

UNIVERSIDAD PÚBLICA DE EL ALTO

CARRERA INGENIERÍA DE SISTEMAS



PROYECTO DE GRADO

“SISTEMA DE INFORMACIÓN WEB PARA EL CONTROL Y SEGUIMIENTO ACADÉMICO”

CASO: INSTITUTO TÉCNICO SAN PABLO

Para Optar al Título de Licenciatura en Ingeniería de Sistemas

MENCIÓN: INFORMÁTICA Y COMUNICACIONES

Postulante: Mirna Tola Mendoza

Tutor Metodológico: Ing. Marisol Arguedas Balladares

Tutor Revisor: Ing. Milton Osvaldo Zurita Benito

Tutor Especialista: Ing. Grover Wilson Quisbert Ibañez

EL ALTO – BOLIVIA

2020

DEDICATORIA

El presente trabajo lo dedico a mis padres Alberto Tola y Flora Mendoza, quienes con su amor, paciencia y esfuerzo me han permitido llegar a cumplir un sueño más; gracias Dios por inculcar en mí el ejemplo de esfuerzo y valentía de no tener las adversidades porque Dios está siempre conmigo.

A mi niño Gael por ser la luz que ilumina mi camino; y a mi compañero de vida Reynaldo, que siempre ha estado conmigo brindándome su apoyo en todo momento.

AGRADECIMIENTOS

A Dios por ser mi guía y acompañarme en el transcurso de mi vida iluminando mi camino brindándome sabiduría para culminar con éxito mis metas propuestas.

A mis padres por ser el pilar fundamental y haberme apoyado incondicionalmente.

A mi tutora metodológico Ing. Marisol Arguedas Balladares por su conocimiento, apoyo, confianza tiempo y persistencia paciencia y motivación que me brindo.

A mi tutor especialista Ing. Grover Wilson Quisbert Ibañez por brindarme sus orientaciones, compartir su conocimiento y las sugerencias durante el desarrollo del presente proyecto.

A mi tutor revisor Ing. Milton Osvaldo Zurita Benito por su disponibilidad de tiempo, su acertada y orientada observaciones brindadas en la realización del proyecto.

Agradezco a la Universidad Pública de El Alto, por acogerme en sus aulas durante los años de estudio así también a la carrera Ingeniería de Sistemas y mis compañeras (os) de estudio por su apoyo incondicional.

RESUMEN

En la actualidad la información es un recurso de gran importancia para toda organización, pública y/o privada que trabaja con enormes volúmenes de datos. Las instituciones de carácter educativo no son la excepción, razón por lo cual, se debe prestar especial atención al uso de datos e información que se genera en las mismas.

El Instituto Técnico “San Pablo”, está experimentando un crecimiento significativo en la cantidad de estudiantes en sus distintas carreras.

Los procesos académicos de inscripción de estudiantes, registro de calificaciones, emisión de historial académico y record académico son manipulados y procesados de forma manual provocando retraso en el trabajo del personal encargado, es así que el principal objetivo del presente trabajo de grado es desarrollar un Sistema de Información Web para el Control y Seguimiento Académico solucionando los inconvenientes anteriores.

Para el análisis, diseño y desarrollo del presente proyecto se aplicó la metodología (UWE) basado en el proceso unificado UML. En cuanto a las herramientas tecnológicas se utilizó el lenguaje de programación PHP complementando con el framework Laravel, como gestor de base de datos MySQL, como servidor de aplicaciones Web se utilizó Apache XAMPP.

Para la calidad que es un aspecto muy importante tratado dentro de la metodología UWE en la fase del cierre, se llega a verificar utilizando parámetros de medición de la ISO 9126. Finalmente para tratar el análisis del costo del sistema desarrollado se utilizó COCOMO II (Constructive Cost Model) orientado a los puntos de función.

ABSTRACT

Today, information is a resource of great importance for any organization, public and / or private that works with huge volumes of data. Educational institutions are no exception, which is why special attention must be paid to the use of data and information generated in them.

The “San Pablo” Technical Institute is experiencing a significant growth in the number of students in its different careers.

The academic processes of student enrollment, record of grades, issuance of academic history and academic record are manipulated and processed manually causing delay in the work of the personnel in charge, so the main objective of this degree project is to develop a System Web Information for Academic Control and Monitoring solving the above problems.

For the analysis, design and development of this project, the methodology (UWE) based on the unified UML process was applied.

Regarding the technological tools, the PHP programming language was used, complementing the Laravel framework, as a MySQL database manager, and Apache XAMPP was used as a web application server.

For quality, which is a very important aspect treated within the UWE methodology in the closing phase, it is verified using measurement parameters of ISO 9126. Finally, to treat the cost analysis of the developed system, COCOMO II (Constructive Cost Model) oriented to function points.

ÍNDICE

1	MARCO PRELIMINAR	1
1.1	Introducción	1
1.2	Antecedentes	2
1.2.1	Antecedentes de la Institución.....	2
1.2.2	Antecedentes a fines al trabajo	3
1.3	Planteamiento del problema	4
1.3.1	Problema principal.....	4
1.3.2	Problemas secundarios	4
1.4	Objetivos	5
1.4.1	Objetivo general	5
1.4.2	Objetivos específicos.....	5
1.5	Justificación	6
1.5.1	Técnica.....	6
1.5.2	Económica.....	6
1.5.3	Social.....	6
1.6	Metodología	6
1.6.1	UWE-UML-based Web Engineering	7
1.6.2	Características.....	7
1.7	Herramientas	7

1.7.1	Lenguaje de Programación PHP	7
1.7.2	Gestor de Base de Datos MySQL	7
1.7.3	Servidor Web Apache.....	8
1.7.4	CSS.....	8
1.7.5	Framework Laravel.....	8
1.8	Límites y Alcances	9
1.8.1	Limites	9
1.8.2	Alcances.....	9
1.9	Aportes.....	10
2	MARCO TEÓRICO	11
2.1	Introducción	11
2.2	Conceptos básicos.....	11
2.2.1	Sistema	11
2.2.2	Sistema de información	12
2.2.3	Ciclo de vida de un sistema de información web.....	14
2.2.4	Control.....	15
2.2.5	Seguimiento Académico.....	16
2.2.6	Intranet	17
2.2.7	Internet	17
2.3	Ingeniería de Software.....	18

2.4	Etapas de la ingeniería del software	19
2.4.1	Análisis de requisitos	19
2.4.2	Especificación.....	19
2.4.3	Diseño y arquitectura.....	19
2.4.4	Programación	20
2.4.5	Prueba.....	20
2.4.6	Mantenimiento	20
2.5	Metodologías de desarrollo de Software.....	21
2.5.1	Metodologías tradicionales	21
2.5.2	Metodologías ágiles.....	22
2.6	Metodología Web	22
2.6.1	Metodología UWE	22
2.6.2	Ciclo de la Metodología UWE.....	25
2.7	Pruebas.....	30
2.7.1	Pruebas de Caja Blanca.....	30
2.7.2	Pruebas de Caja Negra	30
2.8	Arquitectura de Software.....	31
2.8.1	Patrón de diseño Model View Controller (MVC)	31
2.8.2	Definición e historia	31
2.8.3	Componentes	32

2.9	Herramientas	33
2.9.1	Gestor de Base de Datos MySQL	33
2.9.2	Workbench	34
2.9.3	Lenguaje de Programación PHP	34
2.9.4	Apache	35
2.9.5	Framework-Laravel.....	36
2.9.6	Bootstrap	38
2.9.7	HTML.....	38
2.9.8	CSS	39
2.9.9	JSON.....	39
2.9.10	JavaScript.....	40
2.9.11	JQuery.....	40
2.10	Métricas de Calidad	41
2.10.1	Calidad de software.....	41
2.10.2	Métricas de Calidad del Software	41
2.11	Análisis de costos COCOMO II.....	43
2.11.1	Atributo de los costes	45
2.11.2	Atributos de software.....	45
2.11.3	Atributos de hardware	46
2.11.4	Atributos de personal.....	46

2.11.5	Atributos de proyecto.....	46
2.12	Seguridad de la información	47
2.12.1	Estándar ISO/IEC 27000	47
2.12.2	Periodo de seguridad back-up.....	47
2.12.3	Sistema de Gestión de la Seguridad de la Información (SGSI)	47
3	MARCO APLICATIVO	51
3.1	Introducción	51
3.2	Análisis de la Situación actual.....	51
3.3	Obtención de requisitos	51
3.3.1	Definición de actores	52
3.3.2	Lista de Requerimientos del Sistema	53
3.3.3	Descripción de Funciones	56
3.4	Diseño del sistema.....	57
3.4.1	Diseño de los diagramas de casos de uso	57
3.4.2	Diagramas de casos de uso expandidos.....	59
3.5	Diagrama de clases	67
3.6	Modelo de Contenido.....	68
3.7	Diseño de navegación.....	70
3.7.1	Modelo de navegación: Administrador Académico.....	70
3.7.2	Modelo de navegación: Administrador Usuario	71

3.7.3	Modelo de navegación: Docente	71
3.7.4	Modelo de navegación: Estudiante.....	72
3.8	Diseño de presentación	72
3.8.1	Modelo de presentación: Login (inicio de sesión).....	72
3.8.2	Modelo de presentación: Inicio	73
3.8.3	Modelo de presentación: Usuario Administrador	73
3.8.4	Modelo de presentación: Usuario Académico	74
3.8.5	Modelo de presentación: Docente	74
3.8.6	Modelo de presentación: Estudiante	75
3.8.7	Implementación del sistema	75
3.8.8	Interfaz Inicio de Sesión	76
3.8.9	Módulos que integran el sistema	78
3.9	Pruebas de Software.....	80
3.9.1	Pruebas de Caja Blanca.....	80
3.9.2	Pruebas de Caja Negra	81
4	MÉTRICAS DE CALIDAD, ESTIMACIÓN DE COSTO Y SEGURIDAD	85
4.1	Introducción	85
4.2	Métricas de calidad	85
4.2.1	Funcionabilidad	85
4.2.2	Confiabilidad.....	89

4.2.3	Usabilidad.....	90
4.2.4	Mantenibilidad	92
4.2.5	Portabilidad	93
4.3	Estimación de costo de Software.....	93
4.3.1	Método de estimación COCOMO II.....	94
4.4	Sistema de gestión de seguridad de la información ISO-27002	96
4.4.1	Seguridad Lógica.....	96
4.4.2	Seguridad Física.....	97
4.4.3	Seguridad Organizativa	98
5	CONCLUSIONES Y RECOMENDACIONES.....	99
5.1	Conclusiones	99
5.2	Recomendaciones	100
	BIBLIOGRAFÍA	83

ÍNDICE DE TABLAS

Tabla 3.1 Obtención de requisitos.....	52
Tabla 3.2 Lista de Actores	53
Tabla 3.3 Requerimientos Funcionales.....	54
Tabla 3.4 Requisitos No Funcionales.....	55
Tabla 3.5 Especificaciones de los actores de Casos de Uso.....	56
Tabla 3.6 Descripción de Caso de Uso: Administrar Usuario	59
Tabla 3.7 Descripción de Caso de Uso: Inscripción de Estudiantes	61
Tabla 3.8 Descripción de Caso de Uso: Asignación de Materias.....	62
Tabla 3.9 Descripción de Caso de Uso: Registrar Calificaciones de Estudiantes.	64
Tabla 3.10 Descripción de Caso de Uso: Reportes	65
Tabla 3.11 Caso de Prueba Ingreso al Sistema Usuario y Pantalla.....	82
Tabla 4.1 Parámetros de medición	86
Tabla 4.2 Parámetros de medición	86
Tabla 4.3 Parámetros de medición	87
Tabla 4.4 Parámetros de medición	91
Tabla 4.5 Parámetros de medición	91
Tabla 4.6 Valores para determinar la mantenibilidad	92
Tabla 4.7 Coeficientes del modelo COCOMO II.....	94
Tabla 4.8 Ecuaciones del Modelo COCOMO II.....	94
Tabla 4.9 Calculo de los atributos FAE	95

ÍNDICE DE FIGURAS

Figura 2.1 Diagrama de Caso de Uso	26
Figura 2.2 Grafico de Modelos de Caso de Uso	27
Figura 2.3 Gráfico del Diagrama de Contenido de la Metodología UWE	28
Figura 2.4 Gráfico del Diagrama de Navegación de la metodología UWE.....	29
Figura 2.5 Gráfico del Diagrama de Presentación de la Metodología UWE.....	30
Figura 2.6: Interacción entre Modelo, Vista y Controlador	32
Figura 2.7 Enrutamiento básico de Laravel.....	37
Figura 2.8 Características principales para determinar la calidad de software	42
Figura 2.9 Análisis de costo Cocomo II.....	45
Figura 2.10 Estructura de la Norma ISO 27002 (Dominio de Control)	47
Figura 3.1 Diagrama de Caso de Uso General	58
Figura 3.2 Caso de uso administrar usuarios.....	59
Figura 3.3 Caso de uso inscripción de estudiantes.....	60
Figura 3.4 Caso de uso asignación de materias	62
Figura 3.5 Caso de uso registrar calificaciones de estudiantes	63
Figura 3.6 Caso de uso reportes.....	65
Figura 3.7 Diagrama de Clases.....	67
Figura 3.8 Modelo de Datos Relacional	68
Figura 3.9 Diagrama de actividades Gestión de usuarios.....	69
Figura 3.10 Diagrama de actividades Gestión de notas.....	69
Figura 3.11 Modelo de navegación: Administrador Académico	70
Figura 3.12 Modelo de navegación: Administrador Usuario.....	71
Figura 3.13 Modelo de navegación: Docente.....	71

Figura 3.14 Modelo de navegación: Estudiante	72
Figura 3.15 Modelo de presentación: Login (inicio de sesión)	72
Figura 3.16 Modelo de presentación: Inicio	73
Figura 3.17 Modelo de presentación: Usuario	73
Figura 3.18 Modelo de presentación: Usuario Académico.....	74
Figura 3.19 Modelo de presentación: Docente	74
Figura 3.20 Modelo de presentación: Estudiante.....	75
Figura 3.21 Inicio de sesión	76
Figura 3.22 Menú principal Administrador.....	76
Figura 3.23 Menú principal Académico	77
Figura 3.24 Menú principal Docente	77
Figura 3.25 Menú principal Estudiante.....	78
Figura 3.26 Menú de parámetros Administrador Académico	78
Figura 3.27 Pantalla lista de Estudiantes	79
Figura 3.28 Pantalla lista de Docente	79
Figura 3.29 Caja blanca – Técnica Complejidad Dicromática.....	80

CAPITULO I

MARCO PRELIMINAR

1 MARCO PRELIMINAR

1.1 Introducción

Las nuevas tendencias tecnológicas y comunicacionales asociadas a la información, hace que nuestra sociedad este cada día más conectada electrónicamente, creando condiciones de gran competencia y dinamismo en todos los ámbitos del desarrollo del país. Esto condiciona a las Instituciones Educativas Superiores a mantener una continua actualización y un permanente mejoramiento de la calidad de su oferta académica de carreras y de igual forma una constante adecuación de estas y de los procesos de aprendizaje-enseñanza.

Las labores que tradicionalmente eran realizadas manualmente ahora han iniciado con la automatización¹ de los procesos de las organizaciones por medio de sistemas de información. Por estas características, los sistemas educativos superiores, se encuentran inmersos en un proceso de cambios, enmarcados en el conjunto de transformaciones sociales propiciadas por la innovación tecnológica y, sobre todo por el desarrollo de las tecnologías de información y comunicación, por los cambios en las relaciones sociales y por una nueva concepción de las relaciones tecnología-sociedad que determina las relaciones tecnología-educación.

Por tal motivo el Instituto Técnico San Pablo tiene la necesidad de desarrollar un sistema de información académico, que facilite las funciones que desempeñan los administrativos académicos de la institución y la obtención de información oportuna. De tal manera que el presente proyecto de grado se orienta en solucionar los problemas de la institución como ser pérdida de tiempo en la búsqueda de información académica de los estudiantes.

Aplicando la metodología UWE, el cual está pensado para adaptarse a cualquier tipo de proyecto, se pretende modelar el sistema según las necesidades de los

¹ Automatización: Realizar procesos o trabajos utilizando poco o nada la mano del hombre. [Viviana Asensi Artiga]

usuarios, para el desarrollo del sistema se utilizó PHP y como gestor de base de datos MySql.

1.2 Antecedentes

1.2.1 Antecedentes de la Institución

El Instituto Técnico “San Pablo” fue fundado un 6 de marzo de 1996, en honor a las efemérides de la ciudad de El Alto. Desde entonces el Instituto viene trabajando intensivamente en la capacitación y formación de profesionales de excelencia en los campos Técnicos y Administrativos.

➤ **Sistema de Gestión de Calidad ISO 9001:2008**

Con la entrega de los certificados de calidad IBNORCA Nro. EC-183/11, IQNet AE-QS-4327 e IRAM Nro. 9000-4327, el Instituto Técnico San Pablo es el PRIMER y único instituto técnico en Bolivia que trabaja bajo un Sistema de Gestión de Calidad ISO 9001:2008

➤ **Política de calidad**

En el Instituto Técnico San Pablo se forman profesionales técnicos con valores cristianos, competencias laborales y con cultura emprendedora, para este fin se cuenta con programas aprobados, infraestructura necesaria, recursos humanos actualizados y capacitados, y un sistema de gestión de calidad los cuales evidencian, el compromiso de la Institución para la mejora continua y principalmente para el cumplimiento tanto de requisitos de nuestros clientes, como los requisitos legales.

En la actualidad, la institución se rige al reglamento del ministerio de educación, con resolución suprema Ministerial 5/1997 – 095/2012.

- **Misión:** El Instituto Técnico “San Pablo” tiene como misión formar profesionales técnicos, formación concebida de manera integral para la vida y el trabajo, incorporando valores cristianos, competencias laborales, y generando una cultura emprendedora a través de un desarrollo económico sostenible con infraestructura, equipamiento, tecnología de última

generación y el compromiso de nuestros recursos humanos con el servicio, para satisfacer las necesidades del estudiante.

- **Visión:** Ser una institución rentable reconocida, sólida y confiable para la sociedad con presencia nacional e internacional. Brindando ventajas competitivas con nuevos servicios, trabajo social, innovaciones educativas y una estructura organizacional institucionalizada, respondiendo a las necesidades del cliente en diferentes áreas del bienestar personal.

1.2.2 Antecedentes a fines al trabajo

- (Campos, Arque, Mamani, Machaca, & ortega ; Semestre III) ANÁLISIS Y DISEÑO DE UN SISTEMA DE INFORMACIÓN PARA EL SISTEMA MODULAR PARA EL ÁREA DE SECRETARIA ACADÉMICA, se planteó Analizar y diseñar un programa eficaz para el buen funcionamiento del manejo de notas en el área de secretaria académica, para mejorar el funcionamiento en la administración de secretaria, utilizando las siguientes herramientas como SQL server, visual estudio ASP.NET, y para la metodología del Software utiliza RUP. (Instituto superior de educación publico Ayaviri-Perú).
- (Juela & Susana, 2012) DESARROLLO DE UN SISTEMA AUTOMATIZADO PARA LA INSCRIPCIÓN DE ALUMNOS VÍA INTERNET, DEL INSTITUTO TECNOLÓGICO PRIMERO DE MAYO el cual fue desarrollo para otorgar un servicio de eficiencia en benéfico a la institución ya que el sistema será automatizado respecto a las inscripciones de los estudiantes por medio de internet, que fue realizado en la metodología UWE con herramientas de software libre. (Universidad Politécnica Salesiana; Ecuador).
- (Delia Canaza Arias ; 2011) SISTEMA DE CONTROL Y SEGUIMIENTO ACADÉMICO PARA EL INSTITUTO TECNOLÓGICO PUERTO DE MEJILLONES su principal objetivo fue desarrollar un sistema de información que permita un control y seguimiento académico oportuno y confiable de la información de los estudiantes del instituto Puerto de Mejillones, coadyuvando al cumplimiento de sus requerimientos administrativos y

académicos, con el lenguaje de programación PHP, con el Gestor de Base de Datos MySQL, sin embargo, esta realizado con la metodología R.U.P (Universidad Pública de El Alto; El Alto – Bolivia).

- (Edwin Mamani Viscarra, 2016] SISTEMA DE INFORMACIÓN DE SEGUIMIENTO ACADÉMICO VÍA WEB PARA DEPARTAMENTO DE IDIOMAS DE LA UNIVERSIDAD PÚBLICA DE EL ALTO el proyecto planteado desarrollo e implemento un sistema de seguimiento de información académica, aplicando la arquitectura cliente /servidor que coadyuve con la administración y la optimización de la información, en el departamento de idiomas dependiente de la carrera de lingüística e idiomas de la universidad pública de el alto. Usando las herramientas de programación PHP, MySQL, MagicDraw, JQUERY y Framework Codeigniter. Realizado en la metodología Scrum y Uwe. (Universidad Pública de El Alto; El Alto – Bolivia).

1.3 Planteamiento del problema

1.3.1 Problema principal

El Instituto Técnico San Pablo, presenta problemas en la búsqueda de información académica de estudiantes, docentes, materias, historial académico y calificaciones a causa del proceso manual que realiza, ocasionando que mucha información académica sea vulnerable a alteraciones y extravíos.

1.3.2 Problemas secundarios

- Los procesos de toda información académica son realizados de forma semiautomática, provocando retraso y duplicidad de la información en el registro de inscripción.
- La manipulación y registro de calificaciones se realiza de forma manual en documentos Excel, ocasionando que la documentación e información sea desorganizada y propensa al extravió o empapelado.

- La asignación de materias se realiza de forma manual en listas que serán ordenadas posteriormente según el plan de estudios, causando demora ya que estos datos deben ser revisados minuciosamente posteriormente.
- El historial académico del estudiante debe ser llenado a mano ocasionando que sea un proceso exhaustivo y sea propenso a generar información errónea en el proceso.

¿De qué manera el Sistema de Información Web para el Control y Seguimiento Académico podrá optimizar los procesos tal que se genere información precisa y oportuna dentro de la Institución?

1.4 Objetivos

1.4.1 Objetivo general

Desarrollar un Sistema de Información Web para el Control y Seguimiento Académico con el fin de coadyuvar en la generación de información confiable y oportuna para la correcta toma de decisiones.

1.4.2 Objetivos específicos

- Realizar el análisis del flujo de información académica para acelerar los procesos de inscripción.
- Desarrollar un módulo de calificaciones de los estudiantes de acuerdo al requerimiento académico.
- Desarrollar un módulo de asignación de materias y docentes, en función a la cantidad de registrados para su respectivo control académico.
- Generar y mostrar el historial académico de las notas de estudiantes en el tiempo oportuno para acelerar los procesos y toma de decisiones.

1.5 Justificación

1.5.1 Técnica

La institución dispone de recursos para el desarrollo del Sistema de Información Web para el control y Seguimiento Académico, el desarrollo del presente proyecto de grado permitirá a la institución contar con una nueva herramienta de información académica, aplicando métodos, metodologías y técnicas para su desarrollo.

1.5.2 Económica

Con el desarrollo del sistema de información, se reducirá los costos de materiales de escritorio e insumos necesarios utilizados para el procesamiento de la información académica. Trayendo ventajas en maximizar los ingresos.

1.5.3 Social

El personal académico, estudiantes y docentes, de la institución son beneficiados con el Sistema de Información Web para el Control y Seguimiento Académico, prestando una buena imagen del Instituto Técnico San Pablo, el presente sistema facilita el trabajo en el instituto brindando información rápida, oportuna y actualizada de los estudiantes, docentes y procesos administrativos.

1.6 Metodología

Para el desarrollo del presente proyecto se utiliza la metodología UWE², es una de las metodologías ágiles más utilizadas por los desarrolladores de software y empresas.

² UWE (*based web engineering*) Basado en Ingeniería Web

1.6.1 UWE-UML-based Web Engineering

UWE es un método de ingeniería del software, para el desarrollo de aplicaciones web basado en UML cualquier tipo de diagrama de UML puede ser usado, porque UWE es una extensión de UML. (Williams Maximillians, 2010)

1.6.2 Características

- La metodología UWE define vistas especiales representadas gráficamente por diagramas en UML, tales como el modelo de navegación y el modelo de presentación.
- Los diagramas se pueden adaptar como mecanismos de extensión basados en estereotipos que proporciona UML. estos mecanismos de extensión son los que UWE utiliza para definir estereotipos que son los que finalmente se utilizaran en las vistas especiales para el modelo de aplicaciones Web.
- La ventaja de utilizar los perfiles de UML es todas las herramientas CASE de UML los reconocen.
- Los modelos deben ser fácilmente adaptadas al cambio en cualquier etapa del desarrollo.

1.7 Herramientas

1.7.1 Lenguaje de Programación PHP

Es un lenguaje de código abierto muy popular, adecuado para desarrollo web y que puede ser incrustado en HTML. Es popular porque un gran número de páginas y portales web están creadas con PHP. Código abierto significa que es de uso libre y gratuito para todos los programadores que quieran usarlo. (Beati Hernan, 2011, pág. 2)

1.7.2 Gestor de Base de Datos MySQL

Es un sistema de open souce para gestión de base de datos relacionales, que brinda un excelente rendimiento, flexibilidad y velocidad.

Junto a su herramienta Workbench permite la completa administración tanto de registros como de usuarios, permisos y conexiones. Debido a su estabilidad, seguridad y popularidad, elegir MySQL para los proyectos, brinda un alto grado profesionalidad y potencia. (Mussa Yaser, 2017)

1.7.3 Servidor Web Apache

“Es un programa especialmente diseñado para transferir datos de hipertexto, es decir, páginas web con todos sus elementos (textos, widgets, banners, etc).Estos servidores web utilizan el protocolo Http.

Es un servidor web HTTP de código abierto para la creación de páginas y servicios web. Es un servidor multiplataforma, una de las ventajas más grandes de Apache es que es un servidor web multiplataforma, gratuito muy robusto y que descarta por su seguridad y rendimiento.” (McCool Robert, 1995)

1.7.4 CSS

Es un lenguaje de hojas de estilos creado para controlar el aspecto o presentación de los documentos electrónicos definidos con HTML y XHTML. CSS es la mejor forma de separar los contenidos y su presentación es imprescindible para crear páginas web complejas. Separar la definición de los contenidos y la definición de su aspecto presenta numerosas ventajas, ya que obliga a crear documentos HTML/XHTML bien definidos y con significado completo también llamados “documentos semánticos”. (Introducción a CSS, 2008)

1.7.5 Framework Laravel

Laravel es un marco de aplicación web con sintaxis expresiva y elegante. Ya hemos sentido las bases, liberándote para crear sin dudar las cosas pequeñas. Creemos que el desarrollo debe ser una experiencia agradable y creativa para que sea verdaderamente enriquecedora. Laravel busca eliminar el sufrimiento del desarrollo facilitando las tareas comunes utilizadas en la mayoría de los proyectos web, como

la autenticación, enrutamiento, sesiones y almacenamiento en cache. (Richos, 2018)

1.8 Límites y Alcances

1.8.1 Límites

El proyecto de grado contempla como límites los siguientes puntos:

- El sistema solo gestiona el flujo de la información académica de la institución y no así la información administrativa de la institución.
- El sistema no provee de notificaciones, calendarios recordatorios automatizados de los procesos, fechas o actividades que realiza la institución.
- El sistema no presta el servicio para cobros de inscripción y recibos de pagos que realiza el instituto.
- El sistema no almacena información digitalizada académica de los estudiantes.
- El sistema no realiza homologaciones (convalidaciones de materias) de instituciones académicas externas al instituto.

1.8.2 Alcances

Los alcances del Sistema de manera general contemplan lo siguiente:

- **Registro de estudiantes.-** El sistema realiza el registro de estudiantes nuevos y antiguos con todos los datos personales que solicita la institución.
- **Registro de docentes.-** El sistema realiza el registro y asignación de materias de cada docente con los datos correspondientes.
- **Boletines de calificación.-** El sistema genera de manera automatizada los boletines de calificaciones de los estudiantes.
- **Historial académico.-** El sistema genera el historial académico de cada estudiante.
- **Record académico.-** El sistema obtiene el record académico de cada estudiante.

- **Centralizadores de calificación.-** El sistema centraliza de forma bimestral todas las calificaciones de cada carrera, ya que estas calificaciones después de una revisión exhaustiva son enviadas al Ministerio de Educación.
- **Registro plan de estudios.-** El sistema asigna materias de manera fácil y, confiable y rápida a docentes y estudiantes de la institución.
- **Registro de carreras.-** El sistema registra de nuevos y antiguos estudiantes a las carreras con fines de actualizar datos al Instituto Técnico San Pablo.
- **Registro de evaluación.-** Los docentes realizan el registro de notas de los estudiantes de manera remota, en un ambiente intranet proveyendo comodidad y facilidad de este proceso.
- **Consultas y reportes.-** El sistema despliega reportes de acuerdo a las consultas realizadas por el usuario del sistema considerando un comportamiento estadístico.

1.9 Aportes

El sistema web tendrá un gran aporte para el Control y Seguimiento Académico es un aporte para el Instituto Técnico “San Pablo”, puesto que permitirá tener información centralizada en una sola base de datos que posibilitará tener un incremento de funcionalidad y desempeño dentro de la Institución mediante un adecuado flujo de información.

Para todo el proceso de desarrollo del sistema se emplea la metodología UWE, una herramienta de modelado basada en UML para aplicaciones web y además se emplea un patrón de desarrollo muy conocido como MVC (Modelo – Vista - Controlador) implementado en muchos Frameworks en los lenguajes de programación más populares ya que ayuda bastante a los desarrolladores así poder darle una escalabilidad de implementación de más módulos.

CAPITULO II

MARCO TEÓRICO

2 MARCO TEÓRICO

2.1 Introducción

En todo desarrollo de sistemas de software es de suma importancia el seguir alguna especificación que permita a los desarrolladores el tener una disciplina que haga que todas las etapas del desarrollo del sistema, desde la pesquisa inicial de requerimientos hasta las pruebas finales del sistema, sean no solo más coherentes sino también más formales.

El desarrollo de software que este proyecto propone, al ser una herramienta que pretende tener aplicaciones dentro del contexto de un problema real, tiene que seguir un proceso de análisis y diseño que proporcione los cimientos bajo los cuales se va a desarrollar la aplicación conjuntamente. Es por esto que en este capítulo de detallan los procesos de ingeniería de software, análisis y diseño que se involucran para el desarrollo de una aplicación de software que puede utilizarse como auxiliar al tratamiento del trastorno de lateralidad y ubicación espacial.

2.2 Conceptos básicos

2.2.1 Sistema

En informática se entiende por sistema a un conjunto de datos ordenados conforme a una serie de instrucciones y algoritmos, que permiten su ubicación y recuperación rápida. (Marina Estela Raffino, 2020)

Un sistema es módulo ordenado de elementos que se encuentran interrelacionados y que interactúan entre sí. El concepto se utiliza tanto para definir a un conjunto de conceptos como a objetos reales dotados de organización también puede mencionarse la noción del informático, muy común en las sociedades modernas. Este tipo de sistemas denominan al conjunto de hardware, software y soporte humano que forman parte de una empresa u organización. Incluyen ordenadores

con los programas necesarios para procesar datos y las personas encargadas de su manejo. (Roja Daniela, 2018)

Un sistema es un conjunto de partes o elementos organizados y relacionados, que interactúan entre sí para lograr un objetivo.

2.2.2 Sistema de información

Cuando se habla de un sistema de información (SI) se refiere a un conjunto ordenado de mecanismos que tienen como fin la administración de datos y de información, de manera que puedan ser recuperados y procesados fácil y rápidamente. (Maria Estela Raffino, 2020)

Básicamente, un sistema informático, más conocida en el ámbito de la tecnología por sus siglas "SI" es una técnica que permite el almacenamiento y el proceso de información, para lo cual se vale de un grupo de elementos que se relacionan entre sí. Estos elementos no son otros que el hardware, el software y finalmente el usuario, quien es el que requiere de la información procesada, y quien es también el que en definitiva tiene el control total de lo que sucede en el sistema. (TECNOLOGIA&INFORMATICA, 2019)

Existen varias definiciones sobre Sistemas de información, una de las más acertadas indica que un sistema de información es un conjunto de elementos que interactúan entre sí con el fin de apoyar las necesidades de una empresa o negocio. Teniendo en cuenta el equipo computacional necesario para que el sistema de información pueda operar y el recurso humano que interactúa con el Sistema de Información, el cual está formado por las `personas que utilizan el sistema.

2.2.2.1 Actividades Básicas de un Sistema de Información

Entrada de información

Es el proceso mediante el cual el sistema de Información toma los datos que requiere para procesar la información. Las entradas pueden ser manuales o automáticas. Las manuales son aquellas que se proporcionan en forma directa por el usuario, mientras que las automáticas son datos o información que provienen o

son tomados de otros sistemas o módulos. Esto último se denomina interfaces automática. Las unidades típicas de entrada de datos a las computadoras son las terminales, las cintas magnéticas, las unidades de diskette, los códigos de barras, los escáneres, la voz, los monitores sensibles al tacto, el teclado y el mouse, entre otras.

Almacenamiento de Información

El almacenamiento es una de las actividades o capacidades más importantes que tiene una computadora, ya que a través de esta propiedad el sistema puede recordar la información guardada en la sección o proceso anterior. Esta información suele ser almacenada en estructuras de información denominadas archivos. La unidad típica de almacenamiento son los discos magnéticos o discos duros. Los discos duros flexibles y los discos compactos (CD-ROM).

Procesamiento de información

Es la capacidad de Sistema de Información para efectuar cálculos de acuerdo con una secuencia de operaciones preestablecida. Estos cálculos pueden efectuarse con datos introducidos recientemente en el sistema o bien con datos que están almacenados. Esta característica de los sistemas permite la transformación de datos fuente en información que puede ser utilizada para la toma de decisiones, lo que hace posible, entre otras cosas, que un tomador de decisiones genere una proyección financiera a partir de los datos que contiene un estado de resultados o un balance general de un año base.

Salida de información

La salida es la capacidad de un Sistema de Información para sacar la información procesada o bien datos de entrada al exterior. Las unidades típicas de salida son las impresoras y terminales. Es importante aclarar que la salida de un Sistema de Información puede constituir la entrada a otro Sistema de Información o modulo. En este caso, también existe una interfaz automática de salida.

2.2.3 Ciclo de vida de un sistema de información web.

Se utilizará en el sistema web modelo incremental Evolutivo. En este modelo los requerimientos del usuario pueden cambiar en cualquier momento. Investigación preliminar, identificación de fortalezas y amenazas.

La práctica nos demuestra que obtener todos los requerimientos al comienzo del proyecto es muy difícil no solo por la dificultad del usuario transmitir su idea, sino porque los requerimientos evolucionan durante el desarrollo y de esta manera, surgen nuevos requerimientos a cumplir

Existen pautas básicas para el desarrollo de un sistema de información para una organización:

- **Codificación:** Con el algoritmo ya diseñado, se procede a su reescritura en un lenguaje de programación establecido (programación) en la etapa anterior, es decir, en códigos que la máquina pueda interpretar y ejecutar.
- **Conocimiento de la organización:** analizar y conocer todos los sistemas que forman parte de la organización, así como los futuros usuarios del sistema de información. En las empresas (fin de lucro presente), se analiza el proceso de negocio y los procesos transaccionales a los que dará soporte el Sistema de Información.
- **Determinar las necesidades:** este proceso también se denomina elicitación de requerimientos. En el mismo, se procede identificar a través de algún método de recolección de información (el que más se ajuste a cada caso) la información relevante para el sistema de información que se propondrá.
- **Diagnóstico:** en este paso se elabora un informe resaltando los aspectos positivos y negativos de la organización. Este informe formará parte de la propuesta del sistema de información y, también, será tomado en cuenta a la hora del diseño.
- **Diseño del sistema:** una vez aprobado el proyecto, se comienza con la elaboración del diseño lógico del sistema de información; la misma incluye: el diseño del flujo de la información dentro del sistema, los procesos que se realizarán dentro del sistema, el diccionario de datos, los reportes de salida,

etc. En este paso es importante para seleccionar la plataforma donde se apoyará el Sistema de Información y el lenguaje de programación a utilizar.

- **Identificación de problemas y oportunidades:** el segundo paso es relevar las situaciones que tiene la organización y de las cuales se puede sacar una ventaja competitiva (Por ejemplo: una empresa con un personal capacitado en manejo informático reduce el costo de capacitación de los usuarios), así como las situaciones desventajosas o limitaciones que hay que sortear o que tomar en cuenta (Por ejemplo: el edificio de una empresa que cuenta con un espacio muy reducido y no permitirá instalar más de dos computadoras).
- **Implementación:** este paso consta de todas las actividades requeridas para la instalación de los equipos informáticos, redes y la instalación de la aplicación (programa) generada en la etapa de Codificación.
- **Mantenimiento:** proceso de retroalimentación, a través del cual se puede solicitar la corrección, el mejoramiento o la adaptación del sistema de información ya creado a otro entorno de trabajo o plataforma. Este paso incluye el soporte técnico acordado anteriormente.
- **Propuesta:** contando ya con toda la información necesaria acerca de la organización, es posible elaborar una propuesta formal dirigida hacia la organización donde se detalle: el presupuesto, la relación costo-beneficio y la presentación del proyecto de desarrollo del sistema de información.” (María Estela Raffino, 2015)

2.2.4 Control

Es el proceso de verificar el desempeño de distintas áreas o funciones de una organización. Usualmente implica una comparación entre un rendimiento observado, para verificar si están cumpliendo los objetivos de forma eficiente y eficaz.

La función de control se relaciona con la función de planificación, porque el control busca que el desempeño se ajuste a los planes. El proceso administrativo, desde el punto de vista tradicional, es un proceso circular que se retroalimenta. Es por ello que durante la gestión el control permite tomar medidas correctivas.

(Concepto de Control, 2018)

El control es una etapa primordial en la administración, pues, aunque una empresa cuente con magníficos planes, una estructura organizacional adecuada y una dirección eficiente, el ejecutivo no podrá verificar cual es la situación real de la organización si no existe un mecanismo que se cerciore e informe si los hechos van de acuerdo con los objetivos. (Elibeth Yuri, 2020)

El concepto de control es muy general y puede ser utilizado en el contexto organizacional para evaluar el desempeño general frente a un plan estratégico.

2.2.5 Seguimiento Académico

El proceso de seguimiento académico establece la definición de los resultados de las actividades en términos de asistencia de los estudiantes, calificaciones, pertinencia de la información recibida, promoción y recuperación de problemas de aprendizaje.

El seguimiento que se hace a los resultados académicos de los estudiantes es aislado e individual, y no se generan acciones remediales para el logro de los objetivos. El cuerpo docente hace un seguimiento periódico y sistemático al desempeño académico de los estudiantes para diseñar acciones de apoyo a los mismos. (PEI Institucion Educativa La Leona, 1954)

El seguimiento académico es una estrategia que permite implementar acciones tutoriales, para aprovechar mejor las habilidades, capacidades y competencias de estudiantes.

El llenado del formato de seguimiento académico permite al tutor identificar, a través atención personal u uso de la información contenida en la plataforma de autoservicios, para guiar en el diseño de su ruta crítica de forma periódica, lo cual realizara analizando el mapa curricular de su unidad académica para detectar problemas de rezago y bajo rendimiento académico. (Matthew Thomas, 2020)

2.2.6 Intranet

Una definición con enfoque técnico, sostiene que "intranet es un conjunto de Sitios Web que están instalados en la red interna de una institución o empresa y que permiten mostrar datos o documentos a cualquiera de los computadores conectados a ella".

Intranet es una red de ordenadores privados que utiliza tecnología Internet para compartir de forma segura cualquier información o programa del sistema operativo para evitar que cualquier usuario de Internet pueda ingresar. En la arquitectura de las Intranets se dividen el cliente y el servidor. El software cliente puede ser cualquier computadora local (servidor web o PC), mientras que el software servidor se ejecuta en una Intranet anfitriona. No es necesario que estos dos software, el cliente y el servidor, sean ejecutados en el mismo sistema operativo. Podría proporcionar una comunicación privada y exitosa en una organización.

Tiene como función principal proveer lógica de negocios para aplicaciones de captura, informes y consultas con el fin de facilitar la producción de dichos grupos de trabajo; es también un importante medio de difusión de información interna a nivel de grupo de trabajo. Las redes internas corporativas son potentes herramientas que permiten divulgar información de la compañía a los empleados con efectividad, consiguiendo que estos estén permanentemente informados con las últimas novedades y datos de la organización. (Ryan Bernard, 2010)

2.2.7 Internet

Internet proviene de "*interconnetednetworks*" ("redes interconectadas"): básicamente se trata de millones de computadoras conectadas entre sí en una red mundial.

Internet es una red de redes que permite la interconexión descentralizada de computadoras a través de un conjunto de protocolos denominados TCP/IP. Tuvo sus orígenes en 1969, cuando una agencia del Departamento de Defensa de los Estados Unidos comenzó a buscar alternativas ante una eventual guerra atómica

que pudiera incomunicar a las personas. Tres años más tarde se realizó la primera demostración pública del sistema ideado, gracias a que tres universidades de California y una de Utah lograron establecer una conexión conocida como ARPANET (Advanced Research Projects Agency Network).

De esta manera, Internet sirve de enlace entre redes más pequeñas y permite ampliar su cobertura al hacerlas parte de una "red global". Esta red global tiene la característica de utilizar un lenguaje común que garantiza la intercomunicación de los diferentes participantes; este lenguaje común o protocolo (un protocolo es el lenguaje que utilizan las computadoras al compartir recursos) se conoce como TCP/IP.

El desarrollo de Internet ha superado ampliamente cualquier previsión y constituyó una verdadera revolución en la sociedad moderna. El sistema se transformó en un pilar de las comunicaciones, el entretenimiento y el comercio en todos los rincones del planeta. (Enrique Herrera Perez, 2003)

2.3 Ingeniería de Software

La ingeniería del software es una disciplina o área de la Informativa o Ciencias de Computación, que ofrece métodos y técnicas para desarrollar y mantener software de calidad que resuelven problemas de todo tipo. (Pressman, 2003)

- Las herramientas de la ingeniería de software proporcionan un enfoque automático o semiautomático para el proceso y para los métodos.
- Los métodos de la ingeniería del software indican «cómo» construir técnicamente el software.
- El proceso es el fundamento de la ingeniería del software, es la unión que mantiene juntas las capas de tecnología y que permite un desarrollo racional y oportuno.
- Cualquier enfoque de ingeniería del software debe apoyarse sobre un compromiso de organización de calidad.

2.4 Etapas de la ingeniería del software

La ingeniería de software requiere llevar a cabo numerosas tareas, dentro de distintas etapas, las principales son las siguientes: análisis de requisitos, especificación, diseño y arquitectura, programación, prueba y mantenimiento.

2.4.1 Análisis de requisitos

Extraer los requisitos de un producto software es la primera etapa para crearlo. Mientras que los clientes piensan que ellos saben lo que el software tiene que hacer, se requiere habilidad y experiencia en la ingeniería del software para reconocer requisitos incompletos, ambiguos o contradictorios. El resultado del análisis de requisitos con el cliente se plasma en el documento Especificación de Requisitos. Asimismo, se define un diagrama de entidad/relación, en el que se plasman las principales entidades que participarán en el desarrollo de software.

La captura, análisis y especificación de requisitos (incluso pruebas de ellos), es una parte crucial; de esta etapa depende en gran medida el logro de los objetivos finales. Se han ideado modelos y diversos procesos de trabajo para estos fines. Aunque aún no está formalizada, se habla de la Ingeniería de Requisitos.

2.4.2 Especificación

Es la tarea de escribir detalladamente el software a ser desarrollado, en una forma Matemáticamente rigurosa. En la realidad, la mayoría de las buenas especificaciones han sido escritas para entender y afinar aplicaciones que ya estaban desarrolladas. Las especificaciones son más importantes para las interfaces externas, que deben permanecer estables.

2.4.3 Diseño y arquitectura

Se refiere a determinar cómo funcionará el software de forma general sin entrar en detalles.

Consisten en incorporar consideraciones de la implementación tecnológica, como el

Hardware, la red, etc. Se definen los casos de uso para cubrir las funciones que realizará el sistema, y se transformarán las entidades definidas en el análisis de requisitos en clases de diseño, obteniendo un modelo cercano a la programación orientada a objetos.

2.4.4 Programación

Reducir un diseño a código puede ser la parte más obvia del trabajo de ingeniería del

Software, pero no necesariamente es la que demanda mayor trabajo ni la más complicada. La complejidad y la duración de esta etapa está íntimamente relacionada al o a los lenguajes de programación utilizados, así como al diseño previamente realizado.

2.4.5 Prueba

Consiste en comprobar que el software realice correctamente las tareas indicadas en la especificación del problema. Una técnica de prueba es probar por separado cada módulo del Software y luego probarlo de forma integral, para así llegar al objetivo. Se considera una buena práctica que las pruebas sean efectuadas por alguien distinto al desarrollador que la programó.

2.4.6 Mantenimiento

Mantener y mejorar el software para solventar errores descubiertos y tratar con nuevos requisitos. El mantenimiento puede ser de cuatro tipos: perfectivo (mejorar la calidad interna de los sistemas), evolutivo (incorporaciones, modificaciones y eliminaciones necesarias en un producto software para cubrir la expansión o cambio en las necesidades del usuario), adaptativo (modificaciones que afectan a los entornos en los que el sistema opera, por ejemplo, cambios de configuración del hardware, software de base, gestores de base de datos, comunicaciones) y correctivo (corrección de errores).

Para cada una de las fases en que hemos descompuesto el ciclo de vida de un sistema de información se han propuesto multitud de prácticas útiles, entendiendo por prácticas aquellos conceptos, principios, métodos y herramientas que facilitan la consecución de los objetivos de cada etapa. (Berzal, 2020)

2.5 Metodologías de desarrollo de Software

Las Metodologías de Desarrollo de Software surgen ante la necesidad de utilizar una serie de procedimientos, técnicas, herramientas y soporte documental a la hora de desarrollar un producto de Software.

Dichas metodologías pretenden guiar a los desarrolladores al crear un nuevo software, pero los requisitos de un software a otro son variados y cambiantes, que ha dado lugar a que exista una gran variedad de metodologías para la creación del software.

Según la filosofía de desarrollo se pueden clasificar las metodologías en dos grupos. Las metodologías tradicionales, que se basan en una fuerte planificación durante todo el desarrollo, y las metodologías ágiles, en las que el desarrollo de software es incremental, cooperativo, sencillo y adaptado.

2.5.1 Metodologías tradicionales

Las metodologías tradicionales son denominadas, a veces, de forma peyorativa, como metodologías pesadas. Centran su atención en llevar una documentación exhaustiva de todo el proyecto y en cumplir con un plan de proyecto, definido todo esto, en la fase inicial del desarrollo del proyecto.

Otra de las características importantes dentro de este enfoque, son los altos costes al implementar un cambio y la falta de flexibilidad en proyectos donde el entorno es volátil.

2.5.2 Metodologías ágiles

Este enfoque nace como respuesta a los problemas que puedan ocasionar las metodologías tradicionales y se basa en dos aspectos fundamentales, retrasar las decisiones y la planificación adaptativa. Basan su fundamento en la adaptabilidad de los procesos de desarrollo.

Estas metodologías ponen de relevancia que la capacidad de respuesta a un cambio es más importante que el seguimiento estricto de un plan.

2.6 Metodología Web

Llamados procesos que permiten estructurar, comunicar, entender, simplificar y formalizar tanto el dominio como las decisiones de diseño, así como disponer de documentación detallada para posibles cambios de software.

2.6.1 Metodología UWE

Es una metodología que permite modelar de mejor manera una aplicación Web, para el proceso de creación de aplicaciones, con una gran cantidad de definiciones, en el proceso de diseño de sistemas. Procede de manera iterativa e incremental, coincidiendo con UML incluyendo flujos de trabajo y puntos de control. (Roger S. Pressman, 2010)

Es una herramienta para modelar aplicaciones web, utilizadas en la ingeniería web, prestando especial atención en sistematización y personalización (sistemas adaptativos). Su proceso de desarrollo se basa en cuatro fases principales: la fase de captura de requisitos, la fase de análisis y diseño de contenidos, la fase de modelo navegacional y la fase de implementación.

UWE es una propuesta basada en el proceso unificado y UML, pero adaptados a la Web. En requisitos separa las fases de captura, definición y validación. Hace además una clasificación y un tratamiento especial dependiendo del carácter de cada requisito. (Minguez Daniel y Garcia Emilio, 2007, pág. 3)

2.6.1.1 Características de la metodología UWE

Se basa en las características principales siguientes:

- Notación Estándar: el uso de la metodología UML para todos los modelos.
- Métodos definidos: pasos definidos para la construcción de cada modelo.
- Especificación de restricciones: recomendables de manera escrita, para que la exactitud en cada modelo aumente.

2.6.1.2 Actividades de Modelado de UWE

Las actividades base de modelado de UWE son el análisis de requerimientos, el modelo conceptual, el modelo navegacional y el modelo de presentación. A estos modelos se pueden sumar otros modelos como lo son el modelo de interacción y la visualización de escenarios Web.

2.6.1.3 Fases de UWE

Los modelos orientados a objetos se construyen en cada paso que mejora los modelos diseñados en iteraciones anteriores y consta de las siguientes fases:

a. Captura, Análisis y Especificación

Fija los requisitos funcionales de la aplicación web para reflejarlos en un modelo de caso de uso.

Se adquieren, reúnen y especifican las características funcionales y no funcionales que deberá cumplir la aplicación web.

Trata de diferente forma las necesidades de información, las necesidades de navegación, las necesidades de adaptación y las de interfaz de usuario así como algunos requisitos adicionales. Centra el trabajo en el estudio de los casos de uso, la generación de los glosarios y el prototipo de la interfaz de usuario.

b. Diseño del Sistema

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

c. Codificación del software

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

d. Pruebas

Las pruebas se utilizan para asegurar el correcto funcionamiento de secciones de código.

En esta fase veremos la prueba de la caja negra, ya que la prueba de caja blanca no estamos mencionando siendo que es considerada una caja de cristal por la cual está reservada por seguridad.

e. Fase de Implementación

Es el proceso por el cual los programas desarrollados son transferidos apropiadamente al computador destino, inicializados, y, eventualmente, configurados; todo ello con el propósito de ser ya utilizados por el usuario final.

f. Mantenimiento y Seguimiento

Es el proceso de control, mejora y optimización del software ya desarrollado e instalado, que también incluye depuración de errores y defectos que puedan haberse filtrado de la fase de pruebas de control.

Una vez puesta la Pagina Web a Disposición de los usuarios hay que ir cambiando datos y mantener este sitio actualizado, ya que esta página no puede permanecer estática. Los problemas de uso no detectados durante el proceso de desarrollo pueden descubrirse a través de varios métodos, principalmente a través de los mensajes, opiniones de los usuarios, el comportamiento y uso del sitio.

2.6.1.4 Lenguaje Unificado de Modelado (UML)

El Lenguaje de Modelado Unificado (UML: Unified Modeling Language) es la sucesión de una serie de métodos de análisis y diseño orientadas a objetos que aparecen a fines de los 80's y principios de los 90's. El UML, fusiona los conceptos de la orientación a objetos aportados por Booch, OMT y OOSE.

UML ofrece un estándar para describir un "plano" del sistema (modelo), incluyendo aspectos conceptuales tales como procesos de negocios y funciones del sistema, y

aspectos concretos como expresiones de lenguajes de programación, esquemas de bases de datos y componentes de software reutilizables.

UML es un "lenguaje" para especificar y no un método o un proceso. UML se puede usar en una gran variedad de formas para soportar una metodología de desarrollo de software (tal como el Proceso Unificado de Rational) pero no especifica en sí mismo qué metodología o proceso usar. (NUÑES, 2000)

2.6.2 Ciclo de la Metodología UWE

UWE es una metodología dirigida o enfocada al modelado de aplicaciones Web, ya que está basada estrictamente en UML, esta metodología nos garantiza que sus modelos sean fáciles de entender para los que manejan UML.

2.6.2.1 Ciclo de Análisis

El Ciclo de Análisis de Requerimientos realiza la captura de los mismos mediante diagramas de casos de uso acompañado de documentación que detallada.

Diagrama de Casos de Uso

Los diagramas de casos de uso describen las relaciones y las dependencias entre un grupo de casos de uso y los actores participantes en el proceso.

Es importante resaltar que los diagramas de casos de uso no están pensados para representar el diseño y no puede describir los elementos internos de un sistema. Los diagramas de casos de uso sirven para facilitar la comunicación con los futuros usuarios del sistema y con el cliente, resultan especialmente útiles para determinar las características necesarias que tendrá el sistema.

Un diagrama de casos de uso consta de los siguientes elementos:

Actor: Es un rol que un usuario juega con respecto al sistema. Un Actor no necesariamente representa a una persona en particular, sino más bien la labor que realiza frente al sistema.

Caso de Uso: Es una operación/tarea específica que se realiza tras una orden de algún agente externo, sea desde una petición de un actor o bien desde la invocación desde otro caso de uso.

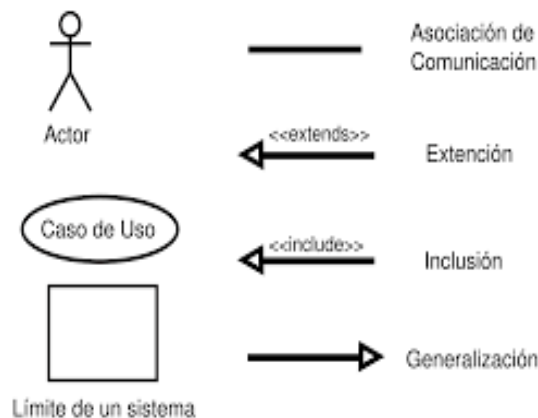
Relaciones:

- **Asociación:** Es el tipo de relación más básica que indica la invocación desde un actor o caso de uso a otra operación (caso de uso). Dicha relación se denota con una flecha simple.
- **Dependencia o Instanciación:** Es una forma muy particular de relación entre clases, en la cual una clase depende de otra, es decir, se instancia (se crea). Dicha relación se denota con una flecha punteada.
- **Generalización:** Este tipo de relación es uno de los más utilizados, cumple una doble función dependiendo de su estereotipo, que puede ser de Uso (<<uses>>) o de Herencia (<<extends>>).

Extends: Se recomienda utilizar cuando un caso de uso es similar a otro (características).

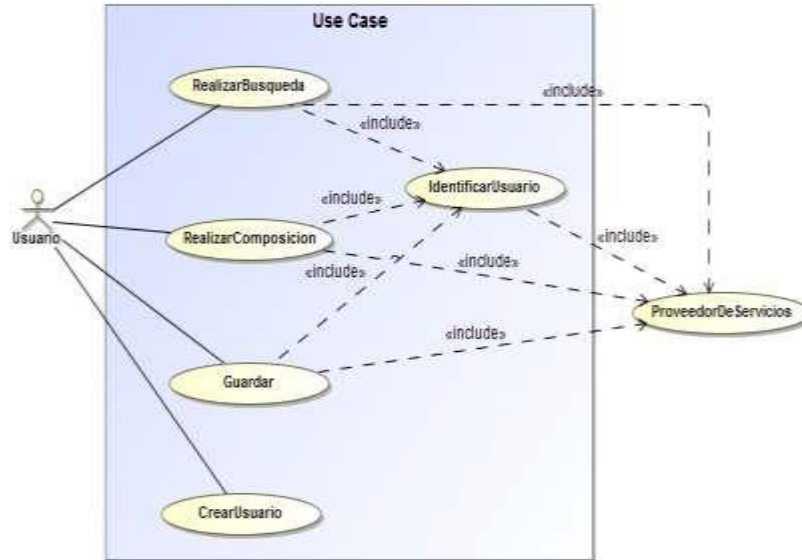
Uses: Se recomienda utilizar cuando se tiene un conjunto de características que son similares en más de un caso de uso y no se desea mantener copiada la descripción de la característica.

Figura 2.1 Diagrama de Caso de Uso



Fuente : (Ludwig-Maximilians-Universität München UWE, 2016)

Figura 2.2 Grafico de Modelos de Caso de Uso



Fuente : (Ludwig-Maximilians-Universität München UWE, 2016)

2.6.2.2 Ciclo de Diseño Conceptual

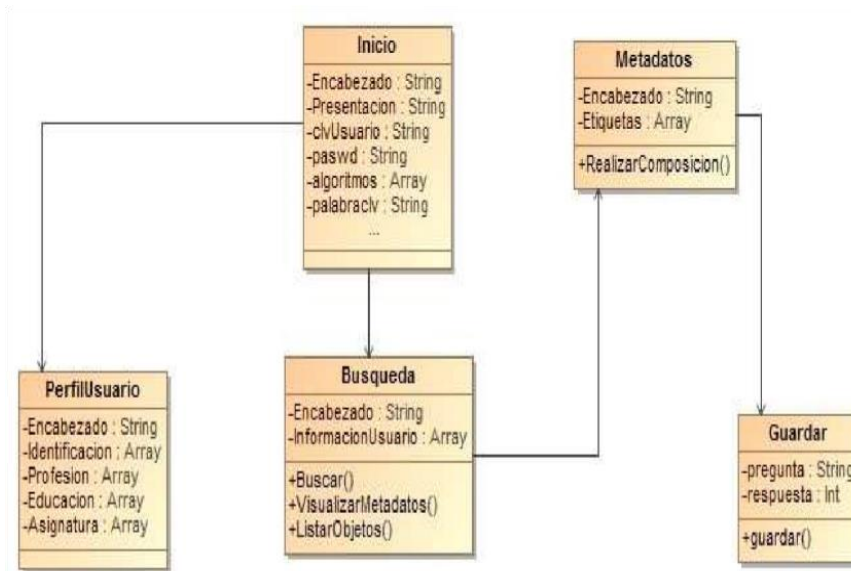
Caracterizado por un Ciclo de dominio, que utiliza los requisitos que se detallan en los casos de uso. En esta etapa se representa el dominio del problema con un diagrama de clases de UML, que permiten determinar, métodos y atributos.

El propósito de este diagrama es construir un modelo del dominio que intenta no considerar el paseo de la navegación, la presentación y los aspectos de interacción. Aspectos que se analizarán en los pasos respectivos de navegación y presentación de la planificación.

Modelo Conceptual

Un diagrama de contenido es un diagrama UML normal de clases. Los diagramas de clases describen la estructura estática de un sistema. UWE provee diferentes estereotipos.

Figura 2.3 Gráfico del Diagrama de Contenido de la Metodología UWE



Fuente: (Ludwig-Maximilians-Universität München UWE, 2016)

2.6.2.3 Ciclo de Diseño Navegacional

Basado en el diagrama de la fase conceptual, donde se especifica los objetos que serán visitados dentro de la aplicación web y la relación entre los mismos.

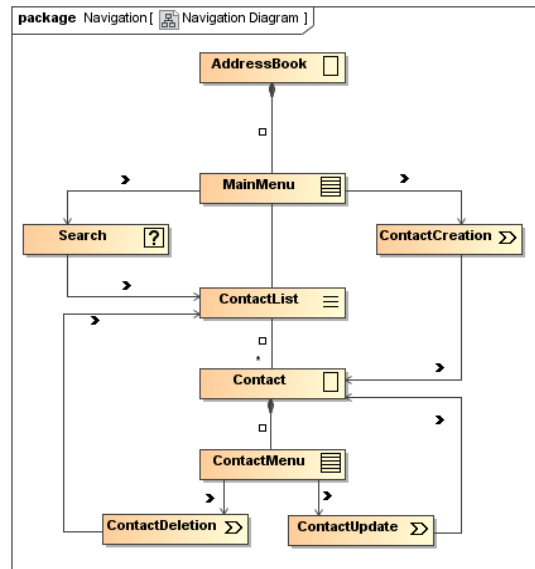
Su objetivo principal es representar el diseño y estructura de las rutas de navegación al usuario para evitar la desorientación en el proceso de navegación.

Modelo Navegacional

Es un sistema para la web es útil saber cómo están enlazadas las páginas.

Ello significa que necesitamos un diagrama contenido nodos (nodes) y enlaces (links).

Figura 2.4 Gráfico del Diagrama de Navegación de la metodología UWE



Fuente: (Ludwig-Maximilians-Universität München UWE, 2016)

2.6.2.4 Ciclo de Diseño de la Presentación

El Ciclo de diseño de presentación tiene como objetivo la representación de las vistas del interfaz del usuario final, la representación gráfica de esta fase se encuentra basada en los diagramas realizados en las fases anteriores.

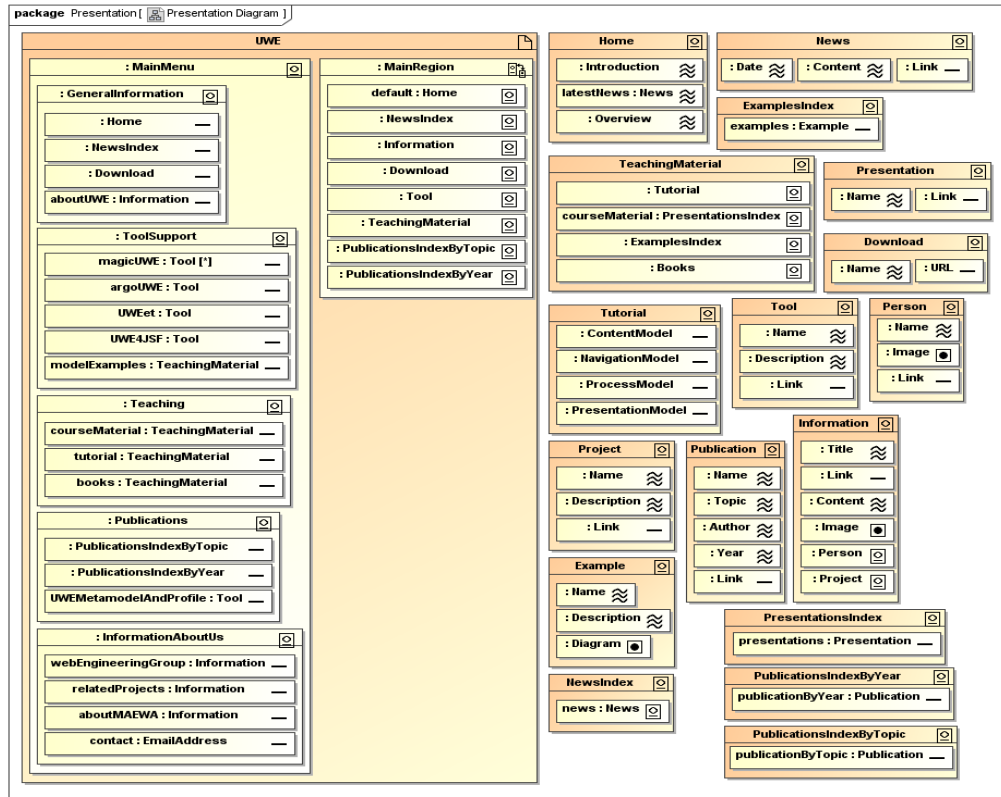
Las clases del modelo de presentación representan páginas Web o parte de ellas, organizando la composición de los elementos de la interfaz de usuario y las jerarquías del modelo de presentación.

El diagrama de esta fase representa los objetos de navegación y elementos de acceso, por ejemplo, en que marco o ventana se encuentra el contenido y que será remplazado cuando se accione un enlace. En la siguiente imagen podremos observar un ejemplo de un diagrama de presentación mediante UWE.

Modelo de Presentación

El Modelo de Navegación no indica cuales son las clases de navegación y de proceso que pertenecen a una página web. Podemos usar un Diagrama de Presentación con el fin de proveer esta información.

Figura 2.5 Gráfico del Diagrama de Presentación de la Metodología UWE



Fuente: (Ludwig-Maximilians-Universität München UWE, 2016)

2.7 Pruebas

2.7.1 Pruebas de Caja Blanca

(White-Box Testing). Son pruebas estructurales. Conociendo el código y siguiendo su estructura lógica, se pueden diseñar pruebas destinadas a comprobar que el código hace correctamente lo que el diseño de bajo nivel indica y otras que demuestren que se comporta adecuadamente ante determinadas situaciones.

2.7.2 Pruebas de Caja Negra

En la teoría de sistemas y física, se denomina caja negra aquel elemento que es estudiado desde el punto de vista de las entradas y salidas o respuestas que produce, sin tener en cuenta su funcionamiento interno. De una caja negra interesará su forma de interactuar con el medio que lo rodea (en ocasiones, otros

elementos que también podrían ser cajas negras) entendiéndolo qué es lo que hace, pero sin dar importancia a cómo lo hace. Por tanto, de una caja negra deben estar muy bien definidas sus entradas y salidas, es decir, su interfaz: en cambio, no se precisa definir ni conocer los detalles internos de su funcionamiento.

Las pruebas de caja negra intentan encontrar errores de las siguientes categorías:

- Funciones incorrectas o ausentes.
- Errores de interfaz.
- Errores de estructuras de datos o en acceso a base de datos externas.
- Errores de rendimiento.
- Errores de inicio y fin.

2.8 Arquitectura de Software

2.8.1 Patrón de diseño Model View Controller (MVC)

El patrón MVC siglas de Model View Controller, en español Modelo Vista Controlador. Esto también se ve reflejado en que cada una de estas palabras representa cada uno de los 3 componentes del patrón MVC, cada parte juega un rol fundamental para la completa integración del sistema.

2.8.2 Definición e historia

“El propósito de este patrón es simplificar la implementación de aplicaciones de acuerdo a las peticiones de los usuarios y los datos a desplegar”. La definición un poco más formal sería: MVC es un patrón de diseño de software que separa los datos de una aplicación, la interfaz de usuario, y la lógica de control en tres componentes distintos de forma que las modificaciones al componente de la vista, o a cualquier parte del sistema puedan ser hechas con un mínimo impacto en el componente del modelo de datos o en los otros componentes del sistema. Este patrón cumple perfectamente el cometido de modularizar un sistema.

El patrón MVC fue descrito por primera vez en 1979 por Trygve Reenskaug, Este patrón se ve frecuentemente utilizado en aplicaciones web, donde la vista es la

página HTML y el código provee de datos dinámicos a la página. Las aplicaciones web complejas continúan siendo más difíciles de diseñar que las aplicaciones tradicionales de escritorio, el patrón MVC se presenta como una solución para ayudar a disminuir dicha complejidad.

2.8.3 Componentes

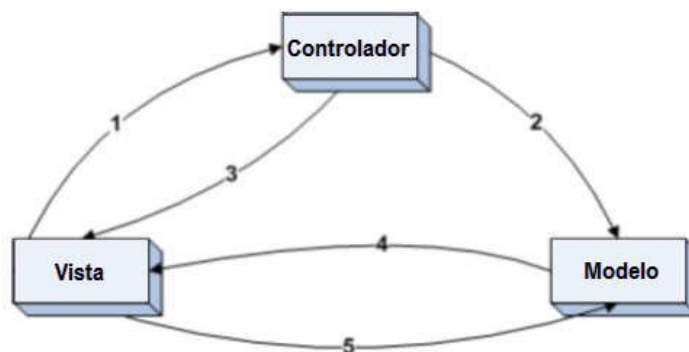
Los tres componentes principales de este patrón son los siguientes:

Modelo: almacena los datos, el estado y la lógica de la aplicación. Este componente debe proveer una interfaz para manipular y averiguar su estado. Desde el punto de vista del patrón Observer, el modelo juega el rol de subject, y cada vez que cambie de estado debe notificárselo a las vistas, que juegan el rol de observers.

Vista: es la responsable de transformar el modelo para que sea visualizada por el usuario, ya sea en un archivo de texto normal o en una página web (HTML o JSP) que el navegador pueda desplegar. En sí el propósito de la vista es convertir los datos para que al usuario le sean significativos y los pueda interpretar fácilmente.

Controlador: es la parte lógica que es responsable del procesamiento y comportamiento de acuerdo a las peticiones (requests) del usuario, construyendo un modelo apropiado, y pasándolo a la vista para su correcta visualización.

Figura 2.6: Interacción entre Modelo, Vista y Controlador



Fuente: (Freeman & Elizabeth, 2004)

A continuación se describen de manera general cada uno de los pasos de esta interacción.

1. El usuario interactúa con la vista, y ésta notifica al controlador lo sucedido.
2. El controlador determina cómo manipular o cambiar el estado del modelo en base a la acción realizada por el usuario.
3. Opcionalmente, el controlador también puede actualizar el estado de la vista en respuesta a determinadas acciones del usuario. Ejemplos de estas situaciones son la habilitación o inhabilitación de componentes de interacción.
4. El modelo notifica a la vista de su cambio de estado.
5. La vista obtiene el estado del modelo, averigua cómo cambió su estado y actualiza su apariencia de acuerdo al cambio ocurrido. (Freeman, 2004)

2.9 Herramientas

2.9.1 Gestor de Base de Datos MySQL

Es un sistema de gestión de bases de datos relacional, fue creada por la empresa sueca MySQL AB, la cual tiene el copyright del código fuente del servidor SQL, así como también de la marca. MySQL es un software de código abierto, licenciado bajo la GPL de la GNU, aunque MySQL AB distribuye una versión comercial, en lo único que se diferencia de la versión libre, es en el soporte técnico que se ofrece, y la posibilidad de integrar este gestor en un software propietario, ya que de otra manera, se vulneraría la licencia GPL.

El lenguaje de programación que utiliza MySQL es Structured Query Language (SQL) que fue desarrollado por IBM en 1981 y desde entonces es utilizado de forma generalizada en las bases de datos relacionales. (Claudio Alberto Nipotti, 2011, pág. 5)

MySQL es un SGBD que ha ganado popularidad por una serie de atractivas características:

- Está desarrollado en C/C++.
- Se distribuyen ejecutables para cerca de diecinueve plataformas diferentes.
- La API se encuentra disponible en C, C++, Eiffel, Java, Perl, PHP, Python, Ruby y TCL.
- Está optimizado para equipos de múltiples procesadores.

- Es muy destacable su velocidad de respuesta.
- Se puede utilizar como cliente-servidor o incrustado en aplicaciones.
- Cuenta con un rico conjunto de tipos de datos.
- Soporta múltiples métodos de almacenamiento de las tablas, con prestaciones y rendimiento diferentes para poder optimizar el SGBD a cada caso concreto.
- Su administración se basa en usuarios y privilegios.
- Se tiene constancia de casos en los que maneja cincuenta millones de registros, sesenta mil tablas y cinco millones de columnas.
- Sus opciones de conectividad abarcan TCP/IP, sockets UNIX y sockets NT, además de soportar completamente ODBC.
- Los mensajes de error pueden estar en español y hacer ordenaciones correctas con palabras acentuadas o con la letra 'ñ'.
- Es altamente confiable en cuanto a estabilidad se refiere.
- MySQL tiene como principal objetivo ser una base de datos fiable y eficiente. Ninguna característica es implementada en MySQL si antes no se tiene la certeza que funcionará con la mejor velocidad de respuesta y, sin causar problemas de estabilidad.

2.9.2 Workbench

“MySQL Workbench es una herramienta visual de diseño de bases de datos que integra desarrollo de software, Administración de bases de datos, diseño de bases de datos, creación y mantenimiento para el sistema de base de datos MySQL. Es el sucesor de DBDesigner 4 de fabFORCE.net, y reemplaza el anterior conjunto de software, MySQL GUI Tools Bundle. (Castell, 2014)

2.9.3 Lenguaje de Programación PHP

El PHP (Hypertext Preprocessor), es un lenguaje interpretado de alto nivel embebido en páginas HTML y ejecutado en el servidor.

A diferencia de Java o JavaScript que se ejecutan en el navegador PHP se ejecuta en el servidor por eso nos permite acceder a los recursos que tenga el servidor como por ejemplo podría ser una base de datos.

El programa PHP es ejecutado en el servidor y el resultado es enviado al navegador. Al ser un lenguaje libre dispone de una gran cantidad de características que lo convierten en la herramienta ideal para la creación de páginas web dinámicas:

- Soporte para una gran cantidad de bases de datos: MySQL, Postgre SQL, Oracle, MS SQL Server, Sybasem SQL, Informix, entre otras.
- Integración con varias bibliotecas externas, permite generar documentos en PDF (documentos de Acrobat Reader) hasta analizar código XML.
- Ofrece una solución simple y universal para las paginaciones dinámicas Web de fácil programación. (Airston G, 2019)

2.9.3.1 Características de PHP

Entre las características que hacen de PHP un lenguaje popular y muy poderoso para desarrollar aplicaciones, podemos citar las siguientes:

- Programación de páginas dinámicas en servidores
- Programación de aplicaciones de escritorio con GTK (PHPGTK)
- Soporte para trabajar con múltiples base de datos
- Soporte para múltiples plataformas
- Facilidad de aprendizaje
- Portabilidad de código entre diferentes plataformas
- Total libertad para distribuir las aplicaciones

2.9.4 Apache

Es un servidor web HTTP de código abierto para la creación de páginas y servicios web. Es un servidor multiplataforma, una de las ventajas más grandes de Apache es que es un servidor web multiplataforma gratuito muy robusto y que destaca por su seguridad y rendimiento. (Robert Mc.Cool, 1995)

Entre las principales características de Apache se encuentran las siguientes:

- Soporte de seguridad SSL TLS.
- Puede realizar autenticación de datos utilizando SGDB.

- Puede dar soporte a diferentes lenguajes como Perl, PHP y Python.

2.9.5 Framework-Laravel

Laravel es un framework PHP. Es uno de los frameworks más utilizados y de mayor comunidad en el mundo de Internet.

Como framework resulta bastante moderno y ofrece muchas utilidades potentes a los desarrolladores, que permiten agilizar el desarrollo de las aplicaciones web.

Laravel pone énfasis en la calidad del código, la facilidad de mantenimiento y escalabilidad, lo que permite realizar proyectos desde pequeños a grandes o muy grandes. Además permite y facilita el trabajo en equipo y promueve las mejores prácticas.

El objetivo de Laravel es el de ser un framework que permita el uso de una sintaxis refinada y expresiva para crear código de forma sencilla, evitando el “código espagueti” y permitiendo multitud de funcionalidades. (Jose M Garcia, 2016)

2.9.5.1 Características de Laravel

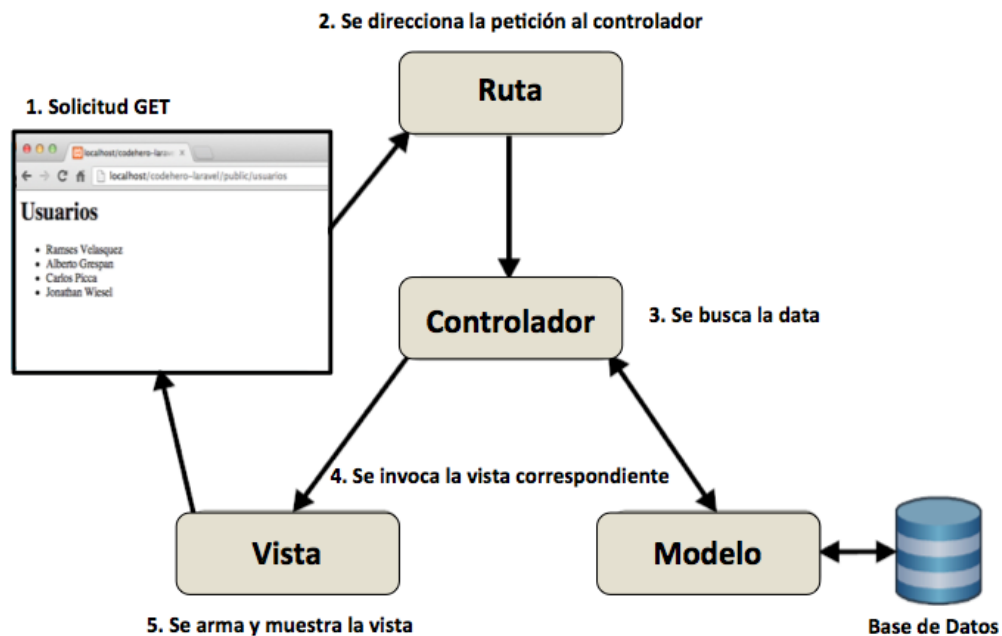
El framework Laravel trabaja con una arquitectura de carpetas avanzada, de modo que promueve la separación de los archivos con un orden correcto y definido, que guiará a todos los integrantes del equipo de trabajo y será un estándar a lo largo de los distintos proyectos. Por supuesto, dispone también de una arquitectura de clases también muy adecuada, que promueve la separación del código por responsabilidades. Su estilo arquitectónico es MVC.

Contiene además un amplio conjunto de características, que sirven para realizar la mayoría de las aplicaciones web. Entre ellas podemos encontrar:

- Un sistema de rutas, mediante las cuales es fácil crear y mantener todo tipo de URLs amistosas a usuarios y buscadores, rutas de API, etc.
- Un sistema de abstracción de base de datos, con un ORM potente pero sencillo de manejar, mediante el que podemos tratar los datos de la base de datos como si fueran simples objetos.

- Varias configuraciones para envío de email, con proveedores diversos
- Un sistema de notificaciones a usuarios, mediante email, base de datos y otros canales
- Gestión de sesiones
- Sistema de autenticación, con todo lo necesario como recordatorios de clave, confirmación de cuentas, recordar un usuario logueado, etc.
- La posibilidad de acceder a datos en realtime y recibir notificaciones cuando éstos se alteran en la base de datos

Figura 2.7 Enrutamiento básico de Laravel



Fuente: (Guía de usuarios,2011)

Así es como funciona el modelo MVC (Model-View-Controller). Supongamos que queremos ingresar a la siguiente URL <http://dominio.com/saludo> y desplegar una página con el mensaje “Bienvenido”. En Laravel la porción /saludo pertenecería a una ruta que regresa una respuesta o una vista dependiendo lo complejo que llegue hacer lo que queramos mostrar. La parte de dominio.com pertenecería a localhost si lo andamos probando de manera local.

2.9.6 Bootstrap

El framework vio la luz en el año 2011. En un principio fue desarrollado por Twitter aunque posteriormente fue liberado bajo licencia MIT. Hoy en día continúa su desarrollo en un repositorio de GitHub. Se trata de un framework que ofrece la posibilidad de crear un sitio web totalmente responsive mediante el uso de librerías CSS. En estas librerías, nos podemos encontrar un gran número de elementos ya desarrollados y listos para ser utilizados como pueden ser botones, menús, cuadros e incluso un amplio listado de tipografías. Desde que vio la luz Bootstrap se ha caracterizado por tratarse de una excelente herramienta para crear interfaces de usuarios limpias y totalmente adaptables a cualquier tipo de dispositivo y pantalla, independientemente de su tamaño. (Antonio Javier Gallego Sanchez, 2018)

2.9.7 HTML

HTML (Hyper Text Markup Language) es lo que se conoce como "Lenguaje de Marcas de Hipertexto", cuya función es preparar documentos escritos aplicando etiquetas de formato. Las etiquetas indican cómo se presenta el documento y cómo se vincula a otros documentos.

HTML se usa también para la lectura de documentos en Internet desde diferentes equipos gracias al protocolo HTTP, que permite a los usuarios acceder, de forma remota, a documentos almacenados en una dirección específica de la red, denominada dirección URL.

Se clasifican en varios grupos según su función.

- **Semántica:** lo que le permite describir con mayor precisión cuál es su contenido.
- **Conectividad:** lo que le permite comunicarse con el servidor de formas nuevas e innovadoras.
- **Desconectado y almacenamiento:** permite a páginas web almacenar datos, localmente, en el lado del cliente y operar fuera de línea de manera más eficiente.

- **Multimedia:** permite hacer vídeo y audio de ciudadanos de primera clase en la Web abierta.
- **Gráficos y efectos 2D/3D:** permite una gama mucho más amplia de opciones de presentación.
- **Rendimiento e Integración:** proporcionar una mayor optimización de la velocidad y un mejor uso del hardware del equipo.
- **Dispositivo de Acceso:** admite el uso de varios dispositivos de entrada y salida.
- **Styling:** deja a los autores escribir temas más sofisticados.

Las principales etiquetas HTML5 nuevas no tienen una representación especial en pantalla.

2.9.8 CSS

Hojas de Estilo en Cascada (Cascading Style Sheets), es un mecanismo simple que describe cómo se va a mostrar un documento en la pantalla.

CSS se utiliza para dar estilo a documentos HTML y XML, separando el contenido de la presentación. Los Estilos definen la forma de mostrar los elementos HTML y XML. CSS permite a los desarrolladores Web controlar el estilo y el formato de múltiples páginas Web al mismo tiempo. Cualquier cambio en el estilo marcado para un elemento en la CSS afectará a todas las páginas vinculadas a esa CSS en las que aparezca ese elemento. (Javier Egluz , 2008)

2.9.9 JSON

JSON (JavaScript Object Notation - Notación de Objetos de JavaScript) es un formato abierto de representación e intercambio de objetos, popularizado a principio de siglo por Douglas Crockford como una alternativa ligera a XML.

Una de sus principales ventajas es que permite intercambiar datos estructurados entre diferentes programas y medios de almacenamiento independientemente del

lenguaje de programación en que estén desarrollados. Además de todo, es fácil de generar y leer tanto por maquinas como por humanos. (Diego Guzman, 2018)

Los tipos de datos que se pueden serializar con JSON son:

Primitivos

- Cadena
- Números
- Constante

2.9.10 JavaScript

JavaScript es un lenguaje de programación que se utiliza principalmente para crear páginas web dinámicas.

Una página web dinámica es aquella que incorpora efectos como texto que aparece y desaparece, animaciones, acciones que se activan al pulsar botones y ventanas con mensajes de aviso al usuario.

Técnicamente, JavaScript es un lenguaje de programación interpretado, por lo que no es necesario compilar los programas para ejecutarlos. En otras palabras, los programas escritos con JavaScript se pueden probar directamente en cualquier navegador sin necesidad de procesos intermedios. (Introduccion a JavaScript, 2010)

2.9.11 JQuery

Para simplificar, podríamos decir que jQuery es un framework Javascript, pero quizás muchos de los lectores se preguntarán qué es un framework. Pues es un producto que sirve como base para la programación avanzada de aplicaciones, que aporta una serie de funciones o códigos para realizar tareas habituales. Por decirlo de otra manera, framework son unas librerías de código que contienen procesos o rutinas ya listos para usar. Los programadores utilizan los frameworks para no tener que desarrollar ellos mismos las tareas más básicas, puesto que en el propio

framework ya hay implementaciones que están probadas, funcionan y no se necesitan volver a programar. (Miguel Angel Alvares, 2009)

2.10 Métricas de Calidad

2.10.1 Calidad de software

La gestión de la calidad de software es una actividad protectora que incorpora tanto control como aseguramiento en la calidad que se aplica a cada paso en el proceso del software.

La actividad adecuada para garantizar la calidad del software requiere recopilar, evaluar y distribuir los datos acerca de los procesos de ingeniería del software (Roger S. Pressman, 2013, pág. 340)

2.10.2 Métricas de Calidad del Software

El concepto de métrica es el término que describe muchos y muy variados casos de medición. Siendo una métrica una medida estadística, estas medidas son aplicables a todo el ciclo de vida del desarrollo, desde la iniciación, cuando debemos estimar los costos, al seguimiento y control de la fiabilidad de los productos finales, y a la forma en que los productos cambian a través del tiempo debido a la aplicación de mejoras. Un ingeniero del Software recopila medidas y desarrolla métricas para obtener indicadores.

2.10.2.1 *Objetivos fundamentales de la medición son:*

- Para indicar la calidad del producto.
- Entender que ocurre durante el desarrollo y el mantenimiento.
- Controlar que es lo que ocurre en nuestros proyectos.
- Mejorar nuestros procesos y nuestros proyectos.
- Para evaluar la productividad de la gente que desarrolla el producto.
- Para evaluar los beneficios en términos de productividad y de calidad, derivados del uso de nuevos métodos y herramientas de la ingeniería de software.

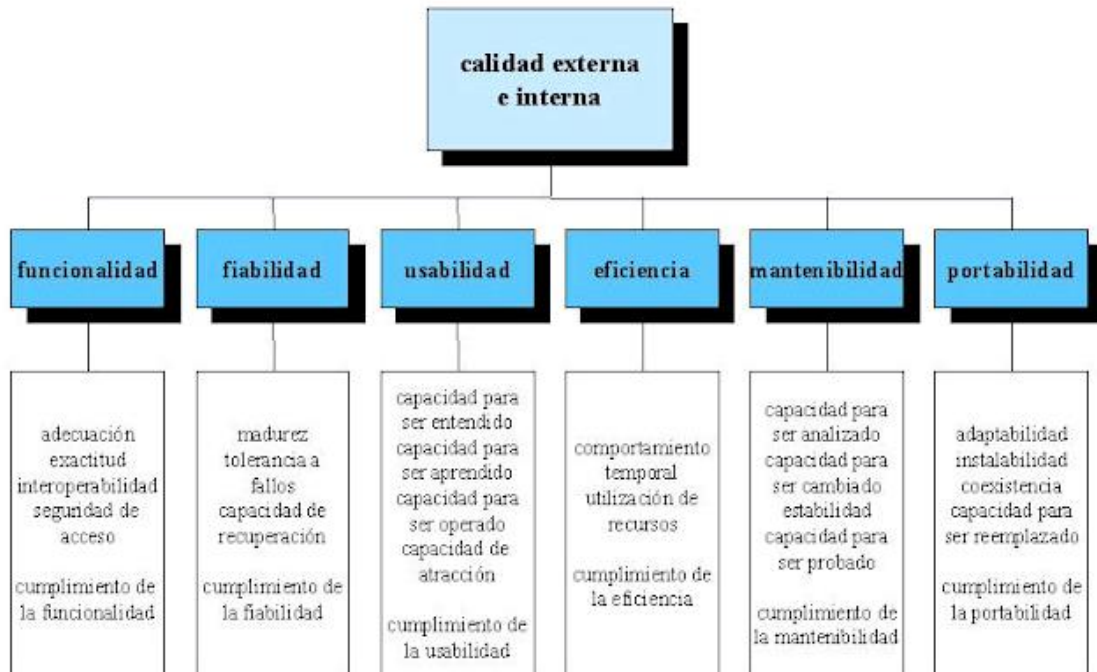
- Para establecer una línea de base para la estimación.
- Para ayudar a justificar el uso de nuevas herramientas o de formación adicional.

2.10.2.2 Calidad de Software - ISO 9126

ISO 9126 es un estándar internacional para la evaluación del software. Está dividido en cuatro partes las cuales dirigen, respectivamente el modelo de calidad, las métricas externas, métricas internas y la calidad en las métricas de uso.

La norma ISO-9126 define un modelo general de calidad basado en seis características principales para determinar la calidad de software a desarrollar.

Figura 2.8 Características principales para determinar la calidad de software



Fuente: (Sergio Matsukawa Maeda)

C1. Funcionalidad

Es la capacidad del software de proveer los servicios necesarios para cumplir con los requisitos funcionales como ser: adecuación, exactitud y seguridad.

C2. Confiabilidad

Cantidad de tiempo que el software está disponible para su uso. Está referido por los siguientes sub atributos: madurez, tolerancia a fallos y facilidad de recuperación.

C3. Usabilidad

Grado en el que el software es fácil de usar. Viene reflejado por los siguientes sub atributos: facilidad de comprensión, facilidad de aprendizaje y operatividad.

C4. Eficiencia

Grado en el que el software hace optimo el uso de los recursos del sistema. Este indicado por los siguientes suba tributos: tiempo de uso y recursos utilizados.

C5. Mantenibilidad

La facilidad con que una modificación puede ser realizada. Está indicada por los siguientes sub atributos: facilidad de análisis, facilidad de cambio, estabilidad y facilidad de prueba.

C6. Portabilidad

La facilidad con que el software puede ser llevado de un entorno a otro. Este referido por los siguientes sub atributos: facilidad de instalación, facilidad de ajuste, facilidad de adaptación al cambio. (Lozano, 2013)

2.11 Análisis de costos COCOMO II

El modelo original COCOMO (Constructive Cost Model) fue publicado por primera vez en 1981 por Barry Boehm y reflejaba las prácticas en desarrollo de software de ese entonces. En las décadas siguientes las técnicas de desarrollo cambiaron drásticamente.

Estos cambios incluyen el gasto de tanto esfuerzo en diseñar y gestionar el proceso de desarrollo de software como en la creación del producto software, un giro total desde los mainframes que trabajan con procesos batch nocturnos hacia los sistemas en tiempo real y un énfasis creciente en la reutilización de software ya existente y en la construcción de nuevos sistemas que utilizan componentes software a medida. (Boehm, 1981)

Los años y el avance tecnológico hicieron que la aplicación del modelo COCOMO original empezara a resultar problemática, como solución se determinó reinventar el modelo para aplicarlo nuevamente y así de después de muchos años de esfuerzo combinado surge COCOMO II, un modelo de estimación de coste que refleja los cambios en la práctica de desarrollo de software profesional.

Este nuevo y mejorado COCOMO resultara de gran ayuda para los estimadores profesionales de coste software. El principal calculo en el modelo COCOMO es el uso de la ecuación del esfuerzo para estimar el número de personas o de meses necesarios para desarrollar el proyecto.

COCOMO define tres modos de desarrollo o tipos de proyecto:

- **Orgánico:** proyectos relativamente sencillos, menores de 50 KDLC líneas de código, en los cuales se tiene experiencia de proyectos similares y se encuentran en entornos estables.
- **Semi-acoplado:** proyectos intermedios en complejidad y tamaño (menores de 300 KDLC), donde 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 se engloban en un entorno de gran innovación técnica. Además, se trabaja con unos requisitos muy restrictivos y de gran volatilidad. Y por otro lado existen diferentes modelos que define COCOMO:
- **Modelo básico:** se basa exclusivamente en el tamaño expresado en LDC y se utiliza para obtener una primera aproximación rápida del esfuerzo.
- **Modelo intermedio:** este añade al modelo básico quince modificadores opcionales para tener en cuenta en el entorno de trabajo incrementando así la precisión de estimación además del tamaño del programa
- incluye un conjunto de medidas subjetivas llamadas conductores de costes.
- **Modelo avanzado:** incluye todo lo del modelo intermedio además del impacto de cada conductor de coste en las distintas fases de desarrollo.

Para realización del COCOMO previamente necesitaremos conocer el número de líneas de código, posteriormente para poder realizar los cálculos del método de estimación usaremos las siguientes ecuaciones:

Figura 2.9 Análisis de costo Cocomo II

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

Fuente: (Prentice Hall, 2005)

2.11.1 Atributo de los costes

Cada atributo se cuantifica para un entorno de proyecto. La escala es muy bajo 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 (por ejemplo, si para un proyecto el atributo DATA es calificado como muy alto, el resultado de la fórmula debe ser multiplicado por 1000). El significado de los atributos es el siguiente, según su tipo:

2.11.2 Atributos de software

- **RELY:** garantía de funcionamiento requerida al software. Indica las posibles consecuencias para el usuario en el caso que existan defectos en el

producto. Va desde la sola inconveniencia de corregir un fallo (muy bajo) hasta la posible pérdida de vidas humanas (extremadamente alto, software de alta criticidad).

- **DATA:** tamaño de la base de datos en relación con el tamaño del programa. El valor del modificador se define por la relación: D/K , donde D corresponde al tamaño de la base de datos en bytes y K es el tamaño del programa en cantidad de líneas de código.
- **CPLX:** representa la complejidad del producto.

2.11.3 Atributos 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.

2.11.4 Atributos de 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.

2.11.5 Atributos 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.

2.12 Seguridad de la información

2.12.1 Estándar ISO/IEC 27000

Es un estándar para la seguridad de la información fue aprobado y publicado como estándar internacional en octubre del 2005 por ISO. Especifica los requisitos necesarios para establecer, implantar, mantener y mejorar un Sistema de Gestión de la Seguridad de la Información (SGSI).

2.12.2 Periodo de seguridad back-up

Debido a las restricciones técnicas del servidor FTP Backup, en esos momentos experimentara fuertes retrasos al restaurar el servidor mediante BackupControl. Calculamos que la restauración podría durar varios días en función del volumen de datos por ello, se recomienda que guarde los datos manualmente.

2.12.3 Sistema de Gestión de la Seguridad de la Información (SGSI)

SGSI (Sistema de Gestión de la Seguridad de la Información), conjunto de políticas y procedimientos que normalizan la gestión de la seguridad de la información, de toda una organización o de uno o varios de sus procesos de negocio debe garantizar la disponibilidad, confidencialidad e integridad de la información”.

Figura 2.10 Estructura de la Norma ISO 27002 (Dominio de Control)



Fuente: (Luis Castellanos,2015)

2.12.3.1 Política de seguridad

Su objetivo es proporcionar a la gerencia la dirección y soporte para la seguridad de la información, en concordancia con los requerimientos comerciales y las leyes y regulaciones relevantes. Esto por supuesto debe ser creado de forma particular por cada organización. Se debe redactar un "Documento de la política de seguridad de la información". Aspectos organizativos de la seguridad de la información

La organización de la seguridad de la información se puede dar de dos formas: organización interna y organización con respecto a terceros. Organización interna: se tiene como objetivo manejar la seguridad de la información dentro de la organización.

2.12.3.2 Organización con respecto a terceros

La organización en materia de seguridad de la información debe también considerarse respecto a terceros. El objetivo de esto es mantener la seguridad de la información y los medios de procesamiento de información de la organización que son ingresados, procesados, comunicados a, o manejados por, grupos externos". (William Pandini, 2018)

2.12.3.3 Gestión de activos

Se deben asignar responsabilidades por cada uno de los activos de la organización, así como poseer un inventario actualizado de todos los activos que se tienen, a quien/quienes les pertenecen, el uso que se les debe dar, y la clasificación de todos los activos.

2.12.3.4 Seguridad ligada a los recursos humanos

El objetivo de esto es asegurar que los empleados, contratistas y terceros entiendan sus responsabilidades, y sean idóneos para los roles para los cuales son considerados, reduciendo el riesgo de robo, fraude y mal uso de los medios

2.12.3.5 Seguridad física y ambiental

La seguridad física y ambiental se divide en áreas seguras y seguridad de los equipos. Respecto a las áreas seguras, se refiere a un perímetro de seguridad física que cuente con barreras o límites tales como paredes, rejas de entrada controladas por tarjetas o recepcionistas, y medidas de esa naturaleza para proteger las áreas que contienen información y medios de procesamiento de información.

2.12.3.6 Gestión de comunicaciones y operaciones

“El objetivo de esto es asegurar la operación correcta y segura de los medios de procesamiento de la información.

Se deben tener en cuenta los siguientes aspectos:

Los procedimientos de operación deben estar bien documentados, pues no basta con tener las ideas en la mente de los administradores, sino que se deben plasmar en documentos que por supuesto estén autorizados por la gerencia”. (William Pandini, 2018)

2.12.3.7 Control de acceso

Se debe contar con una política de control de acceso. Todo acceso no autorizado debe ser evitado y se deben minimizar al máximo las probabilidades de que eso suceda. Todo esto se controla mediante registro de usuarios, gestión de privilegios, autenticación mediante usuarios y contraseña.

Los usuarios deben asegurar que el equipo desatendido tenga la protección apropiada, como por ejemplo la activación automática de un protector de pantalla después de cierto tiempo de inactividad, el cual permanezca impidiendo el acceso hasta que se introduzca una contraseña conocida por quien estaba autorizado para utilizar la máquina desatendida. (William Pandini, 2018)

2.12.3.8 Adquisición, desarrollo y mantenimiento de los sistemas de información

“Contemplar aspectos de seguridad es requerido al adquirir equipos y sistemas, o al desarrollarlos. No solamente se debe considerar la calidad y el precio, sino que la seguridad que ofrecen. Debe existir una validación adecuada de los datos de entrada y de salida, controlando el procesamiento interno en las aplicaciones, y la integridad de los mensajes”. (William Pandini, 2018)

2.12.3.9 Gestión de incidentes en la seguridad de la información

“Se debe trabajar con reportes de los eventos y debilidades de la seguridad de la información, asegurando una comunicación tal que permita que se realice una acción correctiva oportuna, llevando la información a través de los canales gerenciales apropiados lo más rápidamente posible.

Asegurar que se aplique un enfoque consistente y efectivo a la gestión de los incidentes en la seguridad de la información es elemental. (William Pandini, 2018)

2.12.3.10 Gestión de la continuidad del negocio

“Las consecuencias de los desastres, fallas en la seguridad, pérdida del servicio y la disponibilidad del servicio debieran estar sujetas a un análisis del impacto comercial. Se deben desarrollar e implementar planes para la continuidad del negocio para asegurar la reanudación oportuna de las operaciones esenciales. La seguridad de la información debiera ser una parte integral del proceso general de continuidad del negocio, y otros procesos gerenciales dentro de la organización”. (William Pandini, 2018)

CAPITULO III

MARCO APLICATIVO

3 MARCO APLICATIVO

3.1 Introducción

En este capítulo se desarrolla las etapas y modelos correspondientes a la fase de obtención de requisitos, de análisis y diseño del sistema y la fase de implementación, siguiendo el proceso de desarrollo de la metodología UWE, detalladas en el capítulo II.

3.2 Análisis de la Situación actual

Actualmente el Instituto Técnico “San Pablo” administra su propia información en base a procedimientos y reglamentos propios; enmarcados dentro de las normas vigentes del Ministerio de Educación. Esta información se maneja actualmente de forma manual, desde el registro de nuevos estudiantes, inscripciones, historial académico y otros aspectos inherentes a los procesos académicos de la institución realizan el proceso de inscripción, calificaciones, historial académico de forma manual.

La institución cuenta con 6 carreras las cuales son: Sistemas Informáticos, Contabilidad General, Enfermería, Secretariado Ejecutivo, Peluquería y Cosmetología y Laboratorista Dental a las cuales pueden inscribirse previa publicación de una convocatoria pública dirigido a bachilleres. Luego de haber terminado el proceso de inscripción se asigna las materias correspondientes de la carrera elegida una vez que aprueban la materia con promedio de 61 puntos o superior al mismo hasta 100 puntos, luego se emite el reporte de calificaciones.

3.3 Obtención de requisitos

Para realizar un mejor proyecto se requiere la tarea de ingeniería de requisitos ya que es fundamental así que se realizó las actividades de la siguiente tabla:

Tabla 3.1 Obtención de requisitos

+	Característica
Entrevista	Se realizó las entrevistas en la institución al siguiente personal: <ul style="list-style-type: none">➤ Rector del Instituto San Pablo➤ Director del Instituto➤ Secretaria Del Instituto
Observación	Algunas dificultades en el proceso de control de búsqueda para la gestión académica.
Documentación	Se me permitió observar y revisar el proceso que sigue la documentación.

Fuente: (Elaboración propia)

3.3.1 Definición de actores

Los usuarios se presentan por diferentes actores, dependiendo su relación con el sistema, a continuación se muestra en la siguiente tabla los actores que interactúan con el sistema.

Tabla 3.2 Lista de Actores

ACTOR	DESCRIPCIÓN
Administrador del Sistema	Tiene las siguientes funciones: <ul style="list-style-type: none">➤ Asignar privilegios de usuarios
Secretaria Académica	Tiene las siguientes funciones: <ul style="list-style-type: none">➤ Realiza inscripción de estudiantes➤ Realiza asignación de materias➤ Generar historial académico➤ Generar centralizadores de calificación➤ Realiza búsquedas de estudiantes y docentes➤ Administra datos de estudiantes➤ Genera reportes➤ Visualiza información de estudiantes➤ Visualiza información de docentes
Docente	Tiene las siguientes funciones: <ul style="list-style-type: none">➤ Realiza calificaciones➤ Visualiza información del docente➤ Genera reporte de calificaciones
Estudiante	Tiene las siguientes funciones: <ul style="list-style-type: none">➤ Visualiza información del estudiante➤ Visualiza historial académico➤ Visualiza calificaciones actuales

Fuente:(Elaboración Propia)

3.3.2 Lista de Requerimientos del Sistema

La obtención correcta de los requerimientos puede llegar a describir con claridad en forma consistente por ello que se toman en cuenta para diferenciar los conceptos de los requisitos:

3.3.2.1 *Requisitos Funcionales*

Los requisitos funcionales en la siguiente tabla muestran las características que necesita el sistema a partir de la información obtenida como parte de las tareas de obtención de requisitos.

Tabla 3.3 Requerimientos Funcionales

REQUERIMIENTO	DESCRIPCIÓN
Administrar Usuarios	El Administrador del Sistema podrá crear, modificar y eliminar un usuario para ingresar al sistema, y así también le asignará privilegios de usuario.
Generar reportes generales	El administrador académico podrá generar reportes sobre los usuarios del sistema en forma general.
Visualizar información del estudiante	El personal de archivos de Dirección académica y el estudiante, son los que podrán acceder a esta información concerniente al estudiante.
Visualizar información del docente	El personal de archivos de Dirección académica y el docente, son los que podrán acceder a esta información concerniente al docente de la institución.
Realizar búsquedas	El personal de archivos de Dirección académica podrá realizar búsquedas sobre estudiantes y docentes, con el número de documento de identificación u otro parámetro de búsqueda.
Realizar inscripción de estudiantes	El personal de archivos de Dirección académica podrá realizar las inscripciones a estudiantes nuevos y actuales de la institución.
Realizar asignación de materias	El personal de archivos de Dirección académica podrá realizar las asignaciones de materias a estudiantes nuevos y actuales. Así también podrá realizar la asignación de materias a los docentes de la institución.

Generar historial académico de estudiantes	El personal de archivos de Dirección académica podrá generar el historial académico de los estudiantes de la institución.
Generar Centralizadores de calificaciones	El personal de archivos de Dirección académica, podrá generar los centralizadores semestrales de calificaciones de los estudiantes de todas las carreras de la institución.
Realizar consultas	El personal de archivos de Dirección académica podrá realizar consultas sobre los estudiantes y docentes de la institución.
Registrar calificaciones	El docente podrá registrar las calificaciones de los estudiantes que tenga a cargo, una vez registradas las calificaciones no podrá modificarlas sin previa autorización de Dirección académica.

Fuente: (Elaboración propia)

3.3.2.2 Requisitos No Funcionales

En la tabla se muestra los requisitos no funcionales:

Tabla 3.4 Requisitos No Funcionales

Rol	Función
-----	---------

R2-1	El sistema debe visualizarse y funcionar correctamente en cualquier navegador como ser internet Explorer, Mozilla, Chrome.
R2-2	Mantenimiento adecuado de la red local.
R2-3	Respaldo energético del servidor, para asegurar la disponibilidad del sistema
R2-4	Soporte y mantenimiento periódico para asegurar el buen rendimiento del sistema.

Fuente: (Elaboración propia)

3.3.3 Descripción de Funciones

Tabla 3.5 Especificaciones de los actores de Casos de Uso

Actor	Tarea
Administrador del sistema	<ul style="list-style-type: none"> ➤ Asignar privilegios de usuario ➤ Administrar usuario
Personal administrativo	<ul style="list-style-type: none"> ➤ Realizar inscripción de estudiantes ➤ Realizar asignación de materias ➤ Generar historial académico ➤ Generar centralizadores de calificaciones ➤ Realizar búsquedas ➤ Visualizar información de estudiantes ➤ Visualizar información de docentes ➤ Generar reportes
Docente	<ul style="list-style-type: none"> ➤ Registrar calificaciones de estudiantes ➤ Buscar información ➤ Visualizar notas de estudiantes ➤ Visualizar reporte de estudiantes
Estudiante	<ul style="list-style-type: none"> ➤ Visualizar información del estudiante ➤ Visualizar notas del estudiante

Fuente: (Elaboración propia)

3.4 Diseño del sistema

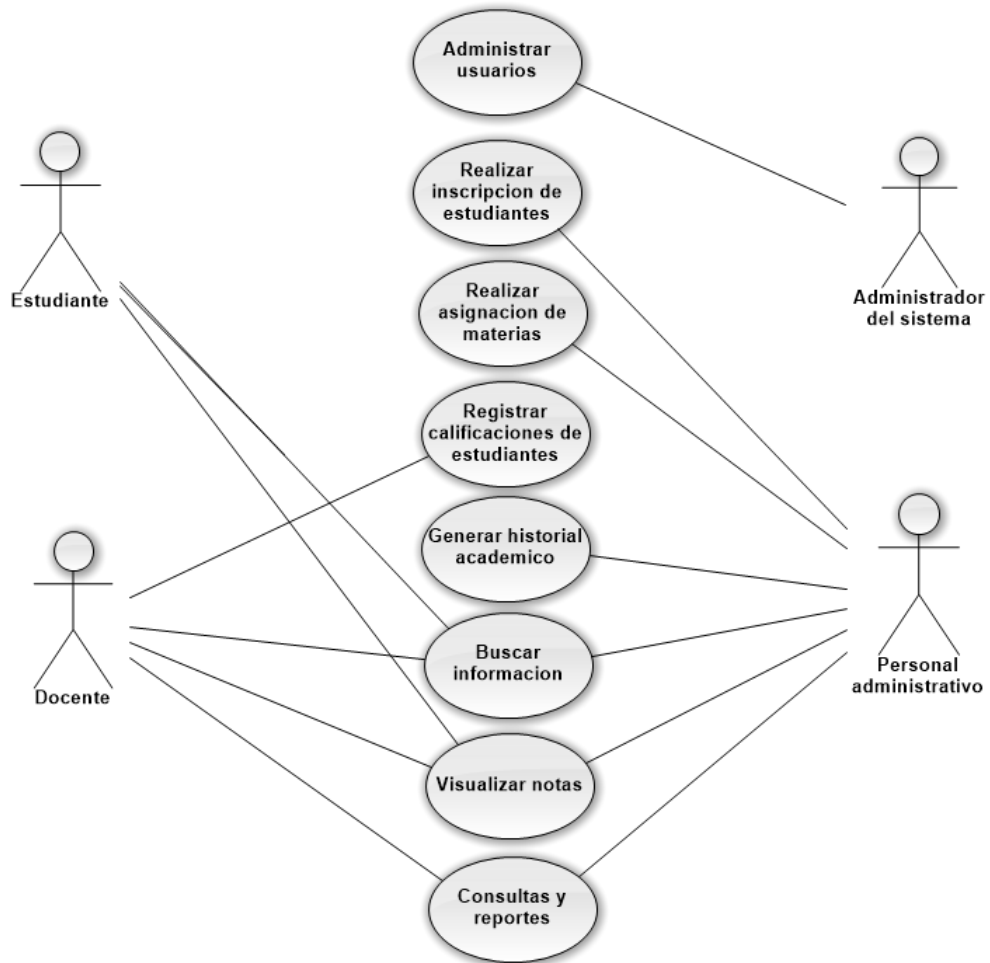
En esta fase se construye un modelo orientado a objetos que represente el dominio de la aplicación, se basa en la especificación de requisitos producido por el análisis de los requerimientos (fase de análisis), el diseño define cómo estos requisitos se cumplirán, la estructura que debe darse a la aplicación web.

3.4.1 Diseño de los diagramas de casos de uso

Los diagramas de casos de uso del sistema ayudaran al modelado y posterior desarrollo que comprenderá lo siguiente:

- La descripción escrita del comportamiento del sistema al afrontar una tarea de negocio o un requisito de negocio. Esta descripción se enfoca en el valor suministrado por el sistema a entidades externas tales como usuarios humanos u otros sistemas.
- La posición o contexto del caso de uso entre otros casos de uso. Dado que es un mecanismo de organización, un conjunto de casos de usos coherentes y consistentes promueven una imagen fácil de comprender del comportamiento del sistema, un entendimiento común entre el cliente/propietario/usuario y el equipo de desarrollo.

Figura 3.1 Diagrama de Caso de Uso General

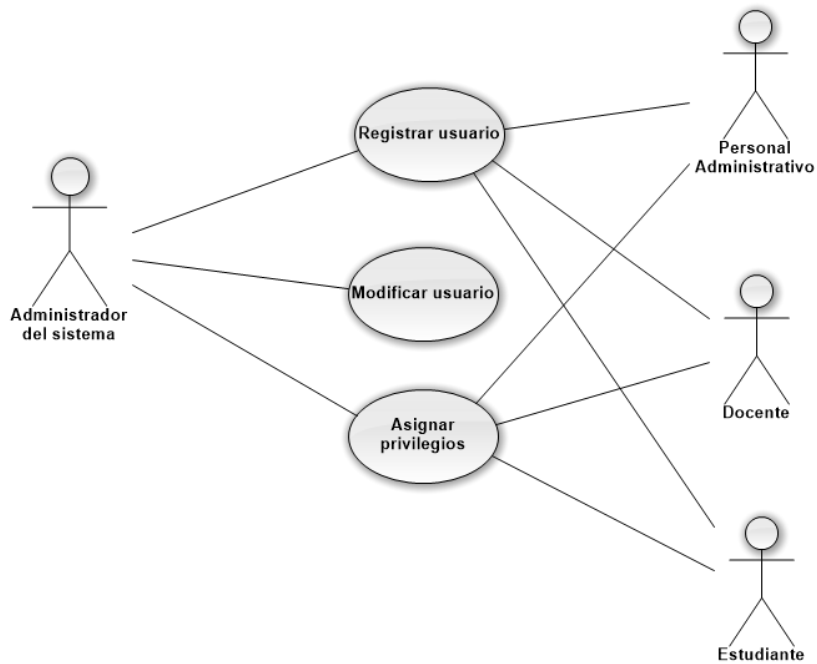


Fuente: (Elaboración propia)

El diagrama de caso de uso general, muestra de manera resumida todas las interacciones de los actores con las funciones del sistema.

3.4.2 Diagramas de casos de uso expandidos

Figura 3.2 Caso de uso administrar usuarios



Fuente: (Elaboración propia)

Tabla 3.6 Descripción de Caso de Uso: Administrar Usuario

Caso de Uso	Caso de uso administrar usuarios
Actores	Administrador del sistema, estudiante, personal administrativo, docente.
Resumen	El administrador del sistema registrará a los usuarios para el acceso al sistema, asignándole el nivel de usuario que pertenece.
Precondiciones	Debe ser un actor administrador del sistema y validado por el sistema.
Tipo	Primario

Curso Normal de Eventos:

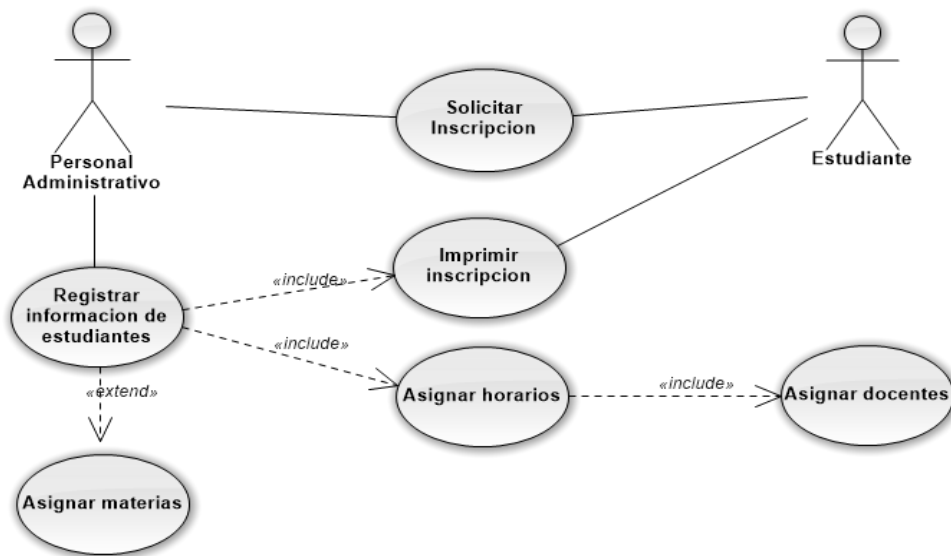
Acción de los Actores:	1. El administrador del sistema una vez autenticado desea registrar al usuario.
	2. El administrador del sistema ingresa los datos y le asigna el tipo de usuario correspondiente.
Respuesta del Sistema	3. Se despliega el formulario de registro de usuario.
	4. El Sistema valida los datos.
	5. El sistema guarda los datos.

Excepciones:

Respuesta del sistema	El sistema no realiza el registro si los datos son ingresados incorrectamente.
------------------------------	--

Fuente: (Elaboración propia)

Figura 3.3 Caso de uso inscripción de estudiantes



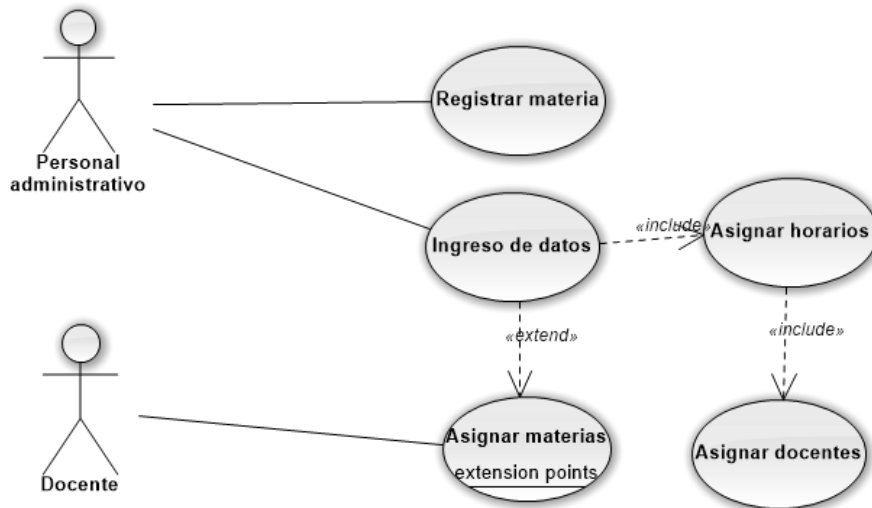
Fuente: (Elaboración propia)

Tabla 3.7 Descripción de Caso de Uso: Inscripción de Estudiantes

Caso de Uso	Caso de uso inscripción de estudiantes
Actores	Personal administrativo, Estudiante.
Resumen	El Personal administrativo (Dirección académica) es el encargado de la recepción y verificar los documentos de los estudiantes para su inscripción en el instituto.
Precondiciones	Debe ser un actor Personal (Dirección académica) y validado por el Sistema para realizar la inscripción del estudiante.
Tipo	Primario
	Curso Normal de Eventos:
Acción de los Actores:	<ol style="list-style-type: none"> 1. El estudiante prepara los documentos personales requeridos. 2. El estudiante entrega los documentos personales al Personal (Dirección académica). 3. El personal (Dirección académica) realiza la recepción de los documentos. 4. Se verifica los documentos por el personal (Dirección académica). 5. El personal (Dirección académica) guarda los documentos en un file nuevo para el estudiante.
	Excepciones:
Acción de los Actores:	El personal (Dirección académica) no realizara la inscripción si los documentos del estudiante no cumplen los requisitos.

Fuente: (Elaboración propia)

Figura 3.4 Caso de uso asignación de materias



Fuente: (Elaboración propia)

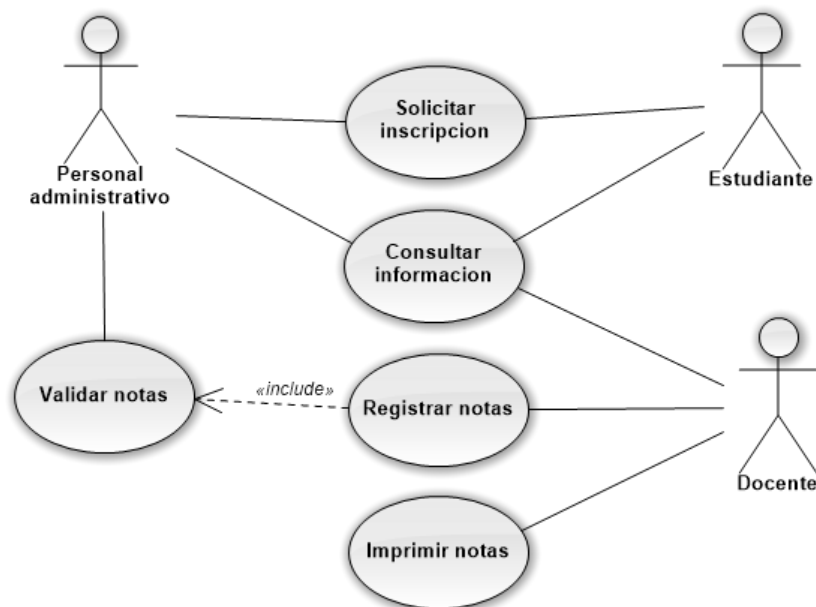
Tabla 3.8 Descripción de Caso de Uso: Asignación de Materias

Caso de Uso	Caso de uso asignación de materias
Actores	Personal (Dirección académica), Docente.
Resumen	El Personal (Dirección académica) realizará la asignación de materias al Docente.
Precondiciones	Debe ser un actor Personal (Dirección académica) para realizar la asignación de materias.
Tipo	Primario
Acción de los Actores	<p>Curso Normal de Eventos:</p> <ol style="list-style-type: none"> 1. El Docente solicita la asignación de materias. 2. El Personal (Dirección académica) ingresa al sistema para la asignación de materias. 4. El Personal (Dirección académica) ingresa el CRD (Código de Registro de Docente) al formulario de solicitud. 6. El Personal (Dirección académica) selecciona las materias y selecciona los paralelos.

		7. El Personal (Dirección académica) confirma las materias a dictar por el docente.
		9. Finalmente la asignación de materias al Docente están registradas en el sistema.
Respuesta del Sistema		3. El sistema despliega un formulario de ingreso de dato.
		5. El Sistema despliega un formulario de asignación de materias.
		8. El Sistema registra la asignación de materias.
		Excepciones:
Acción de actores	los	El Personal (Dirección académica) no realizará las asignaciones de materias a Docentes que no estén registrados.
Respuesta del Sistema	del	El sistema no realizará la asignación de materias si los datos son ingresados incorrectamente.

Fuente: (Elaboración propia)

Figura 3.5 Caso de uso registrar calificaciones de estudiantes



Fuente: (Elaboración propia)

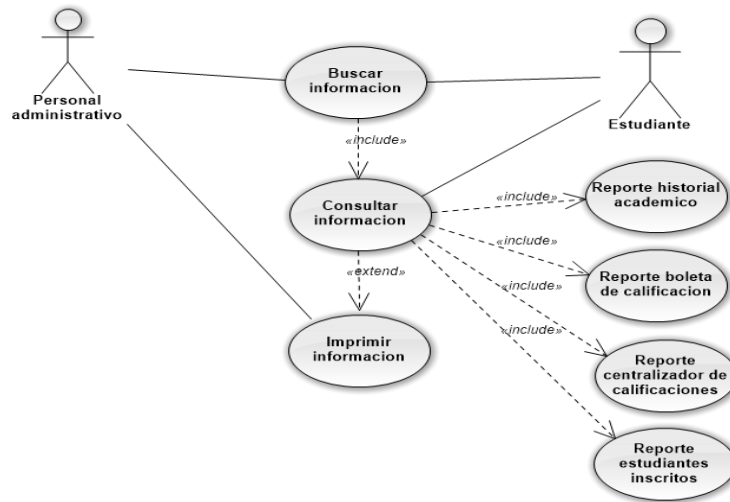
Tabla 3.9 Descripción de Caso de Uso: Registrar Calificaciones de Estudiantes

Caso de Uso	Caso de uso registrar calificaciones de estudiantes
Actores	Docente
Resumen	El Docente registrará las calificaciones de los estudiantes en el Sistema.
Precondiciones	Debe ser un actor Docente para realizar el registro de calificaciones de estudiantes.
Tipo	Primario
	Curso Normal de Eventos:
Acción de los Actores:	<ol style="list-style-type: none"> 1. El Docente prepara los boletines de calificaciones de los Estudiantes. 2. El Docente desea registrar las calificaciones de estudiantes que tiene a cargo. 4. El Docente ingresa calificaciones al Sistema. 6. El Docente confirma los datos. 9. El Docente guarda el documento PDF. 10. Finalmente las calificaciones de Estudiantes ya están registrados en el Sistema.
Respuesta del Sistema	<ol style="list-style-type: none"> 3. El Sistema despliega un formulario de estudiantes que tiene a cargo el Docente. 5. El Sistema solicita confirmación para guardar los datos. 7. El sistema guarda los datos. 8. El Sistema genera un documento PDF de los datos guardados.
	Excepciones:
Acción de los Actores	El docente no registrará las calificaciones de Estudiantes que no estén en el Sistema.

Respuesta del Sistema	El sistema no realizará el registro si los datos son ingresados incorrectamente.
------------------------------	--

Fuente: (Elaboración propia)

Figura 3.6 Caso de uso reportes



Fuente: (Elaboración propia)

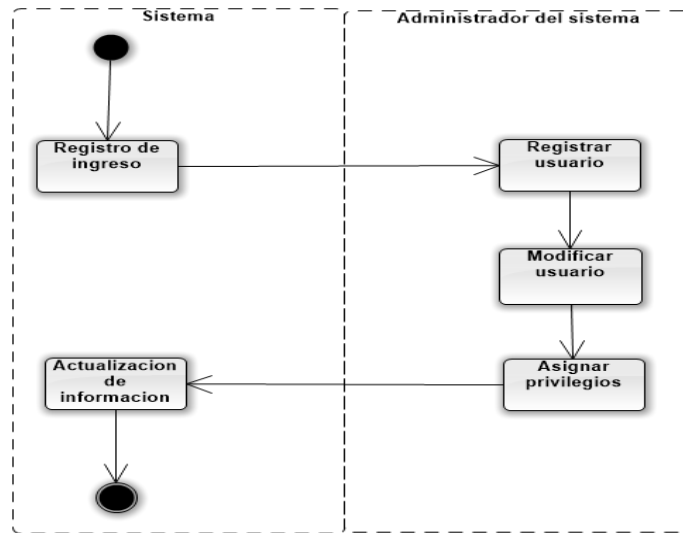
Tabla 3.10 Descripción de Caso de Uso: Reportes

Caso de Uso	Caso de uso reportes
Actores	Personal (Dirección académica), Estudiante.
Resumen	El Personal (Dirección académica) y Estudiante, podrán visualizar toda la información concerniente al estudiante.
Precondiciones	Debe ser un actor que este autenticado por el Sistema.
Tipo	Secundario.
	Curso Normal de Eventos:
Acción de Actores:	<ol style="list-style-type: none"> 1. El actor desea visualizar esta información. 3. El actor elige una opción.
Respuesta del Sistema	<ol style="list-style-type: none"> 2. El Sistema despliega un menú de información que desee ver del estudiante. 4. El Sistema muestra la información seleccionada.

Excepciones:	
Respuesta del Sistema	El sistema no permite modificaciones al Visualizar la Información del Estudiante.

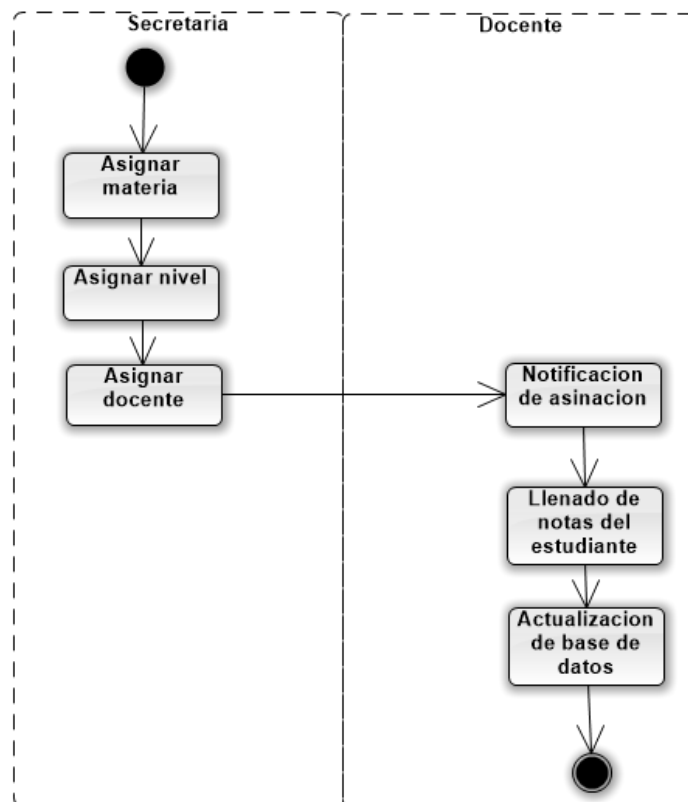
Fuente: (Elaboración propia)

Figura 3.9 Diagrama de actividades Gestión de usuarios



Fuente: (Elaboración propia)

Figura 3.10 Diagrama de actividades Gestión de notas

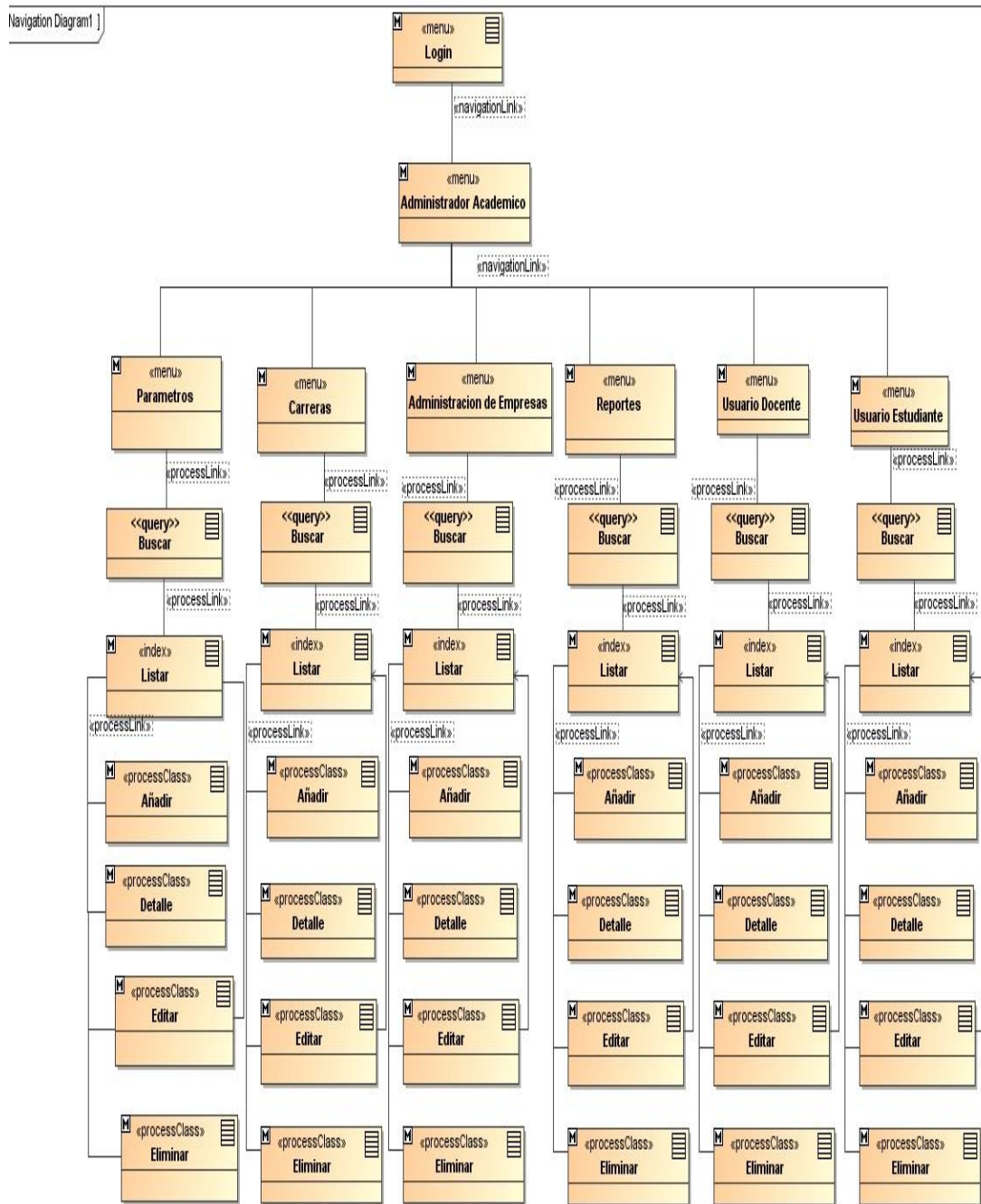


Fuente: (Elaboración propia)

3.7 Diseño de navegación

3.7.1 Modelo de navegación: Administrador Académico

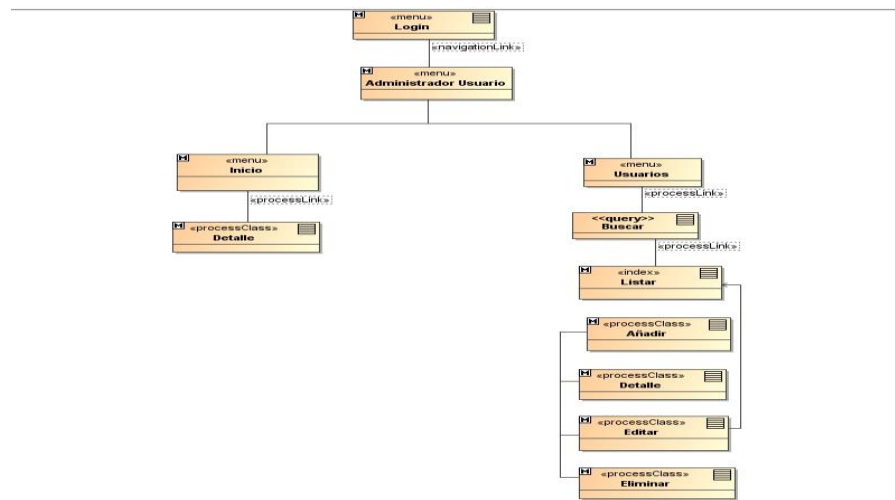
Figura 3.11 Modelo de navegación: Administrador Académico



Fuente: (Elaboración propia)

3.7.2 Modelo de navegación: Administrador Usuario

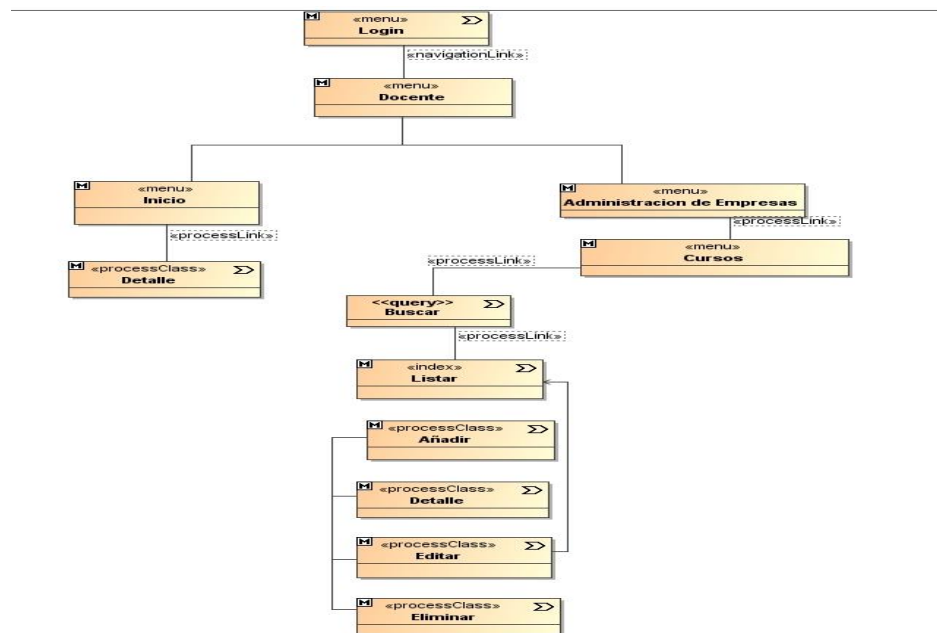
Figura 3.12 Modelo de navegación: Administrador Usuario



Fuente: (Elaboración propia)

3.7.3 Modelo de navegación: Docente

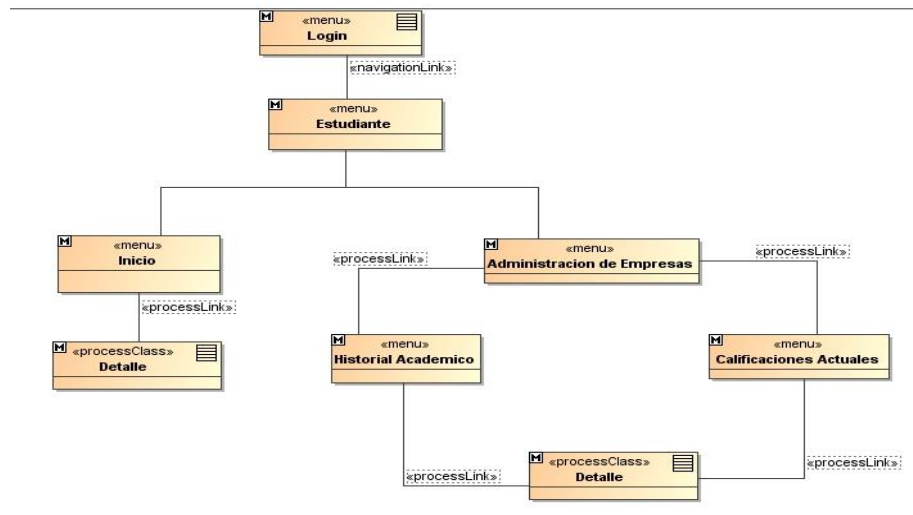
Figura 3.13 Modelo de navegación: Docente



Fuente: (Elaboración propia)

3.7.4 Modelo de navegación: Estudiante

Figura 3.14 Modelo de navegación: Estudiante

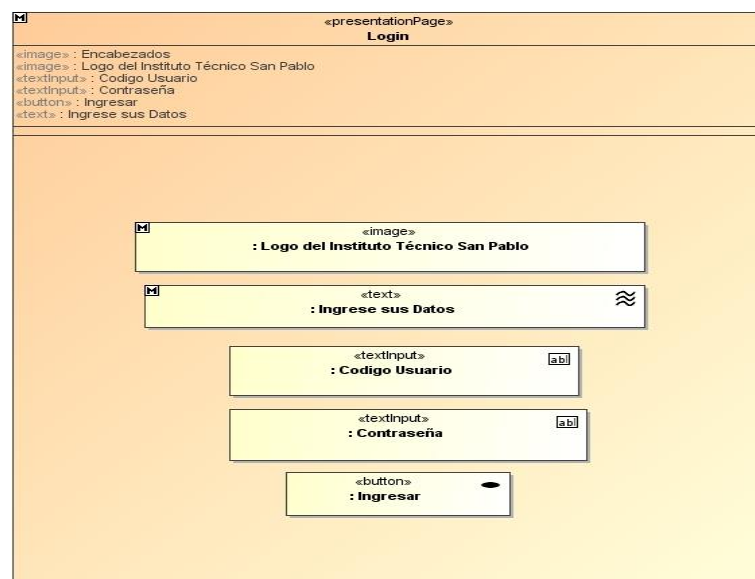


Fuente: (Elaboración propia)

3.8 Diseño de presentación

3.8.1 Modelo de presentación: Login (inicio de sesión)

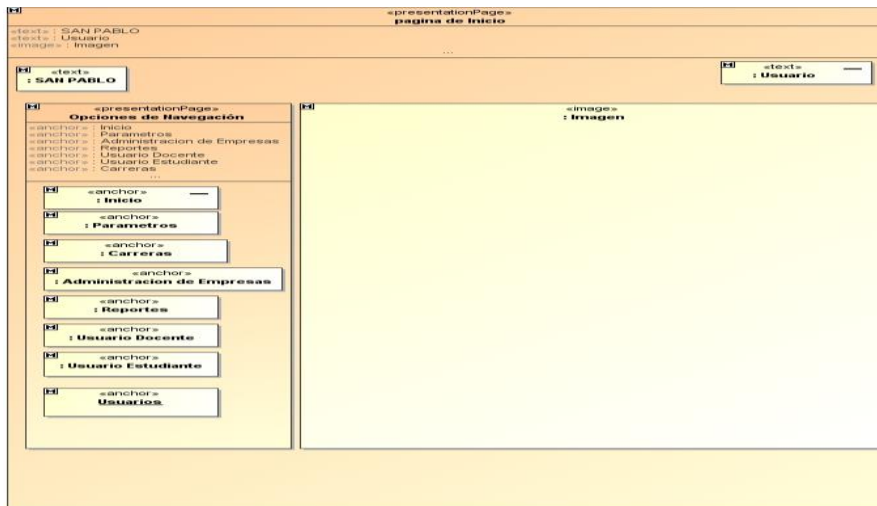
Figura 3.15 Modelo de presentación: Login (inicio de sesión)



Fuente: (Elaboración propia)

3.8.2 Modelo de presentación: Inicio

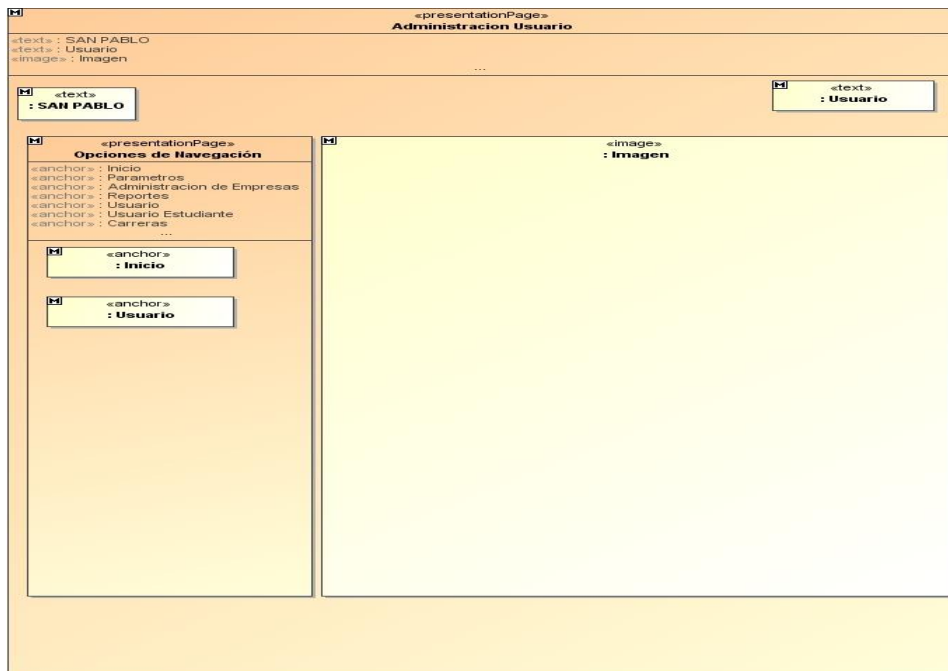
Figura 3.16 Modelo de presentación: Inicio



Fuente: (Elaboración propia)

3.8.3 Modelo de presentación: Usuario Administrador

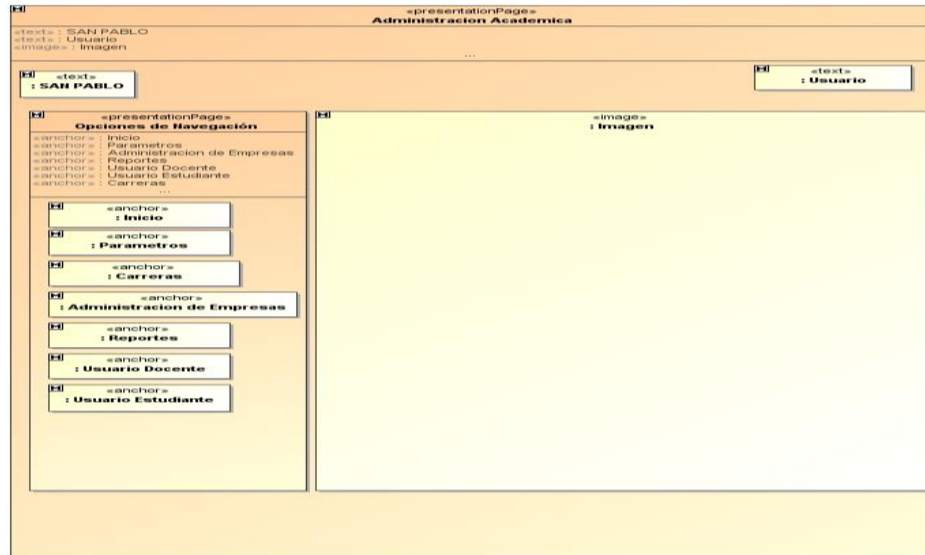
Figura 3.17 Modelo de presentación: Usuario



Fuente: (Elaboración propia)

3.8.4 Modelo de presentación: Usuario Académico

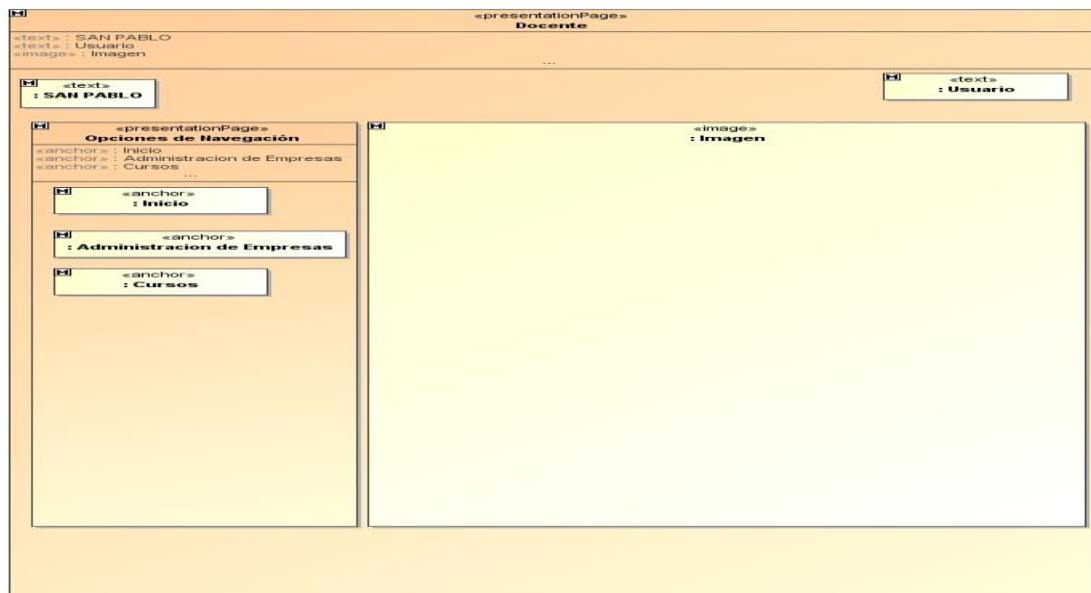
Figura 3.18 Modelo de presentación: Usuario Académico



Fuente: (Elaboración propia)

3.8.5 Modelo de presentación: Docente

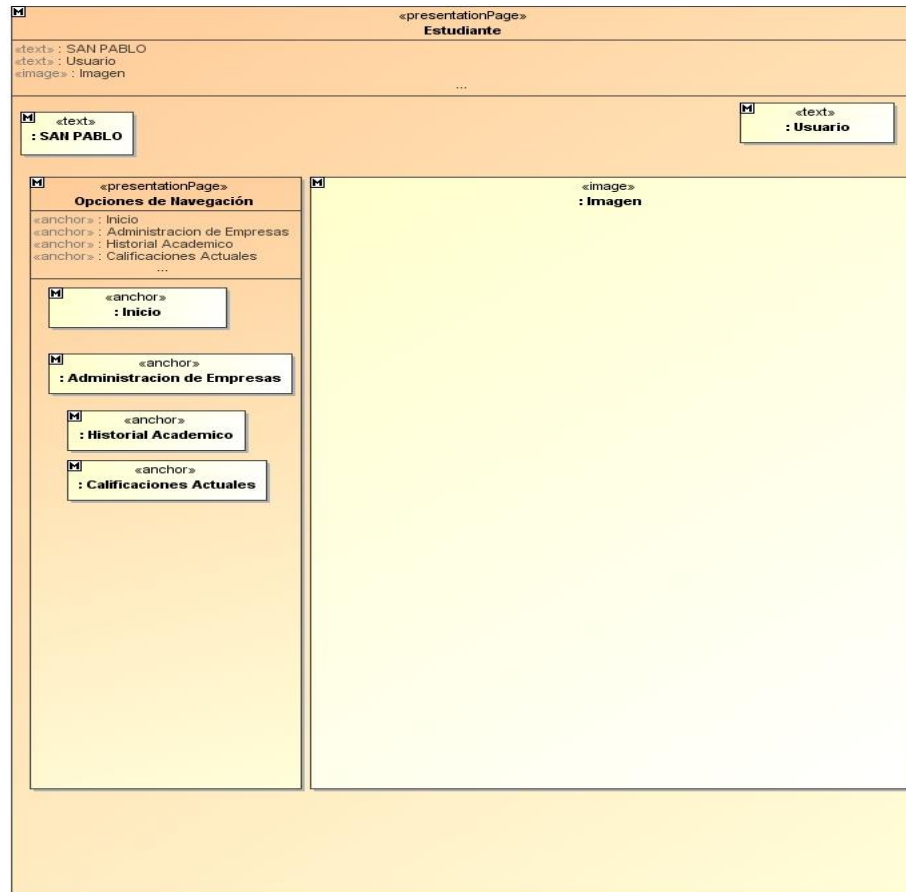
Figura 3.19 Modelo de presentación: Docente



Fuente: (Elaboración propia)

3.8.6 Modelo de presentación: Estudiante

Figura 3.20 Modelo de presentación: Estudiante



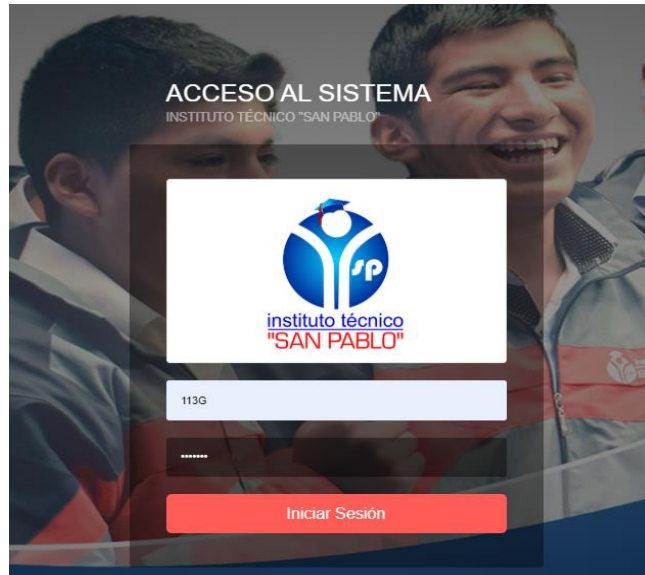
Fuente: (Elaboración propia)

3.8.7 Implementación del sistema

El desarrollo de aplicaciones requiere de metodologías acorde a las características de la plataforma donde sean ejecutadas. La ingeniería web propone nuevas metodologías orientadas al desarrollo y modelación de los procesos asociados a aplicaciones que se ejecutan en la Word Wide Web presentado mediante el modelado UWE ya que es una metodología basada en UML que tiene como finalidad especificar de una manera clara y conocida de una aplicación Web.

3.8.8 Interfaz Inicio de Sesión

Figura 3.21 Inicio de sesión



Fuente: (Elaboración propia)

En la Figura 3.22 observamos el inicio de la interfaz de inicio de sesión, donde el usuario debe autenticarse para acceder al sistema.

Figura 3.22 Menú principal Administrador



Fuente: (Elaboración propia)

En la Figura 3.23 se muestra el menú principal del administrador del sistema.

Figura 3.23 Menú principal Académico



Fuente: (Elaboración propia)

En la Figura 3.24 se aprecia el menú principal del administrador Académico.

Figura 3.24 Menú principal Docente



Fuente: (Elaboración propia)

La Figura 3.25 muestra el menú principal del Docente.

Figura 3.25 Menú principal Estudiante

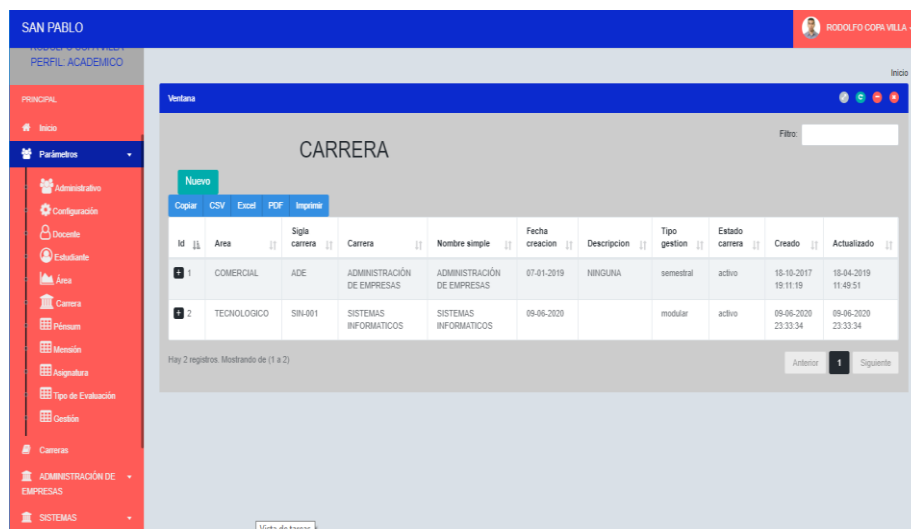


Fuente: (Elaboración propia)

En la Figura 3.26 observamos el menú principal del Estudiante.

3.8.9 Módulos que integran el sistema

Figura 3.26 Menú de parámetros Administrador Académico



Fuente: (Elaboración propia)

Figura 3.27 Pantalla lista de Estudiantes

Id	Paterno	Materno	Nombre	Pais nacimiento	Departamento nacimiento	Provincia nacimiento	Localidad nacimiento	Tipo documento	Numero documento	Expedido
1	LOAYZA	MORALES	RODRIGO	BOLIVIA	LA PAZ	MURILLO	LA PAZ	CI	10109282	LP
2	FLORES	MARTINES	CARLOS	BOLIVIA	LA PAZ	MURILLO	LA PAZ	CI	3466464	LP
3	TICONA	LOAYZA	MARIBEL	Bolivia	LA PAZ	Murillo	EL ALTO	CI	12458978	LP
4	SENTEÑO	MAMANI	GABRIEL	BOLIVIA	LA PAZ	MURILLO	EL ALTO	CI	9104582	LP
5	VERRIOS	HUANCA	MIRIAM	BOLIVIANA	LA PAZ	Murillo	Murillo	CI	4586951	LP
6	CHINO	MAMANI	WILMER	BOLIVIA	LA PAZ	MURILLO	EL ALTO	CI	568847	LP
7	HUANCA	POSTO	HERNAN	Bolivia	LA PAZ	Murillo	Murillo	CI	1254698	LP
8	CONDORI	MAMANI	LISBETH	Bolivia	LA PAZ	Murillo	Murillo	CI	12568975	LP
9	MORALES	ROQUE	ELENA	Bolivia	La Paz	Murillo	Murillo	CI	1568975	LP
10	MARCA	ICHUTA	YAMIL	Bolivia	La Paz	Murillo	Murillo	CI	9004563	LP

Fuente: (Elaboración propia)

En la Figura 3.28 observamos la lista de los estudiantes registrados en el sistema.

Figura 3.28 Pantalla lista de Docente

Id	Paterno	Materno	Nombre	Grado academico	Pais nacimiento	Departamento nacimiento	Provincia nacimiento	Localidad nacimiento	Tipo documento	Numero documento
1	ROMERO	QUILLA	LORENA	LIC.	BOLIVIA	LA PAZ	MURILLO	LA PAZ	CI	2099882
2	Ticona	Sanga	Juan Pablo	Técnico Medio	Bolivia	La Paz	Murillo		CI	1562489

Hay 2 registros. Mostrando de (1 a 2)

Anterior 1 Siguiente

Fuente: (Elaboración propia)

En la Figura 3.29 observamos la lista de los docentes registrados en el sistema.

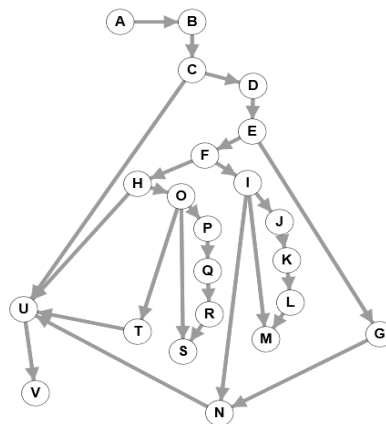
3.9 Pruebas de Software

La construcción de software implica conocimiento, experiencia, talento, capacidad intelectual, las pruebas que se realizaron fueron para detectar el máximo de errores del software.

3.9.1 Pruebas de Caja Blanca

Esta prueba se orienta al cálculo de las regiones que deben ser consideradas como partes independientes del sistema, se realizó los pasos de las entradas del presente trabajo denominado Sistema de Información Web para el Control y Seguimiento Académico, contando desde es el login que es la verificación de usuario A, hasta la salida del sistema V que se ejecutan cada una de las regiones, asegurando así que cada región se ejecuta al menos una vez. De forma general, se debe seguir:

Figura 3.29 Caja blanca – Técnica Complejidad Dicromática



Fuente:(Elaboración propia)

Complejidad Ciclomática de acuerdo a:

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

$$V(G)=P+1$$

Donde:

N= Numero de Nodos. **A=** Numero de aristas **P=** Numero de Nodos Predicados

$$N=22$$

$$A=28$$

$$P=7$$

Remplazando los valores tenemos:

$$V(G) = 28 - 22 + 2$$

$$V(G) = 7 + 1$$

$$V(G) = 8$$

$$V(G) = 8$$

Como vemos el resultado iguala en los tres casos de suma de la matriz y remplazando los valores del ciclo maticas.

El valor de $V(G) = 8$ nos indica que son ocho los casos de pruebas que deben de ejecutarse y diseñar para garantizar que se cubren las sentencias del programa.

Sacamos caminos independientes:

Camino 1: A-B-C-U-V

Camino 2: A-B-C-D-E-G-N-U-V

Camino 3: A-B-C-D-E-F-I-J-K-L-M-N-U-V

Camino 4: A-B-C-D-E-F-H-O-P-Q-R-S-T-U-V

Camino 5: A-B-C-D-E-F-H-O-S-T-U-V

Camino 6: A-B-C-D-E-F-H-U-V

Camino 7: A-B-C-D-E-F-I-N-U-V

Camino 8: A-B-C-D-E-F-H-O-T-U-V

3.9.2 Pruebas de Caja Negra

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 interfaces mostradas a continuación.

Tabla 3.11 Caso de Prueba Ingreso al Sistema Usuario y Pantalla

Caso de Prueba de Ingreso al Sistema Usuario y Pantalla	
Código:	Caso de Prueba 01
Descripción de la Prueba:	Acceder al Sistema usuario y contraseña Permite realizar la verificación del Usuario si tiene autorización al ingresar.
Condiciones de ejecución:	El usuario debe encontrarse en la página para ser aceptado previamente ingresa con su usuario y contraseña.
Pasos de Ejecución:	<ul style="list-style-type: none">- El Sistema muestra la pantalla de autenticación usuario y contraseña.- Los usuarios del Sistema tienen permitido realizar esta tarea, llenar el formulario y presionar el botón de ingresar para que sea habilitado por el sistema.- El usuario ingresa al Sistema para realizar el trabajo correspondiente. <p>El Sistema despliega la pantalla de inicio de Sistema para el usuario correspondiente.</p>
Resultados Esperados:	El Sistema verifica al usuario permitiéndole el acceso al Sistema de acuerdo al nivel de usuario.
Evaluación de Prueba:	Se realiza el ingreso al Sistema con la verificación previa, de manera satisfactoria.

Fuente: (Elaboración Propia)

Tabla 3.12 Caso de Prueba Registrar Personal

Caso de Prueba de Registrar Personal	
Código:	Caso de Prueba 02
Descripción de la Prueba:	Registrar Personal. Permite realizar el registro de diferentes Usuarios (personal) al Sistema.
Condiciones de ejecución:	El Personal (usuario) debe estar verificado por el Sistema para realizar dicha documentación.
Pasos de Ejecución:	<ul style="list-style-type: none">- El Sistema muestra un formulario de registro del personal (Usuario).- El Personal (Director), encargado de realizar el seguimiento y llena los datos al formulario.- El Personal (estudiante) es registrado en el formulario del sistema.
Resultados Esperados:	Registrar al Personal (Usuario) en el Sistema.
Evaluación de Prueba:	Se realizó el registro del Personal correctamente.

Fuente: (Elaboración Propia)

Tabla 3.13 Caso de Prueba Registrar Estudiante

Caso de Prueba Registrar Estudiante	
Código:	Caso de Prueba 03
Descripción de la Prueba:	Permite realizar el registro de Estudiante al Sistema

Condiciones de ejecución:

El Personal (Dirección académica) debe estar autenticado por el Sistema para realizar el registro del Estudiante.

Pasos de Ejecución:

- El Sistema muestra un formulario de registro de Estudiante.
- El Personal (Dirección académica), encargado de realizar este registro llena los datos al formulario.
- El Personal (Dirección académica) después de llenar los datos envía el formulario.
- El Sistema despliega un mensaje de confirmación, el Personal (Dirección académica) confirma él envió.
- El Sistema almacena los datos.

El Estudiante, queda registrado en el Sistema.

Resultados Esperados:

Registrar al Estudiante en el Sistema.

Evaluación de Prueba:

Se realizó en registro del Estudiante correctamente.

Fuente: (Elaboración Propia)

CAPITULO IV

MÉTRICAS DE

CALIDAD Y ESTIMACION

DE COSTO

4 MÉTRICAS DE CALIDAD, ESTIMACIÓN DE COSTO Y SEGURIDAD

4.1 Introducción

En el presente capítulo se describirán los estándares para el ciclo de vida del Software mediante la ISO 9126, se realizaron pruebas después de la codificación para detectar el máximo de errores y poder subsanarlos.

4.2 Métricas de calidad

Se realizará la medición de calidad del software mediante la métrica de ISO 9126, que establece cualquier componente de la calidad de software puede ser descrito en términos de una de seis características básicas las cuales son: funcionalidad, confiabilidad, usabilidad, eficiencia, mantenimiento y portabilidad.

4.2.1 Funcionabilidad

Este atributo valora las características y capacidades del programa, la generalidad de las funciones entregadas y la seguridad del sistema global. La funcionalidad es el grado en que el sistema satisface las necesidades que indican los siguientes sub atributos: estabilidad, exactitud, interoperabilidad, cumplimientos de seguridad.

Punto de función: Para el cálculo de punto función se toma en cuenta cinco características, el dominio de información, como son números de entrada, salida, condiciones, archivos e interfaz externa. Luego se realiza el cálculo de punto de función hallando la suma de estas características, parámetros de medición y el factor de ponderación también llamado punto medio de ponderación.

Número de entrada de usuario: Se cuenta cada entrada de usuario que proporciona datos al sistema.

Número de salida de usuario: Se refiere cada salida que proporciona el sistema al usuario. Entre estos pueden ser informes, reportes y mensajes advertencia, notificaciones y errores.

Número de archivo: Se toma en cuenta cada archivo, estos pueden ser grupos lógicos de datos (tablas de base de datos).

Número de interfaces externas: Se cuentan todas las interfaces legibles por la máquina. Aplicando lo anterior al proyecto se tiene los siguientes datos:

Tabla 4.1 Parámetros de medición

Parámetro de medición	Cuenta
Número de entradas de usuario	22
Número de salidas de usuario	24
Número de consultas de usuario	18
Número de archivos	12
Número de interfaces externas	2

Fuente: (Elaboración propia)

Para calcular el punto de función se tiene que realizar el cálculo de la cuenta total con los factores de ponderación especificados en la siguiente tabla:

Tabla 4.2 Parámetros de medición

Parámetro de medición	Cuenta	Factor	total
Número de entradas de usuario	22	4	88
Número de salidas de usuario	24	5	120
Número de consultas de usuario	18	4	72
Numero de archivos	12	10	120
Numero de interfaces externas	2	7	14
Cuenta Total			414

Fuente: (Elaboración propia)

En la **Tabla 4.2** se muestra la cuenta total que se obtiene de la sumatoria de los factores de ponderación a los parámetros de medición.

Para determinar los valores de ajustes de complejidad se indica según se corresponda a las preguntas de la siguiente tabla:

Tabla 4.3 Parámetros de medición

Importancia	0%	20%	40%	60%	80%	100%	Fi
Escala	No Influencia	Incidental	Moderado	Medio	Significativo	Esencial	
Factor	0	1	2	3	4	5	
1. ¿Requiere el sistema copias de seguridad y de recuperación fiables?					X		4
2. ¿Se requiere comunicación de datos?				X			3
3. ¿Existen funciones de procesamiento distribuido?				X			3
4. ¿Es crítico el rendimiento?				X			3
5. ¿Se ejecuta el sistema en un entorno operativo existente y fuertemente utilizado?					X		4
6. ¿Requiere el sistema entrada de datos interactiva?					X		4
7. ¿Requiere la entrada de datos interactiva que las transacciones de entrada se lleven a cabo sobre múltiples pantallas u operaciones?			X				2
8. ¿Se actualizan los archivos maestros de forma interactiva?			X				2
9. ¿Son complejos las entradas, las salidas, los archivos o peticiones?				X			3
10. ¿Es complejo el procesamiento interno?				X			3
11. ¿Se ha utilizado el código para ser reutilizable?				X			3
12. ¿Están incluidas en el diseño la conversión y la instalación?					X		4

13. ¿Se ha diseñado el sistema para soportar múltiples instalaciones en diferentes organizaciones?	X	4
14. ¿Se ha diseñado la aplicación para facilitar los cambios y para ser fácilmente utilizada por el usuario?	X	4
TOTAL		46

Fuente: (Elaboración Propia)

Calculando el punto de función mediante la siguiente ecuación:

$$PF = CuentaTotal * (0,65 + 0.1 * \sum Fi)$$

Donde:

Cuenta total: es la suma del producto del factor de ponderación y valores de los parámetros.

$\sum Fi$: es la sumatoria de los valores de ajuste de la complejidad.

Calculando:

$$PF = 414 * (0,65 + 0.01 * 46)$$

$$PF = 414 * 1.11$$

$$PF = 459,54$$

Consideramos el máximo valor de complejidad $\sum Fi$: **70** calculamos al 100% el nivel de confianza de la siguiente manera:

$$PF_{max} = CuentaTotal * (0,65 + 0.1 * \sum Fi)$$

$$PF_{max} = 414 * (0,65 + 0.01 * 70)$$

$$PF_{max} = 414 * 1.35$$

$$PF_{max} = 558,9$$

La relación obtenida entre ambos es la funcionalidad:

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

$$\mathbf{Funcionalidad} = \frac{459.54}{558.9} = 0.822$$

$$\mathbf{Funcionalidad} = 0.822 * 100 = 82,2\%$$

Por lo que se concluye que la funcionalidad del sistema es un 82.2 %, esto requiere decir que el sistema tiene un 82.2 % de funcionar sin riesgos a fallar con operatividad constante y un 17,8 % aproximadamente de colapso del sistema.

4.2.2 Confiabilidad

La confiabilidad del sistema se define como la probabilidad de operación libre de fallos de un programa o computadora.

Donde se encuentra:

$P(T \leq t)$ Probabilidad de fallas (el termino en el cual sistema trabaja sin fallas)

$P(T \leq t) = 1 - F(t)$ Probabilidad de trabajo sin fallas (Tiempo en el cual no falla el sistema)

Para calcular la confiabilidad del sistema se toma en cuenta el periodo de tiempo en el que se ejecuta y se obtiene muestras.

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

Donde:

f : Funcionalidad del sistema.

μ : Es la probabilidad de error que puede tener el sistema.

t : Tiempo de duración de gestión en el sistema.

Para lo que consideramos un periodo de 20 días como tiempo de prueba donde se define que cada 10 ejecuciones se presenta una falla.

Calculando:

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

$$F(t) = 0,822 * e^{(-\frac{1}{10} * 20)}$$

$$F(t) = 0,111 * 100 = 11,1\%$$

Reemplazando en las fórmulas de probabilidades:

$$P(T \leq t) = F(t) \quad \rightarrow \quad P(T \leq t) = 0,111 = 11,1\%$$

$$P(T \leq t) = 1 - F(t) \quad \rightarrow \quad P(T \leq t) = 1 - 0,111$$

$$P(T \leq t) = 0,889 = 88,9\%$$

Por lo tanto, la confiabilidad del sistema es el 88,9% en un periodo de 20 días como tiempo de prueba.

4.2.3 Usabilidad

Para conocer si el sistema satisface los requerimientos establecidos por el usuario, se realiza una evaluación del mismo en base a encuestas planteadas a los usuarios del sistema, los cuales califican en una ponderación al 100% los usuarios tienen conocimiento de los procesos que realizan y los resultados se refleja en la **Tabla 4.5**.

Para determinar la usabilidad del sistema se utiliza la siguiente ecuación:

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

Donde:

Xi: Es la sumatoria de valores

n: Es el número de preguntas

Para responder a las preguntas se debe considerar la siguiente tabla:

Tabla 4.4 Parámetros de medición

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

Fuente: Elaboración Propia

Tabla 4.5 Parámetros de medición

Nro	Preguntas	SI	NO	Evaluación
1	¿Puedo utilizar con facilidad el sistema?	5	0	1
2	¿Puedo controlar operaciones que el sistema solicite?	4	1	0,8
3	¿El sistema permitió la retroalimentación de información?	4	1	0,8
4	¿El sistema cuenta con interfaz amigable a la vista?	5	0	1
5	¿La respuesta del sistema es satisfactoria?	5	0	1
6	¿Le parecen complicadas las funciones del sistema?	4	1	0,8
7	¿Los resultados que proporciona el sistema facilitan el trabajo?	5	0	1

8	¿Durante el uso del sistema se produjo errores?	2	3	0,4
Total				6,8

Fuente: Elaboración Propia

Calculamos la usabilidad con la ecuación anterior:

$$FU = \left[\left(\frac{6,8}{8} \right) * 100 \right]$$

$$FU = [0,85 * 100]$$

$$FU = 85\%$$

Por lo tanto, existe un 85% de comprensión o entendimiento de los usuarios con respecto a la capacidad del sistema.

4.2.4 Mantenibilidad

El mantenimiento se da las modificaciones del sistema a los nuevos requerimientos según los usuarios de la institución.

Por lo que el índice de madurez del software (IMS) se determina con la siguiente ecuación:

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

Donde:

Tabla 4.6 Valores para determinar la mantenibilidad

Descripción	Valor
<i>Mt</i> = Número de módulos de la visión actual	4
<i>Fc</i> = Número de módulos en la versión actual que se han modificado	1
<i>Fa</i> = Número de módulos en la versión actual que se han añadido	0

F_d = Número de módulos de la anterior versión que se han borrado en la versión actual. 0

Fuente: Elaboración Propia

Calculando:

$$IMS = \frac{[4-(1+0+0)]}{4}$$

$$IMS = 0.75 * 100 = 75\%$$

Por lo tanto, se puede decir que el sistema tiene un índice de mantenibilidad de 75% que es la facilidad de mantenimiento, el 25% restante es el margen de error corresponde a los cambios y modificaciones que se realizan al sistema.

4.2.5 Portabilidad

El sistema actual está en plataforma de Windows ejecutable desde cualquier plataforma debido a su diseño adaptable el único requisito es que el dispositivo cuente con internet y un navegador.

El sistema del presente proyecto por estar diseñado en un entorno de acceso vía web mide la portabilidad en: lado del servidor y lado del cliente, la portabilidad del software se enfoca en tres aspectos:

- Hardware del servidor
- Sistema operativo del servidor
- Software del servidor

Por lo mencionado anteriormente el sistema es portable en sus diferentes entornos tanto en hardware y software.

4.3 Estimación de costo de Software

Existen distintos métodos para la estimación de costes de desarrollo de software, estos métodos no son otra cosa que establecer una relación matemática entre el esfuerzo y el tiempo de desarrollo.

4.3.1 Método de estimación COCOMO II

La estimación de costos del sistema ha sido desarrollada bajo KLDC (Kilo-Líneas de código) como de detalle a continuación:

$$KLDC=LDC/1000$$

$$KLDC=12,065/1000$$

$$KLDC=12,065 \text{ KLDC}$$

Por lo que la evaluación del sistema ha sido considerada bajo las 12,065 KLDC. Los coeficientes que se usaran los valores que se detallan en la siguiente tabla:

Tabla 4.7 Coeficientes del modelo COCOMO II

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

Fuente (S. Pressman, 2010)

Ecuaciones para calcular el costo de Software:

Tabla 4.8 Ecuaciones del Modelo COCOMO II

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

Fuente (Prentice- Hall, 1981)

Para hallar los valores de FAE se utiliza la **Tabla 4.9**

Tabla 4.9 Calculo de los atributos FAE

Atributos que afectan al Coste	Valor					
	Muy bajo	Bajo	Nominal	Alto	Muy alto	Extra alto
Atributos del Software						
Fiabilidad del software	0,75	0,88	1,00	1,15	1,40	
Tamaño base de datos		0,94	1,00	1,08	1,16	
Complejidad del producto	0,70	0,85	1,00	1,15	1,30	1,65
Atributos del Hardware						
Restricciones de tiempo de ejecución			1,00	1,11	1,30	1,66
Restricciones de memoria			1,00	1,06	1,21	1,56
Volatilidad de 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	
Capacidad de programadores	1,42	1,17	1,00	0,86	0,70	
Experiencia de S.O. usado	1,21	1,10	1,00	0,90		
Experiencia en el lenguaje de programación	1,14	1,07	1,00	0,95		
Atributos del proyecto						
Uso de técnicas actuales de programación	1,24	1,10	1,00	0,91	0,82	
Uso de herramientas de software	1,24	1,10	1,00	0,91	0,83	
Restricciones de tiempo de desarrollo	1,23	1,08	1,00	1,04	1,10	
TOTAL, FAE=	0,377					

Fuente: (Elaboración Propia)

Aplicando las ecuaciones (descritas en la **Tabla 4.8**) así como los coeficientes a y c y los exponentes b y d que en nuestro caso el tipo orgánico será el más apropiado ya que el número de líneas de código no supera los 50 KLDC, (descritos en la **Tabla 4.7**) y el cálculo de los atributos FAE (descrito en la **Tabla 4.9**) Se tiene:

Calculando el Esfuerzo:

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

$$E = 2.4 * (12,065)^{1,05} * 0,377$$

$$E = 12,36 \text{ Personas/Mes}$$

Calculando el Tiempo:

$$T = c * (E)^d$$
$$T = 2,5 * (12,36)^{0,38}$$
$$T = 6,49 \text{ Equivale a 6 Meses}$$

Calculando el Personal Promedio:

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

$$NP = \frac{12,36}{6,49}$$

$$NP = 1,90 \text{ Equivale a 2 Personas}$$

Calculando el Costo Total:

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

$$CT = 500 * 2 * 6$$

$$CT = 6000 \text{ \$us}$$

Entonces se requiere estimando 2 personas un trabajo de 6 meses para el desarrollo del sistema con un costo total de 6000 \$ dólares.

4.4 Sistema de gestión de seguridad de la información ISO-27002

La iso-27002 evalúa y rectifica la implementación mediante el cumplimiento de las normas, así como la mejora continua de un conjunto de controles que permiten reducir el riesgo de sufrir incidentes de seguridad en el funcionamiento de la institución en cuanto a la seguridad de la información, para lo cual se tomó los siguientes tipos de seguridad:

4.4.1 Seguridad Lógica

➤ **Copias de seguridad:**

El Administrador Académico deberá cambiar el password del sistema periódicamente 1 vez cada 20 días o 1 vez al mes.

En caso de ser usuario se recomienda cambiar el password periódicamente.

➤ **Identificación y autenticación:**

Permite prevenir el ingreso de personas que no son usuarios, para ello el sistema cuenta con un control estricto en el ingreso con un Usuario y una contraseña estrictamente controlada.

➤ **Encriptación:**

Se aplica la encriptación de seguridad para la contraseña, un dato de suma importancia para el ingreso al sistema, de este modo se está utilizando lo que es el algoritmo de Hash una encriptación de alta seguridad.

4.4.2 Seguridad Física

➤ **Seguridad física y del entorno:**

Se prevé recomendación de los back-up 3 o hacer copias que sean almacenadas en distintos lugares.

Los back-up de la base de datos deberán ser protegidos en áreas seguras, además será permitido el acceso al personal autorizado.

➤ **Equipamiento:**

Una adecuada protección física y mantenimiento permanente de los equipos e instalaciones que conforman los activos de la institución.

➤ **Control de acceso físico al área de Sistemas:**

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

4.4.3 Seguridad Organizativa

La información referente al sistema debe recibir un nivel de protección apropiada como ser:

➤ **Gestión de archivos**

Etiquetar y manejar los Back-ups de acuerdo a la fecha en que se realizaron los mismos.

➤ **Recursos Humanos**

Una vez que el personal académico que interviene en el proceso concluya con el contrato de prestación de servicio en la institución o por alguna razón fue despedido se deberá dar de baja el acceso al sistema.

CAPITULO V

CONCLUSIONES Y

RECOMENDACIONES

5 CONCLUSIONES Y RECOMENDACIONES

En la culminación del presente proyecto de grado, se llegaron a las siguientes conclusiones y recomendaciones que se citan a continuación:

5.1 Conclusiones

Se concluye con los objetivos planteados habiendo realizado un estudio del sistema actual en el proceso de busca de información de la gestión académica, tomando en cuenta que con el presente proyecto se logró centralizar la información y permite realizar un seguimiento académico en la institución coadyuvando en el cumplimiento de la gestión académica del Instituto Técnico San Pablo.

- Con el sistema académico se resolvió el problema de la automatización de los procesos y registro de información de estudiantes y docentes.
- Se realizó el módulo registro de calificaciones de estudiantes de manera que facilitara al docente en el registro de información la cual es guardada y actualizada.
- La asignación de notas a los estudiantes en las asignaturas que cursa se realiza de manera eficaz y eficiente, brindando al docente una herramienta en línea para el registro de información.
- El historial académico de notas de los estudiantes se realizó de manera oportuna para acelerar los procesos que requiera de este documento para bien de los estudiantes.

Logrando todos los objetivos específicos se concluye con el desarrollo e Implementación del “SISTEMA DE INFORMACION WEB PARA EL CONTROL Y SEGUIMIENTO ACADÉMICO”, por lo que es un aporte tecnológico ya que se redujo el tiempo de registros, consultas, búsquedas de la información perteneciente a la Institución, cabe recalcar que el manejo de esta información se realiza de forma segura y confiable.

5.2 Recomendaciones

Al haber concluido con el presente proyecto de grado, titulado “Sistema de Información Web para el Control y Seguimiento Académico”, se tiene las siguientes recomendaciones:

- Se recomienda mucha discreción en el manejo de sus usuarios y contraseñas ya que el sistema contiene información de mucha importancia.
- Para resguardar la información, el administrador académico del sistema debe realizar copias de seguridad de la base de datos para cada periodo que disponga la institución.
- Se recomienda documentar los procesos y cambios realizados para tener un registro adecuado y pueda ser utilizado para versiones futuras.

BIBLIOGRAFÍA

- (Daniel M.& Emilio J, 2007, pág.3) Metodología para el desarrollo de aplicaciones Web: UWE
- (Citlali G., Juan P., Víctor H. 2014, pág.137) UWE en Sistema de Recomendación de Objetos de Aprendizaje. Aplicando Ingeniería Web: Un Método en Caso de Estudio
- (Claudio Alberto Nipotti,2011, Pag 5) MySQL 8.0 Reference Manual
- (Hernán Beati. 2011. Pag.2) PHP Creación de Páginas Web Dinámicas
- (Javier Eguíluz, 2008, pág. 5) Introducción a CSS.
- (Roger S. Pressman, 2013, pág. 340) Ingeniería de Software. Un enfoque práctico 7ma Edición.
- Schawebe R & D. Olsina. (2008). *conceptos basicos de UWE*.
- Barry W. Boehm. (2001). Estimación de Costos de Software con COCOMO II.
Editorial Prentince Hall.
- (Adriana Gómez, María López, 2017, pág. 7) Un modelo de estimación de proyectos de software Cocomo II.

WEB

- (2008). En J. Eguiluz, *Intoduccion a CSS* (pág. 5).
- (2010). Obtenido de Introduccion a JavaScript:
<https://uniwebsidad.com/libros/javascript/capitulo-1>
- Airston G. (2019). *Aprenderaprogramar*. Obtenido de Lenguaje de Programacion PHP: <http://www.aprenderaprogramar.com/index.php?option>
- Antonio Javier Gallego Sanchez. (2018).

- Beati Hernan. (2011). PHP Creacion de Paginas Web Dinamicas. En H. Beati.
- BinErp. (2018). Obtenido de Metodo Iterativo Incremental: <http://proyectosagiles.org>
- Claudio Alberto Nipotti. (2011). En C. A. Nipotti, *MySQL Reference Manual* (pág. 5).
- *Concepto de Control*. (2018). Obtenido de Concepto de Control: www.conceptosdecontrol.com
- Diego Guzman. (28 de Junio de 2018). Obtenido de <http://codingornot.com/que-es-json>
- Elibeth Yuri. (2020). *monografias.com*. Obtenido de <https://www.monografias.com/trabajos14/control/control.shtml>
- Enrique Herrera Perez. (2003). Obtenido de <http://www.internet.com>
- Javier Egluz . (2008). Obtenido de <http://www.css.com>
- Jose M Garcia. (2016). Obtenido de <Http://www,google.com/amp/s/programacion>
- LMU - . (10 de 08 de 2016). *UWE - ingenieria web basada en UML*. Obtenido de <https://www.uwe.pst.ifi.lmu.de/idex.html>
- Lozano, L. A. (Septiembre de 2013). *Norma ISO-9126*. Obtenido de <http://www.austral.edu.ar>
- Maria Estela Raffino. (20 de Junio de 2020). *Sistema de Informacion*. Obtenido de <https://concepto.de/sistema-de-informacion/#ixzz6XUvTiSig>
- Maria Estela Raffino. (20 de Junio de 2020). *Sistema de Informacion* .

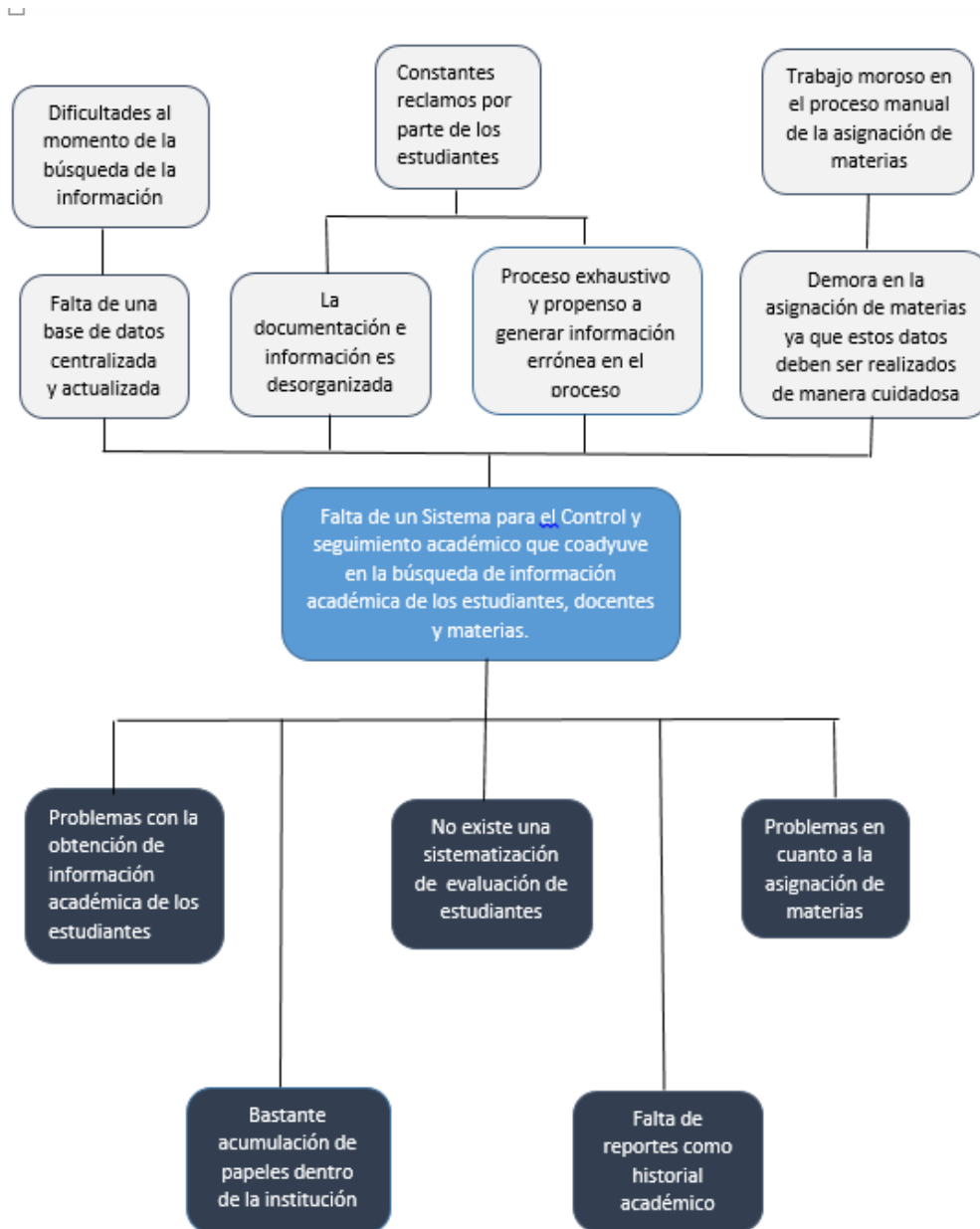
- Marina Estela Raffino. (21 de Junio de 2020). *Sistema*. Obtenido de <https://concepto.de/sistema/>
- Matthew Thomas. (2020). *Scribd*. Obtenido de <https://es.scribd.com/document/230591394/CONCEPTO-A-FORMATO-SEGUIMIENTO-ACADEMICO-docx>
- Miguel Angel Alvares. (25 de Marzo de 2009). *DesarrolloWeb.com*. Obtenido de <https://desarrolloweb.com/articulos/introduccion-jquery.html>
- Minguez Daniel y Garcia Emilio. (2007). En *Metodologia para el desarrollo de aplicaciones web UWE* (pág. 3).
- Mussa Yaser. (2017). *Manual de referencia*. Obtenido de <http://dev.mysql.com/doc/refman/5.7/en/>
- NUÑES. (2000). *Modelado de objetos con UML*. Obtenido de <http://exa.unne.edu.ar/informatica/anasistem1>
- *PEI Institucion Educativa La Leona*. (1954). Obtenido de <https://institucioneducativaleona.wordpress.com>
- Richos. (2018). *Laravel 5 Conceptos basicos*.
- Robert Mc.Cool. (1995). *Servidor Web Apache*. Obtenido de <Http://servidorapache.com>
- Roger S. Pressman. (2013). *Ingenieria del software un enfoque 7ma Edicion*. En R. S. Pressman, *Ingenieria de Software* (pág. 340). 7ma Edicion.
- Roja Daniela. (2018). Obtenido de <http://www.sistema.com>
- Ryan Bernard. (2010).
- Schawebe R & D. Olsina. (2008). *conceptos basicos de UWE*.

- *TECNOLOGIA&INFORMATICA*. (2019). Obtenido de <http://tecnologia-informatica.com/que-es-sistema-informatico/>
- Williams Maximillians. (2010). *Tutorial sobre UWE*. Obtenido de Tutorial sobre UWE: <http://uwe.pst.ifi.lmu.de/teachingTutorialSpanich.html>
- Woodman, L. (1985). *Information mangement in large organization*. Obtenido de <http://www.ecured.cu/Gestion-de-la-Informacion>

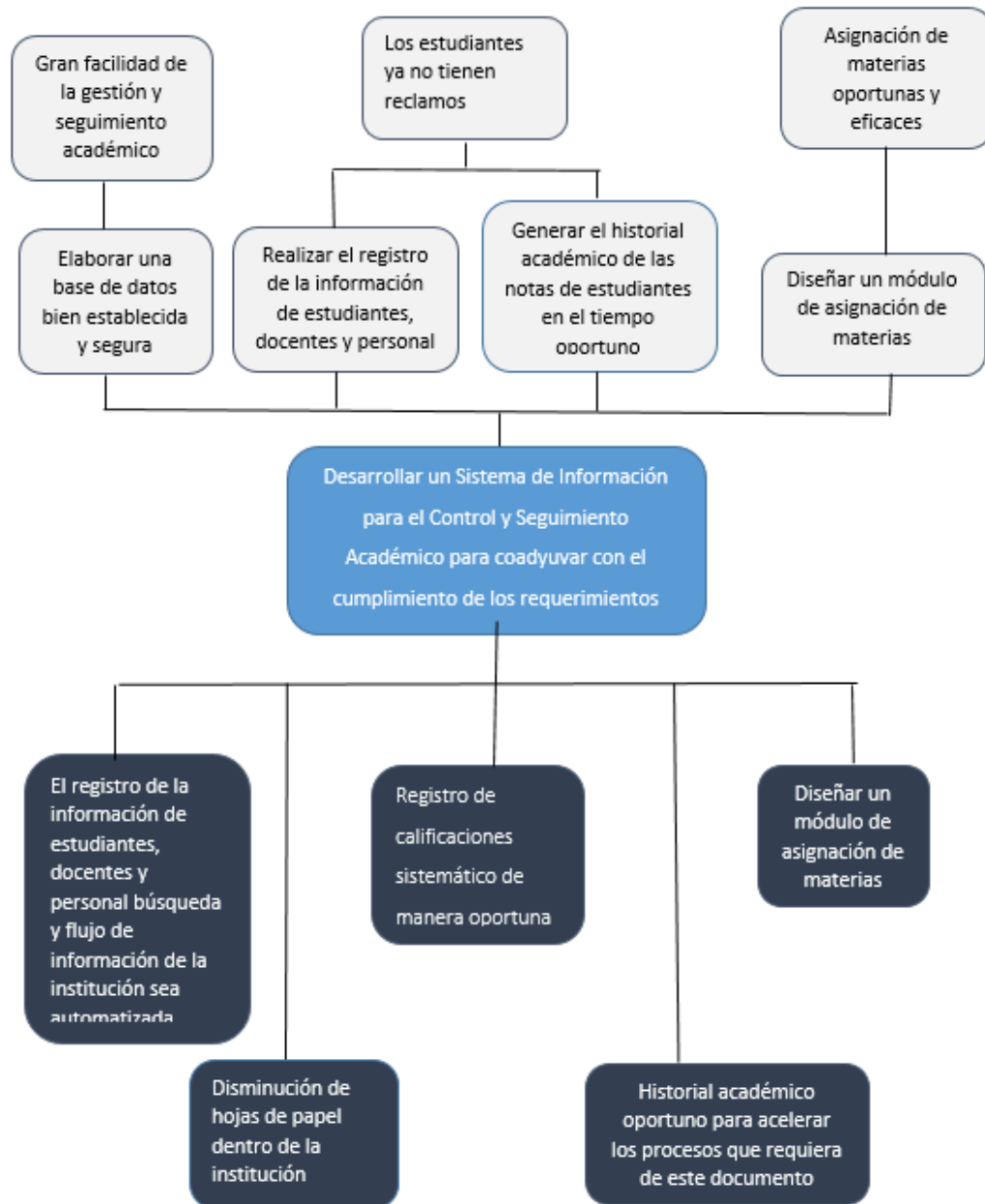
ANEXO

A

Anexo A: Árbol de Problemas



Anexo B: Árbol de Objetivos



ANEXO

B

ANEXO B.1

FORMULARIO DE LA ENTREVISTA INSTITUTO TÉCNICO SAN PABLO

Nombre del Entrevistado.....

Cargo que ocupa.....

Fecha/...../.....

A. ESTABLECER EL PERFIL DE USUARIO

1. ¿Cuáles son sus principales responsabilidades en el Proceso Académico?

2. ¿Qué producción Académica realiza?

3. ¿Para Quién?

4. ¿Cómo determina el éxito en lo que hace?

5. ¿Qué problemas interfieren con sus éxitos?

B. EVALUANDO EL PROBLEMA

6. ¿Cuál es la problemática a resolver?

7. ¿Por qué existe este problema?

8. ¿Cuáles son las características de sistema que se utiliza para resolver el problema actualmente?

9. ¿Cómo le gustaría que se resolviera?

C. EVALUANDO Y ENTENDIENDO EL AMBIENTE USUARIO

10. ¿Tiene los usuarios experiencia con aplicaciones informáticas?

11. ¿Qué herramientas de software se usan en la institución?

12. ¿Qué sistemas operativos se usan en la institución?

13. ¿Cuáles son sus expectativas con respecto a la facilidad de uso de un sistema de Software?

14. ¿Qué tipos de documentación impresa necesita?

D. RESUMEN PARA VALIDAR EL ENTENDIMIENTO DEL PROBLEMA

15. Describa con sus propias palabras los problemas que tiene con el sistema de gestión académico actual.

E. EVALUANDO LAS SOLUCIONES DEL ANALISTA

16. ¿Qué le parece si pudiéramos resolver el problema Desarrollando e implementado un sistema Web para gestionar la información académica en el Instituto Técnico San Pablo?

F. EVALUANDO LA OPORTUNIDAD

17. ¿Quiénes necesitan este sistema en su organización?

18. ¿Cuántos de estos tipos de usuario utilizaría el Sistema?

G. EVALUANDO LAS NECESIDADES DE ESTE SISTEMA EN LA INSTITUCION

19. ¿Cuáles son las expectativas sobre la confiabilidad del sistema?

20. ¿Cuáles son las expectativas sobre la capacidad (Rendimiento) del sistema?

21. ¿Dará usted soporte al sistema. ¿Lo hará alguien más?

22. ¿Cuáles son los requerimientos de seguridad?

MANUAL DE USUARIO

SISTEMA DE INFORMACION WEB PARA EL
CONTROL Y SEGUIMIENTO ACADÉMICO
PARA EL INSTITUTO TÉCNICO SAN PABLO



(SICSA)

MIRNA TOLA MENDOZA

1. INTRODUCCIÓN

En la actualidad las instituciones tanto públicas como privadas, cuentan con un sistema de seguimiento académico, pero no con acceso de tecnologías de innovación que brinde una información oportuna, confiable, precisa y de forma automática, estos procesos se realizan manualmente en Excel, lo que presenta un problema para la secretaria académica, docentes y estudiantes De acuerdo a lo mencionado, se hará el desarrollo del “Sistema de Información Web para el Control y Seguimiento Académico” el objetivo de dicho sistema es brindar información relevante de forma rápida búsqueda de información de la gestión académica.

2. OBJETIVO

Acceder y guiar al usuario mediante este manual para el uso del sistema en forma correcta.

3. REQUERIMIENTO DEL SISTEMA

a) Requerimientos de hardware

Un ordenador (computadora Pc o Laptop)

Conexión a Internet

b) Requerimiento del software

Sistema operativo Windows o Linux

Navegadores (Mozilla Firefox y Google Chrome)

4. TIPOS DE USUARIO

El sistema contiene cuatro tipos de usuario

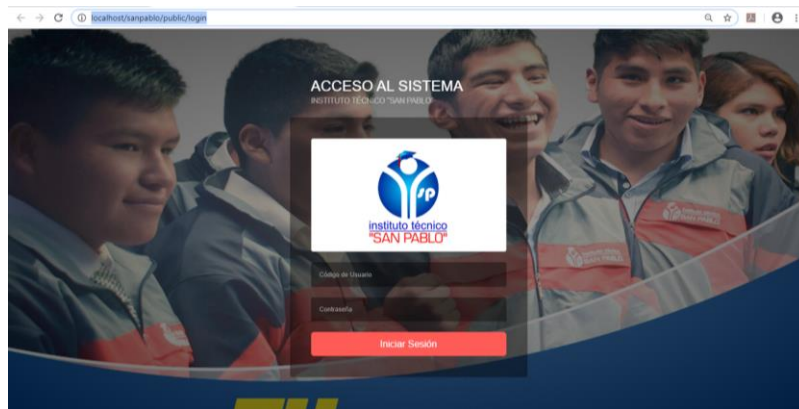
- Usuario Administrador
- Usuario Académico
- Usuario Docente
- Usuario Estudiante

5. IMPLEMENTACIÓN DEL SISTEMA

Ingresar a su navegador dentro del navegador teclee la siguiente dirección electrónica:

- <http://localhost/sanpablo/public/login>

El sistema se redirecciona al inicio de sesión:



5.1 INTERFAZ INCIO DE SESION

Se deberá autenticar en el sistema para acceder a cada uno de los módulos correspondientes.

Se ingresara con una cuenta del tipo usuario proporcionada por el administrador
El sistema verifica al usuario y clave ingresado de ser correcta la verificación, el usuario ingresa al sistema y podrá acceder a las opciones según su rol asignado.
En caso de ingresar los datos erróneos el sistema, muestra un mensaje como se ve en la siguiente imagen.



5.2 FUNCIONALIDAD GENERAL

El sistema se divide en tres zonas las cuales son: zona usuario, zona menú y zona central.



5.3 MÓDULOS QUE INTEGRAN EL SISTEMA

5.3.1 PARÁMETROS

En esta primera parte donde se insertan todos los parámetros que se requerirán en cuanto es docente, estudiante, carrera, inscripción y otros.

5.3.1.1 Administrativo

En este formulario se puede observar la lista del personal administrativo, tiene las opciones de añadir nuevo administrativo (Dirección académica) como también tiene la acción de ver, editar y eliminar.

Pantalla lista de Administrativos

Id	Estado civil	Unidad area	Tipo documento	Numero documento	Expedido	Paterno	Materno	Nombre	Nacionalidad	País nacimiento
7	Casado	Administrativos	CI	1020301	LP	FERRAÑEZ	PACHECO	GONZALO	Boliviana	BOLIVIA
8	Casado	Administrativos	CI	1020302	LP	CORA	VILLA	RODOLFO	Boliviana	BOLIVIA

En la pestaña de color verde podemos registrar un nuevo administrador con los siguientes campos.

Pantalla registro de administrativos

The screenshot shows a web application interface for administrative registration. On the left is a sidebar with a user profile for Rodolfo Copavilla Perfil Académico and a menu with options like Inicio, Parámetros, Carreras, and ADMINISTRACIÓN DE EMPRESAS. The main area is titled 'ADMINISTRATIVO' and contains a form with the following fields:

- Estado_civil: dropdown menu (value: estado_civil)
- Unidad_area: dropdown menu (value: unidad_area)
- Tipo documento: dropdown menu (value: CI)
- Numero documento: text input
- Expedido: dropdown menu (value: LP)
- Paterno: text input
- Materno: text input
- Nombre: text input
- Nacionalidad: dropdown menu (value: nacionalidad)
- País nacimiento: text input
- Departamento nacimiento: text input
- Provincia nacimiento: text input
- Localidad nacimiento: text input

At the top of the form are buttons: Guardar, Guardar y Listar, Cancelar, and Volver a la Lista.

Una vez llenado todos los campos le damos clic en guardar y listar.

5.3.1.2 Configuración

En este formulario se puede observar la configuración que se realizó acerca de la institución.

Pantalla de configuración del instituto

The screenshot shows the 'CONFIGURACIÓN' screen in the web application. It features a sidebar on the left and a main area with a table. The table has columns for various configuration details. Below the table, it indicates 'Hay 1 registros. Mostrando de 1 a 1'.

ID	Logo reporte imagen	Numero patronal	Numero identificador sistema trabajo	Numero empleador	Nombre	R m	Telefono 1	Email 1	Nit	Numero autorizacion	Fecha maxima emision
1		0	100454800-1	12334-1	INSTITUTO TECNICO SAN PABLO		121980 2334333	sarpablo@gmail.com	25664696	24568506	03-11-2019

Apreciamos la configuración realizada en el sistema por el administrador académico.

5.3.1.3 Docente

En este formulario se lista a todos los docentes que trabajan en el instituto

Pantalla lista de docentes

Id	Apellido	Apellido	Nombre	Grado académico	País nacimiento	Departamento nacimiento	Provincia nacimiento	Localidad nacimiento	Tipo documento	Número documento
1	ROMERO	QUILLA	LORENA	LIC.	BOLIVIA	LA PAZ	MURILLO	LA PAZ	CI	2099602
2	Ticona	Sanga	Juan Pablo	Técnico Medio	Bolivia	La Paz	Murillo		CI	1562409

Como se puede ver en la imagen tiene un botón de color verde para crear un nuevo docente con los siguientes campos:

* Expedito: LP

* Fecha nacimiento: 05-jun / 1990

* Sexo: MASCULINO

Direccion provincial: El Alto

Direccion municipal: El Alto

Direccion ciudad localidad: El Alto

Direccion zona barrio urbanizacion: Satelite

Direccion calle: Laureles

Direccion numero: 13

Telefono:

Celular: 72056004

* Estado docente: activo

* Nacionalidad: Boliviana

* Estado_civil: Soltero

* Foto imagen: Seleccionar archivo

Una vez llenado todos los campos le damos click en la parte de arriba guardar y listar para tenerlos registrados correctamente en el sistema.

5.3.1.4 Estudiante

En este formulario nos muestra la lista de estudiantes registrados en el sistema.

Pantalla lista de estudiantes

ID	Paterno	Materno	Nombre	País nacimiento	Departamento nacimiento	Provincia nacimiento	Localidad nacimiento	Tipo documento	Numero documento	Expedido
1	LOAYZA	MORALES	RODRIGO	BOLIVIA	LA PAZ	MURILLO	LA PAZ	CI	10100202	LP
2	FLORES	MARTINES	CARLOS	BOLIVIA	LA PAZ	MURILLO	LA PAZ	CI	3466464	LP
3	TICONA	LOAYZA	MARIBEL	Bolivia	LA PAZ	Murillo	EL ALTO	CI	12458978	LP
4	SENTENO	MAMANI	GABRIEL	BOLIVIA	LA PAZ	MURILLO	EL ALTO	CI	9104502	LP
5	VERRIOS	HUANCA	MIRIAM	BOLIVIANA	LA PAZ	Murillo	Murillo	CI	4588951	LP
6	CHINO	MAMANI	WILMER	BOLIVIA	LA PAZ	MURILLO	EL ALTO	CI	568947	LP
7	HUANCA	POSTO	HERNAN	Bolivia	LA PAZ	Murillo	Murillo	CI	1254688	LP
8	CONDORI	MAMANI	LISBETH	Bolivia	LA PAZ	Murillo	Murillo	CI	1256975	LP
9	MORALES	ROQUE	ELENA	Bolivia	La Paz	Murillo	Murillo	CI	1568975	LP
10	MARCA	ICHUTA	YAMIL	Bolivia	La Paz	Murillo	Murillo	CI	9004563	LP

Tenemos 5 botones actuales en el sistema el primero es copiar donde nos permite copiar en forma de texto, el segundo es ver en formato Excel, el tercero es en formato PDF y el otro es para imprimir si es que así lo requiere el secretario académico.

El botón de color verde nos permite registrar nuevo estudiante con los siguientes campos:

Pantalla registro de estudiantes

ESTUDIANTE

Guardar Guardar y Listar Cancelar Volver a la Lista

* Paterno: Mamani

Materno: Volverto

* Nombre: Magali

País nacimiento: Bolivia

Departamento nacimiento: La Paz

Provincia nacimiento: Murillo

Localidad nacimiento: Murillo

* Tipo documento: CI

* Numero documento: 12546878

* Expedido: LP

* Fecha nacimiento: 21/04/2020

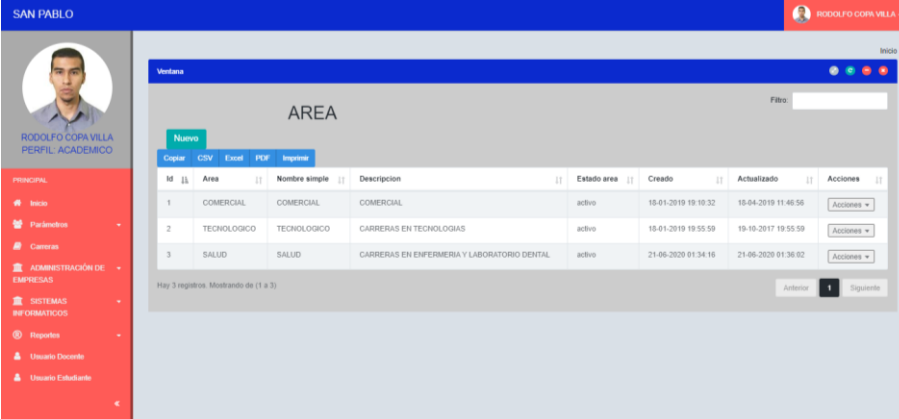
* Sexo: MASCULINO

Una vez llenados correctamente los datos del estudiante proporcionados al instituto, e damos click en guardar y listar en la parte superior.

5.3.1.5 Áreas

En este formulario tenemos las áreas para las carreras que existen en el instituto.

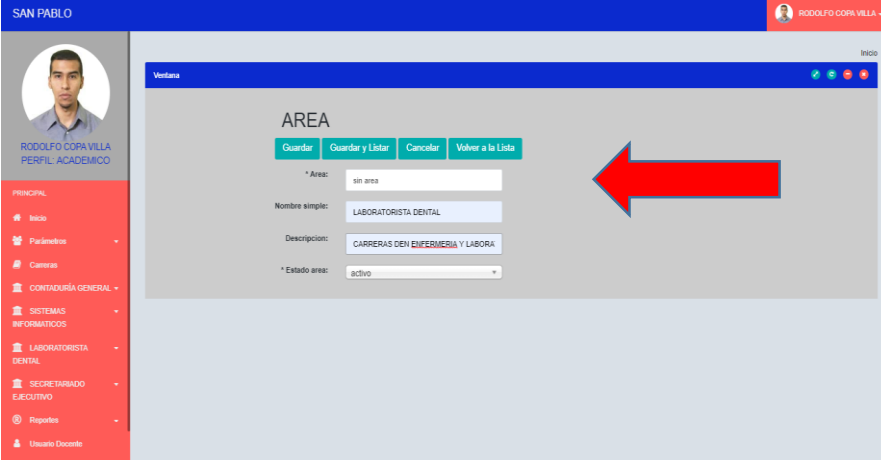
Pantalla de áreas



ID	Area	Nombre simple	Descripción	Estado area	Creado	Actualizado	Acciones
1	COMERCIAL	COMERCIAL	COMERCIAL	activo	18-01-2019 19:10:32	18-04-2019 11:46:56	Acciones
2	TECNOLÓGICO	TECNOLÓGICO	CARRERAS EN TECNOLOGIAS	activo	18-01-2019 19:55:59	19-10-2017 19:55:59	Acciones
3	SALUD	SALUD	CARRERAS EN ENFERMERIA Y LABORATORIO DENTAL	activo	21-06-2020 01:34:16	21-06-2020 01:36:02	Acciones

Se aprecia las áreas que existen dentro del instituto Técnico San Pablo según las carreras que tiene la institución.

Tenemos las siguientes acciones donde podemos ver, editar y eliminar. Con el botón en la parte superior podemos registrar una nueva área.



Formulario de creación de Área:

- * Área:
- Nombre simple:
- Descripción:
- * Estado area:

Botones: Guardar, Guardar y Listar, Cancelar, Volver a la Lista

Una vez llenado los campos le damos click en guardar y listar para que se muestre en el sistema.

5.3.1.6 Carreras

En este formulario observamos las carreras que tenemos registradas en el sistema.

Pantalla lista de carreras

ID	Area	Sigla carrera	Carrera	Nombre simple	Fecha creacion	Descripcion	Tipo gestion	Estado carrera	Creado	Actualizado
1	COMERCIAL	ACE	CONTADURIA GENERAL	CONTADURIA GENERAL	07-01-2019	SINGUNA	semestral	activo	15-10-2017 19:11:19	22-06-2020 21:08:58
2	TECNOLOGICO	SIH-001	SISTEMAS INFORMATICOS	SISTEMAS INFORMATICOS	09-06-2020		modular	activo	09-06-2020 23:33:34	09-06-2020 23:31:34
3	SALUD	SIH-001	LABORATORISTA DENTAL	LABORATORISTA DENTAL	21-03-2006	CARRERAS DEN ENFERMERIA Y LABORATORIO DENTAL	modular	activo	21-06-2020 01:42:50	21-06-2020 01:42:50
4	COMERCIAL	SCEJ-006	SECRETARIADO EJECUTIVO	SECRETARIADO	22-06-2020		semestral	activo	22-06-2020 00:49:29	22-06-2020 00:49:29

Muestra las carreras que cuenta el instituto como también podemos registrar una nueva carrera de la siguiente manera:

Registro de una nueva carrera

CARRERA

[Guardar](#) [Guardar y Listar](#) [Cancelar](#) [Volver a la Lista](#)

* Area: SALUD

Sigla carrera: SIH-001

* Carrera: ENFERMERIA

Nombre simple: ENFERMERIA

Fecha creacion: 21 Jun /2020

Descripcion:

* Tipo gestion: semestral

* Estado carrera: activo

- Elegimos el área al cual pertenece la carrera
- Ponemos la sigla de la carrera a ser creada
- Posteriormente ponemos el nombre de la carrera
- Seguidamente elegimos tipo gestión que puede ser semestral, modular o anual
- Ponemos activo para habilitar la carrera
- Finalizamos haciendo click en el botón Guardar o Guardar y Listar

5.3.1.7 Pensum

En este formulario observamos el pensum para cada carrera con las cuales trabajarán dentro de la institución.

Pantalla lista de pensum

ID	Carrera	Pensum	Plan pensum	Nombre título profesional	Fecha inicio	Fecha fin	Observaciones	Gestion	Modalidad pensum	Estado pensum	Creado
1	CONTADURÍA GENERAL	PLAN1	PLAN 1	TECNICO EN CONTABILIDAD	07-01-2019	07-01-2019	NINGUNA	2019	semestral	activo	16-10-2017 19:12:43
3	SISTEMAS INFORMATICOS	PLAN 1	PLAN 1	TECNICO SUPERIOR EN SISTEMAS INFORMATICOS	03-02-2020	27-01-2020	NINGUNA	2020	modular	activo	22-06-2020 21:22:55
4	SECRETARIADO EJECUTIVO	PLAN 1	PLAN 1	TECNICO SUPERIOR EN SECRETARIADO EJECUTIVO	03-02-2020	27-01-2020	NINGUNA	2020	anual	activo	26-06-2020 09:47:26

Pantalla registro de pensum

Guardar **Guardar y Listar** **Cancelar** **Volver a la Lista**

Carrera:

* Pensum:

Plan pensum:

Nombre título profesional:

Fecha inicio:

Fecha fin:

Observaciones:

* Gestion:

Modalidad pensum:

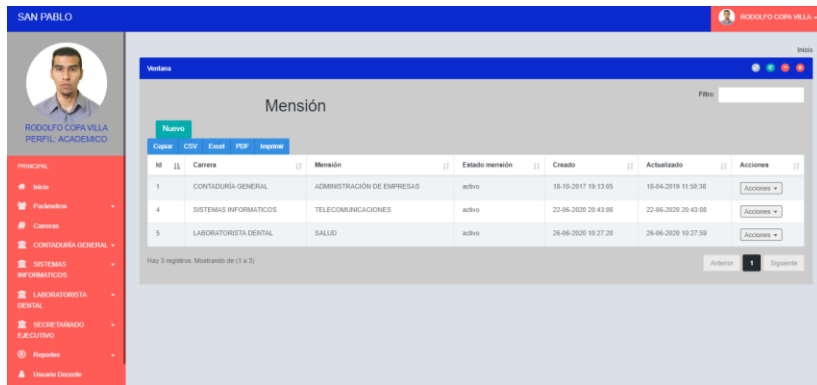
* Estado pensum:

- Para crear un nuevo pensum es necesario seleccionar la carrera
- Posteriormente asignar el pensum
- Crear la fecha de inicio y la fecha fin
- Luego debemos seleccionar la modalidad de pensum ya sea anual o semestral
- Habilitamos el pensum en estado de pensum
- Para finalizar hacer click sobre el botón Guardar o Guarda y listar

5.3.1.8 Mención

En este formulario nos muestra las menciones que tenemos de acuerdo a las carreras que cuenta la institución.

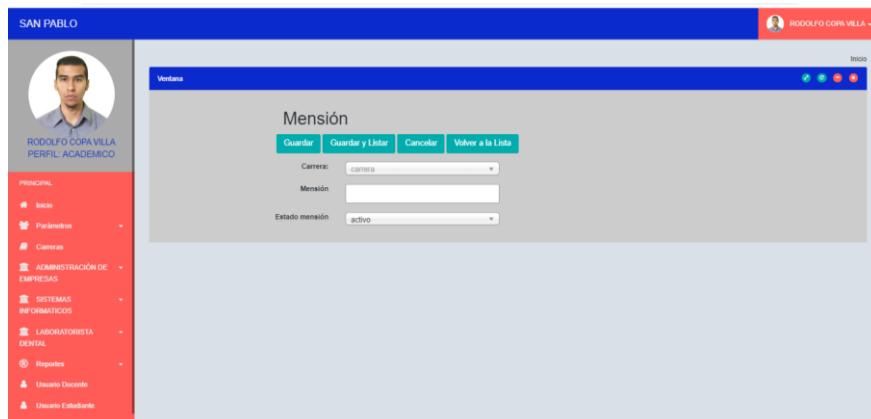
Pantalla lista de mención



M	Carrera	Mención	Estado mención	Creado	Actualizado	Acciones
1	CONTADURÍA GENERAL	ADMINISTRACIÓN DE EMPRESAS	activo	18-10-2017 18:13:05	18-04-2019 11:58:30	Acciones
4	SISTEMAS INFORMÁTICOS	TELECOMUNICACIONES	activo	22-06-2020 20:43:00	22-06-2020 20:43:00	Acciones
5	LABORATORISTA DENTAL	SALUD	activo	26-06-2020 18:27:20	26-06-2020 18:27:59	Acciones

Así también podemos registrar una nueva mención con los siguientes campos a llenar:

Pantalla registro de mención



Formulario de registro de mención:

- Carrera:
- Mención:
- Estado mención:

Botones:

- Para el registro de nueva mención debemos elegir la carrera
- Posteriormente ponemos la mención a ser creada
- También ponemos el estado de la mención como activo o eliminado

5.3.1.9 Asignatura

En este formulario observamos las asignaturas que tenemos dentro del sistema acuerdo a las carreras que cuenta el instituto.

Pantalla lista de asignaturas de las carreras existentes

ID	Carrera	Sigla asignatura	Asignatura	Estado asignatura	Creado	Actualizado	Acciones
1	CONTADURÍA GENERAL	FUA-191	FUNDAMENTOS DE CONTABILIDAD	activo	15-01-2019 19:13:30	22-06-2020 21:17:40	Acciones
2	CONTADURÍA GENERAL	CGG-102	CONTABILIDAD GENERAL PARA ADMINISTRADORES	activo	15-01-2019 19:13:29	15-04-2019 11:52:43	Acciones
3	CONTADURÍA GENERAL	MAF-103	MATEMÁTICA FINANCIERA	activo	15-01-2019 19:14:00	15-04-2019 11:53:14	Acciones
4	CONTADURÍA GENERAL	ESN-104	ESTADÍSTICA PARA LOS NEGOCIOS	activo	15-01-2019 19:14:12	15-04-2019 11:53:45	Acciones
15	CONTADURÍA GENERAL	PLT-105	PRÁCTICA LABORAL	activo	15-01-2019 09:40:18	21-06-2020 01:58:23	Acciones
16	CONTADURÍA GENERAL	LAA-106	INFORMÁTICA APLICADA A LA CONTABILIDAD	activo	15-01-2019 09:41:31	22-06-2020 21:18:19	Acciones
18	SISTEMAS INFORMÁTICOS	ETCA-02	ESTADÍSTICA	activo	26-06-2020 01:29:54	26-06-2020 01:29:54	Acciones
19	SISTEMAS INFORMÁTICOS	RD-003	REDES DE COMPUTADORAS I	activo	22-06-2020 21:14:17	22-06-2020 21:14:17	Acciones

Con el botón de color verde podemos registrar una asignatura nueva en la parte de acciones tenemos las opciones de ver, editar y eliminar.

Registro de asignatura

Ver una

ASIGNATURA

[Guardar](#) [Guardar y Listar](#) [Cancelar](#) [Volver a la Lista](#)

Carrera:

* Sigla asignatura:

* Asignatura:

* Estado asignatura:

- Primeramente elegimos la carrera a la que pertenecerá la nueva asignatura
- Registramos la sigla de la asignatura a ser creada
- Posteriormente ponemos el nombre de la asignatura
- Para finalizar le damos click en el botón de Guardar o Guardar y Listar

5.3.1.10 Tipo de Evaluación

En este formulario tenemos los tipos de evaluación con los que todos los docentes podrán evaluar a los estudiantes llegando a un 100 %.

Pantalla tipo de evaluación

The screenshot shows a web application interface for evaluation management. At the top, there is a header with the name 'SAN PABLO' and a user profile for 'RODOLFO COPÁ VILLA'. Below the header is a sidebar menu with options like 'Inicio', 'Parámetros', 'Carrera', 'ADMINISTRACIÓN DE EMPRESAS', 'SISTEMAS INFORMÁTICOS', 'LABORATORISTA DENTAL', 'Reportes', 'Unidad Docente', and 'Unidad Estudiantil'. The main content area is titled 'Evaluación' and contains a table with the following data:

Id	Evaluación	Casilla 1	Casilla 2	Casilla 3	Casilla 4	Casilla 5	Casilla 6	Casilla 7	Casilla 8	Casilla 9	Casilla 10	Casilla 11
1	NORMAL	Trabajos 20%	Practicas 30%	Participación 30%	Examen 40%							
2	FORMA 1	EXAMEN FINAL 50%	PARTICIPACIÓN 30%	TRABAJO 20%	ASISTENCIA 20%							

Tenemos los botones de copiar, Excel, PDF e imprimir donde si lo ponemos en formato Excel nos muestra de la siguiente manera:

The screenshot shows an Excel spreadsheet with the following data:

Id	Evaluación	Casilla 1	Casilla 2	Casilla 3	Casilla 4	Casilla 5	Casilla 6
1	NORMAL	Trabajos 20%	Practicas 30%	Participación 30%	Examen 40%		
2	FORMA 1	EXAMEN FINAL 50%	PARTICIPACIÓN 30%	TRABAJO 20%	ASISTENCIA 20%		

5.3.1.11 Gestión

En este formulario observamos la gestión actual como también podemos buscar las gestiones pasadas.

Pantalla lista de gestion

Id	Carrera	Periodo gestion	Gestion	Tipo gestion	Fecha inicio gestion	Fecha fin gestion	Descripción gestion	Estado gestion	Creado	Actualizado	Acciones
5	CONTADURÍA GENERAL	ANUAL	2019	REGULAR	15-04-2019	15-12-2019	2019	activo	15-04-2019 12:03:59	15-04-2019 12:03:59	Acciones
6	SISTEMAS INFORMATICOS	ANUAL	2020	REGULAR	22-06-2020	22-07-2021	2020	activo	22-06-2020 20:39:55	22-06-2020 20:40:24	Acciones

Así también tenemos en la parte de acción las opciones de ver, editar y eliminar.

Registro de gestión

Form fields and values:

- Carrera: ADMINISTRACIÓN DE EMPRESAS
- Periodo gestion: ANUAL
- Gestion: 2019
- Tipo gestion: REGULAR
- Fecha inicio gestion: 15/abr/2019
- Fecha fin gestion: 15/dic/2019
- Descripción gestion: 2019
- Estado gestion: activo

- Para la gestión elegimos una carrera
- Seguidamente elegimos el periodo de gestión
- Elegimos el tipo de gestión puede ser regular, invierno o verano
- Creamos la fecha de inicio y la fecha en la que finalizara la gestión

5.3.2 Carreras

En este módulo podemos observar las carreras existentes.



5.3.2.1 Plan de Estudio

Si le damos click en plan de estudio nos muestra el plan de las materias

Id	Pensum	Mención	Codigo Asig.	Asignatura	Nivel	Modalidad	Horas	Label prerequisito	Acciones
1	PLAN1	CONTABLE	FUA-101	FUNDAMENTOS DE CONTABILIDAD	1	semestral	6	NINGUNA	Acciones
2	PLAN1	CONTABLE	CGS-102	CONTABILIDAD GENERAL PARA ADMINISTRADORES	1	semestral	6	NINGUNA	Acciones
3	PLAN1	CONTABLE	MAF-103	MATEMÁTICA FINANCIERA	1	semestral	4	NINGUNA	Acciones
4	PLAN1	CONTABLE	ESN-104	ESTADÍSTICA PARA LOS NEGOCIOS	1	semestral	4	NINGUNA	Acciones
10	PLAN1	CONTABLE	PLT-105	PRÁCTICA LABORAL	1	semestral	4	NINGUNA	Acciones
11	PLAN1	CONTABLE	LAA-106	INFORMÁTICA APLICADA A LA CONTABILIDAD	1	semestral	2	NINGUNA	Acciones
12	PLAN1	CONTABLE	TEE-107	GESTOR EMPRESARIAL	1	semestral	4	NINGUNA	Acciones

Observamos el id, Pensum, Mención, Cod. Asignatura, Asignatura, Nivel, Modalidad, Horas, Prerequisitos y Acciones.

En el botón de acciones tenemos Ver, Editar y Eliminar.

En el botón de Nuevo podemos crear nuevo Plan de Estudio con los siguientes campos:

➤ Primero elegimos el pensum

- Elegimos la mención y la asignatura
- Colocamos el nivel de asignatura
- Escogemos la modalidad de asignatura y las horas académicas
- Para finalizar damos click en el botón Guardar o Guardar y Listar

5.3.2.2 Cursos

En este formulario observamos los cursos de la carrera.

Nro	Período	Gestión	Plan	Mención	Nivel	Paralelo	Turno	Sigla	Asignatura	Acciones
1	ANUAL	2019	PLAN1	CONTABLE	1	A	mañana	FUN-101	FUNDAMENTOS DE CONTABILIDAD	Notas Lista Boletín
2	ANUAL	2019	PLAN1	CONTABLE	1	A	mañana	COC-102	CONTABILIDAD GENERAL PARA ADMINISTRADORES	Notas Lista Boletín
3	ANUAL	2019	PLAN1	CONTABLE	1	A	mañana	MAF-103	MATEMÁTICA FINANCIERA	Notas Lista Boletín
4	ANUAL	2019	PLAN1	CONTABLE	1	A	mañana	ESB-104	ESTADÍSTICA PARA LOS NEGOCIOS	Notas Lista Boletín
5	ANUAL	2019	PLAN1	CONTABLE	1	A	mañana	PLT-105	PRÁCTICA LABORAL	Notas Lista Boletín
6	ANUAL	2019	PLAN1	CONTABLE	1	A	mañana	LAA-106	INFORMÁTICA APLICADA A LA CONTABILIDAD	Notas Lista Boletín

En la parte del lado derecho tenemos tres botones el primero es de notas, lista y boletín. También podemos observar el botón de editar y eliminar.

En el botón de Editar vemos lo siguiente:

Curso

* Gestión: ANUAL - 2019 REGULAR

* Asignatura mención permisi: FUA-101 FUNDAMENTOS DE CONTABILIDAD CONTABLE PLAN1

Docente: 2099882-UP L.C. ROMERO QUILLA LORENA

Forma de Evaluación: FORMA 1

* Paralelo curso: A

Turno curso: mañana

Cupo curso: 50

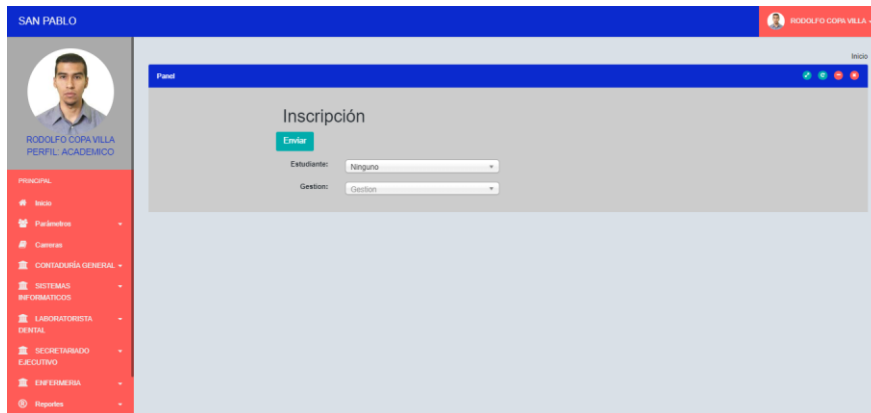
* Estado curso: activo

- En primer lugar elegimos el tipo de gestión
- Posteriormente elegimos la Asignatura y el docente que dictara la materia
- Elegimos la forma de evaluación la normal o la forma 1
- Seguidamente elegimos el paralelo y el turno de curso
- Asignamos en número de cupos para esa materia
- Finalmente damos click en Guardar o Guarda y Listar

Para nuevo curso vemos los siguientes pasos a seguir:

- Para crear curso es necesario seleccionar la gestión, pensum, mención y el turno
- Posteriormente hacer click en nivel para ver las asignaturas
- Seleccionar los paralelos y las asignaturas que se quiere crear
- Para finalizar hacer click en el botón Guardar o Guardar y Listar

5.3.2.3 Inscripción



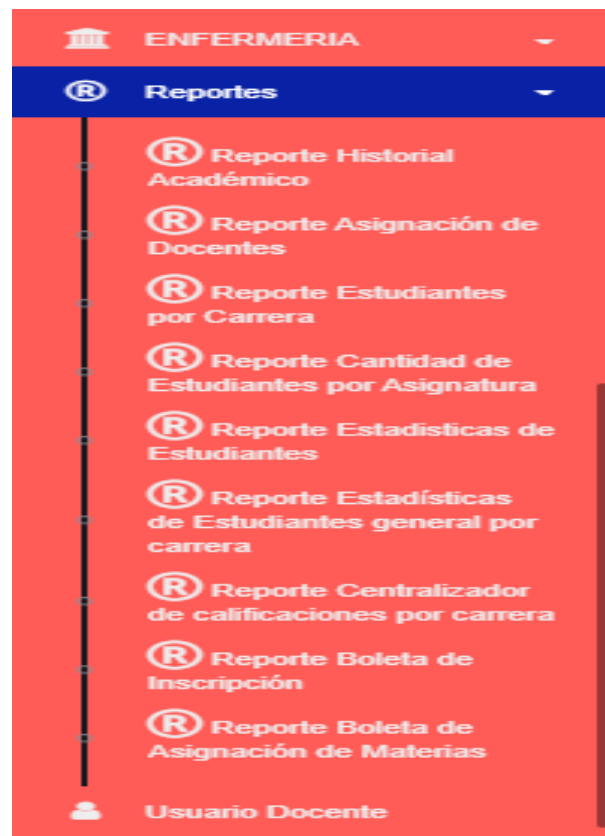
- Lo primero que se debe hacer es elegir el estudiante a inscribirse una vez que se haya registrado en el sistema
- Posteriormente elegimos la gestión
- Para finalizar hacer click en enviar

Una de ves de hacer click en enviar nos muestra el siguiente formulario

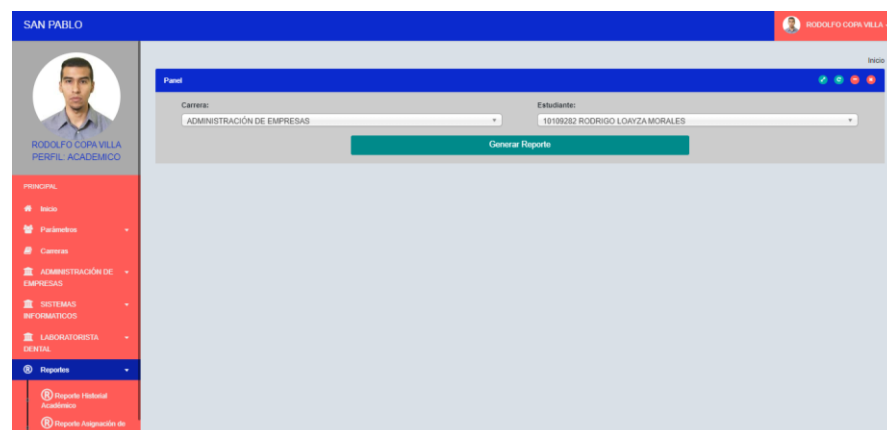


5.3.3 Reportes

Los reportes que genera el sistema son los siguientes:



5.3.3.1 Generación de historial académico



De esta manera realizamos la generación de Reportes.

Período Académico ANUAL 2019		Materia(s)	Nota	Observación	Tipo
Nro.	Nivel	Nota	Observación	Tipo	
1	1	FUA-101	FUNDAMENTOS DE ADMINISTRACIÓN	80	APROBADO NORMAL

Resumen		
Cantidad de Materias Aprobadas	Cantidad de Materias Reprobadas	Cantidad de Materias Abandonadas
1	0	0

Este documento no tiene validez hasta su verificación por Secretaría Académica

5.3.4 Pantalla Usuario Docente

Id	Foto	Usuario	e-mail	Rol	Descripción de Rol	Persona	DOCUMENTO	Estado	Acciones
29		113L	romeroqlorena@gmail.com	docente	docente academico	ROMERO QUILLA LORENA	CI 2099802 LP	activo	Acciones
32		115J	mt_id@gmail.com	docente	docente academico	YICORA SARGA JUAN PABLO	CI 1903489 LP	activo	Acciones
41		116J	conde_ch@hotmail.com	docente	docente academico	CONDE CHACON JIMENA	CI 1457096 LP	activo	Acciones
42		117A	angh_id@gmail.com	docente	docente academico	CARDIA FLORES ANGELA	CI 6734562 LP	activo	Acciones

Aquí vemos la lista de usuario docente que tenemos dentro del instituto donde nos muestra la acción de ver, editar y eliminar.

Como podemos observar tenemos en la parte superior el botón de Nuevo, para poder añadir un nuevo docente.

USUARIO

* docente:

* E-mail:

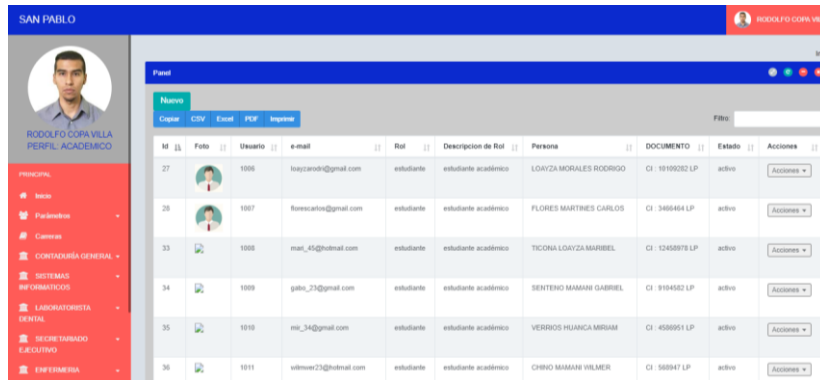
Clave: Para modificar la clave de usuario

* Estado:

Vemos el registro de docente que campos debemos llenar

5.3.5 Pantalla Usuario Estudiante

Apreciamos la lista de estudiantes registrados en el sistema como también vemos en la parte superior Nuevo, para poder registrar un nuevo estudiante.



Donde en acciones tenemos las opciones de ver, editar y eliminar.

5.4 Usuario Docente



En el usuario Docente observamos en la parte de menú inicio y las carreras de la institución, donde en cada una de ellas se encuentra curso.



5.3.6 Usuario Estudiante



Observamos el menú de la carrera que curso el estudiante



Aquí vemos el historial académico del estudiante



Observamos la calificación que obtuvo el estudiante durante la gestión académica.

MANUAL TÉCNICO

1. Requerimiento de Hardware para el desarrollo del Sistema

Tabla 1.1 Requerimiento de Hardware

ÍTEM	DESCRIPCIÓN	CARACTERÍSTICAS
1	Computadora	- Procesador Intel Core I7 2.40 GHz - Disco duro 500 GB - Memoria RAM 8GB
1	Monitor	LCD 19" (Pulgadas)
1	Mouse	Conector USB
1	Teclado	Conector USB

El hardware descrito en la tabla 1.1, será utilizado para el desarrollo del sistema, el cual será provisto por la institución para el desarrollo.

2. Requerimiento de Software para el desarrollo del Sistema

Tabla 1.2 Requerimiento de Software

ÍTEM	DESCRIPCIÓN	CARACTERÍSTICAS
1	Sistema Operativo	Windows 10 Home
1	Gestor de Base de Datos	MySQL
1	Servidor Web	Apache
1	Lenguaje de programación	PHP

La tabla 1.2, describe el software requerido para el desarrollo del sistema, el cual puede ser obtenido de internet, tomando en cuenta que el Instituto Técnico “San Pablo”, cuenta con los equipos mencionados para el desarrollo del Sistema de Información Web para el Control y Seguimiento Académico.