

UNIVERSIDAD PÚBLICA DE EL ALTO

CARRERA INGENIERÍA DE SISTEMAS



PROYECTO DE GRADO

“SISTEMA DE REGISTRO DE ADMISIÓN DE ESTUDIANTES DE LA MODALIDAD DE EXCELENCIA ACADÉMICA”

CASO: VICERRECTORADO – UPEA

Para Optar al Título de Licenciatura en Ingeniería de Sistemas

MENCIÓN: INFORMÁTICA Y COMUNICACIONES

Postulante: Univ. Karen Yhoselin Huaqui Quispe

Tutor Metodológico: Ing. Enrique Flores Baltazar

Tutor revisor: Ing. Yolanda Escobar Mancilla

Tutor Especialista: Ing. Santos Aurelio Limachi Huanca

**EL ALTO – BOLIVIA
2021**

DEDICATORIA

A la persona que fue mi inspiración a realizar el proyecto y fue un pilar fundamental para desarrollar, cumplir con uno de los sueños más importantes de mi vida, por creer en mi capacidad, por su apoyo y confianza quien fue la fortaleza para continuar con este proyecto de grado. (308)

AGRADECIMIENTO

A DIOS por su misericordia, por darme gracia e infinita sabiduría por estar siempre a mi lado, brindarme salud, alegrías, tristezas que me tocó vivir y mucha fortaleza para seguir adelante y superar todas las adversidades para alcanzar mis metas y anhelos

A mi madre Martina Sinforosa Quispe Sanchez quien ha sido fundamental en mi educación, por ser siempre la fortaleza de salir adelante sin importar los obstáculos, por brindarme la confianza y motivación durante el recorrido de toda la carrera. No hay palabras en este mundo para agradecerte, por tanto, muchas gracias mamá.

A mis hermanos y hermanas quienes me han brindado su amor, cariño y la confianza de que lograría culminar la carrera.

Un profundo agradecimiento a mi tutor metodológico Ing. Enrique Flores Baltazar de igual manera agradecerle por su apoyo, por apoyarme con su conocimiento, colaboración y correcciones para la conclusión de este proyecto.

Un especial agradecimiento a mi tutor especialista Ing. Santos Aurelio Limachi Huanca, por todas sus enseñanzas, por el conocimiento compartido y la paciencia prestada, por toda su colaboración y tiempo brindado para el desarrollo del sistema como para la culminación de este proyecto.

Un profundo agradecimiento a mi tutora revisora Ing. Yolanda Escobar Mancilla por todo su apoyo, por el tiempo dedicado a todas las correcciones brindadas, en el desarrollo de este proyecto.

A la unidad de SIE por acogerme en su ambiente de desarrollo y a los ingenieros miembros de tan maravillosa unidad, por brindarme el conocimiento necesario para crecer como profesional.

Finalmente, A la carrera de Ingeniería de Sistemas por acogerme durante estos años y brindarme el conocimiento en sus instalaciones y aulas, donde he adquirido los conocimientos necesarios para poder desempeñarme con éxito en el futuro de toda mi vida, no sólo profesional sino también personal.

RESUMEN

El presente proyecto sistema de registro de admisión de estudiantes de la modalidad de excelencia académica se desarrolló para la Universidad Pública de El Alto en atención a la problemática del registro de los postulantes de la modalidad de excelencia académica para mejorar administración del proceso de admisión estudiantil; centralizando la información de los postulantes, como responder a necesidad de imprimir su formulario, generar reportes de listas de admisión, generar la resolución de vicerrectorado, la generación de certificados de habilitación por parte de los coordinadores.

En la UPEA se tiene un gran movimiento de información y documentación mismos que son registrados y controlados de manera manual y descentralizada.

Para realizar el proyecto se utilizó la metodología 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, se empleó el framework CodeIgniter, que utiliza el modelo de desarrollo MVC que facilita la organización de los archivos del sistema, para un funcionamiento más dinámico del sistema. Para la base de datos se empleó el gestor de base de datos MySQL, para el maquetado del sistema se empleó el uso de Bootstrap4 y jQuery. Después de terminar el desarrollo del sistema de registro de admisión de estudiantes de la modalidad de excelencia académica se procedió a realizar las pruebas de funcionamiento, calidad y seguridad en base a las normas ISO/IEC 25000 e ISO 27000, con las que se pudo constatar que el sistema responde a los requerimientos de la unidad de vicerrectorado de la Universidad Pública de El Alto.

De esta manera se logró proporcionar información confiable, segura y oportuna a los coordinadores, técnicos y a los postulantes de la modalidad de excelencia académica. 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: Software libre Gestión Académica, Excelencia

ABSTRACT

The present project for the registration system for the admission of students of the academic excellence modality was developed for the Public University of El Alto in response to the problem of registering applicants for the academic excellence modality to improve the administration of the student admission process; centralizing the information of the applicants, such as responding to the need to print their form, generate reports of admission lists, generate the resolution of the vice-chancellor, the generation of qualification certificates by the coordinators.

In the UPEA there is a great movement of information and documentation that are registered and controlled in a manual and decentralized way.

The UWE methodology was used to carry out the project, which provides the necessary tools for the development of the web-oriented system. For the development of the system, the PHP tools were used, the CodeIgniter framework was used, which uses the MVC development model that facilitates the organization of the system files, for a more dynamic operation of the system. For the database, the MySQL database manager was used, for the system layout the use of Bootstrap4 and jQuery was used. After completing the development of the student admission registration system of the academic excellence modality, the performance, quality and safety tests were carried out based on the ISO / IEC 25000 and ISO 27000 standards, with which it was possible to verify that the system responds to the requirements of the vice-rector unit of the Public University of El Alto.

In this way, it was possible to provide reliable, safe and timely information to the coordinators, technicians and applicants of the academic excellence modality. 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: Free Software Academic Management, Excellence

ÍNDICE DE GENERAL

CAPÍTULO I

MARCO PRELIMINAR	PÁG.
1.1. INTRODUCCIÓN	1
1.2. ANTECEDENTES	2
1.2.1. Antecedentes de la Institución	2
1.2.2. Antecedente Internacional	5
1.2.3. Antecedente nacional	5
1.2.4. Antecedente local	6
1.3. PLANTEAMIENTO DEL PROBLEMA	7
1.3.1. Formulación del problema	7
1.3.2. Problema General	8
1.3.3. Problemas Específicos	8
1.4. OBJETIVOS.....	8
1.4.1. Objetivo General	8
1.4.2. Objetivos Específicos.....	9
1.5. JUSTIFICACIÓN.....	9
1.5.1. Justificación Técnica.....	9
1.5.2. Justificación Económica	9
1.5.3. Justificación Social	10
1.6. METODOLOGÍA	10
1.6.1. Metodología de Desarrollo UWE Based Web Engineering	10
1.6.2. Métricas de Calidad de Software	11
1.6.3. Métodos de Estimación de Costos	12
1.6.4. Seguridad	12

1.6.5. Pruebas al software	12
1.6.6. Métodos de obtención de datos	13
1.7. HERRAMIENTAS	13
1.8. LÍMITES Y ALCANCES	15
1.8.1. Límites	15
1.8.2. Alcances	15
1.9. APORTES	16
CAPÍTULO II	
MARCO TEÓRICO	
2.1. INTRODUCCIÓN	17
2.2. SISTEMA	17
2.2.1. Tipos de sistema	17
2.2.2. Estructura de los sistemas	18
2.3. GESTIÓN ACADÉMICA	20
2.4. EXCELENCIA	20
2.5. REGISTRO	21
2.6. REGISTRO DE ADMISIÓN DE ESTUDIANTES	21
2.7. REGISTRO DE ADMISIÓN EXCELENCIA ACADÉMICA	23
2.8. INTERNET	23
2.8.1. Ventajas y desventajas del internet	25
2.9. WEB 2.0	25
2.9.1. Características de la Web 2.0	26
2.9.2. Usos de la Web 2.0	27
2.10. INGENIERÍA DE SOFTWARE	27
2.10.1. Etapas de la ingeniería de software	27

2.10.2. Fases de la ingeniería de software	29
2.10.3. Capas de la ingeniería de software	30
2.11. INGENIERÍA WEB	31
2.11.1. Características de la Ingeniería Web	32
2.12. METODOLOGÍA INGENIERÍA WEB BASADA EN UML (UWE)	34
2.12.1. Características	35
2.12.2. Modelos de la metodología UWE	36
2.12.3. Fases de la metodología UWE	37
2.13. ARQUITECTURA DE SOFTWARE (MVC)	41
2.13.1. Modelo	42
2.13.2. Vista	43
2.13.3. Controlador	43
2.13.4. Características del Modelo Vista Controlador	44
2.14. HERRAMIENTAS	45
2.14.1. MagicDraw	45
2.14.2. Gestor de base de datos MySql	48
2.14.3. Lenguaje de Programación PHP	51
2.14.4. Framework CodeIgniter	52
2.14.5. Bootstrap	54
2.14.6. JavaScript	57
2.14.7. jQuery	58
2.14.8. JSON	61
2.14.9. Servidor Apache	62
2.15. MÉTRICAS DE CALIDAD DE SOFTWARE	63
2.15.1. Calidad de Software	64

2.15.2. Norma ISO/IEC 25000	65
2.15.3. Clasificación de las características y sub características	69
2.16. ANÁLISIS DE COSTO DE SOFTWARE COCOMO.....	73
2.16.1. Cocomo II	73
2.16.2. Características	73
2.16.3. Ecuaciones del modelo cocomo.....	73
2.16.4. Modelos de Cocomo.....	74
2.17. SEGURIDAD.....	79
2.17.1. SGSI (Sistema de Gestión de la Seguridad de la Información) ..	79
2.17.2. Aspectos de seguridad.....	81
2.18. PRUEBAS DE SOFTWARE.....	82
2.18.1. Pruebas de caja negra	82
2.18.2. Pruebas de caja blanca	84
2.18.3. Pruebas de estrés	84
 CAPÍTULO III	
MARCO APLICATIVO	
3.1. INTRODUCCIÓN	85
3.2. ANÁLISIS DE LA SITUACIÓN ACTUAL.....	85
3.3. INGENIERÍA DE REQUERIMIENTOS	86
3.4. MODELO DE CASO DE USO.....	88
3.5. MODELO CONTENIDO	97
3.6. MODELO DE NAVEGACIÓN.....	98
3.7. MODELO DE PRESENTACION	99
3.8. MODELO DE IMPLEMENTACIÓN	102
3.9. PRUEBAS.....	111

3.9.1. Prueba de caja blanca	111
3.9.2. Prueba de caja negra	113
CAPÍTULO IV	
CALIDAD Y SEGURIDAD	
4.1. INTRODUCCIÓN	116
4.2. NORMA ISO 25000.....	116
4.2.1. Funcionalidad	116
4.2.2. Confiabilidad	120
4.2.3. Usabilidad.....	121
4.2.4. Eficiencia	123
4.2.5. Mantenibilidad.....	124
4.2.6. Portabilidad	125
4.3. SEGURIDAD.....	127
4.3.1. Seguridad Lógica.....	127
4.3.2. Seguridad Física	127
CAPÍTULO V	
ANÁLISIS DE COSTOS	
5.1. INTRODUCCIÓN	129
5.2. ANÁLISIS DE COSTOS.....	129
CAPÍTULO VI	
CONCLUSIÓN Y RECOMENDACIÓN	
6.1. CONCLUSIONES	133
6.2. RECOMENDACIONES	133
BIBLIOGRAFÍA	
ANEXO A	

ANEXO B

ANEXO C

ANEXO D

ÍNDICE DE FIGURAS

	PÁG.
Figura 1.1 Organigrama Institucional de la Universidad Pública de El Alto	4
Figura 2.1 Estructura General de un Sistema	19
Figura 2.2 Etapa de la ingeniería de software	28
Figura 2.3 Ingeniería de software y sus fases.....	30
Figura 2.4 Capas de la ingeniería de software	31
Figura 2.5 Proceso de la Ingeniería Web.....	34
Figura 2.6 Ingeniería Web Basada en UML.....	37
Figura 2.7 Modelo de Caso de Uso de la Ingeniería Web.....	38
Figura 2.8 Modelo de Caso de Uso de la Ingeniería Web.....	38
Figura 2.9 Modelo Contenido de la Ingeniería Web.....	39
Figura 2.10 Modelo Navegación de la Ingeniería Web	40
Figura 2.11 Modelo de Presentación de la Ingeniería Web	41
Figura 2.12 Flujo de MVC.....	44
Figura 2.13 Herramienta MagicDraw	48
Figura 2.14 Herramienta MySql	50
Figura 2.15 Herramienta Php	52
Figura 2.16 Herramienta CodeIgniter	54
Figura 2.17 Herramienta Bootstrap.....	57
Figura 2.18 Herramienta JavaScript	58
Figura 2.19 Herramienta JQuery	60
Figura 2.20 herramienta JSON.....	62
Figura 2.21 Servidor Apache	63
Figura 2.22 División de calidad de Software	68
Figura 2.23 Calidad del producto de software	69
Figura 3.1 Proceso de inscripción excelencia académica.....	85
Figura 3.2 Casos de uso general.....	88
Figura 3.3 Caso de uso para la admisión de usuarios	89
Figura 3.4 Caso de uso Administración de usuarios	90
Figura 3.5 Caso de uso administración de postulantes.....	92
Figura 3.6 Gestión de la modalidad excelencia académica	93
Figura 3.7 Caso de uso Habilitación de fechas.....	95

Figura 3.8 Caso de uso Gestión de reportes	96
Figura 3.9 Diagrama de Clases para el sistema de admisión excelencia.....	97
Figura 3.10 Modelo de navegación para administrador	98
Figura 3.11 Pantalla página principal.....	99
Figura 3.12 Pantalla de presentación de convocatorias.....	99
Figura 3.13 Pantalla de control de acceso al sistema	100
Figura 3.14 Pantalla de presentación del administrador	100
Figura 3.15 Pantalla de presentación técnico excelencia	101
Figura 3.16 Pantalla de presentación control de cupos	101
Figura 3.17 Pantalla de presentación estudiante	102
Figura 3.18 Pagina Inicio.....	102
Figura 3.19 Autenticación de Usuarios	103
Figura 3.20 Administración de pagina	103
Figura 3.21 Publicación de convocatorias disponibles.....	104
Figura 3.22 Inicio del administrador.....	104
Figura 3.23 Administración de usuarios.....	105
Figura 3.24 Administración de grupos	105
Figura 3.25 Administración técnico excelencia académica	106
Figura 3.26 Reportes.....	106
Figura 3.27 Habilitación de convocatorias	107
Figura 3.28 Control de cupos	107
Figura 3.29 Revisión de documentos	108
Figura 3.30 Seguimiento de excelencia académica	108
Figura 3.31 Registro de formulario para el postulante	109
Figura 3.32 Formulario de Registro	109
Figura 3.33 Administración del estudiante	110
Figura 3.34 Entrega de requisitos.....	110
Figura 3.35 Caja Blanca	111

ÍNDICE DE TABLAS

	PÁG.
Tabla 2.1 Comparación de internet, intranet y extranet.....	24
Tabla 2.2 Constantes de Costes	74
Tabla 2.3 Coeficientes de a,b,c,d de COCOMO II.....	75
Tabla 2.4 Ecuaciones nominales de Coste	76
Tabla 2.5 Atributo de Costes de Software.....	76
Tabla 2.6 Atributo de Costes de Hardware	77
Tabla 2.7 Atributo de Costes Personal.....	77
Tabla 2.8 Atributo de Costes de Proyecto.....	77
Tabla 2.9 Atributos de Costes Totales	78
Tabla 3.1 Herramientas de los requerimientos.....	86
Tabla 3.2 Clasificación de los roles.....	87
Tabla 3.3 Descripción del caso de uso admisión de postulantes	89
Tabla 3.4 Descripción de los casos de uso Administración de usuarios.....	91
Tabla 3.5 Descripción del caso de uso administración de postulantes.....	92
Tabla 3.6 Descripción del caso de uso de la modalidad excelencia académica...94	94
Tabla 3.7 Descripción del caso de uso Habilitación de fechas	95
Tabla 3.8 Descripción del caso de uso Gestión de reportes.....	96
Tabla 3.9 Caso de prueba de Ingreso al sistema	113
Tabla 3.10 Caso de prueba de registro de usuario	114
Tabla 3.11 Caso de prueba de registro de estudiante.....	114
Tabla 4.1 Número de Entradas de usuario.....	116
Tabla 4.2 Número de Salidas Usuario	116
Tabla 4.3 Número de Peticiones del Usuario	117
Tabla 4.4 Parámetros de medición de punto función	117
Tabla 4.5 Punto función sin ajustar	118
Tabla 4.6 Factores de Complejidad	118
Tabla 4.7 Parámetros de medición de usabilidad.....	122
Tabla 4.8 Usabilidad del sistema	122
Tabla 4.9 Escala de valores de eficiencia	123
Tabla 4.10 Valoración de la eficiencia del sistema.....	123
Tabla 4.11 Valores para la mantenibilidad	125

Tabla 4.12 Resultados	126
Tabla 5.1 Coeficientes de a,b,c,d de COCOMO II.....	130
Tabla 5.2 Ecuaciones de Cocomo II	130
Tabla 5.3 Calculo de Atributos FAE	130

**SISTEMA DE REGISTRO DE ADMISIÓN DE ESTUDIANTES
DE LA MODALIDAD DE EXCELENCIA ACADÉMICA
CASO: VICERRECTORADO UPEA**

CAPÍTULO I

MARCO PRELIMINAR



1.1. INTRODUCCIÓN

En la actualidad las organizaciones comprenden la importancia del uso de las tecnologías, el manejo de los sistemas que se han establecido como factores cruciales para la aceleración de procesos, logrando que sus actividades sean más rápidas, logrando que el manejo de grandes cantidades de información sean eficientes , ya que los distintos papeleos, tramites y otras actividades relacionadas con el manejo de información, que tradicionalmente son realizadas manualmente, ahora son realizados por distintos sistemas de información.

La importancia de poner en marcha los sistemas es para tener un control fiable de la información ya que este constituye el eje fundamental que mueve a todos los sistemas, además es esencial para el logro y alcance de los propósitos, la creciente complejidad de las instituciones de educación superior, la competitividad y la diversificación hacen de una mayor inclinación de los sistemas universitarios hacia la sociedad pueda ser un medio poderoso de estimular la sensibilidad de las instituciones para satisfacer las demandas. La Universidad Pública de El Alto está experimentando un crecimiento significativo respecto a la cantidad de estudiantes admitidos en sus distintas carreras, los mismos requieren nuevas formas de atención a sus requerimientos en cuanto a los procesos de carácter académico, así de esta manera los procesos de admisión estudiantil se deben automatizar.

El presente proyecto de grado hace énfasis en la Unidad de Vicerrectorado, el cual es una de las unidades con alta responsabilidad dentro de la Universidad Pública de El Alto, es encargado de administrar el proceso de admisión especial universitario, en ese marco se tiene por objeto el diseño y desarrollo de un sistema web de registro para estudiantes que ingresaran a esta casa de estudios superiores bajo la modalidad de excelencia académica, para mejorar el manejo de datos y agilizar el proceso de ingreso a la Universidad. Ante esta situación, este proyecto propone una solución que permita a la unidad de vicerrectorado UPEA que a través de un sistema web facilite el gestionar documentos y tener un entorno social 2.0 para la comunicación interna.

El desarrollo del sistema se llevará acabo utilizando la metodología de desarrollo web UWE, que utiliza las herramientas UML para la captura de requisitos y el modelado del proceso de admisión de los estudiantes bachilleres bajo la modalidad de excelencia académica para el ingreso a la Universidad Pública de El Alto. Se

utilizará la base de datos MySQL, el lenguaje PHP y el framework codeIgniter para el desarrollo del sistema, también se empleará las herramientas JQuery y el servidor apache para el sistema.

1.2. ANTECEDENTES

1.2.1. Antecedentes de la Institución

El 5 de septiembre de 2000, la UPEA fue creada como Universidad Pública mediante la Ley 2115. El 12 de noviembre de 2003, la UPEA se le considera universidad Plena y Autónoma mediante la Ley 2556.

En 1989 tras la firma de convenios entre institucionales sociales de El Alto y la Universidad Mayor de San Andrés (UMSA) esta casa de estudios creó, a medias, una “Facultad Técnica” con tres carreras técnicas, pero no a nivel licenciatura.

En abril del 2000 ante la convocatoria del Comité Interinstitucional Pro Universidad Autónoma de El Alto (presidida por el Obispado de El Alto) se inscriben más de 10.000 bachilleres y 300 profesionales para docencia ad honorem. Esta convocatoria convirtió al Sindicato de la Prensa de El Alto en una improvisada y efervescente oficina de inscripciones de preuniversitarios y docentes universitarios que concluyó en la multitudinaria marcha del 1ro de mayo de ese año. Al día siguiente comenzó a constituirse el primer Consejo Universitario Provisional (cogobierno universitario) constituido por delegados (un estudiante y un docente) de las nueve áreas o facultades) encabezados por el presidente del Consejo y el ejecutivo de la Federación Universitaria Local (FUL).

En septiembre de 2000 tras más de una veintena de marchas realizadas por dirigentes de base, organizaciones sociales, juntas vecinales, padres de familia, trabajadores, universitarios y docentes de la UPEA. El Congreso finalmente aprobó la Ley 2115 el 5 de septiembre del año 2000, aunque esta fue redactada a ocultas de los representantes de la Universidad y aprobada “por consigna”.

La Ley 2115 determina que la UPEA tendría autonomía dentro de 5 años y que mientras tanto estará a cargo de un “Consejo de Desarrollo Institucional” de la que forman parte el Ministerio de Educación y otros organismos gubernamentales, contradiciendo a la Constitución Política del Estado. El primer rector fue el representante del Ministerio de Educación, por entonces Tito Hoz de Vila. Entre los

responsables de este acuerdo son el ex ministro Walter Guiteras y el entonces senador Reynaldo Venegas. (Reyqui, upea.reyqui.com, 2016)

En noviembre de 2003 durante el gobierno de Carlos Mesa se pone en vigencia la ley que garantiza la autonomía universitaria de la UPEA.² La universidad ha sido un actor principal de las revueltas sociales durante los últimos años.

✓ **Misión**

"Formar profesionales integrales altamente calificados en todas las disciplinas del conocimiento científico-tecnológico, con conciencia crítica y reflexiva; capaz de crear, adaptar y transformar la realidad en que vive; desarrollar la investigación productiva para fomentar el desarrollo local, regional y nacional para que responda al encargo social y las necesidades de las nacionalidades de manera eficiente y oportuna hacia la transformación revolucionaria de la sociedad

✓ **Visión**

"La UPEA es una institución que se proyecta al desarrollo de sus actividades académico-productivas, científicas, tecnológicas de interacción social contemporáneo, para priorizar la investigación científica en todos los campos del conocimiento relacionando la teoría con la práctica para transformar la estructura económica, social, cultural y política vigente en favor de las naciones originarias y clases populares".

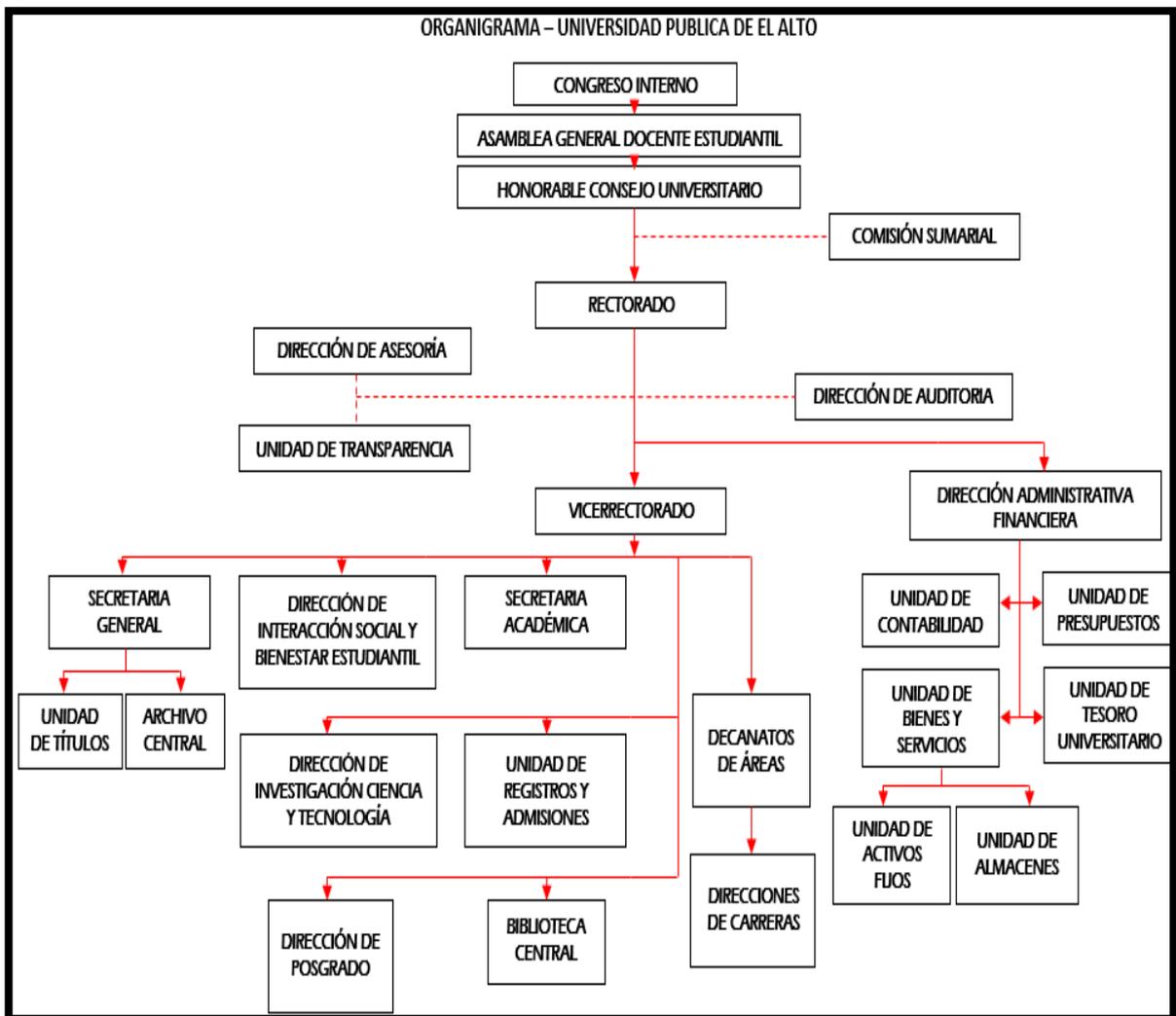
✓ **Fines y objetivo**

La Universidad Pública de El Alto, está en concordancia del Estatuto de la Universidad Boliviana, que fueron incorporados en su plenitud al Estatuto Orgánico de la UPEA.

- ✓ Formar Profesionales idóneos a partir del desarrollo de la ciencia, la tecnología y el conocimiento científico, en un proceso único de interacción entre la teoría y la práctica, que permita transformar y desarrollar la realidad local, regional, nacional promoviendo de múltiples formas del bienestar del pueblo boliviano.
- ✓ Formar profesionales con una concepción crítica contra hegemónica para el logro del poder político "de" y "para" las mayorías nacionales.
- ✓ Desarrollar y difundir ciencia, tecnología y cultura dentro y fuera de la universidad.

- ✓ Orientar, realizar y promover la investigación en todos los campos del conocimiento, conforme a la priorización de problemas de la realidad boliviana.
- ✓ Estimular, orientar, realizar programas de investigación científica.
- ✓ Mejorar continuamente el proceso enseñanza - aprendizaje.
- ✓ Defender, rescatar y desarrollar los valores, la ciencia, la cultura y el pensamiento de las naciones del país en un ambiente de diálogo con los valores de otras culturas.
- ✓ Defender los valores históricos nacionales.
- ✓ La UPEA debe dar respuestas a las necesidades e intereses de la sociedad boliviana en su conjunto.
- ✓ **Organigrama institucional**

Figura 1.1 Organigrama Institucional de la Universidad Pública de El Alto



Fuente: Unidad de vicerrectorado

Haciendo referencia a los trabajos realizados a nivel internacional, nacional y local se pueden citar a las siguientes:

1.2.2. Antecedente Internacional

- ✓ Haide Ponce P. (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”. **Objetivo General:** Implementar un sistema web para el instituto de postgrado UPSE que permita automatizar el proceso de admisión y matriculación. **Herramientas:** Utiliza el framework ZK y el lenguaje java para la realización del sistema de la Universidad Estatal Península de Santa Elena, Ecuador.
- ✓ Paola Andrea Blanco Blanco, Mauricio Hernández Zamudio (2016), “SISTEMA DE INFORMACIÓN PARA LA GESTIÓN DE PROYECTOS”. **Objetivo General:** El desarrollo de un sistema confiable y estable basado en la programación web bajo tecnología asp.net; el cual tiene administración y seguridad que permita gestionar los diferentes roles que se manejen en el sistema, dando la opción a los profesores y directivos de la institución tener un control sobre los proyectos de grado y las investigaciones desarrolladas por los diferentes alumnos. **Herramientas:** Utiliza el framework ZK y el lenguaje java para la realización del sistema de la Fundación Universitaria Los Libertadores, Bogota.
- ✓ Perman B. Nieves (2009), “DESARROLLO DE UN PORTAL WEB PARA LA GESTIÓN DE INFORMACIÓN DE INVESTIGACIÓN”. **Objetivo General:** El desarrollo de un portal web para la gestión de contenidos dentro de una universidad y que pueda adaptarse a cualquier otra universidad. **Herramientas:** Utilizando la tecnología Struts que permite desarrollar aplicaciones en Java, hace uso de la arquitectura MVC de la Universidad Carlos III, Madrid.

1.2.3. Antecedente nacional

- ✓ Julieta Quispe Calle (2017), “SISTEMA DE INFORMACIÓN ACADÉMICA VÍA WEB (SINA)”. **Objetivo General:** El brindar a los profesores y padres de familia de la Unidad Educativa un sistema de información académica a través de una plataforma web que ayude a automatizar los procesos repetitivos, que administre y organice de mejor manera la información. **Herramientas:** La metodología de desarrollo SCRUM y las herramientas de PHP, base de datos

MySQL, servidor Web apache HTTP 2.2 (opensource) de la Universidad Mayor de San Andrés, Bolivia.

- ✓ Marizol Roque Caizana (2014), "SISTEMA WEB DE INFORMACIÓN Y SEGUIMIENTO ACADÉMICO". **Objetivo General:** El desarrollar un sistema web que permita un adecuado control y registro de los procesos realizados por el programa de admisión pre facultativa. **Herramientas:** Utiliza la metodología AUP y el desarrollo del sistema web es llevado a cabo utilizando el framework Spring y el lenguaje Java de la Universidad Mayor de San Andrés, Bolivia.
- ✓ Jesus Marcelo Pinto Arze (2016), "SISTEMA WEB COLABORATIVO INTEGRADO DE CONTROL, ADMINISTRACIÓN Y SEGUIMIENTO BASADO EN GESTIÓN DE PROCESOS DE NEGOCIO". **Objetivo General:** El desarrollar un Sistema Web Colaborativo Integrado, basado en BPM utilizando la metodología de desarrollo ágil ASD para la empresa Educomser SRL, el cual integrará el flujo de información de los subsistemas: Control de personal, Seguimiento académico, Enseñanza virtual, Biblioteca digital, Control de cuentas por cobrar (pago de mensualidades y de servicios), Anuncios y noticias; para satisfacer las necesidades tanto de la empresa como la de los alumnos que ocupan los servicios de la misma. **Herramientas:** La metodología ágil de la Universidad Mayor de San Andrés, Bolivia.

1.2.4. Antecedente local

- ✓ Walter Emilio Siles Paco (2013), "SISTEMA DE INFORMACIÓN AUTOMATIZADO PARA EL CONTROL DE LA GESTIÓN Y ADMINISTRACIÓN DE BIENES Y SERVICIOS". **Objetivo General:** Desarrollo en una arquitectura cliente-servidor lo que significa que todas las gestiones que se realizan se concentran en el servidor de manera que en él se disponen los requerimientos provenientes de los usuarios. **Herramientas:** La metodología web que es UWE (UML-Based WEB Engineering), herramientas de MagigDraw, JQuery, php5, servidor apache, PostgreSQL de la Universidad Pública de El Alto, Bolivia.
- ✓ Nestor Calixto Chura Aguilar (2014), "SISTEMA DE SEGUIMIENTO ACADÉMICO Y GESTIÓN ADMINISTRATIVA CASO: UNIDAD DE POSTGRADO EN INFORMÁTICA". **Objetivo General:** El objetivo es que se pretende mejorar los procesos en la gestión, elaboración y emisión de información mediante el desarrollo de un sistema de seguimiento académico y gestión administrativa, aplicando Extrem Programing (XP) para encontrar la

solución informática adecuada. **Herramientas:** La metodología se siguen las distintas fases de desarrollo, cabe recalcar que se combinan las metodologías XP, WebML y diagramas UML. Realizó el desarrollo del sistema en el lenguaje PHP y usando el framework Laravel de la Universidad Mayor de San Andrés, Bolivia.

- ✓ Manfred Alejandro Maidana Balladares (2020), “IMPLEMENTACIÓN DE UN SISTEMA DE ADMINISTRACIÓN WEB PARA LA GESTIÓN DEL PROCESO DE ADMISIÓN ESTUDIANTIL CASO: UPEA- REGISTRO Y ADMISIONES”. **Objetivo General:** El desarrollo del portal web se realizará utilizando la metodología de desarrollo web UWE, que emplea las herramientas UML para la captura de requisitos y el modelado del proceso de admisión preuniversitario de la UPEA. **Herramientas:** Se empleará el lenguaje PHP y el framework codeigniter para facilitar el desarrollo del portal web, también se utilizará las herramientas de la WEB 2.0 como javascript, css, Ajax y json para un manejo más dinámico del sistema de la Universidad Pública de El Alto, Bolivia.

1.3. PLANTEAMIENTO DEL PROBLEMA

El incremento de la población estudiantil en la Universidad Pública de El Alto hace que la información sea excesiva y difícil de manejar adecuadamente, la Unidad de Vicerrectorado, atraviesa una diversidad de problemas respecto al manejo de documentación y procesamiento de la información que se debe realizar, también existen dificultades respecto al control y seguimiento de información que se realiza a los estudiantes que ingresan bajo la modalidad de excelencia académica para la admisión en la universidad.

Luego de realizar un análisis de problemas con la consecuente construcción del árbol de problemas (ver anexo A), permite identificar que existe la dificultad en la demora de tiempo por parte de la Unidad de Vicerrectorado en la admisión de estudiantes nuevos que concluyeron el nivel de 6to de secundaria de las unidades educativas, que desean ingresar a la universidad con admisión directa bajo la modalidad de excelencia académica.

1.3.1. Formulación del problema

De acuerdo a estos problemas surge la siguiente pregunta o interrogante.

¿De qué manera se podría agilizar y mejorar el registro de postulantes de admisión directa bajo la modalidad de excelencia académica en la unidad de Vicerrectorado de la Universidad Pública de El Alto?

1.3.2. Problema General

La Unidad de Vicerrectorado, no cuenta con un sistema dedicado al registro de la admisión de la modalidad de excelencia académica de los postulantes, este se realiza de manera manual lo que genera que la información se maneje de manera descentralizada y el control de la información sea poco eficiente, generando demoras en los tiempos de búsqueda y presentación de documentos para la gestión del proceso de admisión de los postulantes que ingresan bajo la modalidad de excelencia académica

1.3.3. Problemas Específicos

- El almacenamiento de la información es descentralizado lo que genera burocracia y demora en el manejo de datos.
- No se tiene un adecuado control y organización de los documentos de cada carrera de la postulación de excelencia.
- Perdida de información está sujeto a ser eliminada o extraviada, debido a que son manejadas de forma manual.
- No se tiene un control riguroso de estudiantes que ya se encuentren inscritos en otras carreras, lo que genera duplicidad en los datos, que causa que los estudiantes tengan que realizar papeleos y regularizaciones a futuro.
- Se usa desmesuradamente el papel al producir copias innecesarias de un mismo documento.

1.4. OBJETIVOS

El objetivo del presente proyecto nace como una respuesta a las necesidades de la institución, luego de realizar el análisis al árbol de problemas se ha realizado el árbol de objetivos (ver anexo B).

1.4.1. Objetivo General

Desarrollar e implementar un sistema de registro de estudiantes para la admisión de la modalidad de excelencia académica, para la Unidad de Vicerrectorado de la Universidad Pública de El Alto, a fin de lograr una eficiente administración en el manejo de documentación e información.

1.4.2. Objetivos Específicos

- Analizar la situación actual del proceso de admisión de la modalidad de excelencia académica mediante ingeniería de requerimientos.
- Implementar sistema web para facilitar el registro de información de los postulantes.
- Desarrollar la página dinámica y administrable con una interfaz agradable para el usuario.
- Verificar que los postulantes no estén registrados en la universidad.
- Almacenar los requisitos de admisión de manera digital.
- Generar el formulario de registro de los postulantes a la modalidad de excelencia académica, que sirva para generar la resolución emitida por la Unidad de Vicerrectorado de los postulantes habilitados.
- Automatizar la impresión de reportes que ayuden a tener las listas de los postulantes a la modalidad de excelencia académica.
- Realizar la habilitación de cupos de la modalidad de excelencia académica por carrera.

1.5. JUSTIFICACIÓN

1.5.1. Justificación Técnica

La Unidad de Vicerrectorado de la Universidad Pública de el Alto cuenta con recursos y herramientas como equipos de computación con buena capacidad en memoria, procesamiento, conexiones de red, servicio de internet y también la Universidad Pública de El Alto cuenta con servidores para poder llevar a cabo el desarrollo del sistema. En la parte de software se tienen instalado el sistema operativo Windows, con programas como: Office, PDF, antivirus y otros programas de desarrollo.

El sistema le permitirá a la Unidad de Vicerrectorado, el contar con un recurso tecnológico que facilite el desempeño de sus actividades en el manejo de la información de los estudiantes que ingresen a la universidad bajo la modalidad de excelencia académica, sirviendo así de una herramienta útil y fiable.

1.5.2. Justificación Económica

La información es un recurso económico de consideración en toda universidad, más aún si esta es manipulada en grandes volúmenes de datos, en la medida, en

que las mismas sean automatizadas implica ahorro en tiempo, reducción en material de escritorio, minimiza el trabajo manual.

El sistema será desarrollado bajo la premisa de software libre, que implica la no distribución de gastos en licencias de uso.

1.5.3. Justificación Social

La Unidad de Vicerrectorado, recibe cada año a nuevos postulantes para el ingreso por excelencia académica, los datos de los mismo son manipulados de forma manual, para el procedimiento del registro de los estudiantes son almacenadas en documentación física, en ese sentido el sistema que se propone tiene como finalidad la de coadyuvar con la Unidad de registro y admisiones, para mejorar el proceso de registro de los estudiantes mostrando confiabilidad en el manejo de datos e información, así mismo, reducirá el trabajo excesivo del personal, reducirá los tiempos de respuesta en la emisión de certificados de habilitación, reportes de altas, brindando a los bachilleres de excelencia, una adecuada atención.

1.6. METODOLOGÍA

1.6.1. Metodología de Desarrollo UWE Based Web Engineering

La metodología para el desarrollo del software se hará uso de la metodología UWE (UML-Based Web Engineering, en español (Ingeniería Web Basada en UML) es una metodología que permite modelar de mejor manera una aplicación Web, para el proceso de creación de aplicaciones detalla ésta, con una gran cantidad de definiciones, en el proceso de diseño lista que debe utilizarse. Procede de manera iterativa e incremental, coincidiendo con UML, incluyendo flujos de trabajo y puntos de control.

Modelo de aplicación web según la metodología UWE, se aplicará con las siguientes fases:

- **Modelo de Casos de Uso:** se modela requisitos funcionales de la aplicación Web para ver como interactúa cada uno de ellos.
- **Modelo Conceptual:** Materializa en un modelo de dominio, considerando los requisitos reflejados en los casos de uso.

- **Modelo Navegación:** Especifica el entorno en la cual se realizará el aspecto de navegación de la aplicación Web.
- **Modelo de Presentación:** Representa las vistas del interfaz del usuario mediante modelos estándares de interacción UML.

En cuanto a los requisitos, UWE los clasifica dependiendo del carácter de cada uno. Además, distingue entre las fases de captura definición y validación de requisitos (Sanchez, 2017).

1.6.2. Métricas de Calidad de Software

Se evaluará la calidad del software usando las normas ISO/IEC 25000:

La ISO/IEC 25000, conocida como SQuaRE (System and Software Quality Requirements and Evaluation), es una familia de normas que tiene por objetivo la creación de un marco de trabajo común para evaluar la calidad del producto software. (ISO 25000, 2021)

Esta familia de normas ISO/IEC 25000 se encuentra compuesta por cinco divisiones.

- **ISO/IEC 2500n – División de Gestión de Calidad:** Las normas que forman este apartado definen todos los modelos, términos y definiciones comunes referenciados por todas las otras normas de la familia 25000.
- **ISO/IEC 2501n – División de Modelo de Calidad:** Las normas de este apartado presentan modelos de calidad detallados incluyendo características para calidad interna, externa y en uso del producto software.
- **ISO/IEC 2502n – División de Medición de Calidad:** Estas normas incluyen un modelo de referencia de la medición de la calidad del producto, definiciones de medidas de calidad (interna, externa y en uso) y guías prácticas para su aplicación.
- **ISO/IEC 2503n – División de Requisitos de Calidad:** Las normas que forman este apartado ayudan a especificar requisitos de calidad que pueden ser utilizados en el proceso de licitación de requisitos de calidad del producto software a desarrollar o como entrada del proceso de evaluación.
- **ISO/IEC 2504n – División de Evaluación de Calidad:** Este apartado incluye normas que proporcionan requisitos, recomendaciones y guías

para llevar a cabo el proceso de evaluación del producto software. (ISO 25000, 2021)

1.6.3. Métodos de Estimación de Costos

Para la estimación de costos del software se aplicará el Modelo COCOMO, este método original se convirtió en uno de los modelos de estimación más ampliamente utilizados y estudiados en el mundo.

Está compuesto por tres modelos que corresponden a distintos niveles de detalle y precisión. Mencionados en orden creciente son: Modelo Básico, Intermedio y Detallado. La estimación es más precisa a medida que se toman en cuenta mayor cantidad de factores que influyen en el desarrollo de un producto de software. (Adriana Gomez, 2011)

1.6.4. Seguridad

ISO 27001 es una norma internacional emitida por la Organización Internacional de Normalización (ISO) y describe cómo gestionar la seguridad de la información en una empresa.

El eje central de ISO 27001 es proteger la confidencialidad, integridad y disponibilidad de la información en una empresa. Esto lo hace investigando cuáles son los potenciales problemas que podrían afectar la información (es decir, la evaluación de riesgos) y luego definiendo lo que es necesario hacer para evitar que estos problemas se produzcan (es decir, mitigación o tratamiento del riesgo)

ISO 27001: Especifica los requerimientos necesarios para implantar y gestionar un SGSI. Esta norma es certificable. (Wikipedia, WIKIPEDIA La enciclopedia libre, 2020)

1.6.5. Pruebas al software

✓ **Caja blanca (White box):** El método de pruebas White box (también conocido como clear box testing, glass box testing, transparent box testing, and structural testing) es el cual mira el código y la estructura del producto que se va a probar y usa ese conocimiento para la realización de las pruebas.

Este método se usa en la fase de Unit testing, aunque también puede ocurrir en otras fases como en las pruebas de sistema o de integración. Para la ejecución de este método es necesario que el testero la persona que valla a usar el método

tengan amplios conocimientos de la tecnología y arquitectura usada para desarrollar el programa. (Peño, 2015)

- ✓ **Caja negra (black box):** Es el método en el cual el elemento es estudiado desde el punto de vista de las entradas que recibe y las salidas o respuestas que produce, sin tener en cuenta su funcionamiento interno. Estas pruebas son realizadas desde la interfaz gráfica. (Moderno, 2020)
- ✓ **Prueba de estrés:** Es el proceso en el cual se eligen las actividades a probar en un sitio para codificarlas y ejecutarlas en un tiempo determinado desde una ubicación remota. Nos permiten identificar y planear ante la posibilidad de fallas en el funcionamiento de la plataforma, preferiblemente de manera preventiva. Lo que se hace es medir la capacidad de la infraestructura, los tiempos de respuesta a actividades específicas correspondientes a un número incremental de usuarios generados de manera remota y buscar el número límite de usuarios antes de la negación de servicio de la plataforma. En otras palabras, cuantos usuarios interactuando de manera simultánea con la plataforma puede tolerar la infraestructura de tu sitio (Aliaga, 2020).

1.6.6. Métodos de obtención de datos

En el desarrollo del presente proyecto, se utiliza las técnicas de: observación, entrevista, cuestionarios y encuestas a los usuarios finales lo que nos permite de forma directa, obtener la información necesaria concerniente al movimiento y flujo de información en la unidad de Vicerrectorado.

1.7. HERRAMIENTAS

- **MagicDraw**

MagicDraw, herramienta CASE desarrollada por No Magic. Es compatible con el estándar UML 2.3, desarrollo de código para diversos lenguajes de programación (Java, C++ y C#, entre otros) así como para modelar datos. Cuenta con capacidad para trabajar en equipo y es compatible con varios entornos de desarrollo (IDEs) Diseñada para los analistas del negocio, los analistas del software, los programadores, los ingenieros y los escritores de la documentación.

Facilita el análisis y el diseño de los sistemas Orientado a Objetos y de las bases de datos orientados objeto. (magicdraw, 2021)

- **Bases de datos**

MySQL: Es el sistema de administración de bases de datos (Database Management System, DBMS) más popular, desarrollado y proporcionado por MySQL AB. Es un sistema de gestión de base de datos relacional, multihilo y multiusuario. MySQL fue escrito en C y C++ y destaca por su gran adaptación a diferentes entornos de desarrollo, permitiendo su interacción con los lenguajes de programación más utilizados como PHP, Perl y Java y su integración en distintos sistemas operativos. (MySQL, 2021)

- **PHP**

PHP, es un lenguaje de programación interpretado de alto nivel al lado del servidor para internet, muy similar en su sintaxis al lenguaje C, con algunas diferencias, no compila como al igual que C, ya que es un intérprete, por lo tanto, cada vez que se debe ejecutar un programa, lo interpreta verificando toda su sintaxis.

PHP nos brinda la posibilidad de realizar tareas de forma automatizadas, mejorando la productividad de nuestro sitio web y dando la posibilidad de añadir gran cantidad de funcionalidades que con HTML no podemos hacerlo, ya que HTML no es un lenguaje de programación. (php, 2021)

- **Framework de desarrollo**

CodeIgniter es un **framework** para el desarrollo de aplicaciones en php, que utiliza el MVC. Esto permite a los programadores o desarrolladores Web mejorar su forma de trabajar, además de dar una mayor velocidad a la hora de crear páginas Webs. (Paz, 2021)

- **Framework de diseño**

Bootstrap es un framework CSS de código abierto que favorece el desarrollo web de un modo más sencillo y rápido. Incluye plantillas de diseño basadas en HTML y CSS con la que es posible modificar tipografías, formularios, botones, tablas, navegaciones, menús desplegables, etc. También existe la posibilidad de utilizar extensiones de Javascript adicionales. (Rovira, 2020)

- **JQuery**

JQuery es un Framework JavaScript, nos ofrece una infraestructura con la que tendremos mucha mayor facilidad para la creación de aplicaciones complejas del

lado del cliente. Por ejemplo, con JQuery obtendremos ayuda en la creación de interfaces de usuario, efectos dinámicos, aplicaciones que hacen uso de Ajax, etc. Cuando programemos JavaScript con JQuery tendremos a nuestra disposición una interfaz para programación que nos permitirá hacer cosas con el navegador que estemos seguros que funcionarán para todos nuestros visitantes. (Significados, 2020)

- **Servidor Apache**

Apache es el Servidor Web hecho por excelencia por su configurabilidad, robustez y estabilidad hacen que cada vez millones de servidores reiteren su confianza en este programa. La licencia es una descendiente de la licencia BSD (*Berkeley Software Distribution*), no es GPL (General Public License). Esta licencia permite hacer lo que quieras con el código fuente siempre que les reconozcas su trabajo.

Apache es una tecnología gratuita de código fuente abierta. El hecho de ser gratuita es importante pero no tanto como que se trate de código fuente abierto.

Apache permite personalizar la respuesta ante los posibles errores que se puedan dar en el servidor las características de Apache se pueden extender hasta donde nuestra imaginación y conocimientos lleguen. (Wikipedia, 2021)

1.8. LÍMITES Y ALCANCES

1.8.1. Limites

La presente propuesta se desarrolla para la Unidad de Vicerrectorado de acuerdo a sus necesidades, especificaciones y requerimientos.

- El sistema Web se desarrolla específicamente dentro del marco del proceso de admisión por excelencia académica de la Universidad Pública de El Alto.
- El sistema Web no presta el servicio para cobros de inscripción.
- El sistema Web no se enfoca en el ámbito financiero.

1.8.2. Alcances

Para cumplir con las metas y requerimientos de la institución, el sistema tendrá los siguientes módulos:

- a) Módulo de administración del sistema
 - ✓ Gestión de usuarios con distintos niveles de acceso al sistema.

- ✓ Gestión de cargos/ áreas
- b) Módulo de registro de estudiantes con el beneficio de la modalidad.
 - ✓ Gestión de control de estudiantes
 - ✓ Validación de registro
 - ✓ Digitalización de documentos
- c) Módulo de reportes
 - ✓ Administración de búsqueda de los estudiantes por carrera.
 - ✓ Reporte de estudiantes admitidos (género, carrera).
 - ✓ Gestión de impresión de resoluciones de Vicerrectorado
- d) Módulo de validación y revisión de documentos.
 - ✓ Documentos digitalizados
- e) Módulo de habilitaciones
 - ✓ Convocatorias
 - ✓ Control cupos
- f) Seguridad
- g) Backups

1.9. APORTES

Con la implementación del sistema Web de registro de estudiantes de excelencia académica, se podrá ver reducido el proceso de admisión estudiantil, el manejo de la información, tiempo de respuesta a distintas peticiones por parte de los estudiantes, presentará una mejora considerable por parte de la Unidad de Vicerrectorado, ya que permitirá tener información centralizada en la base de datos, mediante un apropiado flujo de información.

La Universidad Pública de Alto y la Unidad de Vicerrectorado se beneficiará con un sistema confiable y seguro.

**SISTEMA DE REGISTRO DE ADMISIÓN DE ESTUDIANTES
DE LA MODALIDAD DE EXCELENCIA ACADÉMICA
CASO: VICERRECTORADO UPEA**

CAPÍTULO II

MARCO TEÓRICO



2.1. INTRODUCCIÓN

En el presente capítulo se exponen los conceptos básicos que sirvieron como base para poder desarrollar el presente proyecto. Es el sustento que fundamenta el proyecto y que se elabora a partir de los conceptos y teorías ya existentes de fuentes diversas.

Para eso es necesario dar una pequeña descripción de los conceptos fundamentales para empezar el sistema de registro académico, profundizar el conocimiento de la metodología y las herramientas que se aplicaran dentro del proyecto.

2.2. SISTEMA

Un sistema es un conjunto de partes o elementos organizados y relacionados que interactúan entre sí para lograr un objetivo. Los sistemas reciben datos, energía o materia del ambiente (entrada) y proveen información, energía o materia (salida). Los sistemas reales comprenden intercambios de energía, información o materia con su entorno. (Alegsa, 2018)

Un sistema es un conjunto de elementos relacionados entre sí que funciona como un todo. Si bien cada uno de los elementos de un sistema puede funcionar de manera independiente, siempre formará parte de una estructura mayor. Del mismo modo, un sistema puede ser, a su vez, un componente de otro sistema. (Significados, 2020)

Un sistema es módulo ordenado de elementos que se encuentran interrelacionados y que interactúan entre sí.

2.2.1. Tipos de sistema

Existen tres tipos de sistemas reales:

- ✓ **Abiertos** (recibe flujos de su ambiente, adaptando su comportamiento de acuerdo a esto).
- ✓ **Cerrados** (sólo intercambia energía con su entorno).
- ✓ **Aislados** (no realiza ningún tipo de intercambio con su entorno). (Alegsa, 2018)

Los sistemas pueden clasificarse tomando en cuenta diversos criterios, algunos de ellos son: (Luz, 2021)

1) Según la relación que establecen con el medio ambiente:

- ✓ **Sistemas cerrados.** Se caracterizan por su hermetismo, ya que no generan ningún intercambio con el ambiente, por lo que no se ven afectados ni ejercen influencia en él. Tienen un comportamiento totalmente programado y determinado.
- ✓ **Sistemas abiertos.** Se caracterizan por establecer intercambios con el medio ambiente que los rodea, para lograrlo, se valen de salidas y entradas por las que intercambian, de manera constante, energía y materia.

2) Según su constitución:

- ✓ **Sistemas conceptuales o abstractos.** Están constituidos por conceptos que son ajenos a la realidad y que resultan meramente abstractos.
- ✓ **Sistemas físicos.** Están constituidos por objetos o seres concretos y tangibles.

3) Según su origen:

- ✓ **Sistemas artificiales.** Son producto de la creación humana, por lo que dependen de la presencia de otros para poder existir.
- ✓ **Sistemas naturales.** No dependen de la mano de obra del hombre para originarse.

4) Según su movimiento:

- ✓ **Sistemas dinámicos.** Se caracterizan por evolucionar con el tiempo.
- ✓ **Sistemas estáticos.** Se caracterizan por carecer de movimiento.

5) Según la complejidad de los elementos que los conforman:

- ✓ **Sistemas complejos.** Se caracterizan por estar compuestos por subsistemas, lo que vuelve difícil la tarea de identificar los distintos elementos que los componen.
- ✓ **Sistemas simples.** Se caracterizan por no contar con subsistemas, lo que permite identificar fácilmente sus elementos constitutivos.

6) Según su naturaleza:

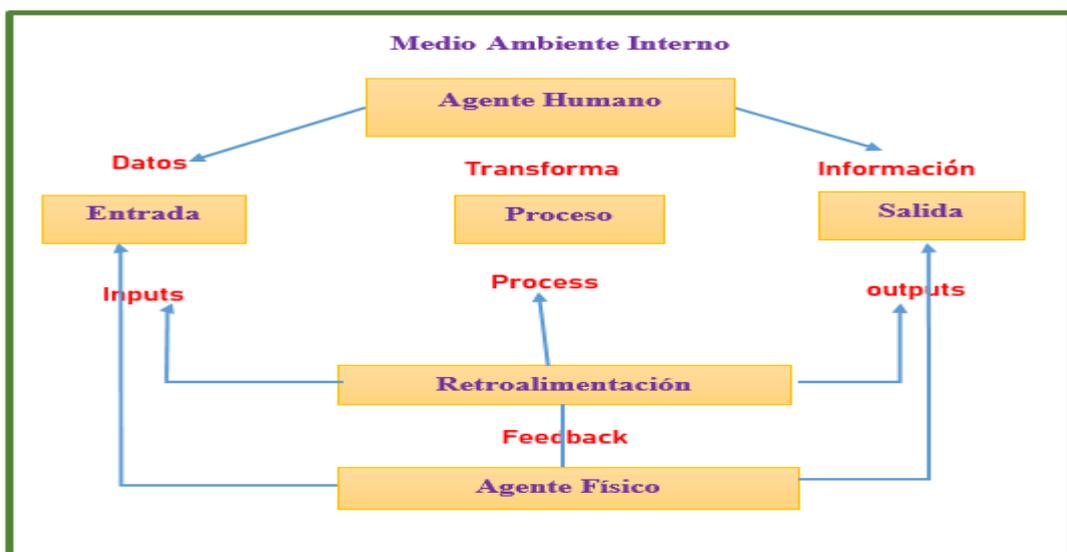
- ✓ **Sistemas inertes.** Se caracterizan por carecer de vida.
- ✓ **Sistemas vivos.** Se caracterizan por tener vida.

2.2.2. Estructura de los sistemas

La estructura de los sistemas se puede clasificar en los siguientes: (Hiamix, 2013)

1. **Medio ambiente externo** un sistema puede ser físico o concreto (una computadora, un televisor, un humano) o puede ser abstracto o conceptual (software).
2. **Medio ambiente interno** este medio ambiente en conjunto es cambiante y nunca esta estático por lo cual es necesario que la organización dedique el tiempo necesario para prever esos cambios internos y haga los ajustes necesarios para establecerlas nuevas directrices que tendrá que adoptar en el mercado. Se refiere a todo lo que está dentro de la organización, es decir los trabajadores, el clima organizacional, la cultura organizacional, etcétera.
3. **Agente humano** es el usuario de la computadora.
4. **Entrada** es una información que es recibida por un sistema para su procesamiento.
5. **Proceso** puede informalmente entenderse como un programa en ejecución. formalmente un proceso es una unidad de actividad que se caracteriza por la ejecución de una secuencia de instrucciones, un estado actual, y un conjunto de recursos del sistema asociado.
6. **Salida** en informática es el proceso de transmitir la información por un objeto.
7. **Retroalimentaciones** un mecanismo de control de los sistemas dinámicos por el cual una cierta proporción de la señal de salida se dirige a la entrada, y así regula su comportamiento.
8. **Agente físico** es el hardware.

Figura 2.1 Estructura General de un Sistema



Fuente: (Hiamix, 2013)

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". (Fuentes, 2018)

La gestión académica se realicen un conjunto de actividades encaminadas a facilitar la transformación de las condiciones institucionales con espíritu de renovación e investigación, en búsqueda de soluciones a los problemas o necesidades identificadas durante el desarrollo del proceso enseñanza aprendizaje. (Salinas, 2018)

La Gestión Académica es aquel ámbito de acción orientada a facilitar y mejorar los procesos formativos de admisión que imparten las instituciones de educación superior.

2.4. EXCELENCIA

Si hablamos de la definición de excelencia en general, tendríamos que decir que la excelencia es una virtud, un talento que día con día se trabaja, es dar lo máximo en cuanto a calidad y se sale de las normas ordinarias.

Hoy en día, cuando nos referimos a la excelencia académica es referirse a los estándares de alta calidad que las instituciones educativas buscan en los ámbitos internacionales; es también elevar el potencial intelectual de cada uno de los estudiantes, quienes pueden fijarse metas. (Anahuac, 2018)

La excelencia es poseer una virtud, un talento o cualidad especial. También lo podemos entender cómo definir un objetivo y alcanzar el máximo rendimiento y productividad. (Llopart, 2016)

La excelencia es también un objetivo para el estándar de rendimiento, lo que está cerca de convertirse en una gran persona con cualidades diferentes.

2.5. REGISTRO

Un registro es el espacio físico o virtual donde se deja constancia de un hecho, o el acto de hacer lo mismo. Esto, con el fin de que terceras personas y las autoridades competentes estén informadas al respecto. (Westreicher, 2020)

Un registro informático es un tipo o conjunto de datos almacenados en un sistema. un registro del sistema viene a ser una base de datos que tiene el fin de almacenar configuración, opciones y comandos propios del sistema operativo. (Bembibre, 2015)

Es decir que un registro es documento donde se relacionan acontecimientos o cosas, especialmente aquellos que deben constatar de manera permanentemente.

2.6. REGISTRO DE ADMISIÓN DE ESTUDIANTES

En la Universidad Pública de El Alto las distintas maneras de registro para la admisión de estudiantes son las siguientes:

1. PRUEBA DE SUFICIENCIA ACADÉMICA

Permite a los bachilleres postulantes seguir estudios universitarios sin el requisito del curso pre-universitario, esta se compone de una prueba de conocimiento y de un diagnóstico psicotécnico.

2. CURSO PRE-UNIVERSITARIO

- ✓ Fotocopia de Diploma de bachiller o documento que acredite la aprobación del sexto de secundaria (Libreta escolar) previa presentación de los documentos originales.
- ✓ Fotocopia de la cédula de identidad (en vigencia), pasaporte o libreta de servicio militar.
- ✓ Fotocopia de certificado de nacimiento (actualizado).
- ✓ Papeleta de depósito bancario en la cuenta Nro. 10000004713083 del BANCO UNIÓN S.A. Costo de Inscripción Bs. 100 para estudiantes bolivianos y equivalentes a \$us 100 para estudiantes extranjeros.
- ✓ Presentar las cinco últimas libretas del nivel secundario y certificado del Director de Colegio que acredite estar cursando el último curso del nivel secundario, válido para las carreras que desarrollen su curso preuniversitario en el segundo semestre de la gestión.

- ✓ Fotocopias del depósito bancario (cantidad de acuerdo al requerimiento de la carrera).
- ✓ Recabar y llenar el formulario 02 (inscripciones), en la jefatura de Registros y Admisiones RR.AA.
- ✓ Fotografías 3x3 fondo rojo (traje formal, cantidad de acuerdo al requerimiento de la carrera).
- ✓ En caso de postulantes extranjeros(as), deberán presentar documentos originales debidamente legalizados por las autoridades nacionales y residencia legal.
- ✓ Los documentos deben ser presentados en fólder colgante de color (de acuerdo al requerimiento de la carrera) con su respectivo fastener y pestaña en el orden establecido.

NOTA: La presentación de la documentación es en folder de color, de acuerdo a la carrera a la cual postula. La prueba de suficiencia académica se establece según la convocatoria de cada carrera. (Reyqui, 2017)

3. EXCELENCIA ACADÉMICA

Incentiva y admite a todos aquellos estudiantes bachilleres sobresalientes de las unidades educativas fiscales de la ciudad de El Alto, área rural, periférica y provisional del departamento de La Paz, que haya obtenido las tres mejores calificaciones del sexto de secundaria. (Reyqui, 2017)

4. ADMISIÓN ESPECIAL

Permite a las personas que deseen iniciar y/o continuar sus estudios universitarios y que poseen Títulos de Licenciatura, Técnico Superior, graduados del Colegio Militar, Academia Nacional de Policía, Institutos Pedagógicos y otros que son resultado de disposiciones y convenios especiales de Universidades del Sistema Universitario, quedar exentas de cumplir con la modalidad de admisión vigente.

Requisitos

- ✓ Fotocopia legalizada de Título profesional en Provisión Nacional
- ✓ Fotocopia legalizada del Diploma de Bachiller.
- ✓ Documentos originales que acrediten y respalden la admisión especial.
- ✓ Certificado de nacimiento original (actualizado) y fotocopia simple de la cédula de identidad.

NOTA: La presentación de los documentos es con nota dirigida a(l)(la) Vicerrector(a) de la UPEA y el respectivo visto bueno de la Carrera a la cual postula. (Reyqui, 2017)

2.7. REGISTRO DE ADMISIÓN EXCELENCIA ACADÉMICA

Este proceso se encuentra regido por normas e instructivos que los postulantes deberán cumplir para ingresar a esta casa de estudios, el procedimiento es el siguiente:

1. Nota dirigida al Vicerrector(a) solicitando la Admisión Estudiantil por la Modalidad de Excelencia Académica (presentar cuando el trámite se realice en Vicerrectorado).
2. Nota dirigida al Director(a) de la Carrera a la cual postula solicitando la Admisión Estudiantil por la Modalidad de Excelencia Académica (autorización y visto bueno del Director(a) de la Carrera.
3. Tres formularios VRC-AEEA-01, autorizado por el director de la carrera.
4. Ser boliviano (a) de nacimiento (certificado de nacimiento original actualizado).
5. Fotocopia(s) simple (s) del diploma de bachiller o de las libretas que le acredite (n) la aprobación del nivel secundario, previa presentación de los documentos originales.
6. Fotocopia de cedula de identidad (en vigencia), pasaporte o libreta de servicio militar.
7. Certificado de la Unidad Educativa Fiscal a la que pertenece que es uno de los tres mejores estudiantes de todo 6 to de secundaria indicando el promedio obtenido y la posición en la que se encuentra (1ero, 2do y 3er lugar) avalado por la Unidad Distrital de Educación correspondiente.
8. Copia del Boletín Centralizador o registro de Notas de Fin de gestión del 6to de secundaria de la Unidad Educativa.
9. Presentar toda la documentación en folder colgante color rojo.

2.8. INTERNET

Internet es una red de redes que permite la interconexión descentralizada de computadoras a través de un conjunto de protocolos denominado TCP/IP. (Porto & Gardey, 2021)

Internet es una red integrada por miles de redes y computadoras interconectadas en todo el mundo mediante cables y señales de telecomunicaciones, que utilizan una tecnología común para la transferencia de datos. (Zamora, 2014)

El internet es un esquema de red donde se utiliza en una plataforma.

Diferencia de Diferencias entre Internet, Intranet y Extranet

- ✓ **Internet:** La comunicación se realiza de forma instantánea. Se define como un red de redes que permite el intercambio de información de forma pública. Podemos acceder de forma ilimitada a páginas web, redes sociales, foros, ect. También podemos acceder a hostings o servicios que proveen a los usuarios espacio para almacenar información, imágenes, vídeos u otros ...
- ✓ **Intranet:** A diferencia de Internet, se trata de una red privada. Se trata de una herramienta de gestión muy útil para sus usuarios, permite la difusión de información y la colaboración entre los mismos. Por motivos de seguridad, su acceso suele estar restringido a los miembros que forman parte de la misma. Normalmente se implementa a través de una infraestructura de red local (LAN).
- ✓ **Extranet:** Se trata de una red privada, que hace uso de protocolos de Internet (TCP/IP, HTTP, etc.) para realizar las comunicaciones, su acceso también es restringido a los componentes de la misma. Se puede entender como dos o más Intranets conectadas a través de Internet.

En la tabla N 2.1 se muestra de manera resumida las diferencias entre las aplicaciones en una entidad:

Tabla 2.1 Comparación de internet, intranet y extranet

	Internet	Intranet	Extranet
Acceso	Publico	Privado	Semi – Publico
Usuarios	Cualquiera	Usuarios de una misma colectividad	Colectividades relacionadas
Información	Fragmentada	Propietaria	Compartida entre colectividades

Fuente: (Zamora, 2014)

2.8.1. Ventajas y desventajas del internet

Ventajas

- ✓ Velocidad en la comunicación.
- ✓ Acceso a múltiples contenidos.
- ✓ Difusión de contenidos propios.
- ✓ Simplificación de procesos.
- ✓ Masificación del conocimiento.
- ✓ Espíritu colaborativo.
- ✓ Interacción con grupos de interés.
- ✓ Generación de nuevos empleos.
- ✓ Acceso a nuevos formatos de entretenimiento.
- ✓ Nuevas dinámicas sociales.

Desventajas

- ✓ Uso de datos personales con fines desconocidos.
- ✓ Estimula el sedentarismo.
- ✓ Delincuencia digital.
- ✓ Acoso en línea.
- ✓ Publicidad invasiva.
- ✓ Spam.
- ✓ Noticias falsas.
- ✓ Exposición a contenido inapropiado.
- ✓ Puede generar adicción.
- ✓ Limita la comunicación cara a cara. (Zamora, 2014)

2.9. WEB 2.0.

La web 2.0 es un concepto que empezó a utilizarse en el año 2003. Su aparición se debió al auge de diversas aplicaciones como los blogs o las redes sociales que permitieron que los usuarios dejaran de ser mejor sujetos pasivos para adoptar un rol mucho más activo y dinámico. Sin embargo, a pesar de que hace años que existe este término.

En pocas palabras, el concepto de web 2.0 es sinónimo de aplicaciones web que permiten a los usuarios compartir información en Internet. En su momento, cuando este término se puso de moda, una de las plataformas preferidas era Blogger. Posteriormente, Facebook se convirtió en la primera red social que no solo permitía compartir información, sino la posibilidad de interactuar de una forma más inmediata. (Gardey, 2021)

2.9.1. Características de la Web 2.0

Es importante tener en cuenta que no existe una definición precisa de Web 2.0, aunque es posible aproximarse a ella estableciendo ciertos parámetros.

- Una página web que se limita a mostrar información y que ni siquiera se actualiza, forma parte de la generación 1.0. En cambio, cuando las páginas ofrecen un nivel considerable de interacción y se actualizan con los aportes de los usuarios, se habla de Web 2.0.

Cabe mencionar que las diferencias entre la primera y la segunda era de la Web no se basan en un cambio a nivel tecnológico en los servidores, aunque naturalmente se ha dado un considerable avance en el hardware; es el enfoque de la Red, los objetivos y la forma en la que los usuarios comenzaron a percibir la información en línea lo que caracteriza este renacer, que tuvo lugar silenciosa pero velozmente, a comienzos del nuevo milenio.

- Algunas de las principales características de la web 2.0 son los servicios de redes sociales, las plataformas de autopublicación (por ejemplo, las herramientas de WordPress de creación de blogs y sitios web), las etiquetas (funcionalidad que permite añadir metadatos a páginas web o contenidos multimedia), el botón «me gusta» y la calificación de contenidos (que permite al usuario expresar su opinión sobre el contenido) y los marcadores sociales.

Hasta entonces, Internet era un universo de datos mayoritariamente estáticos, una fuente de consulta revolucionaria que atraía a millones de personas a contemplarla pasivamente. Si bien los foros y el chat datan de la Web 1.0, éstos se encontraban bien diferenciados de los sitios tradicionales (tal y como ocurre en la actualidad); navegar era similar a visitar un gran centro comercial, con infinidad de tiendas, en las cuales era posible comprar productos, pero no alterar sus vidrieras. (Wikipedia, 2021)

2.9.2. Usos de la Web 2.0

La Web 2.0 se caracteriza principalmente por la participación del usuario como contribuidor activo y no solo como espectador de los contenidos de la Web (usuario pasivo). (Wikipedia, 2021)

Esto queda reflejado en aspectos como:

- ✓ El auge de los blogs.
- ✓ El auge de las redes sociales.
- ✓ Las webs creadas por los usuarios, usando plataformas de auto-edición.
- ✓ El contenido agregado por los usuarios como valor clave de la Web.
- ✓ El etiquetado colectivo (folcsonomía, marcadores sociales...).
- ✓ La importancia de la larga cola.
- ✓ La beta perpetua: la Web 2.0 se inventa permanentemente.
- ✓ Aplicaciones web dinámicas.
- ✓ La World Wide Web como plataforma.

2.10. INGENIERÍA DE SOFTWARE

Es el establecimiento y uso de principios fundamentales de la ingeniería con objeto de desarrollar en forma económica software que sea confiable y que trabaje con eficiencia en máquinas reales. Es la aplicación de un enfoque sistemático, disciplinado y cuantificable al desarrollo, operación y mantenimiento de software; es decir, la aplicación de la ingeniería al software (IEEE). (Vargas A. M., 2019)

La ingeniería de software es una especialidad que consiste en sistemas, instrumentos y técnicas que se emplean en el desarrollo de los programas informáticos. También, incorpora el análisis precedente de la situación, el bosquejo del proyecto, el desarrollo del software, el ensayo necesario para comprobar su funcionamiento correcto y poner en funcionamiento el sistema. (Flores, 2018)

2.10.1. Etapas de la ingeniería de software

La ingeniería de Software consta de siete etapas. A continuación, cada una de ellas brevemente definidas:

- ✓ **Etapa de análisis:** Es el procedimiento de investigación de un problema al que se desea encontrar la solución. Se define con claridad el Problema que hay que resolver o el programa que se desea inventar, identificando los elementos principales que conformarán el producto.
- ✓ **Etapa de Diseño:** Es el procedimiento que emplea la información acumulada en la etapa de análisis al diseño del producto. La labor principal de la etapa de diseño es crear un modelo o las características precisas para el producto o Componentes del Sistema.
- ✓ **Etapa de Desarrollo:** Consiste en el empleo de los diseños creados durante la etapa de diseño para elaborar los elementos a utilizarse en el sistema.
- ✓ **Etapa de Pruebas o Verificación Prueba:** Consiste en garantizar que los elementos individuales que componen el sistema o producto, presentan las características requeridas en la especificación creada durante la etapa de diseño.
- ✓ **Etapa de Implementación o Entrega Implantación:** Consiste en la distribución del producto y hacerlo llegar a manos del cliente. Documentar y comentar adecuadamente el código de los programas. Facilitar la interpretación visual del código utilizando reglas de formato de código previamente consensuadas en el equipo de desarrollo.
- ✓ **Etapa de Mantenimiento:** Consiste en aplicar las soluciones apropiadas a cualquier problema del producto y re- liberar el producto mejorado, dándole una nueva versión.

Figura 2.2 Etapa de la ingeniería de software



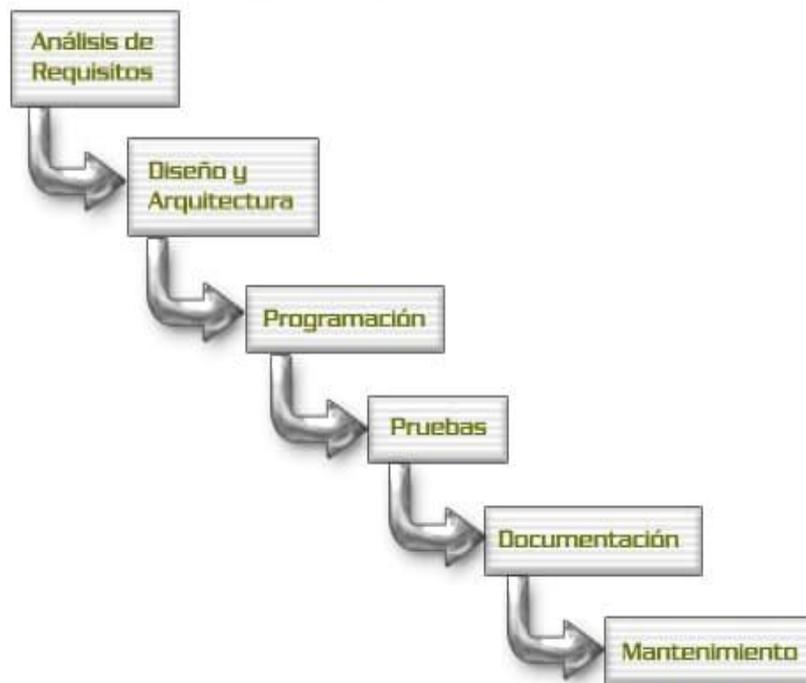
Fuente: (Flores, 2018)

2.10.2. Fases de la ingeniería de software

- ✓ **Análisis de requisitos:** Separar los requerimientos de un producto de software es la primera etapa para poder crearlo. Mientras que los usuarios creen que ellos saben lo que el software tiene que hacer, se necesita de capacidad y pericia en la ingeniería de software para considerar requerimientos no finalizados, confusos o incoherentes
- ✓ **Diseño y arquitectura:** Consiste en acordar cómo funcionará de manera general sin especificar los detalles. Se basa en la inclusión de fundamentos de la implementación tecnológica, como el hardware, la red, etc.
Se determinan los Casos de Uso para abarcar las funciones que efectuará el sistema, y se convierten las entidades definidas en el análisis de requisitos en clases de diseño, dando como resultado un modelo cercano a la programación orientada a objetos.
- ✓ **Programación:** La reducción de un diseño a código puede ser la parte más evidente del trabajo de ingeniería de software, pero no siempre resulta ser la parte más larga. La dificultad y la permanencia de esta etapa está estrechamente ligada al o a los lenguajes de programación utilizados.
- ✓ **Pruebas:** Consiste en verificar que el software ejecute apropiadamente las labores indicadas en la especificación. La tecnología de prueba es aplicada de manera independiente, en cada módulo del software y luego hacer la prueba de integralmente, para así llegar al objetivo.
Es usual, además de recomendado, que las pruebas sean realizadas por alguien distinto al que desarrolló el programa; sin perjuicio de lo anterior el programador debe hacer sus propias pruebas.
- ✓ **Documentación:** Todo lo que se relaciona a la documentación del desarrollo del software y de la administración del proyecto, pasando por modelaciones (UML), diagramas, pruebas, manuales de usuario, manuales técnicos, etc; todo con el propósito de realizar correcciones con regularidad, utilidad, mantenimiento futuro y ampliaciones al sistema.
- ✓ **Mantenimiento:** Conservar y realizar mejoras en el software para afrontar fallas encontradas y nuevos requisitos. Esto puede llevar más tiempo incluso que el desarrollo inicial del software. Alrededor de 2/3 de toda la ingeniería de software tiene que ver con dar mantenimiento.

Figura 2.3 Ingeniería de software y sus fases

Fases del Proceso de Desarrollo del Software



Fuente: (Flores, 2018)

2.10.3. Capas de la ingeniería de software

La ingeniería de software es una tecnología multicapa, cualquier enfoque de ingeniería debe apoyarse sobre un compromiso de organización de calidad.

- **Herramientas:** Soporte de ayuda para construir el software, proporciona un enfoque automático o semiautomático para el proceso y para los métodos.
- **Método:** cómo se debe construir técnicamente el software. abarcan una gran gama de tareas, que incluyen: Análisis de requisitos. Diseño. Construcción de programas. Pruebas. Mantenimiento.
- **Modelo de Proceso:** definen las actividades para poder realizar el software.
- **Enfoque de calidad:** comprende la gestión para obtener un software de calidad, y hace revisiones para hacer mejoras al producto.

Figura 2.4 Capas de la ingeniería de software



Fuente: (Flores, 2018)

2.11. INGENIERÍA WEB

La ingeniería web es el uso de métodos sistemáticos, disciplinados y cuantificables al desempeño eficaz, operatividad y crecimiento de programas de muy buena calidad en la World Wide Web. La ingeniería Web se debe al crecimiento desenfrenado que está teniendo la Web está ocasionando un impacto en la sociedad y el nuevo manejo que se le está dando a la información en las diferentes áreas en que se presenta ha hecho que las personas tiendan a realizar todas sus actividades por esta vía. (Cardenas, 2016)

La ingeniería web es para garantizar el buen funcionamiento y mantenimiento de los sitios web, este debe contar con ciertos atributos y características que en conjunto forman un concepto muy importante, para alcanzar el éxito en cualquier organización, herramienta, y todo aquello que se pueda considerar como servicio.

Dicho concepto es la calidad, que con atributos como, usabilidad, navegabilidad, seguridad, mantenibilidad, entre otros, hace posible por un lado la eficiencia del artefacto Web y por ende la satisfacción del usuario final. (Chavez, 2012)

Cabe destacar que la Ingeniería de la Web hace una diferencia entre un sitio web y un aplicativo, ya que la ingeniería de la Web no se dedica a la construcción de sitios web si no a la construcción de aplicativos web, la principal característica que los distingue (aplicativos de sitios web) es que los sitios web son sitios en la web en donde se publica contenido generalmente estático o un muy bajo nivel de interactividad con el usuario, mientras que los aplicativos son lugares con alto contenido de interactividad y funcionalidades que bien podrían ser de un software convencional, el aplicativo web más sencillo sería uno que contenga formularios

y subiendo de nivel encontramos los que realizan conexión con bases de datos remotas, y administradores de contenidos entre otras. (Lopez, 2010)

La ingeniería web es una de las actividades que se enfoca en el uso de la ingeniería de software para obtener y desarrollar un software eficiente y de alta calidad en aplicaciones y sistemas.

2.11.1. Características de la Ingeniería Web

El desarrollo de aplicaciones Web posee determinadas características que lo hacen diferente del desarrollo de aplicaciones o software tradicional y sistemas de información. La Ingeniería Web es multidisciplinar y aglutina contribuciones de diferentes áreas: arquitectura de la información, ingeniería de hipertexto/hipertexto, ingeniería de requisitos, diseño de interfaz de usuario, usabilidad, diseño gráfico y de presentación, diseño y análisis de sistemas, Ingeniería de Software, Ingeniería de Datos, indexado y recuperación de información, testeo, modelado y simulación, despliegue de aplicaciones, operación de sistemas y gestión de proyectos.

La ingeniería de la Web no es un clon o subconjunto de la ingeniería de software, aunque ambas incluyen desarrollo de software y programación, pues a pesar de que la ingeniería de la Web utiliza principios de Ingeniería de Software, incluye nuevos enfoques, metodologías, herramientas, técnicas, guías y patrones para cubrir los requisitos únicos de las aplicaciones web. (Salguera, 2012)

Los principales aspectos de la ingeniería de la Web incluyen, entre otros, los siguientes temas:

- Diseño de procesos de negocio para aplicaciones web.
- Herramientas CASE para aplicaciones web.
- Generación de código para aplicaciones web.
- Desarrollo web colaborativo.
- Modelado conceptual de aplicaciones web.
- Diseño de Modelos de datos para sistemas de información web.
- Ingeniería web empírica.

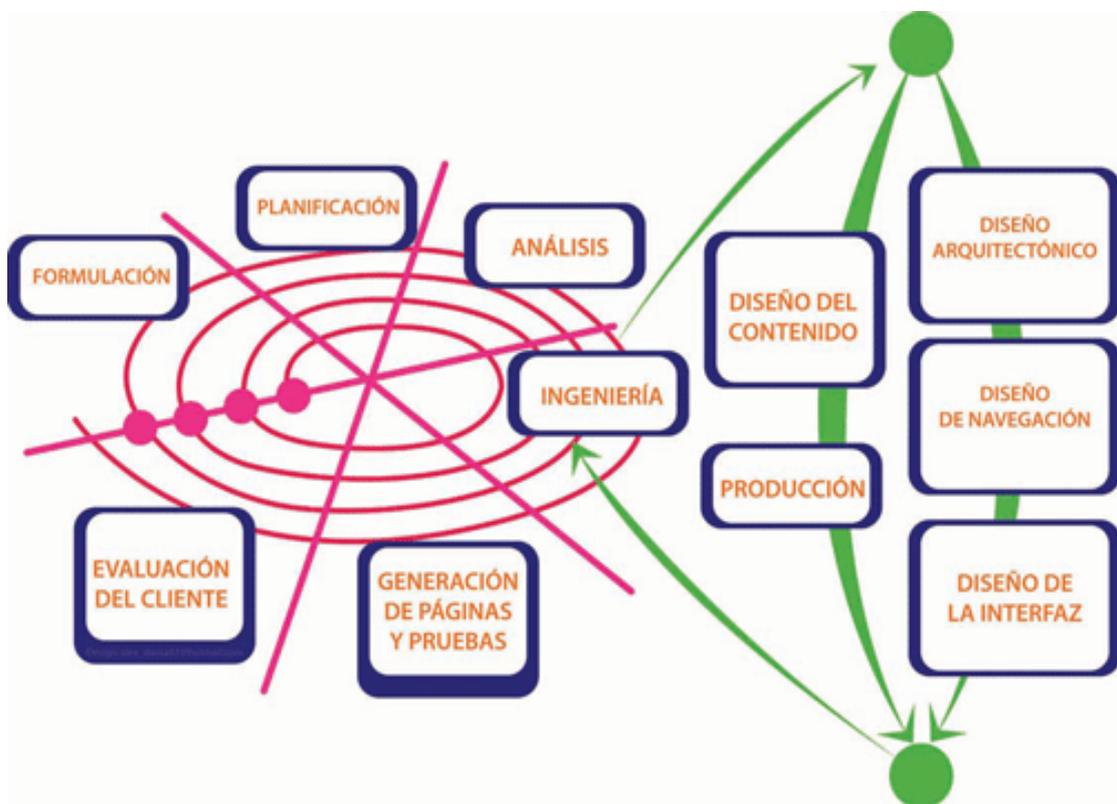
- Entornos de desarrollo de aplicaciones web integrados.
- Herramientas de autor para contenido multimedia.
- Pruebas de rendimiento de aplicaciones basadas en web.
- Personalización y adaptación de aplicaciones web.
- Modelado de procesos para aplicaciones web.
- Herramientas y métodos de prototipado.
- Control de calidad y pruebas de sistemas.
- Ingeniería de requisitos para aplicaciones web.
- Aplicaciones para la Web Semántica.
- Factorías de software para la web.
- Métodos, herramientas y automatización de pruebas de aplicaciones web.
- Aplicaciones web móviles.
- Usabilidad de aplicaciones web.
- Accesibilidad para la web.
- Metodologías de diseño web.
- Formación en ingeniería de la web.
- Diseño de interfaces de usuario.
- Métricas para la web, estimación de costes y medición.
- Gestión de proyectos web y gestión de riesgos.
- Desarrollo y despliegue de servicios web.

Según Roger S. Pressman numera siete actividades que forman parte del proceso de la Ingeniería Web y que son aplicables a cualquier aplicación Web independientemente de su tamaño y complejidad. (Pinzon, 2017)

1. La **Formulación** identifica objetivos y establece el alcance de la primera entrega.

2. La **Planificación** genera estimación de costo, la evaluación de riesgo y el calendario del desarrollo y fechas de entrega.
3. El **Análisis** especifica los requerimientos e identifica el contenido.
4. La **Modelización** Consta de dos partes:
 - Diseño y producción del contenido.
 - Diseño de la arquitectura, navegación e interfaz del usuario.
5. En la **Generación de Páginas** se integran arquitectura, navegación e interfaz para la creación más visible del proyecto, que son las páginas.
6. El **Test** son pruebas en busca de errores en todos los niveles: contenido, funcional, navegación, etc.
7. El resultado final es sometido a **Evaluación del Cliente**.

Figura 2.5 Proceso de la Ingeniería Web



Fuente: (Pinzon, 2017)

2.12. METODOLOGÍA INGENIERÍA WEB BASADA EN UML (UWE)

La Ingeniería web basada en UML es una metodología basada en el Proceso Unificado y UML (Lenguaje Unificado de Modelado) para el desarrollo de aplicaciones Web, cubre todo el ciclo de vida de las aplicaciones Web (SlideShare, 2015)

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, 2016)

2.12.1. Características

La metodología UWE define vistas especiales representadas gráficamente por diagramas UML, tales como el análisis de requerimientos, diseño conceptual modelo de navegación y el modelo de presentación. UWE no limita el número de diagramas posibles de una aplicación

- Es una metodología orientada a objetos, iterativa e incremental basada en UML.
- Se basa también en el proceso de desarrollo del software unificado.
- Proporciona un diseño sistemático y una generación semiautomática en las aplicaciones web a través de un Framework de publicación XML.
- UWE define su propio perfil UML en el cual se definen todos los elementos necesarios para modelar los diferentes aspectos de una aplicación web que son la presentación y la navegación entre otros.
- En esta metodología se proponen 2 tipos de diagramas para el modelado de la navegación que son: El modelo de espacio , el cual se definen todos

los caminos navegacionales, es decir todas aquellas asociaciones de navegación directa entre los distintos objetos de la aplicación más bien conocida como clases de navegación, y el segundo modelo de estructura de navegación el cual se define la estructura de acceso que son utilizadas en la navegación, es decir todo aquello referente a menús, índices y demás.

2.12.2. Modelos de la metodología 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. (Thewolf, 2015)

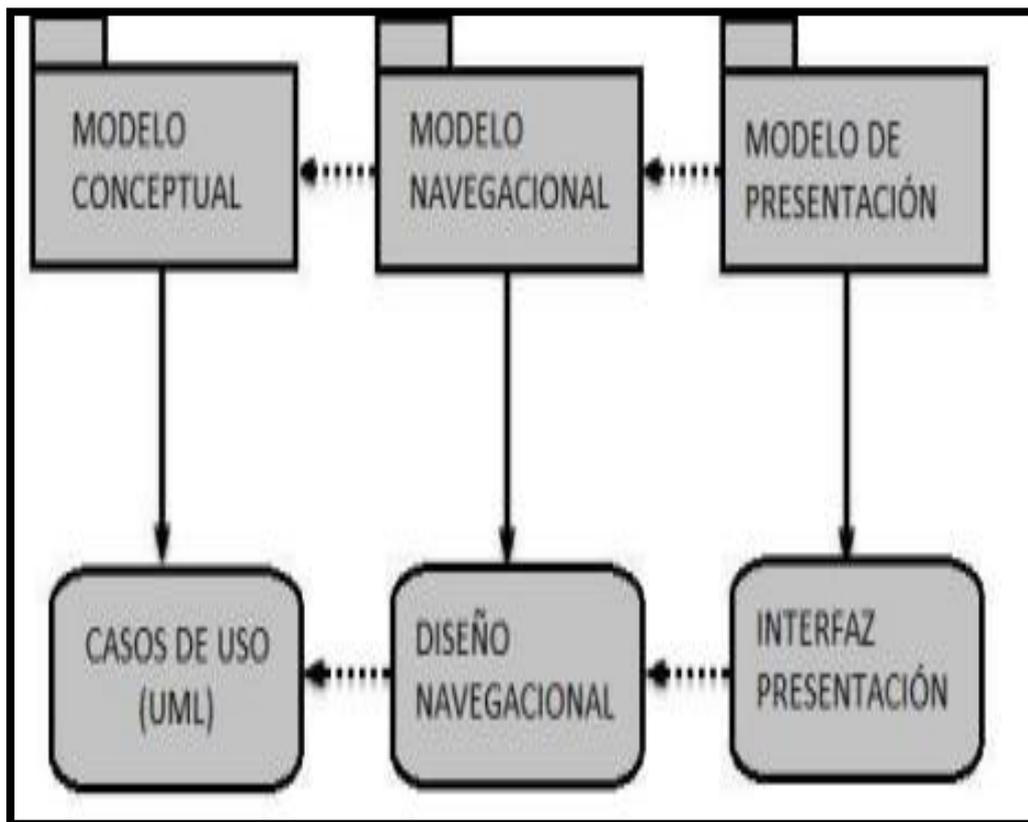
UWE proporciona guías para la construcción de modelos de forma sistemática, enfocándose en la personalización y en estudio de casos de uso.

Los principales artefactos que produce el método de diseño de UWE son los siguientes:

- Un **Modelo de Requerimientos** que captura los requerimientos del sistema.
- Un **Modelo Conceptual** para el contenido (modelo de contenido).
- Un **Modelo de Usuario**.
- Un **Modelo de Navegación** que comprende la estructura de la navegación.
- Un **Modelo de Presentación** que abarca modelos estáticos y dinámicos (modelo de estructura de la presentación, modelo del flujo de la presentación, modelo de interface abstracta de usuario, y modelo de ciclo de vida del objeto).
- Un **modelo de adaptación**.

En cuanto a los requisitos, UWE los clasifica dependiendo del carácter de cada uno. Además, distingue entre las fases de captura definición y validación de requisitos (Sanchez, 2017).

Figura 2.6 Ingeniería Web Basada en UML



Fuente: (Vargas s. m., 2017)

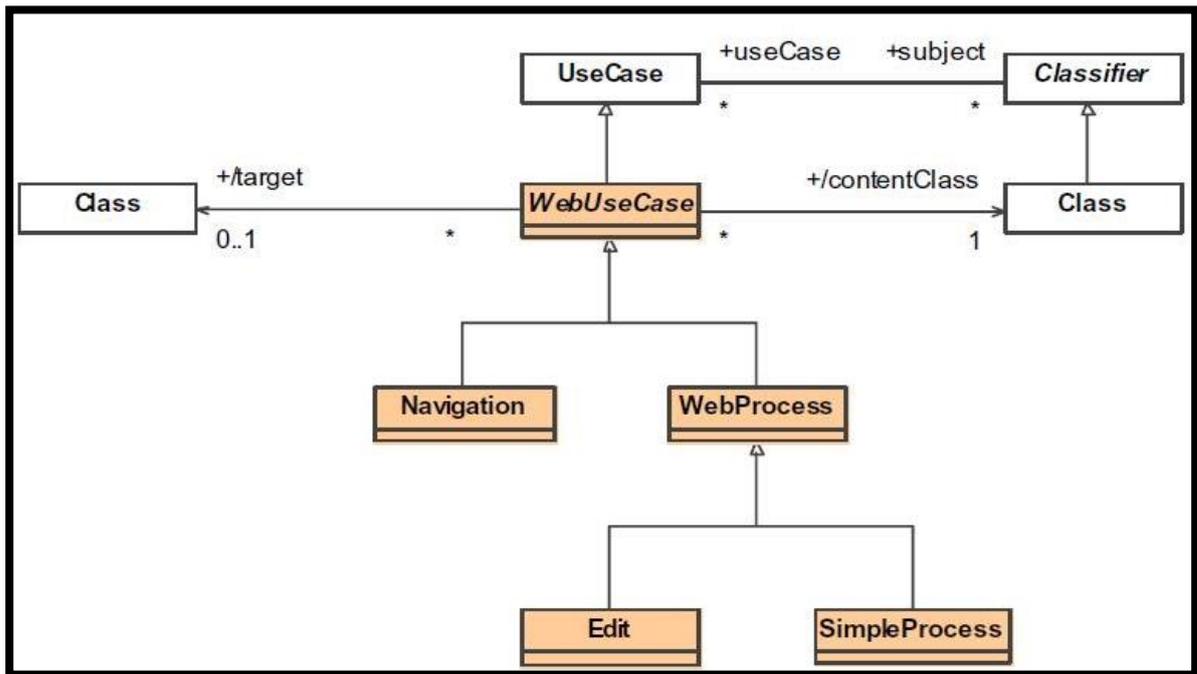
2.12.3. Fases de la metodología UWE

2.12.3.1. Modelo de análisis 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.

Figura 2.7 Modelo de Caso de Uso de la Ingeniería Web

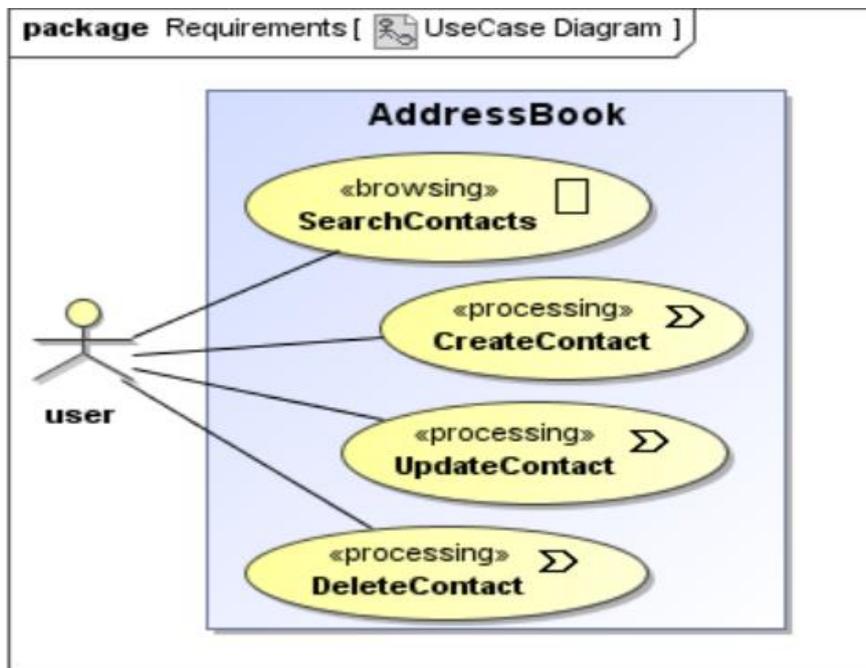


Fuente: (Vargas M. S., 2017)

2.12.3.2. Modelo de Casos de Uso

Un diagrama de casos de uso muestra la relación entre los actores y los casos de uso del sistema. Representa la funcionalidad que ofrece el sistema en lo que se refiere a su interacción externa.

Figura 2.8 Modelo de Caso de Uso de la Ingeniería Web

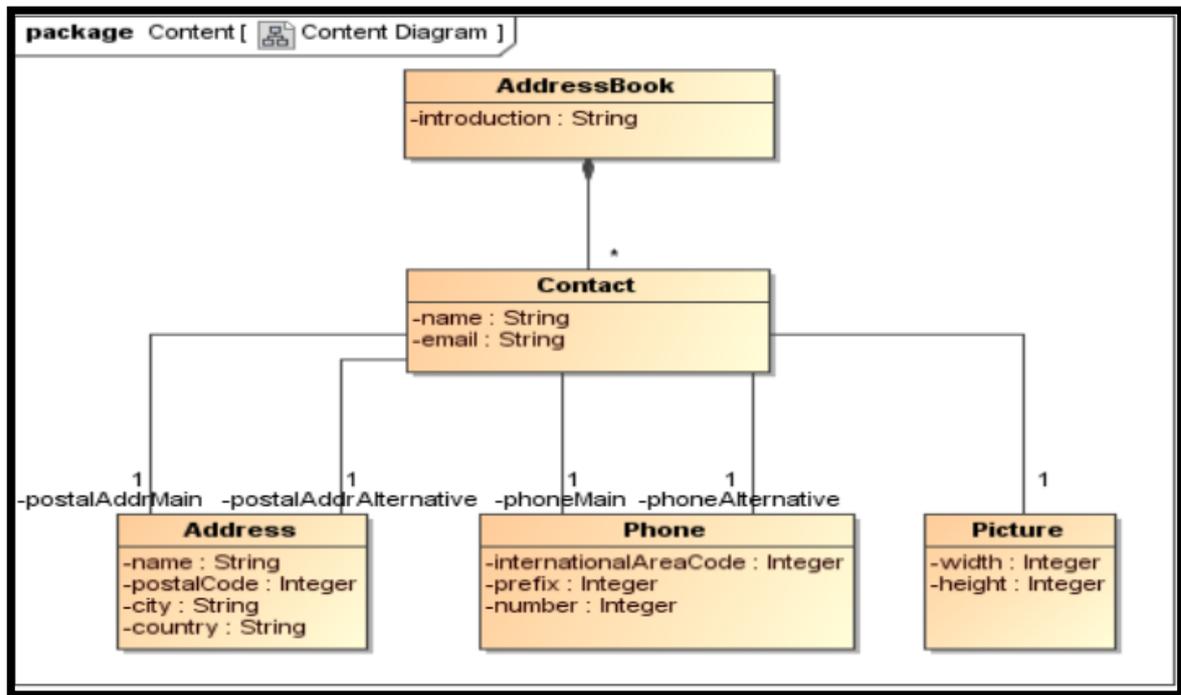


Fuente: (Vargas M. S., 2017)

2.12.3.3. Modelo contenido

El modelo conceptual de un sistema software constituye una abstracción externa que describe mediante diagramas y notaciones con distinto grado de formalidad el conocimiento que debe poseer una persona acerca de un sistema, conocimiento que se encuentra almacenado en la Memoria a Largo Plazo.

Figura 2.9 Modelo Contenido de la Ingeniería Web



Fuente: (Vargas M. S., 2017)

2.12.3.4. 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.

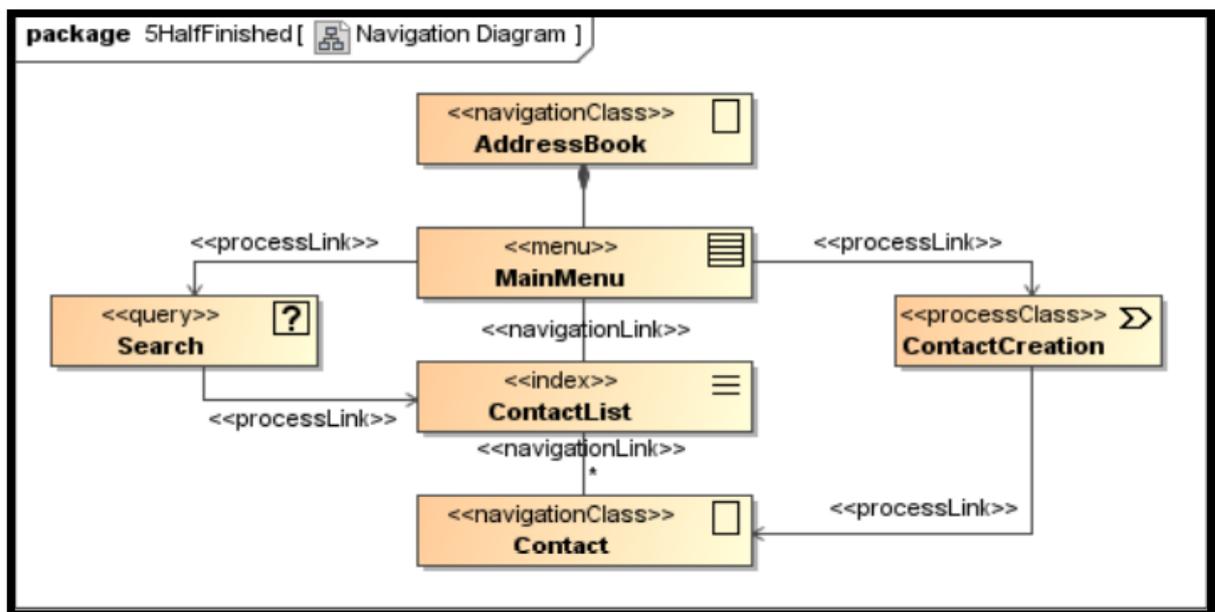
Los elementos de modelado son:

- **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.

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.

Figura 2.10 Modelo Navegación de la Ingeniería Web



Fuente: (Vargas M. S., 2017)

2.12.3.5. Modelo de Presentación

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.

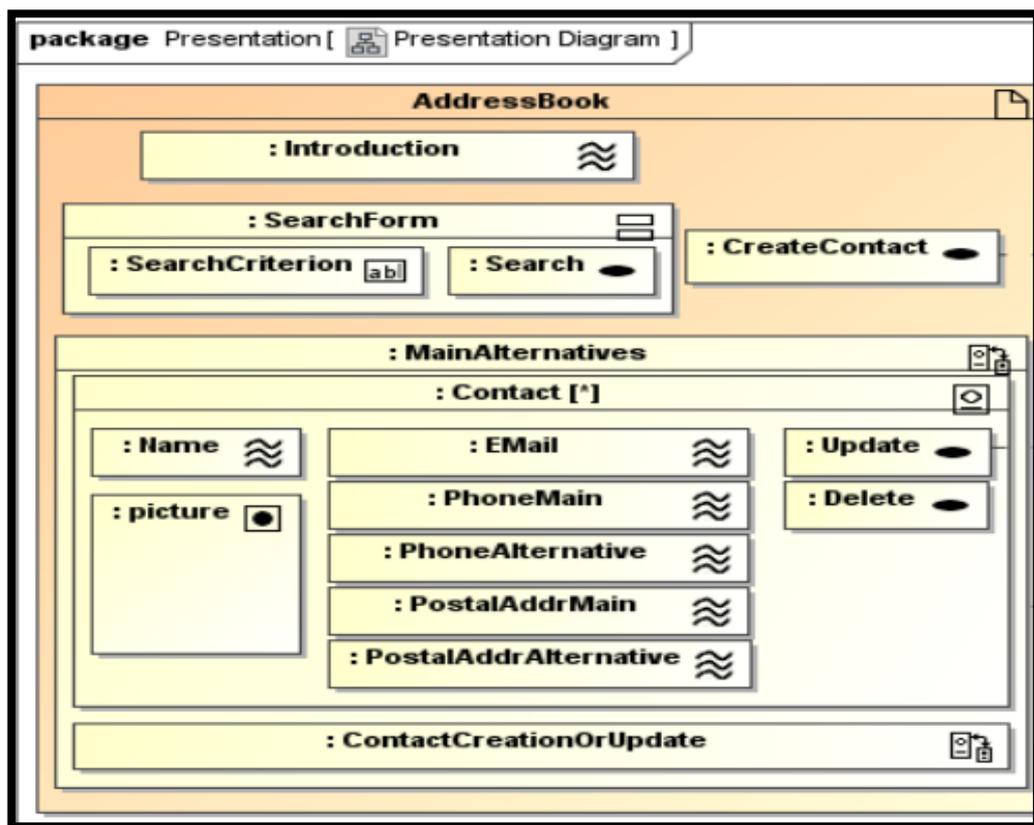
Los elementos básicos del modelo de presentación son:

- **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>>), vínculo (<<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 2.11 Modelo de Presentación de la Ingeniería Web



Fuente: (Vargas M. S., 2017)

2.13. ARQUITECTURA DE SOFTWARE (MVC)

MVC es una propuesta de arquitectura del software utilizada para separar el código por sus distintas responsabilidades, manteniendo distintas capas que se encargan de hacer una tarea muy concreta, lo que ofrece beneficios diversos.

MVC se usa inicialmente en sistemas donde se requiere el uso de interfaces de usuario, aunque en la práctica el mismo patrón de arquitectura se puede utilizar

para distintos tipos de aplicaciones. Surge de la necesidad de crear software más robusto con un ciclo de vida más adecuado, donde se potencie la facilidad de mantenimiento, reutilización del código y la separación de conceptos.

Su fundamento es la separación del código en tres capas diferentes, acotadas por su responsabilidad, en lo que se llaman Modelos, Vistas y Controladores, o lo que es lo mismo, Model, Views & Controllers, si lo prefieres en inglés. En este artículo estudiaremos con detalle estos conceptos, así como las ventajas de ponerlos en marcha cuando desarrollamos.

MVC es un "invento" que ya tiene varias décadas y fue presentado incluso antes de la aparición de la Web. No obstante, en los últimos años ha ganado mucha fuerza y seguidores gracias a la aparición de numerosos frameworks de desarrollo web que utilizan el patrón MVC como modelo para la arquitectura de las aplicaciones web. (Desarrollo web, 2020)

El modelo MVC es un patrón de arquitectura de software que separa los datos de una aplicación, la interfaz de usuario, y la lógica de control en tres componentes distintos.

2.13.1. Modelo

Es la capa donde se trabaja con los datos, por tanto, contendrá mecanismos para acceder a la información y también para actualizar su estado. Los datos los tendremos habitualmente en una base de datos, por lo que en los modelos tendremos todas las funciones que accederán a las tablas y harán los correspondientes selects, updates, inserts, etc..

El modelo es el responsable de:

- ✚ Acceder a la capa de almacenamiento de datos. Lo ideal es que el modelo sea independiente del sistema de almacenamiento.
- ✚ Define las reglas de negocio (la funcionalidad del sistema). Un ejemplo de regla puede ser: "Si la mercancía pedida no está en el almacén, consultar el tiempo de entrega estándar del proveedor".
- ✚ Lleva un registro de las vistas y controladores del sistema.

- ✚ Si estamos ante un modelo activo, notificará a las vistas los cambios que en los datos pueda producir un agente externo (por ejemplo, un fichero por lotes que actualiza los datos, un temporizador que desencadena una inserción, etc.).

2.13.2. Vista

Las vistas, como su nombre nos hacen entender, contienen el código de nuestra aplicación que va a producir la visualización de las interfaces de usuario, o sea, el código que nos permitirá renderizar los estados de nuestra aplicación en HTML. En las vistas nada más tenemos los códigos HTML y PHP que nos permite mostrar la salida.

En la vista generalmente trabajamos con los datos, sin embargo, no se realiza un acceso directo a éstos. Las vistas requerirán los datos a los modelos y ellas se generará la salida, tal como nuestra aplicación requiera

Este presenta el modelo en un formato adecuado para interactuar, usualmente la interfaz de usuario.

Las vistas son responsables de:

- ✚ Recibir datos del modelo y los muestra al usuario.
- ✚ Tienen un registro de su controlador asociado (normalmente porque además lo instancia).
- ✚ Pueden dar el servicio de "Actualización ()", para que sea invocado por el controlador o por el modelo (cuando es un modelo activo que informa de los cambios en los datos producidos por otros agentes).

2.13.3. Controlador

Contiene el código necesario para responder a las acciones que se solicitan en la aplicación, como visualizar un elemento, realizar una compra, una búsqueda de información, etc.

En realidad, es una capa que sirve de enlace entre las vistas y los modelos, respondiendo a los mecanismos que puedan requerirse para implementar las necesidades de nuestra aplicación. Sin embargo, su responsabilidad no es manipular directamente datos, ni mostrar ningún tipo de salida, sino servir de

enlace entre los modelos y las vistas para implementar las diversas necesidades del desarrollo.

Este responde a eventos, usualmente acciones del usuario e invoca cambios en el modelo y probablemente en la vista.

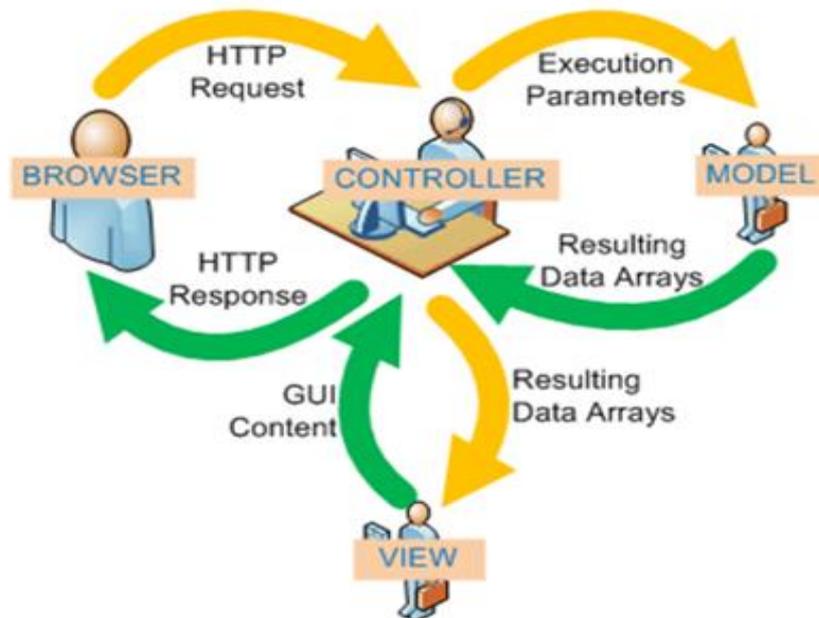
El controlador es responsable de:

- ✚ Recibe los eventos de entrada (un clic, un cambio en un campo de texto, etc.).
- ✚ Contiene reglas de gestión de eventos, del tipo "SI Evento Z, entonces Acción W". Estas acciones pueden suponer peticiones al modelo o a las vistas. Una de estas peticiones a las vistas puede ser una llamada al método "Actualizar ()". Una petición al modelo puede ser "Obtener_tiempo_de_entrega (nueva_orden_de_venta)".

2.13.4. Características del Modelo Vista Controlador

Aunque se pueden encontrar diferentes implementaciones de MVC, la característica del flujo que sigue el control generalmente es el siguiente:

Figura 2.12 Flujo de MVC



Fuente: (Desarrollo web, 2020)

1. El usuario interactúa con la interfaz de usuario de alguna forma (por ejemplo, el usuario pulsa un botón, enlace, etc.)

2. El controlador recibe (por parte de los objetos de la interfaz-vista) la notificación de la acción solicitada por el usuario. El controlador gestiona el evento que llega, frecuentemente a través de un gestor de eventos (handler) o callback.
3. El controlador accede al modelo, actualizándolo, posiblemente modificándolo de forma adecuada a la acción solicitada por el. Los controladores complejos están a menudo estructurados usando un patrón de comando que encapsula las acciones y simplifica su extensión.
4. El controlador delega a los objetos de la vista la tarea de desplegar la interfaz de usuario. La vista obtiene sus datos del modelo para generar la interfaz apropiada para el usuario donde se refleja los cambios en el modelo. El modelo no debe tener conocimiento directo sobre la vista. Sin embargo, el patrón de observador puede ser utilizado para proveer cierta Unidad entre el modelo y la vista, permitiendo al modelo notificar a los interesados de cualquier cambio.
5. La interfaz de usuario espera nuevas interacciones del usuario, comenzando el ciclo nuevamente. (Alicante, 2021)

2.14. HERRAMIENTAS

2.14.1. MagicDraw

Es una herramienta visual de modelado UML y CASE con soporte de trabajo en equipo. Diseñada para analistas de negocio, analistas de software, analistas de sistemas, programadores, ingenieros de control de calidad y redactores de documentación, esta herramienta de desarrollo dinámica y versátil facilita el análisis y el diseño de sistemas orientados a objetos (OO) y bases de datos. Proporciona el mejor mecanismo de ingeniería de código de la industria (con soporte completo de ida y vuelta para los lenguajes de programación J2EE, C#, C++, CORBA IDL, .NET, XML Schema, WSDL), así como el modelado de esquemas de bases de datos, la generación de DDL y las facilidades de ingeniería inversa.

MagicDraw, herramienta CASE desarrollada por No Magic. Es compatible con el estándar UML 2.3, desarrollo de código para diversos lenguajes de programación (Java, C++ y C#, entre otros) así como para modelar datos. Cuenta con capacidad para trabajar en equipo y es compatible con varios entornos de desarrollo (IDEs) Diseñada para los analistas del negocio, los analistas del software, los

programadores, los ingenieros y los escritores de la documentación. (Magicdraw, 2021)

- **Un práctico editor UML.** Una interfaz gráfica de usuario amigable y personalizable le permite expresar sus ideas en UML de la forma más rápida y directa posible. Toda la notación UML 2 y la semántica son compatibles.
- **Una poderosa herramienta de ingeniería de código.** Puede construir un modelo UML, generar código fuente a partir de él, escribir más código a mano o en su IDE favorito, luego revertir el código, hacer algunos cambios en el MagicDraw, y fusionar su modelo con el código de nuevo. No se pierde ningún detalle, no importa el camino (hacia adelante o hacia atrás) que usted vaya. Soporte para Java, C#, C++, CORBA IDL, DDL.
- **Una facilidad de informes del modelo UML.** Basado en los archivos XSL, puede generar informes HTML personalizados para cada elemento del modelo. MagicDraw le permite seleccionar qué partes del modelo incluir y cómo debe ser el informe.
- **Una herramienta de análisis de modelos OO.** MagicDraw genera diagramas de dependencia de paquetes, recupera y muestra árboles de herencia, o rastrea cualquier otro tipo de relación como un parámetro de método o un valor de retorno, tales como dependencias, asociaciones, realizaciones y uso de clases.
- **Un generador de modelos OO personalizado.** Conjunto predefinido de patrones de diseño incluyendo GoF, Java, patrón de diseño JUnit. Todos los patrones pueden ser extendidos y nuevos patrones pueden ser creados usando Java o Jython.
- **Herramienta de modelado en equipo.** Más de un desarrollador puede trabajar con el mismo modelo OO. El modelo UML se almacena en el repositorio de Teamwork Server. Cada desarrollador puede bloquear una parte del modelo y trabajar en esa parte individualmente. Más tarde, los cambios pueden confirmarse en el servidor y compartirse con el equipo. El servidor funciona como un sistema de control de versiones de código fuente.
- **Herramienta de modelado de bases de datos.** Puede diseñar el esquema de la base de datos en un diagrama de clases UML, y luego generar su

código DDL. Si desea analizar la estructura de la base de datos existente, puede realizar ingeniería inversa a través del puente JDBC.

a) Características de MagicDraw

- Compatible con los siguientes IDEs:
 - ✓ Sun Java Studio 8.
 - ✓ Borland CaliberRM 6.0, 6.5 herramienta de requisitos.
 - ✓ Oracle Workshop 8.1.2.
 - ✓ E2E Bridge 4.0
 - ✓ IntelliJ IDEA 4.X or later.
 - ✓ NetBeans 6.X or later.
 - ✓ Eclipse 3.1 o superior (versión Java)
 - ✓ IBM Rational Application Developer
 - ✓ Borland JBuilder 8.0, 9.0, X, 2005, 2006, 2007
 - ✓ Built-in CVS interfaz para almacenar archivos de proyectos.
- Integración con herramientas MDA: Compuware' OptimalJ, AndroMDA, Interactive Objects' ArcStyler, openArchitectureWare, E2E Bridge, y Mia-Software Tools.
- Diseñada para los analistas del negocio, los analistas del software, los programadores, los ingenieros del QA, y los escritores de la documentación.
- Facilita el análisis y el diseño de los sistemas (OO) y de las bases de datos orientados objeto.

b) Ventajas de uso de MagicDraw

- Interfaz elegante e intuitiva, la mayor parte de las opciones accesibles con un solo clic.
- Ayudas en el diseño con autocompletación y corrección automática en tiempo real.
- Permite visualizar el proyecto de diferentes formas.
- Posible derivación de modelos UML a través de códigos fuentes escritos anteriormente.
- Facilidad y rapidez para el cambio del dominio del modelado.
- Generador automático de informes.

- Desarrollo colaborativo directamente con la herramienta a través del Team Work Server(Software que permite trabajar a más de un desarrollador sobre el mismo proyecto en un mismo instante, el modelo está almacenado en un equipo servidor y los desarrolladores).
- Disponible para un gran número de plataformas y sistemas operativos. (EcuRed, s.f.)

Figura 2.13 Herramienta MagicDraw



Fuente: (Magicdraw, 2021)

2.14.2. Gestor de base de datos MySQL

Es el sistema de administración de bases de datos (Database Management System, DBMS) más popular, desarrollado y proporcionado por MySQL AB. Es un sistema de gestión de base de datos relacional, multihilo y multiusuario. MySQL fue escrito en C y C++ y destaca por su gran adaptación a diferentes entornos de desarrollo, permitiendo su interacción con los lenguajes de programación más utilizados como PHP, Perl y Java y su integración en distintos sistemas operativos.

MySQL es un sistema de base de datos disponible gratuitamente. Sin embargo, hay varias ediciones de pago también disponibles con las que puede utilizar la funcionalidad avanzada.

MySQL es fácil de usar en comparación con otros programas de bases de datos como Microsoft SQL Server y Oracle database etc. Puede ser usado con cualquier lenguaje de programación, pero es ampliamente usado con PHP.

MySQL puede ejecutarse en múltiples plataformas como Linux, Windows, Unix y un esquema de información para definir y administrar sus metadatos. Puede instalarlo en su sistema local o incluso en el servidor.

a) Características clave de MySQL

MySQL presenta algunas ventajas que lo hacen muy interesante para los desarrolladores. La más evidente es que trabaja con bases de datos relacionales, es decir, utiliza tablas múltiples que se interconectan entre sí para almacenar la información y organizarla correctamente.

Al ser basada en código abierto es fácilmente accesible y la inmensa mayoría de programadores que trabajan en desarrollo web han pasado usar MySQL en alguno de sus proyectos porque al estar ampliamente extendido cuenta además con una ingente comunidad que ofrece soporte a otros usuarios. Pero estas no son las únicas características como veremos a continuación:

- **Facilidad de gestión:** Es bastante fácil de descargar y utilizar el software.
- **Alto rendimiento:** Le proporciona utilidades de carga rápida con diferente caché de memoria.
- **Escalable:** Con MySQL, puede escalar en cualquier momento. Es realmente fácil crear almacenes de datos que incluyen una enorme cantidad de datos.
- **Compatibilidad:** MySQL es compatible con todas las plataformas modernas como Windows, Linux, Unix.
- **Rendimiento:** MySQL le ofrece resultados de alto rendimiento sin perder la funcionalidad esencial.
- **Seguridad completa de los datos:** Sólo los usuarios autorizados pueden acceder a la base de datos. Completa seguridad de los datos.
- **Bajo Costo:** Es gratis de usar.
- **Eficiencia de memoria:** MySQL tiene baja pérdida de memoria.
- **Arquitectura Cliente y Servidor:** MySQL basa su funcionamiento en un modelo cliente y servidor. Es decir, clientes y servidores se comunican entre sí de manera diferenciada para un mejor rendimiento. Cada cliente puede hacer consultas a través del sistema de registro para obtener datos, modificarlos, guardar estos cambios o establecer nuevas tablas de registros, por ejemplo.
- **Compatibilidad con SQL:** SQL es un lenguaje generalizado dentro de la industria. Al ser un estándar MySQL ofrece plena compatibilidad por lo que

si has trabajado en otro motor de bases de datos no tendrás problemas en migrar a MySQL.

- **Vistas:** Desde la versión 5.0 de MySQL se ofrece compatibilidad para poder configurar vistas personalizadas del mismo modo que podemos hacerlo en otras bases de datos SQL. En bases de datos de gran tamaño las vistas se hacen un recurso imprescindible.
- **Procedimientos almacenados.** MySQL posee la característica de no procesar las tablas directamente, sino que a través de procedimientos almacenados es posible incrementar la eficacia de nuestra implementación.
- **Desencadenantes.** MySQL permite además poder automatizar ciertas tareas dentro de nuestra base de datos. En el momento que se produce un evento otro es lanzado para actualizar registros o optimizar su funcionalidad.
- **Transacciones.** Una transacción representa la actuación de diversas operaciones en la base de datos como un dispositivo. El sistema de base de registros avala que todos los procedimientos se establezcan correctamente o ninguna de ellas. En caso por ejemplo de una falla de energía, cuando el monitor falla u ocurre algún otro inconveniente, el sistema opta por preservar la integridad de la base de datos resguardando la información.

b) Ventajas de usar MySQL

Descritas las principales características de MySQL es fácil ver sus ventajas. MySQL es una opción razonable para ser usado en ámbito empresarial. Al estar basado en código abierto permite a pequeñas empresas y desarrolladores disponer de una solución fiable y estandarizada para sus aplicaciones. Por ejemplo, si se cuenta con un listado de clientes, una tienda online con un catálogo de productos o incluso una gran selección de contenidos multimedia disponible, MySQL ayuda a gestionarlo todo debida y ordenadamente.

Figura 2.14 Herramienta MySql



Fuente: (MySql.com, 2021)

2.14.3. 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, 2021)

Figura 2.15 Herramienta Php



Fuente: (Estrada, 2021)

2.14.4. Framework Codelgniter

Codelgniter 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. Codelgniter le permite centrarse creativamente en su proyecto minimizando la cantidad de código necesario para una tarea determinada. (Instituto de Tecnología de Columbia Británica, 2019)

Codelgniter 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 Codelgniter es distribuido con una licencia MIT y puede descargarse desde la plataforma GitHub. (IONOS, 2020)

a) Estructura de Codelgniter

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** medio 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.

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)

b) Ventajas de usar el framework Codeigniter:

- Las páginas se procesan más rápido, el núcleo de Codeigniter es bastante ligero.
- Es sencillo de instalar, basta con subir los archivos al ftp y tocar un archivo de configuración para definir el acceso a la base de datos.
- Existe abundante documentación en la red.
- Facilidad de edición del código ya creado.
- Facilidad para crear nuevos módulos, páginas o funcionalidades.
- Estandarización del código

- Separación de la lógica y arquitectura de la web, el MVC.
- Cualquier servidor que soporte PHP+MySQL sirve para Codeigniter.
- Codeigniter se encuentra bajo una licencia open source, es código libre.
(IONOS, 2020)

Figura 2.16 Herramienta CodeIgniter



Fuente: (IONOS, 2020)

2.14.5. Bootstrap

Bootstrap es un framework CSS desarrollado por Twitter en 2010, para estandarizar las herramientas de la compañía.

Inicialmente, se llamó Twitter Blueprint y, un poco más tarde, en 2011, se transformó en código abierto y su nombre cambió para Bootstrap. Desde entonces fue actualizado varias veces y ya se encuentra en la versión 4.4.

El framework combina CSS y JavaScript para estilizar los elementos de una página HTML. Permite mucho más que, simplemente, cambiar el color de los botones y los enlaces.

Esta es una herramienta que proporciona interactividad en la página, por lo que ofrece una serie de componentes que facilitan la comunicación con el usuario, como menús de navegación, controles de página, barras de progreso y más.

Además de todas las características que ofrece el framework, su principal objetivo es permitir la construcción de sitios web responsive para dispositivos móviles. 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. (acensTechnologies, 2016)

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

a) Funcionalidad de Bootstrap

Bootstrap ofrece una serie de características que se pueden implementar en un sitio web. A continuación, hablaremos de las posibilidades de esta herramienta.

🚦 **Diseño responsive:** Una de las características principales de Bootstrap es permitir que la adaptación de la página se realice según el tipo de dispositivo utilizado. Para garantizar la responsividad, el framework funciona con:

- Estilización del elemento `<div>`;
- Uso del class container.

En la práctica, el elemento `<div>`, funciona para crear una serie de notas, similar a una tabla, capaz de estructurar la página de forma adaptable.

Ya hubo un intento de utilizar tablas para crear diseños responsivos, sin embargo, existían limitaciones para definir la longitud de las columnas lo que hizo imposible su uso en dispositivos más pequeños, como smartphones.

El elemento `<div>` es más flexible, ya que permite definir y cambiar el tamaño de la longitud fácilmente. Bootstrap le ha asignado al elemento `<div>` una característica de class container, que funciona para determinar las dimensiones apropiadas para los elementos insertados en ese espacio.

Básicamente, el framework funciona con tres tipos de containers:

- **Container:** como un conjunto con una propiedad de ancho máximo, que determina qué tamaño de tela es ideal para crear el diseño de página.
- **Container-fluid:** considera la longitud total de la tela del dispositivo para definir el diseño. Para esto, se considera la propiedad `width —100%` en todos los límites de tamaño de tela—.
- **Container-{ breakpoint}:** considera `width —100%` hasta alcanzar un cierto tamaño—.

a) Biblioteca de componentes: Otra interesante función de este framework es la cantidad de componentes que pueden ser usados para proporcionar una mejor interacción y perfeccionar la comunicación con el usuario.

Enseguida mencionaremos los principales.

- **Alertas:** Bootstrap permite una configuración simple y rápida de diferentes tipos de alertas, con colores específicos, según la situación.
- **Carrusel:** Un componente ampliamente utilizado en Bootstrap es el Carrusel, una presentación de diapositivas, es decir, una herramienta que permite la visualización de imágenes de manera receptiva.
- **Barra de navegación:** Otro poderoso componente de la estructura es la NavBar (barra de navegación), que permite la construcción de un sistema de navegación sensible. Es posible configurar diferentes formas de presentar el menú, elegir entre posicionamiento lateral o superior y, también, definir una forma de visualización que se pueda extender o contraer. También es posible determinar cómo mostrar los enlaces del menú, que pueden ser en forma de botón, enlace, menú suspenso, entre otras configuraciones, para facilitar la implementación de la navegación del sitio. (acensTechnologies, 2016)

b) Ventajas de Bootstrap

Entre las ventajas de utilizar bootstrap podemos enumerar las siguientes:

- Es de código abierto, y todo su código actualizado se encuentra en un repositorio de Github.
- Está mantenido y actualizado por Twitter.
- Es compatible con la mayoría de navegadores (Chrome, Safari, Mozilla...).
- Dispone de gran cantidad de documentación, tanto en su portal como en páginas web especializadas.
- Utiliza componentes vitales para los desarrolladores (HTML5, CSS3, jQuery o GitHub, entre otros).
- Incluye Grid system para maquetar por columnas.
- Sus plantillas son de sencilla adaptación responsive.
- Dispone de un conjunto de elementos web personalizables.
- Se integra con librerías JavaScript.

- Usa Less: un lenguaje de las hojas de estilo CSS preparado para enriquecer los estilos de la web. (Lozada, 2020)

Figura 2.17 Herramienta Bootstrap



Fuente: (Lozada, 2020)

2.14.6. JavaScript

JavaScript es un lenguaje de programación diseñado en un principio para añadir interactividad a las páginas webs y crear aplicaciones web. A pesar de la similitud en el nombre, no está relacionado con Java. Se emplea en el desarrollo de páginas web para tareas como cambiar automáticamente la fecha de una página, hacer que una página aparezca en una ventana emergente al hacer clic en un enlace o que un texto o imagen cambien al pasar el ratón por encima. También suele emplearse para hacer encuestas y formularios. Se ejecuta en el ordenador del visitante a la web, por lo que no requiere descargas constantes desde el sitio web. (Armetrics, 2021)

a) Uso de JavaScript

JavaScript también se utiliza en entornos que no se basan en web, tales como documentos PDF, navegadores específicos del sitio y widgets de escritorio. Máquinas virtuales de JavaScript (VMs) mucho más nuevas y rápidas, así como plataformas construidas sobre este lenguaje, han aumentado la popularidad de JavaScript para aplicaciones web. Por parte del cliente, JavaScript se ha aplicado tradicionalmente como un lenguaje interpretado, pero los navegadores más recientes pueden realizar la compilación al momento.

También se utiliza en el desarrollo de videojuegos, en la creación de aplicaciones de escritorio y móviles y en la programación de servidores con entornos de ejecución como Node.js. Para interactuar con una página web se provee al lenguaje JavaScript de una implementación del Document Object Model (DOM).

Normalmente se utilizaba JavaScript en las páginas web HTML para realizar operaciones y solo en el marco de la aplicación cliente, sin acceso a funciones del servidor. En la actualidad se emplea para enviar y recibir información del servidor junto con ayuda de otras tecnologías como AJAX. JavaScript es interpretado en el agente de usuario al mismo tiempo que las sentencias van descargándose junto con el código HTML. (McFarland, 2014)

Figura 2.18 Herramienta JavaScript



Fuente: (Armetrics, 2021)

2.14.7. jQuery

jQuery a una librería o biblioteca de JavaScript que facilita la programación en este lenguaje, al ofrecer una serie de funciones y métodos con los cuales los usuarios pueden producir páginas web o sitios web más rápido y fácil.

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. (Porto & Gardey, Definición, 2019)

a) Características de JQuery

A continuación, se presentan las principales características de JQuery:

- Es un software libre, por lo que puede ser empleado por cualquier usuario de manera gratuita.
- Su librería permite actualizaciones constantes y rápidas.
- Posee un código abierto y compatible con diferentes navegadores.
- Es fácil de usar, por lo que permite ahorrar tiempo y esfuerzo.
- Su desempeño se integra muy bien con AJAX, una técnica de desarrollo web.
- Permite realizar animaciones, efectos y personalizaciones.
- Es compatible con diferentes buscadores como Google Chrome, Microsoft Edge, Firefox, IE, Safari, Android e IOS, cuyas páginas web se deben programar en formas diferentes. (Parada, 2019)

b) Ventajas y desventajas de JQuery

- JQuery es ampliamente utilizado por los desarrolladores de sitios web porque ofrece una serie de ventajas que facilitan su trabajo y ahorran tiempo y esfuerzo que pueden destinar a otros proyectos.
- Asimismo, JQuery es de uso práctico, es compatible con diversos buscadores, los bugs o errores de software son resueltos con rapidez, entre otros. Cabe mencionar que también se puede hacer uso de JQuery en los dispositivos móviles.
- Ahora bien, entre sus desventajas se pueden mencionar las constantes versiones publicadas y, que, aunque parezca fácil de usar, un error en su aplicación puede generar una larga serie de contratiempos.

A las ventajas de estas dos soluciones, jQuery Mobile agrega otras que lo convierten en una opción ideal para cualquier desarrollador:

- Incluye, listos para usar, elementos como botones, barras de herramientas, formularios, listas y tipografías especialmente optimizadas y estilizadas para aplicaciones móviles. Además, cuenta con un sistema de bloques ideal para construir plantillas.
- Nos permite construir aplicaciones aptas para todo tipo de plataformas casi sin esfuerzo, evitándonos preocupaciones por problemas de compatibilidad.
- Es muy fácil de aprender y utilizar. Si se tienen adecuados conocimientos de HTML y CSS, se pueden construir aplicaciones robustas en muy poco tiempo. El sitio web de jQuery cuenta con abundante documentación para aprender.
- Tiene soporte para Ajax, una tecnología para crear aplicaciones dinámicas que se ejecuten del lado cliente.
- No requiere del uso de programas especiales, a diferencia de otras tecnologías, como Adobe Flash, que exigen una aplicación para desarrollar sobre ellas.

c) Desventajas

- Las funciones que ofrece son muchas, pero resultan difíciles de personalizar. Su aspecto visual es estandarizado y no se integra con el de la plataforma.
- Como es necesario invocar a un archivo para utilizar sus funciones, ralentiza levemente la carga de la página.
- Su manejo de CSS suele resultar innecesariamente complejo. A veces cuesta saber qué clases utilizar.
- No existen muchas plantillas prediseñadas sobre las cuales empezar a construir nuestra aplicación. (Parada, 2019)

Figura 2.19 Herramienta JQuery



Fuente: (Parada, 2019)

2.14.8. JSON

JSON se basa en un subconjunto de JavaScript, un lenguaje de programación imperativo, orientado a objetos e interpretado que fue creado por el estadounidense Brendan Eich. De todos modos, mantiene independencia respecto a este lenguaje. En sus orígenes, JSON surgió como una alternativa a XML (eXtensible Markup Language o Lenguaje de Marcado Extensible), un meta-lenguaje de etiquetas. Por la rapidez de lectura y su menor tamaño, JSON logró una rápida aceptación.

El uso de JSON está creciendo rápidamente en toda la industria, debido a que es una tecnología útil. Recientemente Twitter migró de XML a JSON. Google Web Toolkit también trabaja con este formato. (Porto & Gardey, 2019)

a) Características de JSON:

- JSON es solo un formato de datos.
- Requiere usar comillas dobles para las cadenas y los nombres de propiedades. Las comillas simples no son válidas.
- Una coma o dos puntos mal ubicados pueden producir que un archivo JSON no funcione.
- Puede tomar la forma de cualquier tipo de datos que sea válido para ser incluido en un JSON, no solo arreglos u objetos. Así, por ejemplo, una cadena o un número único podrían ser objetos JSON válidos.
- A diferencia del código JavaScript, en el que las propiedades del objeto pueden no estar entre comillas, en JSON solo las cadenas entre comillas pueden ser utilizadas como propiedades.

Utilizar JSON depende de las circunstancias y de las preferencias que en cada momento se determinen, pues cada uno tiene sus ventajas y desventajas. Aquí te mencionamos algunas: (Barrera, 2021)

b) Ventajas:

- Es autodescriptivo y fácil de entender.
- Su sencillez le ha permitido posicionarse como alternativa a XML.
- Es más rápido en cualquier navegador.
- Es más fácil de leer que XML.
- Es más ligero (bytes) en las transmisiones.
- Se parsea más rápido.
- Velocidad de procesamiento alta.

- Puede ser entendido de forma nativa por los analizadores de JavaScript. (Barrera, 2021)

c) Desventajas:

- Algunos desarrolladores encuentran su escueta notación algo confusa.
- No cuenta con una característica que posee XML: extensibilidad.
- No soporta grandes cargas, solo datos comunes.
- Para la seguridad requiere de mecanismos externos como expresiones regulares. (Barrera, 2021)

Figura 2.20 herramienta JSON



Fuente: (Barrera, 2021)

2.14.9. Servidor Apache

Un servidor web es un programa de tipo informático que se encarga de procesar una aplicación del lado del servidor, cada una de las cuales puede acceder a archivos almacenados en un servidor físico y usarlos para diferentes propósitos, mediante conexiones biunidades o uniunidades con la máquina del cliente, tras lo cual se genera una respuesta del lado del cliente.

El trabajo de un servidor web es servir sitios web en Internet. Para lograr ese objetivo, actúa como un intermediario entre el servidor y las máquinas de los clientes. Extrae el contenido del servidor en cada solicitud de usuario y lo envía a la web. El mayor desafío de un servidor web es servir a muchos usuarios diferentes de la web al mismo tiempo, cada uno de los cuales solicita diferentes páginas. Los servidores web procesan archivos escritos en diferentes lenguajes de programación como PHP, Python, Java y otros. Los convierten en archivos HTML estáticos y le entregan estos archivos al navegador de los usuarios de la web. Cuando escuches la palabra servidor web, piensa que es la herramienta responsable de la comunicación adecuada entre el servidor y el cliente.

Apache HTTP Server es un software de servidor web gratuito y de código abierto para plataformas Unix con el cual se ejecutan los sitios web de todo el mundo. Es mantenido y desarrollado por la Apache Software Fundación. (Bustos, 2021)

a) Ventajas

- ✓ Modular.
- ✓ Código abierto.
- ✓ Multi-plataforma.
- ✓ Extensible.
- ✓ Popular (fácil conseguir ayuda/soporte).

b) Desventajas

- ✓ Formatos de configuración no estándar.
- ✓ No cuenta con una buena administración.
- ✓ Falta de integración.

Figura 2.21 Servidor Apache



Fuente: (Bustos, 2021)

2.15. MÉTRICAS DE CALIDAD 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. En la mayoría de los desafíos técnicos, las métricas nos ayudan a entender tanto el proceso técnico que se utiliza para desarrollar un producto, como el propio producto.

El proceso para intentar mejorarlo, el producto se mide para intentar aumentar su calidad.

El principio, podría parecer que la necesidad de la medición es algo evidente después de todo es lo que nos permite cuantificar y por consiguiente gestionar de forma más efectiva.

Pero la realidad puede ser muy diferente. Frecuentemente la medición con lleva una gran controversia y discusión.

- ✓ ¿Cuáles son las métricas apropiadas para el proceso y para el producto?
- ✓ ¿Cómo se deben utilizar los datos que se recopilan?
- ✓ ¿Es bueno usar medidas para comparar gente, procesos o productos?

Estas preguntas y otras tantas docenas de ellas siempre surgen cuando se intenta medir algo que no sea medido en el pasado.

La medición es muy común en el mundo de la ingeniería. Medimos potencia de consumo, pesos, dimensiones físicas, temperaturas, voltajes, señales de ruidos por mencionar algunos aspectos. Desgraciadamente la medición se aleja de lo común en el mundo de la ingeniería del software. Encontramos dificultades en ponernos de acuerdo sobre que medir y como va evaluar las medidas.

Hay varias razones para medir un producto.

- ✓ Para indicar la calidad del producto.
- ✓ Para evaluar la productividad de la gente que desarrolla el producto.
- ✓ Para evaluar los beneficios en términos de productividad y de calidad, derivados de uso de nuevos métodos y herramientas de la ingeniería de software.
- ✓ Para establecer una línea de base para la estimación
- ✓ Para ayudar a justificar el uso de nuevas herramientas o de formación adicional.

2.15.1. 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:

- **Seguridad.** Debido a iniciativas como cero papeles, 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.

2.15.2. Norma ISO/IEC 25000

La calidad del producto, junto con la calidad del proceso, es uno de los aspectos más importantes actualmente en el desarrollo de Software. Relacionada con la calidad del producto, recientemente ha aparecido la familia de normas ISO/IEC 25000, que proporciona una guía para el uso de la nueva serie de estándares internacionales llamada Requisitos y Evaluación de Calidad de Productos de Software (SQuaRE - System and Software Quality Requirements and Evaluation).

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

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

2.15.2.1. ISO/IEC 2500n – División de Gestión de Calidad

Las normas que forman este apartado definen todos los modelos, términos y definiciones comunes referenciados por todas las otras normas de la familia 25000. Actualmente esta división se encuentra formada por:

- ✓ **ISO/IEC 25000** - Guide to SQuaRE: contiene el modelo de la arquitectura de SQuaRE, la terminología de la familia, un resumen de las partes, los usuarios previstos y las partes asociadas, así como los modelos de referencia.
- ✓ **ISO/IEC 25001** - Planning and Management: establece los requisitos y orientaciones para gestionar la evaluación y especificación de los requisitos del producto software.

2.15.2.2. ISO/IEC 2501n – División de Modelo de Calidad

Las normas de este apartado presentan modelos de calidad detallados incluyendo características para calidad interna, externa y en uso del producto software. Actualmente esta división se encuentra formada por:

- ✓ **ISO/IEC 25010** - System and software quality models: describe el modelo de calidad para el producto software y para la calidad en uso. Esta Norma presenta las características y sub características de calidad frente a las cuales evaluar el producto software.
- ✓ **ISO/IEC 25012** - Data Quality model: define un modelo general para la calidad de los datos, aplicable a aquellos datos que se encuentran almacenados de manera estructurada y forman parte de un Sistema de Información.

2.15.2.3. ISO/IEC 2502n – División de Medición de Calidad

Estas normas incluyen un modelo de referencia de la medición de la calidad del producto, definiciones de medidas de calidad (interna, externa y en uso) y guías prácticas para su aplicación. Actualmente esta división se encuentra formada por:

- ✓ **ISO/IEC 25020** - Measurement reference model and guide: presenta una explicación introductoria y un modelo de referencia común a los elementos de medición de la calidad. También proporciona una guía para que los usuarios seleccionen o desarrollen y apliquen medidas propuestas por normas ISO.
- ✓ **ISO/IEC 25021** - Quality measure elements: define y especifica un conjunto recomendado de métricas base y derivadas que puedan ser usadas a lo largo de todo el ciclo de vida del desarrollo software.
- ✓ **ISO/IEC 25022** - Measurement of quality in use: define específicamente las métricas para realizar la medición de la calidad en uso del producto.
- ✓ **ISO/IEC 25023** - Measurement of system and software product quality: define específicamente las métricas para realizar la medición de la calidad de productos y sistemas software.
- ✓ **ISO/IEC 25024** - Measurement of data quality: define específicamente las métricas para realizar la medición de la calidad de datos.

2.15.2.4. ISO/IEC 2503n – División de Requisitos de Calidad

Las normas que forman este apartado ayudan a especificar requisitos de calidad que pueden ser utilizados en el proceso de licitación de requisitos de calidad del producto software a desarrollar o como entrada del proceso de evaluación. Para ello, este apartado se compone de:

- ✓ **ISO/IEC 25030** - Quality requirements: provee de un conjunto de recomendaciones para realizar la especificación de los requisitos de calidad del producto software.

2.15.2.5. ISO/IEC 2504n – División de Evaluación de Calidad

Este apartado incluye normas que proporcionan requisitos, recomendaciones y guías para llevar a cabo el proceso de evaluación del producto software. Esta división se encuentra formada por:

- ✓ **ISO/IEC 25040** - Evaluation reference model and guide: propone un modelo de referencia general para la evaluación, que considera las entradas al proceso de evaluación, las restricciones y los recursos necesarios para obtener las correspondientes salidas.
- ✓ **ISO/IEC 25041** - Evaluation guide for developers, acquirers and independent evaluators: describe los requisitos y recomendaciones

para la implementación práctica de la evaluación del producto software desde el punto de vista de los desarrolladores, de los adquirentes y de los evaluadores independientes.

- ✓ **ISO/IEC 25042** - Evaluation modules: define lo que la Norma considera un módulo de evaluación y la documentación, estructura y contenido que se debe utilizar a la hora de definir uno de estos módulos.
- ✓ **ISO/IEC 25045** - Evaluation module for recoverability: define un módulo para la evaluación de la su característica Recuperabilidad (Recoverability).

Figura 2.22 División de calidad de Software



Fuente: (ISO, 2021)

La calidad del producto software forma parte central como lo menciona ISO/IEC 25000, relacionando directamente las variables de funcionalidad, fiabilidad, usabilidad, eficiencia, mantenibilidad y portabilidad.

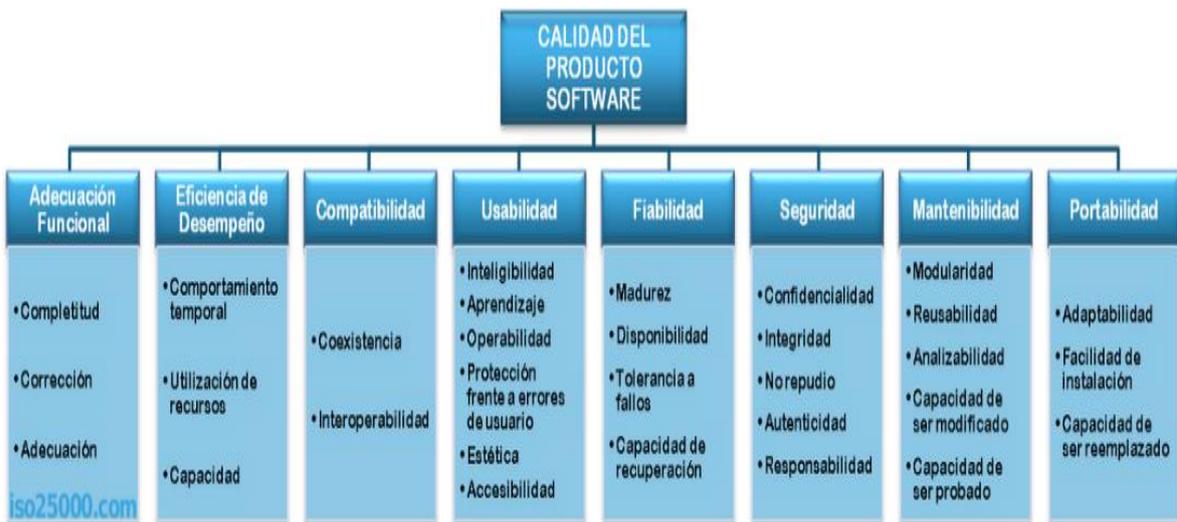
Según Monsalve (2010) “La calidad que pueden alcanzar los productos software, y en general cualquier producto, está sometida a como se desarrolla cada una de las etapas de la vida del producto, partiendo por la definición de la idea del producto hasta la entrega y mantención del mismo”.

ISO/IEC 25000 define tres vistas diferenciadas en el estudio de la calidad de un producto:

- **Vista interna:** Se ocupa de las propiedades del sistema como: el tamaño, la complejidad o la conformidad con las normas de orientación a objetos. Puede utilizarse desde las primeras fases del desarrollo, permitiendo detectar deficiencias en el sistema en etapas tempranas del ciclo de vida.
- **Vista externa:** Analiza el comportamiento del sistema en producción y estudia sus atributos, por ejemplo: el rendimiento de un software en una máquina determinada, el uso de memoria de un programa o el tiempo de funcionamiento entre fallos.
Esta vista necesita que el sistema esté completo y en la fase de producción del producto, siendo muy dependiente de la máquina donde se ejecute.
- **Vista en uso:** Mide la productividad y efectividad del usuario final al utilizar el sistema.

Por último, la tercera vista que también estudia el producto software finalizado será dependiente del usuario y estará condicionada a los factores personales del mismo.

Figura 2.23 Calidad del producto de software



Fuente: (ISO, 2021)

2.15.3. Clasificación de las características y sub características

La Calidad del Software se clasifica en un conjunto estructurado de características y sub características de la siguiente manera:

2.15.3.1. Adecuación funcional

Representa la capacidad del producto software para proporcionar funciones que satisfacen las necesidades declaradas e implícitas, cuando el producto se usa en

las condiciones especificadas. Esta característica se subdivide a su vez en las siguientes sub características:

- **Completitud funcional:** Grado en el cual el conjunto de funcionalidades cubre todas las tareas y los objetivos especificados por el usuario.
- **Corrección funcional:** Capacidad del producto o sistema para proveer resultados correctos con el nivel de precisión requerido.
- **Adecuación funcional:** Capacidad del producto software para proporcionar un conjunto apropiado de funciones para tareas y objetivos de usuario especificados.

2.15.3.2. Eficiencia de desempeño

Esta característica representa el desempeño relativo a la cantidad de recursos utilizados bajo determinadas condiciones. Esta característica se subdivide a su vez en las siguientes sub características:

- **Comportamiento temporal:** Los tiempos de respuesta y procesamiento y las ratios de rendimiento de un sistema cuando lleva a cabo sus funciones bajo condiciones determinadas en relación con un banco de pruebas (benchmark) establecido.
- **Utilización de recursos:** Las cantidades y tipos de recursos utilizados cuando el software lleva a cabo su función bajo condiciones determinadas

2.15.3.3. Compatibilidad

- **Coexistencia:** Capacidad del producto para coexistir con otro software independiente, en un entorno común, compartiendo recursos comunes sin detrimento.
- **Interoperabilidad:** Capacidad de dos o más sistemas o componentes para intercambiar información y utilizar la información intercambiada.

2.15.3.4. Usabilidad

Capacidad del producto software para ser entendido, aprendido, usado y resultar atractivo para el usuario, cuando se usa bajo determinadas condiciones. Esta característica se subdivide a su vez en las siguientes sub características:

- **Capacidad para reconocer su adecuación:** Capacidad del producto que permite al usuario entender si el software es adecuado para sus necesidades.

- **Capacidad de aprendizaje técnico:** Capacidad del producto que permite al usuario aprender su aplicación.
- **Capacidad para ser usado:** Capacidad del producto que permite al usuario operarlo y controlarlo con facilidad.
- **Protección contra errores de usuario:** Capacidad del sistema para proteger a los usuarios de hacer errores.
- **Estética de la interfaz de usuario:** Capacidad de la interfaz de usuario de agrandar y satisfacer la interacción con el usuario.
- **Accesibilidad técnica:** Capacidad del producto que permite que sea utilizado por usuarios con determinadas discapacidades.

Formula:

$$\textit{Funcionalidad} = \frac{PF}{PF \textit{ max}}$$

2.15.3.5. Fiabilidad

Capacidad de un sistema o componente para desempeñar las funciones especificadas, cuando se usa bajo unas condiciones y periodo de tiempo determinados. Esta característica se subdivide a su vez en las siguientes sub características:

- **Madurez:** Capacidad del sistema para satisfacer las necesidades de fiabilidad en
- condiciones normales.
- **Disponibilidad:** Capacidad del sistema o componente de estar operativo y accesible para su uso cuando se requiere.
- **Tolerancia a fallos:** Capacidad del sistema o componente para operar según lo previsto en presencia de fallos hardware o software.
- **Capacidad de recuperación:** Capacidad del producto software para recuperar los datos directamente afectados y restablecer el estado deseado del sistema en caso de interrupción o fallo.

2.15.3.6. Seguridad

Capacidad de protección de la información y los datos de manera que personas o sistemas no autorizados no puedan leerlos o modificarlos. Esta característica se subdivide a su vez en las siguientes sub características:

- **Confidencialidad:** Capacidad de protección contra el acceso de datos e información no autorizados, ya sea accidental o deliberadamente.
- **Integridad:** Capacidad del sistema o componente para prevenir accesos o modificaciones no autorizados a datos o programas de ordenador.
- **No repudio:** Capacidad de demostrar las acciones o eventos que han tenido lugar, de manera que dichas acciones o eventos no puedan ser repudiados posteriormente.
- **Responsabilidad:** Capacidad de rastrear de forma inequívoca las acciones de una entidad.
- **Autenticidad:** Capacidad de demostrar la identidad de un sujeto o un recurso.

2.15.3.7. Mantenibilidad

Esta característica representa la capacidad del producto software para ser modificado efectiva y eficientemente, debido a necesidades evolutivas, correctivas o perfectivas. Esta característica se subdivide a su vez en las siguientes sub características:

- **Modularidad:** Capacidad de un sistema o programa de ordenador (compuesto de componentes discretos) que permite que un cambio en un componente tenga un impacto mínimo en los demás.
- **Reusabilidad:** Capacidad de un activo que permite que sea utilizado en más de un sistema software o en la construcción de otros activos.
- **Analizabilidad:** Facilidad con la que se puede evaluar el impacto de un determinado cambio sobre el resto del software, diagnosticar las deficiencias o causas de fallos en el software, o identificar las partes a modificar.
- **Capacidad para ser modificado:** Capacidad del producto que permite que sea modificado de forma efectiva y eficiente sin introducir defectos o degradar el desempeño.
- **Capacidad para ser probado:** Facilidad con la que se pueden establecer criterios de prueba para un sistema o componente y con la que se pueden llevar a cabo las pruebas para determinar si se cumplen dichos criterios.

Formula:

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

2.15.3.8. Portabilidad

Capacidad del producto o componente de ser transferido de forma efectiva y eficiente de un entorno hardware, software, operacional o de utilización a otro.

Esta característica se subdivide a su vez en las siguientes sub características:

- **Adaptabilidad:** Capacidad del producto que le permite ser adaptado de forma efectiva y eficiente a diferentes entornos determinados de hardware, software, operacionales o de uso.
- **Capacidad para ser instalado:** Facilidad con la que el producto se puede instalar y/o desinstalar de forma exitosa en un determinado entorno.
- **Capacidad para ser reemplazado:** Capacidad del producto para ser utilizado en lugar de otro producto software determinado con el mismo propósito y en el mismo entorno.

2.16. ANÁLISIS DE COSTO DE SOFTWARE COCOMO

2.16.1. Cocomo II

Modelo COCOMO II, modelo de estimación que se encuentra en la jerarquía de modelos de estimación de software con el nombre de COCOMO, por Constructive Cost Model (Modelo Constructivo de Coste). El modelo COCOMO original se ha convertido en uno de los modelos de estimación de coste del software más utilizados y estudiados en la industria. (COCOMO, 2013)

2.16.2. Características

- Es una herramienta basada en las líneas de código la cual la hace muy poderosa para la estimación de costos y no como otros que solamente miden el esfuerzo en base al tamaño.
- Representa el más extenso modelo empírico para la estimación de software.
- Existen herramientas automáticas que estiman costos basados en COCOMO como ser: Costar, COCOMO 81.

2.16.3. Ecuaciones del modelo cocomo

Las ecuaciones que se utilizan en los tres modelos son:

$$E = a(KI)^b * m(X), \text{ en persona - mes.}$$

$$Tdev = c(E)^d, \text{ en meses.}$$

$$P = \frac{E}{Tdev}, \text{ en personas.}$$

Dónde:

- *E* = Es el esfuerzo requerido por el proyecto, en persona – mes.
- *Tdev* = Es el esfuerzo requerido por el proyecto, en meses.
- *P* = Es el numero de personas requerido por el proyecto.
- *a, b, c y d*, Son constantes con valores definidos en una tabla,
- *KI* = Es la cantidad de lineas de codigo, en miles.
- *m(X)* = Es un multiplicador que depende de 15 atributos.

A la vez, cada SubModelo también se divide en modos que representan el tipo de proyecto, y puede ser:

- ✓ **Modo Orgánico:** Un pequeño grupo de programadores experimentados desarrollan software en un entorno familiar. El tamaño del software varía desde unos pocos miles de líneas (tamaño pequeño) a unas decenas de miles (medio).
- ✓ **Modo Semilibre o Semiencajado:** Corresponde a un esquema intermedio entre el orgánico y el rígido; el grupo de desarrollo puede incluir una mezcla de personas experimentadas y no experimentadas.
- ✓ **Modo Rígido o Empotrado:** El proyecto tiene fuertes restricciones, que pueden estar relacionadas con la funcionalidad y/o pueden ser técnicas. El problema a resolver es único y es difícil basarse en la experiencia, puesto que puede no haberla.

2.16.4. Modelos de Cocomo

2.16.4.1. Modelo Básico:

Se utiliza para obtener una primera aproximación rápida del esfuerzo, y hace uso de la siguiente tabla de constantes para calcular distintos aspectos de costes.

Tabla 2.2 Constantes de Costes

Modo	a	b	c	d
Orgánico	2.40			0.38
Semilibre		1.05 2.50		0.35
	3.00	1.12	2.50	
Rígido	3.60	1.20	2.50	0.32

Fuente: (COCOMO, 2013)

Estos valores son para las fórmulas:

- Personas necesarias por mes para llevar adelante el proyecto

$$(MM) = a * (KI^b)$$

- Tiempo de desarrollo del proyecto.

$$(Tdev) = c * (MM^d)$$

- Personas necesarias para realizar el proyecto.

$$(CosteH) = MM/Tdev$$

- Costo total del proyecto

$$(CosteM = CosteH * \text{salario medio entre los programadores y analistas.})$$

Se puede observar que a medida que aumenta la complejidad del proyecto (modo), las constantes aumentan de 2.4 a 3.6, que corresponde a un incremento del esfuerzo del personal. Hay que utilizar con mucho cuidado el modelo básico puesto que se obvian muchas características del entorno

2.16.4.2. Modelo Intermedio

En este modelo se introducen 15 atributos de coste para tener en cuenta en el entorno de trabajo, incrementando así la precisión de la estimación.

Para este ajuste, al resultado de la fórmula general se lo multiplica por el coeficiente surgido de aplicar los atributos que se decidan utilizar.

Los valores de las constantes a reemplazar en la fórmula se muestran en la siguiente tabla:

Tabla 2.3 Coeficientes de a,b,c,d de COCOMO II

Proyecto Software	a	b	c	d
Orgánico	3.2	1.05	2.5	0.38
Semi acoplado	3.0	1.12	2.5	0.35
Empotrado	2.8	1.20	2.5	0.32

Fuente: (COCOMO, 2013)

Se puede observar que los exponentes son los mismos que los del modelo básico, confirmando el papel que representa el tamaño; mientras que los coeficientes de los modos orgánico y rígido han cambiado, para mantener el equilibrio al rededor del SemiLibre con respecto al efecto multiplicador de los atributos de coste.

2.13.4.2.1. Ecuaciones Nominales De Coste

Para cada modo de desarrollo, los 15 atributos del coste intervienen como multiplicadores en el coste nominal, Kn , para producir el coste ajustado.

Las ecuaciones nominales de coste para el modelo intermedio son:

Tabla 2.4 Ecuaciones nominales de Coste

Modo orgánico	$Kn = 3.2 Sk 1.05$
Modo semiencajado	$Kn = 3.0 Sk 1.12$
Modo empotrado	$Kn = 2.8 Sk 1.20$

Fuente: (COCOMO, 2013)

2.13.4.2.2. Atributos de Coste

Cada atributo se cuantifica para un entorno de proyecto. La escala es muy baja bajo - nominal - alto - muy alto - extremadamente alto. Dependiendo de la calificación de cada atributo, se asigna un valor para usar de multiplicador en la fórmula el significado de los atributos es el siguiente, según su tipo de:

Software

Tabla 2.5 Atributo de Costes de Software

RELY	<p>Garantía de funcionamiento requerida al software.</p> <p>Indica las posibles consecuencias para el usuario en el caso que existan defectos en el producto.</p> <p>Va desde la sola inconveniencia de corregir un fallo (muy bajo) hasta la posible pérdida de vidas humanas (extremadamente alto, software de alta criticidad).</p>
DATA	<p>Tamaño de la base de datos en relación con el tamaño del programa. El valor del modificador se define por la relación:</p> <p>D/K</p> <p>Donde:</p> <ul style="list-style-type: none"> ✓ D corresponde al tamaño de la base de datos en bytes. ✓ K es el tamaño del programa en cantidad de líneas de código.
CPLX	Representa la complejidad del producto.

Fuente: (COCOMO, 2013)

HARDWARE

Tabla 2.6 Atributo de Costes de Hardware

TIME	Limitaciones en el porcentaje del uso de la CPU.
STOR	Limitaciones en el porcentaje del uso de la memoria.
VIRT	Volatilidad de la máquina virtual.
TURN	Tiempo de respuesta requerido.

Fuente: (COCOMO, 2013)

PERSONAL

Tabla 2.7 Atributo de Costes Personal

ACAP	Calificación de los analistas.
AEXP	Experiencia del personal en aplicaciones similares.
PCAP	Calificación de los programadores.
VEXP	Experiencia del personal en la máquina virtual.
LEXP	Experiencia en el lenguaje de programación a usar.

Fuente: (COCOMO, 2013)

PROYECTO

Tabla 2.8 Atributo de Costes de Proyecto

MODP	Uso de prácticas modernas de programación.
TOOL	Uso de herramientas de desarrollo de software.
SCED	Limitaciones en el cumplimiento de la planificación.

Fuente: (COCOMO, 2013)

Tabla 2.9 Atributos de Costes Totales

Valor						
Atributos	Muy bajo	Bajo	Nominal	Alto	Muy alto	Extra alto
Atributos de software						
Fiabilidad	0,75	<u>0,88</u>	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		<u>0,87</u>	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	<u>0,86</u>	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	
Valor						
Atributos	Muy bajo	Bajo	Nominal	Alto	Muy alto	Extra alto
Atributos de software						
Fiabilidad	0,75	<u>0,88</u>	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		<u>0,87</u>	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: (COCOMO, 2013)

2.16.4.3. Modelo Detallado

Incluye todo lo del modelo intermedio además del impacto de cada conductor de coste en las distintas fases de desarrollo. Presenta principalmente dos mejoras respecto al modelo básico e Intermedio.

Los factores correspondientes a los atributos son sensibles o dependientes de la fase sobre la que se realizan las estimaciones.

Aspectos tales como la experiencia en la aplicación, utilización de herramientas de software, etc., tienen mayor influencia en unas fases que en otras, y además van variando de una etapa a otra.

Establece una jerarquía de tres niveles de productos, de forma que los aspectos que representan gran variación a bajo nivel, se consideran a nivel módulo, los que representan pocas variaciones, a nivel de subsistema; y los restantes son considerados a nivel sistema.

Para nuestro caso el modelo intermedio será el que usaremos, dado que realiza las estimaciones con bastante precisión.

2.17. SEGURIDAD

2.17.1. 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.

2.17.2. Aspectos de seguridad

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)

2.18. PRUEBAS DE SOFTWARE

2.18.1. 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. (Anonimo, 2011)

2.18.2. 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. (Anonimo, 2011)

2.18.3. Pruebas de estrés

Una prueba de estrés (stress) consiste en probar los límites que un sistema puede soportar. En este tipo de pruebas se suele enviar más peticiones de las que el software podría atender normalmente para saber el comportamiento de la aplicación.

Por otro lado, las pruebas de estrés (stress testing) son realizadas sobrecargando un sistema más allá de sus especificaciones, para verificar cómo y cuándo fallará. Dentro de informática podemos colocar una gran carga en la base de datos, entradas (peticiones) continuas al sistema o almacenar información más allá de la capacidad de memoria del sistema. (Platzi, 2018)

**SISTEMA DE REGISTRO DE ADMISIÓN DE ESTUDIANTES
DE LA MODALIDAD DE EXCELENCIA ACADÉMICA
CASO: VICERRECTORADO UPEA**

CAPÍTULO III

MARCO APLICATIVO



3.1. INTRODUCCIÓN

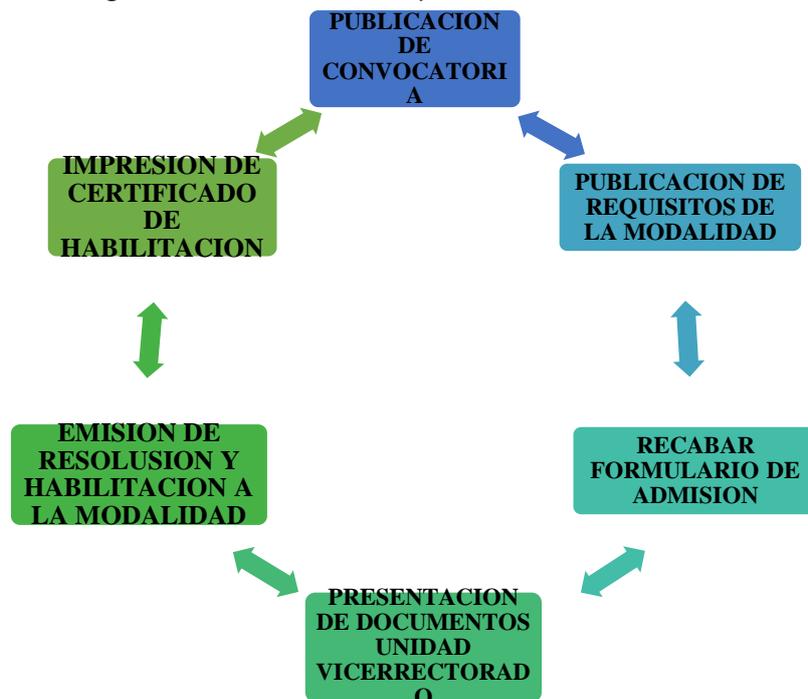
En este capítulo se hará énfasis a todo lo que significa el proceso de desarrollo de software; desde los requerimientos del usuario hasta la prueba final del sistema. Cada una de las actividades que son necesarias para la transformación de los requisitos del usuario en un sistema software, serán detalladas de acuerdo y siguiendo los pasos que emplea la metodología UWE.

3.2. ANÁLISIS DE LA SITUACIÓN ACTUAL

En el proceso de admisión de la modalidad de excelencia académica se realiza actualmente con el procedimiento de que una vez lanzada la convocatoria el estudiante que cuenta con el certificado de ser mejor estudiante y la posición pueda postularse a la modalidad realizando lo siguiente:

- Recaba un formulario de admisión.
- Se presenta a la unidad de vicerrectorado con los requisitos correspondientes que pide la unidad.
- El estudiante espera que la unidad de vicerrectorado emita la resolución que lo habilita a la modalidad.
- El coordinador imprime el certificado de habilitación

Figura 3.1 Proceso de inscripción excelencia académica



Fuente: (Elaboración Propia)

Actualmente en la Unidad de Vicerrectorado existe un gran movimiento de información y documentación generado año tras año se tienen como consecuencia lo siguiente:

- ✓ Pérdida de tiempo
- ✓ Documentos duplicados
- ✓ Mala atención

3.3. INGENIERÍA 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:

Tabla 3.1 Herramientas de los requerimientos

HERRAMIENTAS	DESCRIPCIÓN
OBSERVACION	Se realizó la observación del proceso de la administración de estudiantes así como las funciones desempeñan cada uno de los interesados.
ENCUESTA	Se realizó entrevistas con el personal a cargo del proceso: <ul style="list-style-type: none"> • Personal de vicerrectorado. • Técnico de vicerrectorado. • Postulante.

Fuente: (Elaboración Propia)

Ingeniería de requerimientos es el conjunto de actividades en las cuales, utilizando técnicas y herramientas, se analiza un problema y se concluye con la especificación de una solución (a veces más de una).

Con el análisis de requerimientos se empleó las herramientas de la metodología UWE que son el uso de los diagramas del caso de uso, para eso es la necesaria la identificación de los actores que interactúan en el proceso.

Identificación de tipos de usuario (roles)

La identificación y clasificación de los tipos de usuarios permite organizar la información evitando confusión y otorga mejor control y seguimiento a los usuarios para el funcionamiento a los usuarios para el funcionamiento y acceso al mismo el cual hara que sea eficiente.

En la siguiente tabla 3.2 se muestra la clasificación e identificación de los tipos de usuarios (roles) que tendrán acceso al sistema.

Tabla 3.2 Clasificación de los roles

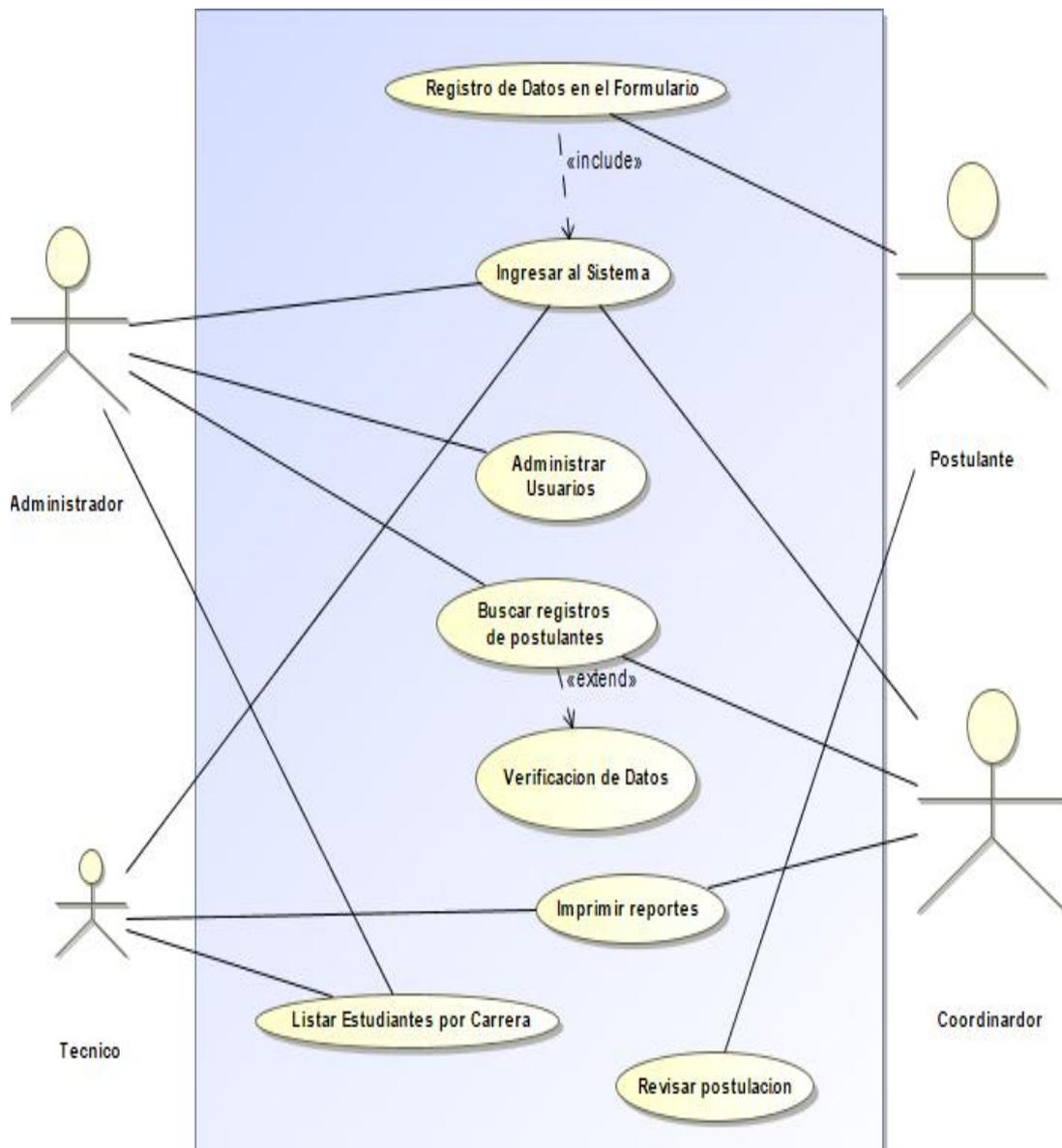
ROL	DESCRIPCION
Administrador del sistema	<ul style="list-style-type: none">✓ Se encarga de administrar los usuarios dentro del sistema.✓ Se encarga de la configuración del sistema.✓ Realiza búsquedas y listados de los postulantes de todas las carreras.
Técnico	<ul style="list-style-type: none">✓ Se encarga de la revisión de los documentos del postulante.✓ Habilitación de postulante a la modalidad.✓ Listado de los postulantes de las distintas carreras.✓ Se encarga del control de postulantes de la modalidad de Excelencia.
Coordinador	<ul style="list-style-type: none">✓ Se encarga de la impresión de certificados de habilitación y reportes de actas de notas.✓ Se encarga de la habilitación de fechas de inicio y conclusión para la inscripción de postulantes, fechas de impresión de actas.
Postulante	<ul style="list-style-type: none">✓ Revisión de la convocatoria.✓ Registro en el sistema mediante el llenado del formulario de postulación.✓ Revisión de datos personales del postulante.

Fuente: (Elaboración Propia)

3.4. MODELO DE CASO DE USO

En el presente numeral se plasmará el análisis de requerimientos del sistema mediante el diseño de casos de uso mismos expresados en el comportamiento del sistema frente a las acciones de los actores del mismo, funcionalidades del sistema y además elementos que permiten la abstracción del problema.

Figura 3.2 Casos de uso general

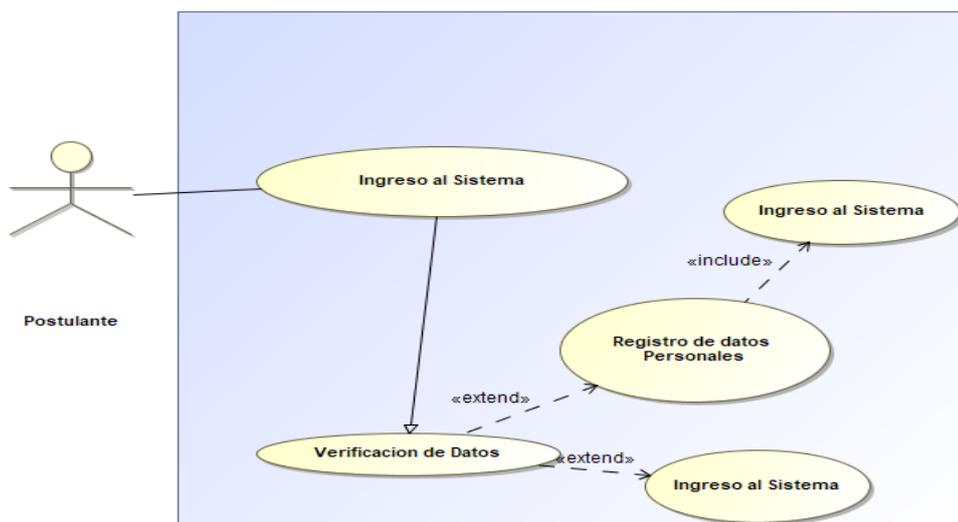


Fuente: (Elaboración propia)

3.1.2.1. Caso de uso para la admisión de usuarios

En la siguiente figura 3.3 y la tabla 3.3 se muestra el caso de uso y la descripción del caso de uso de la admisión de usuarios.

Figura 3.3 Caso de uso para la admisión de usuarios



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 ingresar a la página.
Flujo de Eventos:	<p>El sistema solicita al postulante que ingrese su carnet y fecha de nacimiento.</p> <p>El sistema valida que el postulante no se encuentre inscrito en ninguna de las otras modalidades de ingreso, 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>

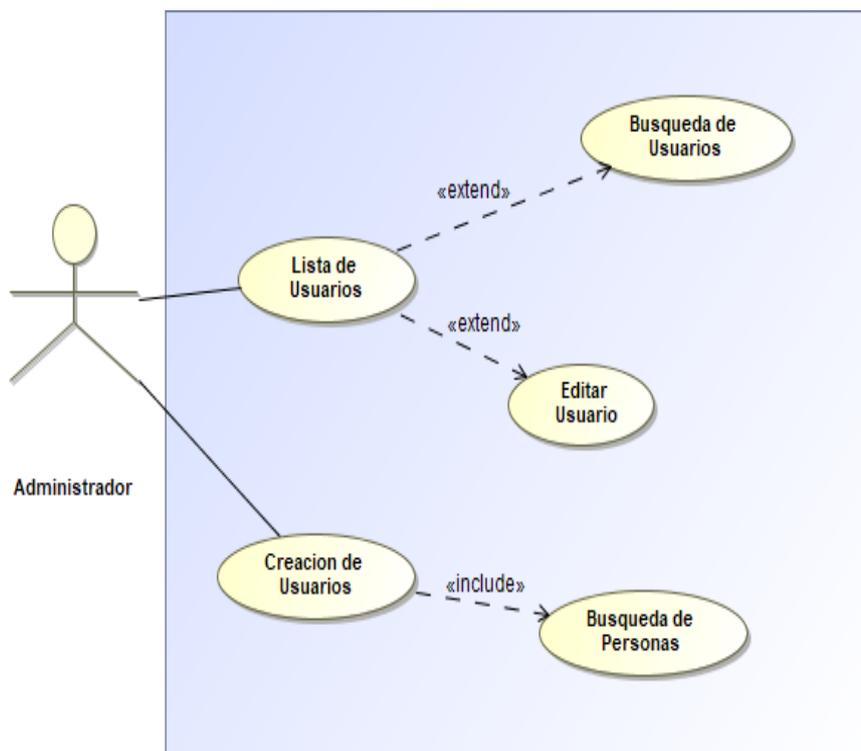
	El sistema valida los datos del formulario, registra los datos y genera el formulario.
Post condición:	El postulante se registra exitosamente
Excepciones:	<ul style="list-style-type: none"> ✓ Si el usuario se encuentra previamente registrado, el sistema muestra un mensaje de error. ✓ Si el postulante se encuentra matriculado, el sistema muestra un mensaje de error. ✓ 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.

Fuente: (Elaboración Propia)

3.1.2.2. Caso de uso para la administración de usuarios

En la siguiente figura 3.4 y la tabla 3.4 se muestra el caso de uso y la descripción de la administración de usuarios.

Figura 3.4 Caso de uso Administración de usuarios



Fuente: (Elaboración Propia)

Tabla 3.4 Descripción de los casos de uso Administración de usuarios

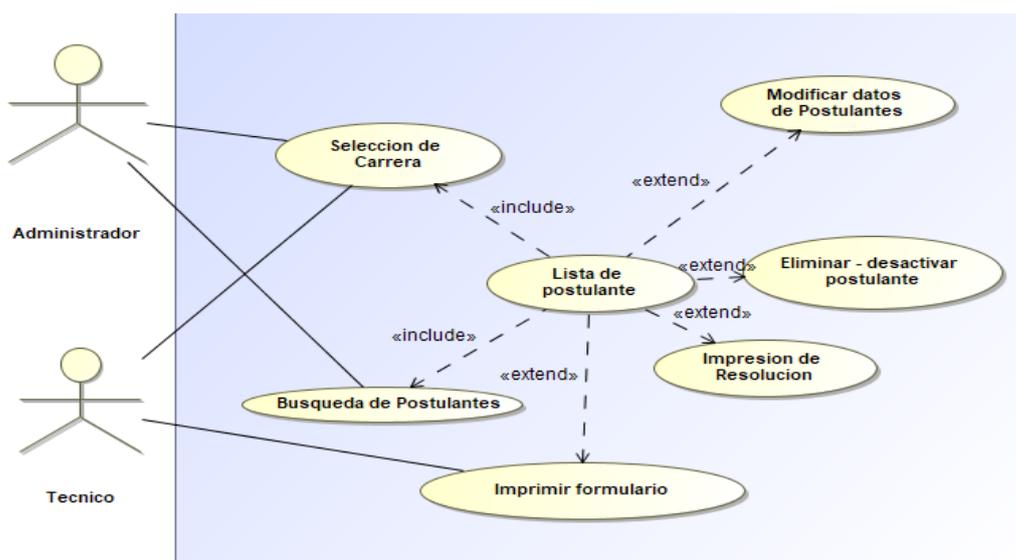
Caso de uso	Administración de usuarios
ID:	CU02
Descripción:	El sistema permite administrar los usuarios registrados en el mismo.
Actores:	Administrador
Precondiciones:	El usuario debe haber iniciado sesión, el usuario debe ser administrador.
Flujo de Eventos:	<p>El usuario solicita el listado de usuarios 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>
Post condición:	El usuario logra crear, modificar o eliminar/desactivar usuarios.
Excepciones	Si el usuario trata de crear o modificar un usuario con campos que ya están registrados sale error y se cancela el caso de uso.

Fuente: (Elaboración propia)

3.1.2.3. Caso de uso para la administración de postulantes

En la siguiente figura 3.5 y la tabla 3.5 se muestra el caso de uso y la descripción de la administración de postulantes.

Figura 3.5 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
Descripción:	Describe el proceso de la administración de los postulantes
Actores:	Técnico /Administrador
Precondiciones:	El usuario debe haber iniciado sesión, el usuario debe ser técnico, administrador o coordinador.
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.</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>

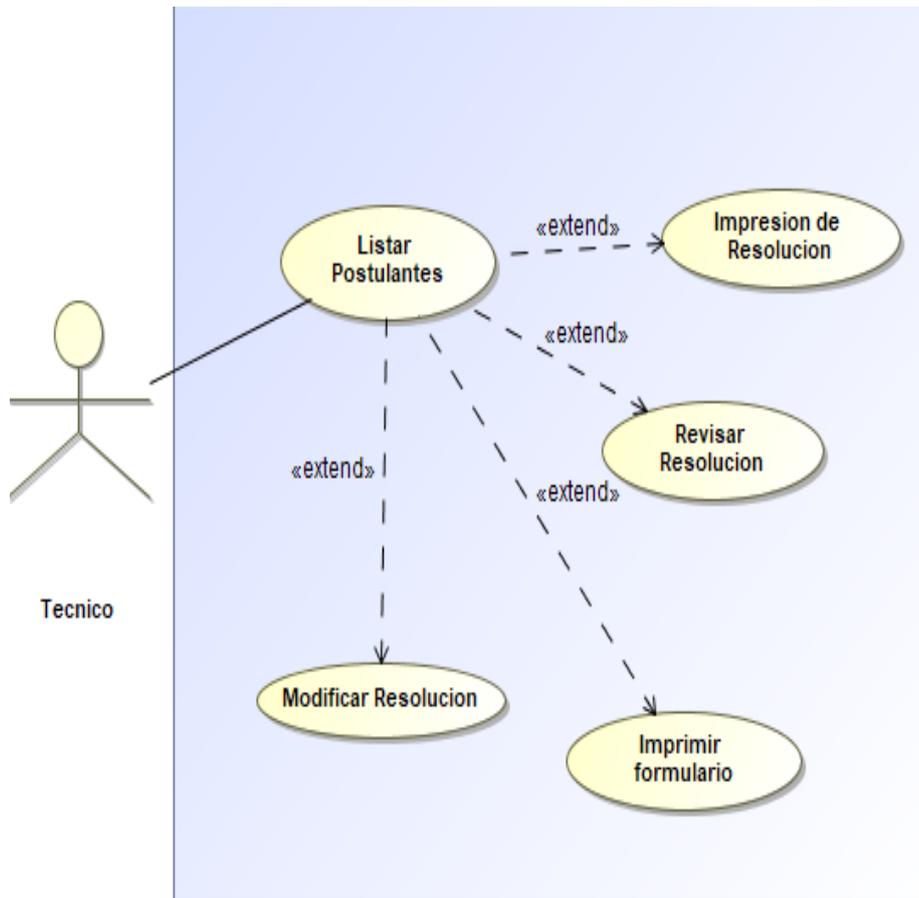
	<p>El usuario solicita hacer búsqueda de usuarios.</p> <p>El sistema le solicita los datos para realizar la búsqueda.</p> <p>El sistema muestra los registros que coinciden con los datos obtenidos.</p>
Post condición:	El usuario habrá administrado los registros del postulante correctamente.
Excepciones	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 modalidad de la excelencia académica

En la siguiente figura 3.6 y la tabla 3.6 se muestra el caso de uso y la descripción de la modalidad de excelencia académica.

Figura 3.6 Gestión de la modalidad excelencia académica



Fuente: (Elaboración Propia)

Tabla 3.6 Descripción del caso de uso de la modalidad excelencia académica

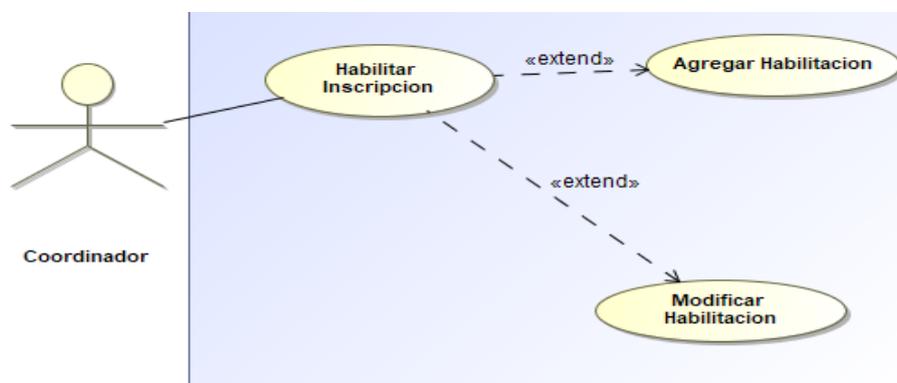
Caso De Uso	GESTIÓN DE MODALIDAD DE EXCELENCIA ACADÉMICA
ID:	CU04
Descripción:	Describe el proceso empleado para la gestión de la modalidad de excelencia académica.
Actores:	Técnico/ Coordinador
Precondiciones:	El usuario debe haber iniciado sesión, el usuario debe ser técnico o coordinador.
Flujo de Eventos:	<p>El usuario técnico solicita al sistema el listado de las carreras habilitadas.</p> <p>El usuario coordinador solicita al sistema el listado de cursos.</p> <p>Si el usuario es técnico el sistema le mostrar la opción de listar postulantes. Si el usuario es coordinador le mostrara las opciones de crear, modificar y listar postulantes.</p> <p>El sistema solicita los datos necesarios para la acción, realiza la validación y efectúa la creación o modificación.</p> <p>El usuario técnico coordinador solicita al sistema el listado de estudiantes.</p> <p>El sistema muestra la lista de estudiantes y le da la opción de descargar certificado de habilitación.</p>
Post condición:	El usuario habrá gestionado a los postulantes.
Excepciones	Si la carrera no está habilitada a esta modalidad no podrá realizar las distintas acciones.

Fuente: (Elaboración Propia)

3.1.2.5. Caso de uso para la habilitación de fechas

En la siguiente figura 3.7 y la tabla 3.7 se muestra el caso de uso y la descripción de la habilitación de fechas.

Figura 3.7 Caso de uso Habilidad de fechas



Fuente: (Elaboración Propia)

Tabla 3.7 Descripción del caso de uso Habilidad de fechas

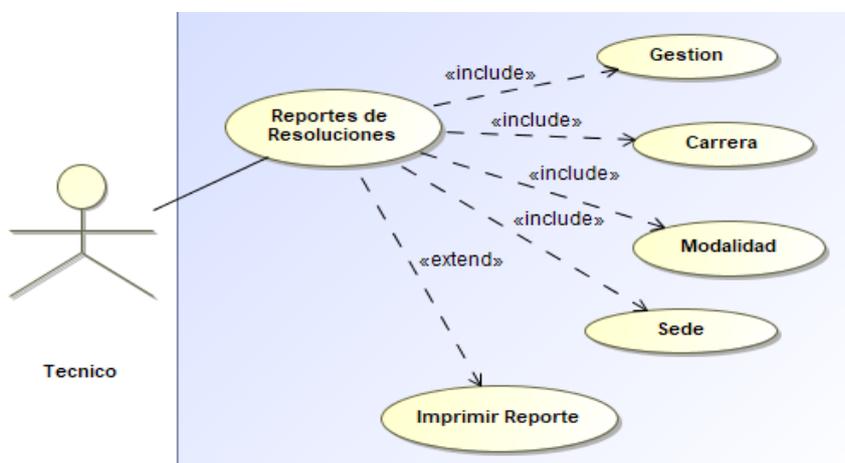
Caso de uso	GESTIÓN DE HABILITACIÓN DE FECHAS
ID:	CU05
Descripción:	Describe el proceso que se realiza para la habilitación de fechas de apertura y finalización de inscripciones de la modalidad de excelencia académica.
Actores:	Coordinador
Precondiciones:	El usuario debe haber iniciado sesión, el usuario debe ser coordinador.
Flujo de Eventos:	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.
Post condición:	El usuario habrá habilitado las fechas para inscripciones.
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)

3.1.2.6. Caso de uso para la gestión de reportes

En la siguiente figura 3.8 y la tabla 3.8 se muestra el caso de uso y la descripción de la gestión de reportes.

Figura 3.8 Caso de uso Gestión de reportes



Fuente: (Elaboración Propia)

Tabla 3.8 Descripción del caso de uso Gestión de reportes

Caso de uso:	GESTIÓN DE REPORTES
ID:	CU06
Descripción:	Describe el proceso empleado para la impresión de reportes
Actores:	Técnico
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 resoluciones.</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</p>

	descarga.
Post condición:	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

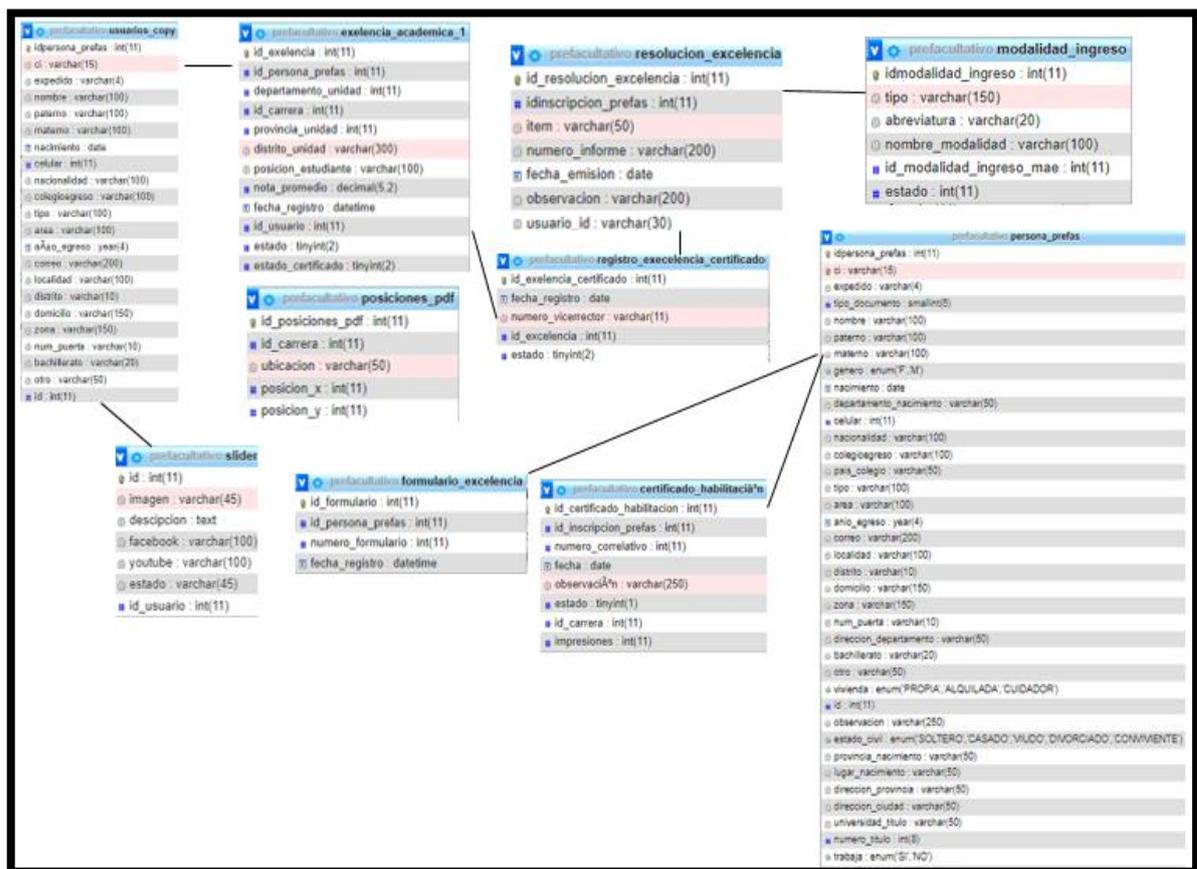
Fuente: (Elaboración Propia)

3.5. MODELO CONTENIDO

El diagrama de contenido tiene por propósito mostrar las relaciones entre las entidades y la estructura de los datos que se encuentran alojados en el sistema el modelo de contenido contiene la información relevante almacenada en el sistema como se estructura como se relaciona. Esto se representa mediante un diagrama de clases UML.

3.5.1. Diagrama de clases

Figura 3.9 Diagrama de Clases para el sistema de admisión excelencia

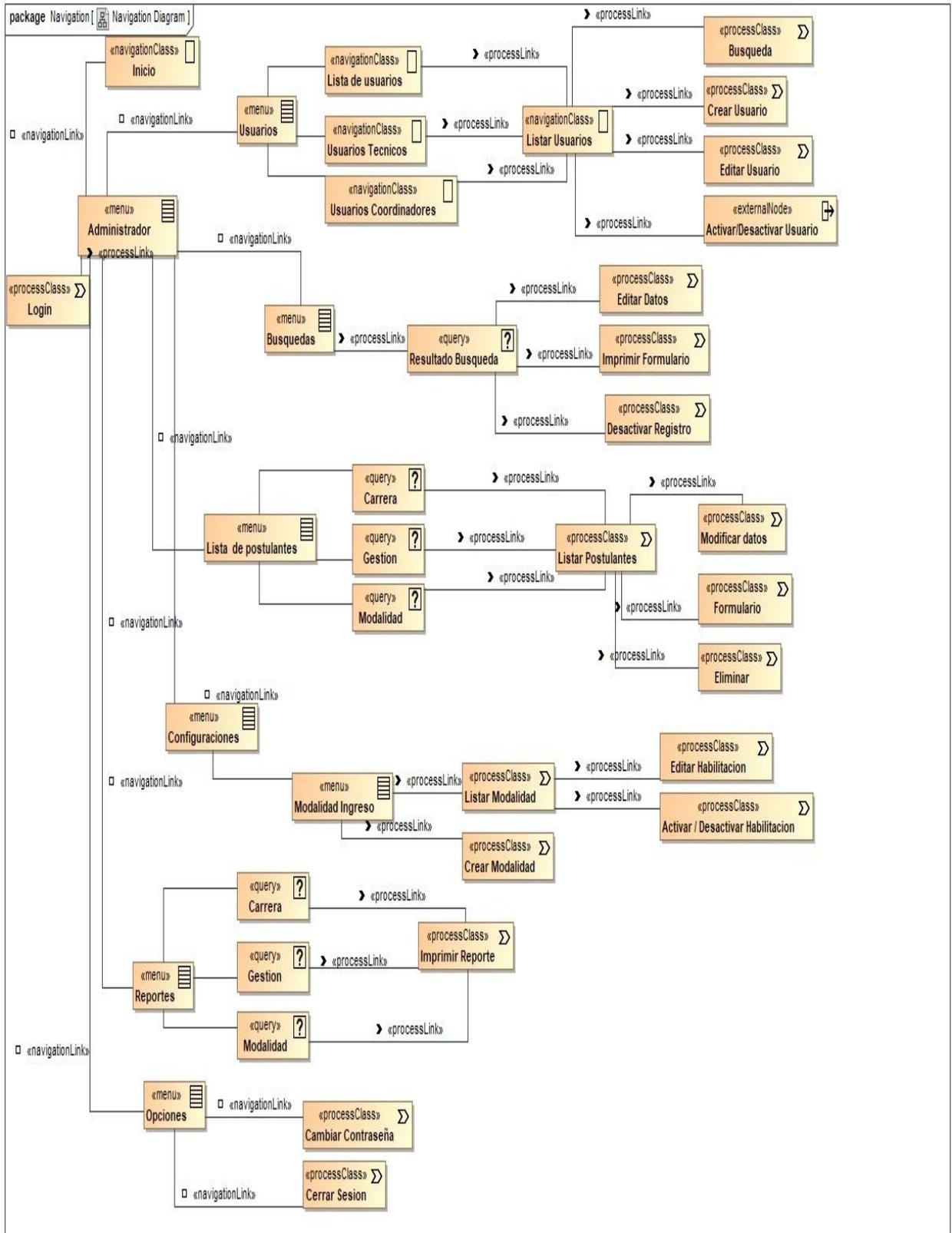


Fuente: (Elaboración propia)

3.6. MODELO DE NAVEGACIÓN

En la siguiente figura 3.10 se presenta el modelo navegación del sistema.

Figura 3.10 Modelo de navegación para administrador

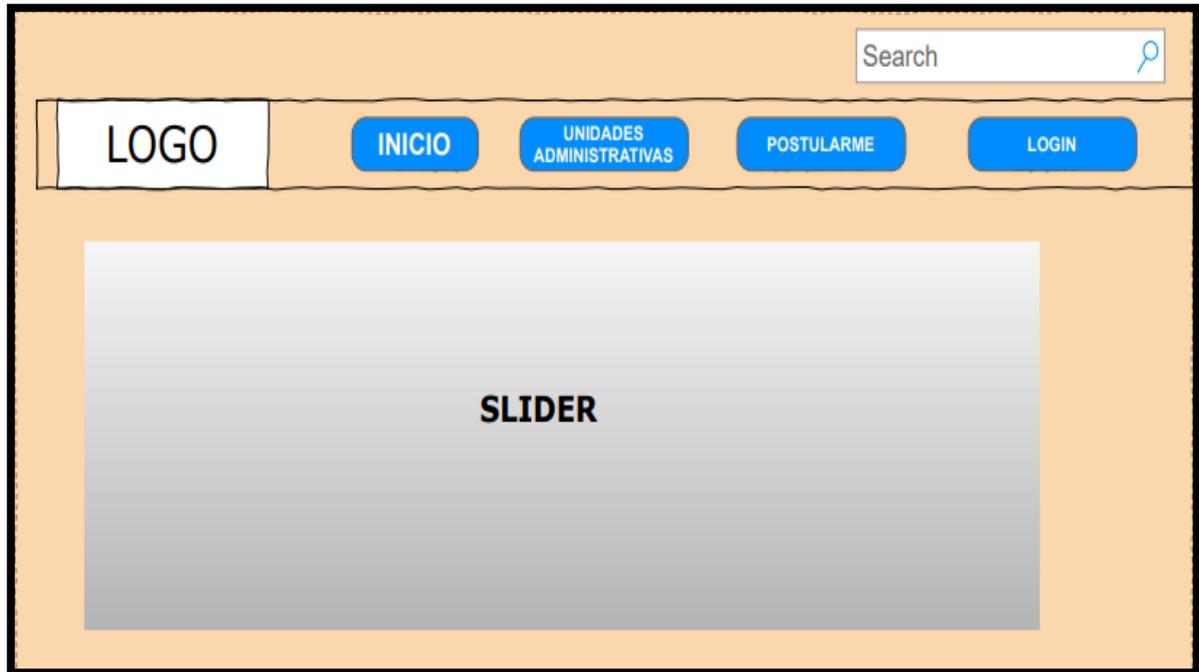


Fuente: (Elaboración propia)

3.7. MODELO DE PRESENTACION

En la siguiente figura 3.11 se presenta el modelo navegaci3n del sistema.

Figura 3.11 Pantalla p3gina principal



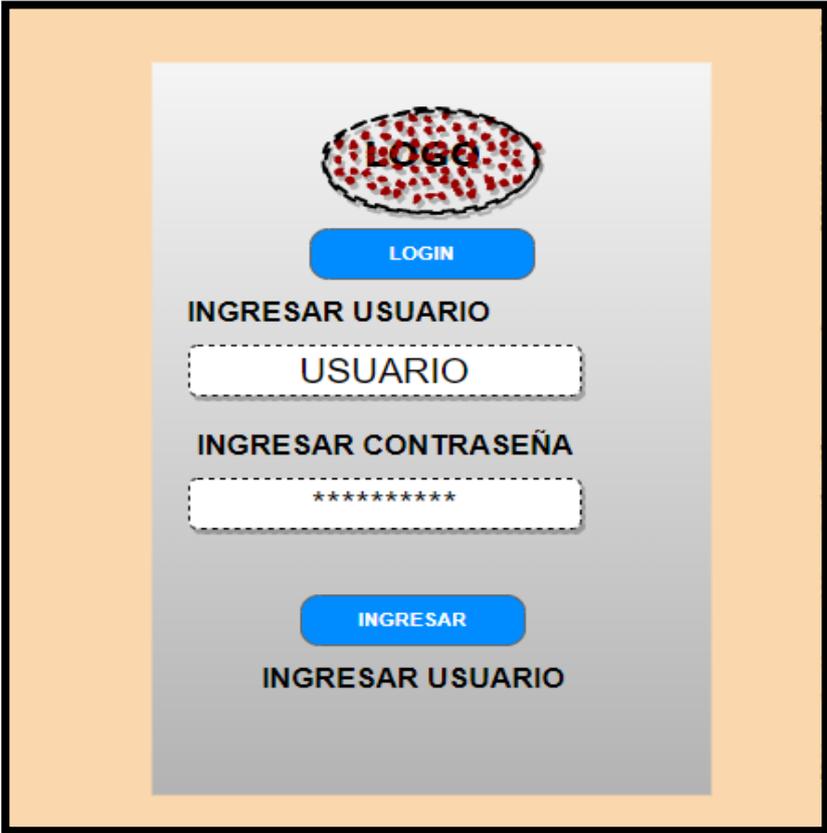
Fuente: (Elaboraci3n propia)

Figura 3.12 Pantalla de presentaci3n de convocatorias



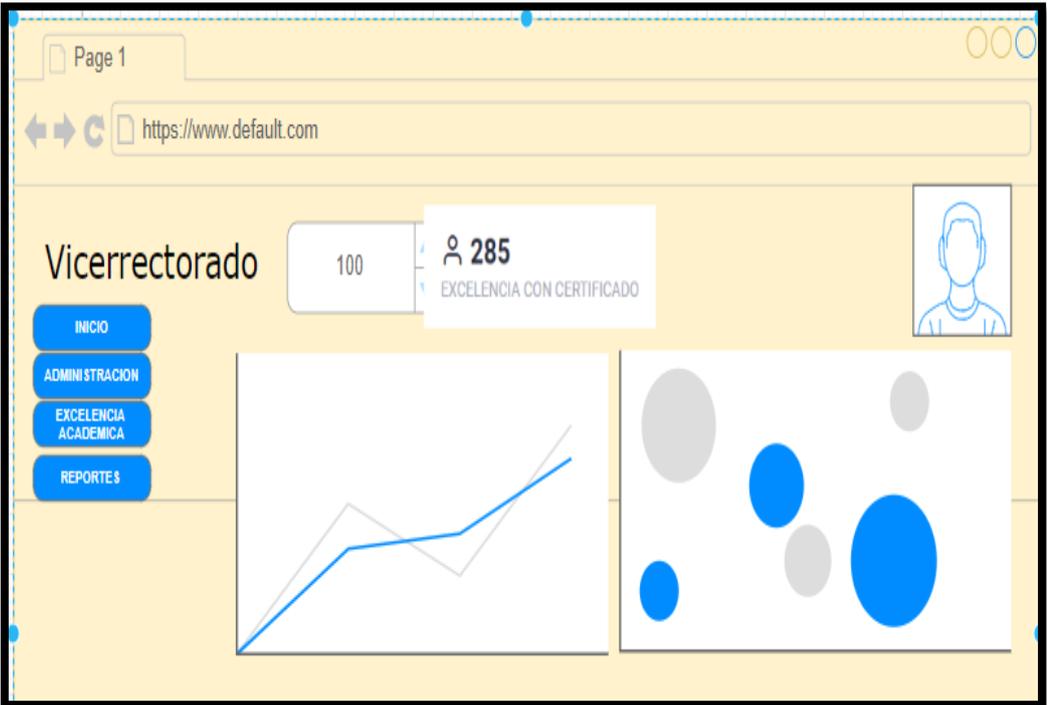
Fuente: (Elaboraci3n propia)

Figura 3.13 Pantalla de control de acceso al sistema



Fuente: (Elaboración propia)

Figura 3.14 Pantalla de presentación del administrador



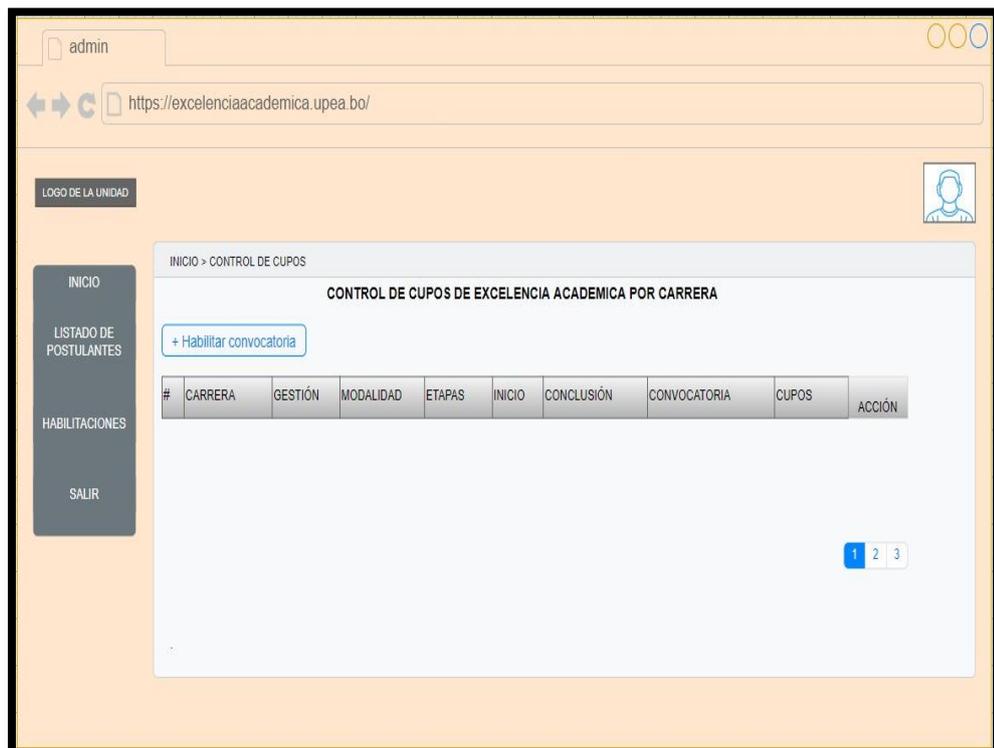
Fuente: (Elaboración propia)

Figura 3.15 Pantalla de presentación técnico excelencia



Fuente: (Elaboración propia)

Figura 3.16 Pantalla de presentación control de cupos



Fuente: (Elaboración propia)

Figura 3.17 Pantalla de presentación estudiante



Fuente: (Elaboración propia)

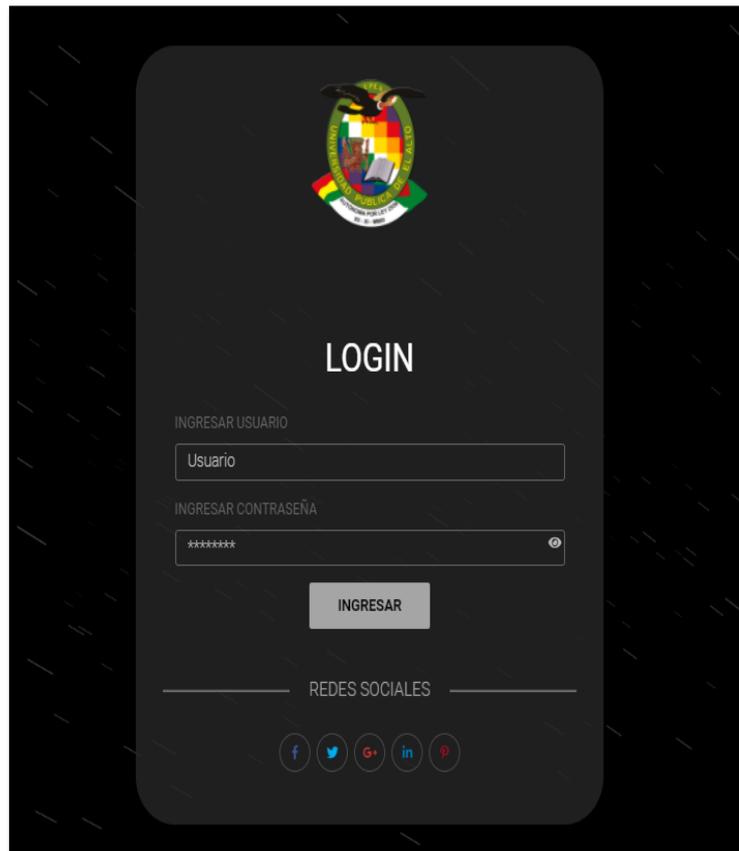
3.8. MODELO DE IMPLEMENTACIÓN

Figura 3.18 Pagina Inicio



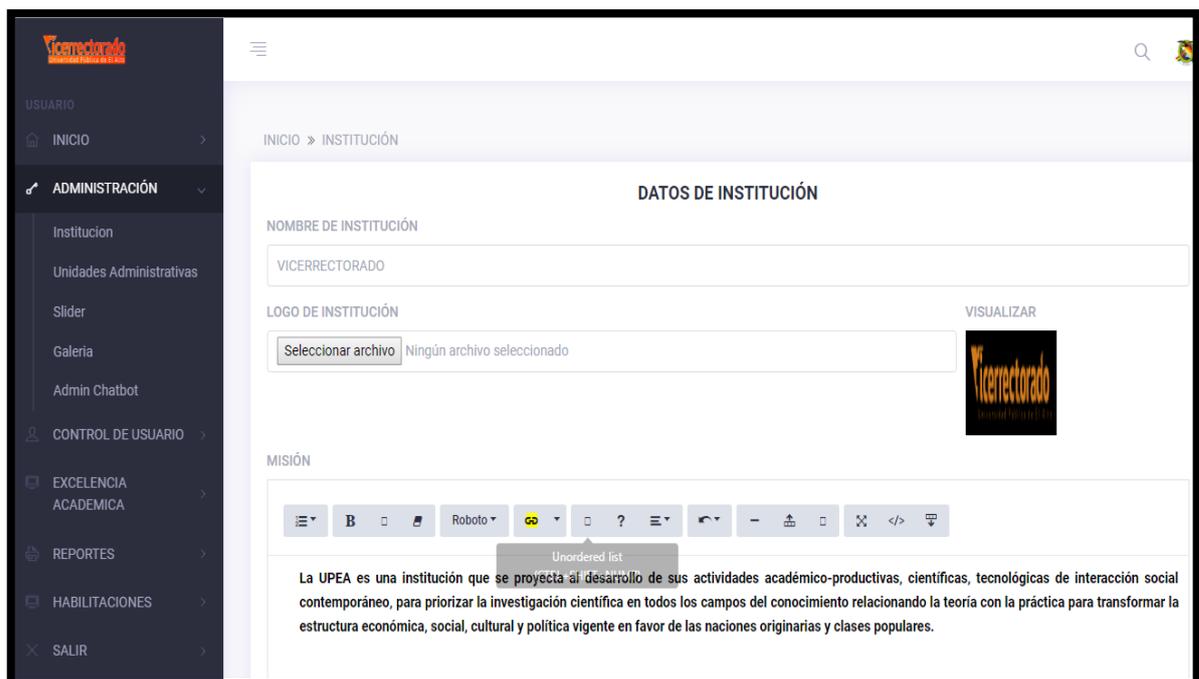
Fuente: (Elaboración propia)

Figura 3.19 Autenticación de Usuarios



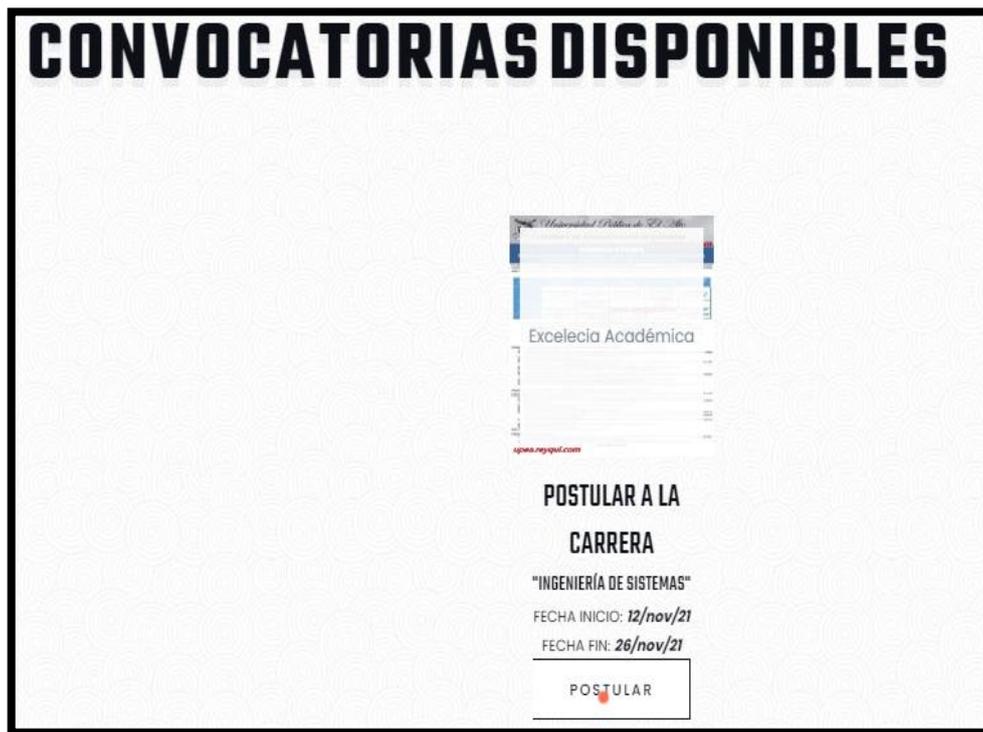
Fuente: (Elaboración propia)

Figura 3.20 Administración de página



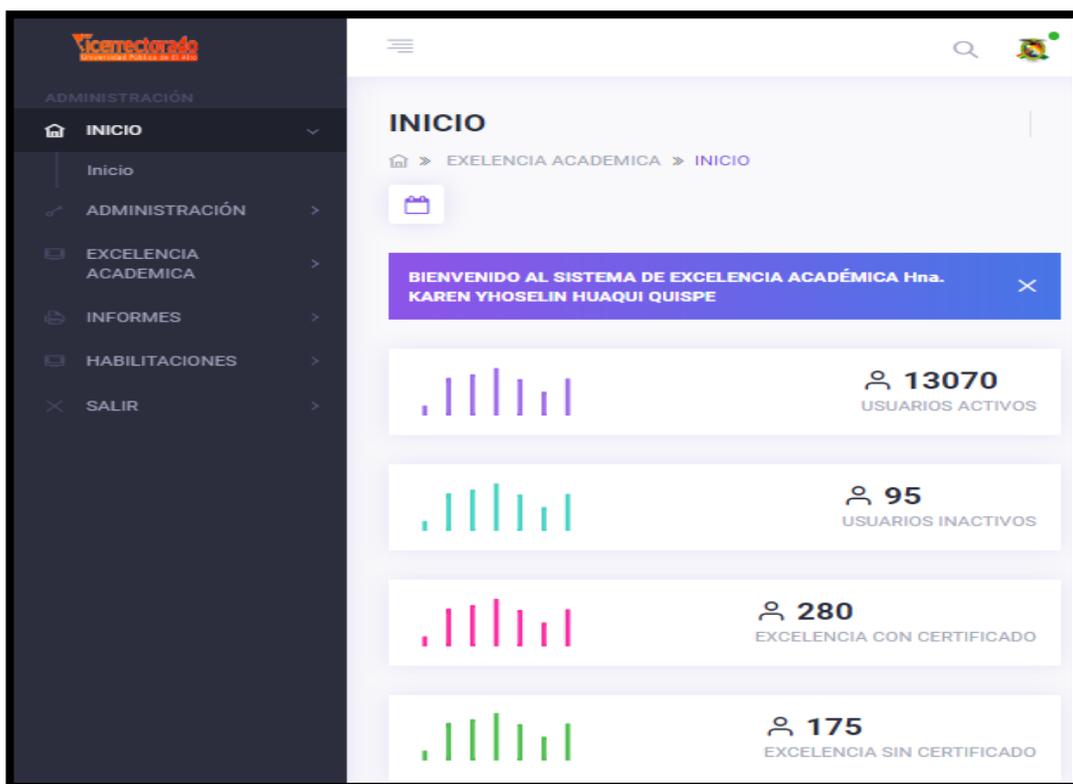
Fuente: (Elaboración propia)

Figura 3.21 Publicación de convocatorias disponibles



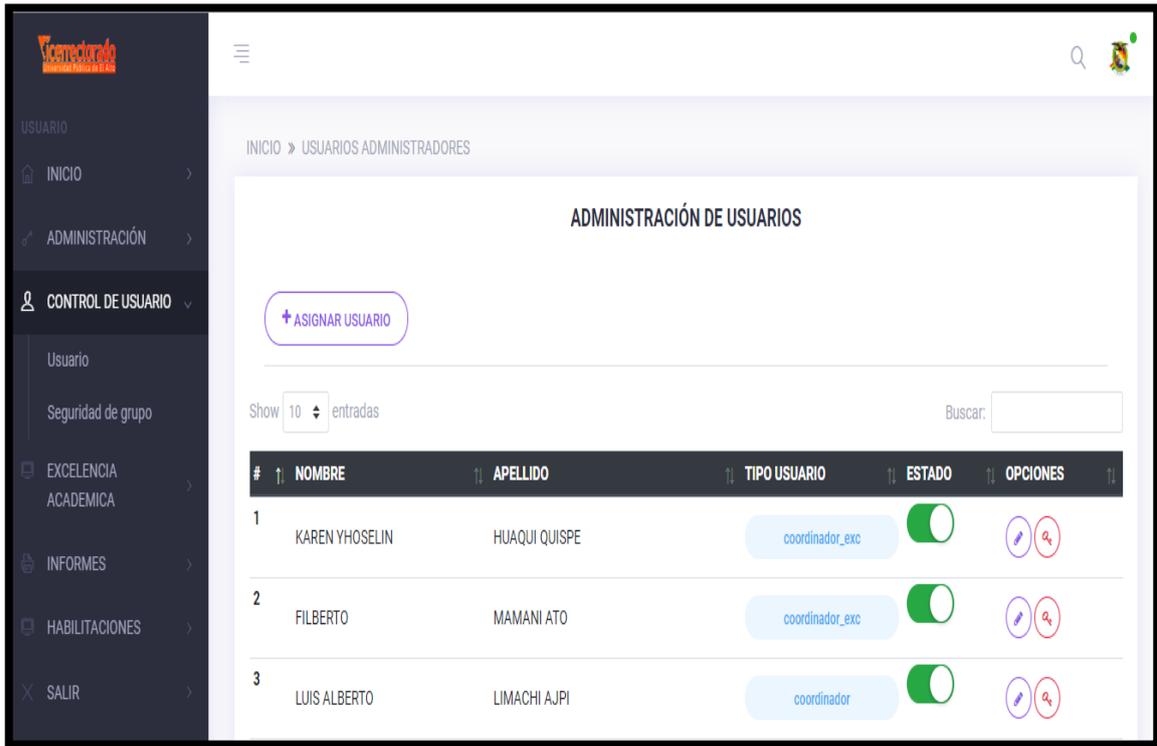
Fuente: (Elaboración propia)

Figura 3.22 Inicio del administrador



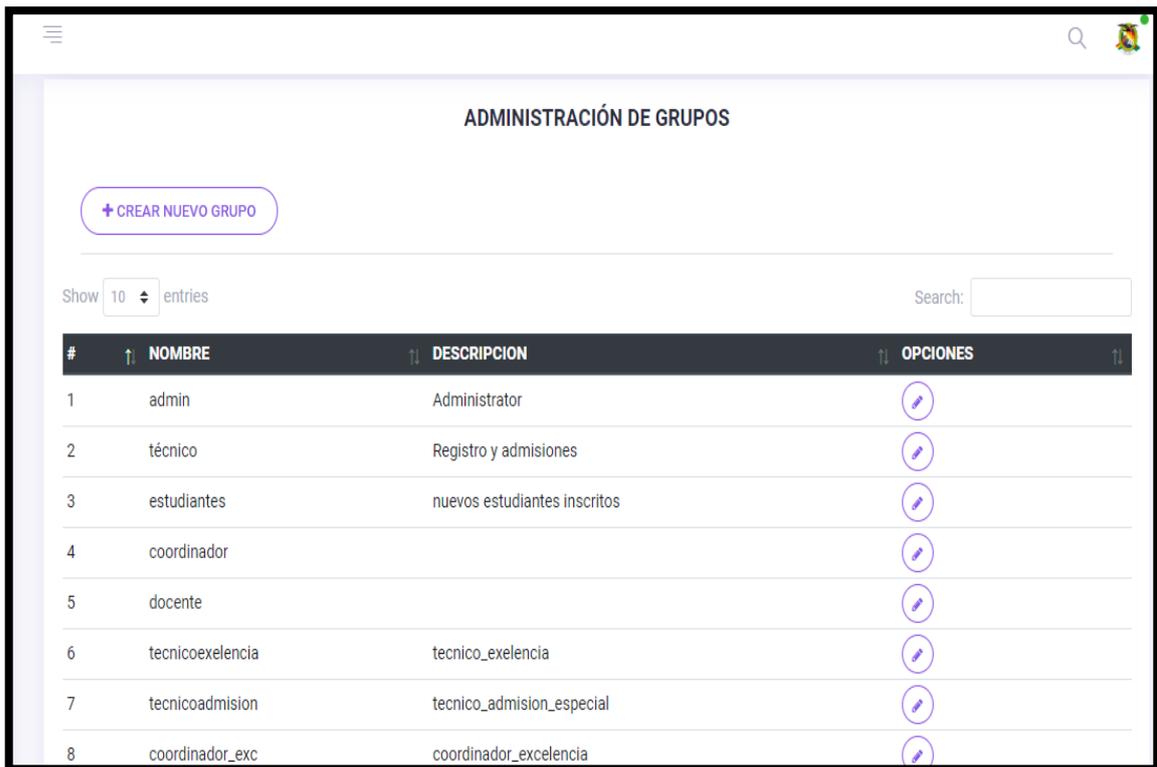
Fuente: (Elaboración propia)

Figura 3.23 Administración de usuarios



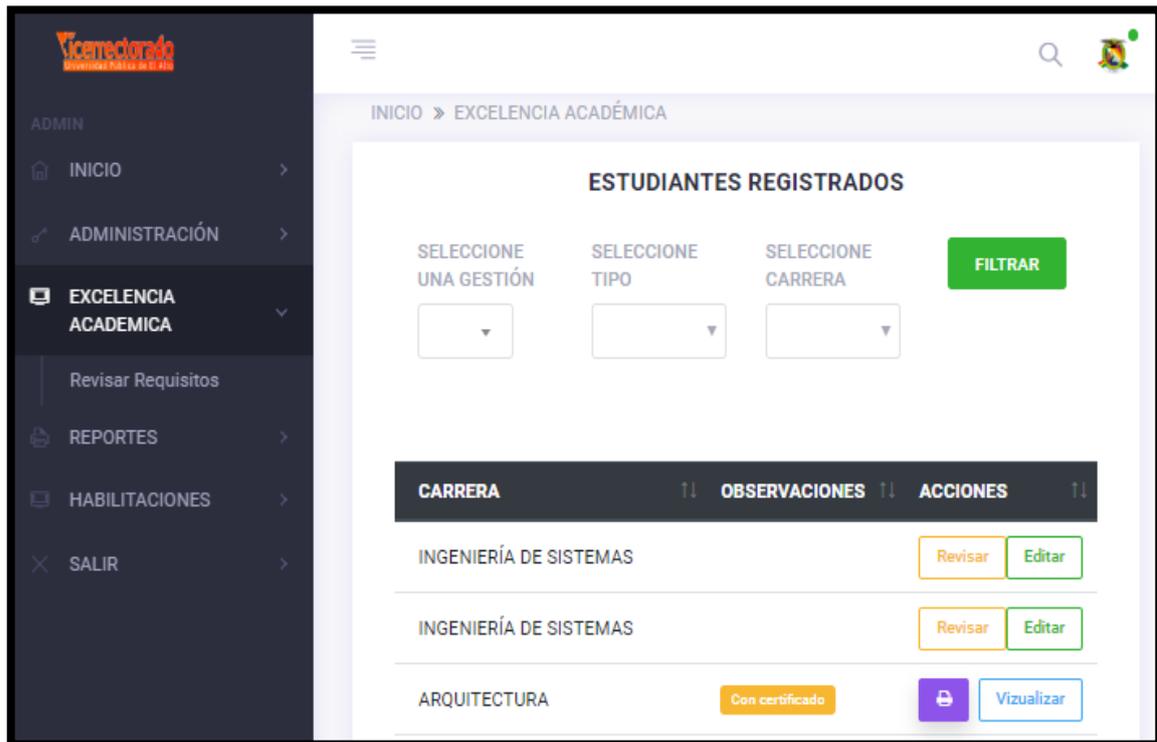
Fuente: (Elaboración propia)

Figura 3.24 Administración de grupos



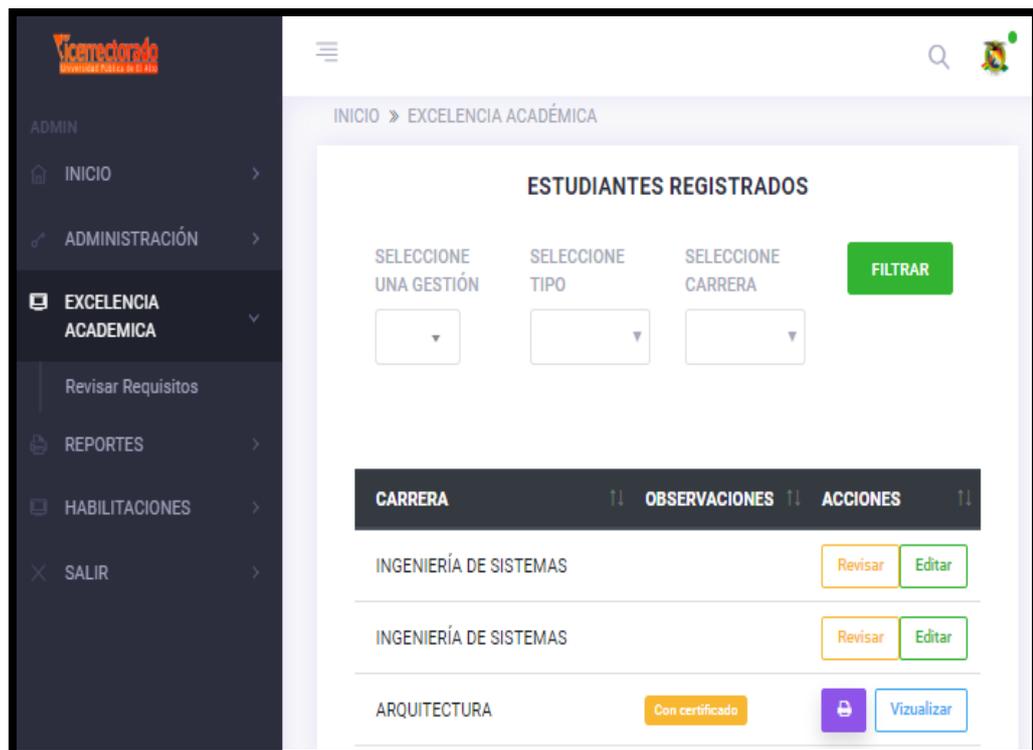
Fuente: (Elaboración propia)

Figura 3.25 Administración técnico excelencia académica



Fuente: (Elaboración propia)

Figura 3.26 Reportes



Fuente: (Elaboración propia)

Figura 3.27 Habilitación de convocatorias

Habilitar fecha

Carrera
- Seleccione carrera -

Fecha inicio
dd/mm/aaaa

Fecha conclusion
dd/mm/aaaa

Fecha de impresion de Certificado de Habilitacion
dd/mm/aaaa

insertar cupos

Adjuntar imagen
Seleccionar archivo Ningún archivo seleccionado

VISUALIZAR

Cancelar Habilitar

Fuente: (Elaboración propia)

Figura 3.28 Control de cupos

INICIO > CONTROL DEL CUPOS

CONTROL DEL CUPOS DE EXCELENCIA ACADÉMICA POR CARRERA

+ Habilitar convocatoria

Show 10 entries Search:

#	CARRERA	GESTIÓN	MODALIDAD	ETAPAS	INCIO	CONCLUSIÓN	FECHA CERTIFICADO DE HABILITACIÓN
1	INGENIERÍA DE SISTEMAS	I/2022-PRE-UNIV	EXCELENCIA ACADEMICA	primera etapa	2021-11-12	2021-11-26	2021-11-26

Fuente: (Elaboración propia)

Figura 3.29 Revisión de documentos

REVISIÓN

POSTULANTE: Sr. ALAIN MAMANI QUISPE

CARRERA: ARQUITECTURA

NOTA: 98.00

#	REQUISITO	ESTADO	ONSERBACIÓN	FECHA RECIVIDA	FECHA REVIS
(1) Carta Vicerrector					
Revisar					

Fuente: (Elaboración propia)

Figura 3.30 Seguimiento de excelencia académica

REVISIÓN

POSTULANTE: Sr. ESTHER QUISPE MAMANI

CARRERA: ARQUITECTURA

NOTA: 98.00

#	REQUISITO	ESTADO	ONSERBACIÓN	FECHA RECIVIDA	FECHA REVISADO	USUARIO REVISOR
1	Carta Vicerrector	Revisado Aprobado		11/nov/21	11/nov/21	Us. KAREN YHOSELIN HUAQUI QUISP
2	Carta Director	Revisado Aprobado		11/nov/21	11/nov/21	Us. KAREN YHOSELIN HUAQUI QUISP
3	Certificado de nacimiento	Revisado Aprobado		11/nov/21	11/nov/21	Us. KAREN YHOSELIN HUAQUI QUISP
4	Título bachiller	Revisado Aprobado		11/nov/21	11/nov/21	Us. KAREN YHOSELIN HUAQUI QUISP

Fuente: (Elaboración propia)

Figura 3.31 Registro de formulario para el postulante

INICIO UNIDADES ADMINISTRATIVAS POSTULARME ACCESO

POSTULARME

SISTEMA DE EXCELENCIA ACADÉMICA

POSTULARME

NÚMERO DE CARNET (Solo número) Complemento

Ingrese documento de identidad complemento

INGRESE FECHA DE NACIMIENTO (Ej: 01-08-2001)

dd/mm/aaaa

INGRESAR PARA EL LLENADO DEL FORMULARIO

CHAT DE CONSULTAS

¡Hola! Mi nombre es Karen y estoy para ayudarte.

Fuente: (Elaboración propia)

Figura 3.32 Formulario de Registro

LLENAR FORMULARIO

Volver a Inicio

CARRERA A LA QUE POSTULA

Fecha de inscripción Carrera Sede

13/11/2021

DATOS PERSONALES

Apellido Paterno Apellido Materno Nombre(s) Nacionalidad

Ingresar apellido paterno Ingresar apellido materno Ingresar nombre(s) Nacionalidad

C.I.(Número) Expedido Tipo de documento Fecha de Nacimiento

2866377

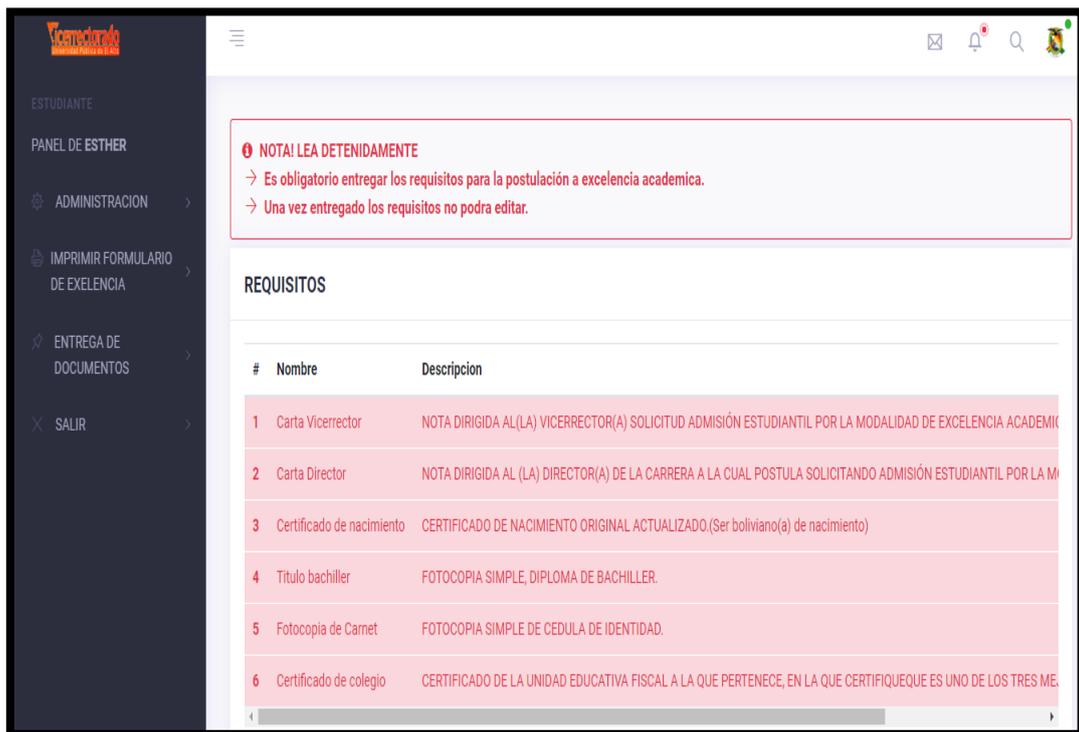
Genero Estado Civil Departamento nacimiento Provincia nacimiento

Lugar nacimiento Telefono/Celular Correo electronico

[MUNICIPIO] telefono/celular Ingrese Correo electronico

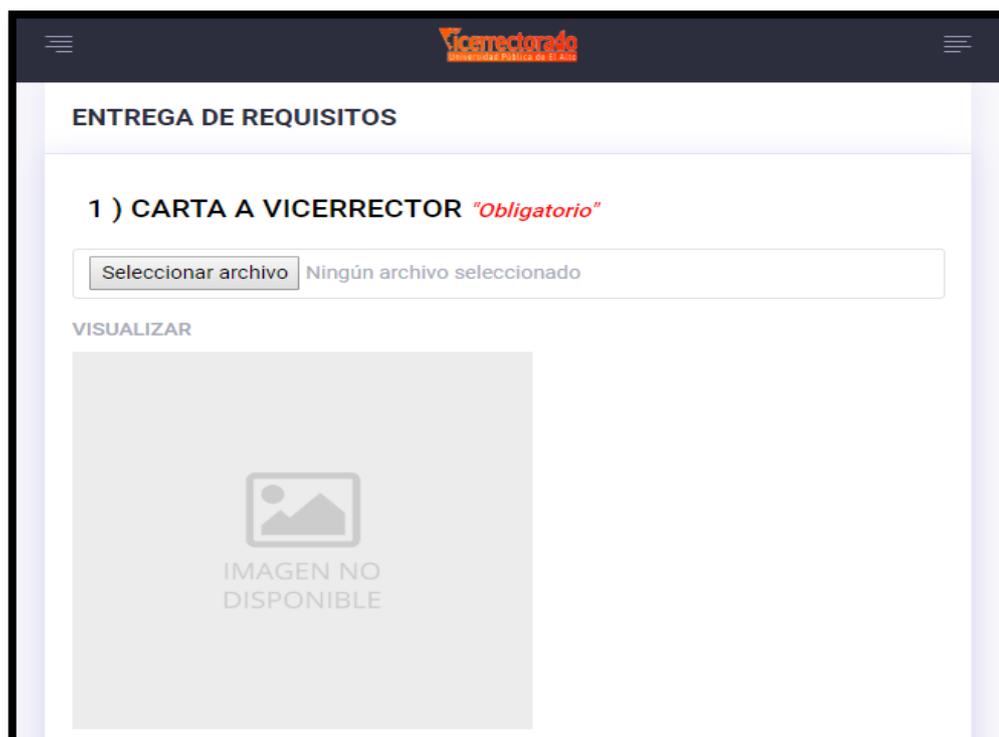
Fuente: (Elaboración propia)

Figura 3.33 Administración del estudiante



Fuente: (Elaboración propia)

Figura 3.34 Entrega de requisitos



Fuente: (Elaboración propia)

3.9. PRUEBAS

3.9.1. Prueba de caja blanca

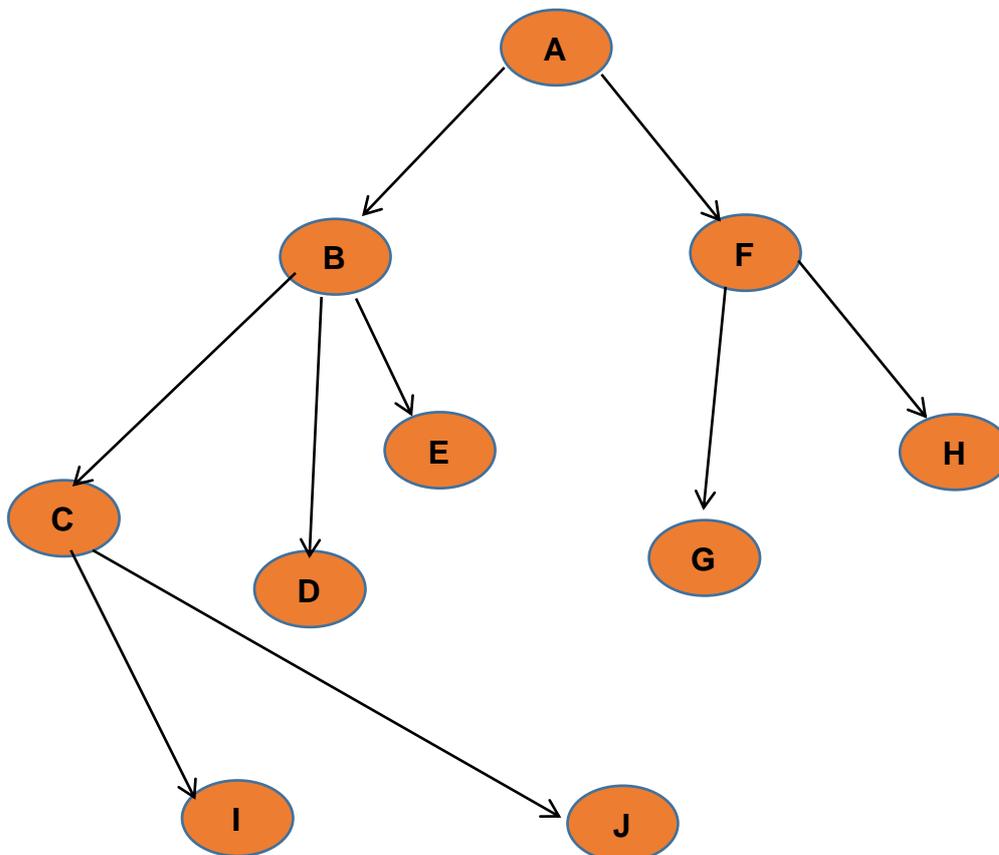
En este tipo de prueba, observamos todo el código con la noción de probar el desenvolvimiento del sistema recorrido por cada uno de los casos presentados por los algoritmos que se utilizaron en la codificación, es decir son casos de prueba que se aplica al código fuente. En las pruebas de caja blanca se consideran lo siguiente:

- Se realizarán las pruebas utilizando el conocimiento del funcionamiento interno del código.
- Las pruebas de caja blanca solo se pueden realizar por programadores.

La aplicación del caso de prueba de caja blanca se lo realiza utilizando métrica de complejidad dicromática el cual brinda la cantidad aproximada de casos de prueba que se deben aplicaren el código fuente como se muestra en la siguiente figura.

De forma general, se debe seguir:

Figura 3.35 Caja Blanca



Fuente: (Elaboración propia)

Donde:

- Inicio del sistema (A)
- Módulo de administración de usuarios (B)
- Creación de usuarios (C)
- Cambiar de estado (D)
- Listado de usuarios (E)
- Módulo de reportes (F)
- Reporte general (G)
- Reporte específico (H)
- Impresión de certificados (I)
- Corroboración de documentos (J)

Después de realizar el gráfico se procede a determinar la complejidad cíclica del grafo mediante la siguiente fórmula:

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

Donde:

A= 8(Aristas)

N= 7(Nodos)

Por tanto:

$$V(G) = 8 - 7 + 2$$

$$V(G) = 3$$

Determinar los caminos linealmente independientes los caminos que deben ser probados.

Camino 1: A – B – C – I

Camino 2: A – B – C – J

Camino 3: A – B – D

Camino 4: A – B – E

Camino 5: A – F – H

Camino 6: A – F – G

Presentar los casos de prueba para forzar ejecución de cada camino. Esta última condición establece que para la ejecución de ciertos caminos.

Camino 1: El usuario podrán ingresar a la página principal revisar los módulos de usuarios, ver los reportes y la impresión de certificados.

Camino 2: El usuario puede ver los reportes y corroborar los documentos.

Camino 3: El usuario ingresa y administra a los usuarios, puede cambiar los estados.

Camino 4: El usuario ingresa y administra a los usuarios, los puede listar.

Camino 5: El usuario ingresa y administra puede generar los reportes generales de la excelencia.

Camino 6: El usuario cuando ingresa y puede generar los reportes específicos.

3.9.2. Prueba 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 3.9 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 3.10 Caso de prueba de registro de usuario

Caso de prueba de registro de usuario	
Código:	Caso de Prueba 02
Descripción de la Prueba	Registro de usuarios
Condiciones de ejecución	El usuario 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. ✓ 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 3.11 Caso de prueba de registro de estudiante

Caso de prueba de registro de estudiante	
Código:	Caso de Prueba 03

Descripción de la Prueba	Registro de postulante mediante sus datos personales.
Condiciones de ejecución	El postulante debe estar en la página y selecciona la opción de llenar el formulario.
Pasos de Ejecución	<ul style="list-style-type: none"> ✓ El sistema muestra una pantalla que le pedirá al postulante su carnet de identidad 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.
Evaluación de prueba	Se realiza el registro del postulante correctamente

Fuente: (Elaboración propia)

**SISTEMA DE REGISTRO DE ADMISIÓN DE ESTUDIANTES
DE LA MODALIDAD DE EXCELENCIA ACADÉMICA
CASO: VICERRECTORADO UPEA**

CAPÍTULO IV

CALIDAD Y SEGURIDAD



4.1. INTRODUCCIÓN

En este capítulo se determina la calidad del sistema el cual es uno de los aspectos más importantes dentro del desarrollo de software del mismo modo de describirá la seguridad del sistema.

Para asegurar que el proyecto se desarrolle se aplicaron las métricas de calidad, donde se calcularon los parámetros que cumple el sistema, que permita especificar ordenadamente dichas características y atributos.

Se pueden destacar varios modelos y criterios de calidad para productos de software, algunos de ellos podemos destacar como el ISO 25000.

4.2. NORMA ISO 25000

4.2.1. Funcionalidad

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

Tabla 4.1 Número de Entradas de usuario

Entradas de Usuario		
1	Administración de usuarios	2
2	Administración de técnicos	2
3	Administración de estudiantes	20
Total		24

Fuente: (Elaboración Propia)

- **Número de salidas de usuario:** El número de salidas que el sistema proporciona al sistema: mensajes, notificaciones, reportes, alertas, etc.

Tabla 4.2 Número de Salidas Usuario

Salidas de Usuario		
1	Administración de usuarios	3
2	Administración de técnicos	3

3	Administración de estudiantes	20
Total		26

Fuente: (Elaboración Propia)

- **Número de peticiones de usuario:** Es el número de entradas interactivas que genera una respuesta del software como salidas interactivas.

Tabla 4.3 Número de Peticiones del Usuario

Peticiones del Usuario		
1	Administración de usuarios	3
2	Administración de técnicos	3
3	Administración de estudiantes	6
Total		12

Fuente: (Elaboración Propia)

- **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.

Tabla 4.4 Parámetros de medición de punto función

Parámetros de medición	Cuenta
Número de entradas de Usuario	24
Número de salidas de usuario	26
Número de peticiones de usuario	12
Número de archivos	20
Número de interfaz externa	2

Fuente: (Elaboración propia)

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

Tabla 4.5 Punto función sin ajustar

Parámetro de medición	Cuenta	Factor	Total
Número de entradas de Usuario	24	6	144
Número de salidas de usuario	26	7	182
Número de peticiones de usuario	12	6	72
Número de archivos	20	15	300
Número de interfaz externa	2	10	20
Cuenta total			718

Fuente: (Elaboración propia)

En la tabla 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 4.6 Factores de Complejidad

Importancia	0%	20%	40%	60%	80%	100%	Fi
Escala	No influencia	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
¿Existe funciones de procesos distribuidos?						X	5
¿Es crítico el rendimiento?					X		4
¿El sistema web será ejecutado el SO. Actual?						X	5

¿Se requiere una entrada interactiva para el sistema?					X		4
¿Se requiere que el sistema tenga entradas a datos con múltiples ventanas?					X		4
¿Se actualiza los archivos de forma interactiva?					X		4
¿Son complejas las entradas, salidas, los archivos o las peticiones?						X	5
¿Es complejo el procesamiento interno del sistema?					X		4
¿Se ha diseñado el código para ser reutilizado?						X	5
¿Se ha diseñado el sistema para facilitar al usuario el trabajo y ayudarlos a encontrar la información?					X		4
TOTAL							54

Fuente: (Elaboración propia)

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

$$PF = \text{Conteo total} [0.65 + 0.01 * \sum fi]$$

Donde:

$$\sum (fi)$$

- Sumatoria de los valores de los factores de ajuste

Se tiene:

$$\sum(fi) = 54$$

$$PF = 718 * [0.65 + 0.01 * 53]$$

$$PF = 854.42$$

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

$$PF_{max} = \text{Cuenta Total} [0.65 + (0.01 * \sum fi)]$$

$$PF_{max} = 718 * [0.65 + (0.01 * 70)]$$

$$PF_{max} = 969.3$$

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

$$Funcionalidad = \frac{PF}{PF_{max}} * 100$$

$$Funcionalidad = \frac{854.42}{969.3}$$

$$Funcionalidad = 0.8815 * 100\%$$

$$\mathbf{Funcionalidad = 88 \%}$$

Por lo tanto, la funcionalidad del sistema desarrollado llega a ser del 88.15 %, esto quiere decir que el sistema tiende a funcionar un 88.15 %, sin riesgo de fallar, y un 11.85 % de que el sistema colapse.

4.2.2. 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:

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

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

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

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

Donde:

- f = funcionalidad del sistema
- μ = Probabilidad de error del sistema
- t = tiempo de prueba del sistema

En un periodo de 20 días como tiempo de prueba se define de cada 10 ejecuciones 1 falla

$$F(t) = 0.88 * e^{-\frac{1}{10} * 20}$$

$$F(t) = 0.1191 * 100\%$$

$$F(t) = 11.91 \%$$

Reemplazando:

$$P(T \leq t) = 0.1191 = 11.91\%$$

$$P(T \leq t) = 1 - 0.1191$$

$$P(T \leq t) = 0.8809 * 100\%$$

$$P(T \leq t) = 88 \%$$

Por lo tanto, la confiabilidad del sistema desarrollado es del 88.09%. en un periodo de 20 días como el tiempo de prueba.

4.2.3. 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:

Formula:

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

Donde:

- x_i = Valores de evaluación de cada pregunta
- n = Numero de preguntas

Se toma en cuenta la siguiente tabla:

Tabla 4.7 Parámetros de medición de usabilidad

Escala	Valor
Muy bueno	5
Bueno	4
Regular	3
Malo	2
Pésimo	1

Fuente: (Elaboración propia)

Tabla 4.8 Usabilidad del sistema

Nº	Pregunta	Si	No	Resultado xi
1	¿Puede Utilizar con facilidad el sistema?	9	1	0.9
2	¿Puede Controlar operaciones que el sistema solicita?	8	2	0.8
3	¿Las Respuestas del sistema son complicadas?	2	8	0.8
4	¿El Sistema permitió la retroalimentación de información?	9	1	0.9
5	¿El sistema cuenta con interface agradable a la vista?	9	1	0.9
6	¿La respuesta del sistema es satisfactoria?	9	1	0.9
7	¿Le parece complicada las funciones del sistema?	8	2	0.8
8	¿Se hace difícil o dificultoso aprender a manejar el sistema?	1	9	0.9
9	¿Los resultados que proporciona el sistema facilitan el trabajo?	10	0	1
10	¿Durante el uso del sistema se produjo errores?	2	8	0.8
TOTAL				8.7

Fuente: (Elaboración propia)

Con los resultados obtenidos del cuestionario obtenemos calculamos la usabilidad:

$$FU = \left[\frac{8.7}{10} * 100 \right]$$

$$FU = [0.87 * 100]$$

$$FU = 87\%$$

Con esto concluimos que la usabilidad del sistema es del 87%

4.2.4. 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 4.9 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 4.10 Valoración de la eficiencia del sistema

Nº	PREGUNTA	PORCENTAJE
1	¿La distribución y estilo de la interfaz permite que un usuario introduzca con eficiencia las operaciones y la información?	5
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?	4
4	¿Las operaciones jerárquicas están organizadas de manera que minimizan la navegación del usuario para hacer que alguna se ejecute?	4

5	¿Procesa y responde adecuadamente cuando realiza alguna consulta o búsqueda?	5
Total		22

Fuente: (Elaboración propia)

Para calcular la eficiencia tenemos la siguiente formula:

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

Donde:

- $\sum x_i$ = Sumatoria de los valores de eficiencia
- n = Numero de preguntas

Reemplazando se tiene lo siguiente:

$$E = \frac{22}{5} * \frac{100}{5}$$

$$E = 88 \%$$

Con esto concluimos que la eficiencia del sistema es del 88%

4.2.5. 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:

- **Mt:** Numero de módulos total de la versión actual
- **Fa:** Numero de módulos de la versión actual que se añadieron.
- **Fc:** Numero de módulos de la versión actual que se cambiaron.
- **Fd:** Numero de módulos de la versión anterior que se eliminaron en la versión actual

Tabla 4.11 Valores para la mantenibilidad

Descripción	Valor
Mt	5
Fa	0
Fc	1
Fd	0

Fuente: (Elaboración propia)

Reemplazando en la ecuación se tiene:

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

$$IMS = 0.8 * 100\%$$

$$IMS = 80\%$$

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

4.2.6. 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 MySQL.

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.

Para obtener la portabilidad, se tiene la siguiente formula:

$$Portabilidad = 1 - \left(\frac{\text{numero de dia para portar el sistema}}{\text{numero de dias para implementar el sistema}} \right)$$

Reemplazando la formula se obtiene el siguiente resultado:

$$Portabilidad = 1 - \left(\frac{1}{7} \right)$$

$$Portabilidad = 0.85 * 100\%$$

$$Portabilidad = 86\%$$

Resultados:

De acuerdo a los resultados obtenidos se puede establecer la calidad total del sistema en base a los parámetros medidos anteriormente. La calidad está directamente relacionada con el grado de satisfacción con el usuario que ingresa al sistema.

Tabla 4.12 Resultados

Características	Resultados
Funcionalidad	88 %
Confiabilidad	88 %
Usabilidad	87%
Eficiencia	88%
Mantenibilidad	80%
Portabilidad	86%
Evaluación total	86 %

Fuente: (Elaboración propia)

Evaluación de calidad total del sistema es de un 86 %.

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 27001 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

Para la seguridad del sistema se consideran las siguientes precauciones

- **Copias de Seguridad:** Las copias de seguridad son respaldos que se tiene de las bases de datos, estos se realizan de manera automática.
- **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.

También la implementación del código captcha.

- **Encriptación:** El sistema permite encriptar las rutas y las contraseñas de los usuarios con hash.
- **Base de datos:** 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.

La información en una institución 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 con seguridad de acceso.

- **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.

**SISTEMA DE REGISTRO DE ADMISIÓN DE ESTUDIANTES
DE LA MODALIDAD DE EXCELENCIA ACADÉMICA
CASO: VICERRECTORADO UPEA**

CAPÍTULO V

ANÁLISIS DE COSTO



5.1. INTRODUCCIÓN

Para realizar el presente proyecto es necesario planificar y estimar los costos durante y hasta la finalización del mismo. Es útil para estimar el costo total del sistema de registro se tomarán en cuenta los siguientes costos:

5.2. ANÁLISIS DE COSTOS

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:

- **ORGANICO:** Proyectos relativamente sencillos, menores a 5000 líneas de código, implica procesamiento de datos, uso de la base de datos se focaliza en transacciones y recuperación de datos.
- **SEMIACOPLADO:** Proyectos intermedios en complejidad y tamaño. La experiencia en este tipo de proyectos es variable y las restricciones intermedias.
- **EMPOTRADO:** Proyectos bastantes complejos, en los que apenas se tiene experiencia y en un entorno de gran innovación técnica. Para la estimación de costos de este proyecto se empleará el submodelo post arquitectura.

Para calcular el esfuerzo, necesitaremos hallar la variable KLDC (kilo-líneas de código). Este proyecto se implementa 8504 Líneas de Código en el lenguaje PHP. Aplicando Conversiones se tiene.

LCD = 8504

$$KLCD = \frac{LCD}{1000}$$

$$KLCD = \frac{8504}{1000}$$

$$KLCD = 8,504$$

Es un modelo Intermedio y semi acoplado.

Tabla 5.1 Coeficientes de a,b,c,d de COCOMO II

Proyecto Software	A	b	c	d
Orgánico	3.2	1.05	2.5	0.38
Semi acoplado	3.0	1.12	2.5	0.35
Empotrado	2.8	1.20	2.5	0.32

Fuente: (COCOMO, 2013)

Debido a que el proyecto no supera las 50000 líneas de código se utilizarán los coeficientes para proyectos semi-acoplado. Ecuaciones para calcular el costo de software:

Tabla 5.2 Ecuaciones de Cocomo II

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

Fuente: (COCOMO, 2013)

Para hallar los valores de FAE, se utilizará la tabla de atributos multiplicadores.

Tabla 5.3 Calculo de 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)

Por tanto, nuestro Factor de ajuste será

$$FAE = 1.15 * 1.00 * 1.15 * 1.00 * 1.00 * 0.87 * 1.00 * 0.86 * 0.82 * 0.86 * 0.90 * 0.95 * 0.82 * 0.91 * 1.00 = 0.4452$$

Aplicando y reemplazando valores a la fórmula de esfuerzo, se tiene:

$$E = a * KLCD^b * FAE(\text{persona/mes})$$

$$E = 3 * 8,504^{1.05} * 0.4452 (\text{persona/mes})$$

$$E = 12.69 (\text{persona/mes})$$

Cálculo del Tiempo

$$T = c * Esfuerzo^d(\text{meses})$$

$$T = 2.5 * 12.69^{0.35}(\text{meses})$$

$$T = 6.08 (\text{meses}) \text{ esto equivale 6 meses}$$

Cálculo del personal requerido

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

$$NP = \frac{12.69}{6.08} \text{ personas}$$

$$NP = 2.08 \cong 2 \text{ persona}$$

Calculo de Costo Total

$$CT = SueldoMes * NP * T$$

$$CT = 350 * 2 * 6$$

$$CT = 4200 \$us$$

En resumen, se requiere 2 personas estimando un trabajo de 6 meses y con costo total de 4200 \$us Equivalente en bolivianos a 29274.5 Bs.

**SISTEMA DE REGISTRO DE ADMISIÓN DE ESTUDIANTES
DE LA MODALIDAD DE EXCELENCIA ACADÉMICA
CASO: VICERRECTORADO UPEA**

CAPÍTULO VI

*CONCLUSIÓN Y
RECOMENDACIÓN*



6.1. CONCLUSIONES

Se concluyó con los objetivos planteados en el presente proyecto, desarrollando un Sistema de Registro Automatizado que colabore a la unidad de vicerrectorado; en el manejo de datos de manera centralizada del estamento estudiantil.

- ✓ Con la implementación del sistema de registro de estudiantes de excelencia académica se facilita la gestión de los postulantes, logrando de esta manera tener un mejor control del proceso admisión a las carreras de la Universidad Pública de El Alto.
- ✓ Se cumplió con la verificación de los estudiantes que no estén en la inscritos en la universidad.
- ✓ Se realizó el análisis del proceso actual conociendo el flujo de la documentación.
- ✓ Se cumplió con la generación del formulario de validación de registro y la resolución vicerrectoral.
- ✓ El sistema de registro permite tener un mayor control en la admisión de los estudiantes de excelencia académica, generando reportes por carreras.
- ✓ Se aplicó con éxito la norma de calidad, metodología UWE, el criterio de Modelo Vista Controlador y las herramientas de programación para que tenga funcionalidad el sistema.
- ✓ Se elaboró documentación necesaria para el uso y administración del sistema

6.2. RECOMENDACIONES

El personal de la Unidad de Vicerrectorado quien se encarga de administrar el sistema una vez concluido el proyecto de grado se recomienda lo siguiente.

- ✓ Realizar el mantenimiento del sistema en un determinado tiempo de (3 meses) para ver el buen funcionamiento, en el almacenamiento de información y procesos concurrentes como consultas, registro, etc
- ✓ Debido a las características del sistema se recomienda llevar a cabo la capacitación correspondiente.
- ✓ En cuanto al análisis y diseño del sistema, cuando la Unidad requiera incluir un módulo nuevo, se recomienda primero revisar el documento para poder tomar una buena decisión, ya que el diseño del sistema representa

componentes reutilizables que pueden ayudar a reutilizar el diseño del sistema actual.

- ✓ Implementar un módulo de firma digital con la finalidad de darle un valor legal al documento digital.
- ✓ Se recomienda realizar convenios con el ministerio de educación para realizar el proceso de registro más simplificado.

BIBLIOGRAFÍA

- acensTechnologies*. (2016). Obtenido de <https://www.acens.com/wp-content/images/2016/10/bootstrap-framework-acens-wp.pdf>
- Adriana Gomez, M. d. (2011). Un modelo de estimacion de Proyectos de Software.
- Alegsa, L. (18 de julio de 2016). Obtenido de <https://www.alegsa.com.ar/Dic/informacion.php>
- Alegsa, L. (13 de junio de 2018). Obtenido de <https://www.alegsa.com.ar/>
- Aliaga, J. M. (20 de noviembre de 2020). *Objectwave*. Obtenido de <https://objectwave.com>
- Alicante. (18 de octubre de 2021). *Servicio de informatica*. Obtenido de <https://si.ua.es/es/documentacion/asp-net-mvc-3/1-dia/modelo-vista-controlador-mvc.html>
- Anahuac, S. j. (4 de julio de 2018). *¿Qué es la excelencia académica y cómo la potenciamos?* Obtenido de <https://www.anahuac.mx/blog/que-es-la-excelencia-academica-y-como-la-potenciamos>
- Anonimo. (2011). *Capacitacion y guia para el desarrollo de Software*. Obtenido de <http://materias.fi.uba.ar/7548/PruebasSoftware.pdf>
- Aranquren, J. (14 de Marzo de 2018). *Qué es lo académico según Josef Pieper*.
- Arimetrics*. (12 de agosto de 2021). Obtenido de <https://www.arimetrics.com/glosario-digital/javascript>
- Barrera, A. (2021). *next U*. Obtenido de <https://www.nextu.com/blog/que-es-json/>
- Bembibre, V. (10 de abril de 2015). *Definicion ABC*. Obtenido de URL: <https://www.definicionabc.com/tecnologia/registro-2.php>
- Bustos, G. (29 de abril de 2021). *Hostinger Tutoriales*. Obtenido de *¿Qué es Apache? Descripción completa del servidor web Apache:* <https://www.hostinger.es/tutoriales/que-es-apache/>
- Calle, J. Q. (2017). *SISTEMA DE INFORMACION ACADEMICA VIA WEB(SINA)*. La Paz.
- Cardenas, O. (26 de mayo de 2016). *google sites*. Obtenido de Ingenieria WEB: <https://sites.google.com/site/ingenieriaweb23/home>
- Carrion, J. (2017). *UDG VIRTUAL*. Obtenido de <http://iibi.unam.mx/voutssasmt/documentos/dato%20informacion%20conocimiento.pdf>
- Chavez. (2012). *Ingenieria Web*. 30-50.
- COCOMO. (28 de enero de 2013). Obtenido de <https://es.wikipedia.org/wiki/COCOMO>

concepto. (5 de agosto de 2021). Obtenido de <https://concepto.de/sistema-de-informacion/>

Desarrollo web. (28 de julio de 2020). Obtenido de <https://desarrolloweb.com/articulos/que-es-mvc.html>

EcuRed. (s.f.). Obtenido de <https://www.ecured.cu/MagicDraw>

Estrada, J. G. (03 de agosto de 2021). *Desarrollo Web con PHP y MySQL*. Obtenido de <https://cupdf.com/document/desarrollo-web-con-php-y-mysql-52.html>

Flores, M. (8 de enero de 2018). *Mi carrera universitaria*. Obtenido de <https://micarrerauniversitaria.com/c-ingenieria/ingenieria-de-software/>

Fuentes, M. (28 de mayo de 2018). *Cnachile*.

Gardey, J. P. (16 de 05 de 2021). *Definicion.de*. Obtenido de <https://definicion.de/web-2-0/>

Haide, P. P. (2016). *Desarrollo e implementacion de un sistema web de admision y matriculacion para el Instituto de Postgrado de la Universidad Estatal Peninsula de Santa Elena*. Santa Elena.

Hernandez, P. A. (2016). *Sistema de Informacion para la gestion de Proyectos*. Bogota.

Hiamix. (13 de marzo de 2013). *slideshare*. Obtenido de <https://es.slideshare.net/HIAMIX/estructura-general-de-un-sistema-17116816>

Informatica. (5 de Agosto de 2021). *Dato en informática*. Editorial Etecé.

Instituto de Tecnologia de comumbia Britanica. (19 de septiembre de 2019). Obtenido de <https://codeigniter.com/userguide3/general/welcome.html>

IONOS. (16 de marzo de 2020). Obtenido de <https://www.ionos.es/digitalguide/paginas-web/desarrollo-web/codeigniter-framework-php-rapido-y-versatil/>

ISO 25000. (2021). Obtenido de <https://iso25000.com>

ISO, E. (2021). Obtenido de <https://danieltrabajoiso25000.blogspot.com/2020/11/ventajas-y-desventajas-norma-iso-25-000.html>

Llopart, C. (14 de Noviembre de 2016). *Novi Cap*. Obtenido de [Cómo iniciar un negocio: 7 hábitos que llevarán a los emprendedores a la excelencia: https://novicap.com/blog/como-llegar-excelencia-habitos/](https://novicap.com/blog/como-llegar-excelencia-habitos/)

Lopez, J. (2010). *Ingenieria WEB. Ingenieria de software*.

Lozada, A. (18 de agosto de 2020). *horizonte*. Obtenido de <https://soyhorizonte.com/blog/ventajas-y-desventajas-de-usar-bootstrap/>

Luz, M. (13 de Febrero de 2021). *Dudalia*. Obtenido de <https://dudalia.com/tarea/qmlr>

magicdraw. (31 de Octubre de 2021). Obtenido de <https://www.magicdraw.com/main.php>

Magicdraw. (31 de octubre de 2021). Obtenido de https://www.magicdraw.com/main.php?ts=download_demo&cmd_go_to_login=1&menu=download_demo&back_cmd=cmd_show

McFarland. (29 de agosto de 2014). Obtenido de <https://www.um.es/docencia/barzana/DAWEB/Lenguaje-de-programacion-JavaScript-1.pdf>

Mentor, I. S. (5 de agosto de 2021). *Aula mentor*. Obtenido de http://descargas.pntic.mec.es/mentor/visitas/demoSeguridadInformatica/isoirc_27001_pdca.html

Moderno, T. (15 de Marzo de 2020). *Tester Moderno*. Obtenido de <https://www.testermoderno.com/caja-blanca-vs-caja-negra/>

Mora, J. G. (2016). La necesidad cambio educativo para la sociedad del conocimiento. *Revistas Ibero Americana*, 35.

MySql. (23 de septiembre de 2021). Obtenido de <https://dev.mysql.com/doc/>

MySql.com. (13 de abril de 2021). Obtenido de <https://www.mysql.com/>

Parada, M. (31 de octubre de 2019). *OpenWebinars*. Obtenido de <https://openwebinars.net/blog/que-es-jquery/>

Paz, P. A. (20 de Marzo de 2021). *SCRIBD*. Obtenido de <https://es.scribd.com/document/405299748/Practical-Codeigniter-3-Espanol-Capitulo-1>

Peño, J. M. (2015). *Pruebas de Software , fundamentos y Tecnicas*. Madrid: sin editorial.

php. (6 de octubre de 2021). Obtenido de <https://www.php.net/manual/es/intro-what-is.php>

Pinto, J. M. (2016). *“SISTEMA WEB COLABORATIVO INTEGRADO DE CONTROL,ADMINISTRACION Y SEGUIMIENTO BASADO EN GESTION DE PROCESOS DE NEGOCIO*. LA PAZ: UMSA.

Pinzon, O. (16 de septiembre de 2017). *Ingenieria Web: Una Metodologia para el Desarrollo de Aplicaciones Web Escalables y Sostenibles*. Obtenido de http://www.laccei.org/LACCEI2017-BocaRaton/student_Papers/SP277.pdf

Platzi. (9 de septiembre de 2018). Obtenido de <https://platzi.com/blog/pruebas-esenciales-para-evaluar-el-rendimiento-de-software/>

Porto, J. P., & Gardey, A. (2019). Obtenido de <https://definicion.de/json/>

Porto, J. P., & Gardey, A. (5 de noviembre de 2019). *Definicion*. Obtenido de <https://definicion.de/jquery/>

- Reyqui. (30 de Abril de 2016). *upea.reyqui.com*. Obtenido de <https://upea.reyqui.com>
- Reyqui. (4 de enero de 2017). Obtenido de <https://upea.reyqui.com/2017/01/conoce-las-4-formas-por-las-que-puedes.html>
- Rovira, H. (27 de julio de 2020). *blog de marketing y SEO*. Obtenido de <https://seocom.agency/es/blog/bootstrap-4-vale-la-pena-actualizar/>
- Salguera, H. (2012). Metodología UWE.
- Salinas, D. V. (2018). Gestión Académica y Desempeño Docente, según los estudiantes de una universidad privada en Lima, Perú. *Industrial Data*, 40-58.
- Sanchez, G. (7 de febrero de 2017). *slideshare*. Obtenido de <https://es.slideshare.net/GermnSnchezDomnguez/metodologia-uwe-uml-based-web-engineering>
- Sanz, D. (9 de septiembre de 2018). *Platzi*. Obtenido de <https://platzi.com/blog/pruebas-esenciales-para-evaluar-el-rendimiento-de-software/>
- Significados*. (16 de marzo de 2020). Obtenido de <https://www.significados.com/sistema/>
- Significados*. (27 de mayo de 2020). Obtenido de <https://www.significados.com/jquery/>
- Siles, W. E. (2013). *SISTEMA DE INFORMACIÓN AUTOMATIZADO PARA EL CONTROL DE LA GESTIÓN Y ADMINISTRACIÓN DE BIENES Y SERVICIOS*. El Alto: UPEA.
- SlideShare. (06 de junio de 2015). *Metodología UWE*. Obtenido de <https://es.slideshare.net/omar1023/uwe-49070035>
- Thewolf, D. (25 de junio de 2015). *Metodología UWE*. Obtenido de <https://metodologiauwe.wordpress.com/>
- UWE, L. -L.-M.-U. (2016). *UWE – UML-based Web Engineering*. Obtenido de LMU – Ludwig-Maximilians-Universität München: <https://uwe.pst.ifi.lmu.de/teachingTutorialRequirementsSpanish.html>
- Vargas, A. M. (2019). Ingeniería de Software I. 1-20.
- Vargas, M. S. (25 de junio de 2017). *Ingeniería basada en modelos*. Obtenido de <http://marcelosalasvargas.blogspot.com/2017>
- vargas, m. s. (25 de junio de 2017). *ingeniería guiada por modelos*. Obtenido de <http://marcelosalasvargas.blogspot.com/2017>
- Vargas, s. m. (25 de junio de 2017). *Ingeniería Guiada por Modelos*. Obtenido de <http://marcelosalasvargas.blogspot.com/2017>
- Vega-Pérez, C. A., Grajales-Lombana, H. A., & Montoya Restrepo, L. A. (2017). Sistemas de información: definiciones, usos y limitantes. En C. A. Vega-Pérez,

H. A. Grajales-Lombana, & L. A. Montoya Restrepo, *Sistema de Información Científica* (págs. 64-72). Villavicencia - Colombia: Orinoquia.

Westreicher, G. (6 de agosto de 2020). *Economipedia*. Obtenido de <https://economipedia.com/definiciones/registro>

Wikipedia. (20 de Agosto de 2020). *WIKIPEDIA La enciclopedia libre*. Obtenido de https://es.wikipedia.org/wiki/ISO/IEC_27000-series

Wikipedia. (9 de octubre de 2021). Obtenido de https://es.wikipedia.org/wiki/Servidor_HTTP_Apache

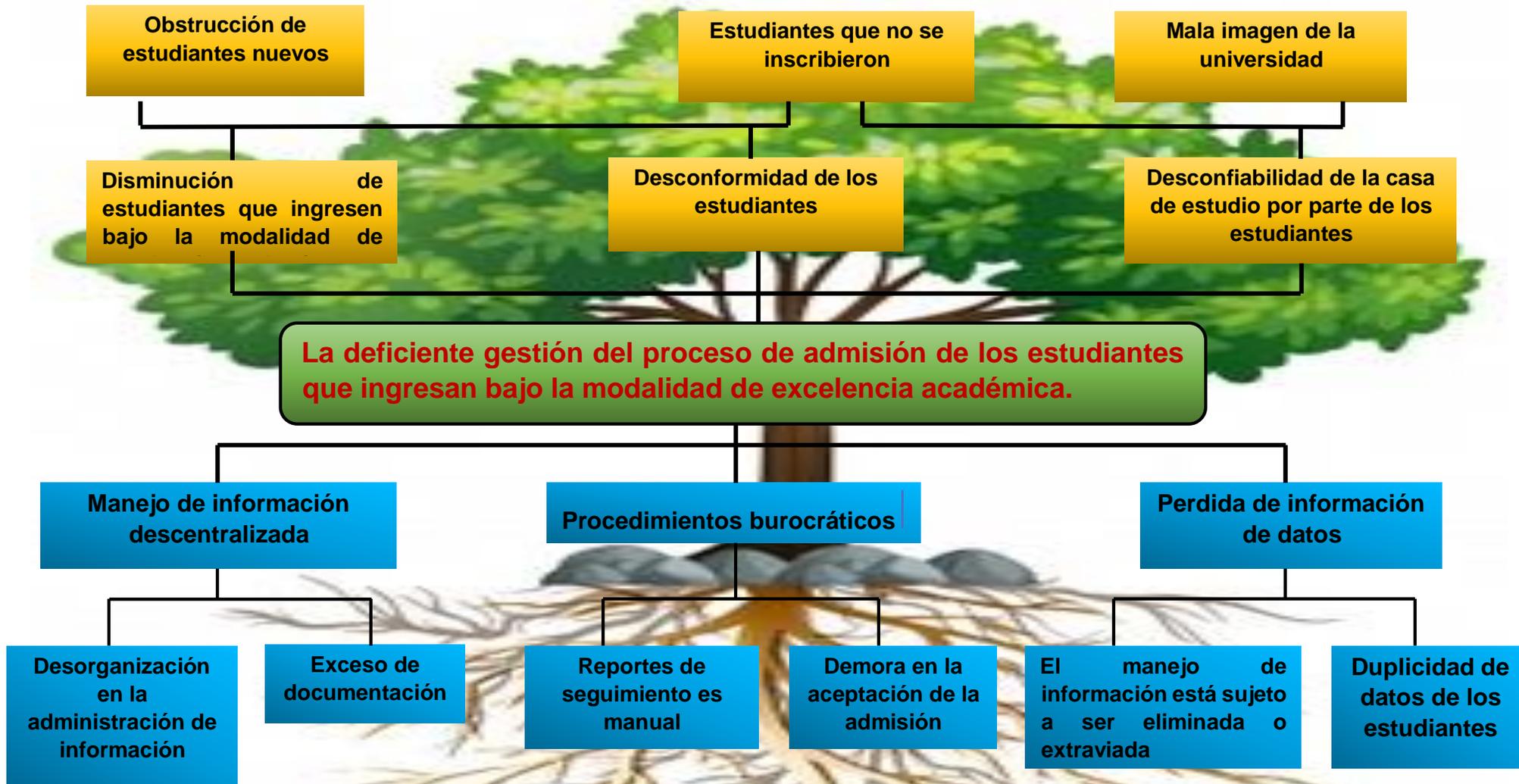
Wikipedia. (31 de octubre de 2021). Obtenido de https://es.wikipedia.org/wiki/Web_2.0

ANEXO

A

ÁRBOL DE
PROBLEMAS

ARBOL DE PROBLEMAS

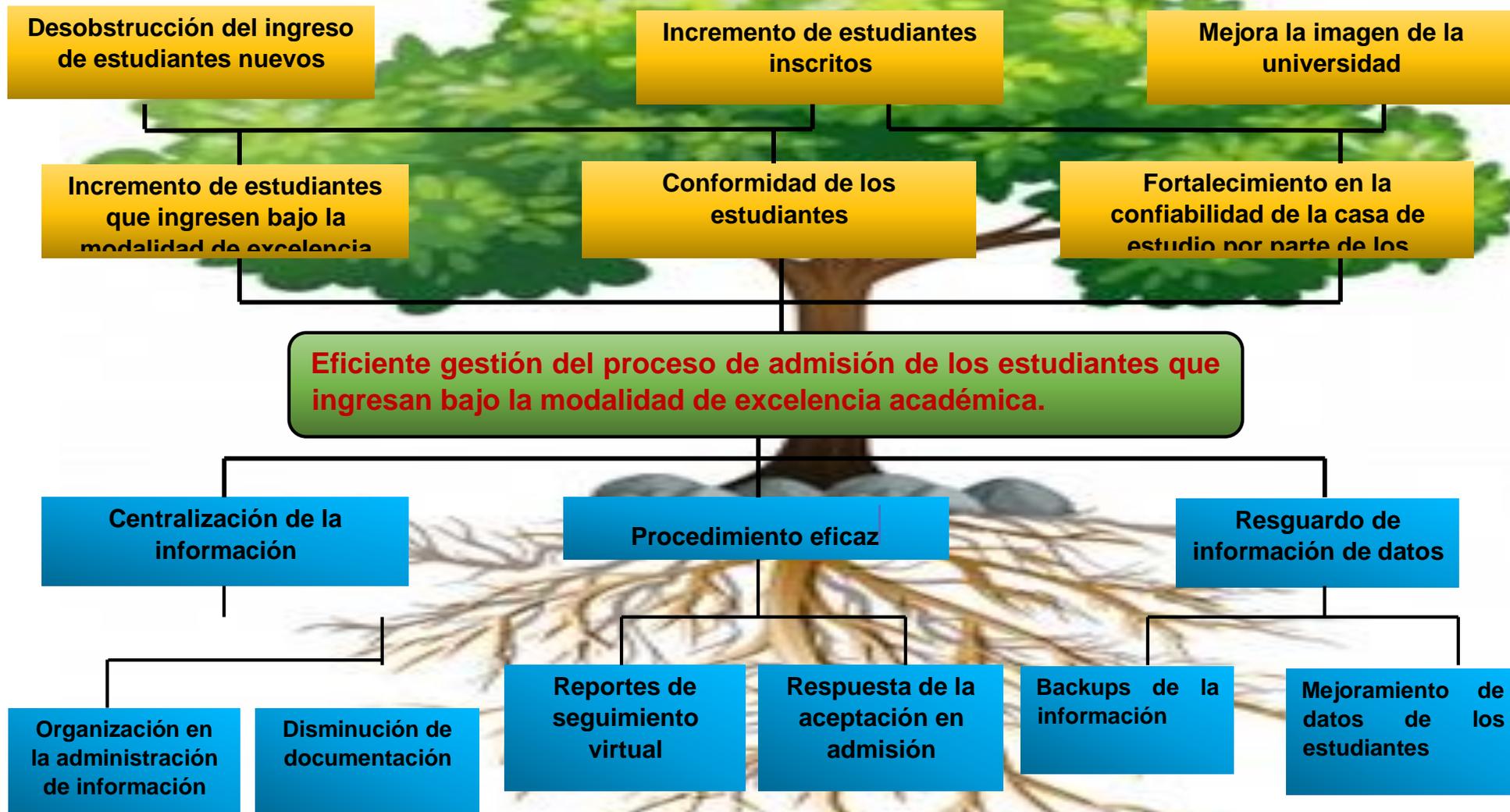


ANEXO

B

**ÁRBOL DE
OBJETIVOS**

ARBOL DE OBJETIVOS





FORMULARIO DE ENTREVISTA

NOMBRE DEL ENTREVISTADO.....

CARGO.....

FECHA.....

1. ¿CUMPLE CON SUS PRINCIPALES REQUERIMIENTOS EL SISTEMA?

Si

No

2. ¿CUENTA CON PROCESOS DISTRIBUIDOS EN EL SISTEMA?

Si

No

3. ¿EL SISTEMA PUEDE SER EJECUTADO EN CUALQUIER SISTEMA OPERATIVO?

Si

No

4. ¿CUENTA CON ESTRADA A DATOS Y MULTIPLES VENTANAS?

Si

No

5. ¿CUENTA CON LOS ARCHIVOS REQUERIDOS?

Si

No

6. ¿IMPRIME LOS TIPOS DE DOCUMENTOS NECESARIOS?

Si

No

7. ¿ CUENTAN CON LAS HERRAMIENTAS DE SOFTWARE NECESARIAS?

Si

No

8. ¿VERIFICO LAS MINIMAS OBSERVACIONES?

Si

No

9. ¿CUÁLES SON LAS EXPECTATIVAS SOBRE LA CONFIABILIDAD DEL SISTEMA?

Si

No

10. ¿CUMPLE CON LAS EXPECTATIVAS DE RENDIMIENTO DEL SISTEMA?

Si

No

11. ¿EXISTE UN ENCARGADO DEL MANTENIMIENTO DEL SISTEMA?

Si

No

12. ¿EXISTE CONOCIMIENTOS BASICOS DE SISTEMA?

Si

No

Gracias por su colaboración.

ANEXO

C

MANUAL DE

USUARIO

CARRERA DE
**INGENIERÍA DE
SISTEMAS**

**SISTEMA DE REGISTRO DE ADMISIÓN DE ESTUDIANTES
DE LA MODALIDAD DE EXCELENCIA ACADÉMICA**

**MANUAL
DE USUARIO**

Versión 1.0

Elaborado:
Karen Yhoselin Huaqui Quispe
GESTIÓN 2021

CONTENIDO

1. INTRODUCCIÓN.....	1
2. OBJETIVO DEL SISTEMA.	1
3. GUIA DEL SISTEMA.	1
3.1. Página principal	1
3.2. Registro de postulante	2
3.3. Sesión de ingreso al sistema	12
3.4. Módulo de administración de la página.	13
3.5. Módulo de administración de usuario	14
3.6. Módulo de técnico excelencia	17
3.7. Módulo coordinador excelencia	24
3.8. Reportes.....	27

MANUAL DE USUARIO

1. INTRODUCCIÓN.

El presente documento pretende servir de guía para el uso adecuado del sistema de registro académico para la Unidad de Vicerrectorado de la Universidad Pública de El Alto, el cual es un proyecto dirigido a los usuarios finales. Con este sistema se pretende administrar el registro de postulantes de la modalidad de excelencia académica.

2. OBJETIVO DEL SISTEMA.

El objetivo del manual, es permitir al usuario final, que pueda tener la administración adecuada del “Sistema de Registro de Admisión de Estudiantes de la Modalidad de Excelencia Académica”.

3. GUIA DEL SISTEMA.

3.1. Página principal

Para ingresar al sistema debe ingresar a la página <https://excelenciaacademica.upea.bo/inicio> donde se mostrará la pantalla de inicio:



En el inicio de la página se nos mostrará los botones de menú donde se pondrá acceder para que el estudiante pueda registrarse en el sistema. También se puede acceder al sistema con los distintos usuarios.

3.2. Registro de postulante

Para realizar el registro del postulante a la excelencia académica debe presionar en la opción POSTULARME como muestra en la siguiente pantalla:



Nos unidada a las convocatorias disponibles como nos muestra en la siguiente pantalla:

2



Una vez que el postulante elija la carrera a la cual desea postular debe hacer click en postular.



SISTEMA DE REGISTRO DE ADMISIÓN DE ESTUDIANTES DE LA MODALIDAD DE EXCELENCIA ACADÉMICA

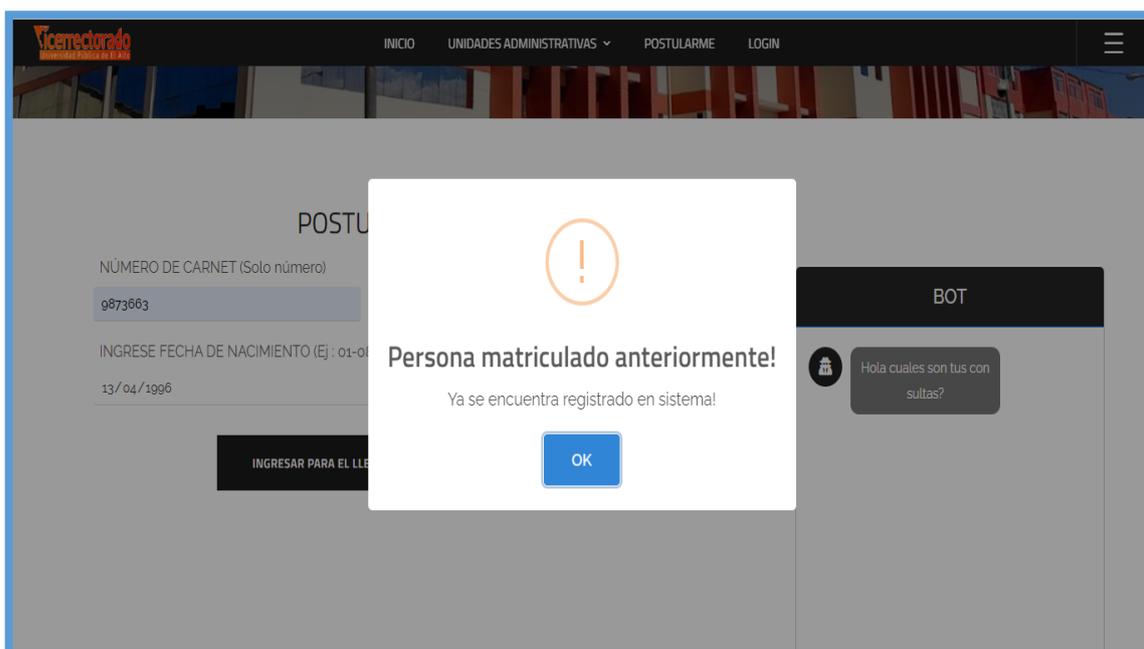
En caso de que la carrera que desea postular no tenga cupos saldrá el siguiente mensaje.

Después de presionar en la opción de postularme nos mandara a la siguiente pantalla:



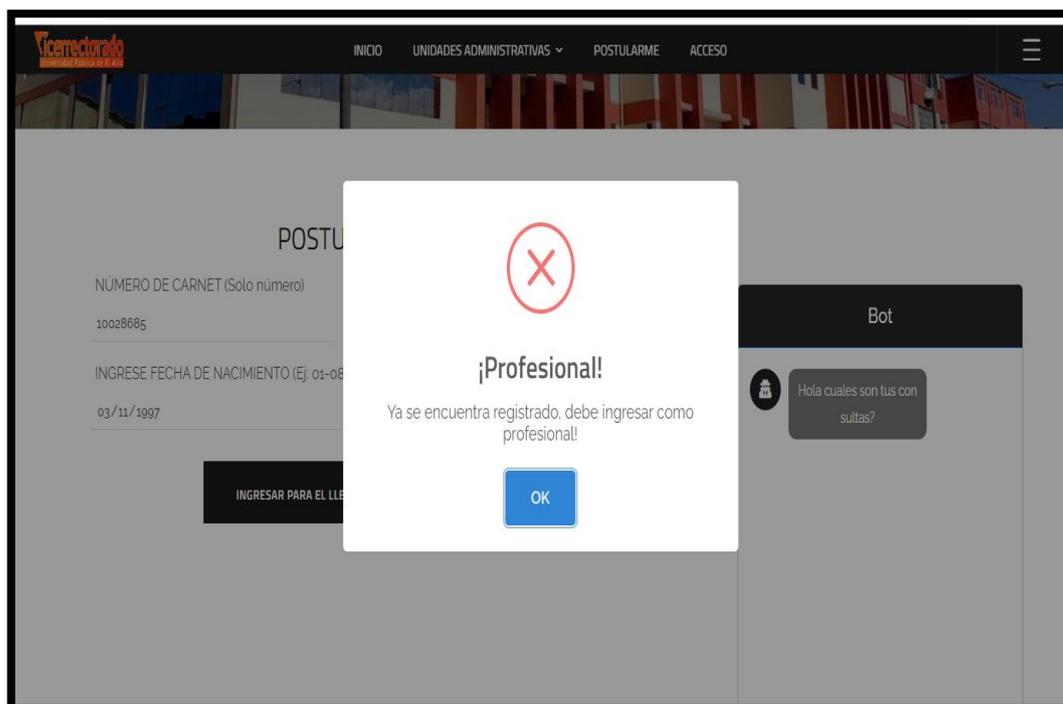
3

Después de completar los campos del registro y el sistema haga la verificación de los datos si es una persona matriculada nos mostrara el siguiente mensaje:



SISTEMA DE REGISTRO DE ADMISIÓN DE ESTUDIANTES DE LA MODALIDAD DE EXCELENCIA ACADÉMICA

También el sistema hace el control de si la persona es profesional, después de completar los campos del registro y se realice la verificación de los datos nos mostrara el siguiente mensaje:



Si el sistema después de haber ingresado los datos del postulante verifica que no es matriculado y no es profesional debe completar los siguientes campos y debe presionar en el siguiente botón:



POSTULARME

NÚMERO DE CARNET (Solo número) Complemento
Ingrese documento de identidad complemento

INGRESE FECHA DE NACIMIENTO (Ej: 01-08-2001)
dd/mm/aaaa

INGRESAR PARA EL LLENADO DEL FORMULARIO

SISTEMA DE REGISTRO DE ADMISIÓN DE ESTUDIANTES DE LA MODALIDAD DE EXCELENCIA ACADÉMICA

Después de que el sistema te permita ingresar para el llenado del formulario nos muestra el siguiente formulario:

LLENAR FORMULARIO

[Volver a Inicio](#)

CARRERA A LA QUE POSTULA

Fecha de inscripción: 13/11/2021
Carrera:
Sede:

DATOS PERSONALES

Apellido Paterno:
Apellido Materno:
Nombre(s):
Nacionalidad:
C.I.(Número): 987366339
Expedido:
Tipo de documento:
Fecha de Nacimiento: 14/01/2003
Genero:
Estado Civil:
Departamento nacimiento:
Provincia nacimiento:
Lugar nacimiento:
Telefono/Celular:
Correo electronico:

DIRECCION

Departamento:
Provincia:
Municipio:
DistritoN°:
Domicilio: Calle o Avenida:
Zona o Villa:
N°puerta:

DATOS DE EGRESO SECUNDARIA

Nombre Colegio Egreso:
Pais colegio:
Año egreso:
Tipo:
Bachillerato:
Otro:
Área:
Ciudad/Localidad:

UNIVERSIDAD QUE EXPIDE EL TITULO DE BACHILLER(Opcional)

Universidad:
Año de titulo:
Nro de titulo:

DATOS SOCIO-ECONOMICOS

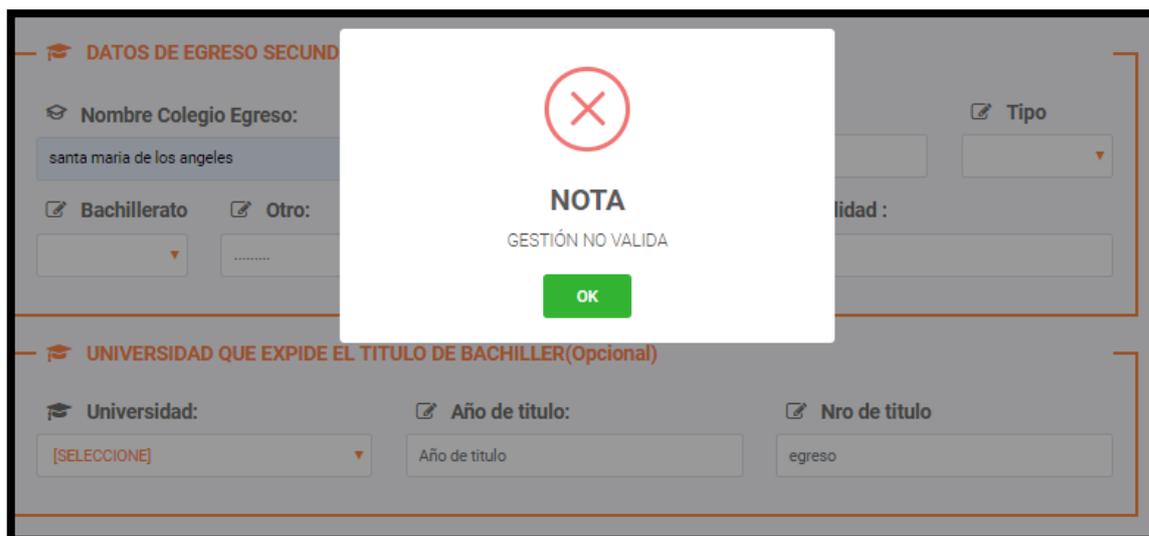
Vivienda que habita:
Características de la vivienda:
Trabaja?:
Jornada laboral:
Nro de hermanos que estudian en la universidad:
DATOS DE LA UNIDAD EDUCATIVA
Seleccione departamento:
Seleccione provincia:
Distrito:
POSICION (1RO, 2DO O 3ER. LUGAR)
Posición:
Promedio:

[GUARDAR DATOS](#)

5

SISTEMA DE REGISTRO DE ADMISIÓN DE ESTUDIANTES DE LA MODALIDAD DE EXCELENCIA ACADÉMICA

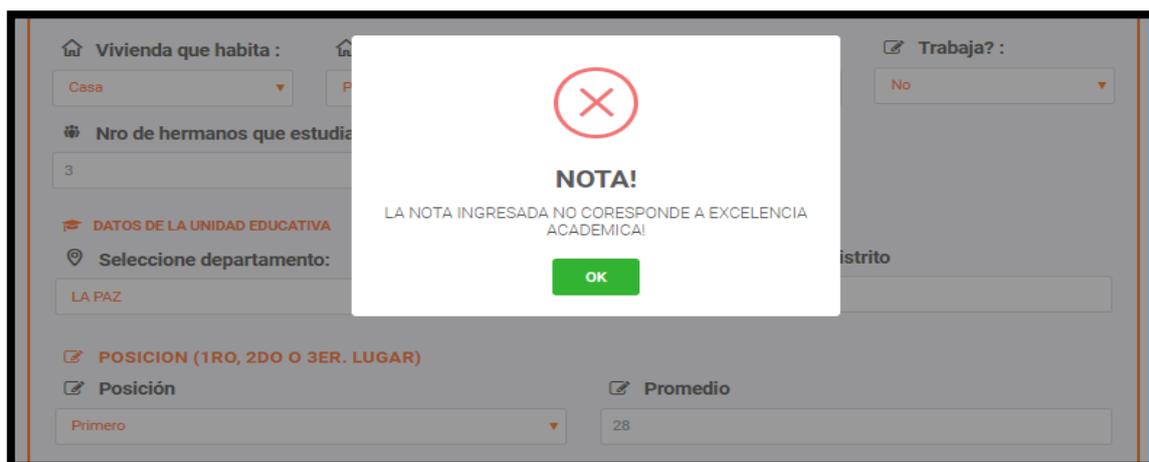
En el formulario solo pueden hacer el registro los postulantes de las dos gestiones anteriores a la fecha de no ser así nos muestra el siguiente mensaje:



The screenshot shows a registration form with a modal error message. The form is titled "DATOS DE EGRESO SECUNDARIO" and includes fields for "Nombre Colegio Egreso:" (filled with "santa maria de los angeles"), "Bachillerato", "Otro:", "Universidad que expide el título de Bachiller (Opcional)", "Universidad:", "Año de título:", and "Nro de título". The error message is a white box with a red 'X' icon, containing the text "NOTA GESTIÓN NO VALIDA" and an "OK" button.

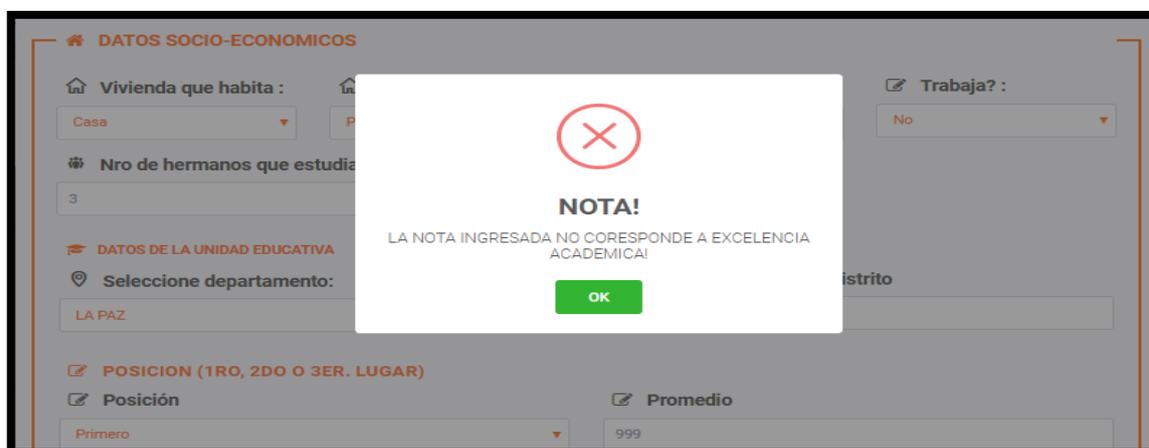
6

También el formulario controla que el postulante evite cometer errores en el registro del promedio inferior a una nota de excelencia.



The screenshot shows a registration form with a modal error message. The form is titled "DATOS DE LA UNIDAD EDUCATIVA" and includes fields for "Vivienda que habita:" (filled with "Casa"), "Nro de hermanos que estudia:" (filled with "3"), "Seleccione departamento:" (filled with "LA PAZ"), "Posición" (filled with "Primero"), and "Promedio" (filled with "28"). The error message is a white box with a red 'X' icon, containing the text "NOTA! LA NOTA INGRESADA NO CORRESPONDE A EXCELENCIA ACADÉMICA!" and an "OK" button.

También en el registro del promedio mayor a una nota de excelencia.



The screenshot shows a registration form with a modal error message. The form is titled "DATOS SOCIO-ECONOMICOS" and includes fields for "Vivienda que habita:" (filled with "Casa"), "Nro de hermanos que estudia:" (filled with "3"), "Seleccione departamento:" (filled with "LA PAZ"), "Posición" (filled with "Primero"), and "Promedio" (filled with "999"). The error message is a white box with a red 'X' icon, containing the text "NOTA! LA NOTA INGRESADA NO CORRESPONDE A EXCELENCIA ACADÉMICA!" and an "OK" button.

SISTEMA DE REGISTRO DE ADMISIÓN DE ESTUDIANTES DE LA MODALIDAD DE EXCELENCIA ACADÉMICA

Una vez que termine de completar los campos del formulario el postulante debe presionar guardar datos como se ve en la imagen.

The image shows a web form titled "DATOS SOCIO-ECONOMICOS". It contains several input fields: "Vivienda que habita:" with a dropdown menu showing "Casa"; "Características de la vivienda" with a dropdown menu showing "Propia de padres"; "Trabaja?:" with a dropdown menu showing "No"; "Nro de hermanos que estudian en la universidad" with a text input field containing "3"; "DATOS DE LA UNIDAD EDUCATIVA" section with "Seleccione departamento:" (LA PAZ), "Seleccione provincia:" (MURILLO), and "Distrito" (3); "POSICION (1RO, 2DO O 3ER. LUGAR)" section with "Posición" (Primero) and "Promedio" (99). At the bottom, a red button labeled "GUARDAR DATOS" is highlighted with a yellow callout box that says "Haga click aquí" with an arrow pointing to the button.

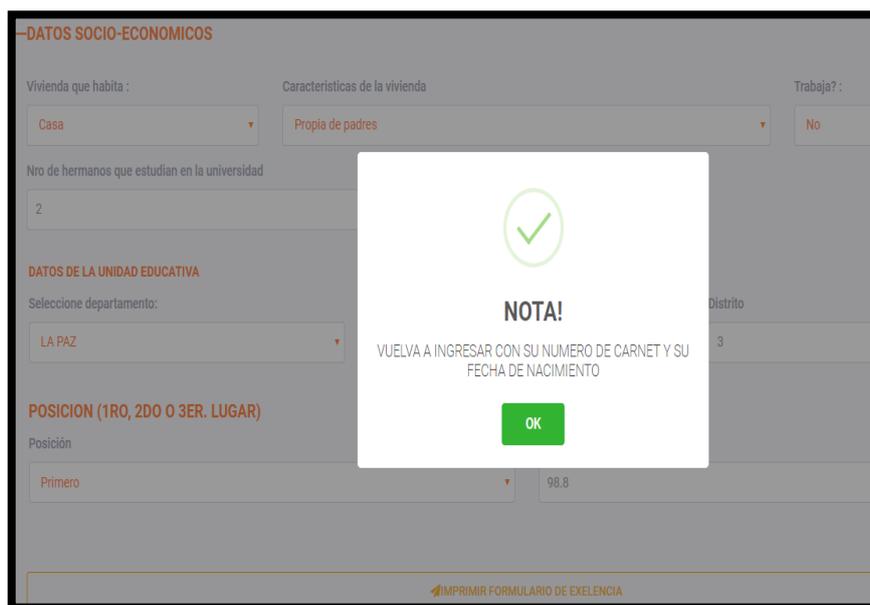
7

Una vez realizado el registro y el postulante presione guardar los datos nos mostrara el siguiente mensaje:

The image shows the same web form as above, but with a modal dialog box overlaid in the center. The dialog box has a white background and a grey border. It features a large orange exclamation mark icon at the top, followed by the text "NOTA!" in bold. Below that, it asks "Esta seguro de todos los datos que ingreso?". At the bottom of the dialog, there are two buttons: a green button labeled "Si, Guardar" and a red button labeled "Cancelar". The background form is dimmed.

SISTEMA DE REGISTRO DE ADMISIÓN DE ESTUDIANTES DE LA MODALIDAD DE EXCELENCIA ACADÉMICA

Una vez que haga la confirmación del registro nos mostrara el siguiente mensaje:



The image shows a web form titled "DATOS SOCIO-ECONOMICOS" with fields for "Vivienda que habita:", "Características de la vivienda", and "Trabaja?". A modal window is overlaid on the form with a green checkmark icon and the text "NOTA! VUELVA A INGRESAR CON SU NUMERO DE CARNET Y SU FECHA DE NACIMIENTO" and an "OK" button. The form also includes sections for "DATOS DE LA UNIDAD EDUCATIVA" and "POSICION (1RO, 2DO O 3ER. LUGAR)".

8

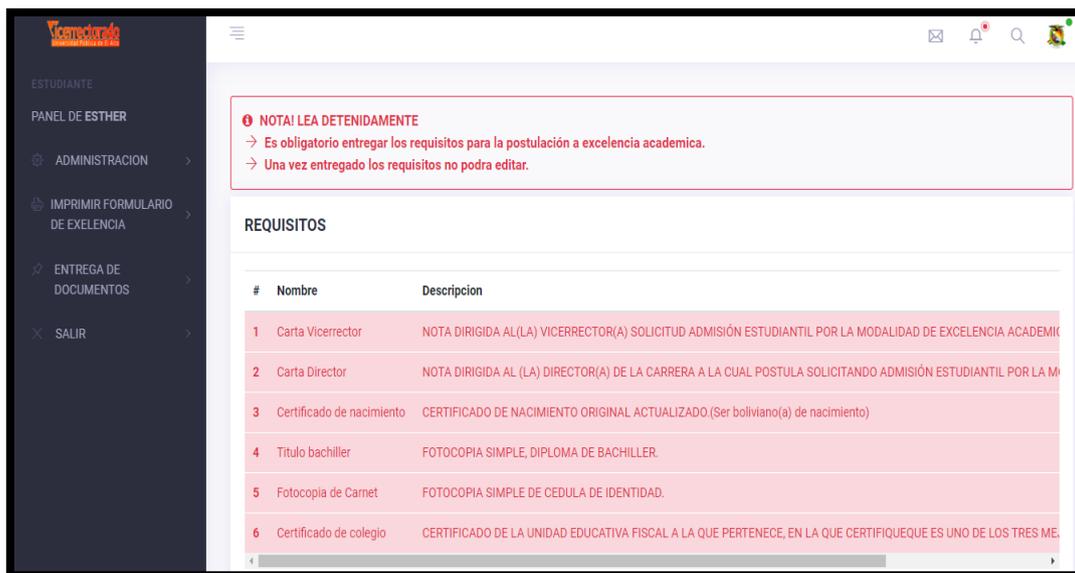
Debe volver a ingresar al sistema completando los campos correspondientes como muestra en la siguiente pantalla:



The image shows a registration screen titled "POSTULARME" with two input fields: "NÚMERO DE CARNET (Solo número)" and "INGRESE FECHA DE NACIMIENTO (Ej: 01-08-2001)". A yellow callout box "Complete este campo" points to the first field, and another yellow callout box "Complete este campo" points to the second field. A black button labeled "INGRESAR PARA EL LLENADO DEL FORMULARIO" is at the bottom, with a green callout box "Haga click aquí" pointing to it.

Una vez que el postulante ingrese sus datos ingresa al panel del estudiante y nos mostrara la siguiente pantalla donde el postulante debe ir agregando cada uno de los requisitos:

SISTEMA DE REGISTRO DE ADMISIÓN DE ESTUDIANTES DE LA MODALIDAD DE EXCELENCIA ACADÉMICA

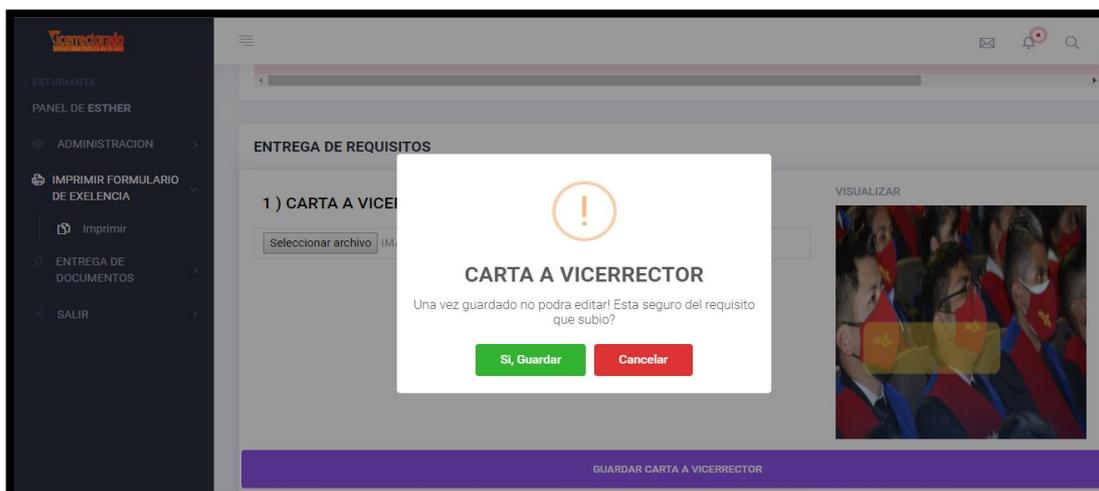


9

El postulante debe subir cada uno de los requisitos como muestra en la siguiente pantalla:

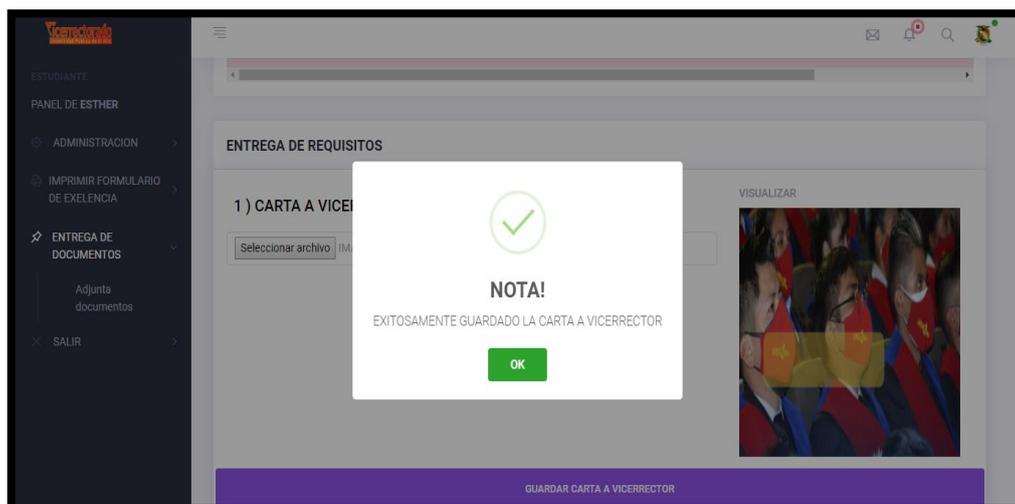


Una vez seleccionado el archivo mostrara el siguiente mensaje de la pantalla:



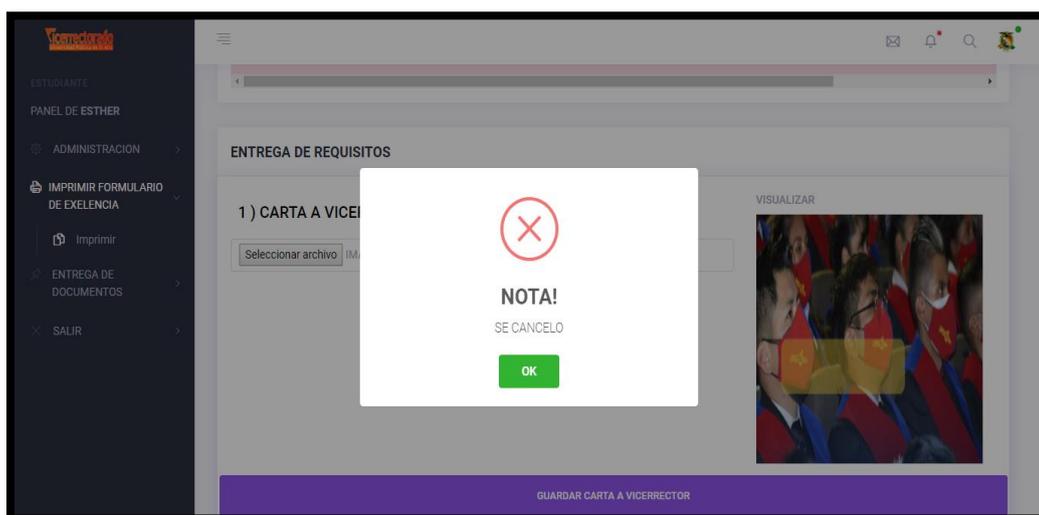
SISTEMA DE REGISTRO DE ADMISIÓN DE ESTUDIANTES DE LA MODALIDAD DE EXCELENCIA ACADÉMICA

Si acepta guardar el archivo seleccionado en la pantalla le mostrara el siguiente mensaje:



10

Si se desea cancelar los cambios de guardar archivos muestra el siguiente mensaje de la pantalla:



Cuando cargue correctamente el archivo la pantalla se mostrará de la siguiente manera:



SISTEMA DE REGISTRO DE ADMISIÓN DE ESTUDIANTES DE LA MODALIDAD DE EXCELENCIA ACADÉMICA

Una vez que se registren los datos del formulario debe presionar en el botón de imprimir formulario.

11

Se procede a la descarga e impresión del formulario como muestra en la siguiente pantalla:

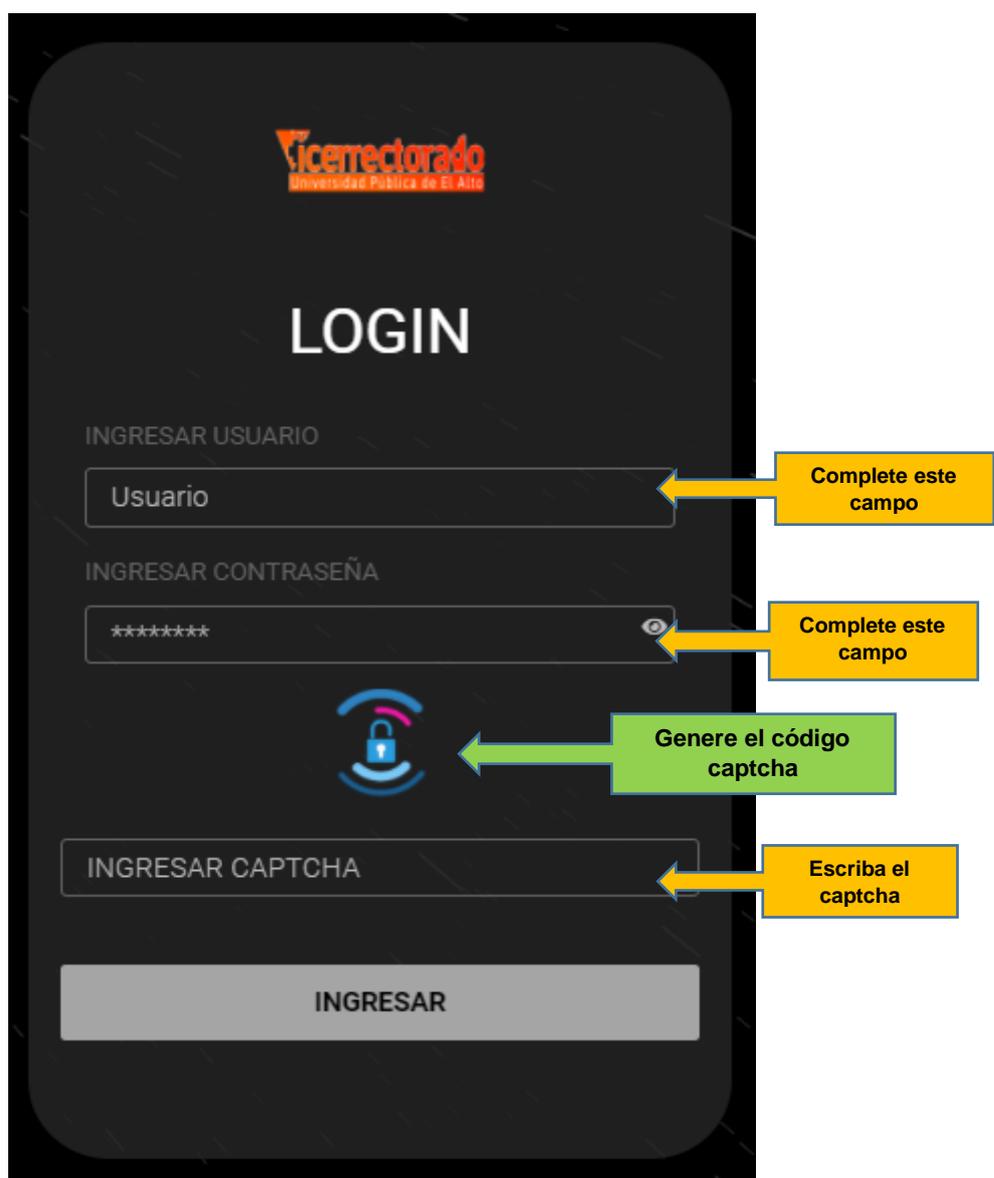
3.3. Sesión de ingreso al sistema

Para ingresar al panel de administración del sistema primeramente debe ingresar a la página web.



Luego presionar en login, este abrirá una pantalla de la siguiente manera donde deberá completar los siguientes campos:

12

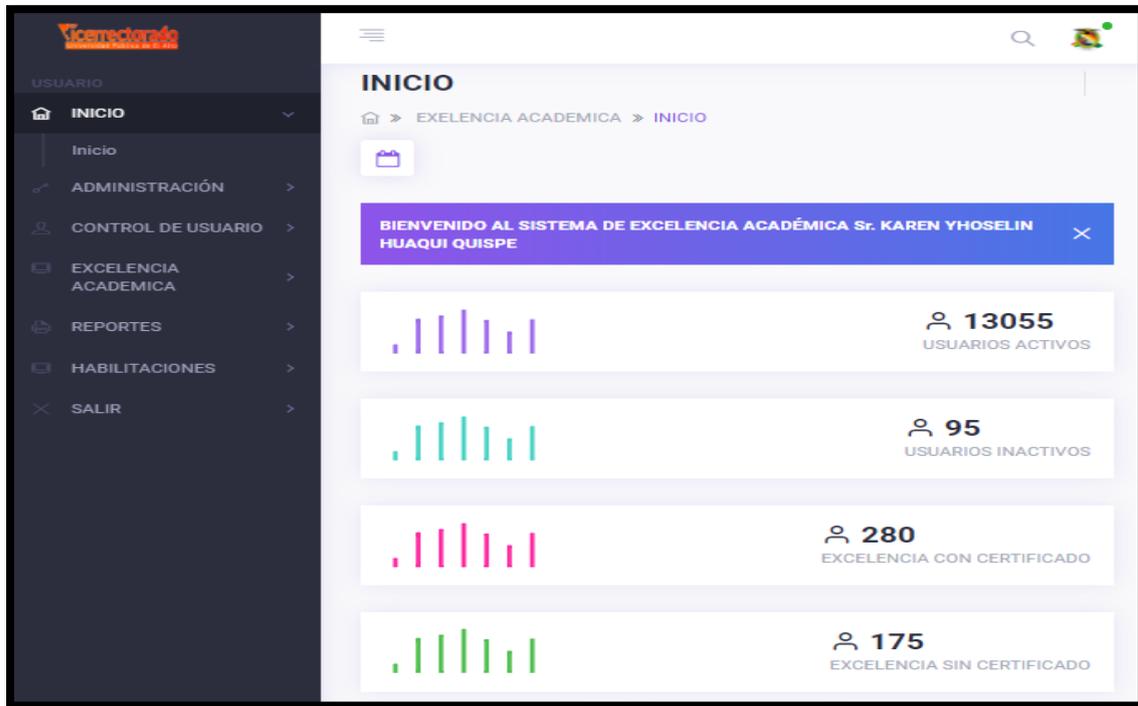


Una vez ingresado al sistema con un usuario y una contraseña proporcionado por el administrador del sistema podrá ingresar, al sistema de administración de la página.

SISTEMA DE REGISTRO DE ADMISIÓN DE ESTUDIANTES DE LA MODALIDAD DE EXCELENCIA ACADÉMICA

3.4. Módulo de administración de la página.

En el inicio de la administración de la página tenemos el conteo de los usuarios como muestra en la siguiente pantalla:



13

En este presente modulo nos permite administrar los datos de la página principal como nos muestra en la siguiente imagen:

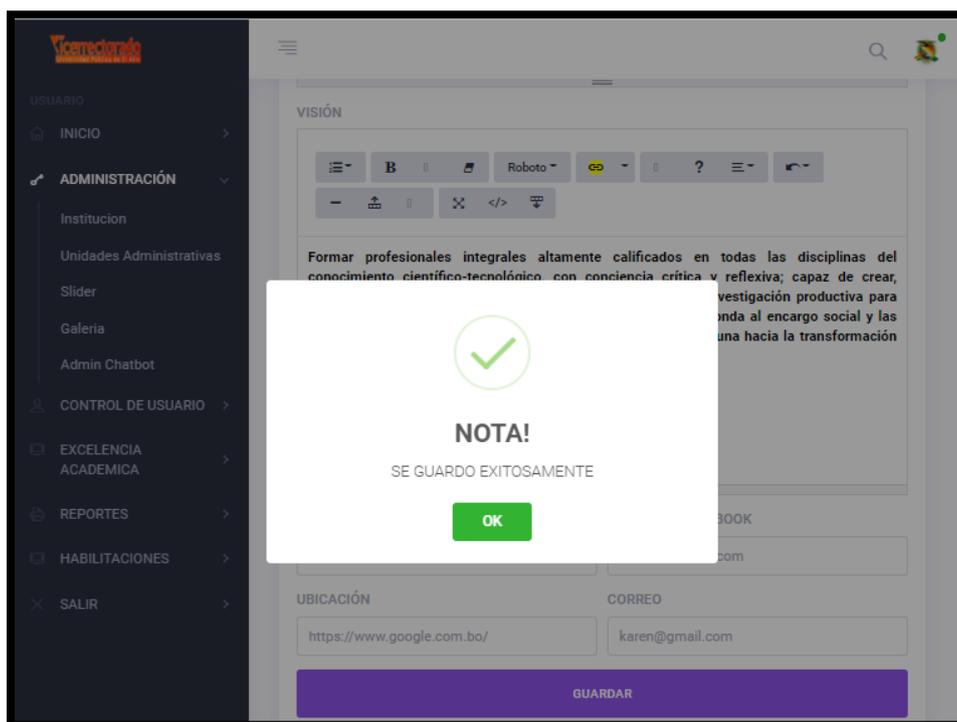


Escriba el texto aquí que desea mostrar en la página de inicio

VICERRECTORADO - UP

SISTEMA DE REGISTRO DE ADMISIÓN DE ESTUDIANTES DE LA MODALIDAD DE EXCELENCIA ACADÉMICA

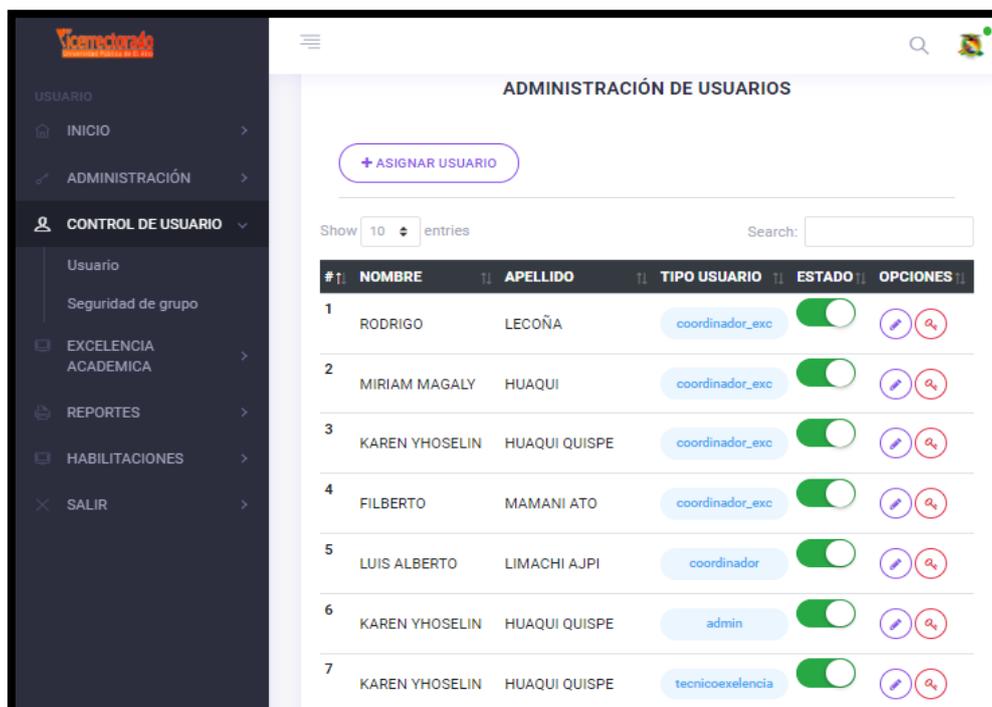
Una vez guardado los datos en pantalla nos envía el siguiente mensaje:



14

3.5. Módulo de administración de usuario

En este módulo, presentamos la administración de usuario de la siguiente manera. Una vez que el usuario administrador ingrese al sistema nos mostrará la siguiente pantalla de inicio:



El administrador podrá realizar las siguientes acciones:

SISTEMA DE REGISTRO DE ADMISIÓN DE ESTUDIANTES DE LA MODALIDAD DE EXCELENCIA ACADÉMICA

Paso 1 Nuevo usuario: En este botón, nos permite crear un usuario presentando un panel de formulario de la siguiente manera.

The screenshot shows a mobile application interface with a dark sidebar menu on the left containing options like 'USUARIO', 'INICIO', 'ADMINISTRACIÓN', 'CONTROL DE USUARIO', 'EXCELENCIA ACADEMICA', 'REPORTES', 'HABILITACIONES', and 'SALIR'. The main content area displays a 'CREAR USUARIO Y CONTRASEÑA' form with the following fields and annotations:

- CI:** A text input field with a purple pencil icon. A yellow callout box points to it with the text: "Ingrese su número de CI el sistema completara los campos de nombres y apellidos".
- NOMBRE:** A text input field with a purple pencil icon.
- APELLIDOS:** A text input field with a purple pencil icon.
- TIPO USUARIO:** A dropdown menu. A yellow callout box points to it with the text: "Tipo de usuario que asignara".
- NUEVO USUARIO:** A section with a purple user icon and a search input field labeled "Buscar usuario...". A yellow callout box points to it with the text: "Se le asignara un usuario".
- NUEVA CONTRASEÑA:** A section with a purple magnifying glass icon and a search input field labeled "Buscar contraseña...". A yellow callout box points to it with the text: "Se genera la contraseña del usuario".

At the bottom of the form are two buttons: "Cerrar" (red) and "GUARDAR" (green). The background shows a list of users with names like "KAREN YHOSELIN HUAQUI QUISPE" and "FILBERTO MAMANI ATO".

15

Paso 2 Editar usuario: En este botón de editar nos permite modificar los datos del usuario, como se puede ver el panel de formulario.

The screenshot shows the 'EDITAR USUARIO Y CONTRASEÑA' form. The fields are pre-filled with the following information:

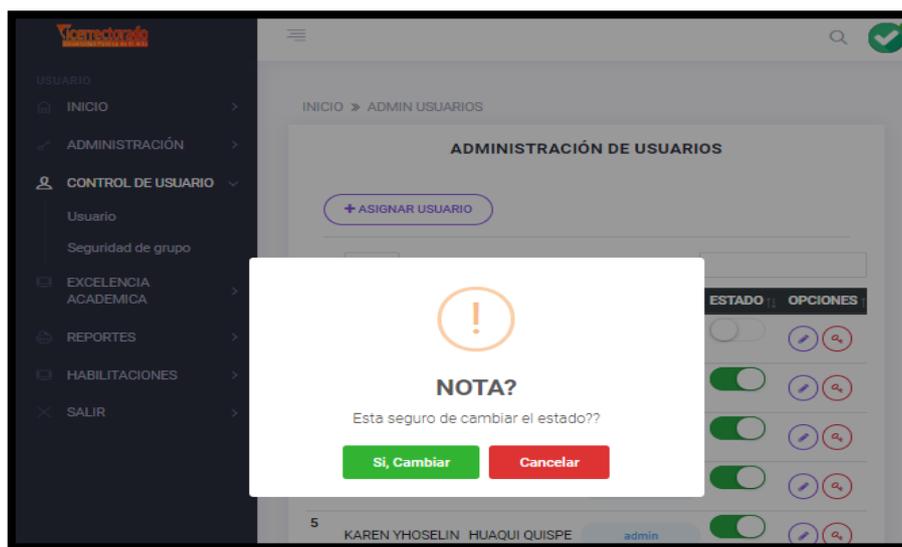
- USUARIO:** 9873660
- NOMBRE:** MIRIAM MAGALY
- APELLIDO:** HUAQUI
- TIPO USUARIO:** coordinador_exc

A yellow callout box points to the 'TIPO USUARIO' dropdown menu with the text: "Podrá editar el tipo de usuario". At the bottom of the form are two buttons: "Cerrar" (red) and "GUARDAR" (green). The background shows a list of users with names like "FILBERTO MAMANI ATO" and "LUIS ALBERTO LIMACHI AJPI".

Paso 3 Cambiar estado: El siguiente enlace nos permite guardar cambiar de

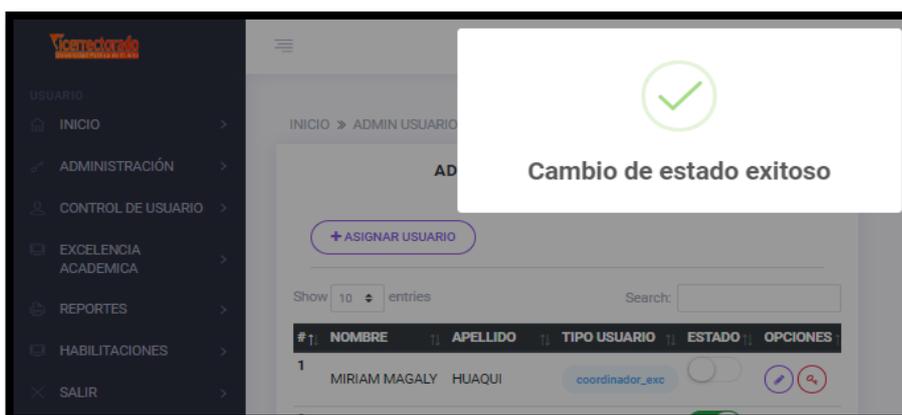
SISTEMA DE REGISTRO DE ADMISIÓN DE ESTUDIANTES DE LA MODALIDAD DE EXCELENCIA ACADÉMICA

estado al usuario de ACTIVO O INACTIVO de la siguiente forma.

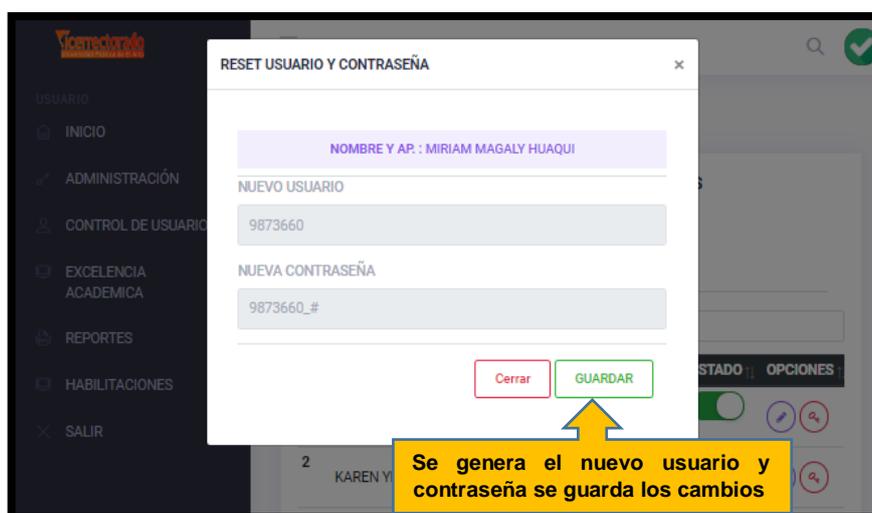


16

Cuando el cambio de estado se realice mostrara el siguiente mensaje:

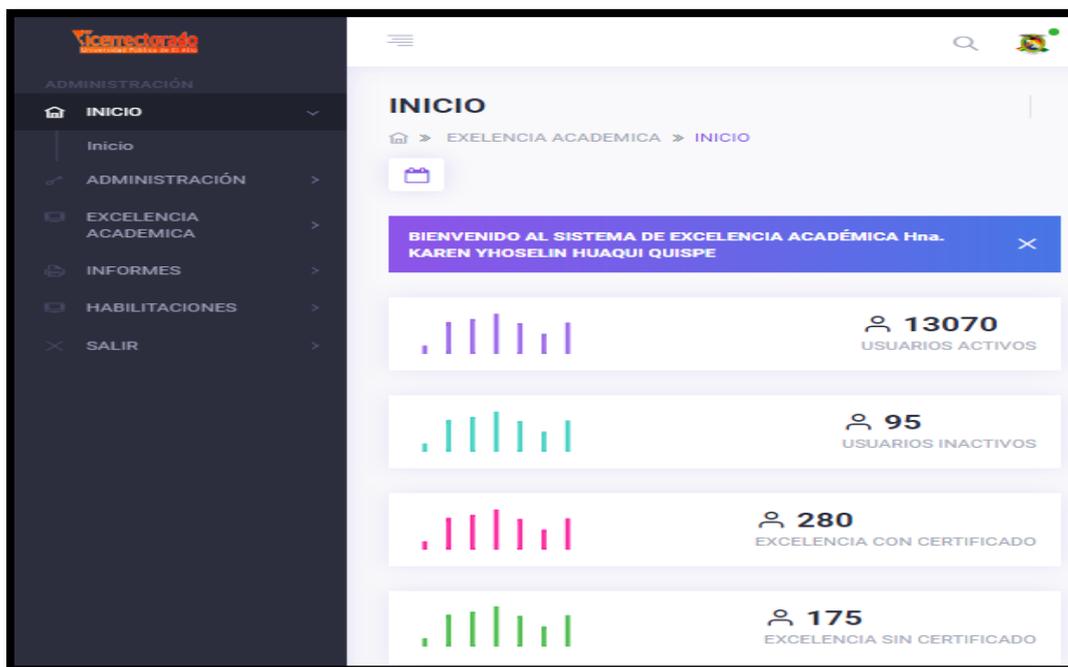


Paso 4 Reseteo usuario: En este botón nos permite resetear el usuario y la contraseña de la siguiente forma.



3.6. Módulo de técnico excelencia

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



17

El técnico de excelencia podrá realizar las siguientes acciones como se muestra en la siguiente pantalla:



Se tiene las siguientes opciones:

- ✓ Revisar: Permite realizar la revisión de documentación del postulante.
- ✓ Editar: Permite realizar modificaciones

SISTEMA DE REGISTRO DE ADMISIÓN DE ESTUDIANTES DE LA MODALIDAD DE EXCELENCIA ACADÉMICA

Cuando el usuario presione en el botón de Revisar mostrar la siguiente pantalla:

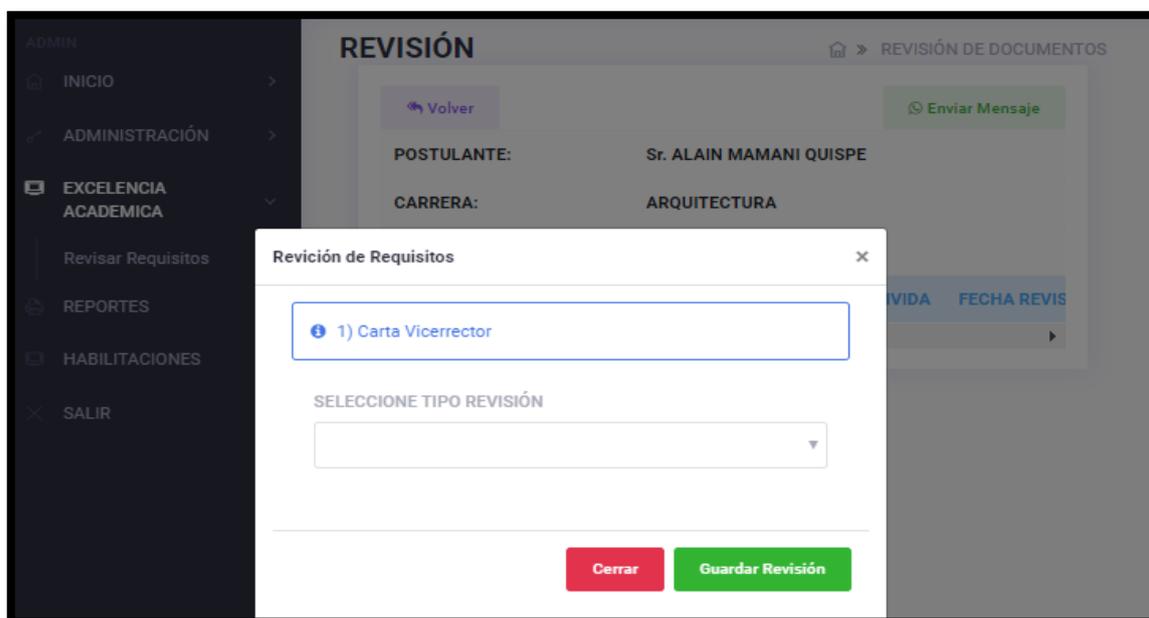


Donde podrá verificar cada uno de los requisitos que el postulante subió al sistema sean los correctos.



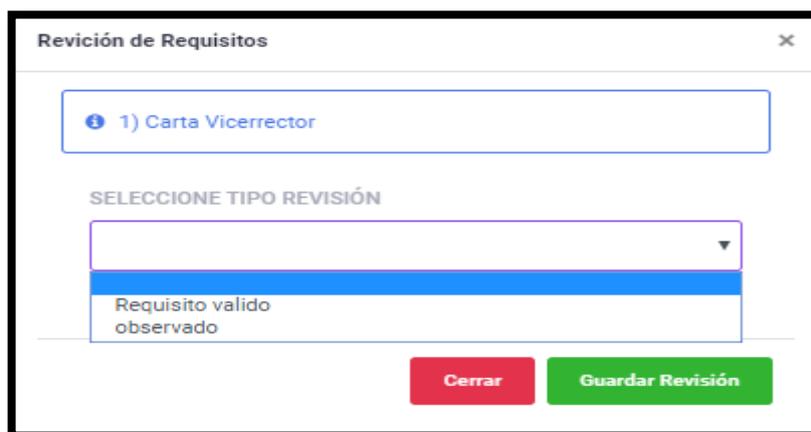
SISTEMA DE REGISTRO DE ADMISIÓN DE ESTUDIANTES DE LA MODALIDAD DE EXCELENCIA ACADÉMICA

Después de presionar en REVISAR mostrara la siguiente pantalla:



19

Donde el técnico de excelencia tendrá las siguientes opciones como se muestra en pantalla:

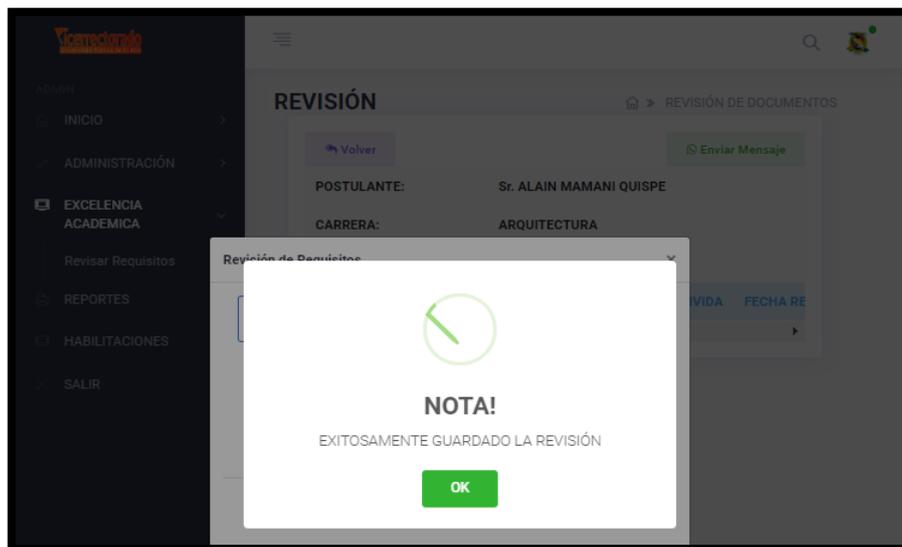


Después de seleccionar la opción de requisito valido el sistema mostrara el siguiente mensaje:



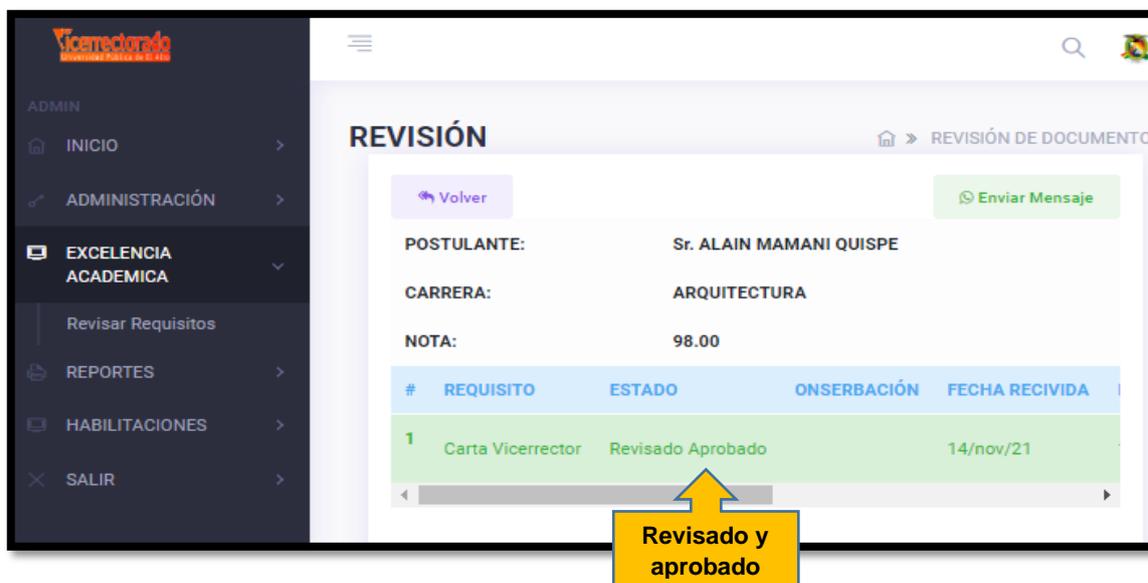
SISTEMA DE REGISTRO DE ADMISIÓN DE ESTUDIANTES DE LA MODALIDAD DE EXCELENCIA ACADÉMICA

Si guarda la revisión de manera correcta el sistema le enviara el siguiente mensaje de pantalla:



20

Después de realizar la revisión de los documentos el sistema mostrada el estado de los requisitos como muestra la siguiente pantalla:



En caso de que el técnico de excelencia observe el requisito que fue subido al sistema mostrados el siguiente estado como muestra en la siguiente pantalla:

SISTEMA DE REGISTRO DE ADMISIÓN DE ESTUDIANTES DE LA MODALIDAD DE EXCELENCIA ACADÉMICA

REVISIÓN

Volver Enviar Mensaje

POSTULANTE: Sr. ALAIN MAMANI QUISPE

CARRERA: ARQUITECTURA

NOTA: 98.00

#	REQUISITO	ESTADO	ONSERBACIÓN	FECHA RECIVIDA	FECHA REVISADO	USUARIO REVISOR
1	Carta Vicerrector	Observado	DOCUM ENTO	14/nov/21	15/nov/21	Us. KAREN YHOSELIN HU

Observado

21

Cuando el técnico de excelencia académica desea realizar modificaciones debe presionar en el siguiente botón como muestra la siguiente pantalla:

REVISIÓN

Volver Enviar Mensaje

POSTULANTE: Sr. ESTHER QUISPE MAMANI

CARRERA: ARQUITECTURA

NOTA: 98.00

#	REQUISITO	ESTADO	ONSERBACIÓN	FECHA RECIVIDA	FECHA REVISADO	USUARIO REVISOR	ACCIONES
1	Carta Vicerrector	Revisado Aprobado		11/nov/21	11/nov/21	Us. KAREN YHOSELIN HUAQUI QUISPE	Vizualizar
2	Carta Director	Revisado Aprobado		11/nov/21	11/nov/21	Us. KAREN YHOSELIN HUAQUI QUISPE	Vizualizar
3	Certificado de nacimiento	Revisado Aprobado		11/nov/21	11/nov/21	Us. KAREN Y	Vizualizar

Haga click para la visualizar cada uno de los requisitos de cada postulante y realizar modificaciones.

Después de realizar la revisión de los documentos el técnico de excelencia académica deberá enviar un mensaje a través de WhatsApp al postulante con la confirmación de su aprobación u observación de ser así.

SISTEMA DE REGISTRO DE ADMISIÓN DE ESTUDIANTES DE LA MODALIDAD DE EXCELENCIA ACADÉMICA

REVISIÓN REVISIÓN DE DOCUMENTOS

[Volver](#) [Enviar Mensaje](#)

POSTULANTE: Sr. ESTHER QUISPE MAMANI

CARRERA: ARQUITECTURA

NOTA: 98.00

#	REQUISITO	ESTADO	ONSERBACIÓN	FECHA RECIVIDA	F
1	Carta Vicerrector	Revisado Aprobado		11/nov/21	1
2	Carta Director	Revisado Aprobado		11/nov/21	1
3	Certificado de nacimiento	Revisado Aprobado		11/nov/21	1

22

El sistema también permite al técnico de excelencia poder editar datos de los postulantes en caso de ser necesario después de presionar en **EDITAR** mostrara la siguiente pantalla:

Editar datos

Nombres: JUAN ELOY

Primer Apellido: VILCA

Segundo Apellido: NINA

promedio: 98.00

Colegio: PEDRO DOMINGO MURILLO

Posición: primero

[Actualizar](#)

**SISTEMA DE REGISTRO DE ADMISIÓN DE ESTUDIANTES DE
LA MODALIDAD DE EXCELENCIA ACADÉMICA**

Para Imprimir la resolución vicerrectoral se debe presionar en el siguiente icono como se muestra en la siguiente pantalla:

primero	ADMINISTRACIÓN DE EMPRESAS	Con certificado		Vizualizar
primero	ADMINISTRACIÓN DE EMPRESAS	Con certificado		Vizualizar
primero	ADMINISTRACIÓN DE EMPRESAS	Con certificado		Vizualizar

Haga click en el siguiente icono para generar la resolución

ANEXO

Vicerrectorado
Universidad Pública de El Alto

RESOLUCIÓN VICERRECTORAL EXC.ACD.VCR. N° 280/2021

VISTOS:
A solicitud en Vicerrectorado con hoja de ruta N° 100/2021 de fecha 11 de noviembre de 2021, de QUISPE MAMANI ESTHER con C.I. 26663776 LP, pidiendo el ingreso a la Universidad Pública de El Alto, por la modalidad de EXCELENCIA ACADÉMICA.

CONSIDERANDO:

Que, habiendo acreditado su filiación personal con el Certificado de Nacimiento Original actualizado, Fotocopia del Diploma de Bachiller, Fotocopia de Cédula de Identidad, Certificado expedido por el(a) Director(a) del Establecimiento, refrendado por la Dirección Distrital correspondiente.

Que, tomando en cuenta el visto bueno de aprobación emitido por el(a) Director(a) de la Carrera de ARQUITECTURA autorizando el ingreso bajo la modalidad de EXCELENCIA ACADÉMICA.

Que, el impetrante ha cumplido con todos los requisitos establecidos en los artículos 2, 26 y 27 del Reglamento de Admisión Estudiantil de la Universidad Pública de El Alto.

POR TANTO, EL VICERRECTOR EN USO DE SUS ATRIBUCIONES, RESUELVE:

Art. Único.- Aprobar el ingreso a la Carrera de ARQUITECTURA, de la Universidad Pública de El Alto bajo la modalidad de EXCELENCIA ACADÉMICA en favor del(a) señor(a) QUISPE MAMANI ESTHER con C.I. 26663776 LP, debiendo el(a) interesado(a) realizar los trámites respectivos en la Dirección de Carrera.

Es dado en la Universidad Pública de El Alto a los 15 días del mes de Noviembre del 2021.

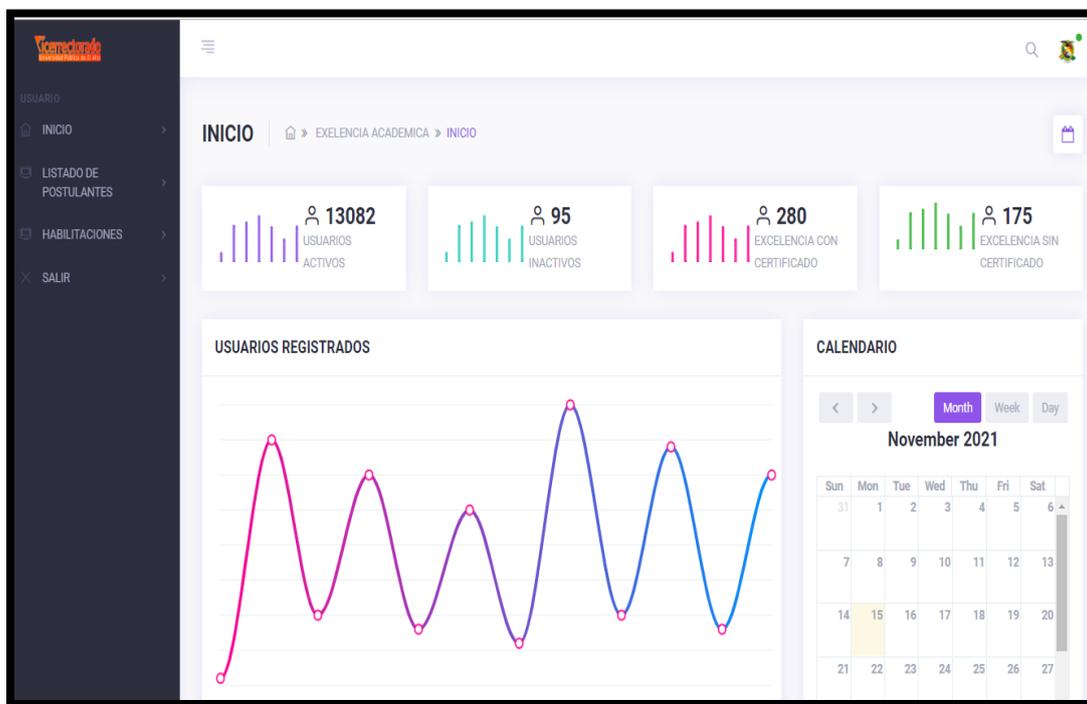
Regístrese, desglósesse, comuníquese, cúmplase y archívese.

ES CONFORME:



3.7. Módulo coordinador excelencia

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



24

En este módulo se tiene las siguientes opciones que el usuario de coordinador podrá habilitar:

Habilitación de convocatorias

The screenshot shows the 'CONTROL DEL CUPOS DE EXCELENCIA ACADÉMICA POR CARRERA' page. At the top, there is a breadcrumb 'INICIO > CONTROL DEL CUPOS'. A prominent yellow callout box with a black border contains the text 'Haga click para habilitar una nueva convocatoria.' with an arrow pointing to a purple button labeled '+ Habilitar convocatoria'. Below the button, there is a 'Show 10 entries' dropdown and a search field. A table with a dark header is visible, containing the following columns: #, CARRERA, GESTIÓN, MODALIDAD, ETAPAS, INICIO, CONCLUSIÓN, and FECHA CERTIFICADO DE HABILITACIÓN. The first row of data shows: 1, INGENIERÍA DE SISTEMAS, I/2022-PRE-UNIV, EXCELENCIA ACADÉMICA, primera etapa, 2021-11-12, 2021-11-26, and 2021-11-26.

**SISTEMA DE REGISTRO DE ADMISIÓN DE ESTUDIANTES DE
LA MODALIDAD DE EXCELENCIA ACADÉMICA**

Después de presionar en habilitar convocatoria mostrara la siguiente pantalla:

Habilitar fecha

Carrera
- Seleccione carrera -

Fecha inicio
dd/mm/aaaa

Fecha conclusion
dd/mm/aaaa

Fecha de impresion de Certificado de Habilitacion
dd/mm/aaaa

insertar cupos

Adjuntar imagen
Seleccionar archivo Ningún ar...ccionado

Seleciones la fecha inicio de la convocatoria

Seleciones la fecha final de la convocatoria

Selecciona la fecha para la impresión de certificado de habilitación

Inserta los cupos de cada convocatoria para la modalidad de excelencia académica

Selecciona la imagen de la convocatoria

Cancelar Habilitar

25

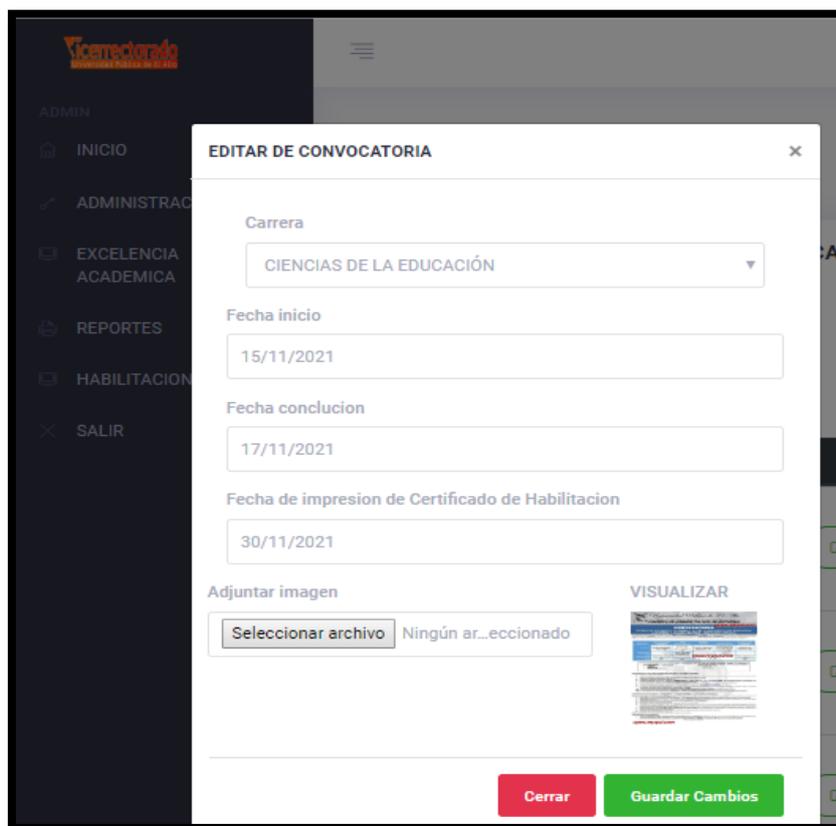
Después de haber realizado las habilitaciones de las convocatorias se podrá realizar las siguientes acciones como muestra en la siguiente pantalla:

CUPOS	ACCIÓN
11	[Editar] [Eliminar] [Cupo]
8	[Editar] [Eliminar] [Cupo]
10	[Editar] [Eliminar] [Cupo]

- **Editar:** Podrá editar el coordinador la convocatoria publicada.
- **Eliminar:** El coordinador podrá eliminar la convocatoria
- **Cupos:** Podrá el coordinador el número de cupos.

SISTEMA DE REGISTRO DE ADMISIÓN DE ESTUDIANTES DE LA MODALIDAD DE EXCELENCIA ACADÉMICA

Cuando se presiona en editar la convocatoria nos mostrará la siguiente pantalla donde se podrá modificar las fechas de inicio y conclusión de la convocatoria.



ADMIN

- INICIO
- ADMINISTRACIÓN
- EXCELENCIA ACADEMICA
- REPORTES
- HABILITACIONES
- SALIR

EDITAR DE CONVOCATORIA

Carrera
CIENCIAS DE LA EDUCACIÓN

Fecha inicio
15/11/2021

Fecha conclusion
17/11/2021

Fecha de impresion de Certificado de Habilitacion
30/11/2021

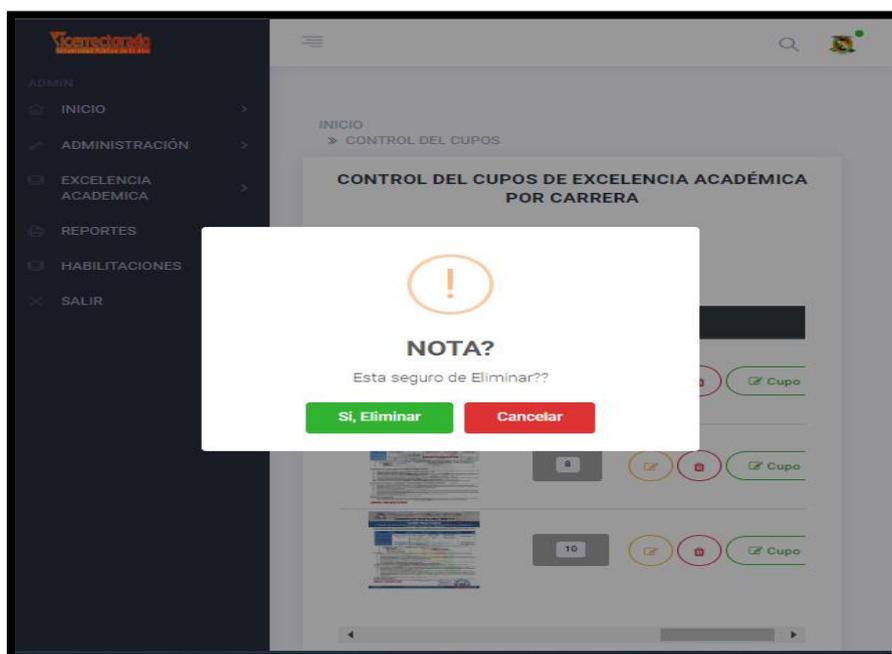
Adjuntar imagen
Seleccionar archivo Ningún archivo adjuntado

VISUALIZAR

Cerrar Guardar Cambios

26

Después de presionar en el icono de eliminar le mostrar el siguiente mensaje el sistema:



ADMIN

- INICIO
- ADMINISTRACIÓN
- EXCELENCIA ACADEMICA
- REPORTES
- HABILITACIONES
- SALIR

CONTROL DEL CUPOS DE EXCELENCIA ACADÉMICA POR CARRERA

NOTA?
Esta seguro de Eliminar??

Si, Eliminar Cancelar

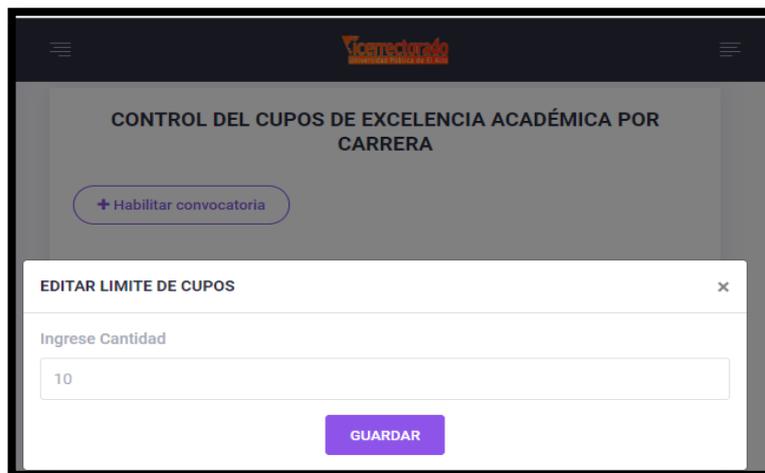
Cupo

Cupo

Cupo

SISTEMA DE REGISTRO DE ADMISIÓN DE ESTUDIANTES DE LA MODALIDAD DE EXCELENCIA ACADÉMICA

Si el coordinador desea editar los cupos de la modalidad de excelencia mostrara la siguiente



CONTROL DEL CUPOS DE EXCELENCIA ACADÉMICA POR CARRERA

+ Habilitar convocatoria

EDITAR LIMITE DE CUPOS

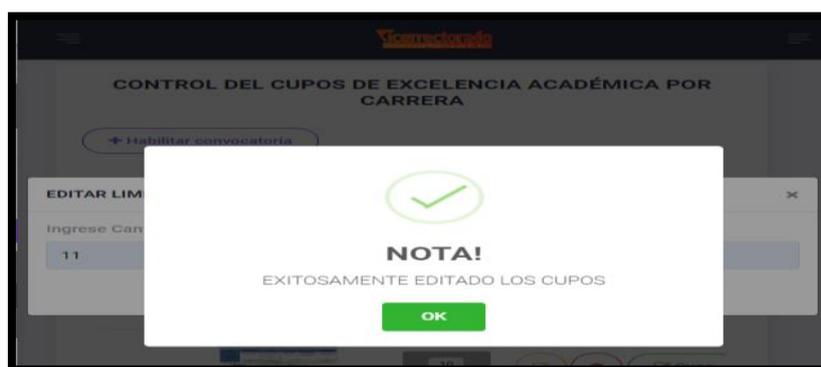
Ingrese Cantidad

10

GUARDAR

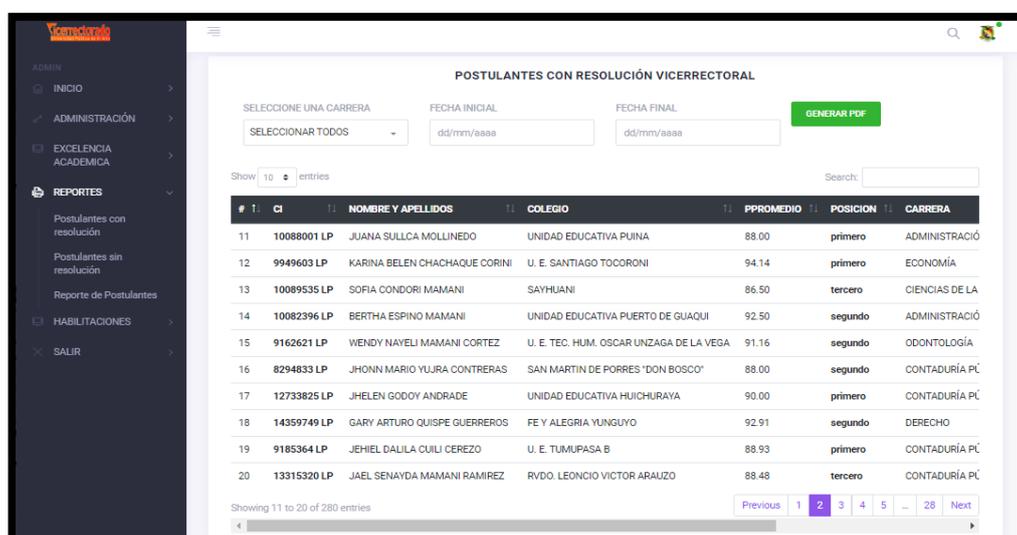
27

Después de que se edite el campo de los cupos de la convocatoria el sistema mostrara el siguiente mensaje:



3.8. Reportes

En este módulo de reportes el sistema mostrar la siguiente pantalla:



POSTULANTES CON RESOLUCIÓN VICERRECTORAL

SELECCIONE UNA CARRERA: SELECCIONAR TODOS

FECHA INICIAL: dd/mm/aaaa

FECHA FINAL: dd/mm/aaaa

GENERAR PDF

Show 10 entries

#	CI	NOMBRE Y APELLIDOS	COLEGIO	PPROMEDIO	POSICION	CARRERA
11	10088001 LP	JUANA SULLCA MOLLINADO	UNIDAD EDUCATIVA PUINA	88.00	primero	ADMINISTRACIÓ
12	9949603 LP	KARINA BELEN CHACHAQUE CORINI	U. E. SANTIAGO TOCORONI	94.14	primero	ECONOMÍA
13	10089535 LP	SOFIA CONDORI MAMANI	SAYHUANI	86.50	tercero	CIENCIAS DE LA
14	10082396 LP	BERTHA ESPINO MAMANI	UNIDAD EDUCATIVA PUERTO DE GUAQUI	92.50	segundo	ADMINISTRACIÓ
15	9162621 LP	WENDY NAVELI MAMANI CORTEZ	U. E. TEC. HUM. OSCAR UNZAGA DE LA VEGA	91.16	segundo	ODONTOLOGÍA
16	8294833 LP	JHONN MARIO YUJRA CONTRERAS	SAN MARTIN DE PORRES "DON BOSCO"	88.00	segundo	CONTADURÍA PÚ
17	1273825 LP	JHELEN GODOY ANDRADE	UNIDAD EDUCATIVA HUICHURAYA	90.00	primero	CONTADURÍA PÚ
18	14359749 LP	GARY ARTURO QUISPE GUERREROS	FE Y ALEGRIA YUNGUYO	92.91	segundo	DERECHO
19	9185364 LP	JHEHIEL DALILA CULI CEREZO	U. E. TUMUPASA B	88.93	primero	CONTADURÍA PÚ
20	13315320 LP	JAEI SENAYDA MAMANI RAMIREZ	RVDO. LEONCIO VICTOR ARAUZO	88.48	tercero	CONTADURÍA PÚ

Showing 11 to 20 of 280 entries

Previous 1 2 3 4 5 ... 28 Next

SISTEMA DE REGISTRO DE ADMISIÓN DE ESTUDIANTES DE LA MODALIDAD DE EXCELENCIA ACADÉMICA

En la sección de reportes el coordinador de excelencia podrá generar los reportes de los postulantes que cuenten con la resolución y los postulantes que aun no cuenten con la resolución:

INICIO > LISTADO DE ESTUDIANTES

POSTULANTES SIN CERTIFICADO VICERECTORAL

SELECCIONE UNA CARRERA: SELECCIONAR TODOS | FECHA INICIAL: dd/mm/aaaa | FECHA FINAL: dd/mm/aaaa | GENERAR PDF

Show 10 entries | Search:

#	CI	NOMBRE Y APELLIDOS	COLEGIO	PROMEDIO	POSICION	CARRERA
21	8295537 LP	Jorge Andres Miranda Illanes	Pedro Poveda	91.00	segundo	DERECHO
22	8295537 LP	Jorge Andres Miranda Illanes	Pedro Poveda	91.00	segundo	DERECHO
23	10091567 LP	JENNY GRISELDA VILLAZANTE CHACHAQUI	LIBERTADOR SIMÓN BOLIVAR	98.00	tercero	MEDICINA
24	10091567 LP	JENNY GRISELDA VILLAZANTE CHACHAQUI	LIBERTADOR SIMÓN BOLIVAR	98.00	tercero	MEDICINA
25	10091567 LP	JENNY GRISELDA VILLAZANTE CHACHAQUI	LIBERTADOR SIMÓN BOLIVAR	98.00	tercero	MEDICINA
26	14046221 LP	ARACELY TURPO CATARI	U.E. AYACUCHO	92.00	primero	CONTADURÍA PÚBL
27	8361655 LP	Juana elizabeth tapia cassa	U. E. Andrés de Santa Cruz	90.00	primero	CONTADURÍA PÚBL
28	8449644 LP	RAUL CONDORI LOPEZ	U. E. 12 DE OCUTUBRE	90.00	tercero	COMERCIO INTERN
29	8449644 LP	RAUL CONDORI LOPEZ	U. E. 12 DE OCUTUBRE	90.00	tercero	ARQUITECTURA
30	14782961 LP	Esther Lima Marca	U. E. San Vicente de Paul Fe y Alegria	92.00	primero	ECONOMÍA

Showing 21 to 30 of 259 entries | Previous 1 2 3 4 5 ... 26 Next

ANEXO

D

DOCUMENTACIÓN