



UNIVERSIDAD NACIONAL DEL ALTIPLANO
FACULTAD DE INGENIERIA ESTADÍSTICA E
INFORMÁTICA
ESCUELA PROFESIONAL DE INGENIERIA ESTADÍSTICA E
INFORMÁTICA



SISTEMA INFORMÁTICO PARA LA GESTIÓN ACADÉMICA
APLICADO EN LA INSTITUCIÓN EDUCATIVA SECUNDARIA
INDUSTRIAL 32 DE PUNO

TESIS

PRESENTADA POR:

Bach. ALEXANDER HUARAYA CANLLAHUE

PARA OPTAR EL TÍTULO PROFESIONAL DE:
INGENIERO ESTADISTICO E INFORMATICO

PUNO – PERÚ

2024



NOMBRE DEL TRABAJO

SISTEMA INFORMÁTICO PARA LA GESTIÓN ACADÉMICA APLICADO EN LA INSTITUCIÓN EDUCATIVA SECUNDARIA INDUSTRIAL

AUTOR

ALEXANDER HUARAYA CANLLAHUE

RECuento de palabras

18714 Words

RECuento de caracteres

110275 Characters

RECuento de páginas

116 Pages

Tamaño del archivo

3.5MB

Fecha de entrega

Jan 26, 2024 8:33 AM GMT-5

Fecha del informe

Jan 26, 2024 8:35 AM GMT-5

● **14% de similitud general**

El total combinado de todas las coincidencias, incluidas las fuentes superpuestas, para cada base de datos

- 13% Base de datos de Internet
- Base de datos de Crossref
- 9% Base de datos de trabajos entregados
- 1% Base de datos de publicaciones
- Base de datos de contenido publicado de Crossref

● **Excluir del Reporte de Similitud**

- Material bibliográfico
- Material citado
- Material citado
- Coincidencia baja (menos de 10 palabras)

 Firmado digitalmente por JUAREZ VARGAS Juan Carlos FAU 20145498170 soft
Motivo: Soy el autor del documento
Fecha: 26.01.2024 14:34:37 -05:00

 **UNA**
PUNO

Firmado digitalmente por COYLA IDME Leonel FAU 20145498170 soft
Motivo: Soy el autor del documento
Fecha: 26.01.2024 11:33:50 -05:00



DEDICATORIA

A Dios, por haberme permitido llegar hasta esta instancia, siempre cuidar de mí y de todos los seres queridos que me rodean, ya que la realización de esta tesis no lo puedo catalogar como algo sencillo, pero puedo afirmar que en el transcurso del trayecto pude aprender de manera concisa todos los procesos que uno tiene que pasar para llegar al objetivo.

Con mucho cariño a mi madre Juana Agustina Canllahue, por ser el motor que empuja mi vida, que incondicionalmente está ahí para poder dar todo el apoyo que requiero en cada trayecto de vida, así también como velar por mi bienestar y por haberme enseñado lo valioso que puede llegar a ser el cariño maternal.

A mis hermanos y hermanas, Emiliana, Richard, Delfina, Jesús, Mary, Irwin y German, que siempre es un agrado tenerlos reunidos y por el apoyo y experiencia que aportan.

He comprendido que no todos tenemos la misma línea de tiempo, evidentemente uno es capaz de alterar esa línea, pero, no todos tenemos la capacidad de intentar levantarse después de haber caído. Inténtalo, créeme, aplica para todo.

Alexander H.C



AGRADECIMIENTOS

A Dios por encaminarme en el transcurso de toda mi carrera, por ser mi apoyo en momentos de fragilidad y tristeza, por haberme enseñado lo grandioso que es la vida universitaria.

A la Facultad de Ingeniería Estadística e Informática ya a la Universidad Nacional del Altiplano por darme la oportunidad de poder adquirir todo el conocimiento necesario para poder sobresalir profesionalmente.

A mi asesor de tesis D.Sc. LEONEL COYLA IDME, por guiarme en cada proceso de este reciente trabajo de investigación,

Alexander H.C



ÍNDICE GENERAL

	Pág.
DEDICATORIA	
AGRADECIMIENTOS	
ÍNDICE GENERAL	
ÍNDICE DE TABLAS	
ÍNDICE DE FIGURAS	
ÍNDICE DE ANEXOS	
ÍNDICE DE ACRÓNIMOS	
RESUMEN	14
ABSTRACT.....	15
CAPÍTULO I	
INTRODUCCIÓN	
1.1. PLANTEAMIENTO DEL PROBLEMA.....	17
1.2. FORMULACIÓN DEL PROBLEMA	17
1.3. HIPÓTESIS RE LA INVESTIGACIÓN	17
1.4. JUSTIFICACIÓN DE LA INVESTIGACIÓN	18
1.5. OBJETIVOS DE LA INVESTIGACIÓN.....	18
1.5.1. Objetivo general	18
1.5.2. Objetivos específicos	18
CAPÍTULO II	
REVISIÓN DE LITERATURA	
2.1. MARCO TEÓRICO	19
2.1.1. Antecedentes de la investigación	19
2.2. MARCO CONCEPTUAL	21



2.2.1. Desarrollo se sistemas	21
2.2.2. Sistema de información	22
2.2.3. Base de datos	22
2.2.4. Sistema de gestión de base de datos.....	23
2.2.5. Lenguaje de modelado Unificado (UML).....	23
2.2.6. Diagramas de clase.....	23
2.2.7. PHP	26
2.2.8. HTML	26
2.2.9. CSS.....	27
2.2.10. MYSQL.....	27
2.2.11. Servidor Apache HTTP	28
2.2.12. Metodologías de desarrollo de software	28
2.2.13. Programación extrema (XP).....	30
2.2.14. Framework	30
2.2.15. Modelo vista controlador (MVC).....	31
2.2.16. Seguridad informática	33
2.2.17. Organización internacional de estandarización 9126 (ISO 9126).....	34
2.2.18. Gestión académica	40
2.2.19. Ingeniería de software	41
2.2.20. Principios de usabilidad	43

CAPÍTULO III

MATERIALES Y MÉTODOS

3.1. LUGAR DE ESTUDIO.....	46
3.2. TIPO DE INVESTIGACIÓN	46
3.3. DISEÑO DE LA INVESTIGACIÓN	46



3.3.1. Cuasi experimental.....	46
3.4. METODOLOGÍA DE DESARROLLO.....	48
3.5. POBLACIÓN Y MUESTRA.....	48
3.5.1. Población.....	48
3.5.2. Muestra.....	49
3.6. TÉCNICAS DE RECOLECCIÓN DE DATOS.....	50
3.7. METODOLOGÍA DE DESARROLLO DE SOFTWARE.....	51
3.8. MÉTODO DE VALIDACIÓN DE SOFTWARE	54
3.9. PRUEBA DE HIPÓTESIS	55
3.10. REQUERIMIENTOS DEL SISTEMA Y MATERIALES.....	55
3.10.1. Requerimientos funcionales	55
3.10.2. Requerimientos no funcionales	56
3.10.3. Requerimientos técnicos	57
CAPÍTULO IV	
RESULTADOS Y DISCUSIÓN	
4.1. DESARROLLO DEL SISTEMA	58
4.1.1. Fases del desarrollador.....	58
4.2. ARQUITECTURA MVC EN SIGAINDUSTRIAL32.....	59
4.3. MÓDULOS DEL SISTEMA.....	61
4.3.1. Módulos.....	61
4.3.2. Historias de Usuario.....	65
4.4. DIAGRAMA Y MODELO DE BASE DE DATOS	73
4.5. DISEÑO E IMPLEMENTACIÓN DEL SISTEMA	76
4.5.1. Implementación del sistema informático SIGAINDUSTRIAL32	76
4.6. VALIDACIÓN DEL SISTEMA	82



4.6.1. Evaluación de calidad de software según la ficha ISO – 9126	82
4.7. CONTRASTACION DE HIPOTESIS	88
4.7.1. Planteamiento de hipótesis	89
4.7.2. Fijación de nivel de significancia (α)	90
4.7.3. Regla de decisión utilizado	90
4.7.4. Prueba estadística	90
4.7.5. Decisión.....	91
4.8. DISCUSIÓN	91
V. CONCLUSIONES.....	93
VI. RECOMENDACIONES.....	94
VII. REFERENCIAS.....	95
ANEXOS.....	100

ÁREA: Informática

TEMA: Base de datos y Sistemas de Información

FECHA DE SUSTENTACIÓN: 30 de enero del 2024



ÍNDICE DE TABLAS

	Pág.
Tabla 1 Tipos De Atributos De Clase	24
Tabla 2 Diferencias Entre Metodologías Agiles Y Tradicionales	29
Tabla 3 Características Y Generalidades De Los Frameworks	31
Tabla 4 Ficha De Evaluación De Calidad De Software Iso 9126.....	54
Tabla 5 Historia De Usuario – Acceso Al Sistema.....	65
Tabla 6 Historia De Usuario – Creación De Tipos De Usuario.....	66
Tabla 7 Historia De Usuario – Gestión De Usuarios	67
Tabla 8 Historia De Usuario – Registro De Administradores Y Docentes.....	67
Tabla 9 Historia De Usuario – Registro De Estudiantes.....	68
Tabla 10 Historia De Usuario – Registro De Padres	69
Tabla 11 Historia De Usuario – Matriculas	70
Tabla 12 Historia De Usuario – Ingreso De Notas	71
Tabla 13 Historia De Usuario – Emisión De Ficha De Matricula	72
Tabla 14 Historia De Usuario – Emisión De Libreta De Notas.....	72
Tabla 15 Valoración Del Iso 9126	87
Tabla 16 Medición De Calidad De Software Iso 9126.....	87
Tabla 17 Resultados Del Instrumento De Evaluación (Anexo B) Del Antes Y Después, Aplicado A Número De Muestra.	88
Tabla 18 Medición De Aceptación O Rechazo Con La Encuesta Aplicada (Instrumento De Evaluacion)	89



ÍNDICE DE FIGURAS

	Pág.
Figura 1 Diagrama De Caso De Uso De Un Sistema De Información.....	25
Figura 2 Proceso De Transferencia Y Acceso A Un Servidor	28
Figura 3 Diagrama De La Estructura De Modelo Vista Controlador (Mvc).	32
Figura 4 Características Básicas De La Iso/Iec 9126	35
Figura 5 Solo Con La Aplicación De Post-Test.	47
Figura 6 Proceso De La Programación Extrema	52
Figura 7 Diagrama De Caso De Uso	73
Figura 8 Diagrama De Diseño De Base De Datos.....	74
Figura 9 Interfaz De Inicio De Sesión Al Sistema Web.....	77
Figura 10 Módulos Principales Del Sistema	78
Figura 11 Formulario De Registro De Usuarios Quienes Administraran El Sistema. .	79
Figura 12 Formulario De Docentes Que Harán Uso Del Sistema, Específicamente En El Ingreso De Notas.	79
Figura 13 Formulario De Registro De Nuevos Estudiantes.	80
Figura 14 Formulario De Registro De Padres De Familia, Donde Se Podrá Enlazar La Información Del Estudiante Y Del Padre.	81
Figura 15 Formulario De Matrículas Para Estudiantes Ya Registrados Previamente Por El Sistema.	81
Figura 16 Resultados Sobre La Encuesta De Calidad De Software Bajo Los Estándares Iso-9126 Funcionalidad.	82
Figura 17 Resultados Sobre La Encuesta De Calidad De Software Bajo Los Estándares Iso-9126 Confiabilidad.	83



Figura 18 Resultados Sobre La Encuesta De Calidad De Software Bajo Los Estándares	
Iso-9126 Usabilidad.	84
Figura 19 Resultados Sobre La Encuesta De Calidad De Software Bajo Los Estándares	
Iso-9126 Eficiencia.	84
Figura 20 Resultados Sobre La Encuesta De Calidad De Software Bajo Los Estándares	
Iso-9126 Mantenimiento.	85
Figura 21 Resultados Sobre La Encuesta De Calidad De Software Bajo Los Estándares	
Iso-9126 Portabilidad.	86



ÍNDICE DE ANEXOS

	Pág.
ANEXO 1 Valuación De Calidad De Software Estándar Iso – 9126.	101
ANEXO 2 Encuesta De Pre-Test Y Post-Test (Instrumento De Evaluación).....	102
ANEXO 3 Manual De Usuario De Sigaindustrial32.....	103
ANEXO 4 Manual De Usuario Docente	108
ANEXO 5 Código Fuente Del Sistema De Información	110



ÍNDICE DE ACRÓNIMOS

MySQL:	Lenguaje de consulta estructurado
PHP:	Pre-procesador de hipertexto
HTML:	Lenguaje de marcas de hipertexto
CSS:	Hojas de estilo en cascada
ISO:	Organización de estándares internacionales
UML:	Lenguaje de modelado unificado
XP:	Programación extrema
SGBD:	Sistemas de gestión de base de datos
MVC:	Modelo vista controlador
FRAMEWORK:	Estructura de software personalizado



RESUMEN

En la actualidad las instituciones educativas del entorno público brindan un servicio de educación primordial en la sociedad, pero las instituciones aún mantienen la tradicional recolección de información y datos de los estudiantes, padres de familia, cursos, docentes, administrativos y de áreas de especialidad al momento de la inscripción o matrícula de los estudiantes, por lo que genera gran cantidad de papeleo y documentación que está muy propensa a pérdida o mala manipulación de las mismas, es el caso de la institución Educativa Industrial 32 de la ciudad de Puno, que conlleva a una mala administración de la documentación, generando demora, costo y aglomeración de diferentes archivos. Entonces como objetivo principal es desarrollar un sistema informático para la gestión académica aplicado en la Institución Educativa Secundaria Industrial 32, que permita a los administrativos recabar, acceder y asegurar la información en una base de datos, en el tiempo más corto posible y así tener un mejor control administrativo de estudiantes y solucionar el principal problema de gestión de información de alumnado. El presente desarrollo de investigación se fundamenta en el procedimiento ágil de desarrollo de software XP programación extrema, lo cual nos permitirá una buena documentación, intercomunicación continua entre los codificadores y una rápida adaptación a cambios. El sistema será un software web, así tendrá ventaja de ser descentralizado, con mayor control de seguridad, hacer copias de seguridad en cualquier momento y tener la rapidez de cargar los datos. Entonces el proceso del sistema se maneja diferentes instrumentos de programación, modelación de datos y almacenamiento de los mismos, lo que permite la integración de la arquitectura MVC (Modelo Vista Controlador). El sistema mejora el control administrativo de la institución educativa lo cual se reduce el tiempo de atención y manejo de documentación entorno a la digitalización de una base de datos estable y de sencillo acceso.

Palabras Clave : Base de datos, Gestión académica, institución educativa, sistema informático.



ABSTRACT

Currently, educational institutions in the public environment provide a primary educational service in society, but the institutions still maintain the traditional collection of information and data from students, parents, courses, teachers, administrators and specialty areas. moment of registration or enrollment of students, which generates a large amount of paperwork and documentation that is very prone to loss or mishandling; this is the case of the Industrial Educational Institution 32 in the city of Puno, which entails to poor documentation management, generating delay, cost and agglomeration of different files. Therefore, the main objective is to develop a computer system for academic management applied in the Industrial Secondary Educational Institution 32, which allows administrators to collect, access and secure information in a database, in the shortest possible time and thus have a better administrative control of students and solve the main problem of student information management. The present research development is based on the agile software development procedure XP extreme programming, which will allow us good documentation, continuous intercommunication between coders and rapid adaptation to changes. The system will be web software, so it will have the advantage of being decentralized, with greater security control, making backup copies at any time and having the speed of loading data. Then the system process is handled by different programming instruments, data modeling and data storage, which allows the integration of the MVC (Model View Controller) architecture. The system improves the administrative control of the educational institution, which reduces the time of attention and handling of documentation around the digitization of a stable and easily accessible database.

Keywords: Database, academic management, educational institution, computer system.



CAPÍTULO I

INTRODUCCIÓN

Hoy en día, el mundo tecnológico donde vivimos, pasa por grandes cambios y constantemente evoluciona de manera jerárquica. Uno de los cambios a mencionar es la sistematización de los procesos administrativos que se da en diferentes lugares, ya sea públicos o privados. Esta sistematización ha logrado reducir considerablemente el tiempo de atención, los recursos, bienes de la institución y, sobre todo, el resguardo de información que es de suma y necesaria importancia para quienes lo requieran y constantemente lo manejan. Se considera que la información es un bien de mucho valor para las entidades, al mismo nivel que los bienes monetarios y seres humanos. Esta información se recolecta de tal forma que al pasar el tiempo llega a acumularse de manera masiva, teniendo así que optar por formas de como almacenar todo ello. En este punto de vista tecnológico cada vez más complejo y de muchas necesidades, la información toma un punto de suma importancia para la tomar una correcta elección y acontecimientos importantes para cambios y por ello, se opta por almacenar tanta cantidad de datos sea posible y de rápido acceso. Y es así como el proceso de almacenaje de datos da lugar a la que hoy conocemos como la base de datos y sistema de información. El sistema de gestión académica (SIGAINDUSTRIAL32), cumple con las necesidades requeridas para registrar, almacenar, reportar y consultar el uso de la información de los educados, padres, docentes y administrativos, que son administradas en la Institución Educativa Secundaria Industrial 32 de la ciudad de Puno (IES Industrial 32) en adelante y así también mejorar la gestión facilitando los registros en el área de administrativa.



1.1. PLANTEAMIENTO DEL PROBLEMA

La información recabada a lo largo de las gestiones, ha generado aglomeración de datos e información de los estudiantes, padres, docentes y administrativos, acumulando documentación en archivadores, folders, cuadernos de registro y hojas de actas. Esta información y datos son llevados al área de almacén, haciendo que las mismas sean propensas a pérdida de información. De tal forma el registro de nuevos estudiantes de primer año o por traslado, la información de los padres, así también el registro de docentes y administrativos genera desorden, causando molestias y extendiéndose el trabajo del personal administrativo.

Por las razones ya mencionadas se propone poner en marcha la implementación del sistema informático para la gestión académica aplicado en la Institución Educativa Secundaria Industrial 32 de la ciudad de Puno.

1.2. FORMULACIÓN DEL PROBLEMA

¿La implementación del sistema de información para la gestión académica mejorará el manejo y almacenamiento de información de la Institución Educativa Secundaria Industrial 32 de Puno?

1.3. HIPÓTESIS RE LA INVESTIGACIÓN

La implementación del sistema informático para la gestión académica mejora significativamente el manejo y almacenamiento de información de la Institución Educativa Secundaria Industrial 32 de Puno.



1.4. JUSTIFICACIÓN DE LA INVESTIGACIÓN

En los últimos años el entorno mundial se ha ido inclinando a nuevas tecnologías de información para la automatización de diferentes tareas administrativas lo que indica una mayor rapidez y simplificación de procesos. La carencia de un sistema informático de gestión académica se ve reflejado en la aglomeración de la documentación de todas las personas vinculadas a la Institución Educativa, lo cual genera y se expone a una pérdida de información de los mismos.

El presente trabajo facilita al personal encargado el registro y almacenamiento de información de los estudiantes, padres de familia, docentes y administrativos teniendo un mejor control, resguardo y orden de la información recabada.

1.5. OBJETIVOS DE LA INVESTIGACIÓN

1.5.1. Objetivo general

Implementar un sistema informático para la gestión académica aplicado en la Institución Educativa Secundaria Industrial 32 de Puno.

1.5.2. Objetivos específicos

- Analizar, diseñar e implementar un sistema informático para la gestión académica, aplicado en la Institución Educativa Secundaria Industrial 32 de Puno.
- Desarrollar módulos relacionados a la información de gestión académica
- Desarrollar una base de datos de acuerdo al diseño del sistema
- Evaluar la calidad del sistema de información bajo el estándar ISO 9126.



CAPÍTULO II

REVISIÓN DE LITERATURA

2.1. MARCO TEÓRICO

2.1.1. Antecedentes de la investigación

Pascagaza (2018), El resultado obtenido en su trabajo, es un software que permite gestionar los trabajos, el cual, a través de la materia informática social del programa de ingeniería de sistemas, da solución a los problemas sociales a través de unos proyectos, lo que generaba un volumen de información y que hay gran cantidad de proyectos donde se guarda en discos y libros. Así también como la mejora en gestión de la documentación de los proyectos de compromiso social, también la medición del impacto que genera tales proyectos y replicar el un modelo para otros proyectos.

Trigoso Inuma (2018) Concluye que el proyecto de investigación llevó a mejorar el proceso de registro académico, el cual permitirá en la actualidad un proceso ágil y eficaz a la información de los estudiantes lo que conlleva una mejor toma de decisión por parte del personal administrativo del centro educativo y mejorando así el registro académico para cada alumno.

Merino & Miranda (2018) Proponen un sistema de matrículas y consulta de notas, Donde concluye que el sistema de información mejora el proceso de matrículas y consulta de notas, brinda información como apoyo a la toma de decisiones y las horas teóricas, indicando que la previa investigación es un punto muy importante para poder realizar un sistema sólido y así garantizar la integridad de los datos, dando así una confianza en obtener consultas e informes.



Torres (2019) En su trabajo de investigación concluye indicando que el sistema de información de control de farmacia, mejora significativamente el manejo de información, lo cual permitió al personal encargado tener un control total del manejo de información, así también como la mejora del manejo de medicamentos e insumos y poder tomar decisiones como desabastecimiento de insumos.

Chachaque (2015) Culmina su trabajo con la implementación de su sistema, con el cual acelera el proceso de emisión de notas de manera eficaz y precisa, así también como acortar el tiempo de atención a los alumnos, cumpliendo con todos los requerimientos del secretariado académico, haciendo que el sistema cumpla satisfactoriamente las necesidades del personal administrativo.

Herrera (2016) Concluye en su investigación que el sistema de información mejora la gestión de información de manera automatizada y eficiente, al implementarlo se confirma que el tiempo de atención hacia los estudiantes en diferentes tareas administrativas se reduce considerablemente, obteniendo la digitalización de la información además de estar segura en una base de datos confiable.

(Torres, 2018) Concluye su investigación donde indica que se desarrolló e implementó la aplicación web con una favorable aceptación en gran parte de los administradores que operan el sistema web y por los usuarios que obtuvieron sus constancias electrónicas, además respaldado por el estándar ISO-9126, logrando que el sistema realiza y cumpla con los requerimientos funcionales dentro de la oficina de repositorio institucional.



2.2. MARCO CONCEPTUAL

2.2.1. Desarrollo se sistemas

El desarrollo de sistemas tiene un fin innumerable de necesidades, estas necesidades dependen mucho de los usuarios humanos y empresas, donde usualmente automatizan ciertos procesos en diferentes modalidades. Al desarrollar un sistema, implica comprender las necesidades de quienes optan este medio para luego poder analizar la información que ingresa constantemente, procesando o convirtiendo esta información, almacenándolos para luego generar datos de cualquier entidad en específica. Entonces haciendo un proceso de evaluación profunda y detallada, los que analizaron esta información buscan distinguir y solucionar datos deteriorados adecuados. Entonces, al iniciar con la evaluación, se comienza a describir diseños para luego así implementar las mejoras, para el apoyo a los usuarios finales, dándoles la sistematización haciendo el manejo de sistemas que son computarizados (Kendall, 2011).

Si al instalar un sistema sin ninguna planificación o proceso necesario, frecuentemente los usuarios finales desisten el sistema, haciendo que desconfíen y abandonen el sistema informático. Así que el análisis y diseño son parte fundamental de la estructura de un sistema informático, además que constituye una actividad muy costosa. La colaboración de quienes harán el uso del proyecto es de suma importancia para un desarrollo exitoso de los sistemas, el usuario avanza en un primera instancia frecuentemente que los grupos de desarrollo se interrelacionan gradualmente en cuanto a su formación, esto precisa que puede aumentar la relación en laborar con los usuarios finales del sistema para así realizar un estudio de su entidad, sus debilidades y objetivos, y en anunciar la



prueba y diseño del sistema que se planificó a todos los participantes (Kendall, 2011).

2.2.2. Sistema de información

Por lo general cuando se habla de sistemas de información se internan en diferentes aspectos de nuestra vida cotidiana, por ejemplo, cuando se hace una transferencia de dinero mediante un dispositivo móvil o cuando ingrese a una página web. Más que nada, en la comercialización, los sistemas de información basados en servidores y computadores producen y almacenan información que se utiliza, de tal modo que se convierten en una de las operaciones empresariales más exitosas y necesarias para la innovación y crecimiento de las mismas. Se puede resolver problemas y toma de decisiones, generar información, administrar los datos y recopilar información útil para los usuarios (Effy, 2008).

En forma general, es aquel conjunto ordenado de elementos que permiten manipular toda aquella información necesaria para implementar aspectos para tomar decisiones, de tal punto que tienen la necesidad de recolectar grandes cantidades de datos (Longatt, 2007).

2.2.3. Base de datos

Es una agrupación de datos relacionados entre sí, que están organizados de manera que pueda accederse a ellos de forma automática. Este aspecto contiene una corriente constante de información acerca de muchos periodos de la vida cotidiana, y no solamente dentro de un contorno personal, sino que también conlleva datos de diferentes temas y magnitudes elementales para una empresa o negocio. La tecnología de la base de datos ayuda a resumir este volumen de datos en información útil para la toma de decisiones a un largo plazo, como invertir en



tecnología, equipos, tiendas, elementos nuevos al inventario e ingresos (Mannino, 2010).

2.2.4. Sistema de gestión de base de datos

Se entiende que es un conjunto de datos no redundantes, almacenados en un soporte informático, organizados independientemente para su utilización y accesible por distintos usuarios y aplicaciones que lo requieran. Lo que se hace es que estos datos sean cada vez más solicitados y con la misma urgencia de gestionarlos. Cualquier dato o elemento informativo que tenga relevancia para un usuario es considerado una información importante y lo que hace es que estos sistemas almacenan y gestionan estos datos de manera eficiente ordenándolos por tablas, campos y registros, enlazándolos de una manera continua (Ricardo, 2009).

2.2.5. Lenguaje de modelado Unificado (UML)

Tal vez una de las innovaciones conceptuales en el desarrollo de software con aparición en muchos lenguajes conocidos por los programadores, así cumpliendo con grandes estándares que permite modelar sistemas de información, lo cual es lograr modelos de sistemas que puedan ser entendidos por los clientes o usuarios de aquello que se modela (Rumbaugh *et al.*, 2007).

2.2.6. Diagramas de clase

Estas gráficas representan la estructura estática de un sistema, es decir detallan las diferentes cualidades de un sistema y diferentes tipos de comunicaciones que se genera entre ellos. Esto se convierte en una técnica más estructurada para el modelado conceptual de un sistema de software, lo que conlleva a tener gráficas que representen infinitos escenarios a los que puede dar

origen un diagrama de clase, donde hay diferentes perspectivas que pueden utilizar los diagramas: (Aguilar & Peñalvo, 2016).

Conceptual: El diagrama representa conceptos del problema que se está estudiando, se crea con toda independencia de la implementación el sistema (Aguilar & Peñalvo, 2016).

Especificación: Refleja las interfaces de las clases, donde las clases son más cercanas a los tipos de datos (Aguilar & Peñalvo, 2016).

Clase: Es el detalle de grupo de elementos que colaboran con las mismas propiedades, cálculos o maneras, donde: (Aguilar & Peñalvo, 2016).

Tabla 1

Tipos de atributos de clase

Tipo	Descripción
(+) visibilidad pública	Quiere decir que será notorio por todos los elementos
(#) visibilidad protegida	Solo notorio por los elementos y sus sucesores
(-) visibilidad privada	No será notorio por nadie, solo por el elemento

Fuente: Diagramas de clase en UML (Aguilar & Peñalvo, 2016)

Asociaciones: Son relaciones más generales entre clases y estas asociaciones pueden ser binarias (Francisco, 2018)

Agregación: Es una asociación con la existencia que se irán estableciéndolas relaciones con cada una de las partes que lo constituyen a medida que se vayan necesitando (Francisco, 2018)

Composición: Representa una variación más restrictiva, es decir, los componentes de un objeto solo pueden pertenecer a un solo objeto agregado, de forma que



cuando este objeto es eliminado todas sus partes serán eliminadas también (Francisco, 2018).

Herencia: Va orientada desde la subclase a la superclase, es decir tiene la facultad de poseer sus propios métodos y atributos, que sean visibles a la superclase (Francisco, 2018).

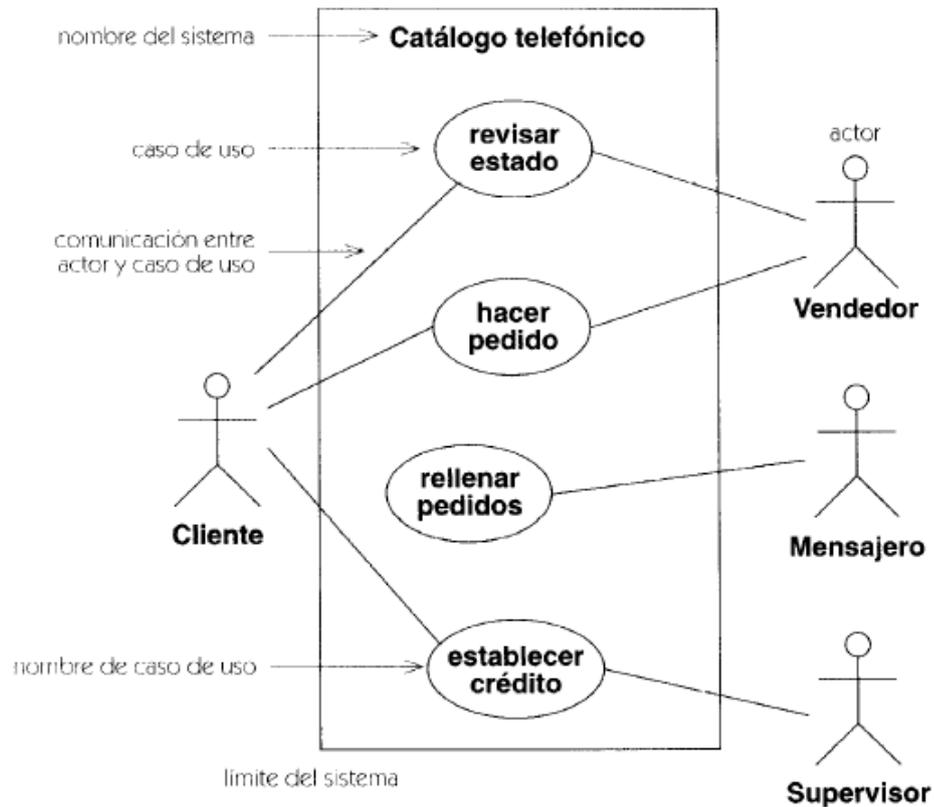
Dependencia: Es cuando una clase depende de otra para su cometido (Francisco, 2018).

Implementación: Representa las clases tal cual aparecen en la implementación (Francisco, 2018).

Diagramas de caso de uso: Hace una recepción de la conducta de un sistema de información desde un área donde el usuario pueda visualizarlo. Las piezas de la funcionalidad interactiva como está en la figura, lo cual. el caso de uso describe una interacción con los actores como una sucesión de mensajes entre el sistema y uno o más usuarios (Rumbaugh *et al.*, 2007).

Figura 1

Diagrama de caso de uso de un sistema de información.



Fuente: Lenguaje unificado de modelado. Manual de referencia (Rumbaugh *et al.*, 2007)

2.2.7. PHP

Lenguaje de programación de código abierto interpretado por un servidor, inicialmente diseñado para desarrollar scripts orientados a la web. Posee un mejor gestor de errores, buen monitor de búsquedas abstracciones de base de datos y otras funcionalidades donde permite al desarrollador hacer sencillas aplicaciones web dinámicas a diferencia de otros lenguajes de programación esta tiene velocidad, estabilidad, seguridad y simplicidad (Bahit, 2012)

2.2.8. HTML

Es un lenguaje diseñado para crear páginas web dándole formato a los textos o imágenes que se pretende ver en el navegador, con la intención de comunicar y compartir información (Vega & Van Der Henst, 2011)



Como su antecesor HTML, tiene características muy limitadas y hoy en día HTML5 con una actualización más reciente y con mejoras notables en compañía de CSS3 hacen de una página web muy dinámica y de excelente contenido. Pero a pesar de los grandes avances e innovaciones introducidas por CSS y JavaScript, HTML aún sigue siendo la parte fundamental del documento donde todo el contenido sea dinámico es la plataforma básica para toda aplicación (Gauchat, 2017).

2.2.9. CSS

Es un lenguaje que hace fácil las instrucciones que se puede usar para dar estilos a los elementos de HTML como color, letra, tamaños e imágenes, luego de implementar estos estilos, se da un mejor diseño visual que se quiere para una página web. Esta última versión agrega mayor control sobre los estilos de las paginas, dándole una mejor visualización por los estilos agregados (Navajas, 2012; Pressman, 2013).

2.2.10. MYSQL

Un sistema de gestión de base de datos de código abierto que puede administrar una gran cantidad de datos, gracias a su sencillez, flexibilidad y velocidad. Muchos lenguajes de programación se adaptan a este modelo de gestión de datos, haciendo que se pueda ejecutar en varios sistemas operativos (Mehta *et al.*, 2018).

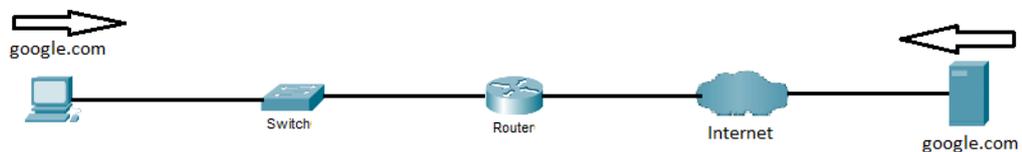
A lo largo de los años, este gestor de base de datos ha facilitado a grandes, medianas y pequeñas empresas a gestionar gran cantidad de datos, ordenándolos de una manera sencilla y fácil de encontrar, reduciendo así notablemente el costo por albergar información confidencial (Gilfillan, 2003).

2.2.11. Servidor Apache HTTP

Una de las plataformas web más utilizadas de código abierto y totalmente compatible con los nuevos estándares de HTTP, que permite que cualquier usuario puede utilizarlo para cualquier fin, ya sea para distribuir o modificarlo. Aunque apache puede utilizarse de muchas maneras, comúnmente se utiliza como servidor web. Un servidor es un software donde su oficio más resaltante es guardar, procesar y presentar páginas web a los usuarios y el protocolo utilizado para las entregas de estas páginas es HTTP (protocolo de transferencia de hipertexto) generalmente llamado como navegador web. El servidor web responde a la página que es solicitada he inmediatamente proporciona la información solicitada en un navegador web (Peicevic, 2016).

Figura 2

Proceso de transferencia y acceso a un servidor



Fuente: Apache HTTP server introduction (Peicevic, 2016).

2.2.12. Metodologías de desarrollo de software

En el área de desarrollo de software, las metodologías son un punto muy importante en para la creación de software, que fueron creadas con el propósito que el software desarrollado sea eficaz, de calidad y a corto tiempo, respetando las fases que permita a los desarrolladores tener un mejor control del proyecto. Existen diferentes metodologías de diferentes características que pueden afrontar de diferentes formas el desarrollo de un software, tratando de obtener el mejor

resultado posible, lo cual se elige una que contenga todos los procesos de creación de un producto de muy alta calidad (Maida & Pacienza, 2015).

Existen dos grandes grupos de metodologías para el desarrollo de software, tenemos las metodologías predictivas o tradicionales, caracterizadas por seguir total y rígidamente los requisitos al inicio de los proyectos y dentro de estas tenemos: cascada, prototipado, espiral, incremental y RAD. Las metodologías ágiles, definidos por un enfoque más sencillo que se basa en una rápida entrega de los proyectos, basado en su flexibilidad y agilidad, donde los ejecutores son más productivos y eficientes, y dentro de esta metodología tenemos: kanban, Scrum, Lean y programación extrema XP (Espinoza, 2013).

Tabla 2

Diferencias entre metodologías ágiles y tradicionales

Metodologías ágiles	Metodologías tradicionales
Basadas en heurísticas provenientes de prácticas de producción de código	Basadas en normas provenientes de estándares seguidos por el entorno de desarrollo
Especialmente preparados en cambios durante el proyecto	Cierta resistencia a los cambios
Impuestas internamente (por el equipo)	Impuestas externamente
Proceso medos controlado, con pocos principios	Proceso mucho más controlado, con numerosas políticas/normas
No existe contrato tradicional o al menos es bastante flexible	Existe un contrato prefijado
El cliente es parte del equipo de desarrollo	El cliente interactúa con el equipo de desarrollo mediante reuniones
Grupos pequeños (<10 integrantes) y trabajando en el mismo sitio	Grupos grandes y posiblemente distribuidos
Pocos artefactos	Mas artefactos
Pocos roles	Mas roles
Menos énfasis en la arquitectura del software	La arquitectura del software es esencial y se expresa mediante modelos
Poca documentación	Documentación exhaustiva
Muchos ciclos de entrega	Pocos ciclos de entrega

Fuente: *Metodologías de desarrollo de software* (Maida & Pacienza, 2015).



2.2.13. Programación extrema (XP)

Una metodología ágil que se basa en la simplicidad, comunicación y realimentación, así minimizando los errores y cambios repentinos del proyecto. Esta metodología se puede modificar de acuerdo a las necesidades de la entidad, adaptándose a los cambios repentinos en marcha que son aspectos naturales de un proyecto y XP facilita que los desarrolladores tengan una realimentación continua con el cliente y el equipo de desarrollo, simplicidad en las soluciones implementadas y rapidez para adaptarse a cambios (Maria *et al.*, 2016). Las variables como costo, tiempo, calidad y alcance, fijadas por la mayoría de los clientes, determinan que la funcionalidad en cada iteración es precisa, dada por los cambios, esta se puede manejar gracias a la máxima comunicación entre los desarrolladores y el cliente que es parte del equipo (Hernán, 2004).

2.2.14. Framework

Se utiliza en diferentes ámbitos del desarrollo de sistemas, una estructura de software compuesta de componentes personalizados e intercambiables para el desarrollo de aplicaciones. En conclusión, un framework es considerado como una app global sin completar y configurable, donde se consigue sumar fragmentos para diseñar una aplicación certera. Entonces al tener un framework se puede añadir diferentes códigos para el desarrollo de sistemas que se pueden reutilizar y así facilita y agiliza el desarrollo de las mismas (Gil Vera *et al.*, 2018).

Características:

Seguidamente, se presenta una serie de atributos que se pueden hallar en los framework más conocidos.

Tabla 3

Características y generalidades de los Frameworks

Generalidades	Descripción
Abstracción de URLs y sesiones	No se tiene que ser preciso al manejar derechamente las URLs ni los ingresos, el framework por oficio ya se encamina a poder realizarlo, incluyendo instrumentos e interfaces que son precisas para poder acoplar con herramientas para acceder a la información en BD, XML, etc.
Acceso a datos	Gran parte de los frameworks incluye diferentes series de computadores para poder administrar eventos, por ejemplo, al insertar datos por medio de formularios o al ingresar a una página web. Los computadores pueden ser adaptarse sencillamente a los requerimientos de un proyecto en específico.
Controladores	Tiene agregado procesos para poder verificar a los usuarios por medio de un nombre de usuario y su contraseña determinada previamente, lo cual permiten controlar el ingreso a determinadas direcciones web a diferentes usuarios.
Autenticación y control de acceso	Son instrumentos para poder mostrar la aplicación en todos aquellos idiomas que se considere oportuno.
Internacionalización	
Separación entre diseño y contenido	

Fuente: *frameworks para el desarrollo de prototipos WEB: un caso de aplicación* (Gutiérrez, 2014)

2.2.15. Modelo vista controlador (MVC)

Es un Patrón de arquitectura de aplicaciones, donde su función es simplificar la implementación de acuerdo a las necesidades de los usuarios y datos a manipular, separando así, los datos de la aplicación, el interfaz de usuario y la lógica de control, en componentes distintos de manera que al momento de hacer una modificación sea hecha con un mínimo impacto (Thanthirige *et al.*, 2016).

- **Modelo:** Encargado de presentar la estructura de tus datos, donde normalmente suele hacerse por interfaces con las mismas bases de datos.

- **Vista:** Realiza una definición de lo que contiene en las plantallas que son compuestas en una página web, así como la información de salida.
- **Controlador:** Controla los request que hace el usuario, se comunica con el modelo (base de datos) y nos muestra la vista (HTML) (TEBAR, 2020).

Definámoslo con un ejemplo la estructura del modelo vista controlador:

Cuando se tiene una aplicación, se tiene una parte donde el usuario interactúa con una ventana, esa ventana podría ser definida como la vista y ahora para que pueda funcionar esa ventana se necesita algo de código, entonces el código es definido como el controlador. Ahora lo que el usuario ingresa en la ventana y lo cual maneja ese código que es el controlador, necesita ser almacenado, entonces se define otra vez como un modelo (los datos), es decir los datos son definidos como un modelo.

Figura 3

Diagrama de la estructura de modelo vista controlador (MVC).





Fuente: Introducción a la arquitectura de software un enfoque práctico (Blancarte, 2020)

2.2.16. Seguridad informática

Al hablar de seguridad informática se entiende que hay criterios en cuando a la ciencia y disciplina que se enfoca a guardar información, lo cual consiste en el bien estar de la información y cuáles son los riesgos que existe al manipularlos. La seguridad de un sistema que limita cierta facultad de ingreso a la información y medios de un sistema mediante procedimientos de reconocimiento y observación, donde se aseguren que los manipulantes de estos recursos tengan permitido ingresar de forma limitada siempre en cuando que se les concedan. Sin embargo, estos mecanismos por lo general causan molestias a los usuarios, ya que las normas y reglamentos se complican cada vez más a medida que la información crezca (Romero *et al.*, 2018).

Seguridad de TI es minimizar los riesgos asociados al acceso y el uso de esa información de forma no autorizada, en ese punto la seguridad implica la necesidad de una gestión para así tener mejor control del acceso a la información y con qué fines se planea utilizar, para ello, se evalúa por diferentes protocolos de acceso, con el fin de determinar paso a paso la identificación de quien solicita la información y que esta solo puede estar sujeta de limitaciones dadas por los administradores (Delgado, 2017).

La ley de protección de datos personales, una normativa peruana, tiene como objetivo proteger los derechos fundamentales de las personas en cuanto al tratamiento de sus datos personales. La ley da ciertos valores y condiciones que se debe realizar en las entidades públicas y privadas que manejan información privada, con el propósito de dar mejor garantía la privacidad y resguardo de la



misma. Se establece que las entidades que manejan datos personales están en la obligación de dar el oportuno conocimiento a los propietarios de dicha descripción sobre la finalidad y el uso que se les dará a sus datos, así como los derechos que tienen para acceder, reciclar o cancelar la información proporcionada. Así mismo la ley establece que las entidades deben contar con medidas de seguridad adecuadas para proteger los datos personales y prevenir su pérdida, acceso no autorizado, modificación o divulgación (Edición *et al.*, 2013; Perú, 2011).

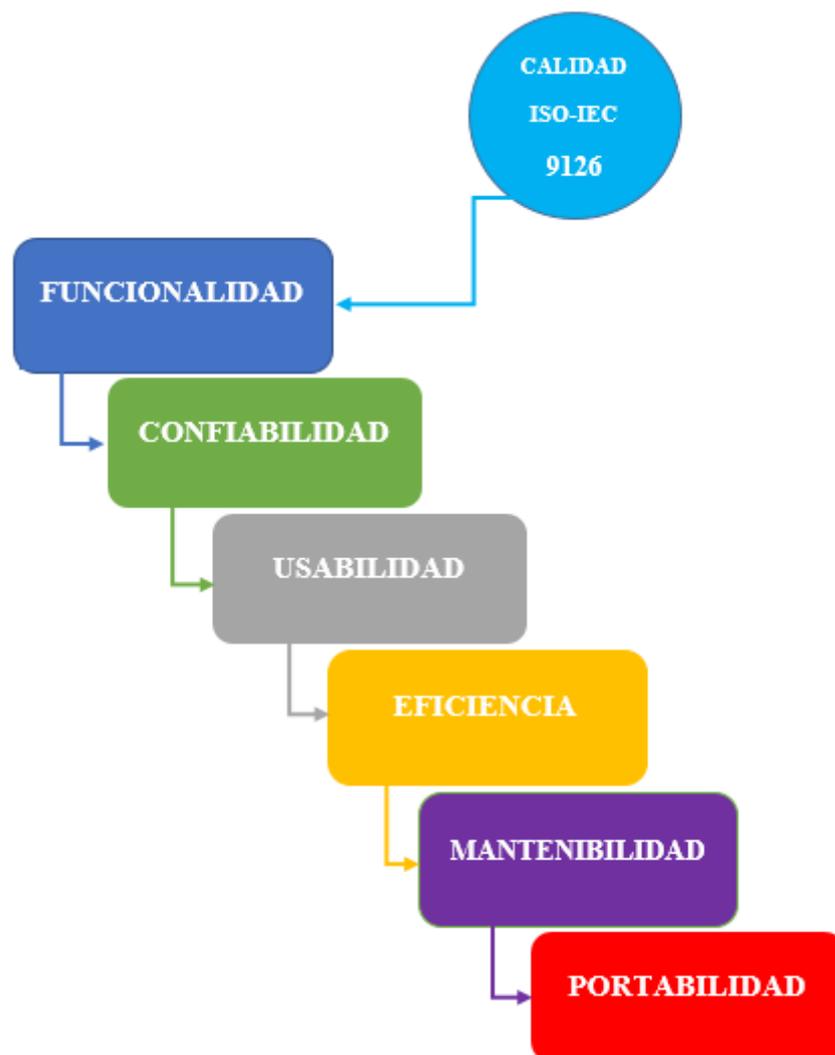
2.2.17. Organización internacional de estandarización 9126 (ISO 9126)

Un estándar internacional creada en 1992 para la evaluación de calidad de software de forma que el conjunto de características y sub características de calidad, que son atributos que se miden mediante métricas. La condición de software es la altitud en el que el usuario distingue que el software realiza con lo previsto, cuando estas métricas de calidad de software no se cumplen, los programas realizados bajo este estándar son calificados como inservibles sin mencionar que abandonan el software en su totalidad, dejando gastos y tiempo. La dificultad en gran parte de los programas realizados bajo este estándar no se puede cualificar simplemente, ya que se implanta de forma peculiar, lo que hace más difícil su medición, ya que se tiene que precisar instaurar normas donde se permita estimar cualitativamente cada una de las características siempre se subordina el tipo de sistema que se procura evaluar. En la ISO 9126 hay elementos que se deben considerar al momento de aplicar la evaluación de software de acuerdo a este estándar de modo que todo aquel que esté dispuesto a aplicar esta norma, pueda generar sus propias métricas bajo la guía que se presenta (Figuerola, 2012).

La ISO 9126 establece un modelo de calidad internacional, donde hay 6 características básicas las cuales son: la funcionalidad, confiabilidad, usabilidad, eficiencia, mantenibilidad y portabilidad, cada una de estas características serán detalladas a continuación y así profundizar la evaluación de calidad de software, pero estas características tienen cuestionamientos que atienden cada una de las características ya mencionadas (Figuroa, 2012).

Figura 4

Características básicas de la ISO/IEC 9126



Fuente: Estándar ISO/IEC 9126.



Características propuestas por ISO-9126:

1. Funcionalidad: Permite calificar si un software puede ser manejado de forma adecuada, es decir, si el conjunto de las funciones que ofrece satisfagan las necesidades para las cuales fue diseñado, es decir determina la capacidad del software de funcionar como el usuario necesita de interactuar con otros sistemas, que exista seguridad y permisos, pero que cumpla con las regulaciones con las leyes, donde se establece los siguientes atributos: (Figuroa, 2012; International *et al.*, 1998; Caso, 2018)

- Adecuación: Enfocado a la evaluación de acciones propias para los trabajos que están asignadas, en función a las tareas específicas y objetivos de los usuarios.
- Exactitud: Examina si en las conclusiones hay efectos de requerimientos que fue creado, dando resultados precisos.
- Interoperabilidad: Define si el software tiene la capacidad de operar o interrelacionarse con otros sistemas, relacionando así resultados finales, como archivos u otros que competen dentro del software.
- Conformidad: Hace lo que debe hacer, es decir realiza las funciones para lo que fue creado, adhiriéndose así a los estándares.
- Seguridad: Tiene una barrera para anticipar el ingreso que no se autoriza a los usuarios, puede ser accidentalmente o intencionado, evitando así la modificación o mala manipulación de los datos

2. Confiabilidad: el software tiene la capacidad de mantener su nivel bajo condiciones establecidas, durante un periodo de tiempo establecido, donde se establece diferentes atributos: (Figuroa, 2012; International *et al.*, 1998; Caso, 2018)



- **Madurez:** Mide la frecuencia de fallas durante la ejecución del software con la finalidad de reducir estas fallas al mínimo.
 - **Tolerancia a fallos:** Indica que tiene la destreza de establecer un grado característico de la función, para que, en caso de tropiezos, el sistema siga funcionando.
 - **Recuperabilidad:** Si se puede reestablecer su funcionalidad normal, es decir, recobra los datos que hayan sido perjudicados directamente por un desperfecto al momento de la ejecución de software.
 - **Conformidad:** Cumple con las relaciones y normas establecidas, relativas a la conformidad, dando así una fiabilidad en el manejo del sistema.
- 3. Usabilidad:** Los atributos en esta característica, permite evaluar el esfuerzo que será necesario para utilizar el sistema, y son los siguientes: (Figueroa, 2012; International *et al.*, 1998; Caso, 2018)
- **Comprensión:** indica al nivel de entendimiento requerido para entender el funcionamiento lógico del sistema, características de la implementación del software y como se debe utilizar para las actividades y circunstancias particulares de sistema.
 - **Aprendizaje:** el atributo donde usuario está en la capacidad de aprender y saber para que se usa el sistema, las extensiones y módulos que se pueden usar a profundidad del sistema.
 - **Operatividad:** Operación y control del sistema, es decir el usuario tiene la facilidad de manejar y ejecutar los diferentes módulos del sistema.
 - **Atractividad:** Es la atracción del sistema en cuando al manejo, lógica e interfaces, siendo de grado y comodidad.



4. **Eficiencia:** Esta característica, permite evaluar la relación entre el nivel de desempeño del software y los recursos usados bajo condiciones establecidas (Figuroa, 2012; International *et al.*, 1998; Caso, 2018).
 - Comportamiento en el tiempo: referido al tiempo de respuesta, tiempo de inicio, tiempo de rendimiento, tiempo de funcionamiento y tiempo al procesar datos.
 - Utilización de recursos: Carácter de medios utilizados y la permanencia en realizar las tareas asignadas.

5. **Mantenibilidad:** Está relacionado con los atributos de modificación, extenderse o corregirse, a todas estas características se le llama mantenibilidades, tiene la capacidad de mantenerse en función al tiempo, de adaptarse, de extenderse o de mejorar en función al uso que se le está dando (Figuroa, 2012; International *et al.*, 1998; Caso, 2018)
 - Análisis: se hace un diagnóstico de las debilidades o causas de averías y el reconocimiento de los módulos a ser alteradas y así dar una corrección.
 - Modificación: tiende a hacer flexible a cambios y modificaciones, dando así los permisos necesarios para que los cambios sean implementados.
 - Estabilidad: tiene la capacidad de tener un desempeño normal y estable, cuando el sistema tiene algunos cambios o modificaciones tiene la capacidad de evitar reacciones inesperadas debido a los cambios recientes.
 - Facilidad de prueba: antes de implementar un sistema, esta tiene que tener la facilidad de hacer test previos al lanzamiento, evaluando el estado del programa.



- 6. Portabilidad:** conjunto de atributos relacionado con la capacidad de ser transferido a diferentes plataformas o versiones y se considera los siguientes aspectos: (Figuroa, 2012; International *et al.*, 1998; Caso, 2018)
- Adaptabilidad: indica si el sistema se puede usar en varias plataformas o distintos momentos sin ser necesario hacerle alteraciones.
 - Simplicidad de instalación: la fácil instalación del sistema en un entorno ya definido
 - Capacidad de reemplazo: facultad del sistema para que se pueda manejar en reemplazo de un diferente sistema, para el mismo trabajo, con las mismas funciones y en el mismo entorno.
 - Conformidad: indica si el sistema tiene la capacidad de apegarse a las características o momentos que tienen relación a lo portable.
- 7. Calidad de uso:** Es aplicado a los diferentes atributos del producto y que permiten determinar posteriormente los niveles de calidad de uso, es decir, la forma como el usuario final logran realizar los procesos con satisfacción, eficiencia y exactitud, lo cual mide la extensión que pueda conseguir sus metas en un ambiente particular (Domínguez, 2016). A continuación, se muestra los atributos de esta característica:
- Eficacia: permite al usuario final realizar los procesos con mayor precisión, integridad y plenitud en el carácter del uso específico.
 - Productividad: característica que relaciona al producto en las asignaciones cotidianas que se realizan por el usuario final.
 - Seguridad: que tan seguro es el sistema para el usuario en cuando al manejo de información sensible, ante robos, plagios o copias.



- Satisfacción: el usuario está satisfecho con el sistema, porque cumple con todas las métricas de calidad, haciendo que esta sea agradable y de fácil uso.

2.2.18. Gestión académica

Organismo que se encarga de adecuar los recursos disponibles en los procesos que permiten lograr los resultados y el propósito institucional. Si bien estos compromisos se han ido ajustando desde las primeras versiones hasta la actualidad, el sentido de cada una de estas se mantiene vigente, sin duda, se ha mejorado el alcance de gestión de varios procesos administrativos, donde se logra mayor precisión y reforma de aquellos compromisos que se requieran. La gestión académica o escolar, requiere de prácticas que permitan generar condiciones favorables y seguras de aprendizaje por la institución, y no solo en temas académicos, sino también en áreas administrativas que van organizando estas diligencias con el fin de promover los recursos y acciones para lograr el objetivo (Minedu, 2015).

Una buena gestión escolar o académica, depende mucho de la eficiente administración de los recursos, así como también, el conjunto de acciones articulares entre sí, que emprende el equipo directivo de una escuela, para promover y posibilitar la consecución de la intencionalidad pedagógica en y con la comunidad educativa (MINEDU, 2014).

La gestión académica apoyadas con las tecnologías de la información y comunicación (TIC) se refiere a la utilización de herramientas para no solo mejorar procedimientos y actividades relacionadas a la educación, sino que estas tecnologías utilizan diferentes recursos con el objetivo de facilitar diferentes tareas



administrativas y pedagógicas, como promover e incentivar y mejorar la calidad de la educación, así como la administración y seguimiento de datos, comunicación y colaboración, plataformas y recursos educativos en línea, aprendizaje virtual y por ultimo evaluación y seguimiento de rendimiento. Tenemos que tener en cuenta que la implementación de las TIC en la gestión académica requiere una infraestructura tecnológica adecuada, promoviendo e incentivando iniciativas para fomentar el uso de las TIC (Espinoza, 2018).

2.2.19. Ingeniería de software

Una disciplina y área de la informática y computación, que muestra métodos y técnicas para el desarrollo y mantener software de calidad que omite problemas de todo tipo, implantado por el mundo laboral, con la finalidad de reducir u omitir considerablemente las actividades del ser humano en las actividades cotidianas. La ingeniería de software es aplicada bajo estándares científicos aplicados en el diseño de elaboración de software para computadores y manualidades requerida relacionada a procesar, intervenir y establecerlos. En general los ingenieros de software adoptan un enfoque sistemático en su trabajo, ya que es la forma más afectiva de producir software de alta calidad, sin embargo, aunque la ingeniería consiste en seleccionar el método más apropiado para un conjunto de circunstancias, más formal y creatividad para el desarrollo efectivo en algunas circunstancias, el cual requiere una mezcla de técnicas de software y diseño gráfico (Sommerville, 2005).

- Gestión del proyecto: orden general para todo tipo de planificaciones de la ciencia de la información y comunicación, y otros ámbitos, como la elaboración, la ingeniería y la administración de empresas, entre otros



ámbitos, incluyen la diligencia y administración de cualquier planificación, con independencia de que la utilidad en concreto se esté elaborando, con el objetivo principal de asegurar su éxito total. En resumen, tiene ciertas características de gestión, como (Miquel & Martos, 2006):

- Estudio de viabilidad.
 - Estimación.
 - Definir claramente la finalidad del proyecto, que aclarará su éxito o fracaso.
 - Asignar el equipo de desarrollo considerando el cálculo de los recursos hecha inicialmente.
 - Establecer hitos
 - Identificar cualquier riesgo que pueda hacer vulnerable el éxito del proyecto
- Identificación y administración de los requisitos: actividad donde implica la comunicación y el apoyo de los usuarios, principalmente para hallar cuales son las condiciones del producto a desarrollarse. Los principales problemas a superar son propios de cualquier actividad de comunicación (Miquel & Martos, 2006):
 - Diferencias respecto a la información con la que trabajan las diferentes partes.
 - Limitaciones del canal utilizado.
 - Limitaciones del lenguaje utilizado.
 - Dificultad de definir el mejor sistema posible.

Para la solución de estos problemas, es habitual usar técnicas de retroalimentación durante el análisis de requisitos cuanto esto es posible.



- **Modelización:** construcción de pautas del sistema que se tiene que elaborar; estas pautas facilitan la comprensión de las condiciones y la estructura del sistema, en cierto modo, es la conformación de maquetas o diseños, con la diferencia de que, al ser el software un fruto intangible, estos diseños no suelen tener una naturaleza física, sino que son propios del software. En la actualidad hay programas donde se puede crear modelos de software, es el caso de UML, que se define como una serie de diagramas que nos permite elaborar nuestros modelos (Miquel & Martos, 2006).
- **Construcción y pruebas:** incluye el código fuente y la realización de evaluaciones necesarias para asegurar el funcionamiento, con la ausencia de fallas y la adaptación del producto a las condiciones. También se tienen en cuenta las actividades relacionadas con la gestión de la configuración y administración de cambios (Miquel & Martos, 2006).
- **Calidad:** esta actividad incluye todas las etapas de desarrollo, donde se define criterios que se admiten enfoques del sistema y en qué momento el plan estará finalizado satisfactoriamente. Normalmente la gestión de calidad recoge todas las métricas que ayuden a determinar si el software cumple con los criterios de calidad y también la documentación formal de los procesos de desarrollo y por último la verificación de su cumplimiento (Miquel & Martos, 2006).

2.2.20 Principios de usabilidad

Los principios de usabilidad son guías y procedimientos para el construcción y diseños basados en interfaces fáciles y amigables entendibles, dando una buena comodidad al usuario final, entonces se presenta los principios comunes de usabilidad:



Visibilidad del estado del sistema: El interfaz del sistema debe ser fácil de comprender, dando paso a los usuarios encaminarse para adaptarse rápidamente en el funcionamiento y usarlo sin dificultad. En este caso, el sistema debe estar siempre informando al usuario de lo que está haciendo. (Nielsen, 2017).

Relación entre el sistema y el mundo real: El software permite a los usuarios finales trabajar las tareas de forma eficiente, reduciendo la cantidad de acciones que no sean necesarias para realizar una tarea, donde proporciona ser más práctico y abreviados posibles. (Nielsen, 2017).

Control y libertad del usuario: El software se adapta a diferentes tipos de usuario y situaciones, donde permite ajustes de personalización, lo cual ayuda al usuario a poder usarlo con todas las necesidades y preferencias posibles. (Nielsen, 2017).

Consistencia y estándares: El sistema debe mantener las características similares a donde ya se estaba trabajando, manteniéndolas siempre sin saber que las situaciones o acciones significan lo mismo. (Nielsen, 2017).

Prevención a errores: Los mensajes de error son importantes, pero se tiene en cuenta que estos mensajes deben prevenir los errores, ayudando al usuario a un entendimiento del sistema. (Nielsen, 2017).

Reconocimiento antes que recuerdo: Por lo general los interfaces del sistema ayudan al usuario a intuir el siguiente proceso en el sistema, estos procesos son correlativos evitando que el usuario memorice las acciones que hizo con anterioridad. (Nielsen, 2017).

Flexibilidad y eficiencia de uso: Los atajos y aceleraciones pueden hacer una iteración rápida para aquellos usuarios que son expertos, de modo que el sistema sea útil tanto para principiantes como avanzados. (Nielsen, 2017).



Diseño estético y minimalista: El sistema no debe contener información extra e irrelevante que confunda al usuario, en dada situación, el sistema solo debe dar a conocer que tipo de contenido se necesita para el usuario (Nielsen, 2017).

Ayuda a los usuarios a reconocer, diagnosticar y recuperar errores: Como ya se mencionó anteriormente, los mensajes de error le dan un valor agregado al sistema, indicándole al usuario una sugerencia constructiva hacia el problema (Nielsen, 2017).

Ayuda y documentación: Por lo general un sistema no requiere una explicación adicional, pero en ciertas situaciones de debe dar una línea de guía para que el usuario entienda y continúe haciendo sus actividades (Nielsen, 2017).



CAPÍTULO III

MATERIALES Y MÉTODOS

3.1. LUGAR DE ESTUDIO

La actual investigación se ejecutó en la Institución Educativa Secundaria Industrial 32, ubicado en la Av. Simón Bolívar N° 1505 de la ciudad de Puno, Provincia de Puno, Departamento de Puno.

3.2. TIPO DE INVESTIGACIÓN

La presente investigación, según la propiedad del problema, así también como los objetivos y la hipótesis se encuentra dentro del tipo cuasi experimental. De acuerdo con (Castell, 2017; Sans *et al.*, 2012) la investigación cuasi experimental es el establecimiento de hipótesis y la determinación de los objetivos del estudio realizado, tratando de definir cuáles serán las variables que se consideran en la investigación, como se controlan y se miden, la manera de filtrar y tratar datos, y por ultimo cual sería la forma de analizar estos datos.

3.3. DISEÑO DE LA INVESTIGACIÓN

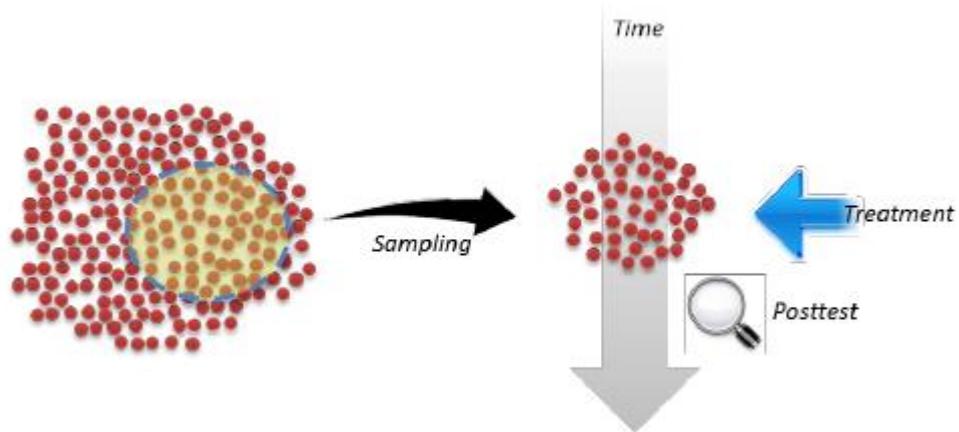
3.3.1 Cuasi experimental

Según (Castell, 2017; Hernández & Mendoza, 2018) indican que en este diseño, el grado de control es mínimo. Se usa para establecer un primer acercamiento al objeto de estudio, donde consiste en administrar un estímulo o tratamiento, después de aplicar en una o más variables. Por lo tanto, presentan menos requerimientos que los diseños experimentales y solo se centran en un grupo, donde se distingue dos tipos de diseños pre-experimentales:

Solo post-test: no se evalúa los cambios del estado inicial al final, sino que solo son mediables distintos parámetros del estado final del grupo.

Figura 5

Solo con la aplicación de post-test.

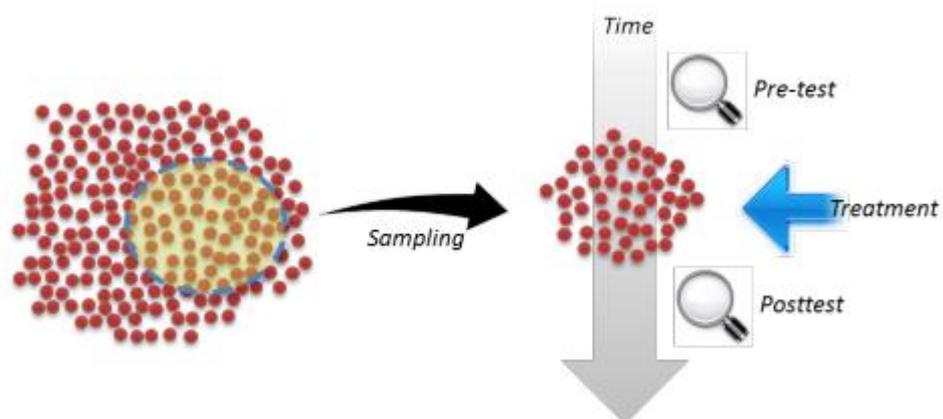


Fuente: metodología de la investigación (Castell, 2017).

Con pre-test y post-test: permite un seguimiento de los cambios que se experimenta una muestra a partir de la medición de ciertos parámetros en el estado inicial y final.

Figura 6

Con la aplicación de pre-test y post-test.



Fuente: metodología de la investigación (Castell, 2017).



En esta investigación, el esquema que le corresponde es de la siguiente forma:

G1:O1X O2

Donde:

G1: Grupo experimental, conformado por los administrativos y docentes que laboran dentro de la institución educativa.

O1: Identificación de la **pre-prueba** antes de aplicar el experimento al grupo seleccionado, para ver si el estado actual de la gestión de información antes de la implementación del sistema informático en la institución.

X: Aplicación con el sistema informático.

O2: Identificación de la **post-prueba** después de aplicar el experimento al grupo seleccionado, para así identificar el efecto de la implementación del sistema informático en la institución.

3.4. METODOLOGÍA DE DESARROLLO

El desarrollo de software se asocia hacia el manejo y control de información que se quiera manipular, en este caso, en el desarrollo de este proyecto se opta por la metodología XP, que es una metodología ágil y una de las más requeridas actualmente y, por consiguiente, se utilizó el estándar de calidad de software ISO/IEC 9126 para la evaluación del sistema

3.5. POBLACIÓN Y MUESTRA

3.5.1. Población

Compuesta por un conjunto de casos, definido, limitado y accesible, que forma parte de la elección de la muestra, y que cumple con una serie de criterios predeterminados. Es necesario aclarar que cuando se habla de población de



estudio, se refiere a los elementos como es el caso de personas, objetos, organismos, familias etc., que participan del análisis definido y delimitado del problema de investigación (Arias-Gómez *et al.*, 2016; Neftali, 2016).

En el caso de este proyecto se toma como población a los administrativos y docentes de la Institución Educativa Secundaria Industrial 32 de Puno, constituida por 65 miembros, quienes usarán el sistema de información.

3.5.2. Muestra

Es parte de una población, donde se define por un subgrupo, porción o parte de la población de interés, que refleje las mismas características de la población (Arias-Gómez *et al.*, 2016).

En este caso la unidad de muestro son los administrativos y docentes que la institución.

Calculo del tamaño de la muestra

Entonces, en la investigación, conocemos que la población es finita, ahora deseamos calcular el tamaño de la muestra, donde se formula de la posterior manera:

- N = Total de la población.
- Z = Valor tabular.
- P = Probabilidad de que ocurra el evento estudiado 50% =0.5
- q = 1 – p probabilidad de que no ocurra el evento estudiado.
- E = Error de muestra 5%

$$n = \frac{N * Z_{\alpha}^2 * p * q}{E^2 * (N - 1) + Z_{\alpha}^2 * p * q}$$



$$n = \frac{65 * 1.96^2 * 0.5 * 0.5}{0.05^2 * (65 - 1) + 1.96^2 * 0.5 * 0.5}$$

$$n = \frac{62.426}{1.1204}$$

$$n = 55.7176$$

$$n = 56$$

Por lo tanto, se obtuvo el tamaño de la muestra $n = 56$, para esta investigación se tomó el tipo de muestreo por fijación proporcional, constituida por los docentes y administrativos.

3.6. TÉCNICAS DE RECOLECCIÓN DE DATOS

Entrevistas: se refiere a una situación de interrogación y dialogo entre personas, el entrevistador y el entrevistado, donde se aplica diferentes modalidades, como entrevistas libres, estructuradas, focalizadas, simultánea y sucesiva (Kerlinger, 2008). Se utiliza este método de recolección de datos para hacer un análisis previo de los requerimientos que demanda la institución, luego tener las declaraciones de los usuarios sobre la administración de información que se maneja dentro de la institución, teniendo así una precepción de la utilización del sistema que se lleva a cabo en la implementación y evaluación del sistema.

Encuestas: establece un contacto con las unidades de observación por medio de cuestionarios previamente establecidos, lo cual se recolecta datos por medio de correos, teléfono, encuestas personales o vía online (Kerlinger, 2008). Por segundo punto, se aplicó la encuesta directa después de haber implementado SIGAINDUSTRIAL32 para así ver el cambio del manejo de información al implementar el sistema tiene significativos.



3.7. METODOLOGÍA DE DESARROLLO DE SOFTWARE

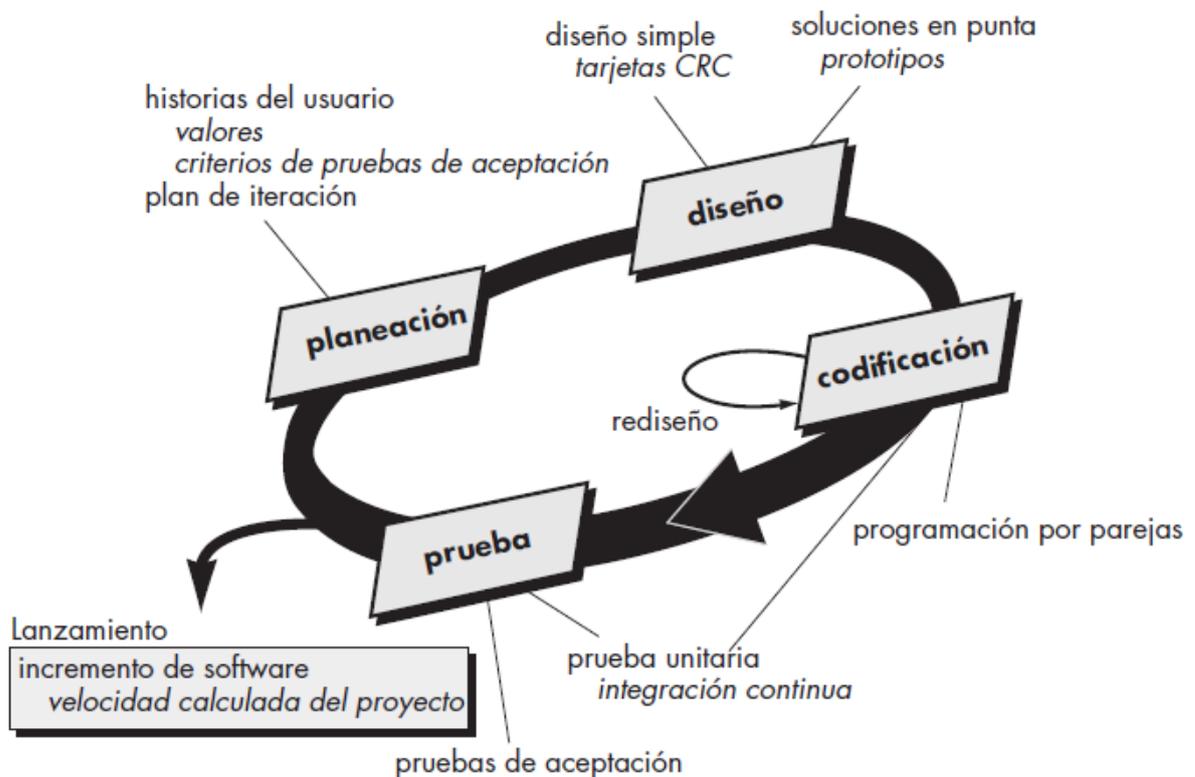
Programación Extrema (XP)

Uno de los procedimientos de desarrollo y ejecución de software ágil, donde se puede conseguir buena comprensión entre participantes, sencillez, comprensión continua, gallardía y atención. Cada una de estos trabajos se utiliza dentro un empuje para las acciones, asignaciones y acciones precisas con propósito de lograr la comunicación eficaz entre los programadores y los participantes (usuarios) que también son parte fundamental al momento de desarrollar un software. Además de poner en un punto alto la colaboración precisa pero formal entre los usuarios y codificadores, establece metáforas para que la comunicación de procesos importantes, en el entendimiento continuo y así evitar la rigurosa documentación innecesaria con el medio de comprensión mutua. Al diseñar e implementar un sistema, emerge una serie de estrategias de pruebas eficaces para que el software tenga toda la retroalimentación al equipo ágil, así XP usa la prueba unitaria como su táctica principal de pruebas, donde, a medida que se desarrolla cada clase, el equipo implementa una prueba unitaria para ejecutar cada operación de acuerdo a su funcionalidad especificada (Pressman, 2013).

XP usa un enfoque orientado a objetos referido al desarrollo, y encierra un conjunto de reglas y prácticas que ocurre en el contexto de cuatro actividades estructurales muy importantes, como: planeación, diseño codificación y pruebas (Pressman, 2013).

Figura 6

Proceso de la programación extrema



Fuente: Ingeniería de software, enfoque práctico (Pressman, 2013).

Actividades clave de XP

- **Planeación:** También llamado juego de planeación donde se empieza a recolectar demandas que permite al grupo comprender la dinámica de la empresa para el sistema y adquieran la flexibilidad de la salida y enfoques precisos que se requieran, creando historias que detallan la salida concisa de las funciones del sistema. Las tareas asignadas son detalladas por los clientes colocando según el nivel de prioridades en tarjetas, dándole un valor general de las características de las funciones de la entidad, ahora es de suma importancia saber cuál es el momento de describir más de una historia. El equipo XP ordenan las historias que serán desarrolladas en una de tres formas; todas las historias se implementan de inmediato, las historias con más valor entrarán a la programación de actividades



y se implementaran en primer lugar o las historias más riesgosas formaran parte de la programación de actividades y se implementan primero (Pressman, 2013).

- **Diseño:** En esta etapa se continua estrictamente la inicialización de mantenerlo, donde un diseño simple siempre se selecciona en vez de una más difícil, además de encontrar problemas, XP da pautas para la creación rápida de una muestra operativa de lo que será la porción del diseño final, entonces esta implementación se llama solución en un punto. La orientación es restar todos los riesgos cuando comience la verdadera ejecución y así poder dar el visto bueno de los cálculos originales para las historias que contienen la dificultad de diseño. También una de las fases de diseño involucra al rediseño, donde este proceso cambia un sistema de software en tal forma que no altere el comportamiento externo del código, pero si mejorar la estructura interna, en sí, es una manera disciplinada de limpiar el código que minimiza la probabilidad de introducir errores (Pressman, 2013).
- **Codificación:** Después de desarrollar las historias y terminar con la estructura prediseñada, el equipo no da como inicio con el desarrollo, entonces se realiza una serie de evaluaciones únicas a cada una de las tareas que se entregan en el área. Una vez realizada esta evaluación, el codificador entiende mejor la estructura para centrarse en lo que se debe implementar, es decir la programación en parejas, este proceso es una característica muy útil dando soluciones a problemas en manera rápida y precisa. A medida que los programadores en pareja terminen su trabajo, el código desarrollado se integra al trabajo de los demás, donde cada pareja tiene diferentes funciones (Pressman, 2013).
- **Pruebas:** Se informó de las evaluaciones de forma adjunta que pueden llegar a ser un elemento resaltante dentro de sus características XP, lo que estas evaluaciones crean, se deben implementarse con el uso de la estructura de tal

manera que puedan realizarse en continuas y varias veces con mucha agilidad y simplicidad, generando siempre que se modifique el código. A medida que las pruebas unitarias se organizan, las evacuaciones de unificación y validación pueden desarrollarse a diario.

Este instrumento es como saber que cuestionamientos vendrán en una evaluación antes de iniciar a estudiar, se convierte mucho más sencillo, porque se concentra toda la atención en los cuestionamientos que se van a dar por respuesta. Estas pruebas de aceptación se derivan de las historias de los usuarios que se han implementado como parte de la liberación del software (Pressman, 2013).

3.8. MÉTODO DE VALIDACIÓN DE SOFTWARE

Estándar de calidad de software ISO/IEC 9126

Como se explicó con anterioridad, la calidad de software está definida como coincidencia en los requerimientos funcionales y de lo que rinde, precisamente que fue establecido, lo que indica, es cumplir con todos los establecimientos y estándares dados por la ISO 9126 mencionadas en una ficha:

Tabla 4

Ficha de evaluación de calidad de software ISO 9126

Característica	Interrogante	Sub características
Funcionalidad	¿las funciones y propiedades satisfacen las necesidades explícitas e implícitas; esto es, el que?	Adecuación Exactitud Interoperabilidad Conformidad
Confiabilidad	¿alcanza permanecer continuamente la altura del funcionamiento, bajo diferentes criterios y por cierto lapso de tiempo?	Madurez Tolerancia a fallas Recuperabilidad

Característica	Interrogante	Sub características
Usabilidad	¿El sistema, es sencillo de manejar y de comprender?	Entendimiento Aprendizaje Operabilidad Atracción
Eficiencia	¿Es ágil y preciso referido al manejo de los suministros , bajo diferentes criterios?	Comportamiento de tiempos Utilización de suministros
Capacidad de mantenimiento	¿Es simple de modificar y testear?	Capacidad de poder ser evaluado Cambiabilidad Estabilidad Facilidad de prueba Adaptable
Portabilidad	¿Es simple de pasar de un momento determinado a otro?	Facilidad de alojar Remplazabilidad Existencia mutua Eficacia

Fuente: Ficha de evaluación ISO 9126.

3.9. PRUEBA DE HIPÓTESIS

- H0: La implementación del Sistema Informático para la gestión académica no mejora significativamente el manejo y almacenamiento de información de la institución educativa secundaria industrial 32 de Puno.
- H1: La implementación del Sistema Informático para la gestión académica mejora significativamente el manejo y almacenamiento de información de la institución educativa secundaria industrial 32 de Puno.

Con un nivel de significancia del 0.05 y con un error del 5%.

3.10. REQUERIMIENTOS DEL SISTEMA Y MATERIALES

3.10.1. Requerimientos funcionales

- Se considera tres usuarios: coordinador, docente y estudiante.
- Coordinador registra, elimina, edita estudiante.



- Coordinador registra, elimina, edita docentes.
- Coordinador registra, elimina, edita curso.
- Coordinador matricula estudiante en sus respectivos cursos.
- Coordinador asigna cursos a los docentes.
- Docente coloca notas únicamente de sus cursos que se le ha asignado.
- Estudiante observa reporte de notas por año.
- Deben existir perfiles para el ingreso e inicio del sistema, donde se comienza a dar mayor realce en cuando a seguridad y manejo de la información.
- El sistema debe registrar a nuevos usuarios como es el caso de administrativos, docentes, estudiantes y padres de familia
- La aplicación debe admitir modificar la información de nuevos administradores como es el caso de administrativos, docentes, estudiantes y padres de familia
- El software debe cumplir con la capacidad de editar datos
- El software debe permitir emitir consultas
- El software debe contar con un módulo donde se genera reportes sobre los datos de los educados, padres, docentes y administrativos.

3.10.2. Requerimientos no funcionales

- Aplicación cooperación mutua
- Una interfaz amigable para el fácil comprensión y manejo del sistema
- Disponible para el uso de cualquier administrador del sistema
- Disponible para todos los días laborales
- Mantenimiento, estabilidad y escalabilidad para posibles alteraciones, actualizaciones y posible incremento del software.



3.10.3. Requerimientos técnicos

Hardware:

- Computadora portátil o de escritorio con procesador Intel Core i3, RAM 4GB, Disco duro de 250 GB en adelante
- Memoria USB 3.0 de 64 GB
- Impresora de sistema de tinta continua
- Monitor LED 19.5
- Disco duro externo 3.0 de 1 TB

Software:

- Sistema operativo Windows 7 en adelante
- Navegador Chrome, Firefox, Microsoft Edge
- Sublime Text 3.2, JavaScript, JQuery 3.2, Css, Html
- Apache
- Servidor web
- Xampp (MySql, PHP 8)
- Netbeans-java
- StartUML, rational rose, rational rose modeler
- Microsoft office (Word, Exel, PowerPoint, 2016)
- RStudio

Materiales de escritorio:

- Información y datos referidos acerca de los procesos de matrículas, cursos, notas, información de padres, estudiantes, docentes y personal administrativo.
- Reglamentos de administración interna
- Manejo de implementos de escritorio



CAPÍTULO IV

RESULTADOS Y DISCUSIÓN

Dentro de la descomposición, se obtuvo en base a la ejecución del sistema web informático de gestión académica aplicado en la Institución Educativa Secundaria Industrial 32 (SIGAINDUSTRIAL32) de la ciudad de Puno, con el propósito de mejorar la gestión académica, información de los administrativos, docentes, estudiantes y padres de familia, mediante la automatización de cada uno de los procesos que son detallados en los módulos a implementar. Así evitar pérdida de información y problemas de documentos que son aglomerados en archivadores, siendo propenso a la mala manipulación de información.

En este capítulo se da a conocer de manera puntualizada los procesos donde se realiza del sistema, validación y comparación de resultados, teniendo un análisis e interpretación de las mismas.

4.1. DESARROLLO DEL SISTEMA

4.1.1 Fases del desarrollador

Definición de roles:

- **Programador:** En este proceso, el investigador asume el rol de programador, donde está encargado de hacer la programación y escribir el código fuente que es necesario para la implementación del sistema.
- **Cliente:** Institución educativa secundaria industrial 32 de la ciudad de Puno, en este caso asume el rol de cliente, quien define las especificaciones del sistema, incluyendo pruebas a las funcionalidades.



- **Tester:** Proceso donde también es asumido por el desarrollador, para que apoye al cliente en las preparaciones y las pruebas funcionales, así explicando los resultados al equipo
- **Tracker:** Acción donde se hace el seguimiento de acuerdo a la planificación, donde se evalúa el tiempo que se toma el equipo en realizar cada tarea, en este caso, el investigador cumple esta función, ayudando a determinar si el proyecto está dentro del tiempo de la iteración,
- **Coach:** Tarea cumplida por el investigador, con el fin de guiar y orientar que todas las normas sean aplicadas correctamente por la guía XP.

4.2. ARQUITECTURA MVC EN SIGAINDUSTRIAL32

Modelo

- El modelo en el sistema está encargado de gestionar los datos académicos y la lógica de negocio relacionada con la gestión académica. Esto incluiría información como registros de estudiantes, calificaciones, horarias de clase, listas de asistencia, etc.
- La capa de modelo proporciona métodos para acceder y modificar los datos de manera coherente. Por ejemplo, puedes tener funciones para agregar estudiantes ingresar calificaciones o generar reportes académicos.
- La lógica de esta capa debe asegurarse de que los datos sean válidos y consistentes.

Vista



- La vista se encarga de la presentación de la información académica a los usuarios, como en este caso se incluyen administradores, profesores, estudiantes y padres de familia.
- Cada tipo de usuario podría tener vistas específicas adaptadas a sus necesidades. Por ejemplo, los profesores podrían ver el registro de las calificaciones de sus clases, mientras que los padres podrían acceder a informes de desempeño de sus hijos.
- La vista debe ser intuitiva y permitir la interacción del usuario, como la búsqueda de información o la actualización de datos personales.

Controlador

- El controlador actuaría como el intermediario entre el modelo y la vista manejando las solicitudes y las acciones del usuario.
- Los controladores pueden ser diferentes para cada tipo de usuario o para cada función en el sistema. Por ejemplo, puedes tener un controlador para la gestión de calificaciones, otro para la inscripción de estudiantes etc.
- Los controladores validarán las solicitudes del usuario, interactuarán con el modelo para obtener o actualizar datos y finalmente actualizarán la vista para mostrar los resultados.

Esta estructura de MVC una separación clara de las responsabilidades y facilita el mantenimiento y la expansión del sistema a medida que se agregan nuevas características o se realizan cambios en los datos o la lógica de la administración.



4.3. MÓDULOS DEL SISTEMA

4.3.1. Módulos

Logín: El acceso al sistema de información web será mediante un usuario y una contraseña que son asignados y registrados por el ADMIN.

Modulo Administrador: El sistema de información web debe contar con un módulo de registro denominado administrador, para el cual se registrarán todos los administradores que darán uso al sistema de información. En este módulo se hará el registro de administradores, lo cual solicitará una serie de datos que serán necesarios para el acceso al sistema, entre los datos a llenar en los campos son:

- DNI
- Nombres
- Apellido paterno
- Apellido materno
- Teléfono/celular
- Correo electrónico
- Usuario
- Contraseña

Modulo docente: el sistema de información web debe tener en cuenta con el módulo de registro llamado DOCENTE, para poder inscribir a los docentes que darán uso al sistema de información, donde la función del docente es ingresar notas de los estudiantes. En el módulo de docente, el sistema solicitará al administrador la información del docente para que pueda tener acceso al sistema, entre los datos a llenar en los campos son:



- DNI
- Nombres
- Apellido paterno
- Apellido materno
- Teléfono/Celular
- Correo electrónico
- Usuario
- Contraseña

Módulo estudiante: el sistema web debe tener en cuenta con el padrón de registro llamado ESTUDIANTE, para hacer el registro de los estudiantes de nuevo ingreso y por modalidad de traslado. Este módulo permite al administrador solicitar datos del estudiante que son esenciales para la continuidad de gestión académica del mismo. Entre los campos y datos a llenar, son:

- DNI
- Nombres
- Apellido paterno
- Apellido materno
- Fecha de nacimiento
- Lugar de nacimiento
- Genero
- Dirección actual
- Numero celular
- Correo electrónico
- Copia de DNI



- Copia partida de nacimiento
- Copia certificada de estudios

Módulo Padres: El sistema web debe tener en cuenta el formulario de registro llamado PADRES, para poder inscribir a los padres y la asimilación de los mismos con los educandos registrados, para así terminar con el registro completo de los estudiantes. Este módulo del sistema solicitará al administrador los datos de los padres de cada estudiante registrado, enlazando así la información entre estudiantes y padres. Entre los datos a ingresar en los campos son:

- DNI (estudiante)
- Padre/madre
- Dni
- Nombres
- Apellido paterno
- Apellido materno
- Fecha de nacimiento
- Lugar de nacimiento
- Vive? Si/no
- Dirección actual
- Correo electrónico
- Grado de instrucción
- Ocupación
- Estado civil
- Numero de celular
- Dirección del centro de trabajo



- Copia de dni

Matriculas: El sistema de información web debe tener en cuenta con un módulo de registro llamado MATRICULA, para poder hacer la matrícula y registro de la misma de los estudiantes, donde el administrador encargado realiza las matriculas al inicio de cada año académico o a estudiantes que ingresar por modalidad de traslado. El sistema permite al administrador matricular a cada estudiante, ingresando datos que ya son previamente almacenados. Entre los datos a ingresar en los campos son:

- Dni (estudiante)
- Año académico
- Grado a ingresar
- Sección
- Especialidad

Ingreso de notas: El sistema de información web, debe tener en cuenta con un módulo de registro llamado INGRESO DE NOTAS, para poder realizar el ingreso de notas de los educados. El sistema concede al docente y al administrador ejercer el ingreso de notas de cada estudiante, de acuerdo al curso que lleva cada docente, entre los datos a ingresar para hacer el registro de notas son:

- Año académico
- Curso
- Trimestre
- Grado
- Sección



Donde al ingresar estos datos permitirá al docente o administrador visualizar la lista de estudiantes que llevan el curso seleccionado e ingresando en la tabla las notas promedio de cada trimestre.

Consulta y emisión de matrículas: El sistema debe consultar y emitir mediante el módulo de REPORTES las matriculas que fueron realizadas, así mismo para poder ver o emitir a ficha de matrícula del estudiante, será necesario ingresar el año académico y el código de matrícula del estudiante para visualizar la ficha de matrícula, dando la opción de poder imprimirla.

Consulta y emisión de libreta de notas: El sistema debe consultar y emitir libreta de notas mediante el módulo de REGISTROS de los diferentes módulos ya mencionados, la visualización de las notas de los estudiantes puede realizarse en cualquier momento, ya sea para una consulta general de todas las notas de un estudiante, adjuntado los promedios de los diferentes trimestres del año académico, con la posibilidad de poder imprimirla.

4.3.2 Historias de Usuario

Tabla 5

historia de usuario – acceso al sistema

Historia de usuario	
Numero : 1	Usuario: Administrador, Docente
Nombre de historia: inicio de sesión al sistema	
Prioridad: Alta (alta, media, baja)	Riesgo de desarrollo: media (alta, media, baja)
Iteración asignada: 1	
Programador responsable: Alexander Huaraya Canllahue	



Historia de usuario

Descripción:

- En el inicio e ingreso en el sistema, en este caso el administrador debe estar previamente registrado con una cuenta.
- Para esta historia hay dos tipos de usuario, administrativos y docentes
- los administrativos o docentes deben identificarse para ingresar al sistema, el ingreso de usuario y contraseña permite saber si el admin es realmente encargado de administrar el software

Observaciones:

- solo los que hace uso del sistema son registrados en el software podrán ingresar al sistema y sus funcionalidades.

Nota: inicio e ingreso al sistema

Tabla 6

historia de usuario – Creación de tipos de usuario

Historia de usuario	
Numero : 2	Usuario: Administrador
Nombre de historia: Composición de tipos de usuario	
Prioridad: Alta (alta, media, baja)	Riesgo de desarrollo: media (alta, media, baja)
Iteración asignada: 1	
Programador responsable: Alexander Huaraya Canllahue	
Descripción:	
<ul style="list-style-type: none"> • El software concede al admin crear o consignar dos diferentes tipos de usuario. • El que hace uso del sistema “administrador”, adquiere todos los premisos y condiciones para la verificación del sistema. • El usuario “docente”, consigue ingresar, modificar y consultar notas de los estudiantes. 	
Observaciones:	
<ul style="list-style-type: none"> • En esta historia el administrador asigna y da el consentimiento a cada navegante que ingrese al software. 	

Nota: Creación de tipos de usuario (módulos).



Tabla 7

historia de usuario – Gestión de usuarios

Historia de usuario	
Numero : 3	Usuario: Administrador
Nombre de historia: Administración de usuarios	
Prioridad: Alta (alta, media, baja)	Riesgo de desarrollo: media (alta, media, baja)
Iteración asignada: 1	
Programador responsable: Alexander Huaraya Canllahue	
Descripción: <ul style="list-style-type: none">• El admin es el encargado de administrar todas las funciones del software, así también puede realizar las actividades dentro del marco del consumidor final	
Observaciones: <ul style="list-style-type: none">• Solo el admin tiene la facultad de ingresar a cada una de las funciones del sistema informático.	

Nota: Administración de usuarios.

Tabla 8

historia de usuario – registro de administradores y docentes

Historia de usuario	
Numero : 4	Usuario: Administrador
Nombre de historia: registro de administradores	
Prioridad: Alta (alta, media, baja)	Riesgo de desarrollo: media (alta, media, baja)
Iteración asignada: 1	
Programador responsable: Alexander Huaraya Canllahue	
Descripción: <ul style="list-style-type: none">• El sistema permite al admin registrar a los administradores quienes manejarán el sistema.• El usuario “administrador”, adquiere con todos los premisos necesarios y condiciones para el control del sistema.• El usuario “docente”, consigue ingresar, modificar y consultar notas de los estudiantes.	



Historia de usuario

Observaciones:

- El manejo del sistema dependerá mucho del tipo de usuario quien esté manipulando el software, ya sea el administrador y el docente.
- En el registro de administradores o docentes, se registra los mismos datos para ambos casos, como: DNI, nombres, apellidos, número de celular, correo electrónico, usuario y contraseña, que serán necesarios para el ingreso al sistema.

Nota: Módulo de registro de padres y administradores

Tabla 9

historia de usuario – registro de estudiantes

Historia de usuario

Numero :5

Usuario: Administrador

Nombre de historia: registro de estudiantes

Prioridad: Alta

Riesgo de desarrollo: media

(alta, media, baja)

(alta, media, baja)

Iteración asignada: 1

Programador responsable: Alexander Huaraya Canllahue

Descripción:

- El software permite al admin inscribir a los estudiantes ya sea de nuevo ingreso o por matrícula.
 - El administrador en el módulo de registro de estudiantes, ingresa todos los datos solicitados, ya que en posteriores búsquedas o modificaciones sea más fácil la búsqueda.
 - El admin al terminar el registro puede guardar la información ingresada en el sistema.
 - Además de registrar estudiantes, también permitirá la visualización de los estudiantes registrados.
 - Por otra parte también el sistema podrá realizar modificaciones autorizadas.
-



Historia de usuario

Observaciones:

- En este módulo el registro de estudiantes, solo basta con registrar solo una vez, ya sea por nuevo ingreso o por traslado, lo que significa que, en un próximo registro, es decir en una matrícula, ya no será necesario ya que los datos estarán almacenados en el sistema.
- El registro de estudiantes se almacenan los siguientes datos: dni, nombres, apellidos, fecha de nacimiento, lugar de nacimiento, genero, dirección actual, numero celular, correo electrónico y por ultimo subir archivos como la copia de DNI, copia de partida de nacimiento y copia de certificado de estudios primarios.

Nota: Registro de estudiantes.

Tabla 10

historia de usuario – registro de padres

Historia de usuario

Numero :6

Usuario: Administrador

Nombre de historia: registro de padres de familia

Prioridad: Alta

Riesgo de desarrollo: media

(alta, media, baja)

(alta, media, baja)

Iteración asignada: 1

Programador responsable: Alexander Huaraya Canllahue

Descripción:

- El software permite al admin inscribir a los padres de los estudiantes previamente ya registrados.
 - El admin, en el módulo de registro de padres, transcribe la información necesaria solicitando el sistema, lo cual enlaza los datos de los educados inscritos con la información de los padres y así ser estar correlacionados para agilizar posteriores búsquedas.
 - El administrador al terminar el registro puede guardar los datos ingresados en el sistema.
-



Historia de usuario

Observaciones:

- En este módulo solo basta registrar una sola vez, ya sea por estudiantes de nuevo ingreso o por estuantes de traslado.
- En el registro de padres se almacenan los siguientes datos: parentesco, DNI, nombres, apellidos, fecha de nacimiento, lugar de nacimiento, vive (si/no), dirección actual, correo electrónico, grado de instrucción, ocupación, estado civil, numero de celular, dirección del centro de trabajo y por último se sube archivos como la copia de DNI.

Nota: Registro de estudiantes.

Tabla 11

historia de usuario – Matriculas

Historia de usuario

Numero :7

Usuario: Administrador

Nombre de historia: matricula de estudiantes

Prioridad: Alta

Riesgo de desarrollo: media

(alta, media, baja)

(alta, media, baja)

Iteración asignada: 1

Programador responsable: Alexander Huaraya Canllahue

Descripción:

- El software permite al encargado realizar matrículas de los estudiantes que ya se registraron previamente, ya sea de nuevo ingreso o por traslado.
 - El sistema permite al administrador visualizar los datos ya consignados, en este caso un estudiante de nuevo ingreso, será matriculado dependiendo a que sección será asignado y que especialidad llevar.
 - El sistema permite al administrador matricular a estudiantes regulares, en este caso el administrador solo asigna al estudiante en la misma sección y con la misma especialidad que llevó con anterioridad.
 - El sistema también permitirá el edición de las mismas, con la finalidad de resarcir las actividades académicas, teniendo el control de los matriculados y teniendo su información relevante, para luego ser solicitado por los administradores o los estudiantes.
-



Historia de usuario

Observaciones:

- En este módulo solo basta hacer un registro de la matrícula, como los datos ya están enlazados, solo se ingresa el dni del estudiante y automáticamente se obtendrá los datos del estudiante y solo es necesario ingresar a que año académico pertenece, grado a ingresar, sección y la especialidad que no sufre algún cambio.
- Al finalizar la matricula, el administrador podrá imprimir la ficha de matrícula donde se visualiza todos los datos pertinentes sobre la matrícula y del estudiante.

Nota: Matriculas de estudiantes.

Tabla 12

historia de usuario – ingreso de notas

Historia de usuario	
Numero : 8	Usuario: Docente
Nombre de historia: matricula de estudiantes	
Prioridad: Alta (alta, media, baja)	Riesgo de desarrollo: media (alta, media, baja)
Iteración asignada: 1	
Programador responsable:	
Descripción:	
<ul style="list-style-type: none"> • El software permite al docente ingresar notas por cada trimestre, por lo general solo se ingresa la nota promedio de los cursos que lleva el estudiante, al igual que el curso de especialidad. • El sistema permite al docente ingresar, consultar y modificar las notas de los alumnos. 	
Observaciones:	
<ul style="list-style-type: none"> • El sistema permite al docente primeramente para el llenado de notas de los educados, ingresa el año académico, curso, trimestre, grado y sección, mostrando así la tabla para ingresar las notas de los estudiantes según a que datos haya ingresado previamente. • Al término de cada año académico se tendrá una libreta de notas de los cursos que se llevaron durante el año, adjuntando las notas de cada trimestre y por último el promedio final del año académico, conjuntamente con la información del estudiante. 	

Nota: Modulo del Docente- ingreso de Notas.



Tabla 13

historia de usuario – Emisión de ficha de matrícula

Historia de usuario	
Numero : 9	Usuario: Administrador
Nombre de historia: emisión de ficha de matrícula	
Prioridad: Alta (alta, media, baja)	Riesgo de desarrollo: media (alta, media, baja)
Iteración asignada: 1	
Programador responsable: Alexander Huaraya Canllahue	
Descripción:	
<ul style="list-style-type: none">• El administrador podrá emitir las fichas de matrícula de los estudiantes cuando ya se haya concluido con la matrícula.	
Observaciones:	
<ul style="list-style-type: none">• El sistema permite imprimir la ficha de matrícula del estudiante, donde se aprecia los cursos, año académico, grado, sección, especialidad y datos relevantes del estudiante.	

Nota: Emisión de Fichas de matrículas.

Tabla 14

historia de usuario – emisión de libreta de notas

Historia de usuario	
Numero : 10	Usuario: Administrador
Nombre de historia: emisión de libreta de notas	
Prioridad: Alta (alta, media, baja)	Riesgo de desarrollo: media (alta, media, baja)
Iteración asignada: 1	
Programador responsable: Alexander Huaraya Canllahue	
Descripción:	
<ul style="list-style-type: none">• El administrador emite libreta de notas de los estudiantes, ya sea por trimestre o por la finalización del año académico.• A demás de emitir, el sistema podrá realizar consultas contantemente, lo que nos permite tener un mejor control de los estudiantes que ya egresan.	

Historia de usuario

Observaciones:

- El sistema admite imprimir una libreta de notas de los educados, donde se encuentra de forma detallada las datos de los estudiantes y los promedios de los cursos que lleva.

Nota: Emisión de libreta de notas.

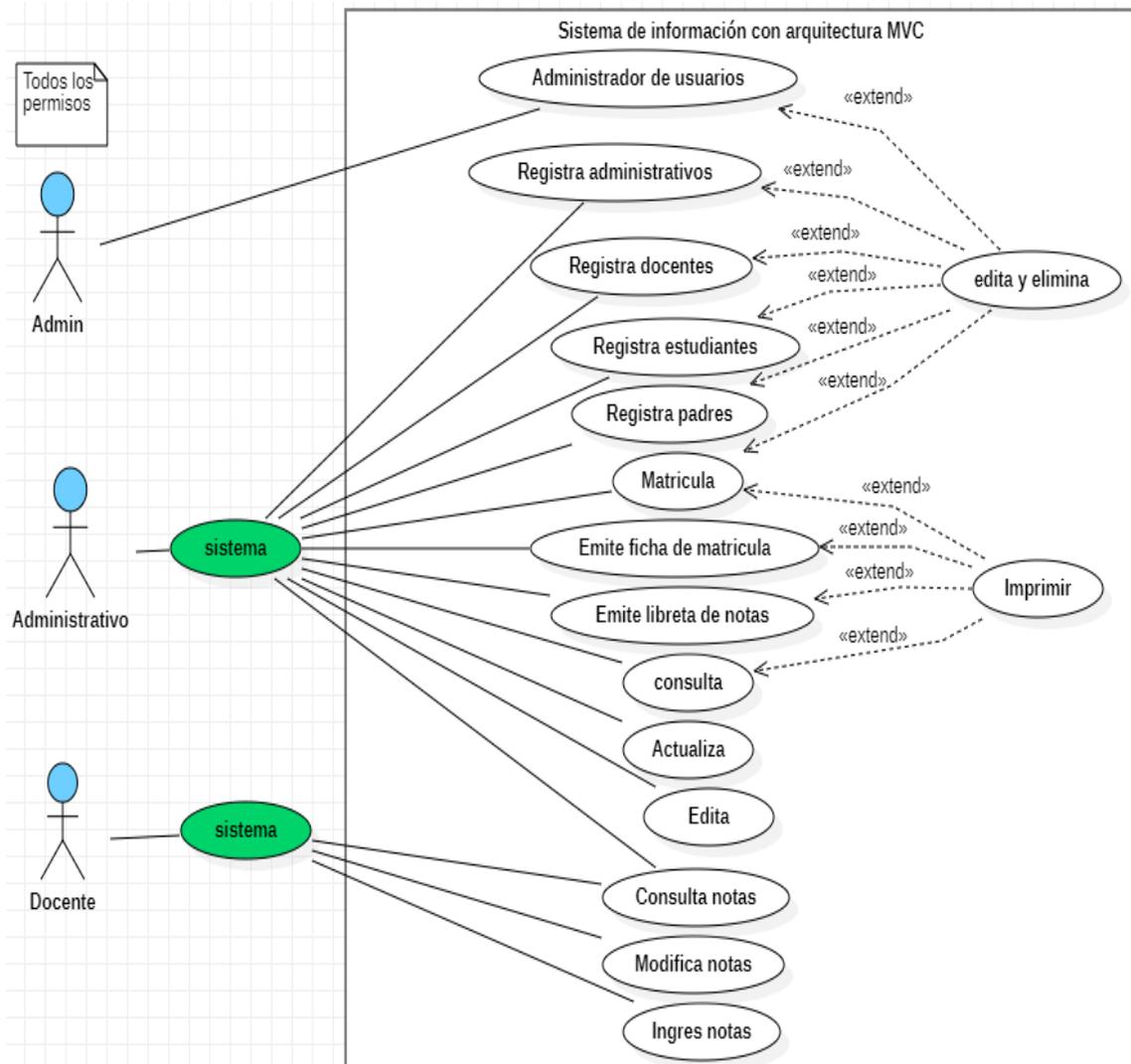
4.4. DIAGRAMA Y MODELO DE BASE DE DATOS

Diagrama de casos de uso

se aprecia el esquema de casos de uso, donde se visualiza todos los componentes y funcionalidades que los usuarios del sistema realizan.

Figura 7

Diagrama de caso de uso





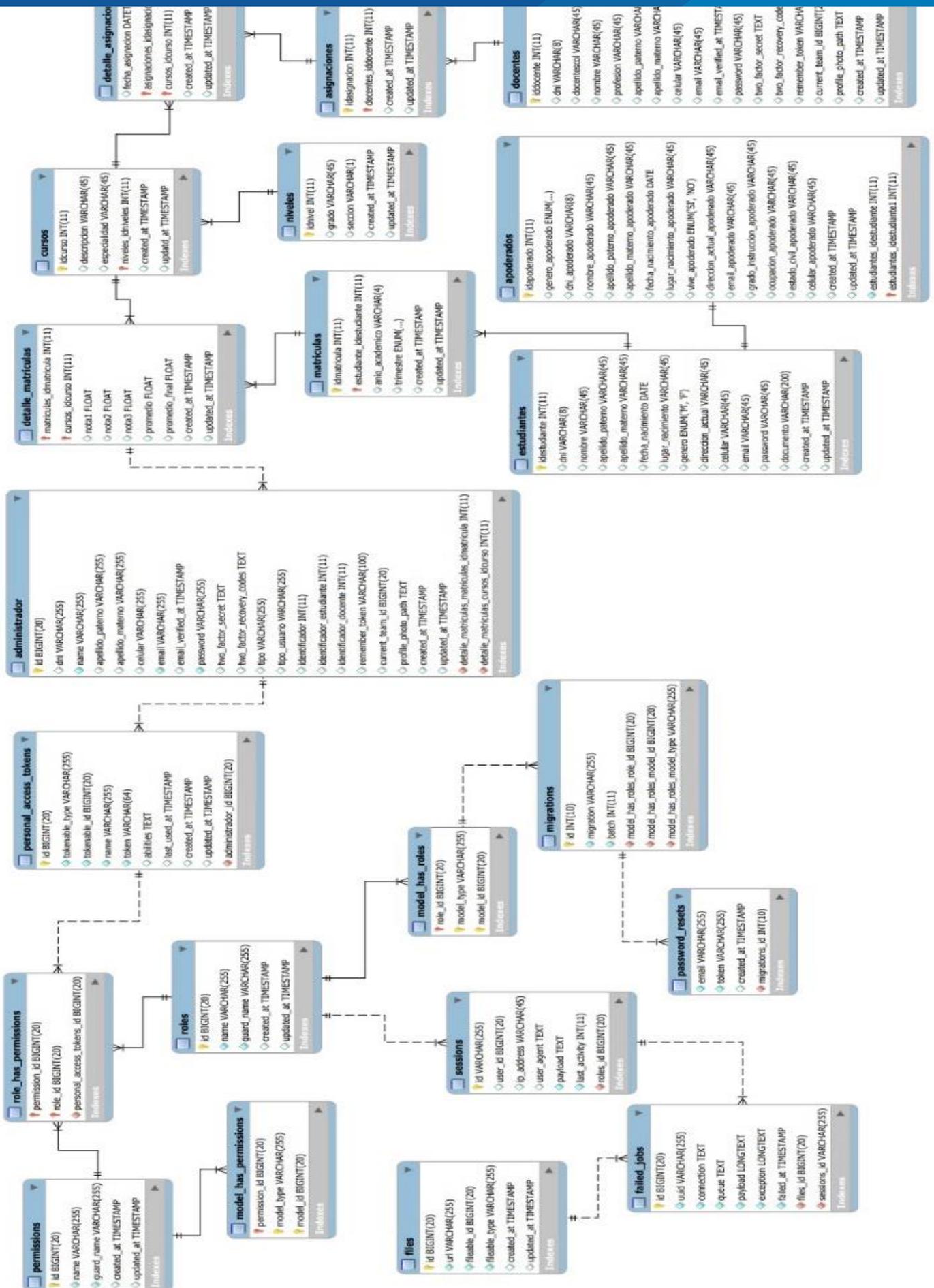
Fuente: Elaboración propia (StarUML)

Diagrama de diseño de base de datos

El diseño de la base de datos está desarrollado de acuerdo al diseño del sistema para su buen funcionamiento, recabando toda la información del módulo y los formularios ya mencionado anteriormente.

Figura 8

Diagrama de diseño de base de datos





Nota: Diseño de la base de datos (MySQL Workbench).

4.5. DISEÑO E IMPLEMENTACIÓN DEL SISTEMA

4.5.1. Implementación del sistema informático SIGAINDUSTRIAL32

La implementación de SIGAINDUSTRIAL32, procede con una continua interacción entre el grupo de codificación y el usuario final, lo que hace es obtener de manera más precisa las funciones que se requieran para que el sistema tenga aceptación, comodidad y facilidad de uso. Además de establecer tiempos para otorgar del sistema, teniendo en consideración las prioridades de cada uno de los módulos desarrollados y así estimar el tiempo que se toma para desarrollar cada una de las historias de los usuarios.

En cuando a las funciones de SIGAINDUSTRIAL32 basándose en la información recabada de los usuarios, indicaron las siguientes funciones: que cuente con un acceso de usuarios, así como también un registro de usuarios, administrativos, docentes, educados y padres, matricular, emitir ficha de matrícula y de notas, consultar, actualizar, editar, subir notas, modificar notas, consular notas e impresión de las mismas.

Iteracion1: Una de las cualidades para el buen funcionamiento del sistema web, es que puede ser compatible con diferentes navegadores como google Chrome, Firefox y Microsoft Edge, ya que el sistema está desarrollado bajo el lenguaje de programación PHP 8, lo que lo hace estable y compatible en diferentes plataformas, dando lugar a un mejor entendimiento y con mayor soporte en la base de datos.

Figura 9

Interfaz de inicio de sesión al sistema web

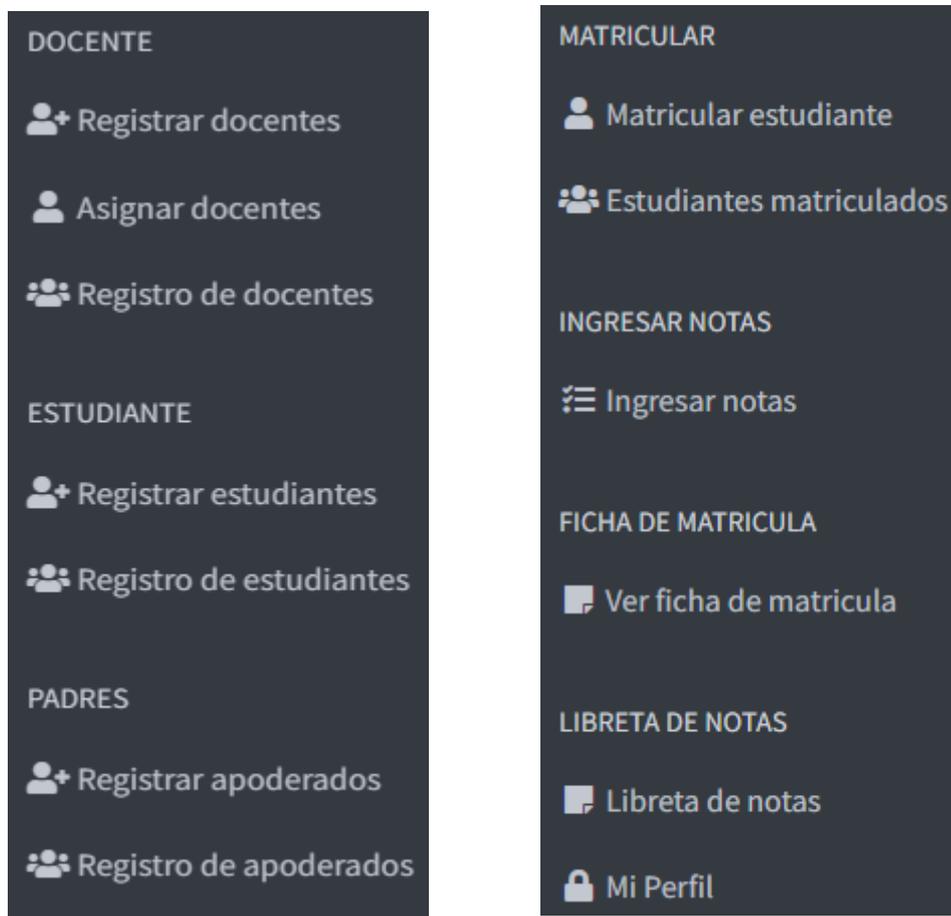


Nota: Portada e inicio de sesión al sistema.

Iteracion2: En esta segunda etapa se diseña todos los módulos que son imprescindible para que el sistema tenga un adecuado funcionamiento, como: USUARIOS, DOCENTE, ESTUDIANTE, PADRES, MATRICULAR, INGRESO DE NOTAS, FICHA DE MATRICULA, LIBRETA DE NOTAS Y REPORTES. Estos módulos contienen todos formularios necesarios para el llenado de las mismas.

Figura 10

Módulos principales del sistema



Nota: Módulos del sistema que el administrador podrá manejar.

Formulario de usuarios: el registro de los usuarios quienes registraran el sistema teniendo todos los permisos que se requieran para la buena administración del sistema.

Figura 11

Formulario de registro de usuarios quienes administraran el sistema

REGISTRO DE ADMINISTRADOR

DNI	<input type="text"/>	Apellido materno	<input type="text"/>
Nombre	<input type="text"/>	Email	<input type="text"/>
Apellido paterno	<input type="text"/>		
Celular	<input type="text"/>		
Usuario	<input type="text"/>		
Password	<input type="text"/>		
Repita su Password	<input type="text"/>		

Nota: Modulo para el registro de administradores.

Formulario de docentes: se diseña el formulario perteneciente al módulo de DOCENTES, creando campos necesarios para el registro como se visualiza en la figura.

Figura 12

Formulario de docentes que harán uso del sistema, específicamente en el ingreso de notas.

REGISTRO DE DOCENTES

DNI	<input type="text"/>	Apellido materno	<input type="text"/>
Nombre	<input type="text"/>	Celular	<input type="text"/>
Profesion	<input type="text"/>	Email	<input type="text"/>
Apellido paterno	<input type="text"/>	Password	<input type="text"/>
		Enviar	<input type="button" value="Guardar datos"/>

Nota: Modulo para el registro de docentes.

Formulario de estudiantes: el desarrollo del módulo ESTUDIANTES, solicita llenar el formulario para tener información relevante de los estudiantes de nuevo ingreso o por traslado, considerando campos necesarios para su registro antes de la matricula como se visualiza en la figura:

Figura 13

Formulario de registro de nuevos estudiantes.

REGISTRAR ESTUDIANTE

DNI	<input type="text"/>	Genero	<input type="text" value="Masculino"/>
Nombre	<input type="text"/>	Direccion	<input type="text"/>
Apellido paterno	<input type="text"/>	Celular	<input type="text"/>
Apellido materno	<input type="text"/>	Email	<input type="text"/>
Fecha de nacimiento	<input type="text" value="dd/mm/aaaa"/>	Password	<input type="text"/>
Lugar de nacimiento	<input type="text"/>	Documento para publicacion	<input type="button" value="Elegir archivo"/> No se eligió ningún archivo

Unir los 3 PDF y subir un unico archivo:

- copia Dni
- copia Partida Nacimiento
- copia Certificado de estudios (primaria)

Nota: Modulo para el registro de estudiantes nuevos.

Formulario de padres: el formulario de registro de padres, solicita registrar a los padres de familia una vez que el estudiante se haya registrado, entonces se podrá relacionarlos padres con los del estudiante que ya fue registrado con anterioridad, teniendo así la ventaja y al momento de hacer consultas, optimizando el proceso considerablemente.

Figura 14 *Formulario de registro de padres de familia, donde se podrá enlazar la información del estudiante y del padre.*

REGISTRAR PADRES

DNI estudiante

No se tiene registro del estudiante en la BASE DATOS...

Datos de apoderado

Genero	<input type="text" value="Padre"/>	DNI	<input type="text"/>
Nombre	<input type="text"/>	Apellido paterno	<input type="text"/>
Apellido materno	<input type="text"/>	Fecha de nacimiento	<input type="text" value="dd/mm/aaaa"/>
Lugar de nacimiento	<input type="text"/>	Vive	<input type="text" value="Si"/>
Grado de instruccion	<input type="text"/>	Estado civil	<input type="text"/>
Email	<input type="text"/>	Celular	<input type="text"/>
Direccion	<input type="text"/>	Ocupacion	<input type="text"/>
		Copia de DNI	<input type="button" value="Elegir archivo"/> No se eligió ningún archivo

Subir copia de DNI en PDF:

Nota: Modulo para el registro de padre, tiene relación con los estudiantes.

Figura 15

Formulario de matrículas para estudiantes ya registrados previamente por el sistema

MATRICULAR A ESTUDIANTE

DNI	<input type="text" value="74832679"/>	Seccion	<input type="text" value="C"/>
Nombre	<input type="text" value="SARAI LEYDI"/>	Grado	<input type="text" value="QUINTO"/>
Apellido paterno	<input type="text" value="TICONA"/>	Año académico	<input type="text" value="2024"/>
Apellido materno	<input type="text" value="TAIPE"/>	Especialidad	<input type="text" value="COMPUTACIÓN"/>
		Enviar	<input type="button" value="Guardar datos"/>

Nota: Modulo para la matrícula de estudiantes registrados.

Formulario de matrículas: Por último, el módulo de matrículas, nos ayuda a matricular a los estudiantes inscritos con anterioridad, teniendo así una mejor visión de la información de los estudiantes, no solo en matrículas, sino también en el control de la información almacenada.

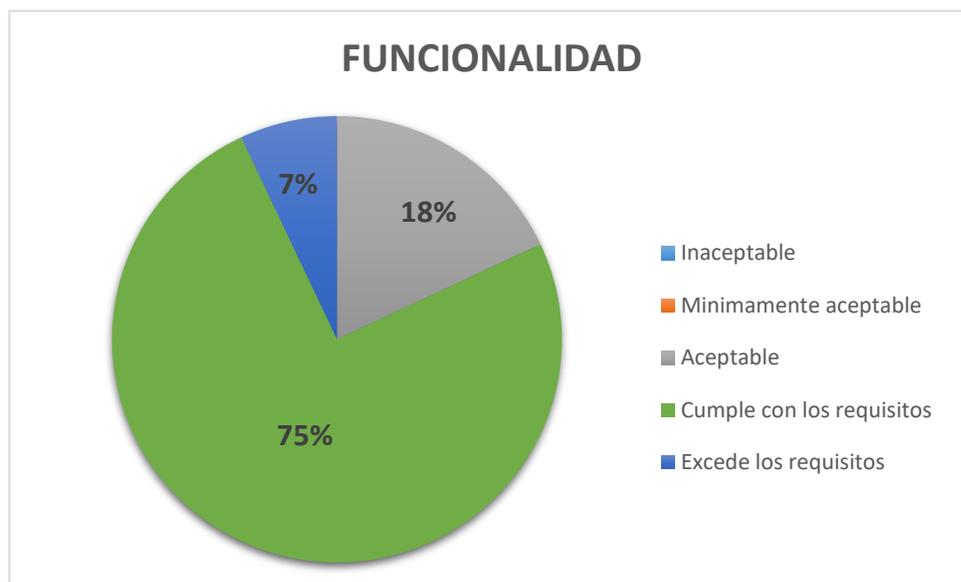
4.6. VALIDACIÓN DEL SISTEMA

4.6.1. Evaluación de calidad de software según la ficha ISO – 9126

La implementación del sistema de información SIGAINDUSTRIAL32 se ejecutó con el propósito esencial de dar mejor manejo a la gestión académica y para examinar el nivel de buena calidad del software se ejecutó las métricas de ISO – 9126, llenando la ficha donde se examina con el alto valor y la medición de calidad de software con sus diferentes características que conlleva las métricas.

Figura 16

Resultados sobre la encuesta de calidad de software bajo los estándares ISO-9126 Funcionalidad.

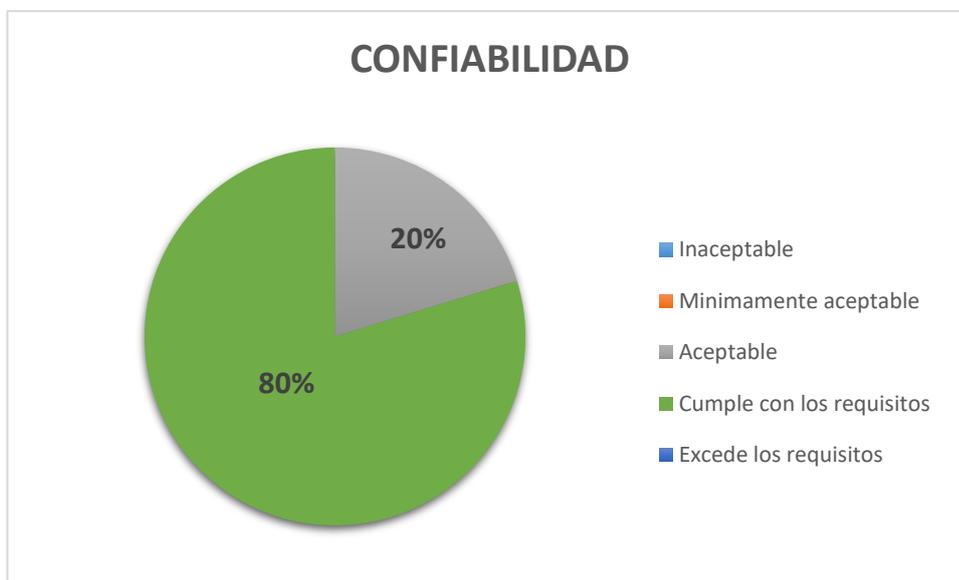


Nota: Resultados sobre la ficha de ISO-9126 calidad de software, funcionalidad.

Interpretación: En la figura se visualiza que del 100% de usuarios, el 7% menciona que el sistema excede los requisitos de funcionalidad, un 18% indica que la funcionalidad del sistema es aceptable y mientras que un 75% indica que cumple con los requisitos de funcionalidad.

Figura 17

Resultados sobre la encuesta de calidad de software bajo los estándares ISO-9126 confiabilidad.



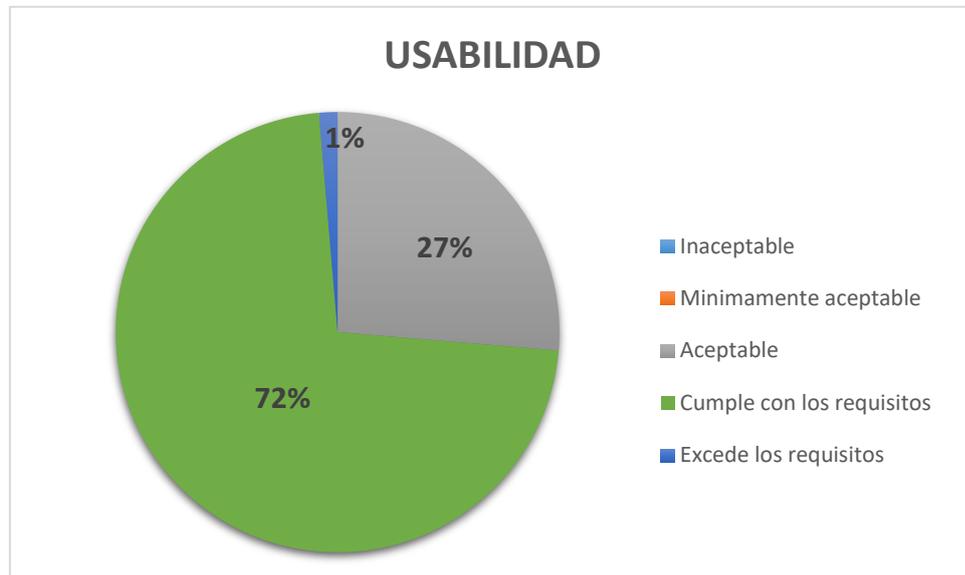
Nota: Resultados sobre la ficha de ISO-9126 calidad de software, Confiabilidad.

Interpretación: En la figura se visualiza que del 100% de usuarios, el 20% menciona que el sistema es aceptable con la confiabilidad, en cuanto al 80% indica que cumple con los requisitos de confiabilidad.

Figura 18

Resultados sobre la encuesta de calidad de software bajo los estándares ISO-9126

Usabilidad



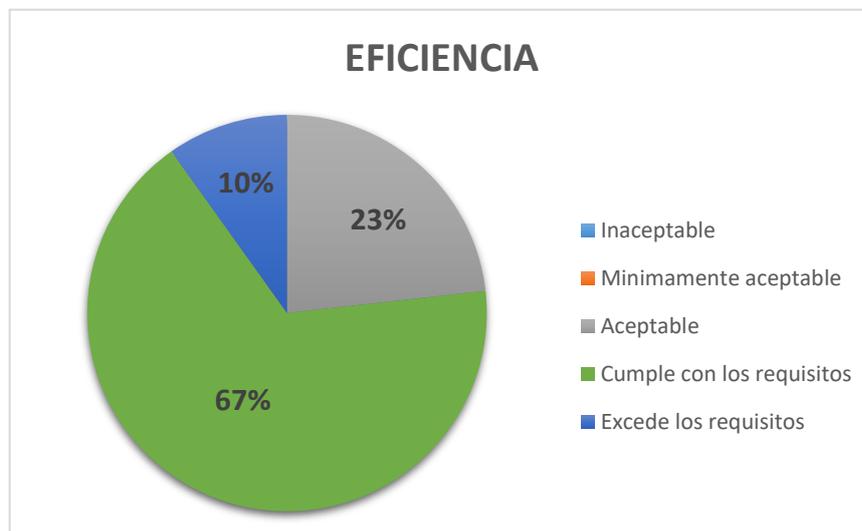
Nota: Resultados sobre la ficha de ISO-9126 calidad de software, Usabilidad.

Interpretación: En la figura se visualiza que del 100% de usuarios, el 1% menciona que el sistema excede los requisitos de usabilidad, seguidamente con un 27% de usuarios indican que la usabilidad del sistema es aceptable y por ultimo con un 72% de usuarios indica que cumple con los requisitos de usabilidad.

Figura 19

Resultados sobre la encuesta de calidad de software bajo los estándares ISO-9126

Eficiencia.



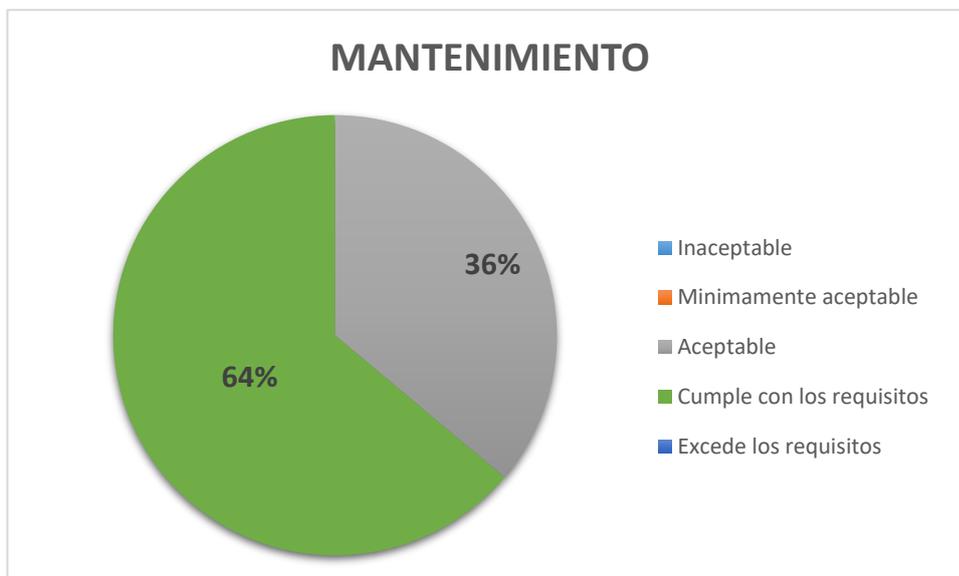
Nota: Resultados sobre la ficha de ISO-9126 calidad de software, Eficiencia.

Interpretación: En la figura se visualiza que del 100% de usuarios, el 10% menciona que el sistema excede los requisitos de eficiencia, mientras que un 23% de usuarios indican que la eficiencia del sistema es aceptable y por ultimo con un 67% indica que cumple con los requisitos de eficiencia en el sistema.

Figura 20

Resultados sobre la encuesta de calidad de software bajo los estándares ISO-9126

Mantenimiento

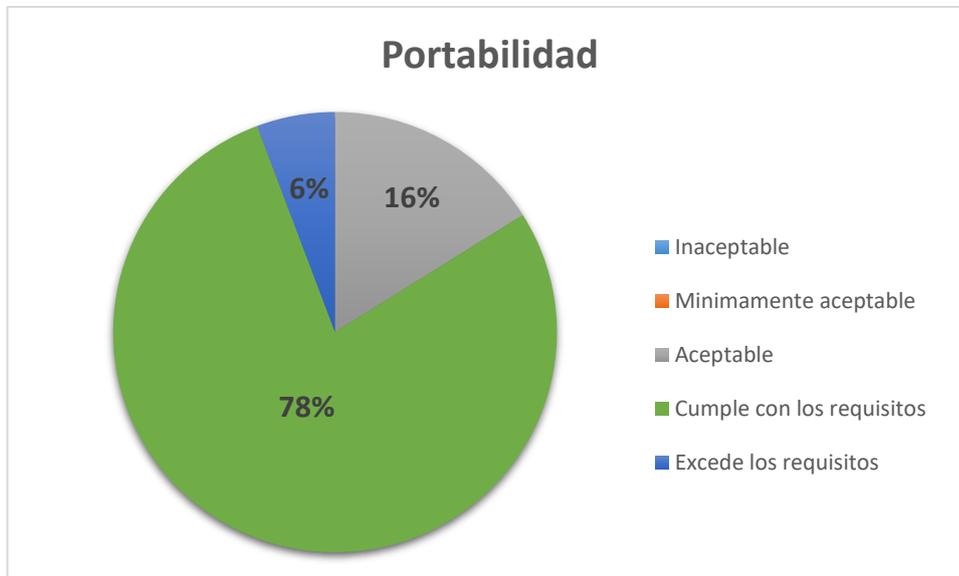


Nota: Resultados sobre la ficha de ISO-9126 calidad de software, Mantenimiento.

Interpretación: En la figura se visualiza que del 100% de usuarios, el 36% menciona que el sistema es aceptable para poder realizar cambios, en cuando al 64% de usuarios restante, indican que cumple con los requisitos de mantenimiento del sistema.

Figura 21

*Resultados sobre la encuesta de calidad de software bajo los estándares ISO-9126
Portabilidad.*



Nota: Resultados sobre la ficha de ISO-9126 calidad de software, Portabilidad.

Interpretación: En la figura se visualiza que del 100% de usuarios, el 6% menciona que el sistema excede los requisitos de portabilidad, mientras que un 16% de usuarios indican que la portabilidad del sistema es aceptable y por último con un 78% de usuarios, indican que cumple con los requisitos de portabilidad del sistema, adaptándose a diferentes navegadores, entornos, ambientes y sistemas operativos.

Validación del sistema:

Evaluación del sistema informático: Para la examinar y evaluar de calidad del sistema, se comprobó haciendo el llenado de la ficha de evaluación con apreciación y parámetros de medición de calidad de software estándar ISO 9126.

Tabla 15

valoración del ISO 9126

Indicadores	Valor
Muy malo	1
Malo	2
Regular	3
Bueno	4
excelente	5

Fuente: Métricas del ISO 9126

Tabla 16

Medición de calidad de software ISO 9126

Indicadores	Intervalo	Decisión
Inaceptable	[20-38>	
Mínimamente aceptable	[38-56>	
Aceptable	[56-74>	
Cumple los requisitos	[74-92>	88
Excede los requisitos	[92-110>	

Fuente: métricas ISO 9126

Decisión: Teniendo en cuenta el resultado final que se obtuvieron bajo las métricas de la ISO 9126 el promedio de puntuación es de 88 puntos, donde nos indica que el sistema web de gestión académica en la institución educativa secundaria industrial 32 de Puno, cumple los requisitos según el estándar ISO 9126.

4.7. CONTRASTACION DE HIPOTESIS

Los datos recolectados provienen de la muestra obtenida con anterioridad, donde el porcentaje total de la población para la implementación del sistema es del 100%. Ahora recordemos que para el análisis estadístico se aplicó el método de diferencia de medias para muestras apareadas, lo cual obtuvimos con anterioridad. Para la contratación de hipótesis se aplicó el instrumento de evaluación (ANEXO B) donde se obtiene datos del antes y después de la implementación del sistema que se muestra a continuación.

Tabla 17

Resultados del instrumento de evaluación (ANEXO B) del antes y después, aplicado a número de muestra.

Puntaje Antes			Puntaje Después		
A	B	C	A	B	C
10	10	10	18	20	18
10	10	12	18	20	20
12	10	8	18	20	16
10	8	8	20	20	16
10	10	10	20	20	18
12	10	10	20	18	16
8	10	8	16	18	18
10	12	8	16	18	18
10	12	8	20	18	18
10	12	10	16	18	18
10	10	10	16	18	18
12	10	10	20	16	16
10	10	12	20	20	16
10	10	12	18	18	16
10	10	10	18	18	16
10	12	10	18	16	18
10	10		18	16	
8	10		20	16	
8	10		20	18	
8	12		18	18	

Nota: Datos obtenidos del instrumento de evaluación.

Tabla 18

Medición de aceptación o rechazo con la encuesta aplicada (INSTRUMENTO DE EVALUACION)

Indicadores	Intervalo	Decisión
Acepta	2 puntos (SI)	Antes (10)
Rechaza	0 puntos (NO)	Después (18)

Nota: Indicadores de evaluación del instrumento.

En la tabla 17 se puede observar en primera instancia, los indicadores, el intervalo y la decisión final respecto a la encuesta realizada. En un primer punto solo se cuenta con dos indicadores (Acepta y Rechaza) los cuales están con una puntuación de 2 y 0 puntos relativamente. Tomando en cuenta el promedio de las puntuaciones de la encuesta que se aplicó Antes y Después de la implementación del sistema, se tiene lo siguiente; antes de la implementación del sistema se obtuvo una nota de 10 puntos en promedio, lo que indica que la administración de la información en la institución es desaprobada o rechazada, seguidamente se vuelve a tomar la encuesta, esta vez con la implementación del sistema, obteniendo una nota de 18 puntos en promedio, indicando que la nueva administración de la información en la institución es aprobada o aceptada por los administradores y docentes de la institución.

4.7.1. Planteamiento de hipótesis

H₀: $\mu_A = \mu_B$

H₀: La implementación del sistema información para la gestión académica no mejora significativamente el manejo y almacenamiento de información de la institución educativa secundaria industrial 32 de Puno.

H₁: $\mu_A \neq \mu_B$



H1: La implementación del sistema informático de gestión académica mejora significativamente el manejo y almacenamiento de información de la institución educativa secundaria industrial 32 de Puno.

4.7.2. Fijación de nivel de significancia (α)

El nivel de significancia $\alpha = 0.05$ (95% de nivel de confianza)

4.7.3. Regla de decisión utilizado

si $p < \alpha$ se rechaza la H_0 y se acepta la H_1

4.7.4. Prueba estadística

El análisis estadístico de diferencia de medias utilizando la prueba Z-Test en RStudio, se lleva a cabo la investigación de diferencia de medias entre dos grupos, en este caso el pre-test y post-test. Esta prueba se utilizó con el propósito de evaluar si existe una diferencia estadísticamente significativa entre los grupos y respaldar las hipótesis planteadas en la investigación. Se incluyen los datos relevantes que se utilizaron en la prueba, que consiste en aceptación o rechazo de la gestión académica antes y después de implementar el sistema informático. La prueba Z-Test se configuró con un nivel de significancia del 0.05 y el tamaño de la muestra para cada grupo fue de 56. El análisis estadístico se realizó utilizando la librería BSDA que proporciona el programa RStudio, seguidamente este análisis en el software ingresando los datos finales ya procesados nos proporciona resultados donde podemos interpretar y a corroborar si la hipótesis alterna es rechazada o aceptada.



4.7.5. Decisión

Los resultados obtenidos con la prueba Z-test aplicado en el software R Studio, se tiene una probabilidad de $p - \text{value} = 0.000$, puesto que la probabilidad es menor que $\alpha = 0.05$, por lo que se decide rechazar la hipótesis nula y aceptar la hipótesis alterna, entonces, la implementación del sistema informático de gestión académica mejora significativamente el manejo y almacenamiento de información de la institución educativa secundaria industrial 32 de Puno.

En conclusión, el análisis estadístico utilizando la prueba Z test en RStudio, respalda la hipótesis alterna de la implementación del sistema informático de gestión académica mejora significativamente el manejo y almacenamiento de la información de la Institución Educativa Secundaria Industrial 32 de Puno. Estos hallazgos son relevantes para la investigación, ya que nos ayuda a la toma de decisiones.

4.8. DISCUSIÓN

En base al resultado obtenido, se puede afirmar que mediante el uso del estándar ISO 2196, se consolida que las características como: funcionalidad, confiabilidad, usabilidad eficiencia, mantenimiento y portabilidad, hacen que el sistema obtuvo un puntaje aprobatorio validado por los usuarios después de utilizar el sistema web, así también como la implementación del sistema SIGAINDUSTRIAL32 tiene una mejora significativa avalado por los resultados de la contratación de hipótesis, lo que hace que el sistema sea eficiente, teniendo un grado de aprobación por parte de los usuarios finales. De la misma manera (Torres, 2019), afirma en su trabajo que el desarrollo e implementación de su software basada en arquitectura MVC, obtuvo resultados favorables según el estándar ISO 9126, solucionando y sistematizando gran parte de



control de insumos y medicamentos en el sistema SISMED . Por otro lado (Torres, 2018), logra desarrollar e implementar la aplicación web, cumpliendo con sus metas satisfactoriamente, además, los usuarios conllevan la aceptación con la funcionalidad de la aplicación web y utilizando también las métricas ISO 9126 ya mencionadas, lo que hace que el sistema cumpla con los requisitos funcionales de la oficina de repositorio institucional. En otro entorno (Herrera, 2016) indica que la implementación de la aplicación web ha permitido que los usuarios puedan acceder a distintos procesos que se realiza en secretaria, afirmando que se redujo el tiempo de atención por parte de los administrativos.



V. CONCLUSIONES

- Se logró implementar satisfactoriamente el sistema informático para la gestión académica aplicado en la Institución Educativa Secundaria Industrial 32 de la ciudad de Puno, lo cual mejoró el manejo de información de los docentes, estudiantes de familia, que son administrados por los trabajadores que están a cargo, permitiendo tener los datos detallados de los estudiantes y docentes, cumpliendo con el objetivo general.
- Se desarrolló los módulos que tienen relación a la información de gestión académica como usuarios, docente, estudiante, padres, matriculas, ingreso de notas, ficha de matrícula, libreta de notas y reportes, siguiendo la metodología XP y aplicando los ciclos que corresponden, generando así el buen entendimiento y adaptación del sistema de manera cómoda para los usuarios.
- Se desarrolló la base de datos de manera amplia, logrando así una buena administración y resguardo de información recabada por los administrativos encargados, teniendo en cuenta un posible incremento de la base de datos, así como también del mismo sistema ya que está desarrollado bajo la metodología XP, lo cual permite hacer modificaciones para poder ampliar en varios aspectos el sistema.
- Para la evaluación del software se aplicó las métricas del estándar internacional ISO 9126, en la que se pudo evaluar la función, lo confiable, el uso, la eficiencia, mantenimiento y portabilidad del sistema, se pudo obtener un puntaje de 88, que indica que el sistema cumple con las exigencias dadas por los administrativos y docentes de la institución educativa secundaria industrial 32 de Puno.



VI. RECOMENDACIONES

- Se hace recomendación a los encargados del uso del sistema tener en consideración que la manipulación de la información del sistema es estrictamente confidencial, ya que se puede hacer mal uso, como eliminar o modificar esta relevante información de suma importancia para la Institución.
- A futuras investigaciones, que pueden tomar como base este proyecto para poder modificar o ampliar las características del sistema, complementando nuevas tecnologías de información.
- Se recomienda a instituciones públicas de similar administración académica, implementar el sistema SIGAINDUSTRIAL32 como referencia para la gestión de información dentro de las instituciones.
- Sistematizar diferentes procesos administrativos de diferentes áreas, agilizando el proceso y además resguardar la información de manera segura.



VII. REFERENCIAS

- Abud Figueroa María. (2012). Calidad en la Industria del Software . La Norma ISO-9126. *Calidad En La Industria Del Software. La Norma ISO-9126*, 255. <http://www.monografias.com/trabajos5/%0>
- Arias-Gómez, J., Villasís-Keever, M. Á., & Miranda-Novales, M. G. (2016). The research protocol III. Study population. *Revista Alergia Mexico*, 63(2), 201–206. <https://doi.org/10.29262/ram.v63i2.181>
- Bahit, E. (2012). *Programador PHP*. [http://46.101.4.154/Libros/El lenguaje PHP.pdf](http://46.101.4.154/Libros/El_lenguaje_PHP.pdf)
- Blancarte, J. (2020). *Introducción a la arquitectura de software – Un enfoque práctico*. 1–653.
- Caso, P. (2018). *innvoacion y desarrollo iso 9126*. http://iso9126uts.blogspot.com/p/i_8.html
- Castell, D. L. (2017). *3.2 Metodología experimental. Figura 1*, 1–15.
- Chachaque Loma, L. (2015). *Sistema Web Para La Administración Académica Del Instituto Superior De Educación Pública “Divino Jesús” Chumbivilcas 2014*. 1–142.
- Delgado, C. A. A. (2017). Fundamentos de seguridad informática. In *Areandina* (Issue 2).
- Domínguez Zárate, R. F. (2016). Aplicación de métricas de calidad en uso utilizando la ISO 9126 para determinar el grado de satisfacción del sistema único de matrícula. *Software Quality Attributes and Trade-Offs*, 447(May), 1–33.
- Edición, P., José Ávila Herrera, H., Álvaro Quiroga León Edición, J., & Diskcopy SAC Jr San Agustín Nro, E. (2013). *Directiva De Seguridad-APDP*. 44.
- Effy, O. (2008). *Administración de los sistemas de información*.
- Espinoza, M. A. A. (2013). Manual Para Elegir Una Metodología De Desarrollo De Software Dentro De Un Proyecto Informático. *Universidad de Piura*, 0(0), 115. <https://pirhua.udep.edu.pe/handle/11042/2747>



- Espinoza, M. J. (2018). Las TICS como factor clave en la gestión académica y administrativa de la universidad. *Gestión En El Tercer Milenio*, 20(39), 35–44. <https://revistasinvestigacion.unmsm.edu.pe/index.php/administrativas/article/view/14141/12473>
- F. Kerlinger. (2008). Técnicas E Instrumentos De RECOLECCION DE Datos. *Metodologia de La Investigacion*, 2, 201–247. <http://iyanu.blogspot.es/i2008-07/>
- Francisco, E. (2018). *Modelo de Clases*. 1–5. https://www.emagister.com/uploads_user_home/Comunidad_Emagister_5401_Modelado_de_Clases.pdf
- Gauchat, J. D. (2017). *El gran libro de HTML, CSS3 Y JavaScript*.
- Gil Vera, V. D., Gomes Da Silva, C. R., Gil Vera, J. C., & Teutsch, J. (2018). Frameworks para el desarrollo de prototipos WEB: Un caso de aplicación. *Lámpsakos*, 20, 40–53. <https://doi.org/10.21501/21454086.2065>
- Gilfillan, I. (2003). *La Biblia de MySQL / MySQL 4*. 880. <http://books.google.com/books?id=EBWyPQAACAAJ&pgis=1>
- González Longatt, F. M. (2007). Introducción a los Sistemas de Información: Fundamentos. *Sistemas de Información*, 1, 7.
- Gutiérrez, J. J. (2014). ¿Qué es un framework? Available in: Http://Www.Lsi.Us.Es/~Javierj/Investigacion_ficheros/Framework.Pdf Accessed May, 12, 1–4. http://www.lsi.us.es/~javierj/investigacion_ficheros/Framework.pdf
- Hernán, S. M. (2004). *Diseño de una Metodología Ágil de Desarrollo de Software*.
- Hernández, R., & Mendoza, C. P. (2018). Metodología de la investigación. In *Mc Graw Hill* (Vol. 1, Issue Mexico).
- Herrera Urtiaga, P. (2016). Sistema de información para el Instituto de Informática de la Universidad Nacional del Altiplano Puno - 2016. *Tesis*, 1–13. <http://repositorio.unap.edu.pe/handle/20.500.14082/2990>
- International, T., Alliance, M., Fishing, C., & Pilot, R. (1998). in F O. *North*, 3–7.
- Kendall, K., & Kendall, J. (2011). *Análisis y Diseños de Sistemas 8va Edición - ISBN:*



9786073205771. http://cotana.informatica.edu.bo/downloads/Id-Analisis_y_Disenio_de_Sistemas_Kendall-8va.pdf
- Maida, E., & Pacienza, J. (2015). Metodologías de desarrollo de software. *Biblioteca Digital de La Universidad Católica Argentina*, 117. <http://bibliotecadigital.uca.edu.ar/repositorio/tesis/metodologias-desarrollo-software.pdf>
- Mannino, M. V. (2010). *Administracion de base de datos diseño y desarrollo de aplicaciones*.
- Maria, B. R., Gaitan, E., Neldin, B. R., Pérez, N., & Documento, C. D. E. L. (2016). Metodología Agil de Desarrollo de Software. *Desarrollo de Software*, 1–146. <https://repositorio.unan.edu.ni/1365/1/62161.pdf>
- Mehta, C., Bhavsar, A., Oza, H., & Shah, S. (2018). *MySQL 8 Administrator 's Guide*.
- Merino Covarrubias, A., & Miranda Pacheco, J. (2018). Sistema De Matrícula Y Consulta De Notas Para La Universidad Peruana Austral Del Cusco (Smcn-Upac). *Tesis*, 232. <http://repositorio.uaustral.edu.pe/handle/UAUSTRAL/11>
- Minedu. (2015). Gestión Escolar. *Compromisos de Gestión Escolar*, 1, 1–34.
- MINEDU. (2014). Marco de buen desempeño del directivo. *Minedu*, 56. http://www.minedu.gob.pe/DeInteres/xtras/marco_buen_desempeno_directivo.pdf
- Miquel, J. P., & Martos, J. R. (2006). Introducción a la Ingeniería del Software. *Chaos*, 1–10.
- Navajas, A. (2012). *CursoCSS3.pdf* (p. 63). <https://openlibra.com/es/book/download/guia-completa-de-css3>
- Neftali, T. (2016). Universidad autónoma del estado de México. *Poblacion y Muestra*, 1–134. <https://core.ac.uk/download/pdf/80531608.pdf>
- Nielsen, J. (2017). *Jakob Nielsen: be user friendly*. <https://profile.es/blog/los-10-principios-de-usabilidad-web-de-jakob-nielsen/>
- Pardo Aguilar, C., & García Peñalvo, F. J. (2016). *Diagramas de Clase en UML 1.1*. 8.



<https://repositorio.grial.eu/bitstream/grial/353/1/DClase.pdf>

- Pascagaza Gitierrez, J. M. (2018). Desarroll de un sistema de informacion para la gestion de los proyectos de responsabilidad social del programa de ingenieria de sistemas de la universidad católica de Colombia. *Bitkom Research*, 63(2), 1–3. <https://repository.ucatolica.edu.co/server/api/core/bitstreams/8afbb665-45be-47b6-94d5-ffe999462cba/content>
- Peicevic, A. (2016). “*Apache HTTP Server introduction.*” 53.
- Perú, C. de la R. (2011). Ley de Protección de Datos Personales Nro 29733. *El Peruano*, 31. http://www.pcm.gob.pe/transparencia/Resol_ministeriales/2011/ley-29733.pdf
- Pressman, R. S. (2013). Ingeniería de Software un enfoque práctico. In *Journal of Chemical Information and Modeling* (Vol. 53, Issue 9).
- Ricardo, C. M. (2009). *Bases de Datos* (Vol. 17).
- Romero, M. I., Figueroa, G. L., Vera, D. S., Álava, J. E., Parrales, G. R., Álava, C. J., Murillo, Á. L., & Castillo, M. A. (2018). Mecanismo Correctivos en seguridad informática. In *Introducción a la seguridad informática y el análisis de vulnerabilidades*.
- Rumbaugh, J., Jacobson, I., & Booch, G. (2007). El Lenguaje Unificado de Modelado. In *Elements*. <http://portal.acm.org/citation.cfm?id=993859&dl=>
- Sans, A., & Atenea Alonso Serrano, Lorena García Sanz, Irene León Rodrigo, Elisa García Gordo, Belén Gil Álvaro, L. R. B. (2012). Métodos de investigación de enfoque experimental. *Metodología de La Investigación Educativa*, 167–193. <http://www.postgradoune.edu.pe/documentos/Experimental.pdf>
- Sommerville, I. (2005). *INGENIERIA DEL SOFTWARE. Séptima edición* (p. 712). <http://danielr.obolog.es/ingenieria-software-355416%5Cn>
- Tebar, E. (2020). Frameworks en el desarrollo web: las mejores prácticas para tu negocio online. *Desarrollo Web*.
- Thanthirige, P., Shanaka, R., Of, A., Contributing, F., Time, T. O., Of, O., Shehzad, A.,



& Keluarga, D. D. (2016). *Modelo Vista Controlador. August*, 1–12.

Torres Ccoa, M. A. (2019). Sistema De Información De Control De Farmacia Para Los Establecimientos De La Red De Salud Puno. *Tesis*, 113.
http://repositorio.unap.edu.pe/bitstream/handle/UNAP/12740/Torres_Ccoa_Milton_Arnold.pdf?sequence=1&isAllowed=y

Torres Torres, guido hernan. (2018). Aplicacion web para la emisión de constancias electronicas en el repositorio instrucional de la Universidad Nacional del Altiplano 2018. *Repositorio Universidad Nacional Del Altiplano, Anexo 1*, 100.
<http://repositorio.unap.edu.pe/handle/20.500.14082/13970>

Trigoso Inuma, A. (2018). “Desarrollo de un sistema de información para el control de registro académico en el centro de educación técnico - productiva, Yurimaguas.” *Universidad Nacional de San Martín*, 52.
<http://repositorio.unsm.edu.pe/handle/11458/574>

Vega, J. F., & Van Der Henst, C. (2011). *Guía HTML 5*. 1–47.
<http://www.maestrosdelweb.com/guias/>



ANEXOS



Anexo 1. valuación de calidad de software estándar ISO – 9126.

INDICADORES	PREGUNTAS	PUNTUACION				
		1	2	3	4	5
FUNCIONALIDAD						
Aplicabilidad	¿Tiene el conjunto de funciones apropiadas para las tareas específicas??					
Precisión	¿Hace lo que fue acordado en forma esperada y correcta?					
Interoperabilidad	¿Interactúa con otros sistemas específicos sin efectuar su funcionamiento?					
Seguridad de acceso	¿El sistema puede impedir el acceso a personal no autorizado?					
CONFIABILIDAD						
Madurez	¿El sistema funciona sin presentar fallas?					
Tolerancia a errores	¿Las fallas ocasionales del sistema no afectan gravemente el funcionamiento del sistema?					
Recuperabilidad	¿Es capaz de recuperar datos en caso de fallas?					
USABILIDAD						
Entendimiento	¿Es fácil entender y reconocer la estructura y la lógica y su aplicabilidad					
Aprendizaje	¿Es fácil de aprender a usar el sistema?					
Operabilidad	¿Es fácil de operar y controlar el sistema?					
Atracción	¿Es atractivo el diseño del software?					
EFICIENCIA						
Comportamiento de tiempos	¿Qué tan parido responde el sistema?					
Utilización de recursos	¿Cuántos recursos usa y durante cuánto tiempo?					
MANTENIMIENTO						
Capacidad de ser analizado	¿Es fácil de diagnosticar una falla o identificar partes a modificar?					
Confiabilidad	¿Es fácil de modificar y adaptar?					
Estabilidad	¿Hay riesgos o efectos inesperados cuando se realizan cambios?					
Facilidad de prueba	¿Son fáciles de validar las modificaciones?					
PORTABILIDAD						
Adaptabilidad	¿Es fácil de adaptar a otros entornos con lo provisto?					
Facilidad de instalación	¿Es fácil de instalar en el ambiente especificado?					
Remplazabilidad	¿Es fácil de usarlo en lugar de otro software para ese ambiente?					
Coexistencia	¿El software puede funcionar con otros sistemas?					
Eficacia	¿La eficaz el software cuando el usuario final realiza los procesos?					



Anexo 3. Manual de usuario de SIGAINDUSTRIAL32

MANUAL DE USUARIO

SISTEMA INFORMÁTICO PARA LA GESTIÓN ACADÉMICA DE LA INSTITUCIÓN EDUCATIVA SECUNDARIA INDUSTRIAL 32



“SIGAINDUSTRIAL32”

www.sigaindustrial32.com

En el presente documento el usuario podrá hacer el correcto uso del sistema informático, como bien sabemos, el sistema informático cumple con ciertos de ingreso y diferentes roles de usuario. El sistema se caracteriza por tener 3 tipos de usuario; el administrador, docente y estudiante. Se hace la descripción detallada de quien será responsable del uso del sistema para así facilitar la gestión académica, administración de información, matrículas y registros de administradores, docentes y estudiantes.



Acceso al sistema

Administrador: En el inicio de sesión al software, el administrador deberá ser registrado previamente y tener todos los permisos que sean necesarios para que luego se haga el uso correcto del sistema y así poder administrar el sistema por completo. Una vez registrado

y obtenido todos los permisos, se procederá a realizar el registro de los diferentes usuarios.

1) Inicio de sesión:

iniciar sesion Responsables

COLEGIO INDUSTRIAL N° 32



COMPUTACIÓN

La Computación es la disciplina que se encarga del estudio de las computadoras, abarcando su diseño, desarrollo y procesamiento (hardware), mientras que la Informática recoge y ordena los datos convirtiéndolos en información sencilla (software).



INDUSTRIA ALIMENTARIA

La industria de alimentos procesa la materia prima para obtener alimentos de consumo humano o animal. En este proceso se incluyen las fases de transporte, recepción, almacenamiento, procesamiento, conservación. Las materias primas utilizadas son de origen agropecuario principalmente.



ELECTRICIDAD

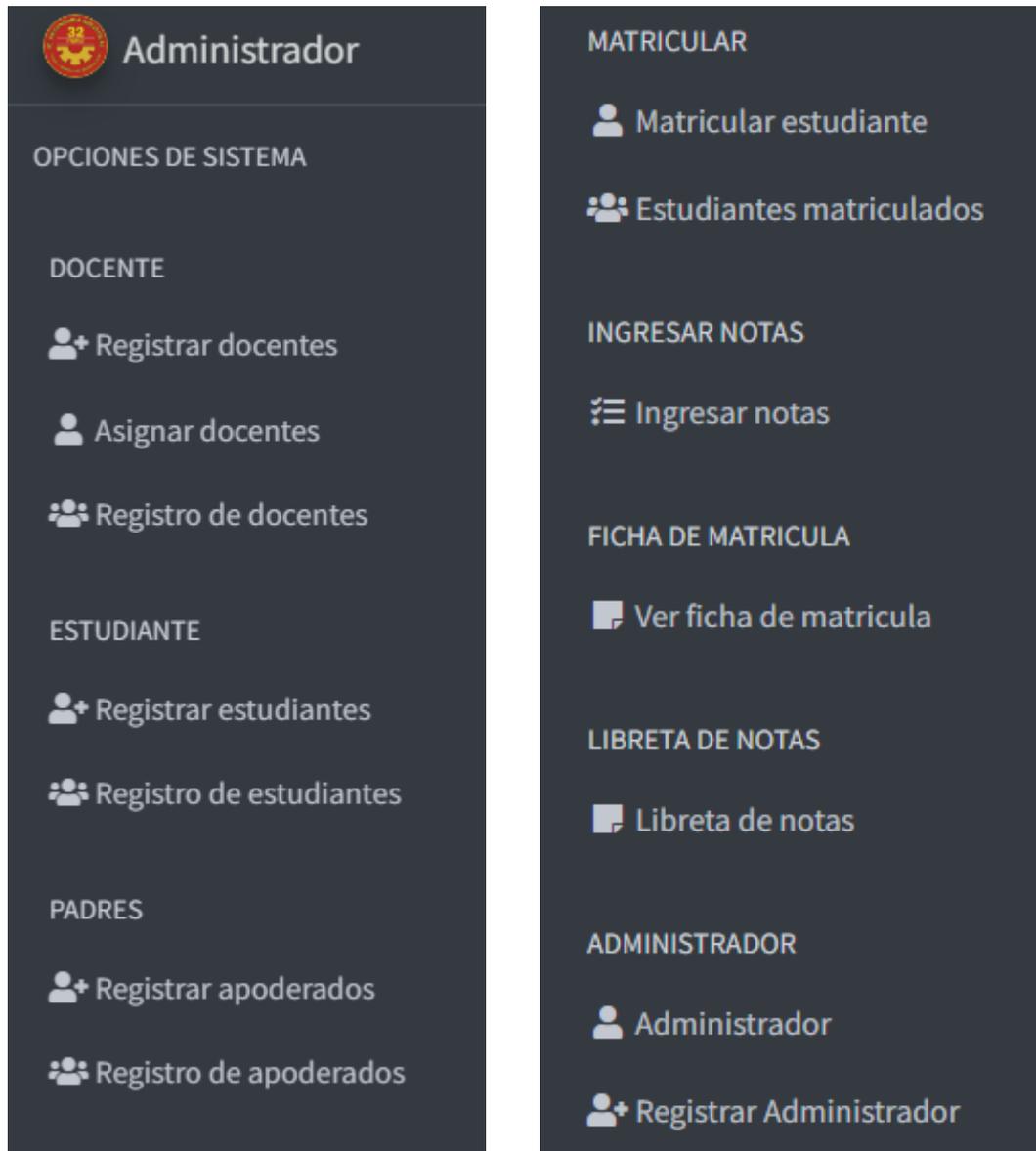
La electricidad es una forma de energía que se manifiesta con el movimiento de los electrones de la capa externa de los átomos que hay en la superficie de un material conductor. La electricidad es un fenómeno íntimamente ligado en la materia y a la vida.

Al dar clickear en el inicio de sesión, se apertura una ventana donde podrá ingresar su usuario y contraseña, para el ingreso de quien sea el administrador, docente o estudiante.



The login screen features a background image of the school building. It includes the school's logo at the top left and right. The central login form has a red circular logo with 'IE SECUNDARIA INDUSTRIAL 32 PUNO' and 'LUZ A LA MENTE A LAS MANOS ACCION'. Below the logo, there are input fields for 'Correo' (with the email 'alexander.huaraya.c@gmail.com') and 'Contraseña' (with masked characters). A 'LOGIN' button is positioned below the password field. At the bottom, a hand icon points to the text 'Colegio inteligente Rumbo al licenciamiento!!!'. On the right side, there is a large blue and white logo for 'IES INDUSTRIAL 32 PUNO' with the motto 'LUZ A LA MENTE A LAS MANOS ACCION'.

En el caso del administrador, se apertura la interface principal donde se puede visualizar todos los módulos que se visualiza a continuación:



En este caso el ADMIN tiene acceso a todos los formularios para que pueda hacer el registro de diferentes usuarios, así también de la información que será necesaria para el registro de los estudiantes y padres como se muestra a continuación:

Registro de docentes, asignación de cursos a los docentes, registro de estudiantes, registro de padres de familia, matrícula de estudiantes e ingreso de notas.



REGISTRAR ESTUDIANTE

DNI	<input type="text"/>	Genero	Masculino <input type="button" value="v"/>
Nombre	<input type="text"/>	Direccion	<input type="text"/>
Apellido paterno	<input type="text"/>	Celular	<input type="text"/>
Apellido materno	<input type="text"/>	Email	<input type="text"/>
Fecha de nacimiento	dd/mm/aaaa <input type="button" value="📅"/>	Password	<input type="text"/>
Lugar de nacimiento	<input type="text"/>	Documento para publicacion	<input type="button" value="Elegir archivo"/> No se eligió ningún archivo

Unir los 3 PDF y subir un unico archivo:

- copia Dni
- copia Partida Nacimiento
- copia Certificado de estudios (primaria)

REGISTRO DE DOCENTES

DNI	<input type="text"/>	Apellido materno	<input type="text"/>
Nombre	<input type="text"/>	Celular	<input type="text"/>
Profesion	<input type="text"/>	Email	<input type="text"/>
Apellido paterno	<input type="text"/>	Password	<input type="text"/>
		Enviar	<input type="button" value="Guardar datos"/>

ASIGNAR DOCENTE

DNI	<input type="text" value="Buscar por DNI del docente"/>	Seccion	A <input type="button" value="v"/>
		Grado	PRIMERO <input type="button" value="v"/>
		Año académico	2022 <input type="button" value="v"/>
		Especialidad	AIP - ROBÓTICA <input type="button" value="v"/>
		Curso	ARTE <input type="button" value="v"/>
		Enviar	<input type="button" value="Guardar datos"/>



REGISTRAR PADRES

DNI estudiante

Datos de apoderado

Genero	<input type="text" value="Padre"/>	DNI	<input type="text"/>
Nombre	<input type="text"/>	Apellido paterno	<input type="text"/>
Apellido materno	<input type="text"/>	Fecha de nacimiento	<input type="text" value="dd/mm/aaaa"/>
Lugar de nacimiento	<input type="text"/>	Vive	<input type="text" value="Si"/>
Grado de instruccion	<input type="text"/>	Estado civil	<input type="text"/>
Email	<input type="text"/>	Celular	<input type="text"/>
Direccion	<input type="text"/>	Ocupacion	<input type="text"/>
		Copia de DNI	<input type="button" value="Elegir archivo"/> No se eligió ningún archivo

Subir copia de DNI en PDF:

MATRICULAR A ESTUDIANTE

DNI	<input type="text" value="buscar por DNI"/>	Seccion	<input type="text" value="A"/>
		Grado	<input type="text" value="PRIMERO"/>
		Año académico	<input type="text" value="2022"/>
		Especialidad	<input type="text" value="AIP - ROBÓTICA"/>
		Enviar	<input type="button" value="Guardar datos"/>

Anexo 4. Manual de usuario docente

El sistema (módulo docente) de gestión de notas es una herramienta diseñada para facilitar la tarea de ingresar, modificar y consultar las calificaciones de los estudiantes, lo cual se proporciona instrucciones detalladas sobre cómo utilizar el sistema de manera efectiva.

Para acceder al sistema, siga los siguientes pasos:

1. Abre tu navegador web favorito e ingresa la URL proporcionada por el administrador del sistema, introduce tu nombre de usuario y contraseña proporcionada por el administrador, luego haz clic en el botón Login.



2. Después de iniciar sesión, serás dirigido a la pantalla principal del sistema, aquí encontrarás las siguientes secciones:





Ficha de matricula



PERÚ

Ministerio
de Educación

INSTITUCION EDUCATIVA SECUNDARIA INDUSTRIAL 32

FICHA DE MATRICULA

MARLENI TARAPA QUISPE

Año académico 2023

N° Matricula / Dni: 73814424

Grado y Seccion: 5° - B

Cursos

- ARTE
- CIENCIA TECNOLOGIA Y AMBIENTE
- COMUNICACIÒN
- FORMACION CIUDADANA Y CIVICA
- INGLES
- EDUCACION FISICA
- EDUCACION RELIGIOSA
- HISTORIA GEOGRAFIA Y ECONOMIA
- MATEMATICA
- PERSONA FAMILIA Y RELACIONES HUMANAS
- EDUCACION PARA EL TRABAJO

En la parte izquierda de la pantalla, encontraras un menú que te permita acceder a las diferentes funciones del sistema, así mismo podrás realizar búsquedas para encontrar estudiantes y asignaturas específicas que le fueron asignadas para dictar clases y en el módulo “ingresar notas” podrá realizar el ingreso de notas a los cursos asignados.

Anexo 5. Código fuente del sistema de información

Rutas del sistema.

```
routes > web.php
1 <?php
2
3 > use Illuminate\Support\Facades\Route; ...
16
17 Route::resource('estudiante', EstudianteController::class); //INTERFAZ
18 Route::resource('docente', DocenteController::class); // INTERFAZ
19 Route::resource('matricula', MatriculaController::class); // INTERFAZ
20 Route::resource('reportenotas', ReporteEstudianteController::class); //INTERFAZ
21 Route::resource('asignacion', AsignacionController::class);// INTERFAZ
22 Route::resource('curso', CursoController::class);//INTERFAZ
23 Route::resource('colocacionnotas', ColocacionNotasController::class);//INTERFAZ
24 Route::resource('fichamatricula', FichaMatriculaController::class);
25 Route::resource('apoderado', ApoderadoController::class);
26
27 Route::resource('users', UserController::class)->names('admin.users');
28
29
30 Route::get('/', [PostController::class, 'index']->name('posts.index'));
31
32
33 Route::middleware(['auth:sanctum', 'verified']->get('/dashboard', function () {
34     return view('dashboard');
35 }->name('dashboard'));
36
37 Route::get('/reporte_estudiante_padre', [EstudianteController::class, 'reporte_estudiante_padre']->name('reporte_estudiante_padre');
38 Route::get('/reporte_estudiante_matriculado', [MatriculaController::class, 'reporte_estudiante_matriculado']->name('reporte_estudiante_mat');
39 Route::get('/reporte_docente_asignado', [AsignacionController::class, 'reporte_docente_asignado']->name('reporte_docente_asignado');
40 Route::get('/reporte_docente_estudiantes', [AsignacionController::class, 'reporte_docente_estudiantes']->name('reporte_docente_estudiantes');
```

Registrar estudiante:

```
EstudianteController.php M X index.blade.php ...estudiante estudiante-index.blade.php index.blade.php ...asignacion web
app > Http > Controllers > EstudianteController.php > App\Http\Controllers\EstudianteController > store()
45
46 0 references | 0 overrides
47 public function store(Request $request)
48 {
49     $request->validate(
50         [ ...
51     ],
52     [ ...
53     ]
54 );
55
56 $datosEstudianteApoderado = request()->except('_token');
57 $estudiante = Estudiante::create(request()->all());
58
59 if($request->file('documento')){ ...
60 }
61
62 //agregar usuario tipo estudiante
63 $user = new User();
64 $user->name = $datosEstudianteApoderado["nombre"];
65 $user->email = $datosEstudianteApoderado["email"];
66 $user->password = Hash::make( $datosEstudianteApoderado["password"] );
67 $user->identificador_estudiante = $estudiante->idestudiante;
68 $user->assignRole('EstudianteUsuario');
69 $user->save();
70
71 return $this->index();
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101 0 references | 0 overrides
```

Registro de Docentes:

```
DocenteController.php 5 X users-index.blade.php UsersIndex.php EstudianteIndex.php ApoderadoIndex.php
app > Http > Controllers > DocenteController.php > App\Http\Controllers\DocenteController > index()
0 references | 0 overrides
47 public function store(Request $request)
48 {
49     $request->validate(
50         [
51             'dni' => ['required', 'size:8'],
52             'nombre' => 'required',
53             'apellido_paterno' => 'required',
54             'apellido_materno' => 'required',
55             'profesion' => 'required',
56             'celular' => 'required',
57             'email' => ['required', 'email', 'unique:users,email'],
58             'password' => ['required', 'Password::min(8)],
59         ],
60         [
61             'dni.required' => 'El campo no puede estar vacio',
62             'dni.numeric' => 'DNI debe ser numerico',
63             'dni.max' => 'DNI debe ser maximo 8 digitos',
64             'nombre.required' => 'El campo no puede estar vacio',
65             'apellido_paterno.required' => 'El campo no puede estar vacio',
66             'apellido_materno.required' => 'El campo no puede estar vacio',
67             'profesion.required' => 'El campo no puede estar vacio',
68             'celular.required' => 'El campo no puede estar vacio',
69             'email.required' => 'El campo no puede estar vacio',
70             'email.email' => 'El campo debe ser correo electrónico',
71             'password.required' => 'El campo no puede estar vacio'
72         ]
73     );
74
75     $datosDocente = request()->except('_token');
76     $docente = Docente::create(request()->all());
77     Docente::insert($datosDocente);
78
79     //agregar usuario tipo estudiante
80     $user = new User();
81     $user->name = $datosDocente["nombre"];
82     $user->email = $datosDocente["email"];
83     $user->password = Hash::make( $datosDocente["password"] );
84     $user->identificador_docente = $docente->iddocente;
85     $user->assignRole('DocenteUsuario');
86     $user->save();
87
88     return redirect('/docente')->with('mensaje', 'Docente agregado con éxito');
89
90
91
```

Matriculas de Estudiantes:

```
lex.blade.php index.blade.php ..\matricula MatriculaIndex.php matricula-index.blade.php MatriculaController.php M X docente-index.blade.php
app > Http > Controllers > MatriculaController.php > App\Http\Controllers\MatriculaController > store()
0 references | 0 overrides
45 public function store(Request $request)
46 {
47     $datosForm = request()->except('_token');
48
49     // HECHO. Consultar la tabla 'niveles' where grado = 1 y seccion = C para extraer el valor de 'idniveles' "3".
50     $nivel = Nivel::where('grado', $datosForm['grado'])
51         ->where('seccion', $datosForm['seccion'])->first();
52
53     // // Consultar la tabla 'cursos' where 'idniveles' "3" para extraer el array de cursos e insertarlo en 'detall_matricula'.
54     $cursos = DB::select('select * from cursos where niveles_idniveles = :id', ['id' => $nivel->idnivel]);
55     $estudiante = Estudiante::where('idestudiante', '=', $datosForm['idestudiante']);
56
57     DB::table('matriculas')->insert([
58         'estudiante_idestudiante' => $datosForm['idestudiante'],
59         'anio_academico' => $datosForm['anio_academico']
60     ]);
61
62     $matricula = Matricula::where('estudiante_idestudiante', $datosForm['idestudiante'])->first();
63
64     foreach($cursos as $curso){
65         DB::table('detalle_matriculas')->insert(
66             [
67                 'matriculas_idmatricula' => $matricula->idmatricula,
68                 'cursos_idcurso' => $curso->idcurso
69             ]
70         );
71     }
72 }
73
```

Asignación de curso a docentes:

```
p ..\matricula MatriculaIndex.php matricula-index.blade.php MatriculaController.php M AsignacionController.php M X docente-index.blade.php
app > Http > Controllers > AsignacionController.php > App\Http\Controllers\AsignacionController > store()
79
0 references | 0 overrides
80 public function store(Request $request)
81 {
82     // DATOS DESDE 'asignacion.index'
83     $datosForm = request()->except('_token');
84
85     // HECHO. Consultar la tabla 'niveles' where grado = 1 y seccion = C para extraer el valor de 'idniveles' "3".
86     $nivel = Nivel::where('grado', $datosForm['grado'])
87         ->where('seccion', $datosForm['seccion'])->first();
88
89     $cursos = Curso::where('niveles_idniveles', '=', $nivel->idnivel)->where('descripcion', '=', $datosForm['Curso'])->first();
90
91     DB::table('asignaciones')->insert(
92         ['docentes_iddocente' => $datosForm['IdDocente']]
93     );
94
95     $asignacion = Asignacion::where('docentes_iddocente', $datosForm['IdDocente'])->first();
96
97     DB::table('detalle_asignaciones')->insert(
98         [
99             'asignaciones_idasignacion' => $asignacion->idasignacion, 'cursos_idcurso' => $cursos->idcurso
100         ]
101     );
102
103     return redirect('/asignacion')->with('mensaje', 'Asignacion agregado con exito');
104 }
105
0 references | 0 overrides
public function show($id)
```

Ingreso de notas:

```
matricula-index.blade.php MatriculaController.php M AsignacionController.php M ColocacionNotasController.php M X docente-index.bl...
app > Http > Controllers > ColocacionNotasController.php > App\Http\Controllers\ColocacionNotasController > update()
33
34 0 references | 0 overrides
35 function update(Request $request, $id)
36 {
37     $datosColocacion = request()->except('_token', '_method');
38
39     $tamano = sizeof($datosColocacion)/9 ; // 9 elementos que se envian
40
41     for ($i=1; $i <= $tamano ; $i++) {
42         # code...
43
44         $promedio = ($datosColocacion['nota1'.$i]+$datosColocacion['nota2'.$i]+$datosColocacion['nota3'.$i])/3;
45
46         $proemdio = number_format($promedio,2);
47
48         DB::table('detalle_matriculas')
49         ->where('matriculas_idmatricula', $datosColocacion['idmatricula'.$i] )
50         ->where('cursos_idcurso', $datosColocacion['idcurso'.$i] )
51         ->update([
52             'nota1' => $datosColocacion['nota1'.$i],
53             'nota2' => $datosColocacion['nota2'.$i],
54             'nota3' => $datosColocacion['nota3'.$i],
55             'promedio' => $promedio
56         ]);
57     }
58
59     return $this->index();
60 }
```

Ficha de matricula:

```
MatriculaController.php M AsignacionController.php M ColocacionNotasController.php M FichaMatriculaController.php M X docente-index...
app > Http > Controllers > FichaMatriculaController.php > ...
1 <?php
2
3 namespace App\Http\Controllers;
4
5 use Illuminate\Http\Request;
6 use Illuminate\Support\Facades\DB;
7
8 4 references | 0 implementations
9 class FichaMatriculaController extends Controller
10 {
11     //
12     0 references | 0 overrides
13     public function index()
14     {
15         $id = auth()->user()->identificador_estudiante;
16         $datos['matriclass'] = DB::table('estudiantes')
17         ->join('matriculas', 'estudiantes.idestudiante', '=', 'matriculas.estudiante_idestudiante')
18         ->select('matriculas.idmatricula','estudiantes.*')
19         ->where('estudiantes.idestudiante', '=', $id)
20         ->get();
21
22         return view('fichamatricula.index', $datos);
23     }
24 }
25
26 }
```



DECLARACIÓN JURADA DE AUTENTICIDAD DE TESIS

Por el presente documento, Yo **ALEXANDER HUARAYA CANLLAHUE**,
identificado con **DNI 48217827** en mi condición de egresado de:

Escuela Profesional, Programa de Segunda Especialidad, Programa de Maestría o Doctorado

INGENIERÍA ESTADÍSTICA E INFORMÁTICA,

informo que he elaborado el/la Tesis o Trabajo de Investigación denominada:

**“SISTEMA INFORMÁTICO PARA LA GESTIÓN ACADÉMICA APLICADO EN LA
INSTITUCIÓN EDUCATIVA SECUNDARIA INDUSTRIAL 32 DE PUNO”**

Es un tema original.

Declaro que el presente trabajo de tesis es elaborado por mi persona y **no existe plagio/copia** de ninguna naturaleza, en especial de otro documento de investigación (tesis, revista, texto, congreso, o similar) presentado por persona natural o jurídica alguna ante instituciones académicas, profesionales, de investigación o similares, en el país o en el extranjero.

Dejo constancia que las citas de otros autores han sido debidamente identificadas en el trabajo de investigación, por lo que no asumiré como tuyas las opiniones vertidas por terceros, ya sea de fuentes encontradas en medios escritos, digitales o Internet.

Asimismo, ratifico que soy plenamente consciente de todo el contenido de la tesis y asumo la responsabilidad de cualquier error u omisión en el documento, así como de las connotaciones éticas y legales involucradas.

En caso de incumplimiento de esta declaración, me someto a las disposiciones legales vigentes y a las sanciones correspondientes de igual forma me someto a las sanciones establecidas en las Directivas y otras normas internas, así como las que me alcancen del Código Civil y Normas Legales conexas por el incumplimiento del presente compromiso

Puno 29 de ENERO del 20 24



FIRMA (obligatoria)



Huella



AUTORIZACIÓN PARA EL DEPÓSITO DE TESIS O TRABAJO DE INVESTIGACIÓN EN EL REPOSITORIO INSTITUCIONAL

Por el presente documento, Yo **ALEXANDER HUARAYA CANLLAHUE**, identificado con **DNI 48217827** en mi condición de egresado de:

Escuela Profesional, Programa de Segunda Especialidad, Programa de Maestría o Doctorado

INGENIERÍA ESTADÍSTICA E INFORMÁTICA,

informo que he elaborado el/la Tesis o Trabajo de Investigación denominada:

“SISTEMA INFORMÁTICO PARA LA GESTIÓN ACADÉMICA APLICADO EN LA INSTITUCIÓN EDUCATIVA SECUNDARIA INDUSTRIAL 32 DE PUNO”

para la obtención de Grado, Título Profesional o Segunda Especialidad.

Por medio del presente documento, afirmo y garantizo ser el legítimo, único y exclusivo titular de todos los derechos de propiedad intelectual sobre los documentos arriba mencionados, las obras, los contenidos, los productos y/o las creaciones en general (en adelante, los “Contenidos”) que serán incluidos en el repositorio institucional de la Universidad Nacional del Altiplano de Puno.

También, doy seguridad de que los contenidos entregados se encuentran libres de toda contraseña, restricción o medida tecnológica de protección, con la finalidad de permitir que se puedan leer, descargar, reproducir, distribuir, imprimir, buscar y enlazar los textos completos, sin limitación alguna.

Autorizo a la Universidad Nacional del Altiplano de Puno a publicar los Contenidos en el Repositorio Institucional y, en consecuencia, en el Repositorio Nacional Digital de Ciencia, Tecnología e Innovación de Acceso Abierto, sobre la base de lo establecido en la Ley N° 30035, sus normas reglamentarias, modificatorias, sustitutorias y conexas, y de acuerdo con las políticas de acceso abierto que la Universidad aplique en relación con sus Repositorios Institucionales. Autorizo expresamente toda consulta y uso de los Contenidos, por parte de cualquier persona, por el tiempo de duración de los derechos patrimoniales de autor y derechos conexos, a título gratuito y a nivel mundial.

En consecuencia, la Universidad tendrá la posibilidad de divulgar y difundir los Contenidos, de manera total o parcial, sin limitación alguna y sin derecho a pago de contraprestación, remuneración ni regalía alguna a favor mío; en los medios, canales y plataformas que la Universidad y/o el Estado de la República del Perú determinen, a nivel mundial, sin restricción geográfica alguna y de manera indefinida, pudiendo crear y/o extraer los metadatos sobre los Contenidos, e incluir los Contenidos en los índices y buscadores que estimen necesarios para promover su difusión.

Autorizo que los Contenidos sean puestos a disposición del público a través de la siguiente licencia:

Creative Commons Reconocimiento-NoComercial-CompartirIgual 4.0 Internacional. Para ver una copia de esta licencia, visita: <https://creativecommons.org/licenses/by-nc-sa/4.0/>

En señal de conformidad, suscribo el presente documento.

Puno, 29 de ENERO del 2024


FIRMA (obligatoria)

