la UML en lugar de una técnica de modelado patentada es la aceptación de la UML en el desarrollo de sistemas de software, la flexibilidad para la definición de un lenguaje de modelado específico de dominio web: un llamado perfil UML y un amplio soporte de modelado visual por parte de las herramientas existentes de UML CASE.

UWE utiliza tipos de notación UML "pura" y diagrama UML siempre que sea posible para el análisis y diseño de aplicaciones web, es decir, sin extensiones de ningún tipo. Para las características específicas de Web, como nodos y vínculos de la estructura de hipertexto, el perfil UWE incluye estereotipos, valores etiquetados y restricciones definidas para los elementos de modelado. La extensión UWE abarca aspectos de navegación, presentación, procesos de negocio y adaptación. La notación UWE se define como una extensión "ligera" de la UML. (Recuperado de uwe.pst.ifi.lmu.de, 21 de octubre de 2020).

2.6.1.1. Modelos de UWE

El método UWE consiste en la construcción de seis modelos de análisis y diseño. Dicha construcción se realiza dentro del marco de un proceso de diseño iterativo e incremental. Las actividades de modelado abarcan: el análisis de requerimientos, diseño conceptual, modelo de usuario, diseño de la navegación, de la presentación y diseño de la adaptación.

Los principales artefactos que produce el método de diseño de UWE son los siguientes: (Blanco, 2015)

¹ Meta-Object Facility (MOF) es un estándar del Object Management Group (OMG) sobre ingeniería dirigida por modelos. Su propósito es proporcionar un sistema de tipos para entidades en la arquitectura CORBA y un conjunto de interfaces a través de las cuales esos objetos se pueden crear y manipular.

² QVT (Consulta / Vista / Transformación) es un conjunto estándar de lenguajes para la transformación de modelos definidos por el Grupo de Gestión de Objetos.

2.6.1.1.1. Modelo de Requerimientos

El primer paso para el desarrollo de un sistema web que se especificará con UWE, es realizar la identificación de los requerimientos y plasmarlos en un modelo de requerimientos.

Los requerimientos pueden ser documentados en diferentes niveles de detalle, para este caso, UWE propone dos niveles de granularidad. En primera instancia se deben describir detalladamente las funcionalidades del sistema, las cuales son modeladas con casos de uso UML. Como segundo paso, se debe elaborar una descripción de los casos de uso más detallada, por ejemplo, realizando diagramas de actividad UML donde se delimiten las responsabilidades y acciones de los actores involucrados.

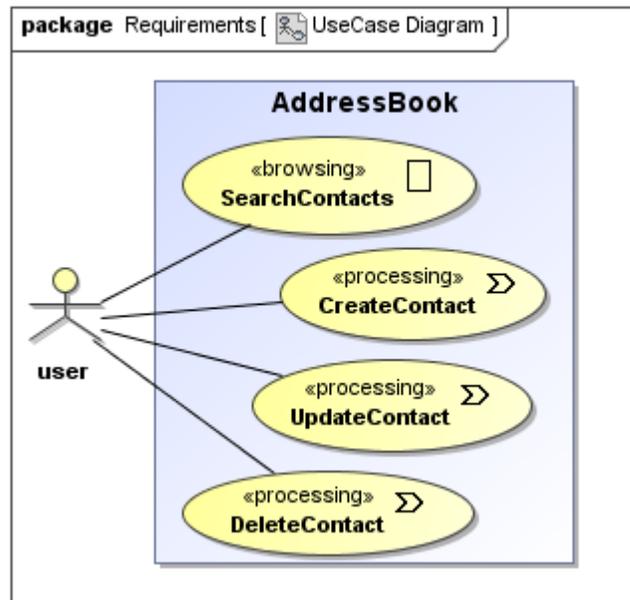
UWE distingue tres tipos de casos de uso: navegación, proceso, y casos de uso personalizados.

Los casos de uso de navegación, se distinguen con el estereotipo <<navigation>> (0) y se utilizan para modelar el comportamiento típico del usuario cuando interactúa con una aplicación web, tal como navegar a través del contenido de la WebApp o buscar información por medio de palabras claves.

Los casos de uso de proceso, se utilizan para describir las tareas del negocio que los usuarios finales realizarán con el sistema, tales como acciones transaccionales sobre la Base de Datos. No se denota con ningún estereotipo específico, por lo tanto, se utiliza en este caso la notación pura de UML.

Los casos de usos personalizados, implican la personalización de un sistema web, la cual es desencadenada por el comportamiento del usuario. Estos casos de uso se denotan con el estereotipo <<personalized>>

Figura 2.2 Nombres estereotipados y sus iconos



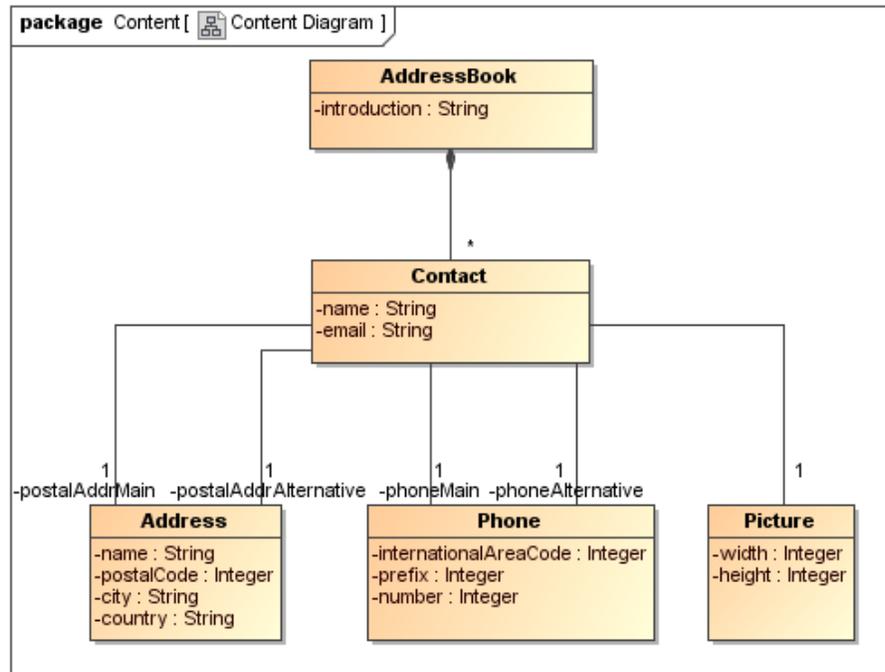
Fuente: Ludwing-Maximiliam-Universität München

2.6.1.1.2. Modelo de Contenido y Modelo de Usuario

El diseño conceptual se basa en el modelo de análisis e incluye los objetos involucrados en las actividades típicas que los usuarios realizan con la aplicación.

El propósito del modelo de contenido es proporcionar una especificación visual de la información relevante para el dominio del sistema web, que comprende principalmente el contenido de la aplicación Web.

Figura 2.3 Modelo de contenido



Fuente: Ludwig-Maximilian-Universität München

2.6.1.1.3. Modelo de Navegación

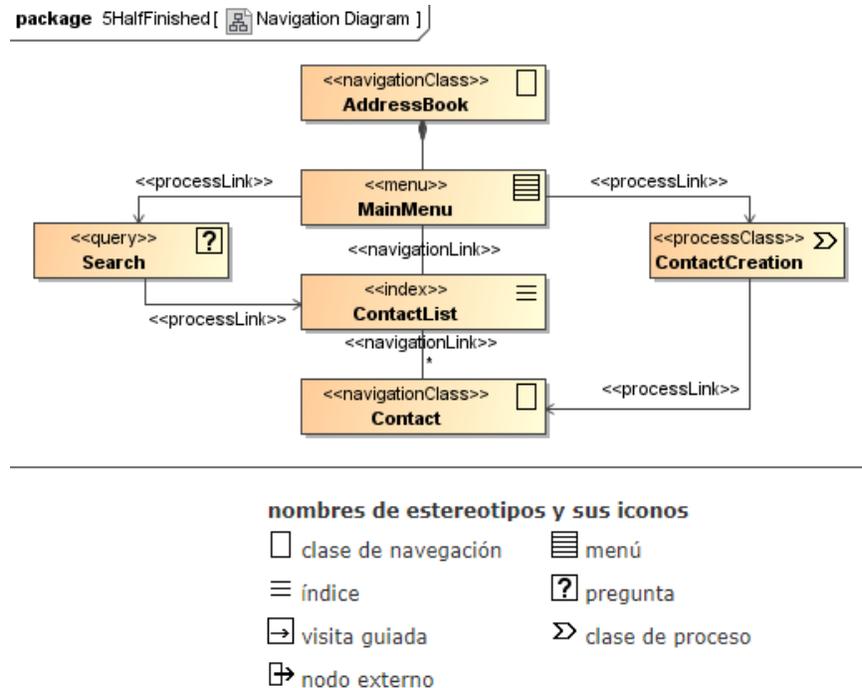
El modelo de estructura de navegación define la estructura de nodos y links de una WebApp mostrando cómo se puede realizar la navegación utilizando elementos de acceso tales como índices, visitas guiadas, consultas y menús.

Los elementos de modelado son:

- **Clases de navegación**, que se denotan con (0), representan los nodos navegables de la estructura de hipertexto.
- **Links de navegación**, que muestran el vínculo directo entre las clases de navegación.
- **Caminos de navegación alternativos**, los cuales son visualizados con el estereotipo <<menu>> ().
- **Primitivas de acceso**, las cuales se utilizan ya sea para llegar a múltiples instancias de una clase de navegación (<<index>> o <<guided tour>>) o para seleccionar ítems (<<query>>).

- **Clases de procesos** (), las cuales modelan los puntos de entrada y de salida de los procesos de negocio. Cada clase de proceso está asociada a un caso de uso de proceso.
- **Links de procesos**, que representan el vínculo entre las clases de proceso y de navegación.

Figura 2.4 Modelo de navegación



Fuente: Ludwig-Maximilian-Universität München

El modelo de estructura de navegación se representa mediante diagramas de clases UML estereotipados con las clases de navegación y procesos, menús y primitivas de acceso y así también los links de navegación y proceso.

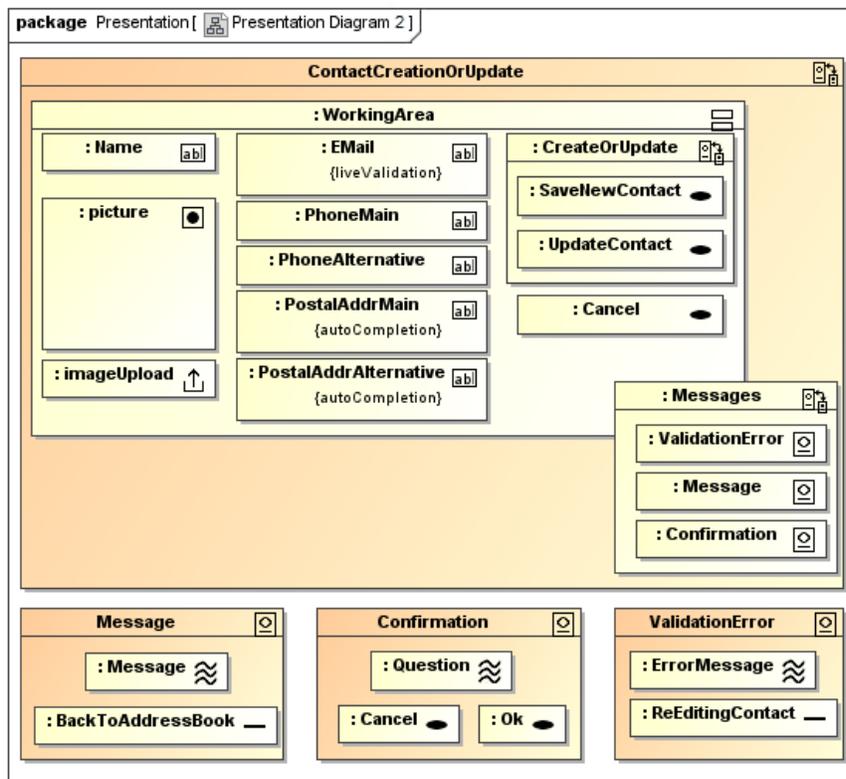
2.6.1.1.4. Modelo de Representación

El modelo de presentación proporciona una vista abstracta de la interfaz de usuario (UI) de la aplicación web. Se basa en el modelo de navegación y describe qué elementos (por ejemplo, texto, elementos, links, formularios) se utilizarán para presentar los nodos de navegación.

Los elementos básicos del modelo de presentación son:

- **Clases de presentación**, las cuales se basan directamente en los nodos del modelo de navegación. Una clase de presentación () está compuesta por elementos de UI tales como, texto (<<text>>), vinculo (<<anchor>>), botón (<<button>>), imagen (<<image>>), formulario (<<form>>), y colección de vínculos (<<anchored collection>>)
- **Páginas web** (<<page>>), que se utilizan para modelar la información proveniente de varios nodos de navegación y que se presentan en una misma página web.
- **Grupo de presentación** (<<presentation group>>), el cual es un contenedor de clases de presentación, y a su vez de otros grupos de presentación.

Figura 2.5 Modelo de presentación



Fuente: Ludwig-Maximilian-Universität München

2.7. HERRAMIENTAS

2.7.1. Gestor de Base de Datos MariaDB

MariaDB es un Sistema de gestión de bases de datos derivado de MySQL con licencia GPL (General Public License). Es desarrollado por Michael (Monty) Widenius (fundador de MySQL), la fundación MariaDB y la comunidad de desarrolladores de software libre. Introduce dos motores de almacenamiento nuevos, uno llamado Aria - que reemplaza con ventajas a MyISAM- y otro llamado XtraDB -en sustitución de InnoDB. Tiene una alta compatibilidad con MySQL ya que posee las mismas órdenes, interfaces, APIs y bibliotecas, siendo su objetivo poder cambiar un servidor por otro directamente. Este SGBD surge a raíz de la compra de Sun Microsystems -compañía que había comprado previamente MySQL AB - por parte de Oracle. MariaDB es un fork³ directo de MySQL que asegura la existencia de una versión de este producto con licencia GPL. Monty decidió crear esta variante porque estaba convencido de que el único interés de Oracle en MySQL era reducir la competencia que MySQL suponía para el mayor vendedor de bases de datos relacionales del mundo, que es Oracle. (EcuRed c. , 2017)

MariaDB es una bifurcación popular de MySQL creada por los desarrolladores originales de MySQL. Creció de preocupaciones relacionadas con la adquisición de MySQL por parte de Oracle. Ofrece soporte tanto para datos pequeños tareas de procesamiento y necesidades empresariales. Su objetivo es ser un reemplazo directo de MySQL requiriendo solo una simple desinstalación de MySQL y una instalación de MariaDB. MariaDB ofrece las mismas características de MySQL y mucho más. (Tutorials Point Ltd, 2016)

Las características importantes de MariaDB son:

- Todo MariaDB está bajo licencia GPL, LGPL o BSD.

³ En ingeniería de software se considera una bifurcación (en inglés fork) al desarrollo de un proyecto informático tomando como base un código fuente que ya existe o a la ramificación de un proyecto madre en varios proyectos que son independientes entre sí y que cuentan con objetivos o desarrolladores diferentes.

- MariaDB incluye una amplia selección de motores de almacenamiento, incluidos los de alto rendimiento motores de almacenamiento, para trabajar con otras fuentes de datos RDBMS.
- MariaDB utiliza un lenguaje de consulta estándar y popular.
- MariaDB se ejecuta en varios sistemas operativos y admite una amplia variedad de lenguajes de programación.
- MariaDB ofrece soporte para PHP, uno de los programas de desarrollo web más populares Idiomas
- MariaDB ofrece tecnología de clúster Galera.
- MariaDB también ofrece muchas operaciones y comandos no disponibles en MySQL, y elimina / reemplaza características que impactan negativamente en el desempeño.

2.7.2. Lenguaje de Programación PHP

PHP es un lenguaje de secuencia de comandos de servidor diseñado específicamente para la Web. Dentro de una página Web puede incrustar código PHP que se ejecutara cada vez que se visite una página. El código PHP es interpretado en el servidor Web y genera código HTML y otro contenido que el visitante verá.

PHP fue concebido en 1994 y es fruto del trabajo de un hombre, Rasmus Lerdorf. Ha sido adoptado por otras personas de talent0 y ha experimentado tres transformaciones importantes hasta convertirse en el producto actual. En octubre de 2002, era utilizado por más de nueve millones de dominios de todo el mundo y su número crece rápidamente. Si desea conocer el número actual de sitios que utilizan este lenguaje, visite el sitio <http://www.php.net/usage.php>. PHP es un producto de código abierto, lo que quiere decir que puede acceder a su código. Puede utilizarlo, modificarlo y redistribuirlo sin coste alguno.

Entre los competidores principales de PHP se puede citar a Perl, Microsoft Active Server Pages (ASP), Java Server Pages (JSP) y Allaire ColdFusion.

En comparación con estos productos, PHP cuenta con muchas ventajas, entre las que se encuentran las siguientes: (Welling & Thompson, 2004)

- Alto rendimiento
- Interfaces para una gran cantidad de sistemas de base de datos diferentes
- Bibliotecas incorporadas para muchas tareas Web habituales
- Bajo coste
- Facilidad de aprendizaje y uso
- Portabilidad
- Acceso al código abierto

El lenguaje PHP es un lenguaje de programación de estilo clásico, es decir, es un lenguaje de programación con variables, sentencias condicionales, ciclos (bucles), funciones.... No es un lenguaje de marcado como podría ser HTML, XML o WML. Está más cercano a JavaScript o a C, para aquellos que conocen estos lenguajes.

Al ser PHP un lenguaje que se ejecuta en el servidor no es necesario que su navegador lo soporte, es independiente del browser, pero sin embargo para que las páginas PHP funcionen, el servidor donde están alojadas debe soportar PHP. (Estrada, 2018)

2.7.3. Framework CodeIgniter

CodeIgniter es un marco de desarrollo de aplicaciones - un kit de herramientas - para las personas que construyen sitios web usando PHP. Su objetivo es permitirle desarrollar proyectos mucho más rápido de lo que podría si estuviera escribiendo código desde cero, proporcionando un amplio conjunto de bibliotecas para tareas comúnmente necesarias, así como una interfaz simple y una estructura lógica para acceder a estas bibliotecas. CodeIgniter le permite centrarse creativamente en su proyecto minimizando la cantidad de código necesario para una tarea determinada. (Instituto de Tecnología de Columbia Británica, 2019)

CodeIgniter es un entorno de desarrollo web escrito en PHP que presume de acelerar y optimizar el desarrollo de aplicaciones web gracias a un compacto diseño de software. La compañía de software norteamericana EllisLab fue la encargada de su creación y de la publicación de su primera versión en febrero de 2006. Un año después de anunciar, el 9 de julio de 2013, que la compañía ya no disponía de los recursos

necesarios para continuar desarrollando el software, el proyecto se vio beneficiado por su adquisición por el British Columbia Institute of Technology (BCIT).

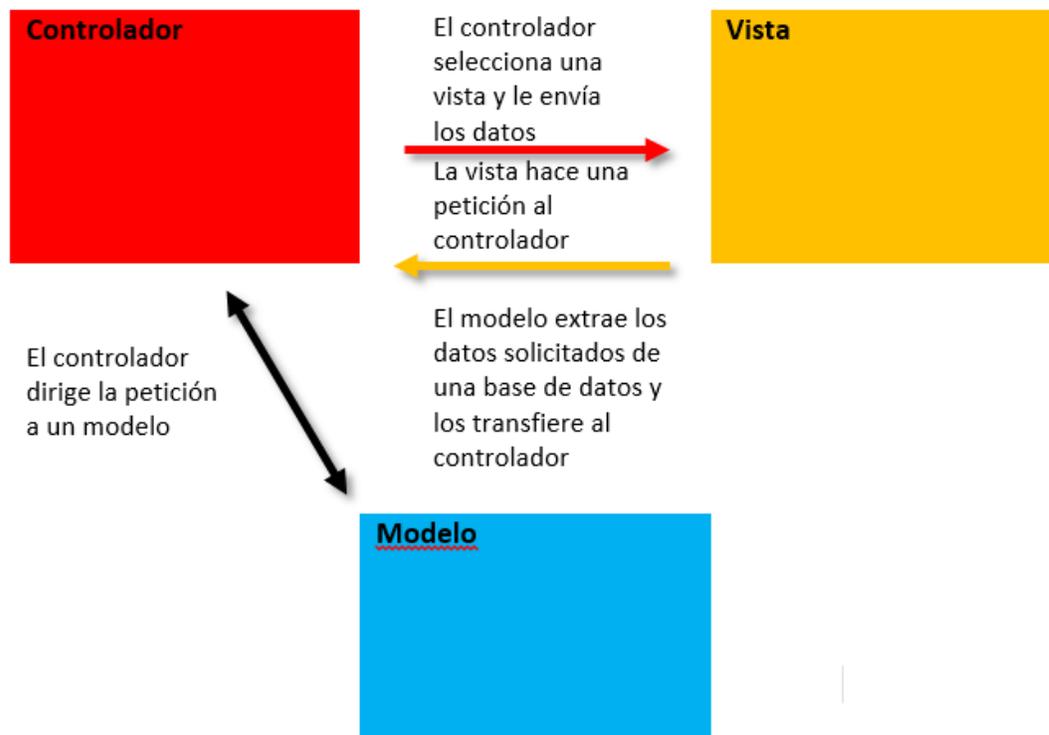
El código fuente de CodeIgniter es distribuido con una licencia MIT y puede descargarse desde la plataforma GitHub. (IONOS, 2020)

2.7.4. Estructura de CodeIgniter

El diseño orientado al rendimiento de este framework de desarrollo web se revela en su parca arquitectura, pues se basa en el patrón Modelo-Vista-Controlador (MVC). El principio fundamental que sustenta a la arquitectura de desarrollo MVC es la estricta separación entre el código y la presentación, gracias a una estructura modular de software y a la externalización del código PHP. Esta separación se realiza en estos tres grupos: el modelo (model), la vista (view) y el controlador (controller), que explicamos a continuación:

- **El modelo** representa la estructura de datos de una aplicación web desarrollada con CodeIgniter. Para ello, en el código fuente se definen las denominadas clases (“model classes”), que contienen funciones especiales con las cuales se puede recibir, insertar o actualizar la información de la base de datos.
- **La vista** es aquello que se le presenta al usuario final. Por lo general, se trata de un documento HTML en el cual se ha insertado contenido de forma dinámica con PHP, convirtiéndose en una especie de plantilla. CodeIgniter también permite definir fragmentos de una página web como la cabecera y el pie de página o páginas RSS como vista. Normalmente las aplicaciones web utilizan varias vistas, que toman su contenido desde el mismo modelo, de tal forma que es posible presentar diversas características del programa en vistas diferentes.
- **El controlador** media entre el modelo, la vista y cualquier otro recurso necesario para procesar una petición HTTP o generar una página web de forma dinámica. Este componente recibe las peticiones entrantes, valida la entrada, selecciona la vista deseada y le entrega el contenido que el modelo ha cargado desde una base de datos.

Figura 2.6 Interacción de los componentes modelo, vista y controlador de la arquitectura MVC



Fuente: IONOS (2020)

La estructura MVC permite diseñar software de forma flexible, ya que se pueden substituir, editar y reutilizar los módulos individuales de programación muy fácilmente. Los cambios que se realizan en un componente no suelen tener ningún efecto en el código fuente de otros componentes, siempre y cuando estos cambios no tengan lugar en los puntos de contacto entre unos y otros. (IONOS, 2020)

2.7.5. Lenguaje de Hipertexto HTML

HTML5 es la actualización de HTML, el lenguaje en el que es creada la web. HTML5 también es un término de marketing para agrupar las nuevas tecnologías de desarrollo de aplicaciones web: HTML5, CSS3 y nuevas capacidades de Javascript.

La versión anterior y más usada de HTML, HTML4, carece de características necesarias para la creación de aplicaciones modernas basadas en un navegador. El uso fuerte de Javascript ha ayudado a mejorar esto, gracias a frameworks como

jQuery,1 jQuery UI2, Sproutcore3, entre otros. Flash en especial ha sido usado en reemplazo de HTML para desarrollar web apps que superaran las habilidades de un navegador: Audio, video, webcams, micrófonos, datos binarios, animaciones vectoriales, componentes de interfaz complejos, entre muchas otras cosas. Ahora HTML5 es capaz de hacer esto sin necesidad de plugins y con una gran compatibilidad entre navegadores. (Vega & Vander Henst, 2011)

2.7.6. Bootstrap

El framework Bootstrap vio la luz en el año 2011. En un principio fue desarrollado por Twitter aunque posteriormente fue liberado bajo licencia MIT. Hoy en día continúa su desarrollo en un repositorio de GitHub. Se trata de un framework que ofrece la posibilidad de crear un sitio web totalmente responsive mediante el uso de librerías CSS. En estas librerías, nos podemos encontrar un gran número elementos ya desarrollados y listos para ser utilizados como pueden ser botones, menús, cuadros e incluso un amplio listado de tipografías.

Desde que vio la luz, Bootstrap se ha caracterizado por tratarse de una excelente herramienta para crear interfaces de usuarios limpias y totalmente adaptables a cualquier tipo de dispositivo y pantalla, independientemente de su tamaño.

Ya hemos dicho que Bootstrap se ha convertido en una de las herramientas más utilizadas hoy en día a la hora de realizar cualquier diseño web. Esto es debido a algunas de sus principales características, que lo han hecho muy atractivo a los ojos de los desarrolladores. (AscensTechnologies, 2016)

- Fácil e intuitivo
- Compatibles con todos los navegadores
- Optimizado para dispositivos móviles
- Amplia comunidad de desarrolladores tras el proyecto

2.7.7. JavaScript

JavaScript es un lenguaje de programación que le permite potenciar su HTML con animación, interactividad y efectos visuales dinámicos.

JavaScript puede hacer que las páginas web sean más útiles proporcionando comentarios inmediatos. Por ejemplo, una página de carrito de compras con JavaScript puede mostrar instantáneamente un costo total, con impuestos y envío, en el momento en que un visitante selecciona un producto para comprar. JavaScript puede producir un mensaje de error inmediatamente después de que alguien intente enviar un sitio web formulario en el que falta la información necesaria.

JavaScript también le permite crear interfaces divertidas, dinámicas e interactivas. Por ejemplo, con JavaScript, puede transformar una página estática de imágenes en miniatura en una presentación de diapositivas animada. O puede hacer algo más sutil como enviar más información en una página sin hacer que parezca abarrotada al organizar el contenido en pequeños paneles a los que los visitantes pueden acceder con un simple clic del mouse. O agregar algo útil y atractivo, como información emergente que proporciona información adicional información de los elementos de su página web. (McFarland, 2014)

2.7.8. jQuery

JavaScript tiene un pequeño secreto vergonzoso: escribirlo puede ser difícil. Si bien es más simple que muchos otros lenguajes de programación, JavaScript sigue siendo un lenguaje de programación. Y muchas personas, incluidos los diseñadores web, encuentran difícil programarlo.

Para complicar aún más las cosas, los diferentes navegadores web entienden JavaScript de manera diferente, por lo que un programa que funciona en, digamos, Chrome puede no responder por completo en Internet Explorer 9. Esta situación común puede costar muchas horas de prueba en diferentes máquinas y diferentes navegadores para asegurarse de que un programa funcione correctamente para toda la audiencia de su sitio.

Ahí es donde entra jQuery. JQuery es una librería de JavaScript destinada a hacer la programación JavaScript más fácil y divertida. Una librería de JavaScript es un complejo conjunto de código JavaScript que simplifica las tareas difíciles y resuelve los problemas de compatibilidad en navegadores. En otras palabras, jQuery resuelve los dos mayores problemas de JavaScript: complejidad y la naturaleza meticulosa de los diferentes navegadores web.

jQuery es el arma secreta de un diseñador web en la batalla de la programación JavaScript. Con jQuery, puede realizar tareas en una sola línea de código que podrían requerir cientos de líneas de programación y muchas horas de pruebas en el navegador para lograrlo con su propio código JavaScript. (McFarland, 2014)

2.8. METRICAS DE CALIDAD DE SOFTWARE

2.8.1. Calidad de Software

La calidad de un producto software depende del cumplimiento de un amplio listado de variables y de indicadores, que se deben tener en cuenta en las fases iniciales del ciclo de vida. Para alcanzarla es necesario darle mayor importancia a cuestiones como los requisitos no-funcionales, y a características como interoperabilidad, flexibilidad, usabilidad, confiabilidad, desempeño, escalabilidad, seguridad, etc. Entre los atributos más relevantes se encuentran: (Serna M., 2013)

- **Seguridad.** Debido a iniciativas como cero papel, mejoramiento ambiental, gestión documental, y la tendencia actual hacia la digitalización de la información de las empresas y la sociedad en general, y debido a que prácticamente todos los dispositivos electrónicos tienen conectividad, se vuelve sensible la información confidencial y protegida por marcos legales. Es por esto que este atributo toma importancia, y se hace imperante mejorar la seguridad y sus atributos de calidad, como la integridad, la privacidad y la confidencialidad, incrementado el nivel de la complejidad y la efectividad de los mecanismos software que prevengan las diferentes modalidades de ataque.
- **Disponibilidad y Confiabilidad.** La dependencia social, organizacional y de gobierno de los sistemas de información, exige el aseguramiento de una funcionalidad ininterrumpida de todas sus funciones, y cada vez son menos tolerables las interrupciones o fallas.
- **Usabilidad.** El acceso cada vez mayor a todo tipo de usuario hace que el desarrollo de software sea más exigente en el diseño de interfaz y factores relacionados con la usabilidad, y cada vez se va ir agudizando más este problema en el análisis y diseño, lo cual incrementará los costos.

2.8.2. Control de calidad y pruebas

Las razones por las que ocurren los errores en el software:

- **Especificaciones:** Al momento de redactar las especificaciones estas están incompletas, ambiguas, variables o simplemente no están realizadas.
- **Diseño:** El diseño de la solución está incompleto o inadecuado o bien las especificaciones no se comprendieron correctamente.
- **Codificación:** el código es incorrecto porque se hizo rápidamente, el programador no conoce bien el lenguaje o no comprendió bien el diseño.

La calidad es un concepto difícil de definir, toda vez que contiene múltiples facetas, en primer lugar según Pressman (2010), es una característica que se reconoce de inmediato aunque no pueda ser definida explícitamente, desde el punto de vista del usuario es un producto o servicio que satisface sus necesidades, para el fabricante es el cumplimiento de las especificaciones del producto fabricado o servicio prestado, la calidad también se encuentra relacionada con el precio que está dispuesto a pagar el comprador a cambio de un producto con tales características.

En el desarrollo de software, la calidad del diseño es el grado en el que éste cumple con las funciones y características especificadas en el modelo de requerimiento. La calidad de la conformidad se centra en el grado en el que la implementación se apega al diseño y en el que el sistema resultante cumple sus metas de requerimientos y desempeño.

La calidad del software, continua Pressman (2010), se presenta de la siguiente manera:

- Un proceso eficaz de software envuelve la calidad del proceso con el cual es llevado a cabo, envolviendo los aspectos de gestión y de buenas prácticas de ingeniería.
- En segundo lugar, se relaciona con la producción de un producto útil que cumpla con las expectativas del usuario de forma confiable y libre de errores.

- Añade valor agregado al productor y el usuario del producto, beneficiando a la organización que lo utiliza, lo produce y a la comunidad de usuarios finales.
- Así el producto final redunda en una serie de beneficios, menor esfuerzo en mantenimiento, menores errores, menos costo de producción, mayor rentabilidad y disponibilidad de información.

Señala Pressman (2010) los factores de calidad dictada por la norma ISO 9126, que sirven de guía de acción a la calidad:

- **Funcionalidad:** Es el grado en el que el software satisface las necesidades planteadas según su: adaptabilidad, exactitud, interoperatividad, cumplimiento y seguridad.
- **Confiabilidad:** Cantidad de tiempo que el software se encuentra disponible para su uso, según lo indican los atributos tales como: madurez, tolerancia a fallos y recuperación.
- **Usabilidad:** Grado en el que el software es fácil de usar, que sea entendible, aprendible y operable.
- **Eficiencia:** Grado en el que el software emplea óptimamente los recursos del sistema.
- **Facilidad de recibir mantenimiento:** Facilidad con la que pueden efectuarse reparaciones al software, que sea analizable, cambiable, estable y susceptible de someterse a pruebas.
- **Portabilidad:** Facilidad con la que el software puede llevarse de un ambiente a otro, cumpliendo con atributos de adaptabilidad, instalable, conformidad y sustituible.

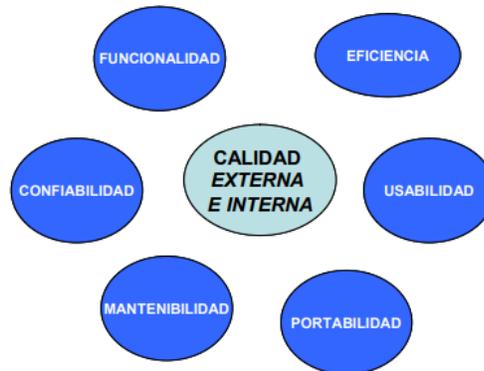
2.9. CALIDAD DEL SISTEMA

2.9.1. Norma ISO/IEC 9126

Permite especificar y evaluar la calidad del software desde diferentes criterios asociados con adquisición, requerimientos, desarrollo, uso, evaluación, soporte, mantenimiento, aseguramiento de la calidad y auditoria de software. Los modelos de calidad para el software se describen así:

Calidad interna y externa: Especifica 6 características para calidad interna y externa, las cuales, están subdivididas. Estas divisiones se manifiestan externamente cuando el software es usado como parte de un sistema Informático, y son el resultado de atributos internos de software.

Figura 2.7 Calidad interna y externa de la norma ISO 9126



Fuente: (García, 2005)

Calidad en uso: Calidad en uso es el efecto combinado para el usuario final de las 6 características de la calidad interna y externa del software. Especifica 4 características para la calidad en uso.

Al unir la calidad interna y externa con la calidad en uso se define un modelo de evaluación más completo, se puede pensar que la usabilidad del modelo de calidad externa e interna pueda ser igual al modelo de calidad en uso, pero no, la usabilidad es la forma como los profesionales interpretan o asimilan la funcionabilidad del software y la calidad en uso se puede asumir como la forma que lo asimila o maneja el usuario final. Si se unen los dos modelos, se puede definir que los seis indicadores del primer modelo tienen sus atributos y el modelo de calidad en uso sus 4 indicadores pasarían a tener sus atributos, mirándolo gráficamente quedaría así:

Figura 2.8 Funcionalidad ISO/IEC 9126



Fuente: (García, 2005)

Se establecen categorías para las cualidades de la calidad externa e interna y calidad en uso del software, teniendo en cuenta estos 7 indicadores (funcionalidad, confiabilidad, utilidad, eficiencia, capacidad de mantenimiento, portabilidad y calidad en uso), que se subdividen a su vez en varios indicadores; estas se pueden medir por métrica interna o externa.

Figura 2.9 Evaluación Interna, externa y Calidad de Uso ISO/IEC 9126



Fuente: (García, 2005)

Las definiciones se dan para cada característica y sub característica de calidad del software que influye en la calidad. Para cada característica y sub característica, la capacidad del software es determinada por un conjunto de atributos internos que pueden ser medidos. Las características y sub características se pueden medir externamente por la capacidad del sistema que contiene el software.

2.9.1.1. FUNCIONALIDAD

Funcionalidad es la capacidad del software de cumplir y proveer las funciones para satisfacer las necesidades explícitas e implícitas cuando es utilizado en condiciones específicas. A continuación, se muestra la característica de Funcionalidad y las sub características que cubre:

Formula:

$$\text{Funcionalidad} = \frac{\text{PF}}{\text{PF}_{\text{max}}}$$

La funcionalidad se divide en 5 criterios:

- **Adecuación:** La capacidad del software para proveer un adecuado conjunto de funciones que cumplan las tareas y objetivos especificados por el usuario.
- **Exactitud:** La capacidad del software para hacer procesos y entregar los resultados solicitados con precisión o de forma esperada.
- **Interoperabilidad:** La capacidad del software de interactuar con uno o más sistemas específicos.
- **Seguridad:** La capacidad del software para proteger la información y los datos de manera que los usuarios o los sistemas no autorizados no puedan acceder a ellos para realizar operaciones, y la capacidad de aceptar el acceso a los datos de los usuarios o sistemas autorizados
- **Conformidad de la funcionalidad:** La capacidad del software de cumplir los estándares referentes a la funcionalidad.

Figura 2.10 Característica de funcionalidad ISO/IEC 9126



Fuente: (García, 2005)

2.9.1.2. CONFIABILIDAD

La confiabilidad es la capacidad del software para asegurar un nivel de funcionamiento adecuado cuando es utilizado en condiciones específicas. En este caso a la confiabilidad se amplía sostener un nivel especificado de funcionamiento y no una función requerida.

Figura 2.11 Característica de Confiabilidad ISO/IEC 9126



Fuente: (GARCIA, 2005, pág. 16)

Formula:

$$F(t) = (\text{Funcionalidad}) * e^{-\lambda t}$$

La confiabilidad se divide en 4 criterios:

- **Madurez:** La capacidad que tiene el software para evitar fallas cuando encuentra errores. Ejemplo, la forma como el software advierte al usuario cuando realiza operaciones en la unidad de diskett vacia, o cuando no encuentra espacio suficiente el disco duro donde esta almacenando los datos.
- **Tolerancia a errores:** La capacidad que tiene el software para mantener un nivel de funcionamiento en caso de errores.
- **Recuperabilidad:** La capacidad que tiene el software para restablecer su funcionamiento adecuado y recuperar los datos afectados en el caso de una falla.
- **Conformidad de la fiabilidad:** La capacidad del software de cumplir a los estándares o normas relacionadas a la fiabilidad.

2.9.1.3. USABILIDAD

La usabilidad es la capacidad del software de ser entendido, aprendido, y usado en forma fácil y atractiva. Algunos criterios de funcionalidad, fiabilidad y eficiencia afectan la usabilidad, pero para los propósitos de la ISO/IEC 9126 ellos no clasifican como usabilidad. La usabilidad está determinada por los usuarios finales y los usuarios indirectos del software, dirigidos a todos los ambientes, a la preparación del uso y el resultado obtenido.

Figura 2.12 Característica de Usabilidad ISO/IEC 9126



Fuente: (García, 2005)

La usabilidad se divide en 5 criterios:

- **Entendimiento:** La capacidad que tiene el software para permitir al usuario entender si es adecuado, y de una manera fácil como ser utilizado para las tareas y las condiciones particulares de la aplicación. En este criterio se debe tener en cuenta la documentación y de las ayudas que el software entrega.
- **Aprendizaje:** La forma como el software permite al usuario aprender su uso. También es importante considerar la documentación.
- **Operabilidad:** La manera como el software permite al usuario operarlo y controlarlo.
- **Atracción:** La presentación del software debe ser atractiva al usuario. Esto se refiere a las cualidades del software para hacer más agradable al usuario, ejemplo, el diseño gráfico.
- **Conformidad de uso:** La capacidad del software de cumplir los estándares o normas relacionadas a su usabilidad.

2.9.1.4. EFICIENCIA

La eficiencia del software es la forma del desempeño adecuado, de acuerdo a al número recursos utilizados según las condiciones planteadas. Se debe tener en cuenta otros aspectos como la configuración de hardware, el sistema operativo, entre otros.

Figura 2.13 Característica de Eficiencia



Fuente: (García, 2005)

Formula:

$$\text{Eficiencia} = \frac{\sum_{i=0}^n i}{n} * \frac{100}{n}$$

La eficiencia se divide en 3 criterios:

- **Comportamiento de tiempos:** Los tiempos adecuados de respuesta y procesamiento, el rendimiento cuando realiza su función en condiciones específicas. Ejemplo, ejecutar el procedimiento más complejo del software y esperar su tiempo de respuesta, realizar la misma función, pero con más cantidad de registros.
- **Utilización de recursos:** La capacidad del software para utilizar cantidades y tipos adecuados de recursos cuando este funciona bajo requerimientos o condiciones establecidas. Ejemplo, los recursos humanos, el hardware, dispositivos externos.
- **Conformidad de eficiencia:** La capacidad que tiene el software para cumplir con los estándares o convenciones relacionados a la eficiencia.

2.9.1.5. CAPACIDAD DE MANTENIMIENTO

La capacidad de mantenimiento es la cualidad que tiene el software para ser modificado. Incluyendo correcciones o mejoras del software, a cambios en el entorno, y especificaciones de requerimientos funcionales.

Figura 2.14 Característica de Mantenimiento



Fuente: (García, 2005)

Formula:

$$IMS = \frac{Mt - (Fc + Fa + FE)}{Mt}$$

El mantenimiento se divide en 5 criterios:

- **Capacidad de ser analizado:** La forma como el software permite diagnósticos de deficiencias o causas de fallas, o la identificación de partes modificadas.
- **Cambiabilidad:** La capacidad del software para que la implementación de una modificación se pueda realizar, incluye también codificación, diseño y documentación de cambios.
- **Estabilidad:** La forma como el software evita efectos inesperados para modificaciones del mismo.
- **Facilidad de prueba:** La forma como el software permite realizar pruebas a las modificaciones sin poner el riesgo los datos.

- **Conformidad de facilidad de mantenimiento:** La capacidad que tiene el software para cumplir con los estándares de facilidad de mantenimiento.

2.9.1.6. PORTABILIDAD

La capacidad que tiene el software para ser trasladado de un entorno a otro.

Figura 2.15 Característica de portabilidad



Fuente: (García, 2005)

La usabilidad se divide en 5 criterios:

- **Adaptabilidad:** Es como el software se adapta a diferentes entornos especificados (hardware o sistemas operativos) sin que implique reacciones negativas ante el cambio. Incluye la escalabilidad de capacidad interna (Ejemplo: Campos en pantalla, tablas, volúmenes de transacciones, formatos de reporte, etc.).
- **Facilidad de instalación:** La facilidad del software para ser instalado en un entorno específico o por el usuario final.
- **Coexistencia:** La capacidad que tiene el software para coexistir con otro o varios softwares, la forma de compartir recursos comunes con otro software o dispositivo.
- **Reemplazabilidad:** La capacidad que tiene el software para ser reemplazado por otro software del mismo tipo, y para el mismo objetivo. Ejemplo, la reemplazabilidad de una nueva versión es importante para el usuario, la propiedad de poder migrar los datos a otro software de diferente proveedor.

- **Conformidad de portabilidad:** La capacidad que tiene el software para cumplir con los estándares relacionados a la portabilidad.

2.9.1.7. CALIDAD EN USO

Calidad en uso es la calidad del software que el usuario final refleja, la forma como el usuario final logra realizar los procesos con satisfacción, eficiencia y exactitud. La calidad en uso debe asegurar la prueba o revisión de todas las opciones que el usuario trabaja diariamente y los procesos que realiza esporádicamente relacionados con el mismo software.

Figura 2.16 Característica Calidad de uso



Fuente: (García, 2005)

La calidad de uso se divide en 4 criterios:

- **Eficacia:** La capacidad del software para permitir a los usuarios finales realizar los procesos con exactitud e integridad.
- **Productividad:** La forma como el software permite a los usuarios emplear cantidades apropiadas de recursos, en relación a la eficacia lograda en un contexto específico de uso. Para una empresa es muy importante que el software no afecte a la productividad del empleado
- **Seguridad:** Se refiere al que el Software no tenga niveles de riesgo para causar daño a las personas, instituciones, software, propiedad intelectual o entorno. Los riesgos son normalmente el resultado de deficiencias en la funcionalidad (Incluyendo seguridad), fiabilidad, usabilidad o facilidad de mantenimiento.

- **Satisfacción:** La satisfacción es la respuesta del usuario a la interacción con el software, e incluye las actitudes hacia el uso del mismo. A continuación, se describe un cuadro donde podemos resumir las características y cada uno de sus atributos, este cuadro le ayudara a visualizar el proceso de evaluación.

2.10. COSTOS

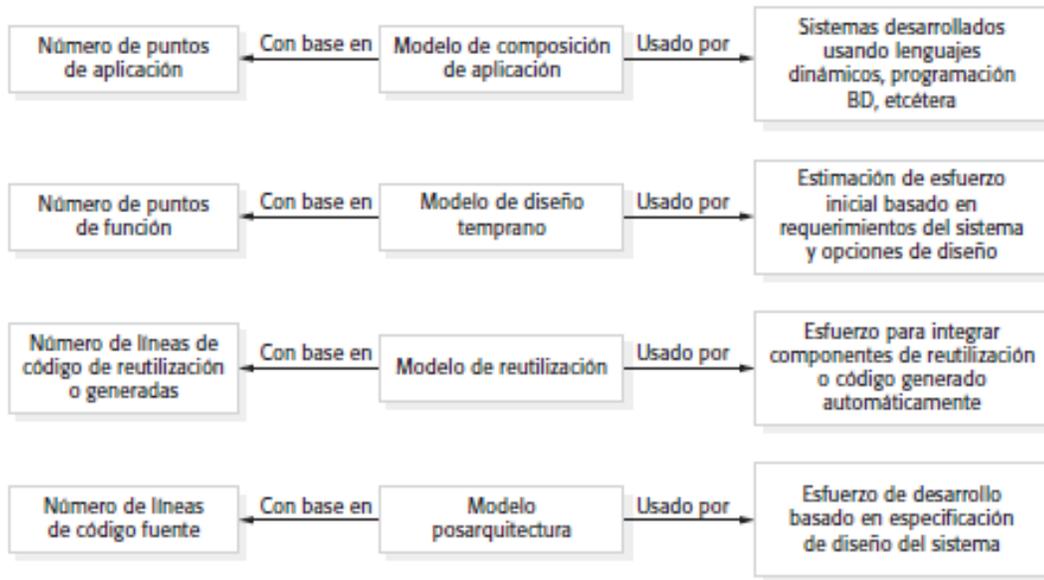
2.10.1. El Modelo COCOMO II

Éste es un modelo empírico que se derivó al recopilar datos a partir de un gran número de proyectos de software. Dichos datos se analizaron para descubrir qué fórmulas se ajustaban mejor con las observaciones. Dichas fórmulas vinculan el tamaño del sistema y los factores del producto, proyecto y equipo, con el esfuerzo para desarrollar el sistema. COCOMO II es un modelo de estimación bien documentado y no registrado.

COCOMO II se desarrolló a partir de los primeros modelos de estimación de costos COCOMO, que se basaron principalmente en el desarrollo de código original (Boehm, 1981; Boehm y Royce, 1989). El modelo COCOMO II toma en cuenta enfoques más modernos para el desarrollo de software, tales como el desarrollo rápido que usa lenguajes dinámicos, el desarrollo mediante composición de componentes y el uso de programación de base de datos. COCOMO II soporta el modelo en espiral de desarrollo, e incrusta submodelos que producen estimaciones cada vez más detalladas.

Los submodelos que se consideran parte del modelo COCOMO II son: (Somerville, 2011)

Figura 2.17 Modelos de estimación COCOMO



Fuente: (Somerville, Pág. 637)

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 [Boe00]. Como su predecesor, COCOMO II en realidad es una jerarquía de modelos de estimación que aborda las áreas siguientes: (Pressman, 2010)

- 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.
- Modelo de etapa postarquitectónica. Se usa durante la construcción del software.

2.11. SEGURIDAD DE LA INFORMACIÓN

2.11.1. SGSI (Sistema de Gestión de la Seguridad de la Información)

Toda la información almacenada y procesada por una organización está expuesta ante amenazas de ataque (por intereses comerciales, intelectuales y/o chantaje y extorsión), error (intencionado o por negligencia), ambientales (por ej. inundación o incendio), fallo en los sistemas (de almacenamiento de datos, informáticos, redes telemáticas), entre otras y también está sujeta a vulnerabilidades que representan puntos débiles inherentes a su propio uso en el ciclo de vida representado a continuación.

Permitir que una información precisa y completa esté disponible de manera oportuna para aquellos autorizados que tienen una necesidad es un catalizador para la eficiencia del negocio.

Para poder interrelacionar y coordinar las actividades de protección para la seguridad de la información, cada organización necesita establecer su propia política y objetivos para la seguridad de la información dentro de la coherencia del marco de globales de la organización.

Una vez fijados los objetivos en seguridad de la información necesitamos asegurar el modo de poder lograrlos eficazmente, en definitiva, un sistema de gestión de la seguridad de la información o SGSI en su forma abreviada.

Por tanto, un SGSI consiste en el conjunto de políticas, procedimientos y directrices junto a los recursos y actividades asociados que son administrados colectivamente por una organización, en la búsqueda de proteger sus activos de información esenciales.

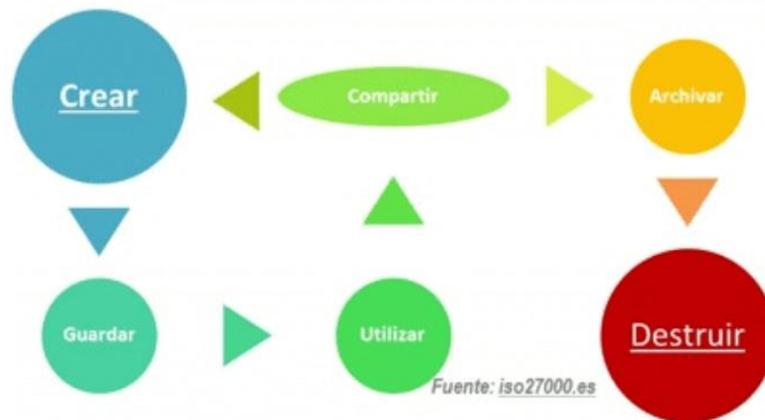
Un SGSI desde la visión del estándar internacional ISO/IEC 27001 es un enfoque sistemático para establecer, implementar, operar, monitorear, revisar, mantener y mejorar la seguridad de la información de una organización y lograr sus objetivos comerciales y/o de servicio (p.ej. en empresas públicas, organizaciones sin ánimo de lucro, etc.).

El alcance de un SGSI puede incluir, en función de dónde se identifiquen y ubiquen los activos de información esenciales, total o sólo una parte de la organización, funciones específicas e identificadas de la organización, secciones específicas e identificadas de la organización, o una o más funciones en un grupo de organizaciones.

El término seguridad de la información generalmente se basa en que la información se considera un activo que tiene un valor que requiere protección adecuada, por ejemplo, contra la pérdida de disponibilidad, confidencialidad e integridad.

- **Confidencialidad:** la información no se pone a disposición ni se revela a individuos, entidades o procesos no autorizados.
- **Integridad:** mantenimiento de la exactitud y completitud de la información y sus métodos de proceso.
- **Disponibilidad:** acceso y utilización de la información y los sistemas de tratamiento de la misma por parte de los individuos, entidades o procesos autorizados cuando lo requieran.

Figura 2.18 Ciclo de vida de un SGSI



Fuente: (iso27000.es)

Cada organización puede extender e integrar en un SGSI las tres características básicas iniciales de definición de la seguridad a otras adicionales como suelen ser la autenticidad, trazabilidad, no repudio, auditabilidad, etc. según se considere oportuno para cumplir con los requerimientos internos y/o externos aplicables en cada actividad. (ISO2700.es, 2005)

2.11.2. Aspectos de seguridad para los sistemas tradicionales y los basados en web

La seguridad de las instalaciones de cómputo, almacén de datos y la información generada es parte de una conversión exitosa. El reconocimiento de la necesidad de seguridad es una consecuencia natural de la creencia de que la información es un recurso organizacional importante. Con las transacciones cada vez más complejas y muchos intercambios innovadores, la Web ha producido un incremento en las preocupaciones de seguridad para el mundo profesional de SI. Es útil pensar en la seguridad de sistemas, datos e información en un continuo imaginario que va desde seguridad total hasta acceso abierto por completo. Aunque no hay tal cosa como un sistema totalmente seguro, las acciones de los analistas y usuarios pretenden mover los sistemas hacia el lado más seguro del espectro, disminuyendo la vulnerabilidad del sistema. Se debe observar que conforme más personas en la organización obtienen mayor potencia de cómputo, obtienen acceso a la Web, o se conectan a las intranets y extranets, la seguridad se vuelve cada vez más difícil y compleja.

La seguridad tiene tres aspectos interrelacionados: físico, lógico y conductual. Los tres deben trabajar juntos si deseamos que la calidad de seguridad permanezca alta.

Seguridad física

La seguridad física se refiere a proteger el sitio donde se encuentra la computadora, su equipo y software a través de medios físicos. Puede incluir acceso controlado a las salas de cómputo por medio de signos legibles por la máquina, sistemas biométricos o un registro de entrada y salida del sistema por un humano, así como el uso de cámaras de televisión de circuito cerrado para supervisar las áreas de cómputo, respaldando con frecuencia los datos y almacenando los respaldos en un área a prueba de fuego o de agua, a menudo en una ubicación remota segura. Además, el equipo de cómputo pequeño se debe asegurar para que un usuario típico no pueda moverlo y se debe garantizar el suministro ininterrumpido de energía eléctrica. Las alarmas que notifican a las personas apropiadas en caso de fuego, inundación o intrusión no autorizada de una persona deben estar en todo momento en funcionamiento activo.

Seguridad lógica

La seguridad lógica se refiere a los controles lógicos en el software. Los controles lógicos son conocidos por la mayoría de los usuarios como contraseñas o códigos de autorización de alguna clase. Cuando se usan, permiten al usuario entrar al sistema o a una parte específica de una base de datos con una contraseña correcta. Sin embargo, las contraseñas se manejan de manera descuidada en muchas organizaciones. Los empleados han escuchado por casualidad gritar una contraseña en las oficinas atestadas, grabar las contraseñas para sus pantallas y compartir las contraseñas personales con empleados autorizados que han olvidado las suyas.

Seguridad conductual

Las expectativas conductuales de una organización están implícitas en sus manuales de políticas e incluso en letreros anunciados en las salas de trabajo y los comedores. Sin embargo, la conducta que los miembros de la organización interiorizan también es crítica para el éxito de los esfuerzos de seguridad.

Parte del aspecto conductual de la seguridad es supervisar la conducta a intervalos irregulares para cerciorarse de que se están siguiendo los procedimientos apropiados y corregir cualesquier conductas que se podrían deteriorar con el tiempo. Hacer que el sistema registre el número de inicios de sesión fallidos de los usuarios es una forma de supervisar si usuarios no autorizados están intentando iniciar sesión en el sistema.

La salida generada por el sistema se debe reconocer por su potencial de poner a la organización en riesgo en algunas circunstancias. Los controles para la salida incluyen pantallas que sólo se pueden acceder mediante la contraseña, la clasificación de información (es decir, a quién se puede distribuir y cuándo) y el almacenamiento seguro de documentos impresos y almacenados, sin importar el formato. (Kendall & Kendall, 2011)

2.12. PRUEBAS DE SOFTWARE

2.12.1. Pruebas de caja negra

Estas pruebas permiten obtener un conjunto de condiciones de entrada que ejerciten completamente todos los requisitos funcionales de un programa. En ellas se ignora la

estructura de control, concentrándose en los requisitos funcionales del sistema y ejercitándolos.

La prueba de Caja Negra no es una alternativa a las técnicas de prueba de la Caja Blanca, sino un enfoque complementario que intenta descubrir diferentes tipos de errores a los encontrados en los métodos de la Caja Blanca.

Muchos autores consideran que estas pruebas permiten encontrar:

1. Funciones incorrectas o ausentes.
2. Errores de interfaz.
3. Errores en estructuras de datos o en accesos a las Bases de Datos externas.
4. Errores de rendimiento.
5. Errores de inicialización y terminación.

Para preparar los casos de pruebas hacen falta un número de datos que ayuden a la ejecución de los estos casos y que permitan que el sistema se ejecute en todas sus variantes, pueden ser datos válidos o inválidos para el programa según si lo que se desea es hallar un error o probar una funcionalidad. Los datos se escogen atendiendo a las especificaciones del problema, sin importar los detalles internos del programa, a fin de verificar que el programa corra bien.

Para desarrollar la prueba de caja negra existen varias técnicas, entre ellas están:

Técnica de la Partición de Equivalencia: esta técnica divide el campo de entrada en clases de datos que tienden a ejercitar determinadas funciones del software.

Técnica del Análisis de Valores Límites: esta Técnica prueba la habilidad del programa para manejar datos que se encuentran en los límites aceptables.

Técnica de Grafos de Causa-Efecto: es una técnica que permite al encargado de la prueba validar complejos conjuntos de acciones y condiciones.

Dentro del método de Caja Negra la técnica de la Partición de Equivalencia es una de las más efectivas pues permite examinar los valores válidos e inválidos de las entradas

existentes en el software, descubre de forma inmediata una clase de errores que, de otro modo, requerirían la ejecución de muchos casos antes de detectar el error genérico. La partición equivalente se dirige a la definición de casos de pruebas que descubran clases de errores, reduciendo así en número de clases de prueba que hay que desarrollar.

2.12.2. Pruebas de caja Blanca

La prueba de caja blanca se basa en el diseño de casos de prueba que usa la estructura de control del diseño procedimental para derivarlos. Mediante la prueba de la caja blanca el ingeniero del software puede obtener casos de prueba que:

- a) Garanticen que se ejerciten por lo menos una vez todos los caminos independientes de cada módulo, programa o método.
- b) Ejerciten todas las decisiones lógicas en las vertientes verdadera y falsa.
- c) Ejecuten todos los bucles en sus límites operacionales.
- d) Ejerciten las estructuras internas de datos para asegurar su validez.

Es por ello que se considera a la prueba de Caja Blanca como uno de los tipos de pruebas más importantes que se le aplican al software, logrando como resultado que disminuya en un gran porcentaje el número de errores existentes en los sistemas y por ende una mayor calidad y confiabilidad.

3. MARCO APLICATIVO

En el desarrollo del presente proyecto necesariamente se tiene que seguir y ejecutar el conjunto de actividades que se encuentra en el proceso de desarrollo que se ha optado por emplear.

En este capítulo se verán las técnicas y herramientas aplicadas para el desarrollo de sistema, siguiendo los pasos que emplea la metodología UWE.

3.1. ANÁLISIS DE REQUERIMIENTOS

Para el correcto desarrollo de esta fase se empleó el uso de herramientas provistas por la ingeniería de requerimientos para la captura de las necesidades a las que debe responder el sistema, empleándose las herramientas descritas en la tabla 3.1:

Tabla 3.1 Herramientas usadas para la ingeniería de requerimientos

Herramienta	Características
Observación	Se hizo la observación del proceso de la admisión de estudiantes así como las funciones que desempeñan cada uno de los actores involucrados en el proceso.
Documentación	Se revisaron los documentos que se manejan dentro del proceso, el tipo de presentación y los formatos para la realización de reportes.
Entrevistas	Se realizó entrevistas con el personal involucrado dentro del proceso, quienes son: <ul style="list-style-type: none">• Personal de registro y admisiones.• Coordinadores de preuniversitario las diferentes carreras.• Docentes y estudiantes.

Fuente: (Elaboración propia)

Después del análisis de requerimientos se empleó las herramientas que provee la metodología UWE que son el uso de los diagramas de casos de uso. Para esto fue necesaria la identificación de los actores que interactúan en el proceso

3.1.1. Actores

Para definir los actores que interactúan con el sistema se tomó en cuenta los roles que cumplen las diferentes personas y las entidades dentro de la Universidad que forman parte del proceso de admisión estudiantil, como resultado de ello se definieron los siguientes actores:

Tabla 3.2 Actores

ACTOR	DESCRIPCION
Administrador del sistema	<ul style="list-style-type: none">• Se encarga de gestionar los usuarios dentro del sistema• Se encarga de las configuraciones del sistema web• Realiza búsquedas y listados de los postulantes de todas las carreras•
Coordinadores de prefacultativos	<ul style="list-style-type: none">• Se encarga de listado de los postulantes a la carrera que coordina• Se encarga de la habilitación de estudiantes que presenten sus files• Se encarga del llenado de notas en la modalidad de admisión de prueba de suficiencia académica• Se encarga del control de llenado de notas de los docentes en la modalidad de admisión de curso preuniversitario• Se encarga del control de postulantes de las modalidades de admisión especial: Excelencia, Discapacidad y Profesionales.• Se encarga de la impresión de certificados de habilitación y reportes de actas de notas.

	<ul style="list-style-type: none"> • Se encarga de la habilitación de fechas de inicio y conclusión para la inscripción de postulantes, fechas de impresión de actas de notas y fechas de llenados de notas para la modalidad de curso preuniversitario.
Docentes de cursos prefacultativos	<ul style="list-style-type: none"> • Se encarga del llenado de notas de la materia a la que fue asignado en el curso preuniversitario. • Búsqueda de estudiantes inscritos en su materia.
Postulantes	<ul style="list-style-type: none"> • Registro en el sistema mediante el llenado e impresión del formulario 02 • Revisión de las carreras a las que está postulando • Revisión de los datos personales que ingresó al sistema

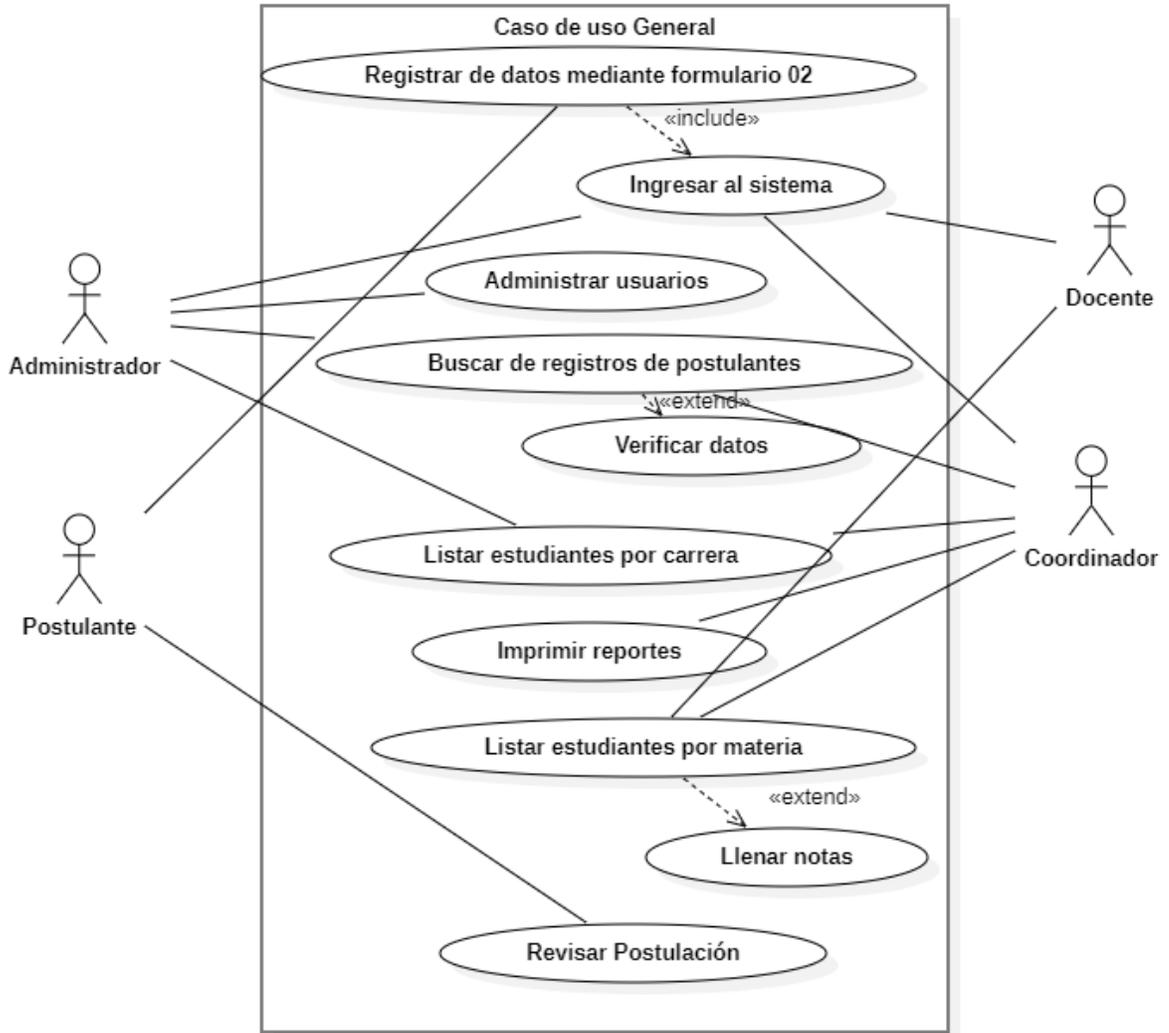
Fuente: (Elaboración Propia)

3.1.2. Casos de Uso

Para obtener un mejor entendimiento del proceso y llegar a un modelado correcto UWE utiliza los diagramas de casos de uso para describir las diferentes etapas dentro de un proceso.

Los casos de uso realizados para el desarrollo del sistema web son los siguientes:

Figura 3.1 Casos de uso general

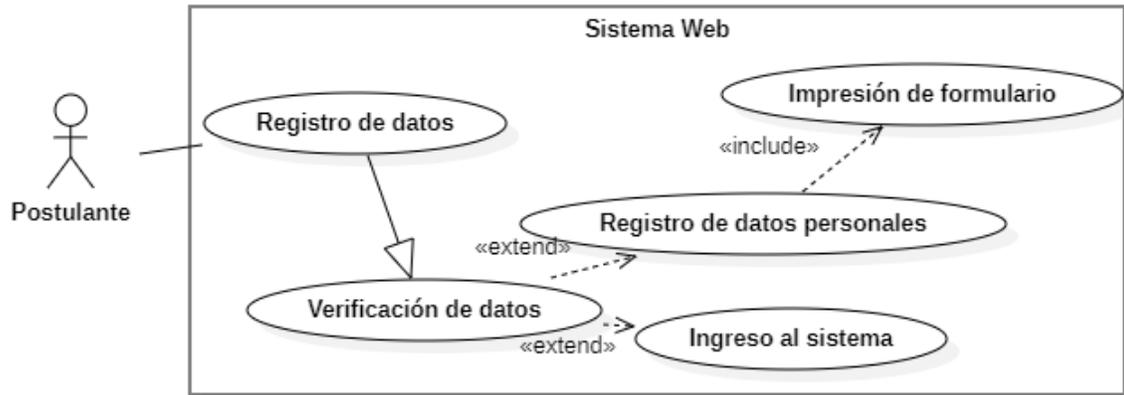


Fuente: (Elaboración propia)

3.1.2.1. Caso de uso para la admisión de postulantes

En este escenario se representa el proceso para la admisión de postulantes.

Figura 3.2 Caso de uso para la admisión de postulantes



Fuente: (Elaboración propia)

Tabla 3.3 Descripción del caso de uso admisión de postulantes

Caso de uso:	Administración de Usuarios
ID:	CU01
Descripción:	El postulante llena sus datos personales para registrar sus datos en el sistema.
Actores:	Postulante
Precondiciones:	El postulante debe haber ingresado a la pagina
Flujo de Eventos:	<p>El sistema solicita al postulante que ingrese su carnet, depósito bancario y fecha de nacimiento.</p> <p>El sistema valida que el postulante no se encuentre matriculado, sea profesional o se encuentre registrado en el sistema.</p> <p>El sistema muestra el formulario para el llenado de los datos personales.</p> <p>El sistema valida los datos del formulario, registra los datos y genera el formulario para su posterior impresión</p>
Postcondición:	El postulante se registra exitosamente

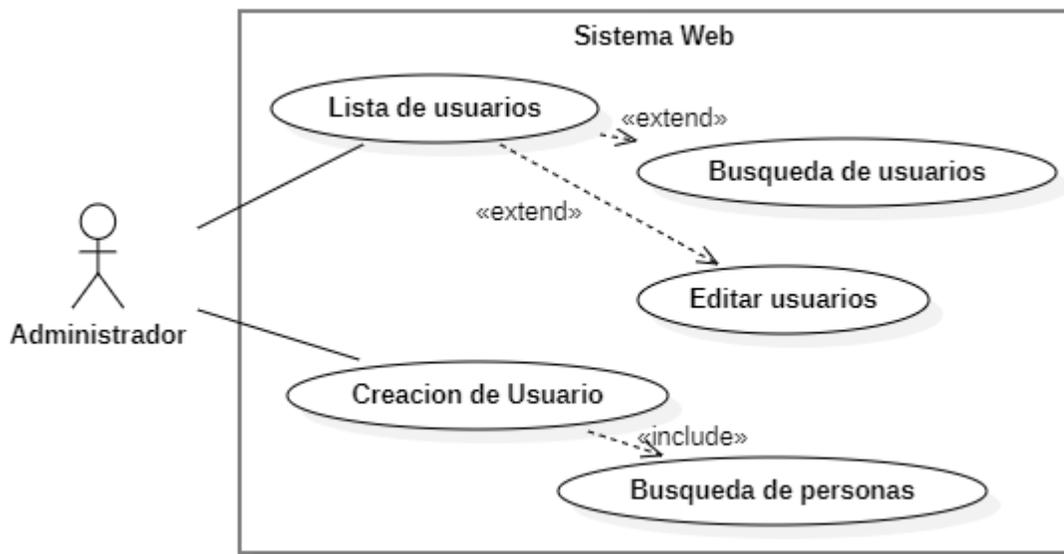
Excepciones	<ul style="list-style-type: none"> • Si el postulante se encuentra matriculado, el sistema muestra un mensaje de error y se cancela el caso de uso. • Si el postulante es profesional y se encuentra registrado en la universidad, el sistema muestra un mensaje de error y se cancela el caso de uso. • Si el usuario se encuentra previamente registrado, el sistema inicia sesión como postulante y le muestra las inscripciones que tiene activa
-------------	---

Fuente: (Elaboración propia)

3.1.2.2. Casos de uso para la administración de usuarios

En este escenario se representan todos los procesos para la administración de usuarios del lado del administrador.

Figura 3.3 Diagrama de caso de uso Administración de usuarios



Fuente: (Elaboración propia)

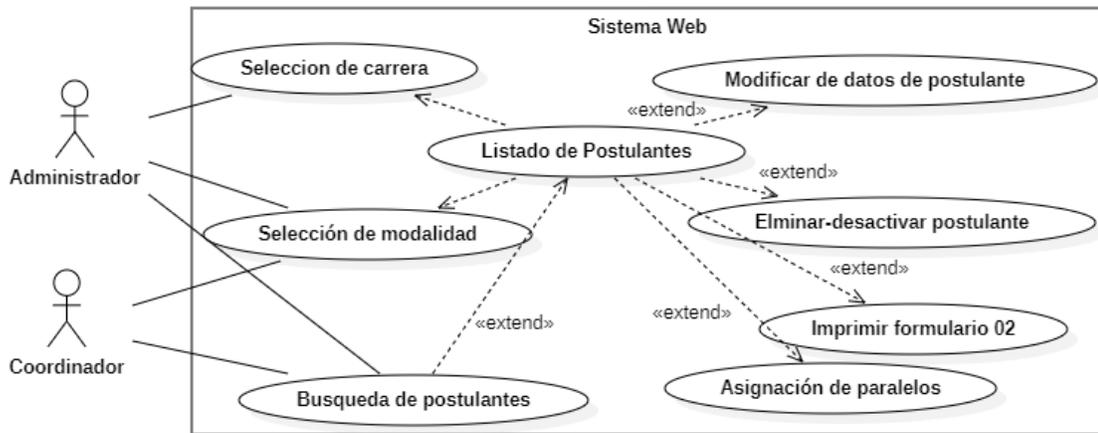
Tabla 3.4 Descripción de los casos de uso Administración de usuarios

Caso de uso:	Administración de Usuarios
ID:	CU02
Descripción:	El sistema permite gestionar los usuarios registrados en el mismo
Actores:	Administrador
Precondiciones:	El usuario debe haber iniciado sesión, el usuario debe ser administrador.
Flujo de Eventos:	<p>El usuario solicita el listado de usuarios al sistema.</p> <p>El sistema muestra todos los usuarios registrados.</p> <p>El usuario solicita crear, modificar o eliminar/desactivar un usuario.</p> <p>El sistema solicita la información para la creación, modificación y eliminación/desactivación de un usuario.</p> <p>El sistema valida los datos y realiza la acción solicitada.</p>
Postcondición:	El usuario logra crear, modificar o eliminar/desactivar usuarios.
Excepciones	Si el usuario trata de crear o modificar un usuario con campos clave que ya están registrados se cancela el caso de uso

Fuente: (Elaboración propia)

3.1.2.3. Caso de uso: Administración de postulantes

Figura 3.4 Caso de uso administración de postulantes



Fuente: (Elaboración propia)

Tabla 3.5 Descripción del caso de uso administración de postulantes

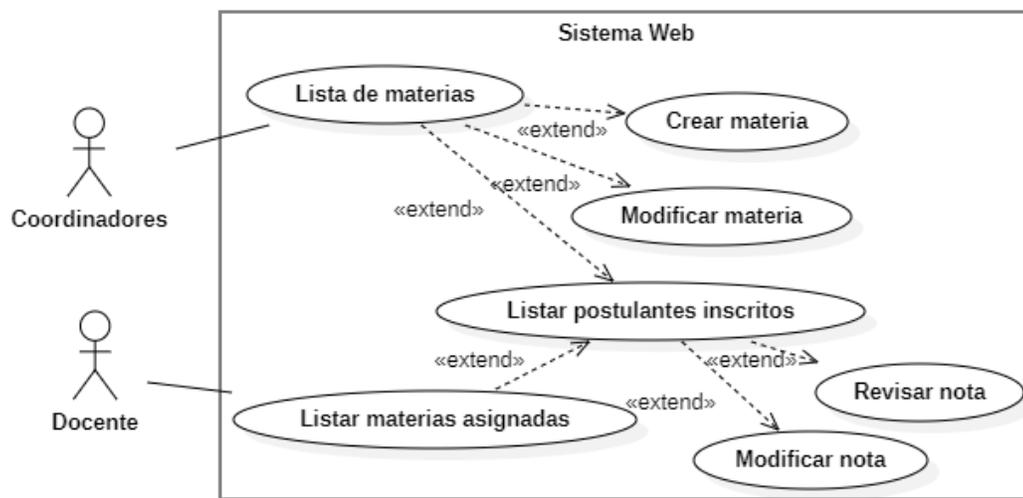
Caso de uso:	Administración de Postulantes
ID:	CU03
Breve descripción:	Describe el proceso para la gestión y administración de los postulantes
Actores:	Coordinador/Administrador
Precondiciones:	El usuario debe haber iniciado sesión, el usuario debe ser coordinador o administrador.
Flujo de Eventos:	<p>El usuario solicita al sistema el listado de los postulantes</p> <p>El sistema solicita que el usuario elija la carrera si es administrador, la modalidad y la sede.</p> <p>El sistema muestra los datos de los postulantes de las gestiones activas.</p> <p>El usuario solicita la modificación, eliminación de los registros de postulación.</p> <p>El sistema solicita los datos necesarios para la modificación o eliminación del registro.</p>

	<p>El usuario solicita hacer búsqueda de usuarios.</p> <p>El sistema le solicita los datos para realizar la búsqueda.</p> <p>El sistema muestra los registros que coinciden con los datos obtenidos.</p>
PostCondicion	El usuario habrá administrado los registros del postulante correctamente.
Excepciones	Si el usuario modifica un registro e ingresa datos clave ya existentes el sistema mostrara una alerta y cancelará el caso de uso.

Fuente: (Elaboración Propia)

3.1.2.4. Caso de uso: Gestión de la modalidad de curso preuniversitario

Figura 3.5 Gestión de la modalidad de curso preuniversitario



Fuente: (Elaboración propia)

Tabla 3.6 Descripción del caso de uso gestión de modalidad de curso preuniversitario

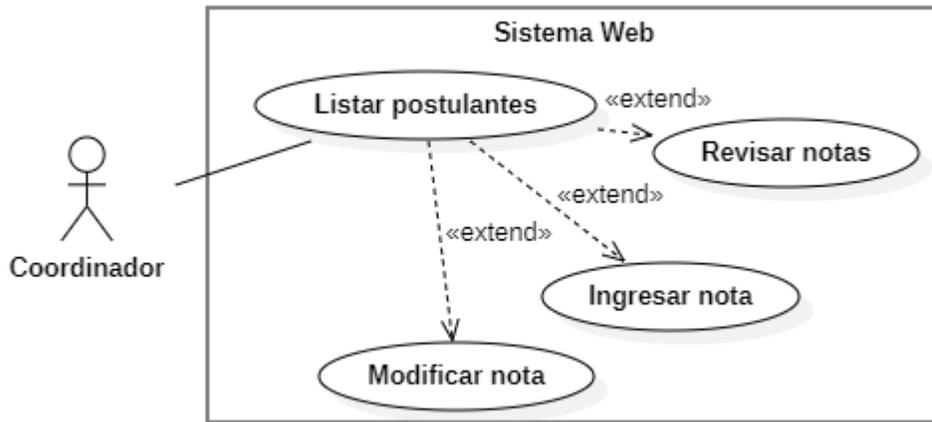
Caso de uso:	Gestión de la modalidad de curso preuniversitario
ID:	CU04

Breve descripción:	Describe el proceso empleado para la gestión del curso preuniversitario
Actores:	Coordinador/Docente
Precondiciones:	El usuario debe haber iniciado sesión, el usuario debe ser coordinador o docente.
Flujo de Eventos:	<p>El usuario coordinador solicita al sistema el listado de cursos, el docente solicita el listado de materias al que fue asignado.</p> <p>El sistema muestra la lista de materias de la gestión actual. Si el usuario es coordinador el usuario le mostrar las opciones de crear, modificar y listar postulantes. Si el usuario es docente el sistema le mostrar la opción de listar postulantes e insertar notas.</p> <p>El usuario coordinador crea o modifica cursos.</p> <p>El sistema solicita los datos necesarios para la acción, realiza la validación y efectúa la creación o modificación.</p> <p>El usuario coordinador o docente solicita al sistema el listado de estudiantes.</p> <p>El sistema muestra la lista de estudiantes y le da la opción de insertar/modificar notas.</p> <p>El usuario coordinador/docente ingresa las notas.</p> <p>El sistema hace la validación de las notas y las guarda automáticamente.</p>
Postcondicion	El usuario habrá gestionado las materias del curso preuniversitario.
Excepciones	Si los docentes no han sido asignados, la materia se creara con el campo de docente vacío.

Fuente: (Elaboración propia)

3.1.2.5. Caso de uso: Gestión de la modalidad de prueba de suficiencia académica

Figura 3.6 Caso de uso de la gestión de la modalidad de prueba de suficiencia académica



Fuente: (Elaboración propia)

Tabla 3.7 Descripción del caso de uso de la modalidad de prueba de suficiencia académica

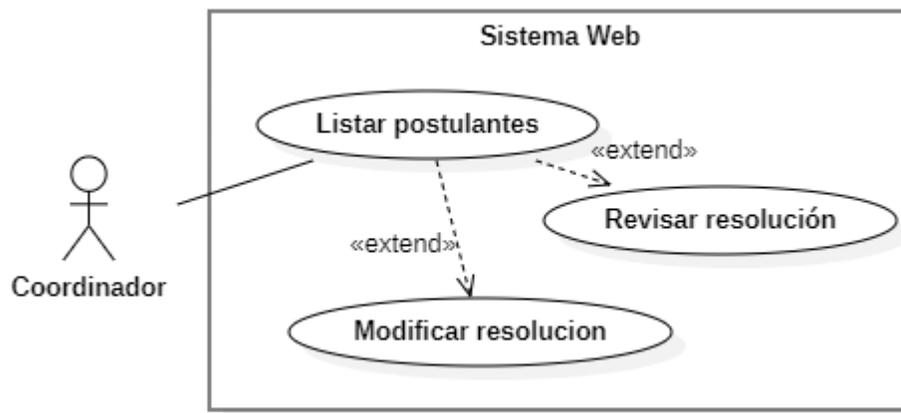
Caso de uso:	Gestión de la modalidad de prueba de suficiencia académica
ID:	CU05
Breve descripción:	Describe el proceso empleado para la gestión de la modalidad de admisión de prueba de suficiencia académica.
Actores:	Coordinador
Precondiciones:	El usuario debe haber iniciado sesión, el usuario debe ser coordinador.
Flujo de Eventos:	El usuario solicita el listado de los postulantes inscritos a la modalidad de prueba de suficiencia académica. El sistema muestra la lista de los postulantes inscritos en la gestión actual y le da la opción de llenar las notas correspondientes.

	El usuario modifica o ingresa la nota. El sistema valida la nota ingresada y la guarda automáticamente la nota.
Postcondicion	El usuario habrá gestionado las notas de la modalidad de prueba de suficiencia académica.
Excepciones	Si la nota ingresada no corresponde a rango establecido (0 - 100) el sistema mostrará una alerta y le pedirá ingresar una nota correcta al usuario.

Fuente: (Elaboración propia)

3.1.2.6. Caso de uso: Gestión de la modalidad de admisión especial

Figura 3.7 Caso de uso gestión de la modalidad especial



Fuente: (Elaboración propia)

Tabla 3.8 Descripción del caso de uso gestión de la modalidad especial

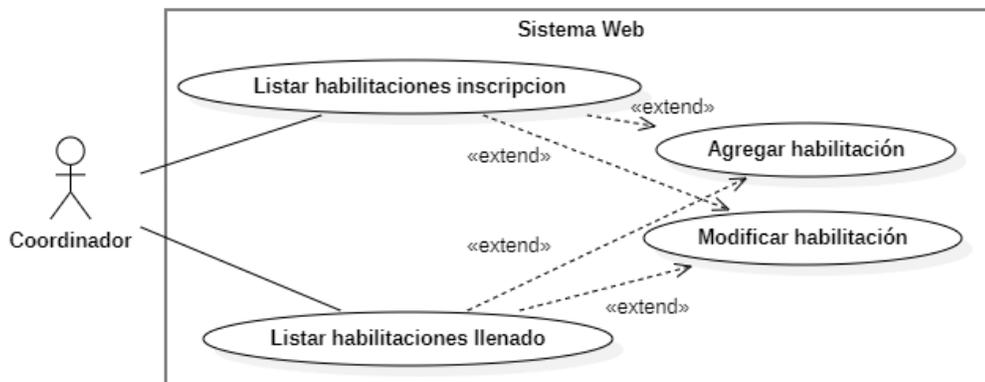
Caso de uso:	Gestión de la modalidad de admisión especial
ID:	CU06
Breve descripción:	Describe el proceso empleado para la gestión de la modalidad de admisión especial ya sea excelencia académica, discapacidad o profesionales.
Actores:	Coordinador
Precondiciones:	El usuario debe haber iniciado sesión, el usuario debe ser coordinador.

Flujo de Eventos:	<p>El usuario solicita el listado de los postulantes inscritos a la modalidad de una de las modalidades de admisión especial</p> <p>El sistema la mostrará el listado de los postulantes a la modalidad de admisión especial y le permitirá gestionar las resoluciones de vicerrectorado.</p> <p>El usuario solicita modificar la resolución de vicerrectorado.</p> <p>El sistema valida y verifica la resolución y realiza la modificación.</p>
Postcondición	El usuario habrá gestionado y revisado las resoluciones de vicerrectorado para la modalidad de admisión especial.

Fuente: (Elaboración propia)

3.1.2.7. Caso de uso: Habilitación de fechas

Figura 3.8 Caso de uso Habilitación de fechas



Fuente: (Elaboración propia)

Tabla 3.9 Descripción del caso de uso Habilitación de fechas

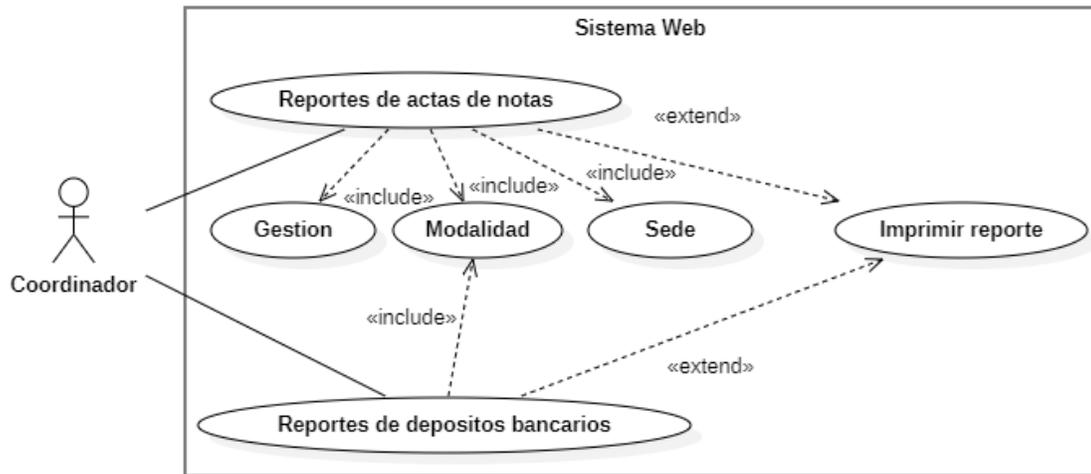
Caso de uso:	Gestión de la modalidad de admisión especial
ID:	CU07
Breve descripción:	Describe el proceso empleado para la habilitación de fechas de apertura y finalización de inscripciones y la habilitación de las fechas para llenado de notas.

Actores:	Coordinador
Precondiciones:	El usuario debe haber iniciado sesión, el usuario debe ser coordinador.
Flujo de Eventos:	El usuario solicita el listado de las habilitaciones. El sistema muestra una lista de las habilitaciones pertenecientes a la actual gestión. El usuario crea/modifica una habilitación. El sistema solicita los datos para la creación/modificación de la habilitación, valida los datos y efectúa la operación.
Postcondición	El usuario habrá habilitado las fechas para inscripciones y llenado de notas
Excepciones	Si el usuario llena datos que ya existen para una gestión y modalidad, el sistema mostrará un mensaje de error y no efectuará la creación/modificación.

Fuente: (Elaboración propia)

3.1.2.8. Caso de uso: Gestión de reportes

Figura 3.9 Caso de uso Gestión de reportes



Fuente: (Elaboración propia)

Tabla 3.10 Descripción del caso de uso Gestión de reportes

Caso de uso:	Gestión de reportes
ID:	CU08
Breve descripción:	Describe el proceso empleado para la impresión de reportes
Actores:	Coordinador
Precondiciones:	El usuario debe haber iniciado sesión, el usuario debe ser coordinador, deben haberse cargado las notas en el sistema
Flujo de Eventos:	<p>El usuario solicita la impresión la reportes.</p> <p>El sistema solicita que se selecciones la gestión, modalidad y sede para la impresión de reportes de actas de notas y solicita la selección de una modalidad para la impresión de reportes de depósitos bancarios.</p> <p>El usuario elije las opciones que se adecuen al reporte que necesita.</p> <p>El sistema genera el reporte en formato pdf listo para su descarga.</p>
Postcondición	El usuario habrá generado los reportes para su presentación.
Excepciones	Si el usuario trata de imprimir un reporte y no se cargaron las notas el sistema generara un reporte vacío.

Fuente: (Elaboración propia)

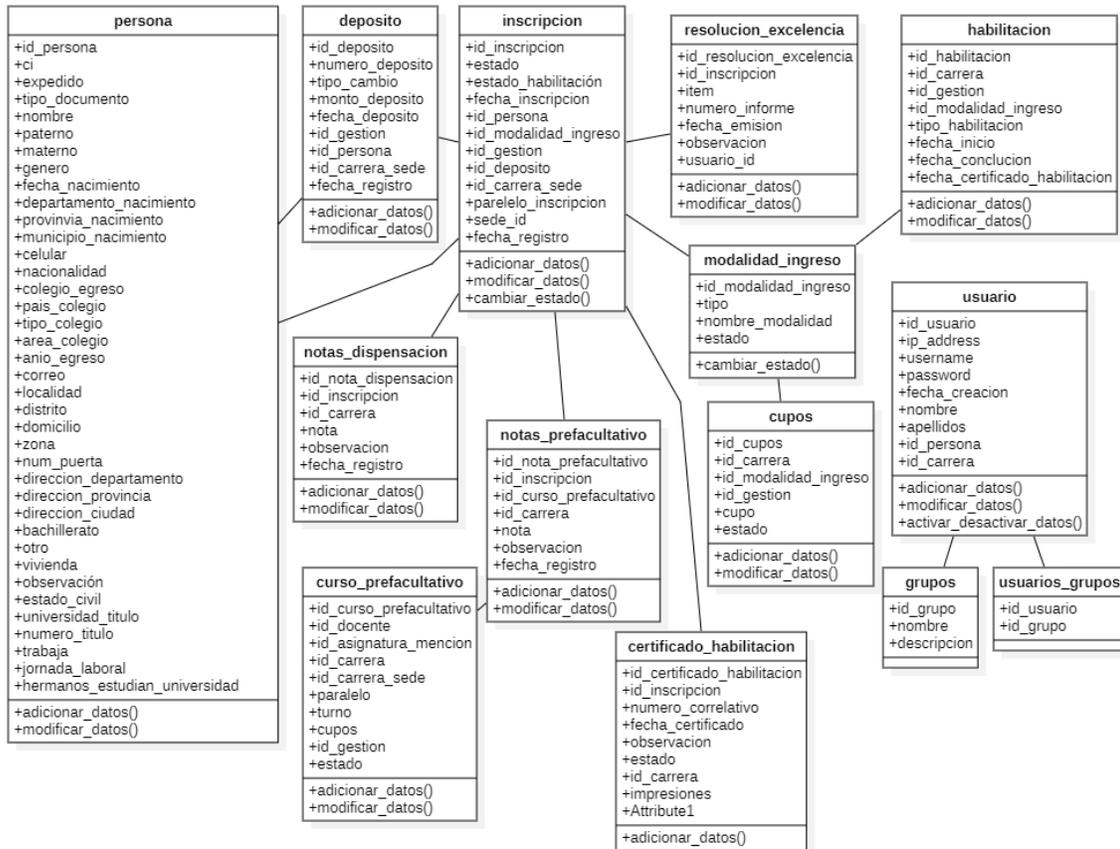
3.2. MODELO DE CONTENIDO

Esta fase de UWE se basa en la herramienta proporcionada por UML que son los diagramas de clases, que describen los datos que se manejaran dentro del desarrollo del sistema web.

3.2.1. Diagrama de clases

Diagrama de Clases para el sistema de admisión

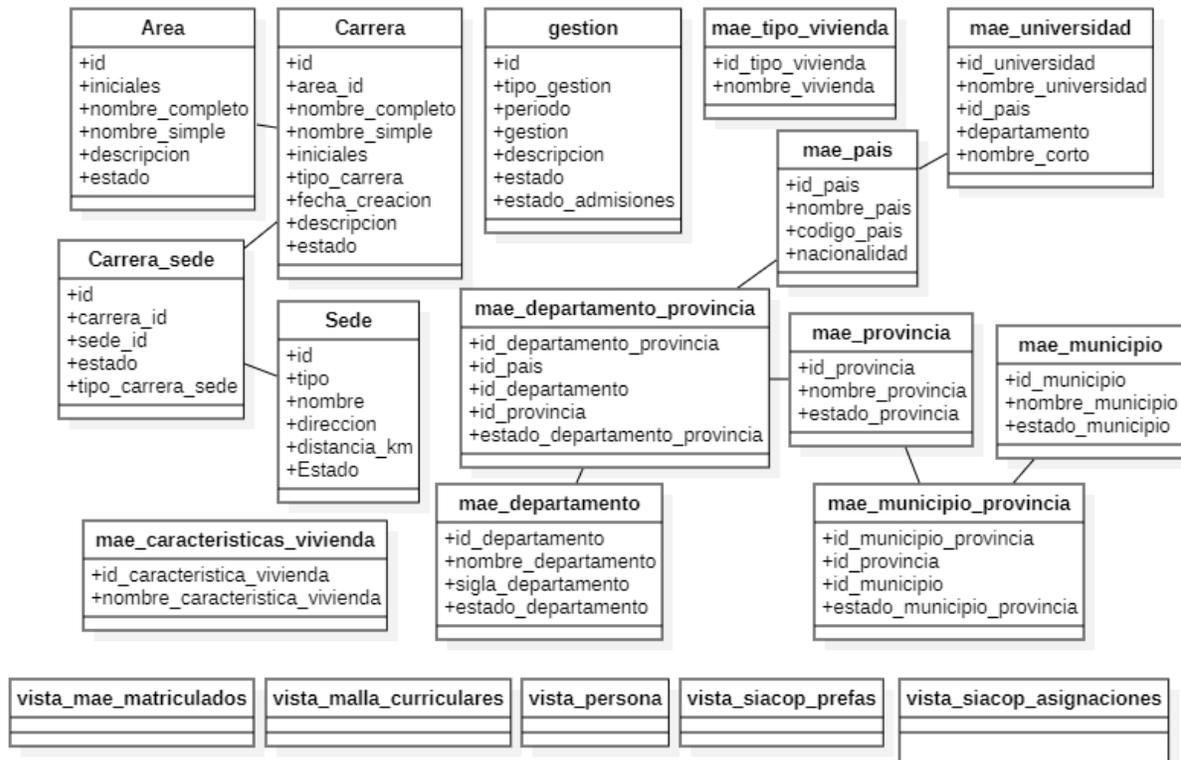
Figura 3.10 Diagrama de clases sistema de admisión estudiantil



Fuente: (Elaboración propia)

Diagrama de clases para la base de datos Base_upea

Figura 3.11 Diagrama de clases Base_upea

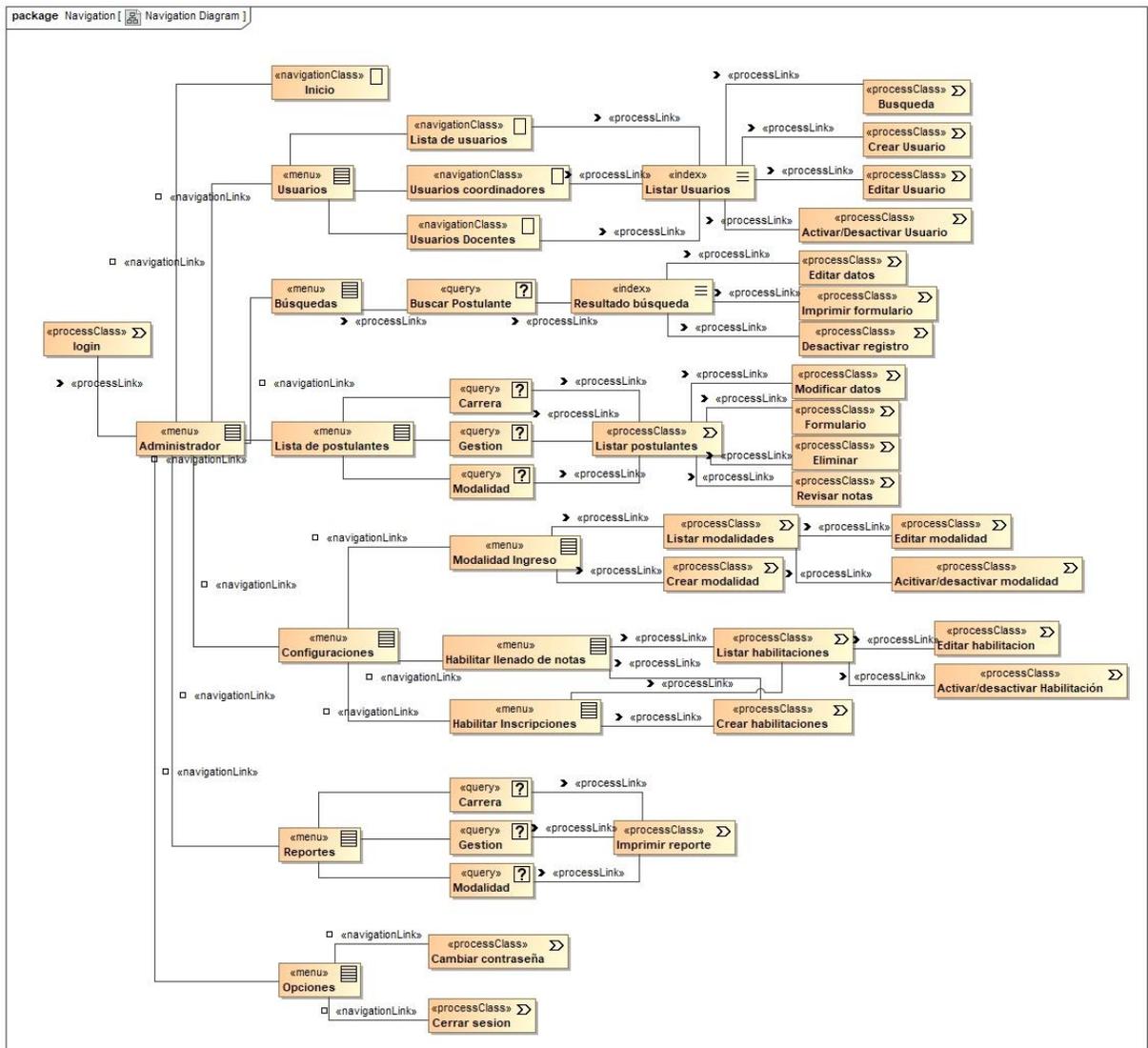


Fuente: (Elaboración propia)

Para la realización del proyecto se emplearon las tablas de la base de datos de la universidad, que se usaron para diseñar el formulario de admisión estudiantil a fin de interpolar la información del sistema de admisión estudiantil con el sistema de matriculación.

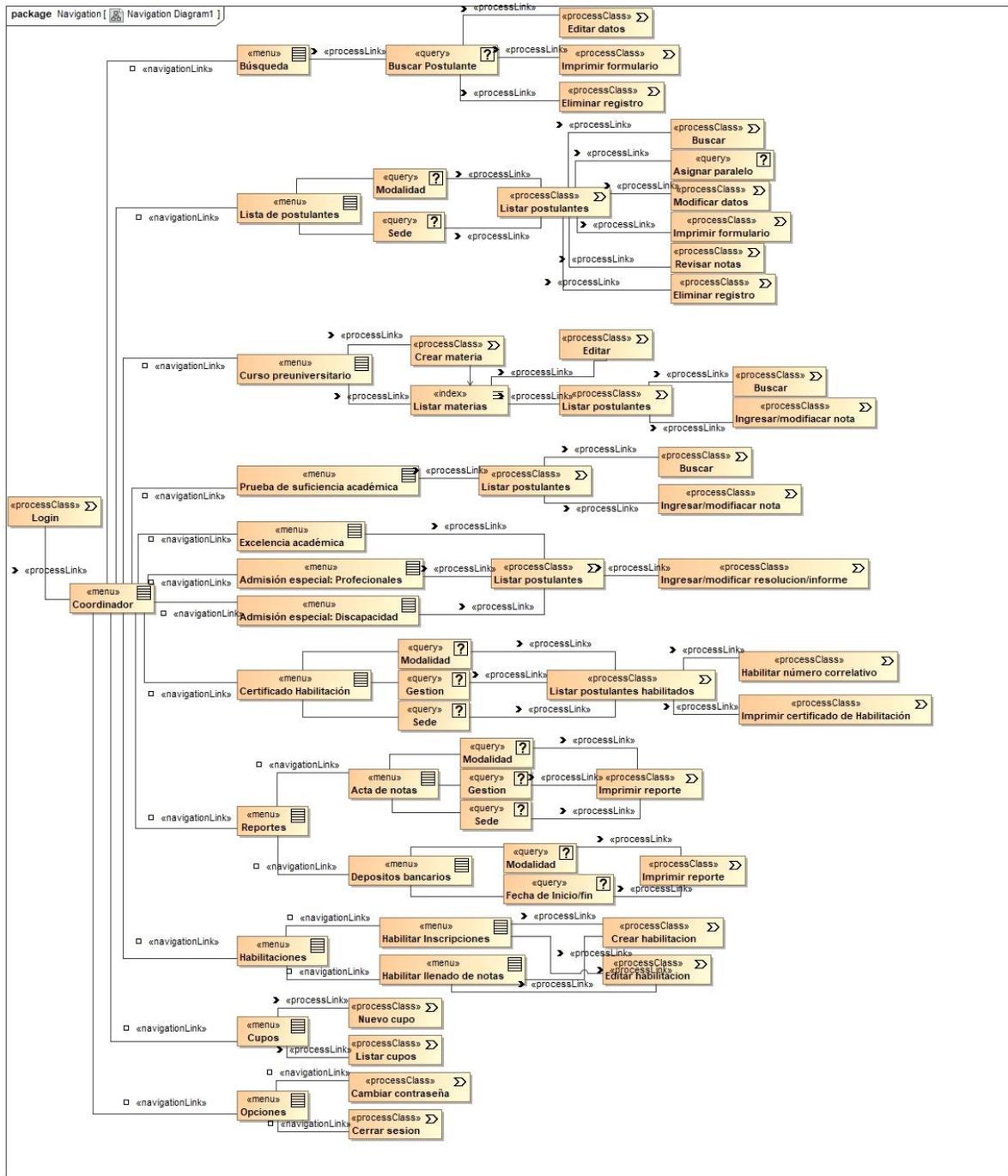
3.3. MODELOS DE NAVEGACIÓN

Figura 3.12 Modelo de navegación para administrador



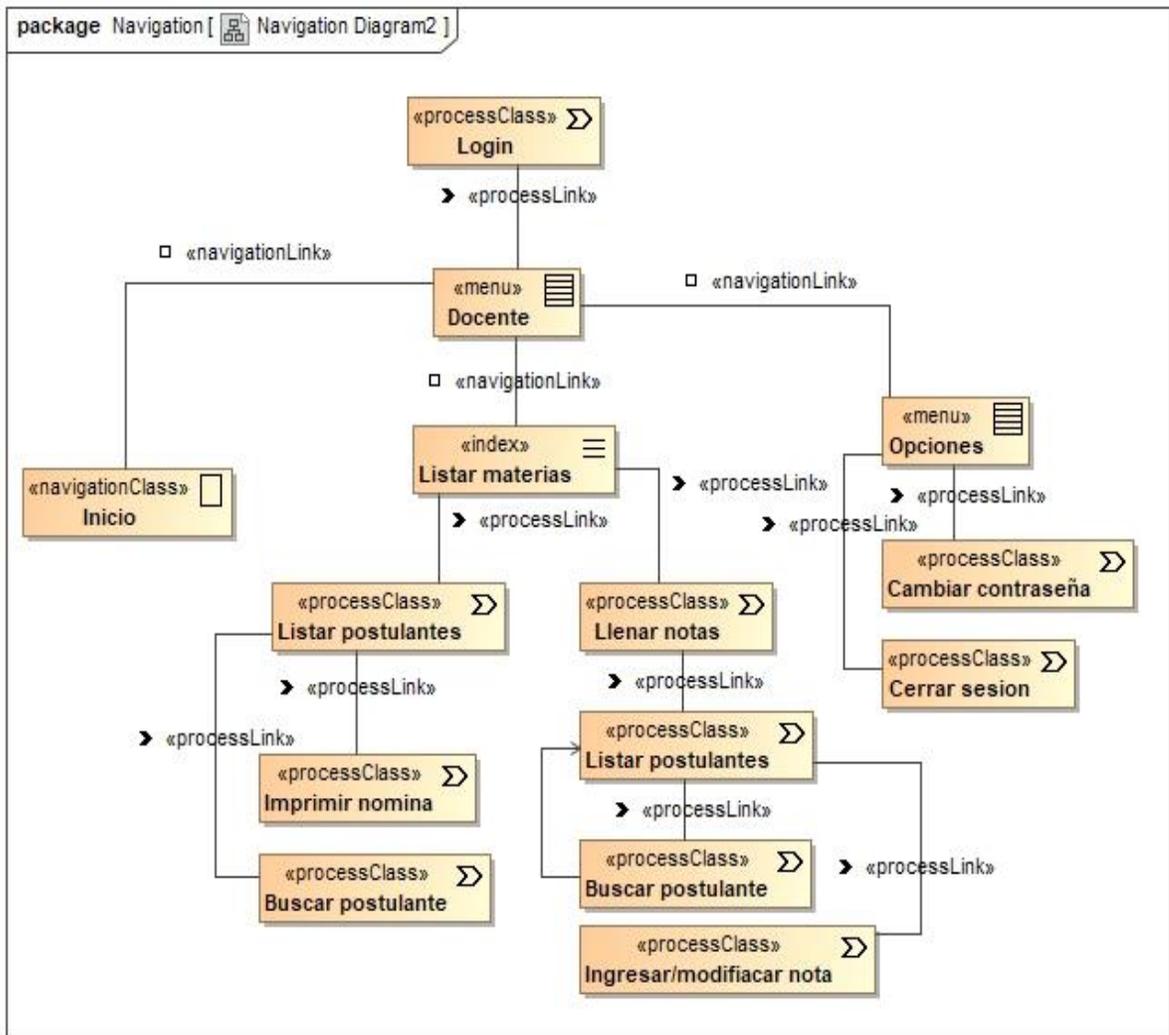
Fuente: (Elaboración propia)

Figura 3.13 Modelo de navegación para el módulo de coordinador



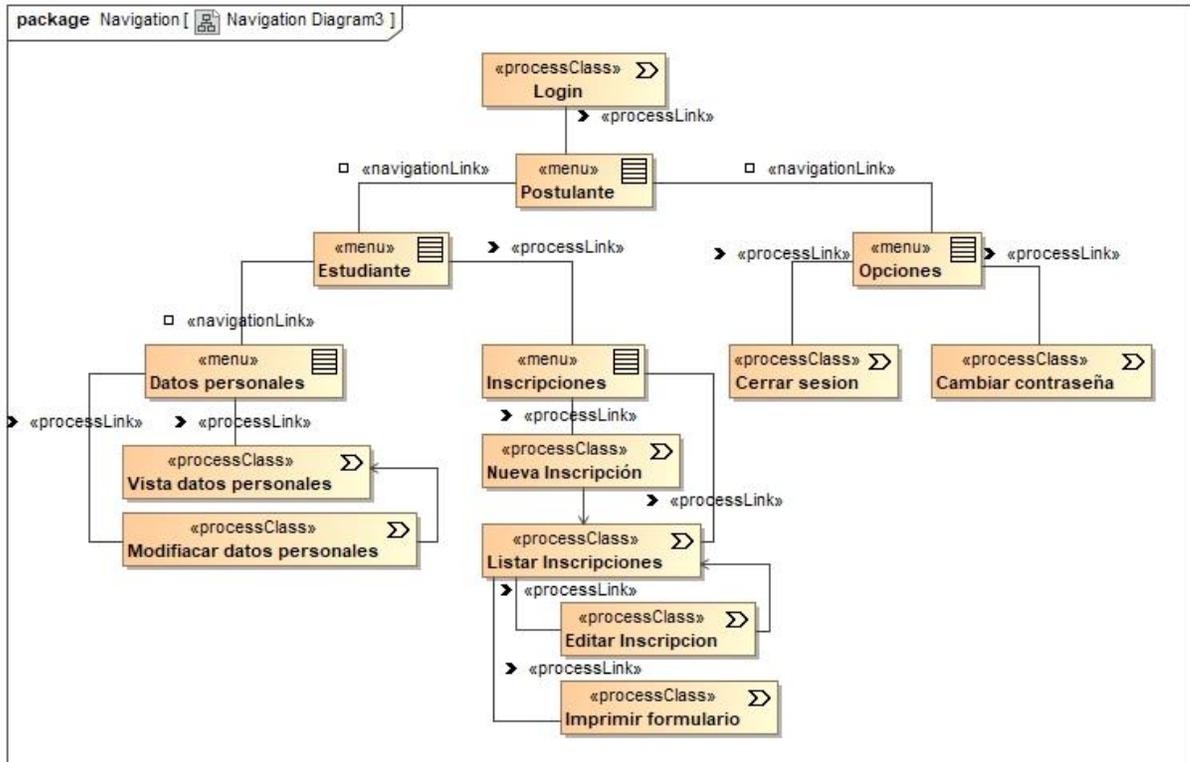
Fuente: (Elaboración propia)

Figura 3.14 Modelo de navegación para el módulo docente



Fuente: (Elaboración propia)

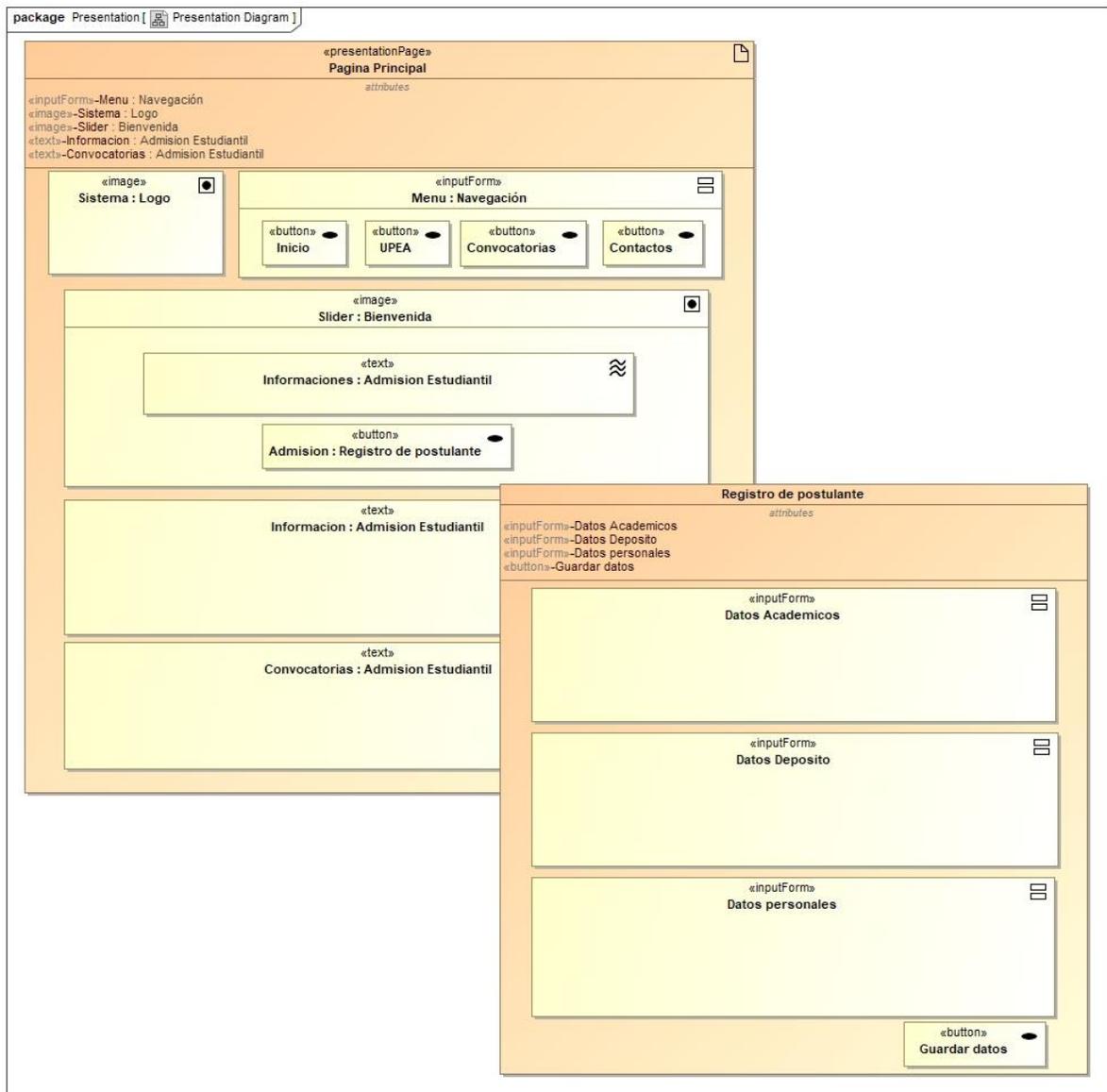
Figura 3.15 Modelo de navegación para el módulo postulante



Fuente: (Elaboración propia)

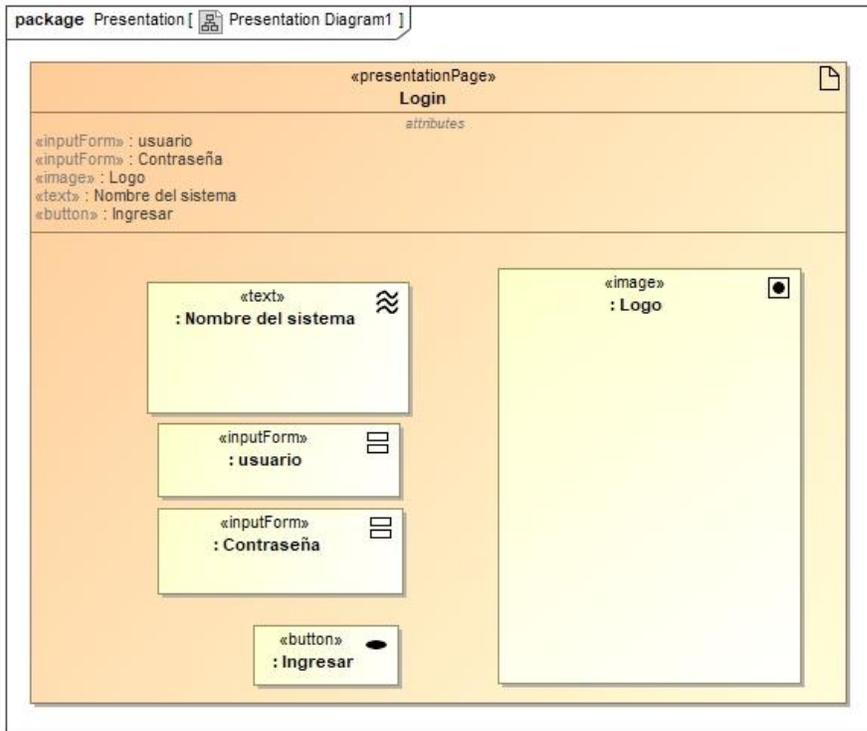
3.4. MODELO DE PRESENTACIÓN

Figura 3.16 Diagrama de presentación Página principal



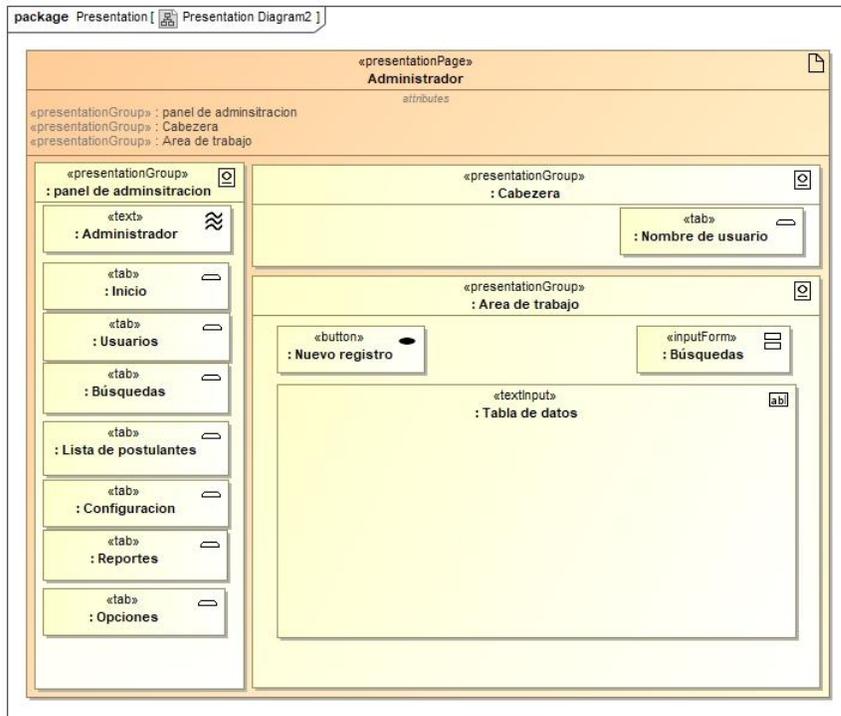
Fuente: (Elaboración propia)

Figura 3.17 Diagrama de presentación Login



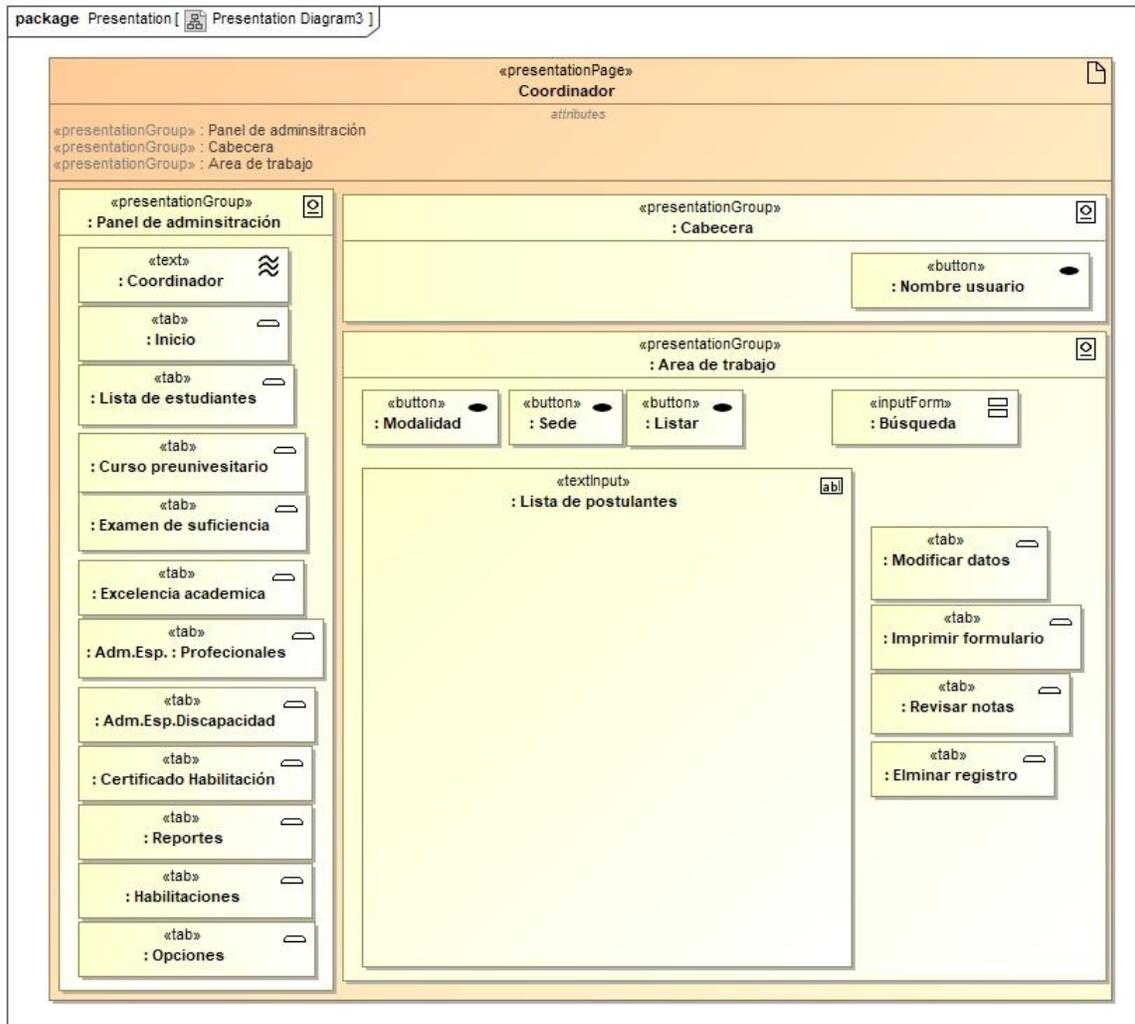
Fuente: (Elaboración propia)

Figura 3.18 Diagrama de presentación Administrador



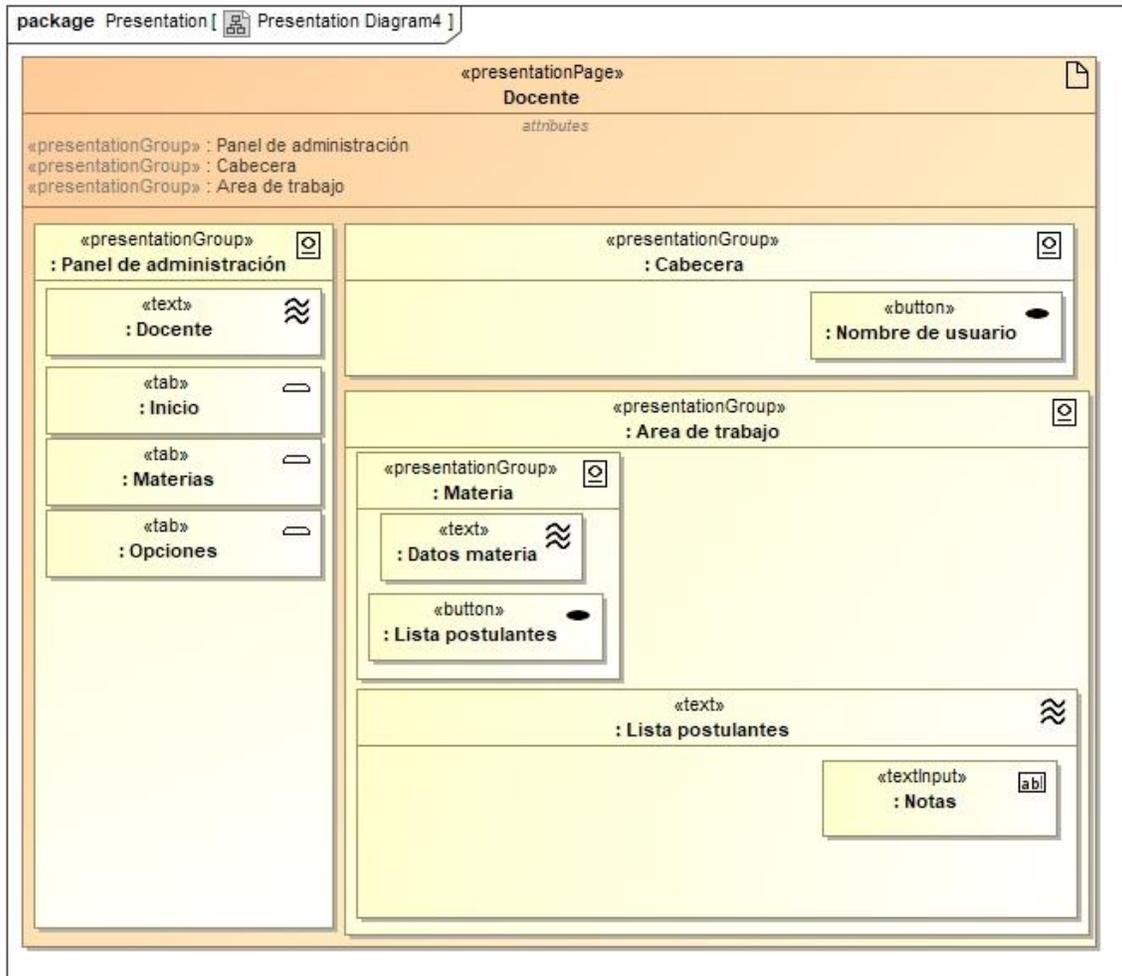
Fuente: (Elaboración propia)

Figura 3.19 Diagrama de presentación Coordinador



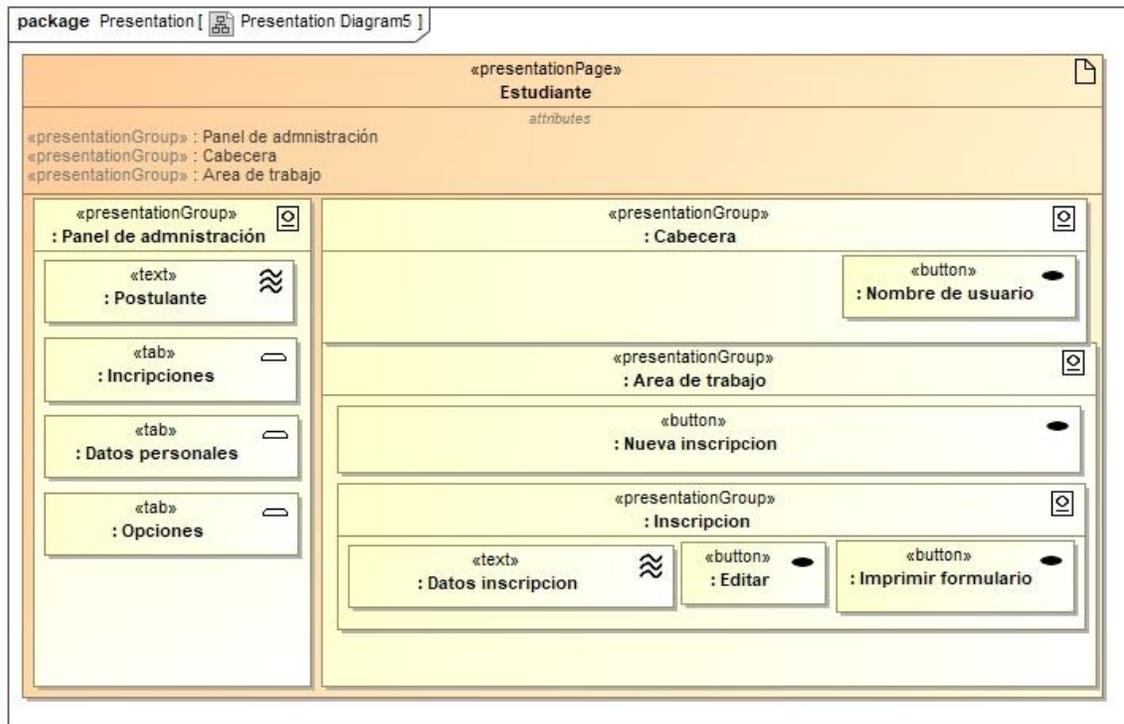
Fuente: (Elaboración propia)

Figura 3.20 Diagrama de presentación Docente



Fuente: (Elaboración propia)

Figura 3.21 Diagrama de presentación Estudiante



Fuente: (Elaboración propia)

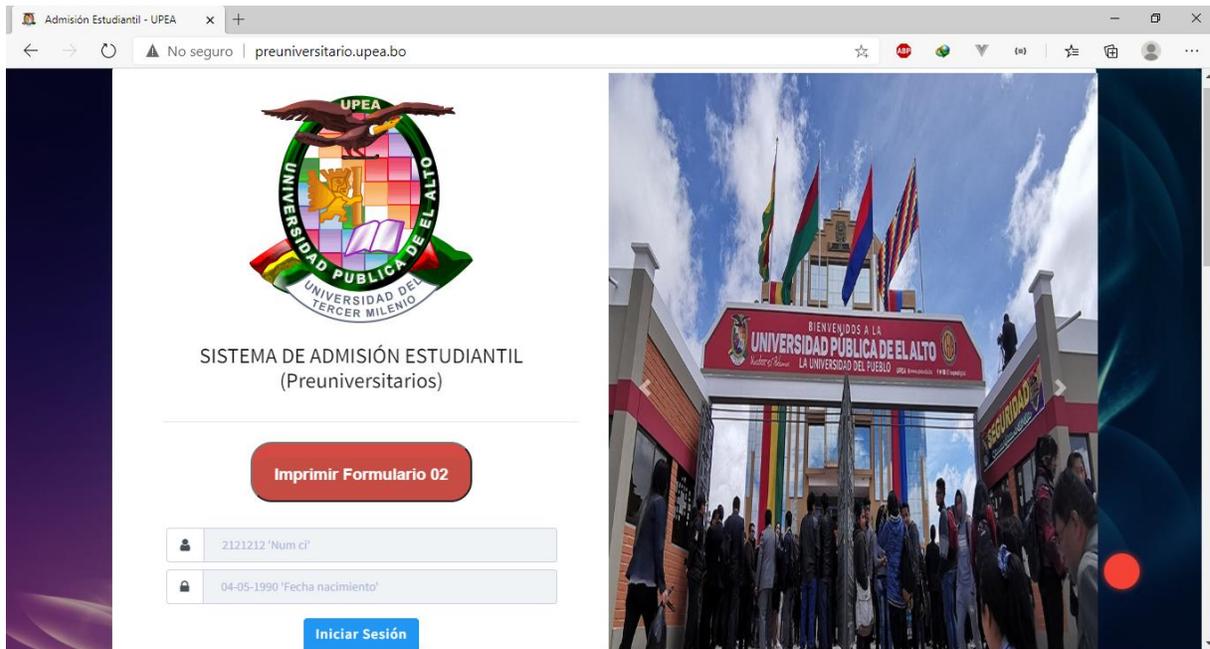
3.5. DESARROLLO DE SISTEMA

El desarrollo de sistema se realizó haciendo el uso del framework CodeIgniter para php, haciendo uso del gestor de base de datos mariadb. Primero se desarrolló en un entorno de pruebas utilizando la aplicación xampp y después se migro a un servidor de pruebas dentro de los servidores de la Universidad, finalmente se implementó el sistema usando alojándolo en los servidores de la universidad con el subdominio www.preuniversitario.upea.bo

3.5.1. Interfaz de Inicio de sesión y registro de estudiantes

En esta interfaz se muestran la página de inicio de sesión y para el registro de postulantes.

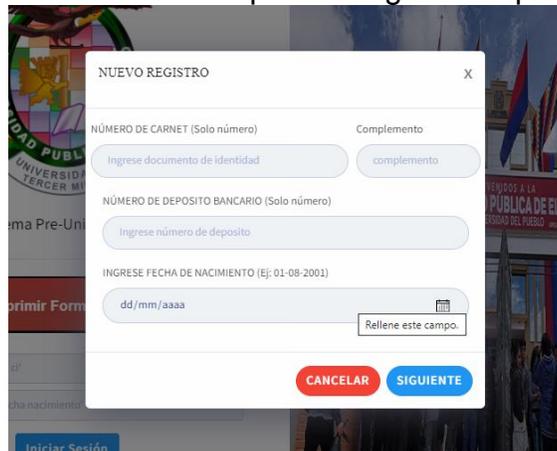
Figura 3.22 Inicio de sesión y registro de postulantes



Fuente: (Elaboración propia)

Como se puede apreciar para el ingreso al sistema se pide el usuario y contraseña que posteriormente se validará en sistema y permitirá el ingreso al sistema de acuerdo a los privilegios que tenga el usuario

Figura 3.23 Verificación para el registro de postulantes



Fuente: (Elaboración propia)

Adicionalmente se tiene el modal para el registro de postulantes, donde el postulante deberá ingresar su ci, y su fecha de nacimiento, después de la verificación se muestra el siguiente formulario:

Figura 3.24 Formulario de inscripción de postulante

Inscripción Nuevo

Carrera a la que Postula:

Fecha de inscripción	Modalidad_ingreso	Carrera	Sede
10/12/2020	[MODALIDAD]		

Número Compte. Bancario	Fecha Deposito Bancario	Monto deposito	Tipo moneda
78954654	dd/mm/aaaa		Bolivia

Datos Personales:

Apellido Paterno	Apellido Materno	Nombre(s)	Nacionalidad
Ingresar apellido pat	Ingresar apellido ma	Ingresar nombre(s)	Nacionalid

C.I.(Número)	Expedido	Tipo de documento	Fecha de Nacimiento
7895465			dd/mm/aaaa

Genero	Estado Civil	Departamento nacimiento	Provincia nacimiento
		[DEPARTAMENTC]	[PROVINCIA]

Lugar nacimiento	Telefono/Celular	Correo_electronico
[MUNICIPIO]	telefono/celular	Ingrese Correo electronico

Dirección:

Departamento	Provincia	Municipio	Distriton*
[DEPARTAMENTC]	[PROVINCIA]	[MUNICIPIO]	

Domicilio: Calle o Avenida	Zona o Villa	N°puerta
Ingresar Av/calle	Ingresar Zona/Villa	

Datos Egreso Secundaria:

Nombre Colegio Egreso:	País colegio:	Año egreso	Tipo
Nombre colegio Egreso	[PAIS]	egreso	

Bachillerato	Otro:	Área :	Ciudad/Localidad :
		Ingrese localidad

Universidad que expide el título de bachiller(Opcional)

Universidad:	Año de título:	Nro de título
[SELECCIONE]	Año de título	egreso

Datos socio-economicos

Vivienda que habita :	Características de la vivienda	Trabaja? :	Jornada laboral
[VIVIEN]	[TIPO VIVIENDA]		Jornada Laboral

Nro de hermanos que estudian en la universidad

Nro de hermanos

[Cancel](#) [IMPRIMIR Form.02](#)

Fuente: (Elaboración propia)

3.5.2. Interfaz de Administrador

En esta interfaz se observan las opciones que tiene el administrador que son la gestión de usuarios, el listado de postulantes las configuraciones dentro del sistema que se tiene.

Figura 3.25 Administración de usuarios

#	Nombres	Apellidos	Correo	Username	Carrera	Grupo	Estado
1	Sie	UPEA	sie@upea.bo	admin	-	admin	
2	JIMENA SILVIA	LIMACHI GUACHALLA	Upea.ninguno@gmail.com	JIMENA_4280824	ADMINISTRACIÓN DE EMPRESAS	coordinador	Activo
3	GUSTAVO	DIAZ MACHICADO	Upea.ninguno@gmail.com	GUSTAVO_6027652	ADMINISTRACIÓN DE EMPRESAS	coordinador	Activo
4	RANDY	CHAVEZ GARCIA	Upea.ninguno@gmail.com	RANDY_5483189	CIENCIAS DE LA COMUNICACIÓN SOCIAL	coordinador	Activo
5	SANTOS	QUISPE CONDORI	Upea.ninguno@gmail.com	SANTOS_5472625	GESTIÓN TURÍSTICA Y HOTELERA	coordinador	Activo
6	RENE	CHOQUEHUANCA MARCA	Upea.ninguno@gmail.com	RENE_4992984	DERECHO	coordinador	Inactivo
7	JONATHAN	YANA CHAVEZ	Upea.ninguno@gmail.com	JONATHAN_5984904	ECONOMÍA	coordinador	Activo
8	JOHNNY	SANGALLI CHUJIMA	Upea.ninguno@gmail.com	JOHNNY_5479762	MEDICINA	coordinador	Activo

Fuente: (Elaboración propia)

3.5.3. Interfaz de Coordinador

En esta interfaz se tienen las opciones para la gestión de postulantes para los coordinadores, es aquí donde el coordinador podrá registrar y revisar notas, consultar el número de inscritos dentro del sistema

Aquí es también donde se generan los reportes y se generan las actas de notas

Figura 3.26 Lista de postulantes

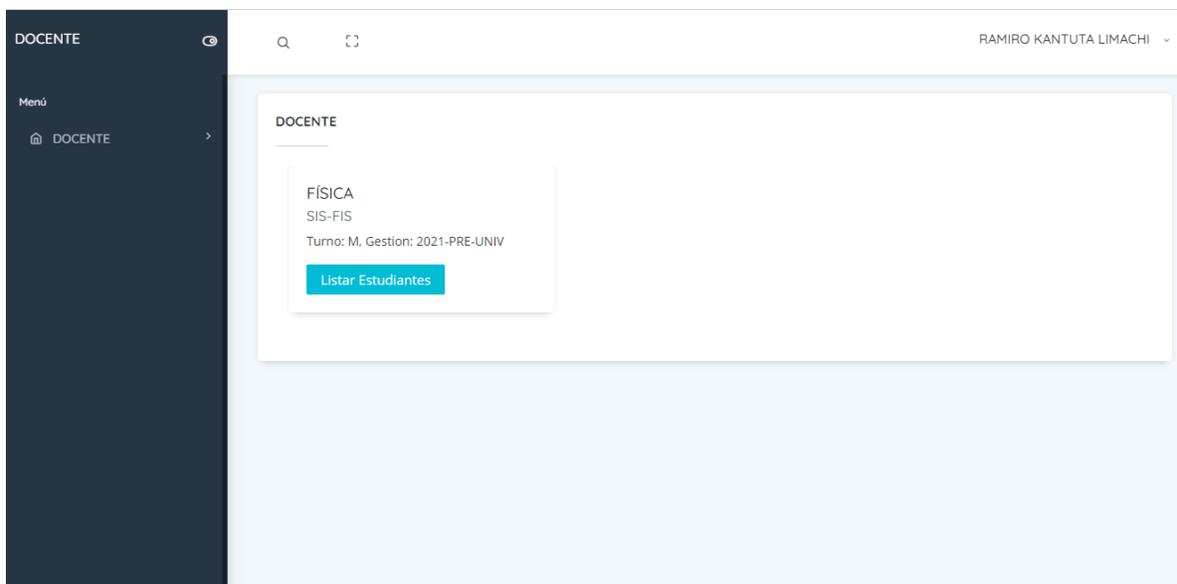
#	Carnet Identidad	Apellidos y Nombres	Celular	Modalidad	Gestión	Fecha inscripción	Numero dep
1	8322345 LP	RAMOS MAURICIO CHRISTIAN	74866376	PRUEBA DE SUFICIENCIA ACADÉMICA	2021-PRE-UNIV	2020-12-02	54555472
2	6153395 LP	MAYTA CALIZAYA JOSE CARLOS	63227904	PRUEBA DE SUFICIENCIA ACADÉMICA	2021-PRE-UNIV	2020-12-04	54878770
3	10010215 LP	AGUIRRE COPA BRAYAN	70645512	PRUEBA DE SUFICIENCIA ACADÉMICA	2021-PRE-UNIV	2020-11-25	53836157
4	13297738 LP	ALANOCA APAZA MIRIAM	74031905	PRUEBA DE SUFICIENCIA ACADÉMICA	2021-PRE-UNIV	2020-12-07	55016153
5	11063967 LP	ALANOCA ULO MIRIAM NATALY	71517330	PRUEBA DE SUFICIENCIA ACADÉMICA	2021-PRE-UNIV	2020-12-07	55143502
6	14047209 LP	APAZA MICHEL NATHALIA BELEN	65600560	PRUEBA DE SUFICIENCIA ACADÉMICA	2021-PRE-UNIV	2020-11-19	53334912
7	9209665 LP	ATACURURI VASQUEZ BRYAN ARIEL	73230721	PRUEBA DE SUFICIENCIA ACADÉMICA	2021-PRE-UNIV	2020-12-08	55119659

Fuente: (Elaboración propia)

3.5.4. Interfaz de Docente

En esta interfaz el docente puede listar las materias que tiene asignadas, y una vez que el coordinador habilite el llenado de notas el docente podrá hacer el llenado correspondiente de las notas.

Figura 3.27 Listado de materias



Fuente: (Elaboración propia)

3.5.5. Interfaz de postulante

En esta interfaz se muestra las opciones que tiene el postulante, donde puede listar las inscripciones activas que tiene, así también como modificar sus datos personales

Figura 3.28 Listado de inscripciones

#	Carrera	Sede	Modalidad Ingreso	Gestión	Acciones
1	ARQUITECTURA	VILLA ESPERANZA	CURSO PREUNIVERSITARIO	2021-PRE-UNIV	Editar Formulario

Fuente: (Elaboración propia)

3.6. PRUEBAS DE SOFTWARE

Para comprobar que el sistema desarrollado tenga el mínimo de errores se realizan diferentes pruebas. Para ello se aplican las técnicas de pruebas de caja negra y pruebas de caja blanca:

3.6.1. Pruebas de caja negra

Para hacer las pruebas de caja negra, se hace el uso de la técnica de partición de equivalencia que es una de las más efectivas para analizar los valores válidos e inválidos de las siguientes interfaces:

Tabla 3.11 Caso de prueba de Ingreso al sistema

Caso de Prueba de Ingreso al Sistema	
Código:	Caso de Prueba 01
Descripción de la Prueba	Ingresar al sistema mediante usuario y contraseña, después de la validación correspondiente
Condiciones de ejecución	El usuario debe encontrarse registrado previamente.
Pasos de Ejecución	<ul style="list-style-type: none"> • El sistema muestra la pantalla de autenticación, donde pide usuario y contraseña para ingresar. • El usuario llena el formulario y presiona el botón de iniciar sesión.
Resultado esperado	El sistema valida los datos ingresados y permite al usuario el ingreso de acuerdo a su rol asignado.
Evaluación de prueba	Se realiza el ingreso al sistema con la verificación previa, en caso de que la verificación falle, el usuario es devuelto a la página de inicio.

Fuente: (Elaboración propia)

Tabla 3.12 Caso de prueba de registro de usuario

Caso de Prueba de Registro de usuarios	
Código:	Caso de Prueba 02
Descripción de la Prueba	Registro de usuarios
Condiciones de ejecución	El usuario debe ser administrador y debe haber iniciado sesión en el sistema.
Pasos de Ejecución	<ul style="list-style-type: none"> • El sistema muestra la pantalla para buscar personas registradas en la universidad. • El usuario encuentra la persona y presiona la opción de asignar usuario. • El sistema le muestra un formulario para la creación de nuevo usuario.

	<ul style="list-style-type: none"> • El sistema valida los datos ingresados y crea al nuevo usuario.
Resultado esperado	El sistema valida y crea el nuevo usuario
Evaluación de prueba	Se realizó la creación de usuario de manera efectiva.

Fuente: (Elaboración propia)

Tabla 3.13 Caso de prueba de registro de estudiante

Caso de Prueba de Registro de postulante	
Código:	Caso de Prueba 02
Descripción de la Prueba	Registro de postulante mediante sus datos personales
Condiciones de ejecución	El postulante debe estar en la página, y haber seleccionado la opción de imprimir formulario
Pasos de Ejecución	<ul style="list-style-type: none"> • El sistema muestra una pantalla que le pedirá al postulante su carnet de identidad, depósito bancario y fecha de nacimiento. • El sistema valida que el postulante no este matriculado ni sea profesional o egresado de la universidad o que ya este registrado en el sistema. • El sistema después de la verificación le muestra un formulario para que ingrese sus datos. • El postulante debe ingresar sus datos y después presionar el botón de imprimir formulario.
Resultado esperado	El sistema valida los datos ingresados y genera el formulario de inscripción.
Evaluación de prueba	Se realiza el registro del postulante correctamente

4. MÉTRICAS DE CALIDAD, ESTIMACIÓN DE COSTOS DE SOFTWARE

Para asegurar que el proyecto se desarrolle de aplicaron una serie de métricas de calidad seguridad, donde se calcularon los parámetros de calidad que cumple el sistema.

También se hizo la estimación de costos para conocer el esfuerzo aplicado, el número de personas necesarias para desarrollar el trabajo y el tamaño del sistema desarrollado

4.1. MÉTRICAS DE CALIDAD

Para medir la calidad del software desarrollado se empleará las métricas de calidad de la ISO 9126 o ISO/IEC9126 que proporciona las métricas necesarias para demostrar la calidad del sistema.

4.1.1. Funcionalidad

Es un conjunto de atributos y características que debe cumplir el sistema para satisfacer los requerimientos funcionales. La funcionalidad es el grado en que el sistema satisface las necesidades que se indican los siguientes sub atributos: estabilidad, exactitud, interoperabilidad, cumplimientos de seguridad.

4.1.1.1. Punto función

El desarrollo de esta técnica de estimación de software, toma en cuenta 5 características:

- *Número de entradas de usuario:* El número de entradas en la que el usuario proporciona datos al sistema
- *Número de salidas de usuario:* El número de salidas que el sistema proporciona al sistema: mensajes, notificaciones, reportes, alertas, etc.
- *Número de peticiones de usuario:* Es el número de entradas interactivas que genera una respuesta del software como salidas interactivas
- *Número de Archivos:* Es el número de archivos, se toma en los grupos lógicos de datos, o archivos independientes.

- *Número de interfaces externas:* Es la suma de todas las interfaces legibles por la máquina.

Tomando en cuanto lo anteriormente descrito, se generaron las siguientes tablas:

Tabla 4.1 Parámetros de medición de punto función

Parámetros de medición	Cuenta
Número de entradas de Usuario	26
Número de salidas de usuario	35
Número de peticiones de usuario	25
Número de archivos	43
Número de interfaz externa	3

Fuente: (Elaboración propia)

Habiendo realizado el conteo de los parámetros de función en la tabla 4.1 se hace el cálculo de los puntos de función sin ajustar en la siguiente tabla:

Tabla 4.2 Punto función sin ajustar

Parámetro de medición	Cuenta	Factor	Total
Número de entradas de Usuario	26	4	104
Número de salidas de usuario	35	7	245
Número de peticiones de usuario	25	4	100
Número de archivos	43	10	430
Número de interfaz externa	3	5	15
Cuenta total			894

Fuente: (Elaboración propia)

En la tabla 4.2 se puede apreciar la cuenta total de los puntos de función si ajustar, para determinar los valores ajustados se hace el empleo de los factores de complejidad listados a continuación:

Tabla 4.3 Factores de complejidad

Importancia	0%	20%	40%	60%	80%	100%	Fi
Escala	No influyente	Incidental	Moderado	Medio	Significativo	Esencial	
Factor	0	1	2	3	4	5	
¿Requiere el sistema copias de seguridad y de recuperación fiable?						X	5
¿Se requiere comunicación de datos?						X	5
¿Existen funciones de procesos distribuidos?					X		4
¿Es crítico el rendimiento?					X		4
¿El sistema web será ejecutado en el SO. Actual?					X		4
¿Se requiere una entrada interactiva para el sistema?			X				2
¿Se requiere que el sistema tenga entradas a datos con múltiples ventanas?				X			3
¿Se actualizan los archivos de forma interactiva?			X				2
¿Son complejas las entradas, salidas, los archivos o las peticiones?			X				2
¿Es complejo el procesamiento interno del sistema?				X			3
¿Se ha diseñado el código para ser reutilizado?				X			3

¿Se ha diseñado el sistema para facilitar al usuario el trabajo y ayudarlos a encontrar la información?					X		4
Total							39

Fuente: (Elaboración propia)

Para el cálculo del punto de función ajustado se utiliza la siguiente formular:

$$PF = \text{conteo total} * [0.65 + 0.01 * \Sigma(fi)]$$

Donde:

$$\Sigma(fi)$$

Sumatoria de los valores de los factores de ajuste

Se tiene:

$$PF = 894 * [0.65 + 0.01 * 39]$$

$$PF = 894 * 1.04$$

$$PF = 929.76$$

Considerando el máximo valor en la sumatoria de los factores de ajuste $\Sigma(fi) = 70$, se calcula el siguiente valor:

$$PF_{max} = 894 * [0.65 + 0.01 * 70]$$

$$PF_{max} = 894 * 1.35$$

$$PF_{max} = 1209.9$$

El cálculo de la funcionalidad se obtiene de la relación entre el valor máximo con el valor obtenido del punto de función ajustado:

$$\text{Funcionalidad} = \frac{PF}{PF_{max}}$$

$$\text{Funcionalidad} = \frac{926.76}{1209.9}$$

$$\text{Funcionalidad} = 0.766 * 100\%$$

$$\text{Funcionalidad} = 76.6 \%$$

Por lo tanto, la funcionalidad del sistema desarrollado llega a ser del 76.6%, esto quiere decir que el sistema tiende a funcionar un 76.6 % sin riesgo de fallar, y un 23.4% de que el sistema colapse.

4.1.2. Confiabilidad

Es la capacidad de un sistema para mantener su nivel de ejecución bajo condiciones normales en un periodo de prueba establecido, cuyo criterio son la madurez, recuperabilidad, tolerancia a fallos y cumplimiento de fiabilidad

Para determinar la confiabilidad del sistema especificamos el tiempo en que empieza a funcionar, a partir de allí se realiza las observaciones hasta un tiempo determinado.

Donde se toma en cuenta que:

$$P(T \leq t) = F(t) \quad \Rightarrow \quad \text{Probabilidad de fallas}$$

$$P(T \leq t) = 1 - F(t) \quad \Rightarrow \quad \text{Probabilidad de trabajo sin fallas}$$

Para calcular la confiabilidad del sistema se toma en cuenta la siguiente función:

$$F(t) = f * e^{-\mu * t}$$

Donde:

$$f = \text{funcionalidad del sistema}$$

$$\mu = \text{Probabilidad de error del sistema}$$

$$t = \text{tiempo de prueba del sistema}$$

Tomando 20 días como periodo de prueba se tiene lo siguiente:

$$F(t) = 77 * e^{\frac{1}{15} * 20}$$

$$F(t) = 0.114 * 100\%$$

$$F(t) = 11.4\%$$

Reemplazando

$$P(T \leq t) = 0.114 = 11.4\%$$

$$P(T \leq t) = 1 - 0.114 \rightarrow P(T \leq t) = 0.886 * 100\% = 88.6 \%$$

Por lo tanto, la confiabilidad del sistema desarrollado es del 88.6%

4.1.3. Usabilidad

Para calcular la usabilidad de un sistema se toma en cuenta el factor humano, para conocer si el sistema cumple con los requisitos establecidos para el usuario se realiza la evaluación del sistema mediante encuestas a los usuarios del sistema.

Para determinar la usabilidad del sistema, se utiliza la siguiente formular:

$$FU = \left[\left(\frac{\sum x_i}{n} \right) * 100 \right]$$

Donde:

x_i = Valores de evaluación de cada pregunta

n = Numero de preguntas

Se toma en cuenta la siguiente tabla:

Tabla 4.4 Parámetros de medición de usabilidad

Escala	Valor
Muy bueno	5
Bueno	4

Regular	3
Malo	2
Pésimo	1

Fuente: (Elaboración propia)

Tabla 4.5 Tabla de usabilidad del sistema

N°	Pregunta	Si	No	Resultado
1	¿El sistema facilita el trabajo?	10	0	1
2	¿El sistema es comprensible?	8	2	0.8
3	¿El sistema es agradable a la vista?	8	2	0.8
4	¿El sistema hace lo que dice que hace?	9	1	0.9
5	¿Las Respuestas del sistema son satisfactorias?	9	1	0.9
6	¿Es fácil aprender a manejar el sistema?	8	2	0.8
7	¿El sistema satisface las necesidades que usted requiere?	9	1	0.9
Total				6.1

Fuente: (Elaboración propia)

Con los resultados obtenidos del cuestionario obtenemos calculamos la usabilidad:

$$FU = \left[\left(\frac{6.2}{7} \right) * 100 \right]$$

$$FU = 0.886 * 100$$

$$FU = 88.6\%$$

Con esto concluimos que la usabilidad del sistema es del 88.6%

4.1.4. Eficiencia

La eficiencia mide si el sistema emplea óptimamente los recursos del sistema. Para calcular el grado de eficiencia que tiene el sistema se toma en cuenta la siguiente escala:

Tabla 4.6 Escala de valores de eficiencia

ESCALA	VALOR
Excelente	5
Bueno	4
Aceptable	3
Deficiente	2
Pésimo	1

Fuente: (Elaboración propia)

Para valorar la eficiencia del sistema se tiene la siguiente tabla:

Tabla 4.7 Valoración de la eficiencia del sistema

N°	PREGUNTA	PORCENTAJE
1	¿La distribución y estilo de la interfaz permite que un usuario introduzca con eficiencia las operaciones y la información?	4
2	¿Una secuencia de operaciones (o entrada de datos) puede realizarse con facilidad de movimientos?	4
3	¿Los datos de salida están presentados de modo que se entienden de inmediato?	3
4	¿Las operaciones jerárquicas están organizadas de manera que minimizan la navegación del usuario para hacer que alguna se ejecute?	5
5	¿Procesa y responde adecuadamente cuando realiza alguna consulta o búsqueda?	4
Total		20

Fuente: (Elaboración propia)

Para calcular la eficiencia tenemos la siguiente formular:

$$E = \frac{\sum x_i}{n} * \frac{100}{n}$$

Donde:

$\sum x_i =$ Sumatoria de los valores de eficiencia

$n =$ Numero de preguntas

Reemplazando se tiene lo siguiente:

$$E = \frac{20}{5} * \frac{100}{5}$$

$$E = 80\%$$

Con esto concluimos que la eficiencia del sistema es del 80%

4.1.5. Mantenibilidad

Permite medir el esfuerzo necesario para realizar modificaciones del sistema, ya sea por corrección de errores o por el incremento de módulos o funciones dentro del sistema. El estándar IEE94 sugiere la implementación del índice de madurez del sistema que proporciona un indicador de estabilidad del sistema desarrollado.

De esto modo tenemos que la siguiente ecuación para el índice de madurez del sistema:

$$IMS = \frac{[Mt - (Fa + Fc + Fd)]}{Mt}$$

Donde:

Tabla 4.8 Valores para la mantenibilidad

Descripción	Valor
Mt: Numero de módulos total de la versión actual	5
Fc: Numero de módulos de la versión actual que se cambiaron.	1
Fa: Numero de módulos de la versión actual que se añadieron.	0
Fd: Numero de módulos de la versión anterior que se eliminaron en la versión actual.	0

Fuente: (Elaboración propia)

Remplazando en la ecuación se tiene:

$$IMS = \frac{[5 - (1 + 0 + 0)]}{5}$$

$$IMS = 0.8 * 100\%$$

$$IMS = 80\%$$

Se concluye que el índice de mantenibilidad del sistema desarrollado es del 80%

4.1.6. Portabilidad

Es la capacidad de un software para ser transferido de un ambiente a otro donde se considera lo siguiente:

- **Adaptabilidad:** Evalúa la oportunidad para adaptar el software a diferentes ambientes sin necesidad de aplicarle modificaciones.
- **Facilidad de Instalación:** Es el esfuerzo necesario para instalar el software en un ambiente determinado.
- **Conformidad:** Permite evaluar si el software se adhiere a estándares o convenciones relativas a portabilidad.
- **Capacidad de reemplazo:** Se refiere a la oportunidad y el esfuerzo usado en sustituir el software por otro producto con funciones similares.

El sistema está desarrollado para ser implementado en cualquier servidor que tenga instalado los servicios de apache, y las herramientas de php y mariadb.

En el caso de del sistema desarrollado, al estar bajo el marco de trabajo Codeigniter, se cuenta con la facilidad de configurar el software para casi la mayoría de los ambientes.

En cuanto a la ejecución, al estarse aplicando una tecnología web, el sistema puede ser ejecutado en cualquier dispositivo que cuente con conexión a internet y un navegador web.

4.2. ESTIMACIÓN DE COSTOS DE SOFTWARE

4.2.1. Método de estimación COCOMO II

COCOMO II permite hacer estimaciones en base al tamaño del software, y un conjunto de factores de coste y de escala.

COCOMO II posee tres modelos:

- Composición de aplicación
- Diseño temprano
- Post-arquitectura

Para la estimación de costos de este proyecto se empleará el submodelo post-arquitectura.

Primeramente, se hace la estimación de las líneas de código, para esto se utilizará el modelo de estimación de puntos de función no ajustados, que fueron calculados anteriormente.

Para el cálculo de las líneas de código se utilizó la siguiente función:

$$LDC = UPF * TCF$$

Donde:

UPF = Puntos de función no ajustados

TCF = Factores de complejidad

Reemplazando en la formula se tiene:

$$LDC = 894 * 39$$

$$LDC = 34866$$

Para la estimación de costos de se debe convertir las líneas de código a KLDC (líneas de código por miles):

$$KLDC = \frac{LDC}{1000}$$

$$KLDC = \frac{34866}{1000}$$

$$KLDC = 34.866$$

Los coeficientes que se usaran se detallan en la siguiente tabla:

Tabla 4.9 Coeficientes de modelo COCOMO II

Proyecto de Software	A	b	c	d
Orgánico	2,4	1,05	2,5	0,38
Semi-acoplado	3,0	1,12	2,5	0,35
Empotrado	3,6	1,20	2,5	0,32

Fuente: (Pressman, 2010)

Debido a que el proyecto no supera las 50000 líneas de código se utilizaran los coeficientes para proyectos semi-acoplado.

Ecuaciones para calcular el costo de software:

Tabla 4.10 Ecuaciones COCOMO II

Variable	Ecuación	Tipo/Unidad
Esfuerzo requerido por el proyecto	$E = a * (KLDC)^b * FAE$	Persona/mes
Tiempo requerido por el proyecto	$T = c * (E)^d$	Meses
Número de personas requeridos para el proyecto	$NP = \frac{E}{T}$	Personas
Costo Total	$CT = SueldoMes * NP * T$	\$us

Fuente: (Prentica-Hall, 1981)

Tabla 4.11 Cálculo de los atributos FAE

Atributos	Valor					
	Muy bajo	Bajo	Nominal	Alto	Muy alto	Extra alto
Atributos de software						
Fiabilidad	0,75	0,88	1,00	1,15	1,40	
Tamaño de Base de datos		0,94	1,00	1,08	1,16	
Complejidad	0,70	0,85	1,00	1,15	1,30	1,65
Atributos de hardware						
Restricciones de tiempo de ejecución			1,00	1,11	1,30	1,66
Restricciones de memoria virtual			1,00	1,06	1,21	1,56
Volatilidad de la máquina virtual		0,87	1,00	1,15	1,30	
Tiempo de respuesta		0,87	1,00	1,07	1,15	
Atributos de personal						
Capacidad de análisis	1,46	1,19	1,00	0,86	0,71	
Experiencia en la aplicación	1,29	1,13	1,00	0,91	0,82	
Calidad de los programadores	1,42	1,17	1,00	0,86	0,70	
Experiencia en la máquina virtual	1,21	1,10	1,00	0,90		
Experiencia en el lenguaje	1,14	1,07	1,00	0,95		
Atributos del proyecto						
Técnicas actualizadas de programación	1,24	1,10	1,00	0,91	0,82	
Utilización de herramientas de software	1,24	1,10	1,00	0,91	0,83	

Restricciones de tiempo de desarrollo	1,22	1,08	1,00	1,04	1,10	
---------------------------------------	------	------	-------------	------	------	--

Fuente: (Elaboración propia en base a criterios de Pressman,2010)

Haciendo el cálculo del FAE se tiene:

$$FAE = 1.15 * 1.00 * 1.00 * 1.00 * 1.00 * 0.87 * 1.00 * 0.86 * 0.91 * 0.86 * 0.90 * 0.95 \\ * 0.82 * 0.91 * 1.00$$

$$FAE = 0.429$$

Reemplazando los coeficientes en las fórmulas de COCOMO II:

Esfuerzo:

$$E = a * (KLDC)^b * FAE$$

$$E = 2.4 * (34.866)^{1.05} * 0.429$$

$$E = 42.87 \text{ Personas/mes}$$

Tiempo:

$$T = c * (E)^d$$

$$T = 2.5 * 42.87^{0.38}$$

$$T = 10.42 \text{ equivalente a 10 meses}$$

Personal Promedio:

$$NP = \frac{E}{T}$$

$$NP = \frac{42.87}{10.42}$$

$$NP = 4,11 \cong 4 \text{ Personas}$$

Costo total:

$$CT = \text{SueldoMes} * NP * T$$

$$CT = 240\$ * 4 * 10$$

$$CT = 9600 \$us$$

4.3. SEGURIDAD

Uno de los aspectos más importantes a tomar en cuenta en este proyecto, es la implementación de normas de seguridad. La norma ISO 27002 evalúa y rectifica el cumplimiento de las normas, así como la mejora continua en base a un conjunto de controles que permiten reducir el riesgo de sufrir incidentes de seguridad en el funcionamiento del sistema dentro de la institución, para ello se toman en cuenta los siguientes tipos de seguridad:

4.3.1. Seguridad Lógica

Copias de Seguridad

Las copias de seguridad son respaldos que se tiene de las bases de datos, estos deben hacerse periódicamente para asegurarse que la información se encuentre respaldada.

Autenticación

Para la seguridad de datos del sistema se tienen registrado de nombre de usuario y contraseña de acceso encriptadas, según su nivel de acceso pueda realizar actividades en el sistema.

Para asegurar la seguridad de la información dentro del sistema se aplicaron los controles mediante el uso de sesiones y su verificación. En el sistema se tiene 4 niveles de usuario, cada uno con diferentes privilegios

Encriptación

El sistema permite almacenar las contraseñas de los usuarios de manera encriptada. Para lograr esto se utiliza el método de encriptación bcrypt que llega a ser más segura que hash y md5.

Base de datos

La mayoría de las aplicaciones web son usadas como un conducto entre muchas fuentes de datos y el usuario, las aplicaciones web son usadas frecuentemente para interactuar con una base de datos.

Aunque el tema de la seguridad en las bases de datos merece un tratamiento diferente al de las aplicaciones, se encuentran íntimamente relacionados

En cuanto a la forma de resguardo se realizó los siguientes puntos:

- Una vez que se accedió a los registros o se realizó las diferentes acciones dentro del sistema se hace la desconexión de la base de datos para que no se tenga ningún problema con terceros.
- Los respaldos o back-up de la base de datos se deberá realizar una vez por semana.

La información en una empresa es muy valiosa, por tanto, su resguardo es fundamental, la conexión a la base de datos y el cierre de la conexión es de forma automática.

4.3.2. Seguridad Física

Seguridad física y del entorno

Se prevé que los equipos donde se almacenan los datos y el sistema se encuentran resguardados en lugares seguros, con las condiciones necesarias para funcionar con el menor número de riesgos.

Equipamiento

Se tiene una adecuada protección física y mantenimiento de los servidores donde se almacenan los sistemas de la universidad, y el completo resguardo de las instalaciones donde se encuentran.

Acceso físico

Se restringe el acceso físico a las áreas críticas a toda persona no autorizada, para reducir el riesgo de accidentes fraudulentos y robos de contraseñas.

5. CONCLUSIONES Y RECOMENDACIONES

5.1. CONCLUSIONES

Habiendo concluido y realizado las tareas descritas en este proyecto, se logró implementar el sistema de admisión estudiantil para la gestión académica de las carreras en la Universidad Pública de El Alto, logrando de este modo alcanzar los objetivos que se habían planteado.

- Con la implementación del sistema de admisión estudiantil se facilitó la gestión de los registros de los postulantes, logrando de esta manera tener un mejor control del proceso de selección y evaluación de los postulantes que ingresan a las carreras de la Universidad Pública de El Alto.
- Se facilitó la tarea a los coordinadores de la gestión de la documentación que se entrega en Registros y Admisiones, logrando que el sistema imprima de manera eficaz los diferentes reportes que se manejan en el proceso de admisión estudiantil.
- El sistema permite tener un mayor control en la admisión estudiantil, lo que permite que este se realice de acuerdo a las normas e instructivos de la universidad en cuanto a la postulación de nuevos estudiantes.
- Se desarrollaron los formularios para la admisión estudiantil de manera que puedan interpolarse con los datos del sistema de Matriculación de estudiantes, permitiendo de esta manera simplificar los datos que se ingresan a la hora de la matriculación
- El sistema fue desarrollado en base a las normas de calidad ISO9126 e ISO27000, por lo tanto, el sistema cumple con las normas de confidencialidad, Integridad y Disponibilidad.

5.2. RECOMENDACIONES

Se proponen las siguientes recomendaciones:

- Se recomienda que los administradores de sistema sean afines a las ramas de Ingeniería de Sistemas.
- Se recomienda hacer copias de seguridad al finalizar cada semana para resguardar la información de la base de datos.

- Se recomienda la capacitación de los usuarios del sistema para que el uso del mismo se haga de manera correcta y evitar inconvenientes a futuro.

BIBLIOGRAFÍA

- AscensTechnologies. (2016). *Bootstrap, un framework para diseñar*. Obtenido de <https://www.acens.com/wp-content/images/2016/10/bootstrap-framework-acens-wp.pdf>
- Blanco, S. D. (25 de Junio de 2015). *Metodología UWE*. Obtenido de <https://metodologiauwe.wordpress.com/>
- Cabello, E. (29 de Octubre de 2013). *Fases o etapas de la metodologia UWE*. Obtenido de <http://evangellyscarolinacabellorodriguez.blogspot.com/2013/10/fases-o-etapas-de-la-metodologia-uwe.html>
- EcuRed, c. (29 de Agosto de 2017). *MariaDB*. Obtenido de <https://www.ecured.cu/index.php?title=MariaDB&oldid=2943520>
- EcuRed, c. (20 de junio de 2019). *CSS*. Obtenido de <https://www.ecured.cu/index.php?title=CSS&oldid=3417373>
- Estrada, J. G. (2018). *Desarrollo Web con PHP y MySQL*.
- Instituto de Tecnología de Columbia Británica. (19 de Septiembre de 2019). *CodeIgniter*. Obtenido de <https://codeigniter.com/userguide3/general/welcome.html>
- IONOS. (16 de Marzo de 2020). *CodeIgniter, el peso pluma de los frameworks PHP*. Obtenido de <https://www.ionos.es/digitalguide/paginas-web/desarrollo-web/codeigniter-framework-php-rapido-y-versatil/>
- ISO2700.es. (2005). *SGSI*. Obtenido de <https://www.iso27000.es/sgsi.html>
- Kendall, K. E., & Kendall, J. E. (2011). *Análisis y Diseño de Sistemas*. Mexico: PEARSON EDUCACIÓN.
- Kimell, P. (2008). *Manual de UML*. Mexico: MacGraw Hill Interamericana.
- Koch, N., Knapp, A., Zhang, G., & Baumeister, H. (2008). *UML-Based Web Engineering, An approach based on standards*. Germany: Ludwig-Maximilians-Universität München.
- Lapedra, R., Devece, C., & Guiral, J. (2011). *Introducción a la gestión de sistemas de Información en la empresa*. Publicacions de la Universitat Jaume I. Servei de Comunicació i Publicacions.
- MariaDB, F. (15 de 07 de 2020). *MariaDB*. Obtenido de <https://mariadb.org>
- McFarland, D. S. (2014). *JavaScript & JQuery*. Sebastopol: O'Really Media Inc.
- Minera, F. (2010). *Users: PHP6 Sitios dinámicos con el lenguaje más robusto*. Grandi: Fox Andina.
- Pressman, R. (2010). *Ingeniería de Software*. Mexico: Mc Graw Hill Educación.
- Sanz, D. M., & Morales, E. J. (2011). *Metodologías para el desarrollo de aplicaciones Web: UWE*. Obtenido de <https://jorgeportella.files.wordpress.com/2011/11/analisis-diseño-y-desarrollodeaplicacionesweb.pdf>
- Serna M., E. (2013). *Libro Blanco de la Ingeniería de Software en Latino America*. Medellín, Antioquia: Instituto Antioqueño de Investigación.

Somerville, I. (2011). *Ingenieria de Software*. Mexico: PEARSON EDUCACIÓN.

Tutorials Point Ltd. (2016). *MariaDB*.

Vega, J. F., & Vander Henst, C. (2011). *Guia HTML5*. Mexico.

Welling, L., & Thompson, L. (2004). *Desarrollo Web con PHP y MySql*. Mexico: Anaya multimedia.

ANEXOS

Anexo A. MANUAL DE USUARIO

Ingreso al sistema

Para ingresar al sistema debe ingresarse a la página www.preuniversitario.upea.bo donde se mostrará la pantalla de inicio.

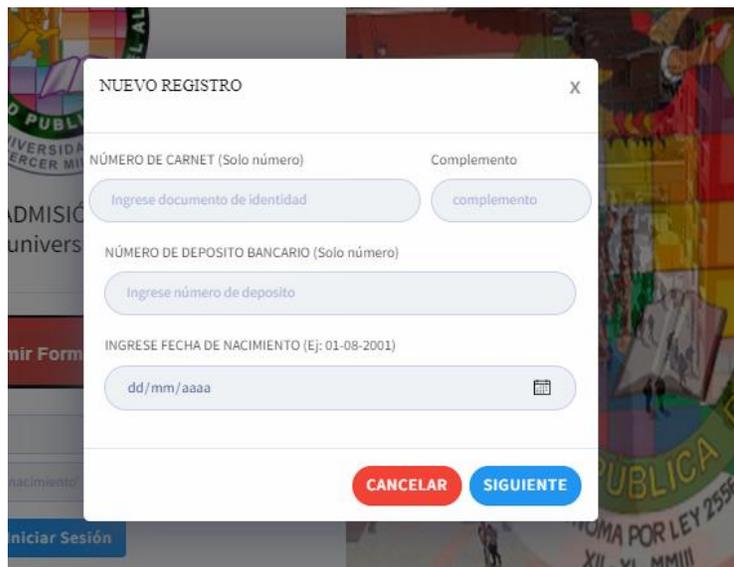


Se nos mostrará la pantalla donde se pedirá el ingreso de usuario y contraseña para el ingreso al sistema.

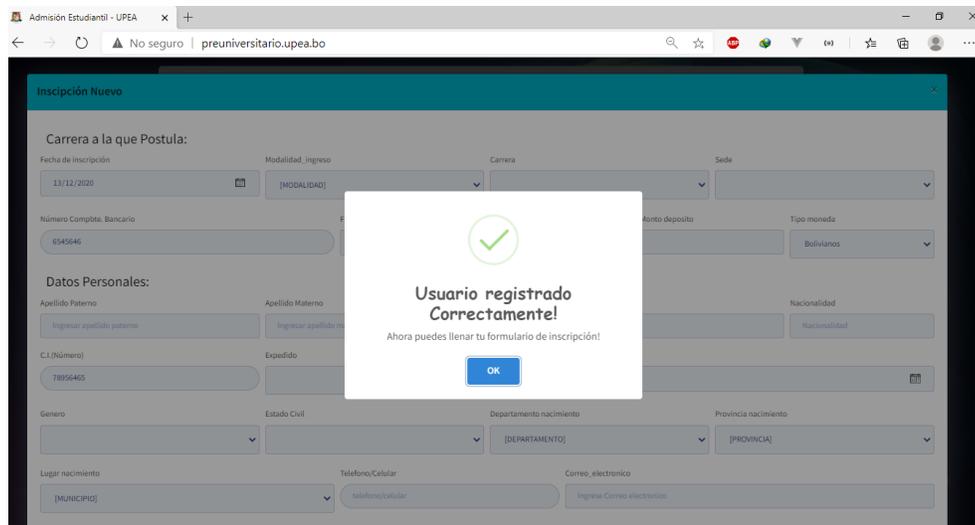
El sistema tiene 4 niveles de usuario: Administrador, Coordinador, Docente y Postulante.

Admisión Estudiantil

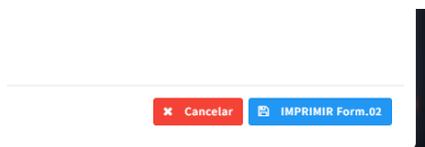
Para la admisión estudiantil se debe presionar en la opción de imprimir formulario02 que nos mostrará la siguiente pantalla:



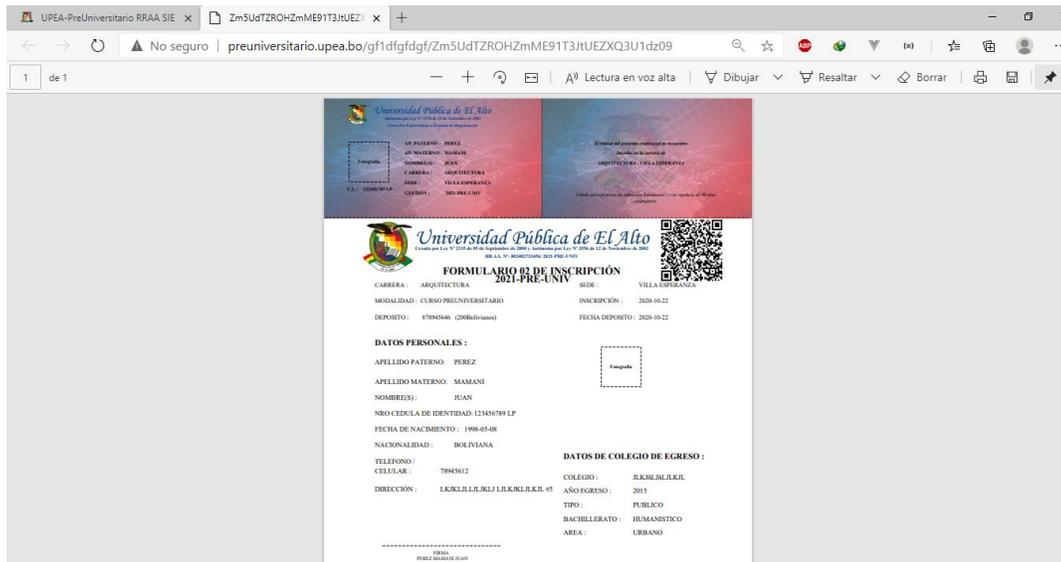
Después de ingresar los datos y el sistema haga la verificación de los datos se nos muestra el formulario de postulación:



Una vez que se llenen los datos del formulario. Debe presionarse el botón de imprimir formulario.

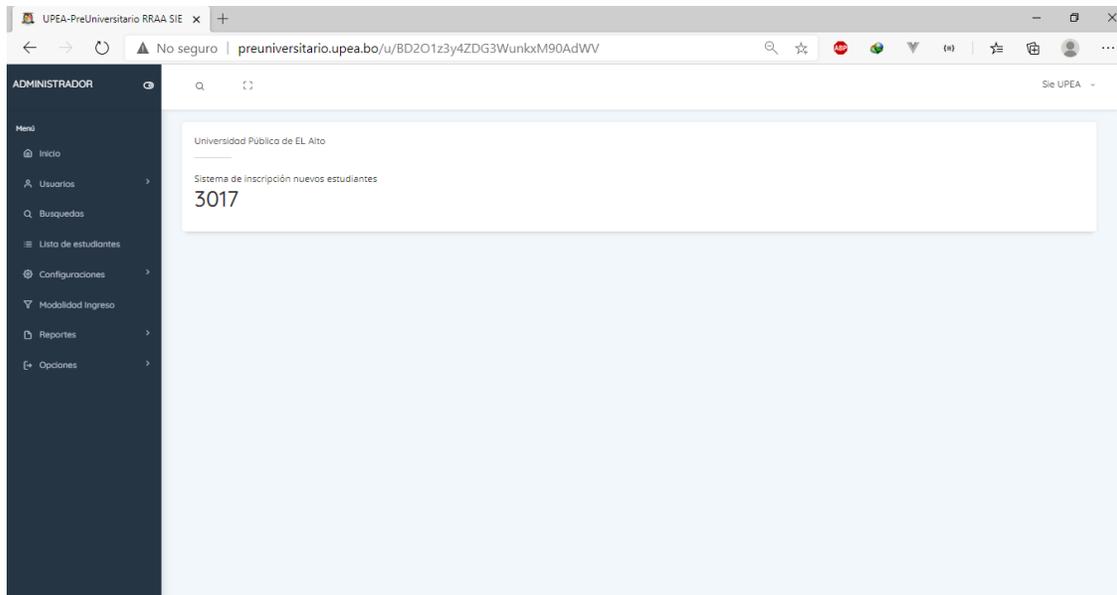


El sistema volverá a verificar los datos ingresados y generará el formulario de inscripción:



Módulo Administrador

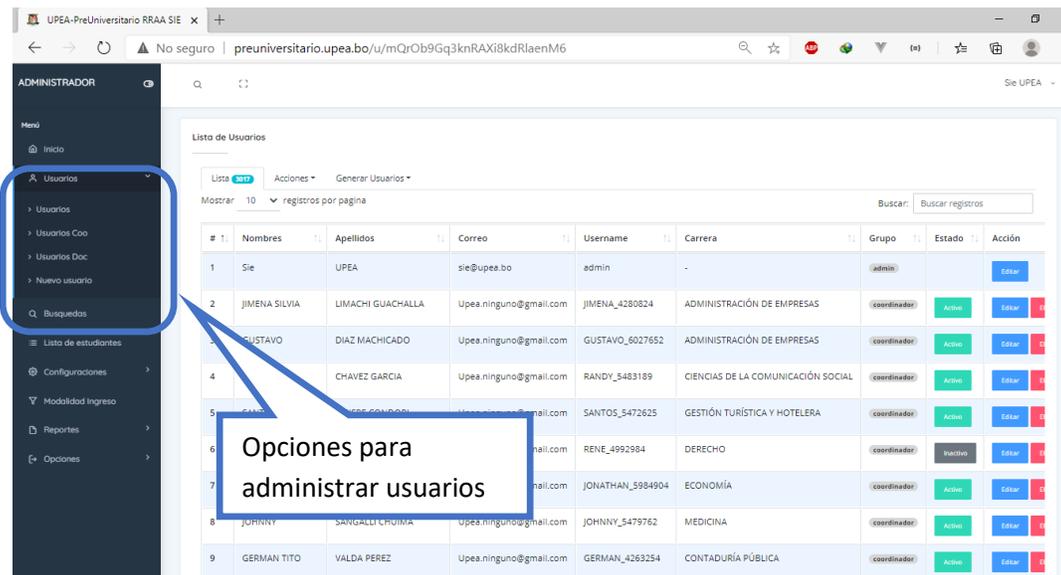
Después de iniciar sesión se nos muestra la siguiente pantalla:



Se tiene las siguientes opciones:

Inicio: Se muestra la página principal y de bienvenida.

Usuarios: Esta opción permite la administración de los usuarios del sistema:



Se tiene las siguientes opciones:

- Usuarios: Despliega la lista de todos los usuarios dentro del sistema.
- Usuarios Coo.: Muestra la lista de todos los usuarios que son coordinadores.
- Usuarios Doc.: Muestra la lista de todos los usuarios que son Docentes.
- Nuevo usuario: permite la creación de nuevos usuarios.

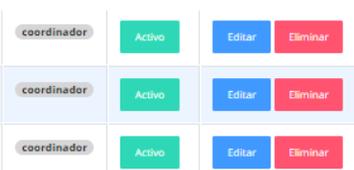
En la opción que nos despliegan usuarios tenemos las siguientes funciones



Acciones: Nos permite las creaciones de nuevos usuarios y exportar los usuarios a una lista Excel.

Generar Usuarios: Nos permite generar usuarios de Coordinadores y Docentes automáticamente, para ello los coordinadores y docentes deben tener la asignación docente de vicerrectorado.

En las opciones para gestionar usuarios se tiene:



Editar: Permite la edición de usuarios, al presionar el botón nos muestra un formulario para editar el usuario.

Eliminar: Permite eliminar un usuario.

Activo/Inactivo: Cambia el estado de un usuario.

Búsquedas:



Aquí es donde se hacen búsquedas de postulantes mediante la cedula de identidad del mismo. Una vez que se ingresen los datos el sistema desplegara la información del postulante. En caso de que el postulante no cumpla con los criterios para la admisión se mostrara un mensaje en la parte de observaciones.

Aquí también se puede hacer la inscripción de nuevos postulantes haciendo clic en la opción de Nueva Inscripción que mostrara la pantalla con el formulario de admisión estudiantil.

En la sección de búsqueda, cuando el sistema encuentra un postulante con el ci listado, mostrará el registro del postulante y las mostrará las opciones respectivas.

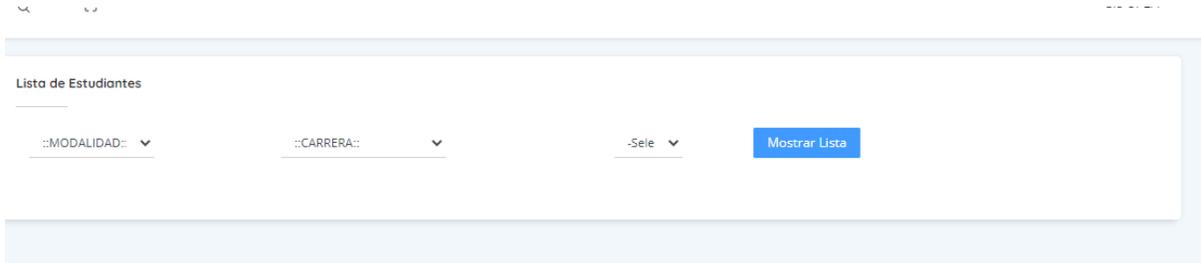
Ingreso...	Carrera	Sede	Carnet Identidad	Nombres & Apellidos	Gestión	Acciones
UNIVERSITARIO	ARQUITECTURA	VILLA ESPERANZA	123456789 LP	Perez Mamani Juan	2021-PRE-UNIV	Editar Formulario ... Eliminar...
UNIVERSITARIO	ADMINISTRACIÓN DE EMPRESAS	VILLA ESPERANZA	123456789 LP	Perez Mamani Juan	2021-PRE-UNIV	Activar
EFICIENCIA ACADÉMICA	INGENIERÍA DE SISTEMAS	VILLA ESPERANZA	123456789 LP	Perez Mamani Juan	1 2021-PRE-UNIV	Activar

Se tiene las siguientes opciones:

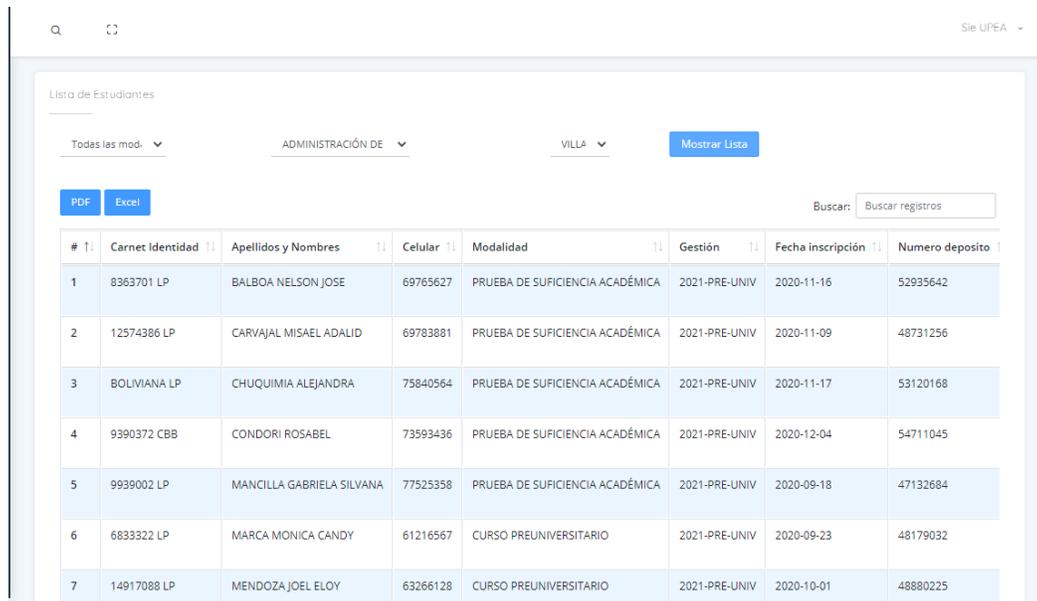
- **Editar:** Permite editar los datos del formulario.
- **Formulario:** Permite imprimir el formulario del postulante.
- **Icono de whatsapp:** Abre una ventana en whatsapp, con la opción de enviarle un mensaje por whatsapp al postulante.
- **Eliminar:** Se desactiva un registro
- **Activar:** Permite activar un registro.

Lista de estudiantes

Esta opción permite listar los postulantes registrados en el sistema, para ello se tienen las siguientes opciones.

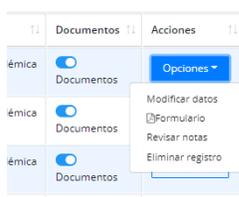


Para hacer el listado se tienen las opciones de filtrar los datos por modalidad, carrera y gestión. Una vez seleccionados las opciones que nos piden, podemos listar los datos.



#	Carnet Identidad	Apellidos y Nombres	Celular	Modalidad	Gestión	Fecha inscripción	Numero deposito
1	8363701 LP	BALBOA NELSON JOSE	69765627	PRUEBA DE SUFICIENCIA ACADÉMICA	2021-PRE-UNIV	2020-11-16	52935642
2	12574386 LP	CARVAJAL MISAEL ADALID	69783881	PRUEBA DE SUFICIENCIA ACADÉMICA	2021-PRE-UNIV	2020-11-09	48731256
3	BOLIVIANA LP	CHUQUIMIA ALEJANDRA	75840564	PRUEBA DE SUFICIENCIA ACADÉMICA	2021-PRE-UNIV	2020-11-17	53120168
4	9390372 CBB	CONDORI ROSABEL	73593436	PRUEBA DE SUFICIENCIA ACADÉMICA	2021-PRE-UNIV	2020-12-04	54711045
5	9939002 LP	MANCILLA GABRIELA SILVANA	77523358	PRUEBA DE SUFICIENCIA ACADÉMICA	2021-PRE-UNIV	2020-09-18	47132684
6	6833322 LP	MARCA MONICA CANDY	61216567	CURSO PREUNIVERSITARIO	2021-PRE-UNIV	2020-09-23	48179032
7	14917088 LP	MENDOZA JOEL ELOY	63266128	CURSO PREUNIVERSITARIO	2021-PRE-UNIV	2020-10-01	48880225

Se nos despliegan los registros de inscripciones y por cada registro se nos presenta una serie de opciones:



Modificar datos: nos permite realizar la modificación de los datos.

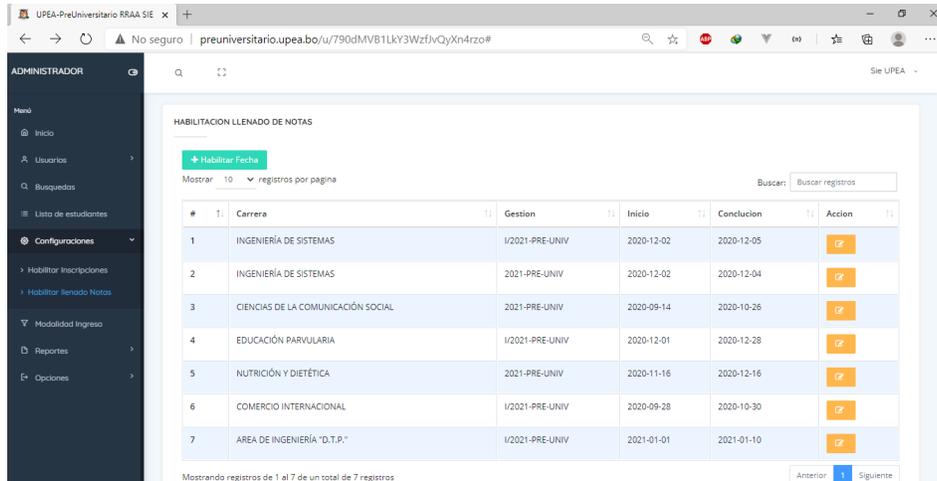
Formulario: nos permite imprimir el formulario.

Revisar notas: Permite ver las notas de los postulantes

Eliminar registro: Desactiva un registro.

Configuraciones

En esta opción se nos permite hacer la habilitación de fechas de inscripciones y llenado de notas de las diferentes carreras.



The screenshot shows a web application interface for 'UPEA-PreUniversitario RRAA SIE'. The main content area is titled 'HABILITACION LLENADO DE NOTAS' and features a '+ Habilitar Fecha' button. Below this is a table with 7 rows of data. The table has columns for '#', 'Carrera', 'Gestión', 'Inicio', 'Conclusion', and 'Accion'. Each row represents a different career and its associated dates for the 'I/2021-PRE-UNIV' session. The 'Accion' column contains an edit icon for each row. The table is displayed on a page that also includes a search bar and pagination controls.

#	Carrera	Gestión	Inicio	Conclusion	Accion
1	INGENIERÍA DE SISTEMAS	I/2021-PRE-UNIV	2020-12-02	2020-12-05	[Edit]
2	INGENIERÍA DE SISTEMAS	2021-PRE-UNIV	2020-12-02	2020-12-04	[Edit]
3	CIENCIAS DE LA COMUNICACIÓN SOCIAL	2021-PRE-UNIV	2020-09-14	2020-10-26	[Edit]
4	EDUCACIÓN PARVULARIA	I/2021-PRE-UNIV	2020-12-01	2020-12-28	[Edit]
5	NUTRICIÓN Y DIETÉTICA	2021-PRE-UNIV	2020-11-16	2020-12-16	[Edit]
6	COMERCIO INTERNACIONAL	I/2021-PRE-UNIV	2020-09-28	2020-10-30	[Edit]
7	AREA DE INGENIERÍA "D.T.P."	I/2021-PRE-UNIV	2021-01-01	2021-01-10	[Edit]

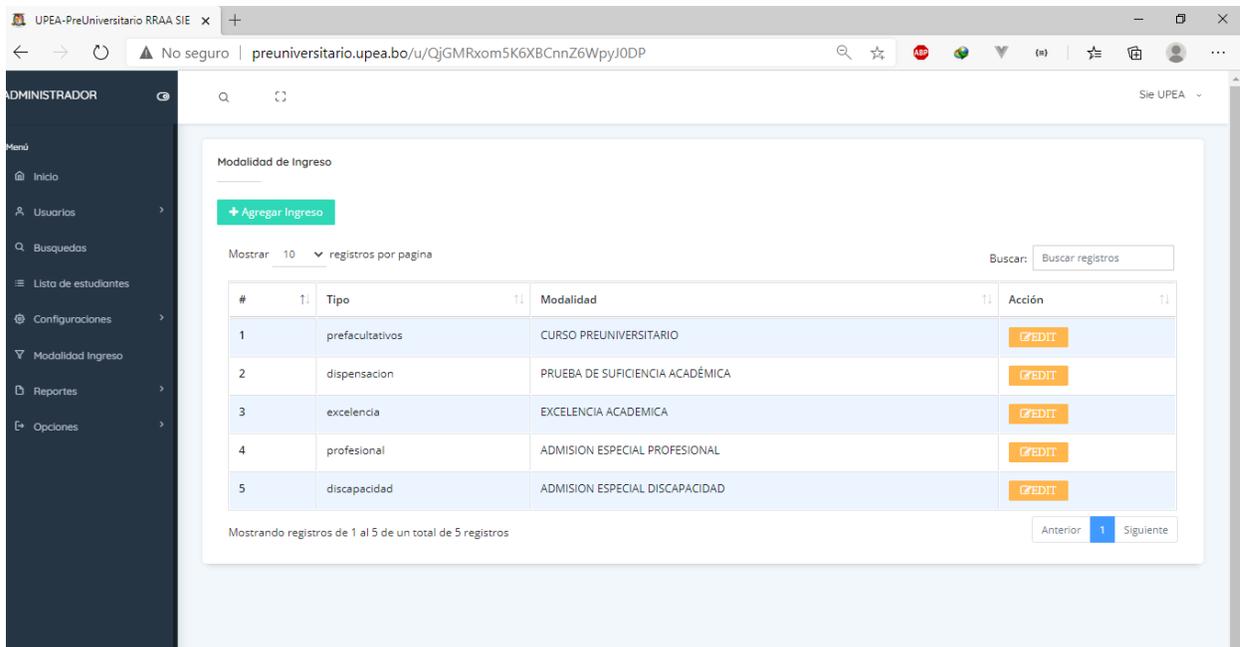
Al seleccionar cualquiera de las opciones para habilitación de nos desplegara la lista de las habilitaciones existentes. Aquí se nos presente dos opciones:

Habilitar fecha: Nos permite habilitar la fecha de inicio y conclusión ya sea de Inscripción o de Llenado de notas.

Editar fecha: Nos permite editar las fechas de habilitación.

Modalidad Ingreso:

Esta opción permite gestionar las modalidades dentro del sistema:



Las opciones que se tienen son las siguientes:

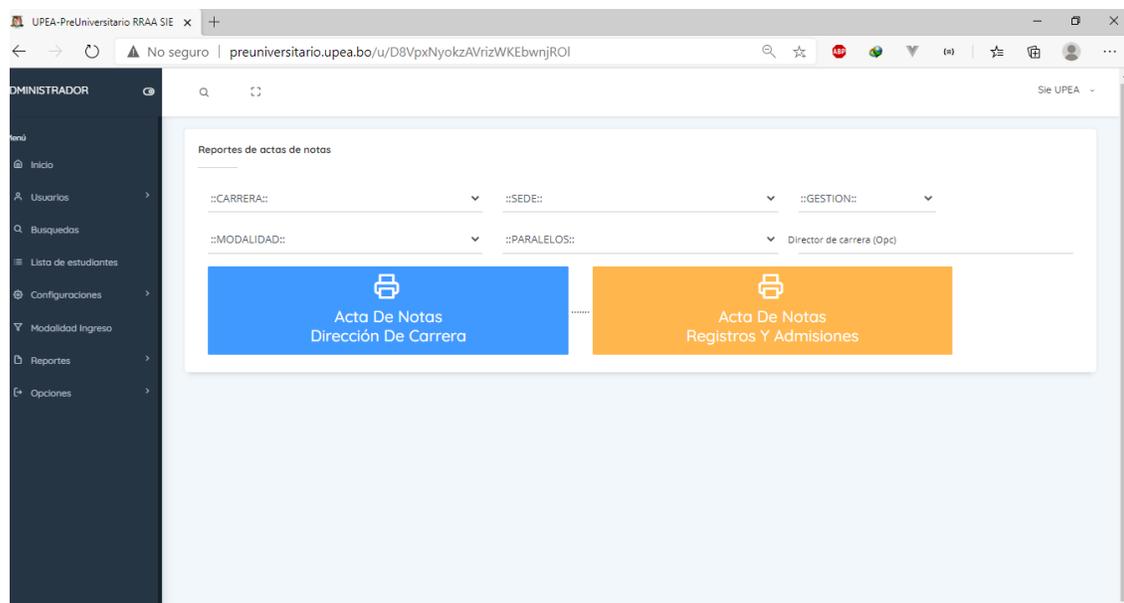
Agregar modalidad: Permite adicionar una nueva modalidad.

Editar: Permite editar la modalidad.

Esta parte debe manejarse con sumo cuidado, ya que es la parte primordial del sistema.

Reportes

Permite generar los reportes de acta de notas de todos los postulantes.



Para generar los reportes de actas de notas el sistema nos pedirá seleccionar las opciones de filtros.

Una vez seleccionado las opciones el sistema generará las actas:

ACTA DE CALIFICACIONES
PRUEBA DE SUFICIENCIA ACADÉMICA
 Área: CIENCIAS ECONÓMICAS, FINANCIERAS Y ADMINISTRATIVAS
 Carrera: ADMINISTRACIÓN DE EMPRESAS
 Director(a):
 Coordinador(a):
 Gestión: 2020

Página 1 de 11

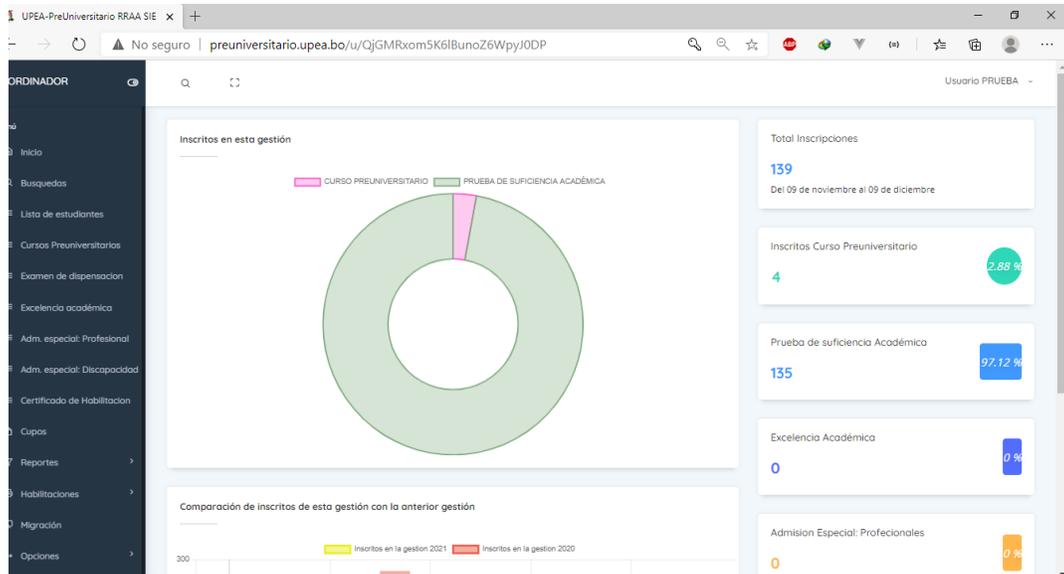
Nº DE IDENTIDAD	NOMINA DE POSTULANTES	C.F.	CALIFICACIÓN LITERAL	RESULTADO	
APPELLIDO PATERNO	APPELLIDO MATERNO	NOMBRE(S)	g/100		
1 8435029 LP	APAZA	LAURA	---	NO SE PRESENTO	
2 13408466 LP	CHURA	DAVID JOSUE	87	OCHENTA Y SIETE APROBADO	
3 9088838 LP	HUANCA	PATRICIA	25	VEINTICINCO REPROBADO	
4 9167170 LP	ACHU	FANNY CHRISTY	---	NO SE PRESENTO	
5 9962740 LP	ADUVIRI	GUAYGUA	61	SESENTA Y UNO APROBADO	
6 10077820 LP	AGUAYO	QUISPE	52	CINCUENTA Y DOS APROBADO	
7 9082336 LP	AGUILAR	AGUILAR	65	SESENTA Y CINCO APROBADO	
8 8368886 LP	AGUIRRE	SUKO	---	NO SE PRESENTO	
9 4066195 DR	AJAYA	CHOQUE	---	NO SE PRESENTO	
10 8349243 LP	AJLLAHUANCA	MAMANI	---	NO SE PRESENTO	
11 11093232 LP	AJPE	CHINO	EDGAR	55 CINCUENTA Y CINCO APROBADO	
12 7056213 LP	ALAVI	OSCORI	JUAN DANIEL	---	NO SE PRESENTO
13 7049477 LP	ALBORTA	AGUILAR	JHOVITH	---	NO SE PRESENTO
14 9149752 LP	ALBORTA	PACO	LISBETH	61 SESENTA Y UNO APROBADO	
15 13277911 LP	ALEGRIA	CAYO	JOHN KEVIN	75 SESENTA Y CINCO APROBADO	
16 9215456 LP	ALI	MAMANI	TANIA	---	NO SE PRESENTO
17 10945785 LP	ALI	PACASI	IVAN ADOLFO	---	NO SE PRESENTO
18 9148844 LP	ALMARAZ	MENDOZA	LEONOR JAEI	---	NO SE PRESENTO
19 13404676 LP	ALVAREZ	CASTRO	VLADMIR	56 CINCUENTA Y SEIS APROBADO	
20 9905462 LP	ALVAREZ	LAURA	GABRIELA	56 CINCUENTA Y SEIS APROBADO	
21 9905470 LP	ALVAREZ	VALENCIA	JOSE LUIS	52 CINCUENTA Y DOS APROBADO	

Opciones

Salir: Permite cerrar la sesión

MODULO COORDINADOR

Después de ingresar al sistema con el usuario y contraseña se nos mostrara la siguiente pantalla:



En la pantalla principal se nos muestra el número de estudiantes inscritos así como algunas estadísticas de las inscripciones.

A. Inicio

Aquí se nos muestra la pantalla principal, y las estadísticas del sistema.

B. Búsquedas



Busqueda de datos sobre los estudiantes...

Ingrese datos (CI)

Ingrese C.I.

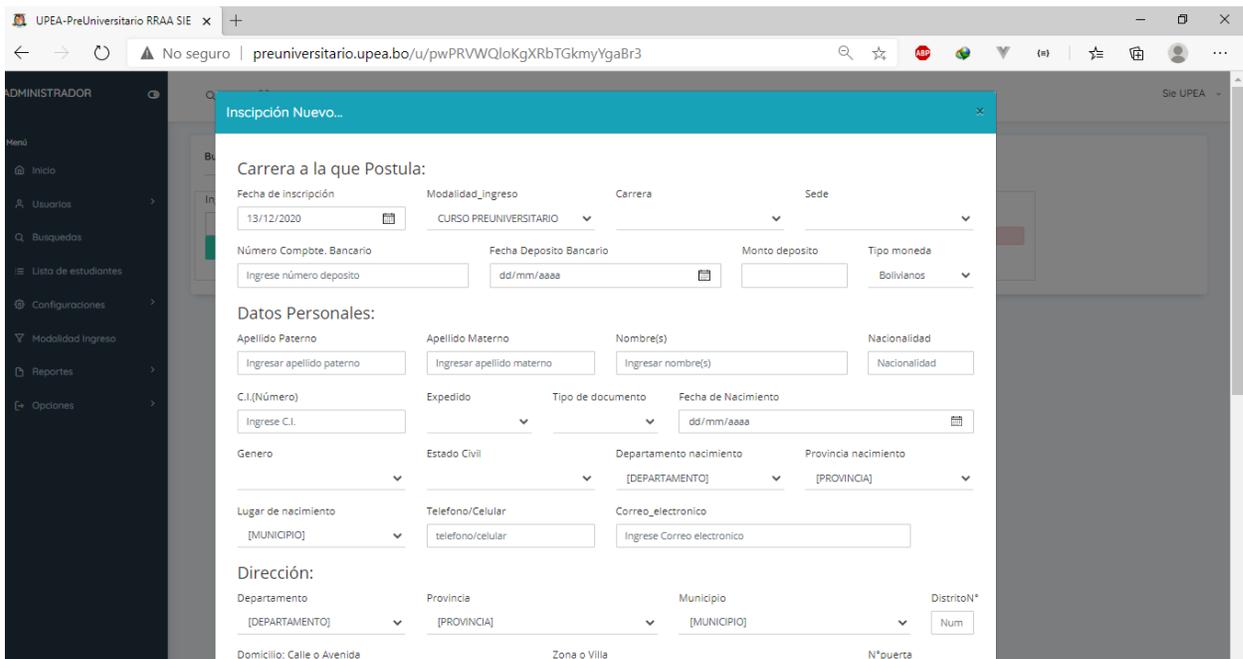
+ Buscar

Observaciones:

+ Inscripción Nuevo

Aquí es donde se hacen búsquedas de postulantes mediante la cedula de identidad del mismo. Una vez que se ingresen los datos el sistema desplegara la información del postulante. En caso de que el postulante no cumpla con los criterios para la admisión se mostrara un mensaje en la parte de observaciones.

Aquí también se puede hacer la inscripción de nuevos postulantes haciendo clic en la opción de Nueva Inscripción que mostrara la pantalla con el formulario de admisión estudiantil.



UPEA-PreUniversitario RRAA SIE x +

No seguro | preuniversitario.upea.bo/u/pwPRVWQloKgXRbTGkmyYgaBr3

ADMINISTRADOR

Inscripción Nuevo...

Carrera a la que Postula:

Fecha de inscripción: 13/12/2020

Modalidad_ingreso: CURSO PREUNIVERSITARIO

Carrera: [dropdown]

Sede: [dropdown]

Número Compte. Bancario: [input]

Fecha Deposito Bancario: dd/mm/aaaa

Monto deposito: [input]

Tipo moneda: Bolivianos

Datos Personales:

Apellido Paterno: [input]

Apellido Materno: [input]

Nombre(s): [input]

Nacionalidad: [input]

C.I.(Número): [input]

Expedido: [dropdown]

Tipo de documento: [dropdown]

Fecha de Nacimiento: dd/mm/aaaa

Genero: [dropdown]

Estado Civil: [dropdown]

Departamento nacimiento: [DEPARTAMENTO]

Provincia nacimiento: [PROVINCIA]

Lugar de nacimiento: [MUNICIPIO]

Telefono/Celular: [input]

Correo_electronico: [input]

Dirección:

Departamento: [DEPARTAMENTO]

Provincia: [PROVINCIA]

Municipio: [MUNICIPIO]

DistritoN*: [input]

Domicilio: Calle o Avenida

Zona o Villa

N°puerta

En la sección de búsqueda, cuando el sistema encuentra un postulante con el CI listado, mostrará el registro del postulante y las mostrará las opciones respectivas.

greso...	Carrera	Sede	Carnet Identidad	Nombres & Apellidos	Gestión	Acciones
IIVERSITARIO	ARQUITECTURA	VILLA ESPERANZA	123456789 LP	Perez Mamaní Juan	2021-PRE-UNIV	Editar Formulario ... Eliminar...
IIVERSITARIO	ADMINISTRACIÓN DE EMPRESAS	VILLA ESPERANZA	123456789 LP	Perez Mamaní Juan	2021-PRE-UNIV	Activar
EFICIENCIA ACADÉMICA	INGENIERÍA DE SISTEMAS	VILLA ESPERANZA	123456789 LP	Perez Mamaní Juan	2021-PRE-UNIV	Activar

Se tiene las siguientes opciones:

- **Editar:** Permite editar los datos del formulario.
- **Formulario:** Permite imprimir el formulario del postulante.
- **Icono de whatsapp:** Abre una ventana en whatsapp, con la opción de enviarle un mensaje por whatsapp al postulante.
- **Eliminar:** Se desactiva un registro
- **Activar:** Permite activar un registro.

C. Lista de estudiantes

Esta opción permite listar los postulantes registrados en el sistema, para ello se tienen las siguientes opciones.

Lista de Estudiantes

MODALIDAD: [v] Sede: [v] [Mostrar Lista](#)

Para hacer el listado se tienen las opciones de filtrar los datos por modalidad y la sede. Una vez seleccionados las opciones que nos piden, podemos listar los datos.

Usuario PRUEBA

Lista de Estudiantes

Todas las mod. ▼ Toda ▼ [Mostrar Lista](#)

[PDF](#) [Excel](#) Buscar:

#	Carnet Identidad	Apellidos y Nombres	Celular	Modalidad	Gestión	Fecha inscripción	Numero dep
1	8322345 LP	RAMOS MAURICIO CHRISTIAN	74866376	PRUEBA DE SUFICIENCIA ACADÉMICA	2021-PRE-UNIV	2020-12-02	54555472
2	6153395 LP	MAYTA CALIZAYA JOSE CARLOS	63227904	PRUEBA DE SUFICIENCIA ACADÉMICA	2021-PRE-UNIV	2020-12-04	54878770
3	10010215 LP	AGUIRRE COPA BRAYAN	70645512	PRUEBA DE SUFICIENCIA ACADÉMICA	2021-PRE-UNIV	2020-11-25	53836157
4	13297738 LP	ALANOCA APAZA MIRIAM	74031905	PRUEBA DE SUFICIENCIA ACADÉMICA	2021-PRE-UNIV	2020-12-07	55016153
5	11063967 LP	ALANOCA ULO MIRIAM NATALY	71517330	PRUEBA DE SUFICIENCIA ACADÉMICA	2021-PRE-UNIV	2020-12-07	55143502
6	14047209 LP	APAZA MICHEL NATHALIA BELEN	65600560	PRUEBA DE SUFICIENCIA ACADÉMICA	2021-PRE-UNIV	2020-11-19	53334912
7	9209665 LP	ATAUCURI VASQUEZ BRYAN ARIEL	73230721	PRUEBA DE SUFICIENCIA ACADÉMICA	2021-PRE-UNIV	2020-12-08	55119659

Se nos despliegan los registros de inscripciones de la gestión actual y por cada registro se nos presenta una serie de opciones:

INSCRIPCIÓN DE ▼ VILLA ▼ [Mostrar Lista](#)

Buscar:

Inscripción	Numero deposito	Fecha deposito	Paralelo	Documentos	Acciones
5	52935642	2020-11-16	Prueba de Suficiencia Académica	<input checked="" type="checkbox"/> Documentos	Opciones
3	48731256	2020-09-30	Prueba de Suficiencia Académica	<input checked="" type="checkbox"/> Documentos	Opciones
7	53120168	2020-11-17	Prueba de Suficiencia Académica	<input checked="" type="checkbox"/> Documentos	Opciones
4	54711045	2020-12-03	Prueba de Suficiencia Académica	<input checked="" type="checkbox"/> Documentos	Opciones
3	47132684	2020-09-10	Prueba de Suficiencia Académica	<input checked="" type="checkbox"/> Documentos	Opciones
3	48179032	2020-09-23		<input checked="" type="checkbox"/> Documentos	Opciones

Modificar datos: nos permite realizar la modificación de los datos.

Formulario: nos permite imprimir el formulario.

Revisar notas: Permite ver las notas de los postulantes

Eliminar registro: Desactiva un

registro.

D. Cursos preuniversitarios

En esta opción se nos muestra las materias que se llevarán en el curso preuniversitario.

Lista de Cursos

[Añadir Curso](#)

#	Gestion	Sigla / Asignatura	Sede	Docente	Paralelo	Turno	Estado	Acciones
1	I/2021-PRE-UNIV	PRIN-UNIV PRINCIPIOS UNIVERSITARIOS	VILLA ESPERANZA	MARIA GUISELA HUANCA TORREZ	A	Mañana	Activo	 
2	I/2021-PRE-UNIV	PRIN-UNIV PRINCIPIOS UNIVERSITARIOS	VILLA ESPERANZA	CLAUDIA ROCIO CONDORI CHOQUE	C	Tarde	Activo	 
3	I/2021-PRE-UNIV	SIS-ALG ÁLGEBRA	VILLA ESPERANZA	DIONICIO HENRY PACHECO RIOS	A	Mañana	Activo	 
4	I/2021-PRE-UNIV	SIS-ALG ÁLGEBRA	VILLA ESPERANZA	NEIL RAMIRO GONZALES BURGOA	C	Tarde	Activo	 
5	I/2021-PRE-UNIV	SIS-COM COMPUTACIÓN	VILLA ESPERANZA	WILLIAM ROQUE ROQUE	A	Mañana	Activo	 
6	I/2021-PRE-UNIV	SIS-COM COMPUTACIÓN	VILLA ESPERANZA	FRANCISCO LUIS PAEZ ROCHA	C	Tarde	Activo	 
7	I/2021-PRE-UNIV	SIS-FIS FÍSICA	VILLA ESPERANZA	RAMIRO KANTUTA LIMACHI	A	Mañana	Activo	 

Aquí se nos presenta una serie de opciones:

Añadir curso: Permite adicionar una materia al curso preuniversitario

Editar: Permite hacer la edición del registro de curso

Listar: nos muestra una lista con los postulantes asignados al paralelo al que pertenece el curso

E. Examen de dispensación

En esta opción se nos muestra una lista de todos los inscritos a la modalidad de prueba de suficiencia académica.

LISTA INSCRITOS A EXAMEN DE DISPENSACIÓN

#	Paterno	Materno	Nombre	Carrera	Sede	Celular	CI	Nota
1	.	TORRES	JHANNET XIMENA	INGENIERÍA DE SISTEMAS	VILLA ESPERANZA	79683976	11074583 LP	<input type="text" value="51"/>
2	.	CONDORI	SERGIO OLIVER	INGENIERÍA DE SISTEMAS	VILLA ESPERANZA	74841317	13424784 LP	<input type="text" value="24"/>
3	ACUÑA	CANAVIRI	ALVIN LIVAN	INGENIERÍA DE SISTEMAS	VILLA ESPERANZA	60501132	11079595 LP	<input type="text" value="24"/>
4	AGUILAR	CONDORI	ELMER JOSE	INGENIERÍA DE SISTEMAS	VILLA ESPERANZA	73207881	12638601 LP	<input type="text" value="54"/>
5	ANTE	SALINAS	DIEGO ARMANDO	INGENIERÍA DE SISTEMAS	VILLA ESPERANZA	63122917	8443815 LP	<input type="text" value="6"/>
6	ANTI	PONGO	YUOSETT	INGENIERÍA DE SISTEMAS	VILLA ESPERANZA	73086274	9176370 LP	<input type="text" value="28"/>
7	APAZA	APAZA	WILSON	INGENIERÍA DE SISTEMAS	VILLA ESPERANZA	70125098	10038658 LP	<input type="text" value="0"/>
8	APAZA	HUANCA	BENJAMIN JOSUE	INGENIERÍA DE SISTEMAS	VILLA ESPERANZA	63224884	10096016 LP	<input type="text" value="51"/>

Aquí se le presenta la opción de llenar o modificar las notas de los postulantes.

F. Excelencia académica

En esta opción se nos muestra la lista de todos los postulantes que se encuentren inscritos en la modalidad de excelencia académica.

LISTA DE EXCELENCIA ACADEMICA

#	CI	Paterno	Materno	Nombre	Sede	INFORME VCR	Fecha Emision	Celular
---	----	---------	---------	--------	------	-------------	---------------	---------

IMPRIMIR ACTA DE NOTAS

Se nos permite revisar o modificar el registro de informe de vicerrectorado

G. Adm. Especial: Profesionales

En esta opción se nos permite listar a los postulantes inscritos por la modalidad de admisión especial profesionales.

LISTA DE ADMISION ESPECIAL PROFESIONALES

#	CI	Paterno	Materno	Nombre	Sede	INFORME VCR	Fecha Emision	Celular
---	----	---------	---------	--------	------	-------------	---------------	---------

IMPRIMIR ACTA DE NOTAS

Se nos permite revisar y modificar los informes de vicerrectorado.

H. Adm. Especial Discapacidad

En esta opción se nos despliega la lista de los postulantes inscritos en la modalidad de admisión especial discapacidad.

LISTA DE ADMISION ESPECIAL DISCAPACIDAD

#	CI	Paterno	Materno	Nombre	Sede	INFORME VCR	Fecha Emision	Celular
---	----	---------	---------	--------	------	-------------	---------------	---------

IMPRIMIR ACTA DE NOTAS

Se nos permite revisar y modificar los informes de vicerrectorado.

I. Certificado de Habilitación

En esta opción se permite al coordinador generar los certificados de habilitación de los postulantes aprobados

Lista de Estudiantes

::MODALIDAD: ▾ -Sede ▾ -Gestio ▾ **Mostrar Lista**

Config. Margen superior: 2
 Config. Margen Izquierdo: 2

Después de elegir las opciones de filtrado. Se nos muestra la lista de los postulantes aprobados

Lista de Estudiantes

Todas las mod. ▼ Toda ▼ II/2020 ▼ [Mostrar Lista](#)

Config. Margen superior: 2
Config. Margen izquierdo: 2

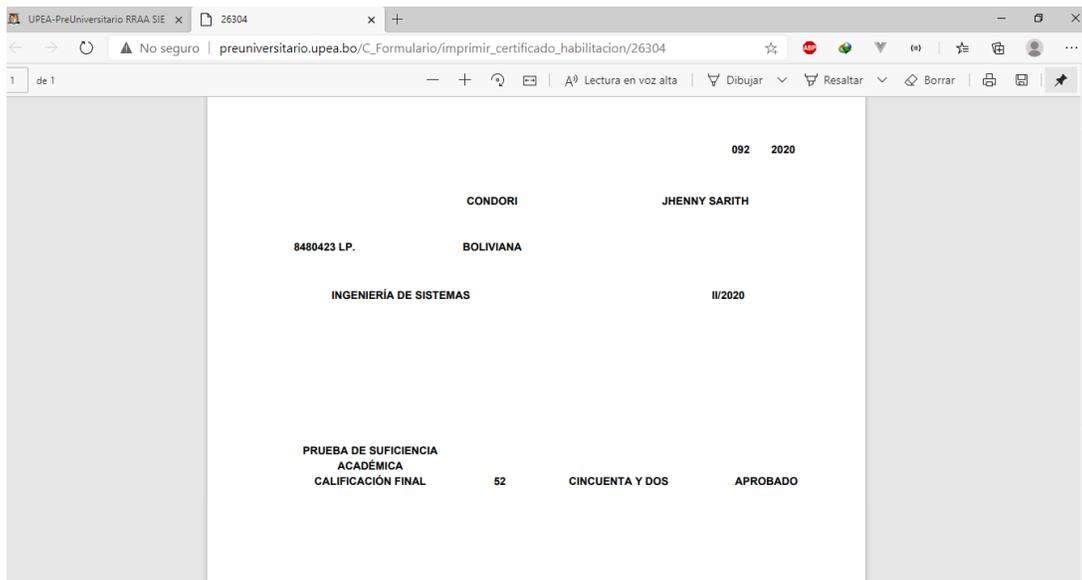
Mostrar 10 registros por página ▼ Buscar:

#	Carnet Identidad	Apellidos y Nombres	Nota	Modalidad	Paralelo	Hab.corre.	
1	8480423 LP	CONDORI JHENNY SARITH	52	PRUEBA DE SUFICIENCIA ACADÉMICA	(VILLA ESPERANZA)	92	<input checked="" type="checkbox"/> Imprimir
2	9070663 LP	COPAJIRA GUSTAVO RAMIRO	67	PRUEBA DE SUFICIENCIA ACADÉMICA	(VILLA ESPERANZA)	93	<input checked="" type="checkbox"/> Imprimir
3	12453600 LP	PACO GLADYS LAURA	86	PRUEBA DE SUFICIENCIA ACADÉMICA	(VILLA ESPERANZA)	94	<input checked="" type="checkbox"/> Imprimir
4	13377250 LP	ADUVIRI LARUTA JORGE ANTONIO	51	PRUEBA DE SUFICIENCIA ACADÉMICA	(VILLA ESPERANZA)	95	<input checked="" type="checkbox"/> Imprimir
5	14620920 LP	AGUILAR APAZA JOSE ANGEL	79	PRUEBA DE SUFICIENCIA ACADÉMICA	(VILLA ESPERANZA)		<input checked="" type="checkbox"/> Imprimir

Se nos presenta dos opciones.

Habilitar correlativo: Se habilita seleccionando la casilla. Lo que hace es habilitar el numero correlativo de impresión del postulante.

Imprimir: Genera el certificado de habilitación que debe ser impreso en los certificados de habilitación que se adquieren en valores:



J. Cupos

En esta opción el coordinador tiene la facultad de asignar cupos para las modalidades de inscripción, esta asignación es importante, ya que habilita el número de certificados de habilitación que se puede imprimir.

Cupos

[Añadir Cupo](#)

#	Carrera	Modalidad	Gestion	Cupo	Estado
1	INGENIERÍA DE SISTEMAS	CURSO PREUNIVERSITARIO	I/2021	<input type="text" value="200"/>	Activo
2	INGENIERÍA DE SISTEMAS	PRUEBA DE SUFICIENCIA ACADÉMICA	I/2021	<input type="text" value="101"/>	Activo

Se tienen las siguientes opciones:

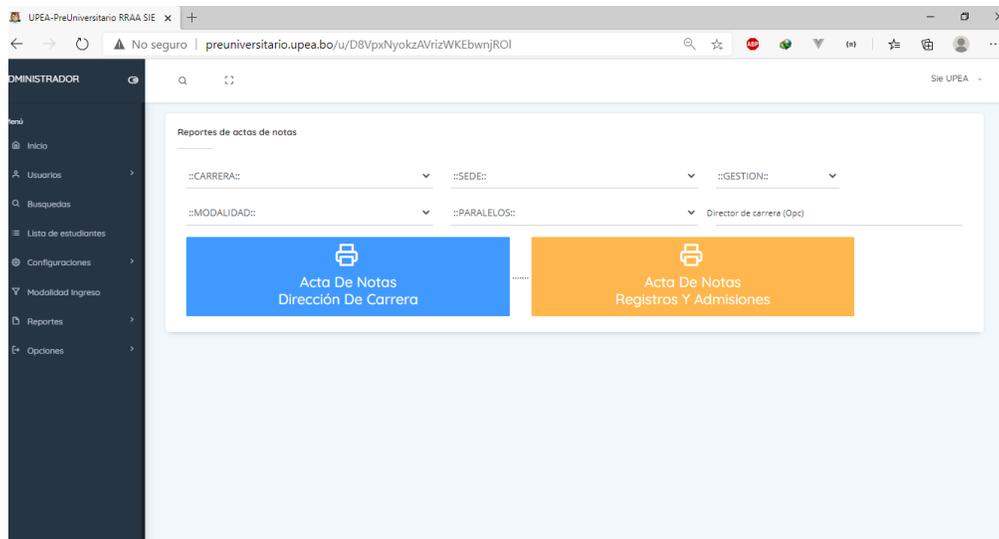
Añadir cupo: permite crear un nuevo cupo para una modalidad.

Cupos: permite modificar el número de cupos asignado.

K. Reportes

Acta de notas

Permite generar los reportes de acta de notas de todos los postulantes.



Para generar los reportes de actas de notas el sistema nos pedirá seleccionar las opciones de filtros.

Una vez seleccionado las opciones el sistema generará las actas:

UPEA-PreUniversitario RRAA SIE x PW82YQLvA8QGIAPZpg9odDyR x

preuniversitario.uepa.bo/u/PW82YQLvA8QGIAPZpg9odDyR

1 de 11

Universidad Pública de El Alto
Creada por Ley N° 2115 de 05 de Septiembre de 2009 y Autónoma por Ley N° 2556 de 12 de Noviembre de 2003

ACTA DE CALIFICACIONES
PRUEBA DE SUFICIENCIA ACADÉMICA

Área: CIENCIAS ECONÓMICAS, FINANCIERAS Y ADMINISTRATIVAS
Carrera: ADMINISTRACIÓN DE EMPRESAS
Director(a):
Coordinador(a):

Gestión: 2020

Página 1 de 11

N°	N° CEDULA DE IDENTIDAD	NOMINA DE POSTULANTES			C.F.	CALIFICACION LITERAL	RESULTADO
1	2	3	4	5	6	7	8
		APELLIDO PATERNO	APELLIDO MATERNO	NOMBRE(S)	s/100		
1	8435029 LP	APAZA		LAURA	---	---	NO SE PRESENTO
2	13408466 LP	CHURA		DAVID JOSUE	87	OCHENTA Y SIETE	APROBADO
3	9288338 LP	HUANCA		PATRICIA	25	VEINTICINCO	RESPONDIDO
4	9167170 LP	ACHU		LIMACHI	---	---	NO SE PRESENTO
5	9862740 LP	ADUVIRI		GUAYGUA	61	SESENTA Y UNO	APROBADO
6	10077820 LP	AGUAYO		QUISPE	52	CINCUENTA Y DOS	APROBADO
7	9062338 LP	AGUILAR		WILMER DAVID	---	---	APROBADO
8	9368886 LP	AGUIRRE		SUXO	---	---	NO SE PRESENTO
9	4066195 OR	AJATA		CHOQUE	---	---	NO SE PRESENTO
10	8348243 LP	AJLLAHUANCA		MAMANI	---	---	NO SE PRESENTO
11	1109232 LP	AJPE		CHINO	55	CINCUENTA Y CINCO	APROBADO
12	7056213 LP	ALAVI		OSCORI	---	---	NO SE PRESENTO
13	7049477 LP	ALBORTA		AGUILAR	---	---	NO SE PRESENTO
14	9149752 LP	ALBORTA		PACO	61	SESENTA Y UNO	APROBADO
15	13277911 LP	ALEGRIA		CAYO	75	SESENTA Y CINCO	APROBADO
16	9215456 LP	ALI		MAMANI	---	---	NO SE PRESENTO
17	10945785 LP	ALI		PACASI	---	---	NO SE PRESENTO
18	9148844 LP	ALMADAZ		MENDOZA	---	---	NO SE PRESENTO
19	13404676 LP	ALVAREZ		CASTRO	---	---	APROBADO
20	9905462 LP	ALVAREZ		LAURA	56	CINCUENTA Y SEIS	APROBADO
21	9905470 LP	ALVAREZ		VALENCIA	52	CINCUENTA Y DOS	APROBADO

L. Habilitaciones

En esta opción se nos permite hacer la habilitación de fechas de inscripciones y llenado de notas de las diferentes carreras.

UPEA-PreUniversitario RRAA SIE x

preuniversitario.uepa.bo/u/790dMVb1LkY3WzfvYqYXn4rzo#

ADMINISTRADOR

Inicio

Usuarios

Busquedas

Lista de estudiantes

Configuraciones

Habilitar Inscripciones

Habilitar llenado Notas

Modalidad Ingreso

Reportes

Opciones

HABILITACION LLENADO DE NOTAS

+ Habilitar Fecha

Mostrar 10 registros por pagina

Buscar: Buscar registros

#	Carrera	Gestion	Inicio	Conclusion	Accion
1	INGENIERÍA DE SISTEMAS	I/2021-PRE-UNIV	2020-12-02	2020-12-05	[Editar]
2	INGENIERÍA DE SISTEMAS	2021-PRE-UNIV	2020-12-02	2020-12-04	[Editar]
3	CIENCIAS DE LA COMUNICACIÓN SOCIAL	2021-PRE-UNIV	2020-09-14	2020-10-26	[Editar]
4	EDUCACIÓN PARVULARIA	I/2021-PRE-UNIV	2020-12-01	2020-12-28	[Editar]
5	NUTRICIÓN Y DIETÉTICA	2021-PRE-UNIV	2020-11-16	2020-12-16	[Editar]
6	COMERCIO INTERNACIONAL	I/2021-PRE-UNIV	2020-09-28	2020-10-30	[Editar]
7	AREA DE INGENIERÍA "D.T.P."	I/2021-PRE-UNIV	2021-01-01	2021-01-10	[Editar]

Mostrando registros de 1 al 7 de un total de 7 registros

Anterior 1 Siguiente

Al seleccionar cualquiera de las opciones para habilitación de nos desplegara la lista de las habilitaciones existentes. Aquí se nos presente dos opciones:

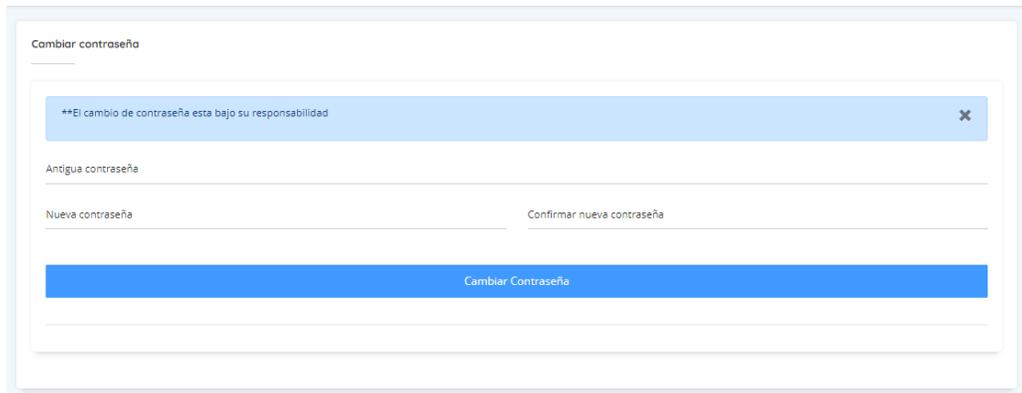
Habilitar fecha: Nos permite habilitar la fecha de inicio y conclusión ya sea de Inscripción o de Llenado de notas.

Editar fecha: Nos permite editar las fechas de habilitación.

M. Opciones

Aquí se nos permite las siguientes opciones

Cambio de contraseña:



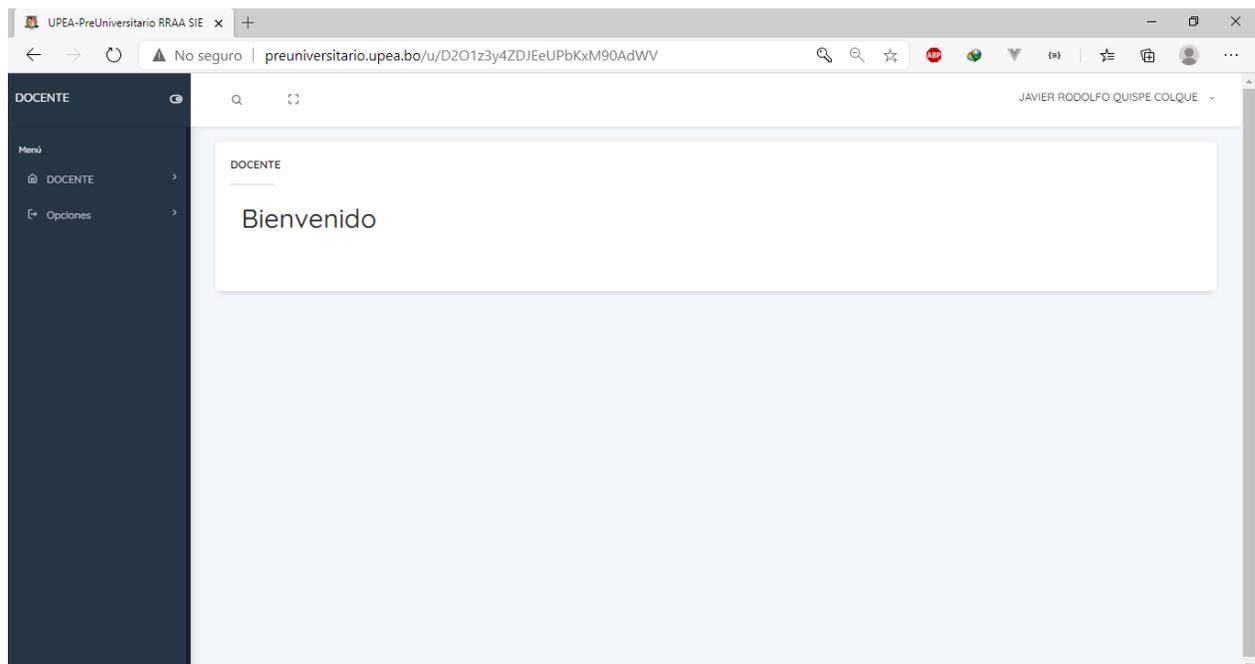
Formulario de "Cambiar contraseña" con un mensaje de advertencia: "**El cambio de contraseña esta bajo su responsabilidad". Incluye campos para "Antigua contraseña", "Nueva contraseña" y "Confirmar nueva contraseña", y un botón "Cambiar Contraseña".

Aquí se permitirá al usuario cambiar la contraseña, teniendo que poner su antigua contraseña.

Salir: Cierra la sesión actual y se nos redirige a la página principal

MODULO DOCENTE

Después de haber ingresado el usuario y contraseña se nos permitirá ingresar al módulo de docente mostrándonos la siguiente pantalla:

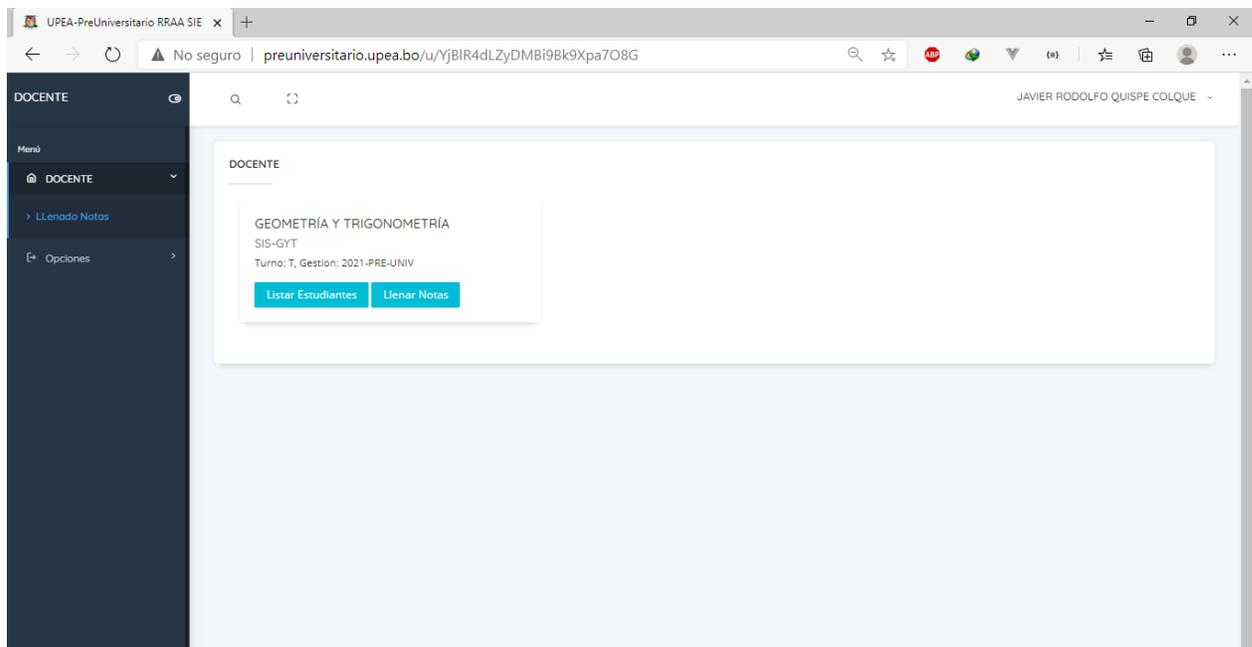


Se tienen las siguientes opciones:

A. Docente

Llenado de notas

Se muestran las materias que tiene asignado el docente.



Aquí se le permite al docente hacer las siguientes acciones:

Listar Estudiantes:

Lista de Estudiantes por curso

PDF Excel Search:

#	CI	Paterno	Materno	Nombres	Numero Celular
1	12451098		BAUTISTA	XIMENA FLORIANA	62334590
2	14917088		MENDOZA	JOEL ELOY	63266128
3	6096212	ACHA	MAMANI	GROVER GEOVANNI	67344785
4	12896838	AGUILAR	SIRPA	ALAN KEVIN	60631936
5	9110441	ALANOCA	ALTAMIRANO	ELA LOLA	63219226
6	13849566	ALANOCA	GUARACHI	RONALD	60548092
7	9905660	ALANOCA	LEON	DIEGO ENRRIQUE	61110629
8	9247936	ALANOCA	SOTO	CRISTIAN JOSE	61231032
9	13116435	AMARU	ARUQUIPA	EDWIN	78754728
10	8425852	AMARU	CHAMBI	MIGUEL ANGEL	69747566

Showing 1 to 10 of 224 entries

Previous 1 2 3 4 5 ... 23 Next

Aquí el docente podrá ver todos los postulantes asignados a su materia, así como imprimir reportes en Excel y pdf, también puede realizar búsquedas de postulantes.

Llenar notas:

Aquí se le permite al docente hacer el llenado de notas:

DISPENSACIÓN					
Cerrar Llenado De Notas					
#	Paterno	Materno	Nombre	CI	Nota
1		BAUTISTA	XIMENA FLORIANA	12451098 LP	<input type="text" value="40"/>
2		MENDOZA	JOEL ELOY	14917088 LP	<input type="text" value="40"/>
3	ACHA	MAMANI	GROVER GEOVANNI	6096212 LP	<input type="text" value="36"/>
4	AGUILAR	SIRPA	ALAN KEVIN	12896838 LP	<input type="text" value="56"/>
5	ALANOCA	ALTAMIRANO	ELA LOLA	9110441 LP	<input type="text" value="42"/>
6	ALANOCA	GUARACHI	RONALD	13849566 LP	<input type="text" value="44"/>
7	ALANOCA	LEON	DIEGO ENRIQUE	9905660 LP	<input type="text" value="80"/>
8	ALANOCA	SOTO	CRISTIAN JOSE	9247936 LP	<input type="text" value="16"/>

Para realizar el llenado de notas solo se debe poner el valor de la nota, que debe ir entre 0 y 100, y este se guardará automáticamente.

Una vez que se hayan llenado las notas, se tiene la opción para cerrar el llenado de notas. Una vez hecho esto ya no se podrá modificar las notas.

B. Opciones

Aquí se nos permite las siguientes opciones

Cambio de contraseña:

Cambiar contraseña

**El cambio de contraseña esta bajo su responsabilidad ✕

Antigua contraseña

Nueva contraseña Confirmar nueva contraseña

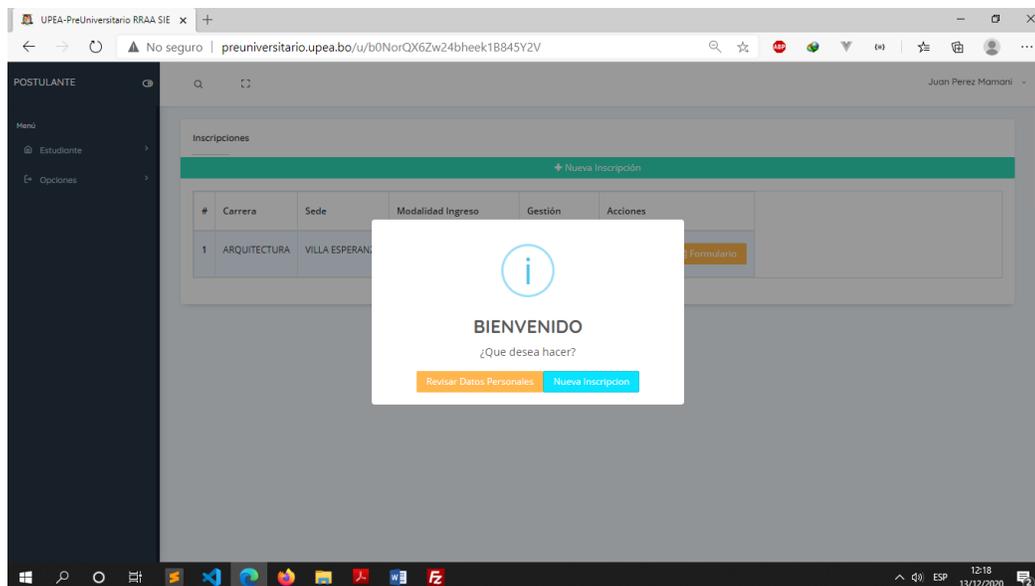
Cambiar Contraseña

Aquí se permitirá al usuario cambiar la contraseña, teniendo que poner su antigua contraseña.

Salir: Cierra la sesión actual y se nos redirige a la página principal

MODULO ESTUDIANTE

Después de ingresar un usuario y contraseña se desplegará la siguiente pantalla.



Se tiene las siguientes opciones:

A. Estudiante

En esta parte se le permite a los postulantes gestionar su información personal y la información de sus inscripciones:

Inscripciones: Esta opción permite gestionar las inscripciones del postulante

Inscripciones					
+ Nueva Inscripción					
#	Carrera	Sede	Modalidad Ingreso	Gestión	Acciones
1	ARQUITECTURA	VILLA ESPERANZA	CURSO PREUNIVERSITARIO	2021-PRE-UNIV	Editar Formulario

Se le presenta las siguientes opciones:

Nueva inscripción: Permite que el postulante realice una nueva inscripción.

Editar: Permite modificar los datos ingresados en el formulario.

Formulario: Permite imprimir el formulario 02 del postulante.

Datos personales: Permite al postulante la modificación de sus datos personales:

DATOS PERSONALES

Datos Personales:

Apellido Paterno <input type="text" value="Perez"/>	Apellido Materno <input type="text" value="Mamani"/>	Nombre(s) <input type="text" value="Juan"/>	Nacionalidad <input type="text" value="BOLIVIANA"/>
C.I. / RUN / Pas. (Número) <input type="text" value="123456789"/>	Expedido <input type="text" value="La paz"/>	Tipo de documento <input type="text" value="Cedula de Ide"/>	Fecha de Nacimiento <input type="text" value="08/05/1998"/>
Genero <input type="text" value="Femenino"/>	Estado Civil <input type="text" value="Soltero/a"/>	Departamento nacimiento <input type="text" value="LA PAZ"/>	Provincia nacimiento <input type="text" value="MURILLO"/>
Lugar de nacimiento <input type="text" value="La Paz"/>	Telefono/Celular <input type="text" value="78945612"/>	Correo_electronico <input type="text" value="lakjsfjasf@asdkljf"/>	

Dirección:

Departamento	Provincia	Municipio	DistritoNº
--------------	-----------	-----------	------------

B. Opciones

Aquí se nos permite las siguientes opciones

Cambio de contraseña:

Cambiar contraseña

**El cambio de contraseña esta bajo su responsabilidad

Antigua contraseña

Nueva contraseña Confirmar nueva contraseña

Cambiar Contraseña

Aquí se permitirá al usuario cambiar la contraseña, teniendo que poner su antigua contraseña.

Salir: Cierra la sesión actual y se nos redirige a la página principal

Anexo B. MODELOS DE ACTAS DE NOTAS Y CERTIFICADO DE HABILITACION



UNIVERSIDAD PÚBLICA DE EL ALTO

Creada por Ley 2115 del 5 de Septiembre del 2000 y Autónoma por Ley 2556 del 12 de Noviembre del 2003

Dentro del Sistema de la Universidad Boliviana

XI – CONGRESO NACIONAL DE UNIVERSIDADES RESOLUCIÓN N° 2/09

CERTIFICADO DE HABILITACIÓN ADMISIÓN ESTUDIANTIL

N° 000086

SERIE "B-001"

Costo Bs. 4

N° 002 / 2016

El(la) Postulante:

MAMANI	TORREZ	KAREN GABRIELA
<small>Apellido paterno</small>	<small>Apellido materno</small>	<small>Nombre(s)</small>
8302001 LP.	BOLIVIANA	CIENCIAS ECONÓMICAS, FINANCIERAS Y ADMINISTRATIVAS
<small>N° Cédula de identidad</small>	<small>Nacionalidad</small>	<small>Área</small>
ADMINISTRACIÓN DE EMPRESAS		Gestión: 2016
<small>Carrera</small>		

Habiendo obtenido el siguiente resultado:

MODALIDAD DE ADMISIÓN Y ASIGNATURAS	CALIFICACIÓN		RESULTADO
	NUMERAL	LITERAL	
PRUEBA DE SUFICIENCIA ACADÉMICA CALIFICACIÓN FINAL	89	OCHENTA Y NUEVE	APROBADO

En consecuencia con el resultado de aprobación obtenido por el(la) postulante, queda HABILITADO para tramitar su condición de Estudiante Regular en la Universidad Pública de El Alto.

El Alto, 29 de ENERO de 20 16

Coordinador(a)
o Director(a) de Carrera

Sello

Registros y Admisiones

Sello

ADVERTENCIA: Este documento queda nulo si en el hubiese hecho raspaduras, anotaciones o enmiendas.
 Nota: La aprobación de una modalidad de admisión es un requisito para ingresar a la Universidad Pública de El Alto y tiene validez por una sola gestión.
 ORIGINAL: Interesado.
 COPIAS: Registros y Admisiones y Dirección de Carrera

COPIA 3



Universidad Pública de El Alto

El Alto, 1 de Diciembre de 2015. Aprobado el 20 de Noviembre de 2015. Expediente No. 2015/0112/01. Versión 04 de 2015

ACTA DE CALIFICACIONES CURSO PREUNIVERSITARIO

Area: DESARROLLO TECNOLÓGICO PRODUCTIVO
Carrera:
Asignatura:
Asignatura:
Asignatura:
Asignatura: PRINCIPIOS UNIVERSITARIOS
Coordinador(a):

Gestión: 2016
Paralelo: "B"
Docente:
Docente:
Docente:
Facilitador(a):



N° de Nombramiento del(a) Coordinador(a):

Página 1 de 1

N°	N° DE CÉDULA DE IDENTIFICACION	NÓMINA DE POSTULANTES			N° DE CÉDULA DE IDENTIFICACION AL TRABAJAR	N° DE CÉDULA DE IDENTIFICACION AL SERVICIO	N° DE CÉDULA DE IDENTIFICACION AL SERVICIO	N° DE CÉDULA DE IDENTIFICACION AL SERVICIO	C.F. x/100 pts. (PROMEDIO)		RESULTADO
		APELLIDO PATERNO	APELLIDO MATERNO	NOMBRE(S)					NUMERAL	CALIFICACIÓN LITERAL	
1											NSP
2											NSP
3											NSP
4											NSP
5											NSP
6											NSP
7											NSP
8											NSP
9											NSP
10											NSP
11											NSP
12											NSP
13											NSP
14											NSP
15											NSP
16											NSP
17											NSP
18											NSP
19											NSP
20											NSP
21											NSP
22											NSP
23											NSP
24											NSP
25											NSP
26											NSP
27											NSP
28											NSP
29											NSP
30											NSP
31											NSP
32											NSP
33											NSP
34											NSP
35											NSP
36											NSP
37											NSP
38											NSP
***	XXXXXXXXXX	XXXXXXXXXXXXXX	XXXXXXXXXXXXXX	XXXXXXXXXXXXXX	XXXXX	XXXXX	XXXXX	XXXXX	XXXXX	XXXXXXXXXXXXXXXXXXXX	XXXXX

ADVERTENCIA: Este documento queda nulo si en él hubiese hecho reparaciones, adiciones o modificaciones

Lugar y Fecha: El Alto, 09 de Diciembre de 2015

C.F. x/100 pts = Calificación Final sobre 100 puntos
RESULT = Resultado
APR = Aprobado
REPR = Reprobado
NSP = No se Presentó

CUADRO RESUMEN	N°	%
APROBADOS	0	0.0%
REPROBADOS	0	0.0%
NO SE PRESENTÓ	38	100.0%
TOTAL POSTULANTES	38	100%

SELLO
CARRERA

COORDINADOR(A)

PROFESOR EN CARRERA Y/O FACULTAD DE AREA



Universidad Pública de El Alto

Creada por Ley 1115 del 5 de Septiembre de 2000 y Autónoma por Ley 2536 del 12 de Noviembre de 2003

ACTA DE CALIFICACIONES PRUEBA DE SUFICIENCIA ACADÉMICA

Área: DESARROLLO TECNOLÓGICO PRODUCTIVO

Carrera:

Director(a):

Gestión: 2016

Coordinador(a):

N° de Nombramiento del(a) Coordinador(a):

Página 1 de 1

N°	N° CÉDULA DE IDENTIDAD	NÓMINA DE POSTULANTES			CALIFICACIÓN FINAL SOBRE 100 PUNTOS		RESULTADO
		APELLIDO PATERNO	APELLIDO MATERNO	NOMBRE(S)	NUMERAL	CALIFICACIÓN LITERAL	
1							NO SE PRESENTÓ
2							NO SE PRESENTÓ
3							NO SE PRESENTÓ
4							NO SE PRESENTÓ
5							NO SE PRESENTÓ
6							NO SE PRESENTÓ
7							NO SE PRESENTÓ
8							NO SE PRESENTÓ
9							NO SE PRESENTÓ
10							NO SE PRESENTÓ
11							NO SE PRESENTÓ
12							NO SE PRESENTÓ
13							NO SE PRESENTÓ
14							NO SE PRESENTÓ
15							NO SE PRESENTÓ
16							NO SE PRESENTÓ
17							NO SE PRESENTÓ
18							NO SE PRESENTÓ
19							NO SE PRESENTÓ
20							NO SE PRESENTÓ
21							NO SE PRESENTÓ
22							NO SE PRESENTÓ
23							NO SE PRESENTÓ
24							NO SE PRESENTÓ
25							NO SE PRESENTÓ
26							NO SE PRESENTÓ
27							NO SE PRESENTÓ
28							NO SE PRESENTÓ
29							NO SE PRESENTÓ
30							NO SE PRESENTÓ
31							NO SE PRESENTÓ
32							NO SE PRESENTÓ
33							NO SE PRESENTÓ
34							NO SE PRESENTÓ
35							NO SE PRESENTÓ
36							NO SE PRESENTÓ
37							NO SE PRESENTÓ
38							NO SE PRESENTÓ
xxx	XXXXXXXXXXXX	XXXXXXXXXXXXXXXX	XXXXXXXXXXXXXXXX	XXXXXXXXXXXXXXXXXXXXXXXX	XXXXXX	XXXXXXXXXXXXXXXXXXXXXXXXXXXX	XXXXXXXXXXXX

ADVERTENCIA: Este documento queda nulo si en el hubiese hecho raspaduras, anotaciones o enmiendas.

Lugar y Fecha: El Alto, 09 de Diciembre de 2015

CUADRO RESUMEN	N°	%
APROBADOS	0	0.0%
REPROBADOS	0	0.0%
NO SE PRESENTÓ	38	100.0%
TOTAL POSTULANTES	38	100%