

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



**DESARROLLO DE UN SISTEMA DE REGISTRO CIVIL PARA LA
MUNICIPALIDAD DISTRITAL DE PLATERIA PUNO- 2017**

TESIS

PRESENTADA POR:

BACH. ELIZABETH AROPAZA ARPASI

PARA OPTAR EL TÍTULO PROFESIONAL DE:

INGENIERO ESTADÍSTICO E INFORMÁTICO

PUNO – PERÚ

2018

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

DESARROLLO DE UN SISTEMA DE REGISTRO CIVIL PARA LA
MUNICIPALIDAD DISTRITAL DE PLATERIA PUNO- 2017

TESIS PRESENTADA POR:

Bach. ELIZABETH AROPAZA ARPASI

PARA OPTAR EL TÍTULO PROFESIONAL DE:

INGENIERO ESTADÍSTICO E INFORMÁTICO



APROBADA POR EL JURADO REVISOR CONFORMADO POR:

PRESIDENTE :


M.Sc. ERNESTO NAYER TUMI FIGUEROA

PRIMER MIEMBRO :

M.SC. CHARLES IGNACIO MENDOZA MOLLOCONDO

SEGUNDO MIEMBRO :


ING. RONALD MAMANI MAYTA

DIRECTOR / ASESOR :


Dr. JUAN REYNALDO PAREDES QUISPE

Área : Informática
Tema : Sistema de Información
Fecha de Sustentación : 12/04/2018

DEDICATORIA

*Al ser más supremo del universo, por concederme vida y sabiduría DIOS
nuestro Señor Todo poderoso.*

*Con infinito cariño a mi padre Isauro que en todo momento me proporciono
siempre todo lo que necesité a cambio de su sacrificio, esfuerzo y
comprensión, a mi madre Isabel que desde el cielo ilumino mi camino;
hicieron posible la culminación de mi carrera profesional.*

*A mis hermanos Veronica, David, Mery y Mary que con sus consejos me
han ayudado a afrontar los retos que se me han presentado a lo largo de la
vida y en especial en mis años de estudio en nuestra casa de estudios la
Universidad Nacional del Altiplano.*

*A mi novio Howald quien me apoyo y alentó para continuar, cuando
parecía que me iba a rendir. Gracias.*

Elizabeth

AGRADECIMIENTO

A Dios, por haberme dado sabiduría, fortaleza, salud, coraje y no dejarme solo en los momentos difíciles, y haberme permitido llegar a la meta en este gran trabajo.

A mis padres y a mis hermanos, que siempre me han dado su apoyo incondicional y a quienes debo este triunfo profesional, por todo su trabajo y dedicación para darme una formación académica.

A todos mis docentes de la Escuela Profesional de Ingeniería Estadística e Informática por su gran apoyo y motivación para la culminación de nuestros estudios profesionales y para la elaboración de esta tesis.

ÍNDICE GENERAL

RESUMEN.....	10
ABSTRACT.....	11
CAPITULO I INTRODUCCIÓN.....	12
1.1. Planteamiento del Problema.....	13
1.2. Formulación del Problema.....	14
1.3. Hipótesis de la Investigación	14
1.4. Justificación de la Investigación.....	14
1.5. Objetivos de la Investigación	15
CAPÍTULO II REVISION DE LA LITERATURA	16
2.1. Marco Teórico	16
2.1.1. Antecedentes de la Investigación	16
2.2. Marco Conceptual	20
2.2.1. Sistemas de Información	20
2.2.2. Tipos de Sistema de Información	22
2.2.3. Sistemas Web	24
2.2.4. Ingeniería de Software	25
2.2.5. Software	25
2.2.6. Programación Extrema (XP).....	26
2.2.7. Fases de Desarrollo Con XP	26
2.2.8. Pruebas Software	28
2.2.9. Norma de Evaluación ISO/IEC 9126	29
2.2.10. Base de Datos.....	39
2.2.11. Modelado de software con UML	39
2.2.12. UML	40
2.2.13. Gestor de base de datos	45
2.2.14. framework	56
2.3. Definición de Términos Básicos.....	59
2.4. Municipalidad distrital de Plateria	61

CAPITULO III MATERIALES Y METODOS	64
3.1. Tipo de investigación	64
3.2. Población y Muestra del Estudio.....	65
3.3. Operacionalización de Variables	66
3.4. Método de Recopilación de Datos	67
3.4.1. Prueba de Hipótesis	68
3.5. Arquitectura de Software	70
3.5.1. Ciclo de Desarrollo de la Arquitectura	70
3.6. Desarrollo del Sistema	72
3.7. Material Experimental.....	77
3.7.1. Software	77
3.7.2. Hardware.....	78
CAPITULO IV RESULTADOS Y DISCUSION.....	79
4.1. Resultados	79
4.1.1. Ámbito del Problema	79
4.1.2. Especificación de Requerimientos del Sistema	79
Identificación y descripción de historias de usuarios y modulos del sistema	80
4.1.3. Elaboración de diagramas de caso de uso	80
4.1.4. Elaboración de diagrama de secuencia	84
4.1.5. Diseño	85
4.1.6. Pruebas.....	86
CAPÍTULO V CONCLUSIONES.....	92
CAPÍTULO VI RECOMENDACIONES	93
CAPÍTULO VII REFERENCIAS BIBLIOGRAFIAS	95
ANEXOS	100

ÍNDICE DE FIGURAS

Figura N° 1 Elementos básicos de un sistema de información	21
Figura N° 2 Resumen las áreas principales del conocimiento de los sistemas de información que necesitan los profesionales de los negocios...22	22
Figura N° 3 Tipos de información	24
Figura N° 4 Ciclo de pruebas de calidad de software	29
Figura N° 5 Modelo de Calidad Externa e Interna y Calidad de Uso	29
Figura N° 6 Funcionalidad y sus características	30
Figura N° 7 Confiabilidad	31
Figura N° 8 Usabilidad	33
Figura N° 9 Eficiencia.....	34
Figura N° 10 Capacidad de Mantenimiento	35
Figura N° 11 Portabilidad	36
Figura N° 12 Calidad en Uso.....	38
Figura N° 13 Proceso UML	41
Figura N° 14 Modelo de Diagrama de Caso de Uso.....	42
Figura N° 15 Ubicación del Distrito de Platería.....	63
Figura N° 16 Diseño de investigación	67
Figura N° 17 Programación Extrema XP	72
Figura N° 18 Acciones del Administrador del Sistema.....	81
Figura N° 19 Diagrama de Uso del Requerimiento Registrar Actas.....	82
Figura N° 20 Diagrama de Secuencia Ingreso de Usuarios	84
Figura N° 21 Entidad Relación	85
Figura N° 22 Modelo desarrollo web	86
Figura N° 23 Resultados de la Validación de la Calidad del Producto de Software del Sistema	90

ÍNDICE DE TABLAS

Tabla N° 1 Lenguajes de Programación Actuales.....	51
Tabla N° 2 Población de usuario área de registro civil (Mes de enero 2018) ...	64
Tabla N° 3 Operacionalización de variables independientes	66
Tabla N° 4 Operacionalización de variables dependientes	67
Tabla N° 5 Acciones Generales del Administrador	82
Tabla N° 6 Acciones generales del usuario	83
Tabla N° 7 Valoraciones del dominio de información del desarrollo de un sistema de Registro Civil para la municipalidad distrital de Plateria.....	87
Tabla N° 8 Valoraciones de ajuste de la complejidad del desarrollo de un registro civil para la municipalidad distrital de Plateria	88
Tabla N° 9 Resultados de la Validación de la Calidad del Producto de Software del Sistema	90

ÍNDICE DE ACRÓNIMOS

XP	: Programación Extrema
SUBLIME TEXT	: Editor de códigos y texto
MYSQL	: Sistema de gestor de base de datos relacionales
WORKBENCH	: Herramienta visual para diseñar base de datos
XAMPP	: servidor web de plataforma, software libre
HTML	: Lenguaje de Marcado de Hiper Texto
PHP	: Pre-procesador de hiper-texto
Jquery	: Biblioteca multiplataforma de JavaScript
ISO	: Organización para la creación De Estándares Internacionales
AJAX	: Acrónimo de Asynchronous JavaScript And XM

RESUMEN

Los tramites y atenciones en la unidad de registro civil son lentas e ineficientes debido al sistema manual con que vienen laborando, justamente por ello se plante el siguiente objetivo, desarrollar un Sistema de Registro Civil para Municipalidad Distrital de Platería- 2017. La metodología que se utilizó para el desarrollo del sistema fue metodología XP (eXtreme Programming), el cual es un modelo evolutivo que nos lleva paso a paso desde el análisis hasta la aplicación web desarrollada, incidiendo en la comunicación fluida entre todos los actores del sistema proponiendo simplicidad en las soluciones implementadas y coraje para enfrentar los cambios. Las conclusiones de la investigación son: Se ha diseñado el aplicativo usando tecnologías de desarrollo basado en Xampp, con MySQL. Se ha modelado la base de datos con el software MySQL Workbench, como herramienta de diseño del diagrama. La interfaz del Sistema web es ergonómica de fácil acceso e intuitiva utilizando representaciones visuales y gráficas, ventanas, cuadros de diálogo, botones y comandos. La seguridad del sistema en el control de la información, dio como resultado que el 100% aprueba el Estándar ISO-9126 de calidad de software, ubicándose mayoritariamente en al escala de “Cumple los Requisitos”

Palabras Clave: Registro Civil, Desarrollo, html , Php, Sistema web, xampp y xp.

ABSTRACT

The procedures and attentions in the civil registry unit are slow and inefficient due to the manual system with which they are working, precisely because of this the following objective is set, to develop a Civil Registry System for District Municipality of Silverware- 2017. The methodology used for the development of the system was XP methodology (eXtreme Programming), which is an evolutionary model that takes us step by step from the analysis to the developed web application, influencing the fluid communication between all the actors of the system proposing simplicity in the implemented solutions and courage to face the changes. The conclusions of the research are: The application has been designed using Xampp-based development technologies, with MySQL. The database has been modeled with the MySQL Workbench software, as a diagram design tool. The web system interface is ergonomically easy to access and intuitive using visual and graphic representations, windows, dialog boxes, buttons and commands. The security of the system in the control of the information, resulted that 100% approves the ISO-9126 standard of software quality, being located mostly in the escape of "Meets the Requirements".

Keywords: Civil Registry, Development, html, Php, Web System, xampp and xp.

CAPITULO I

INTRODUCCIÓN

El principal objetivo de la presente Tesis se basó el desarrollo de un Sistema de Registro Civil para la Municipalidad Distrital de Plateria-2017, para el registro Mejora atención al usuario y dar la facilidad para hacer sus trámites de solicitud y consulta de formatos a través de la Web. Ayuda y apoyo a los trabajadores del registro civil.

Se utilizó el Lenguaje Unificado de Modelado (UML) para proporcionar una serie de modelos que permiten describir esquemáticamente al prototipo de sistema y su aplicación. La metodología que se utilizó para el desarrollo del sistema es la Metodología en XP; el cual nos lleva paso a paso desde el análisis hasta la aplicación web desarrollada de software de Registro Civil del Distrito de Plateria para poder contar con una herramienta confiable, rápida, ágil y oportuna que ayuda en el desarrollo de tareas de almacenamiento, consulta, solicitud y envío de formatos de Registros Civiles.

Esta herramienta, mejoró la calidad de atención al usuario. El prototipo de sistema permitió agilizar la solicitud de formatos a través del Web y en el mismo

registro civil, resolver sugerencias del usuario que solicita formatos.

A través de la encuesta, que el 100% de los operadores sistema califican de óptimo, confiable y confidencial el control de inscripciones de nacimientos, matrimonios, defunciones,

El sistema es confiable y confidencial el acceso a la información de datos de nacimiento y defunción vía Internet, con lo cual se ha reducido enormemente el espacio y tiempo en que se perdía al estar físicamente en el Registro Civil.

1.1. Planteamiento del Problema

En la actualidad la Unidad de Registro Civil de la municipalidad distrital de Plateria, no cuenta con un sistema automático de registro de datos de los usuarios y consulta e impresiones de actas de nacimientos, defunciones y matrimonios, así mismo que permita mejorar el tiempo de atención de los usuarios. Sin este sistema se ocasiona la mucha demora de trámite de las actas y reporte del estado de trámite documentos.

A través de las entrevistas previstas, se obtuvo como resultado que es necesaria la automatización del área de Registro Civil para la municipalidad distrital de Plateria, un Sistema que administre la información y genere los resultados óptimos e inmediatos, reduciendo el tiempo de respuesta satisfactoriamente, dando mayor oportunidad de realizar otras labores.

1.2. Formulación del Problema

¿De qué manera mejorará la atención en la oficina de Registro Civil en la Municipalidad Distrital de Platería 2017?

1.3. Hipótesis de la Investigación

El Desarrollo de Sistema para el Registro Civil en el Distrito de Platería- puno- 2017; mejorará la atención en la oficina de Registro Civil de la Municipalidad Distrital de Platería.

1.4. Justificación de la Investigación

En la oficina de Registro Civil de la Municipalidad Distrital de Platería, desempeña un papel fundamental en la conformación del Sistema de Registro Nacional de Identificación, ya que a través de las actas del estado civil se podrá certificar y acreditar la veracidad de la identidad de las personas; actualmente no cuenta ningún sistema de información, se ha notado una deficiencia en cuanto a las inscripciones y solicitudes de partidas de nacimientos, matrimonios, defunciones y también en los reportes estadísticos mensuales que se emiten a la RENIEC.

Con el desarrollo de un Sistema web de registro civil para la municipalidad distrital de Platería, que se aplicó, ayudó a los usuarios como al personal encargado de la oficina de registro civil a registrar, agilizar, organizar y gestionar todo tramite documentario en la municipalidad distrital de Platería, con un interfaz de fácil acceso proporcionado al administrador para poder realizar los registros de los clientes, proveedores y productos, haciéndole un seguimiento al cliente.

Con el sistema web de registro civil no genera pérdida de tiempo, seguridad en los datos y descontrol.

Con la investigación se pretende demostrar que el Desarrollo de un Sistema de Registro Civil permitirá a la oficina de Registro Civil ser más eficiente y eficaz en la calidad de los servicios que presta a la población.

1.5. Objetivos de la Investigación

Objetivo General

Desarrollar un Sistema de Registro Civil para Municipalidad Distrital de Platería- 2017.

Objetivos Específicos

- Diseñar el sistema web de administración para la Municipalidad distrital de Platería- 2017.
- Modelar la Base de Datos para el manejo de la información documentario de la Municipalidad distrital de Platería- 2017.
- Desarrollar una interfaz ergonómica para el sistema de administración trámite documentario aplicado tecnologías PHP, AJAX, JAVA SCRIPT, además de MySQL como gestor de base de datos para el sistema.
- Establecer el grado de seguridad del sistema de administración trámite documentario.

CAPÍTULO II

REVISION DE LA LITERATURA

2.1. Marco Teórico

2.1.1. Antecedentes de la Investigación

Tesis Locales

Castillo Barriga (2005) Desarrollo de Sistema de Información Orientado a Objetos para la Gestión del Registro Civil de la Municipalidad Provincial de Chucuito Juli – 2003” La fase de análisis consistió en desarrollar un modelo de lo que debería hacer el sistema, independientemente de la forma en que esté implantado. La fase del diseño consistió en optimizar, refinar y extender los modelos de objetos, dinámico y funcional hasta que fueron suficientemente detallados para la implementación. Sistema de información para la gestión de registro civil (Software) se realizó a través de programación orientada a objetos mediante la herramienta de desarrollo Power Builder versión 8.0. El desarrollo del sistema, siguió fases de desarrollo de un prototipo.

Coaquira Pinto (2015) “Rediseño de Procesos de Negocio Aplicando la Tecnología Workflow para el Proceso de Trámite Documentario de la Unidad de Gestión Educativa Local Puno – 2014”, realizó las pruebas con el objeto de obtener información sobre la estabilidad del sistema con las métricas de puntos de función, para verificar que esta cumpla con los requerimientos de la organización. Finalmente, en la etapa operacional se realizó la instalación del sistema y el entrenamiento necesario a los usuarios del sistema, con la ayuda de las respuestas proporcionadas por los usuarios del sistema podemos afirmar que se logró facilitar un software adecuado para el proceso de trámite documentario que está plasmada.

Flores Paredes & Roque Choque (2015) Desarrollo de un Sistema de registro único y gestión documentaria de los juzgados de paz del distrito judicial de Puno – 2015, La presente tesis tiene por objetivo; Desarrollar e implementar un sistema de registro único y gestión documentaria que acelere el tiempo de búsqueda y emisión de los documentos requeridos de los juzgados de paz del distrito judicial de Puno - 2015. Para el desarrollo e implementación del sistema se utilizó la metodología de desarrollo de software XP. Para el diseño se utilizó el UML, se midió la calidad del software aplicando la ficha de evaluación ISO 9126 a los operadores del sistema.

Tesis Nacionales

Tejada Coaguila (2016) “Prototipo de un Sistema de Flujo de Trabajo para la Universidad Católica de Santa María”, teniendo como conclusión final que: es posible integrar un conjunto de componentes, incluyendo una herramienta de definición de procesos, un software de gestión de flujo de trabajo, una base de datos; que hagan posible la simplificación de los procesos.

Hayamares Huaman (2012) “Implementación de un Sistema de Información para el Área de Registro Civil, en la Municipalidad Distrital de Pueblo Nuevo”, el objetivo de la presente investigación es demostrar que la aplicación del software contribuye a la mejora de los procesos que se realizan en el área de registro civil haciéndolos más rápidos y eficientes.

El estudio es de tipo aplicado, es decir, analiza cómo es y cómo influye el software en la mejora de los procesos del área de registro civil. La hipótesis plantea que el software de información mejora eficientemente el desarrollo de los procesos en el área de registro civil. La muestra de estudio estuvo conformada por la población del distrito de pueblo nuevo y por el personal del área.

Meza Cardama (2012) El siguiente documento consta en la planificación y el monitoreo de la Actas registrales de identidad, mediante el presente plan de desarrollo de software, el análisis del sistema Informático de Tramite Documentario de Registro Civil de la Municipalidad Distrital de Punchana, que permitirá dar un mejor servicio a los usuarios administradores del sistema que administran los datos, así como también

a los usuarios clientes y contribuidores del distrito de Punchana.

Tesis Internacionales

Font Aranda (2013) "Implementación de un Sistema de Gestión Documental en la Universidad Central "Marta Abreu" de las Villas Cuba: Facultad de Ciencias de la Información y de Educación" El objetivo de esta investigación es la implementación de un Sistema de gestión documental en la "Facultad de Ciencias de la Información y de la Educación" de la "Universidad Central "Marta Abreu" de las Villas", (UCLV) Cuba. La metodología empleada es de tipo no experimental, exploratoria-descriptiva y con un enfoque cualitativo, se puede clasificar como investigación Aplicada ya que, a partir de los postulados teóricos, intenta solucionar los problemas detectados, sentando las bases para la aplicación inmediata de los resultados obtenidos y modificar así la realidad descrita.

Blanco Arteaga (2005) "Sistema Automatizado para el Control de Envío y Recepción de la Correspondencia del Banco Provincial de Caracas" el problema está enfocado hacia la comunicación escrita, que establece entre los diferentes departamentos dentro del BBVA de Caracas. Y teniendo como objetivo implantar un sistema de control de correspondencia que facilite el proceso de envío y recepción de remesas y concluye que la realización de éste proyecto, permitió elaborar un sistema automatizado para controlar el envío y recepción de interna de las remesas que se procesan en las instalaciones del centro financiero provincial y su red de oficinas, fue desarrollado como una manera de darle u a solución a las fallas que se presentan al cumplir con esas funciones en forma manual.

Cox SanMiguel (2012) “sistemas de información, en la gestión y control de hojas de seguridad electrónica en la dirección general de registro civil del Ecuador”; La importancia de contar con un sistema de información para la implementación y control de las hojas de seguridad electrónica en la emisión de los servicios de actas, inscripciones, defunciones entre otras, en la Dirección General de Registro Civil del Ecuador, se centra en descubrir enfoques innovadores para cumplir los objetivos de valor de negocio (servicios públicos) y valor para el cliente de la institución. Para aquello, conducirse mediante un proceso de planeación conducirá al desarrollo de estrategias y modelos de gestión y control de información para las nuevas aplicaciones, procesos, productos y servicios que oferta la institución.

2.2. Marco Conceptual

2.2.1. Sistemas de Información

Un sistema de información (SI) es un conjunto de elementos orientados al tratamiento y administración de datos e información, organizados y listos para su uso posterior, generados para cubrir una necesidad o un objetivo. Dichos elementos formarán parte de alguna de las siguientes categorías:

- Personas
- Actividades o técnicas de trabajo
- Datos
- Recursos materiales en general (recursos informáticos y de comunicación, generalmente, aunque no necesariamente).

Todos estos elementos interactúan para procesar los datos (incluidos los procesos manuales y automáticos) y dan lugar a información más elaborada, que se distribuye de la manera más adecuada posible en una determinada organización, en función de sus objetivos. Si bien las existencias de la mayor parte de sistemas de información son de conocimiento público, recientemente se ha revelado que desde finales del siglo XX diversos gobiernos han instaurado sistemas de información para el espionaje de carácter secreto **(Ralph M. & George W., 2010)**.

Habitualmente el término "sistema de información" se usa de manera errónea como sinónimo de sistema de información informático, en parte porque en la mayoría de los casos los recursos materiales de un sistema de información están constituidos casi en su totalidad por sistemas informáticos. Estrictamente hablando, un sistema de información no tiene por qué disponer de dichos recursos (aunque en la práctica esto no suele ocurrir). Se podría decir entonces que los sistemas de información informáticos son una subclase o un subconjunto de los sistemas de información en general.

Figura N° 1 Elementos básicos de un sistema de información



FUENTE: (wikipedia, es.wikipedia.org, s.f.)

2.2.2. Tipos de Sistema de Información

El área de sistemas de información abarca muchas tecnologías complejas, conceptos abstractos de comportamiento y aplicaciones especializadas en incontables áreas de negocio y en las que no son de negocios. Como profesional en los negocios, exactamente no se tiene que absorber todo este conocimiento abstracto, sino recurrir a las partes más necesarias e importantes de los sistemas de información. Para aquello se debe basar en el siguiente esquema (M. Stair & W. Reynolds, 2010).

Figura N° 2 Resumen las áreas principales del conocimiento de los sistemas de información que necesitan los profesionales de los negocios



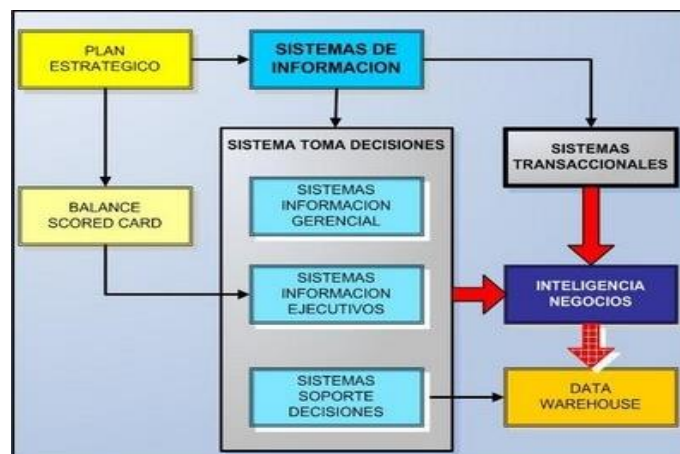
FUENTE: (M. Stair & W. Reynolds, 2010)

El sistema de información se encarga de transformar los datos en información y se define como un "conjunto formal de procesos operando sobre una colección de datos, estructurados de acuerdo con las necesidades de una organización, que recopilan, elaboran y distribuyen la información necesaria para las actividades de dicha organización y para las actividades de dirección y control correspondientes" (Martinez, 2001)

- **Apoyo a sus procesos y operaciones de negocios.**- como trabajador o consumidor a menudo se encuentra con sistemas de información que apoyan los procesos y operaciones de negocio en muchas de las instituciones públicas y privadas. Ejemplo, muchas empresas gubernamentales ha desarrollados un sistema basado en computadora para medir la satisfacción del cliente, entre otros.
- **Apoyo a sus empleados y directivos en la toma de decisiones.**- Los sistemas de información también apoyan o ayudan a los gerentes de las instituciones gubernamentales a la toma de decisiones. Por ejemplo. El Presidente de la republica puede tomar decisiones en función de los indicadores económicos que está presente en la Web o en la plataforma informática del Gobierno Nacional
- **Apoyo a sus estrategias para conseguir una ventaja competitiva.**- lograr una ventaja competitiva en los sectores gubernamentales se requiere de una aplicación innovadora de las tecnologías de la información. Por ejemplo el vínculo de las plataformas informáticas de todas las entidades gubernamentales para informar y ofrecer los servicios de forma paralela a todos los ecuatorianos.

Los tipos de sistemas de información que se ha implementado en el mundo de los negocios de la actualidad pueden clasificarse en diferentes maneras. Es decir, se puede clasificar como sistemas de información operativos o administrativos. En el siguiente cuadro se presentan los siguientes tipos de sistemas de información.

Figura N° 3 Tipos de información



Fuente: O'BRIEN, James A (2001).

2.2.3. Sistemas Web

Los “Sistemas Web” o también conocido como “aplicaciones Web” son aquellos que están creados e instalados no sobre una plataforma o sistemas operativos (Windows, Linux). Sino que se aloja en un servidor en Internet o sobre una intranet (red local). Su aspecto es muy similar a páginas Web que vemos normalmente, pero en realidad los ‘sistemas Web’ tienen funcionalidades muy potentes que brindan respuestas a casos particulares **(Baez, 2012)**.

Los sistemas Web se pueden utilizar en cualquier navegador Web (chrome, firefox, Internet Explorer, etc) sin importar el sistema operativo. Para utilizar las aplicaciones Web no es necesario instalarlas en cada computadora ya que los usuarios se conectan a un servidor donde se aloja el sistema.

Las aplicaciones Web trabajan con bases de datos que permite procesar y mostrar información de forma dinámica para el usuario.

2.2.4. Ingeniería de Software

La ingeniería de software es el establecimiento que comprende todos los aspectos de la traducción del software y uso de principios robustos de la ingeniería a fin de obtener económicamente software que sea fiable y que funciones eficientemente sobre maquinas reales.

Es la aplicación práctica del conocimiento científico en el diseño y construcción de programas de computadoras y la documentación asociada requerida para desarrollar, operar y mantenerlos. Se conoce también como desarrollo de software o producción de software.

La ingeniería de software no solo comprende los procesos técnicos del desarrollo del software sino también con actividades tales como la gestión del proyecto y desarrollo de herramientas, métodos y teorías de apoyo a la producción de software.

2.2.5. Software

Se conoce como software al equipamiento lógico o soporte lógico de una computadora digital; comprende el conjunto de los componentes lógicos necesarios que hacen posible la realización de tareas específicas, en contraposición a los componentes físicos del sistema, llamados hardware. Tales componentes lógicos incluyen, entre muchos otros, aplicaciones informáticas como el procesador de textos, que permite al usuario realizar todas las tareas concernientes a la edición de textos o el software de sistema tal como el sistema (**González, marzo 2005**) operativo, que, básicamente, permite al resto de los programas funcionar

adecuadamente, facilitando la interacción con los componentes físicos y el resto de las aplicaciones, proporcionando también una interfaz para el usuario.

2.2.6. Programación Extrema (XP)

“... centrada en potenciar las relaciones interpersonales como clave para el éxito en desarrollo de software, promoviendo el trabajo en equipo, preocupándose por el aprendizaje de los desarrolladores, y propiciando un buen clima de trabajo.” **(Penades, 2006)**. El principal objeto de XP es la continua interacción entre el cliente y el equipo desarrollador, una comunicación correctamente organizada entre ellos puede asegurar un producto de calidad y la implementación de soluciones certeras.

2.2.7. Fases de Desarrollo Con XP

Fase I: Exploración

Se usan las historias de usuario para recabar la información acerca de cómo debería funcionar el sistema. Además, se deciden las tecnologías a usar para organizar los equipos **(Pérez A., 2011)**.

Fase II: Planificación de la Entrega

Se priorizan las historias de usuario, permitiendo definir un cronograma de entregables, estimando el esfuerzo para el desarrollo de cada sección. Se puede establecer al punto como método de medición, siendo 1 punto una semana de programación, entonces las historias se categorizarán entre 1 y 3 puntos según el esfuerzo **(Pérez A., 2011)**.

Fase III: Iteraciones

Incluye entregas del proyecto en versiones utilizables. La primera entrega puede estar descrita como la arquitectura del sistema, la misma que será utilizada a lo largo del desarrollo **(Pérez A., 2011)**.

Hay que mencionar que el cliente es quien define qué es lo que necesita ser entregado con mayor prioridad. Entonces las interacciones estarán dispuestas por el cliente.

Fase IV: Producción

La fase de producción requiere de pruebas adicionales y revisiones del funcionamiento y rendimiento del sistema. Es posible que se realicen cambios, los mismos que serán documentados para realizarlos en la fase de mantenimiento **(Pérez A., 2011)**.

Fase V: Mantenimiento

Mientras la versión del sistema se encuentra en producción, se pueden empezar a generar los cambios surgidos y al mismo tiempo desarrollando las funciones que se haya decidido en las interacciones anteriores **(Pérez A., 2011)**.

Esta fase puede considerarse, para su ejecución, una vez que el sistema ya ha sido puesto en marcha y está totalmente funcional. Inclusive puede ser vista como un nuevo proyecto de actualización del sistema.

Fase VI: Muerte del Proyecto

Se produce cuando el cliente ha cubierto sus necesidades, con respecto al sistema, tanto en rendimiento como en confiabilidad. Puede también generarse la muerte del proyecto cuando el sistema no satisface las necesidades o el presupuesto para el mantenimiento ha sido eliminado **(Pérez A., 2011)**.

2.2.8. Pruebas Software

El objetivo de la prueba del software es asegurar la apropiada navegación dentro del sistema, ingreso de datos, procesamiento y recuperación apropiada de datos, y la implementación apropiada de las reglas de negocio. Este tipo de pruebas se basan en *técnicas de caja negra*, esto es, verificar el sistema (y sus procesos internos), la interacción con las aplicaciones que lo usan vía GUI y analizar las salidas o resultados **(Pérez A., 2011)**.

En esta prueba se determina que pruebas de sistema (usabilidad, volumen, desempeño, etc.) aseguran que la aplicación alcanzara sus objetivos.

Figura N° 4 Ciclo de pruebas de calidad de software



FUENTE: www.panel.es

2.2.9. Norma de Evaluación ISO/IEC 9126

Se establecen categorías para las cualidades de la calidad externa e interna y calidad en uso del software, teniendo en cuenta estos 7 indicadores (funcionalidad, confiabilidad, utilidad, eficiencia, capacidad de mantenimiento, portabilidad y calidad en uso), que se subdividen a su vez en varios indicadores; estas se pueden medir por métrica interna o externa (Pressman, 2010).

Figura N° 5 Modelo de Calidad Externa e Interna y Calidad de Uso



FUENTE: www.panel.es/

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

Las definiciones se dan para cada característica y subcaracterística de calidad del software que influye en la calidad. Para cada característica y subcaracterísticas, la capacidad del software es determinada por un conjunto de atributos internos que pueden ser medidos. Las características y subcaracterísticas se pueden medir externamente por la capacidad del sistema que contiene el software (**Pressman, 2010**).

Funcionalidad

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

Figura N° 6 Funcionalidad y sus características



FUENTE: www.panel.es/

Característica de funcionalidad

La funcionalidad se divide en 5 criterios:

Adecuación: La capacidad del software para proveer un adecuado conjunto de funciones que cumplan las tareas y objetivos especificados por

el usuario.

Exactitud: La capacidad del software para hacer procesos y entregar los resultados solicitados con precisión o de forma esperada.

Interoperabilidad: La capacidad del software de interactuar con uno o más sistemas específicos.

Seguridad: La capacidad del software para proteger la información y los datos de manera que los usuarios o los sistemas no autorizados no puedan acceder a ellos para realizar operaciones, y la capacidad de aceptar el acceso a los datos de los usuarios o sistemas autorizados

Conformidad de la funcionalidad: La capacidad del software de cumplir los estándares referentes a la funcionalidad.

confiabilidad

La confiabilidad es la capacidad del software para asegurar un nivel de funcionamiento adecuado cuando es utilizando en condiciones específicas. En este caso a la confiabilidad se amplía sostener un nivel especificado de funcionamiento y no una función requerida (**Pressman, 2010**).



FUENTE: www.panel.es/

Característica de Confiabilidad

La confiabilidad se divide en 4 criterios:

Madurez: La capacidad que tiene el software para evitar fallas cuando encuentra errores. Ejemplo, la forma como el software advierte al usuario cuando realiza operaciones en la unidad de diskett vacía, o cuando no encuentra espacio suficiente el disco duro donde esta almacenando los datos (**Pressman, 2010**).

Tolerancia a errores: La capacidad que tiene el software para mantener un nivel de funcionamiento en caso de errores.

Recuperabilidad: La capacidad que tiene el software para restablecer su funcionamiento adecuado y recuperar los datos afectados en el caso de una falla.

Conformidad de la fiabilidad: La capacidad del software de cumplir a los estándares o normas relacionadas a la fiabilidad.

Usabilidad

La usabilidad es la capacidad del software de ser entendido, aprendido, y usado en forma fácil y atractiva. Algunos criterios de funcionalidad, fiabilidad y eficiencia afectan la usabilidad, pero para los propósitos de la ISO/IEC 9126 ellos no clasifican como usabilidad. La usabilidad está determinada por los usuarios finales y los usuarios indirectos del software, dirigidos a todos los ambientes, a la preparación del uso y el resultado obtenido (**Pressman, 2010**).

Figura N° 8 Usabilidad

FUENTE: www.panel.es

Característica de Usabilidad

La usabilidad se divide en 5 criterios:

Entendimiento: La capacidad que tiene el software para permitir al usuario entender si es adecuado, y de una manera fácil como ser utilizado para las tareas y las condiciones particulares de la aplicación. En este criterio se debe tener en cuenta la documentación y de las ayudas que el software entrega.

Aprendizaje: La forma como el software permite al usuario aprender su uso. También es importante considerar la documentación.

Operabilidad: La manera como el software permite al usuario operarlo y controlarlo.

Atracción: La presentación del software debe ser atractiva al usuario. Esto se refiere a las cualidades del software para hacer más agradable al usuario, ejemplo, el diseño gráfico.

Conformidad de uso: La capacidad del software de cumplir los estándares o normas relacionadas a su usabilidad.

eficiencia

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

Figura N° 9 Eficiencia



FUENTE: www.panel.es/

Característica de Eficiencia

La eficiencia se divide en 3 criterios:

Comportamiento de tiempos: Los tiempos adecuados de respuesta y procesamiento, el rendimiento cuando realiza su función en condiciones específicas. Ejemplo, ejecutar el procedimiento más complejo del software y esperar su tiempo de respuesta, realizar la misma función, pero con más cantidad de registros.

Utilización de recursos: La capacidad del software para utilizar cantidades y tipos adecuados de recursos cuando este funciona bajo requerimientos o condiciones establecidas. Ejemplo, los recursos

humanos, el hardware, dispositivos externos.

Conformidad de eficiencia: La capacidad que tiene el software para cumplir con los estándares o convenciones relacionados a la eficiencia.

Capacidad de mantenimiento

La capacidad de mantenimiento es la cualidad que tiene el software para ser modificado. Incluyendo correcciones o mejoras del software, a cambios en el entorno, y especificaciones de requerimientos funcionales (Pressman, 2010).

Figura N° 10 Capacidad de Mantenimiento



FUENTE: www.panel.es/

Característica de Mantenimiento

El mantenimiento se divide en 5 criterios:

Capacidad de ser analizado: La forma como el software permite diagnósticos de deficiencias o causas de fallas, o la identificación de partes modificadas.

Cambiabilidad: La capacidad del software para que la implementación de una modificación se pueda realizar, incluye también codificación, diseño y

documentación de cambios.

Estabilidad: La forma como el software evita efectos inesperados para modificaciones del mismo.

Facilidad de prueba: La forma como el software permite realizar pruebas a las modificaciones sin poner el riesgo los datos

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

Portabilidad

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

Figura N° 11 Portabilidad



FUENTE: www.panel.es/

Característica de portabilidad

La usabilidad se divide en 5 criterios:

Adaptabilidad: Es como el software se adapta a diferentes entornos especificados (hardware o sistemas operativos) sin que implique

reacciones negativas ante el cambio. Incluye la escalabilidad de capacidad interna (Ejemplo: Campos en pantalla, tablas, volúmenes de transacciones, formatos de reporte, etc.).

Facilidad de instalación: La facilidad del software para ser instalado en un entorno específico o por el usuario final.

Coexistencia: La capacidad que tiene el software para coexistir con otro o varios softwares, la forma de compartir recursos comunes con otro software o dispositivo (**Pressman, 2010**).

Reemplazabilidad: La capacidad que tiene el software para ser reemplazado por otro software del mismo tipo, y para el mismo objetivo. Ejemplo, la reemplazabilidad de una nueva versión es importante para el usuario, la propiedad de poder migrar los datos a otro software de diferente proveedor.

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

calidad en uso

Calidad en uso es la calidad del software que el usuario final refleja, la forma como el usuario final logra realizar los procesos con satisfacción, eficiencia y exactitud. La calidad en uso debe asegurar la prueba o revisión de todas las opciones que el usuario trabaja diariamente y los procesos que realiza esporádicamente relacionados con el mismo software (**Pressman, 2010**).

Figura N° 12 Calidad en Uso

FUENTE: www.panel.es

Característica Calidad de uso

La calidad de uso se divide en 4 criterios:

Eficacia: La capacidad del software para permitir a los usuarios finales realizar los procesos con exactitud e integridad.

Productividad: La forma como el software permite a los usuarios emplear cantidades apropiadas de recursos, en relación a la eficacia lograda en un contexto específico de uso. Para una empresa es muy importante que el software no afecte a la productividad del empleado.

Seguridad: Se refiere al que el Software no tenga niveles de riesgo para causar daño a las personas, instituciones, software, propiedad intelectual o entorno. Los riesgos son normalmente el resultado de deficiencias en la funcionalidad (Incluyendo seguridad), fiabilidad, usabilidad o facilidad de mantenimiento (**Pressman, 2010**).

Satisfacción: La satisfacción es la respuesta del usuario a la interacción con el software, e incluye las actitudes hacia el uso del mismo. A continuación, se describe un cuadro donde podemos resumir las

características y cada uno de sus atributos, este cuadro le ayudara a visualizar el proceso de evaluación **(Pressman, 2010)**.

2.2.10. Base de Datos

Una base de datos o banco de datos es un conjunto de datos pertenecientes a un mismo contexto y almacenados sistemáticamente para su posterior uso. En este sentido; una biblioteca puede considerarse una base de datos compuesta en su mayoría por documentos y textos impresos en papel e indexados para su consulta. Actualmente, y debido al desarrollo tecnológico de campos como la informática y la electrónica, la mayoría de las bases de datos están en formato digital, siendo este un componente electrónico, por tanto se ha desarrollado y se ofrece un amplio rango de soluciones al problema del almacenamiento de datos **(Nevado Cabello, 2010)**.

Una base de datos es una colección de datos estructurados según un modelo que refleja las relaciones y restricciones existentes en el mundo real.

Los datos, son compartidos por diferentes usuarios y aplicaciones, y deben mantenerse independientes de estas. Asimismo, los tratamientos que sufran estos datos tendrán que conservar la integridad y seguridad **(Sabana, 2006)**.

2.2.11. Modelado de software con UML

Un modelo es una representación de la realidad en la que se toma en cuenta solo los detalles relevantes con la finalidad de obtener soluciones

de un problema **(Matzukawa, 2002)**

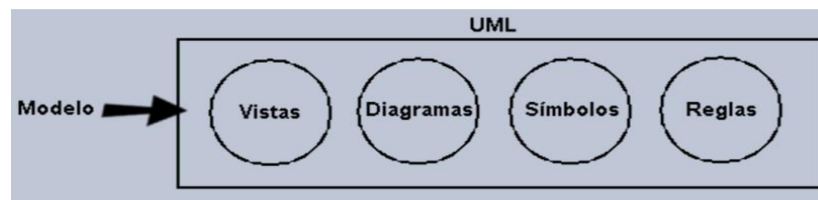
Un modelo proporciona los planos de un sistema. Los modelos pueden estar formados por planos detallados, así como por planos más generales que presentan una visión global del sistema en estudio. Un buen modelo permite cumplir los siguientes objetivos:

- Visualizar como es o como queremos que sea un sistema.
- Especificar la estructura y el comportamiento del sistema.
- Proporcionar plantillas que sirvan como guías en la construcción del sistema.

2.2.12. UML

UML es un lenguaje para hacer modelos y es independiente de los métodos de análisis y diseño. Existen diferencias importantes entre un método y un lenguaje de modelado. Un *método* es una manera explícita de estructurar el pensamiento y las acciones de cada individuo. Además, el método le dice al usuario qué hacer, cómo hacerlo, cuándo hacerlo y por qué hacerlo; mientras que el lenguaje de modelado carece de estas instrucciones. Los métodos contienen modelos y esos modelos son utilizados para describir algo y comunicar los resultados del uso del método **(Pressman, 2010)**.

Figura N° 13 Proceso UML

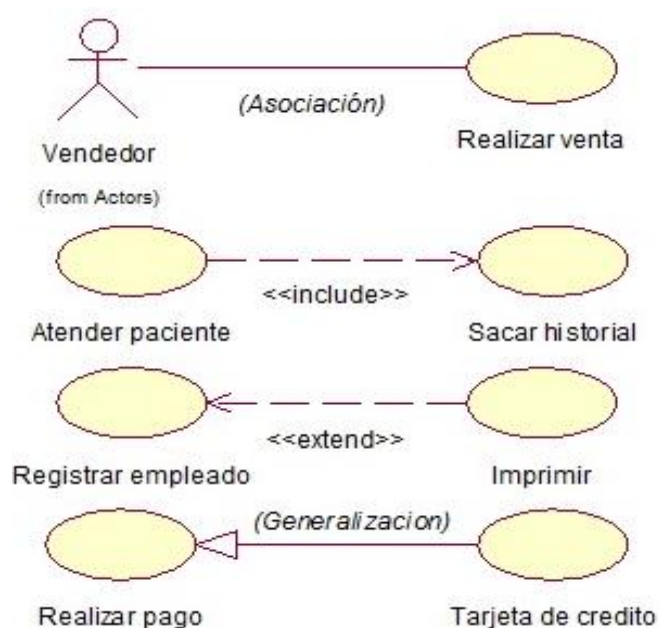
FUENTE: www.panel.es

Vistas: Las vistas muestran diferentes aspectos del sistema modelado. Una vista no es una gráfica, pero sí una abstracción que consiste en un número de diagramas y todos esos diagramas juntos muestran una "fotografía" completa del sistema (**Pressman, 2010**). Las vistas también ligam el lenguaje de modelado a los métodos o procesos elegidos para el desarrollo. Las diferentes vistas que UML tiene son:

- **Vista Use-Case:** Una vista que muestra la funcionalidad del sistema como la perciben los actores externos.
- **Vista Lógica:** Muestra cómo se diseña la funcionalidad dentro del sistema, en términos de la estructura estática y la conducta dinámica del sistema.
- **Vista de Componentes:** Muestra la organización de los componentes de código.
- **Vista Concurrente:** Muestra la concurrencia en el sistema, direccionando los problemas con la comunicación y sincronización que están presentes en un sistema concurrente.
- **Vista de Distribución:** muestra la distribución del sistema en la arquitectura física con computadoras y dispositivos llamados *nodos*.

Diagramas: Los diagramas son las gráficas que describen el contenido de una vista. UML tiene nueve tipos de diagramas que son utilizados en combinación para proveer todas las vistas de un sistema: diagramas de caso de uso, de clases, de objetos, de estados, de secuencia, de colaboración, de actividad, de componentes y de distribución (**Pressman, 2010**).

Figura N° 14 Modelo de Diagrama de Caso de Uso



FUENTE: www.diagramas.com

Símbolos o Elementos de modelo: Los conceptos utilizados en los diagramas son los elementos de modelo que representan conceptos comunes orientados a objetos, tales como clases, objetos y mensajes, y las relaciones entre estos conceptos incluyendo la asociación, dependencia y generalización. Un elemento de modelo es utilizado en varios diagramas diferentes, pero siempre tiene el mismo significado y simbología.

Reglas o Mecanismos generales: Proveen comentarios extras, información o semántica acerca del elemento de modelo; proveen

mecanismos de extensión para adaptar o extender UML a un método, organización o usuario.

2.2.12.1. fases de desarrollo de un sistema

Las fases del desarrollo de sistemas que soporta UML son: *Análisis de requerimientos, Análisis, Diseño, Programación y Pruebas.*

Análisis de Requerimientos

UML tiene casos de uso (use-cases) para capturar los requerimientos del cliente. A través del modelado de casos de uso, los actores externos que tienen interés en el sistema son modelados con la funcionalidad que ellos requieren del sistema (los casos de uso). Los actores y los casos de uso son modelados con relaciones y tienen asociaciones entre ellos o éstas son divididas en jerarquías. Los actores y casos de uso son descritos en un diagrama use-case. Cada use-case es descrito en texto y especifica los requerimientos del cliente: lo que él (o ella) espera del sistema sin considerar la funcionalidad que se implementará. Un análisis de requerimientos puede ser realizado también para procesos de negocios, no solamente para sistemas de software (**Pressman, 2010**)

Análisis

La fase de análisis abarca las abstracciones primarias (clases y objetos) y mecanismos que están presentes en el dominio del problema. Las clases que se modelan son identificadas, con sus relaciones y descritas en un diagrama de clases. Las colaboraciones entre las clases para ejecutar los casos de uso también se consideran en esta fase a través de

los modelos dinámicos en UML. Es importante notar que sólo se consideran clases que están en el dominio del problema (conceptos del mundo real) y todavía no se consideran clases que definen detalles y soluciones en el sistema de software, tales como clases para interfaces de usuario, bases de datos, comunicaciones, concurrencia, etc.

Diseño

En la fase de diseño, el resultado del análisis es expandido a una solución técnica. Se agregan nuevas clases que proveen de la infraestructura técnica: interfaces de usuario, manejo de bases de datos para almacenar objetos en una base de datos, comunicaciones con otros sistemas, etc. Las clases de dominio del problema del análisis son agregadas en esta fase. El diseño resulta en especificaciones detalladas para la fase de programación (**Kendall, 2011**).

Programación

En esta fase las clases del diseño son convertidas a código en un lenguaje de programación orientado a objetos. Cuando se crean los modelos de análisis y diseño en UML.

Pruebas

Normalmente, un sistema es tratado en pruebas de unidades, pruebas de integración, pruebas de sistema, pruebas de aceptación, etc. Las pruebas de integración integran componentes y clases en orden para verificar que se ejecutan como se especificó. Las pruebas de sistema ven al sistema como una "caja negra" y validan que el sistema tenga la

funcionalidad final que le usuario final espera. Las pruebas de aceptación conducidas por el cliente verifican que el sistema satisface los requerimientos y son similares a las pruebas de sistema **(Kendall, 2011)**.

2.2.13. Gestor de base de datos

Un sistema gestor de base de datos (SGBD) es un conjunto de programas que permiten el almacenamiento, modificación y extracción de la información en una base de datos, además de proporcionar herramientas para añadir, borrar, modificar y analizar los datos.

2.2.13.1. MYSQL

Es un sistema administrativo relacional de bases de datos (RDBMS por sus siglas en inglés Relational Database Management System). Este tipo de bases de datos puede ejecutar desde acciones tan básicas, como insertar y borrar registros, actualizar información ó hacer consultas simples, hasta realizar tareas tan complejas (Oppel, 2009).

MySQL se ejecuta en prácticamente todas las plataformas, incluyendo Linux, UNIX y Windows. A pesar de que se puede utilizar en una amplia gama de aplicaciones, MySQL se asocia más con las aplicaciones basadas en la web y la publicación en línea y es un componente importante de una pila empresarial de código abierto llamado LAMP. LAMP es una plataforma de desarrollo web que utiliza Linux como sistema operativo, Apache como servidor web, MySQL como sistema de gestión de base de datos relacional y PHP como lenguaje de programación orientado a objetos (a veces, Perl o Python se utiliza en lugar de PHP) **(Gilfillan, 2003)**.

2.2.13.2. XAMPP

XAMPP es un servidor web de plataforma, software libre, que consiste principalmente en el sistema de gestión de bases de datos MySQL, el servidor web Apache y los intérpretes para lenguajes de script PHP y Perl. El nombre es en realidad un acrónimo: X (para cualquiera de los diferentes sistemas operativos), Apache, MariaDB, PHP, Perl. A partir de la versión 5.6.15, XAMPP cambió la base de datos MySQL por MariaDB, fork de MySQL con licencia GPL (Baiker, 2016).

El programa se distribuye con la licencia GNU y actúa como un servidor web libre, fácil de usar y capaz de interpretar páginas dinámicas. A esta fecha, XAMPP está disponible para Microsoft Windows, GNU/Linux, Solaris y Mac OS X.

2.2.13.3. Workbench

Workbench es una herramienta visual de diseño de bases de datos que integra desarrollo de software, Administración de bases de datos, diseño de bases de datos, creación y mantenimiento para el sistema de base de datos MySQL. Es el sucesor de DBDesigner 4

2.2.13.4. PhpAdmin

PhpMyAdmin es una herramienta de software libre escrito en PHP para ocuparse de la administración de MySQL sobre la Red en el ámbito mundial. PhpMyAdmin es compatible con una amplia gama de operaciones con MySQL. La mayoría de las operaciones de uso frecuente son compatibles con la interfaz de usuario (administrar bases de datos, tablas,

campos, relaciones, índices, usuarios, permisos, etc.); mientras se tiene la capacidad de ejecutar cualquier sentencia SQL directamente **(Welling, Luke ; Thomson, Laura, 2009)**.

2.2.13.5. Modelo Entidad – Relación

El Modelo de Entidad Relación es un modelo de datos basado en una percepción del mundo real que consiste en un conjunto de objetos básicos llamados entidades y relaciones entre estos objetos, implementándose en forma gráfica a través del Diagrama Entidad Relación **(Nevado Cabello, 2010)**.

2.2.13.6. Clave Principal

Se denomina Clave principal o primaria al atributo o conjunto mínimo de atributos (uno o más campos) que permiten identificar en forma única cada instancia de la entidad, es decir, a cada registro de la tabla. En un principio se puede identificar más de un atributo que cumpla las condiciones para ser clave, los mismos se denominan Claves candidatas.

Si la clave primaria se determina mediante un solo atributo de la entidad, entonces se dice que la misma es una Clave simple. En caso de estar conformada por más de un atributo, la misma se conoce como Clave compuesta **(Nevado Cabello, 2010)**.

La Clave foránea (también llamada externa o secundaria) es un atributo que es clave primaria en otra entidad con la cual se relaciona.

2.2.13.7. Tipos de Relaciones

- **Relación Uno a Uno:** Cuando un registro de una tabla sólo puede estar relacionado con un único registro de la otra tabla y viceversa.
- **Relación Uno a Muchos:** Cuando un registro de una tabla (tabla secundaria) sólo puede estar relacionado con un único registro de la otra tabla (tabla principal) y un registro de la tabla principal puede tener más de un registro relacionado en la tabla secundaria. En este caso la clave foránea se ubica en la tabla secundaria.
- **Relación Muchos a Muchos:** Cuando un registro de una tabla puede estar relacionado con más de un registro de la otra tabla y viceversa. En este caso las dos tablas no pueden estar relacionadas directamente, se tiene que añadir una tabla entre las dos (Tabla débil o de vinculación) que incluya los pares de valores relacionados entre sí (**Nevado Cabello,2010**).

2.2.13.8. Estructura Del Sistema

Es un procesador de textos orientado para escribir código fuente de aplicaciones en general en lenguajes de programación. Generalmente los editores de código soportan varios lenguajes y son capaces de abrir varios archivos a la vez, resaltar su sintaxis y ofrecer ayudas contextuales a la hora de escribir o visualizar el código fuente de las aplicaciones

2.2.13.9. HTML5 (Hyper Text Markup Language)

El HTML no es más que una aplicación del SGML (Standard

Generalized Markup Language), un sistema para definir tipos de documentos estructurados y lenguajes de marcas para representar esos mismos documentos.

El HTML, Hyper Text Markup Language es el lenguaje de marcas de texto utilizado normalmente en la WWW (World Wide Web). Fue creado en 1986 por el físico nuclear Tim Berners-Lee; el cual tomó dos herramientas preexistentes: el concepto de Hipertexto (conocido también como link o ancla) el cual permite conectar dos elementos entre sí y el SGML (Lenguaje Estándar de Marcación General) el cual sirve para colocar etiquetas o marcas en un texto que indique como debe verse.

HTML no es propiamente un lenguaje de programación como C++, Visual Basic, etc., sino un sistema de etiquetas. HTML no presenta ningún compilador, por lo tanto, algún error de sintaxis que se presente este no lo detectará y se visualizara en la forma como éste lo entienda. **(Daniel, 2005).**

2.2.13.10. *SUBLIME TEXT*

Es un editor de código multiplataforma, ligero y con pocas concesiones a las florituras. Es una herramienta concebida para programar sin distracciones. Su interfaz de color oscuro y la riqueza de coloreado de la sintaxis, centra nuestra atención completamente. Permite tener varios documentos abiertos mediante pestañas, e incluso emplear varios paneles para aquellos que utilicen más de un monitor. Dispone de modo de pantalla completa, para aprovechar al máximo el espacio visual disponible de la pantalla.

Lenguaje de Programación

Un lenguaje de programación es un lenguaje formal diseñado para realizar procesos que pueden ser llevados a cabo por máquinas como las computadoras. Pueden usarse para crear programas que controlen el comportamiento físico y lógico de una máquina, para expresar algoritmos con precisión, o como modo de comunicación humana **(Fresno F,2012)**.

Está formado por un conjunto de símbolos y reglas sintácticas y semánticas que definen su estructura y el significado de sus elementos y expresiones. Al proceso por el cual se escribe, se prueba, se depura, se compila (de ser necesario) y se mantiene el código fuente de un programa informático se le llama programación.

También la palabra programación se define como el proceso de creación de un programa de computadora, mediante la aplicación de procedimientos lógicos, a través de los siguientes pasos:

- El desarrollo lógico del programa para resolver un problema en particular.
- Escritura de la lógica del programa empleando un lenguaje de programación específico (codificación del programa).
- Ensamblaje o compilación del programa hasta convertirlo en lenguaje de máquina.
- Prueba y depuración del programa.

Tabla N° 1 Lenguajes de Programación Actuales

LENGUAJE	PRINCIPAL ÁREA DE APLICACIÓN	COMPILADO/ INTERPRETADO
ADA	Tiempo real	Lenguaje compilado
BASIC	Programación para fines educativos	Lenguaje interpretado
C	Programación de sistema	Lenguaje compilado
C++	Programación de sistema orientado a objeto	Lenguaje compilado
Cobol	Administración	Lenguaje compilado
Fortran	Cálculo	Lenguaje compilado
Java	Programación orientada a Internet	Lenguaje intermediario
MATLAB	Cálculos matemáticos	Lenguaje interpretado
Cálculos matemáticos	Cálculos matemáticos	Lenguaje interpretado
LISP	Inteligencia artificial	Lenguaje intermediario
Pascal	Educación	Lenguaje compilado
PHP	Desarrollo de sitios web dinámicos	Lenguaje interpretado
Inteligencia artificial	Inteligencia artificial	Lenguaje interpretado
Perl	Procesamiento de cadenas de caracteres	Lenguaje interpretado

FUENTE: Elaboración Propia

2.2.13.11. CSS 3

Las hojas de estilos en cascada (CSS) permiten dar a los documentos HTML una apariencia atractiva y coherente. Al vincular muchas páginas Web a la misma hoja de estilos externa. Puede definir un aspecto y un diseño coherentes para todo un sitio Web.

Cualquier explorador Web compatible con HTML 4.0 o posterior admitirá la mayoría de los atributos de estilos CSS. En el desarrollo de la aplicación se usó estilos CSS para definir la ubicación y la apariencia de los elementos, del texto de las páginas HTML y los formularios Web Forms.

Las hojas de estilos en cascada usadas, contienen definiciones de estilos que se aplican a los elementos de los documentos HTML. Los estilos CSS definen la forma de mostrar los elementos y su posición en la página. En lugar de asignar individualmente atributos a cada elemento de la página, se creó una regla general que aplica atributos específicos.

Siempre que el explorador Web encuentre una instancia de un elemento o un elemento asignado a un cierto estilo CLASS, se aplicara la configuración definida en la hoja de estilo **(Gauchat, 2012)**.

2.2.13.12. PHP (Personal Home Page)

El sistema fue desarrollado originalmente en el año 1994 por Rasmus Lerdorf como un CGI escrito en C que permitía la interpretación de un número limitado de comandos.

El sistema fue denominado Personal Home Page Tools y adquirió relativo éxito gracias a que otras personas pidieron a Rasmus que les permitiese utilizar sus programas en sus propias páginas. Dada la aceptación del primer PHP su creador diseñó un sistema para procesar formularios al que le atribuyó el nombre de FI (FormInterpreter) y el conjunto de estas dos herramientas, sería la primera versión compacta del lenguaje: PHP/FI **(Welling, Luke ; Thomson, Laura, 2009)**.

PHP es uno de los lenguajes más populares dentro del software libre en la programación para Web, su diversidad y soporte han sido fundamentales en lograr esta popularidad. Principales características de PHP:

- Acceso a gran número de gestores de bases de datos (Adabas D, dbm,dBase, filePro, Hyperwave, Informix, Internase, LDAP, Microsoft SQL server, mSQL, MySQL, ODBC, Oracle, PostgreSQL, Solid y Sybase).
- Envío de correo con SMTP.
- Acceso a servidores de FTP.
- Acceso a SNMP para gestión de redes y equipos.
- Generación dinámica de gráficos y documentos PDF.
- Análisis de documentos XML.
- Generación de datos en WDDX (Intercambio Web de Datos distribuidos).
- Soporte de hilos de ejecución a partir de PHP 4

2.2.13.13. Javascript

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

Una página web dinámica es aquella que incorpora efectos de texto, animaciones, acciones que se activan al pulsar botones y ventanas con mensajes de aviso al usuario **(Gauchat, 2012)**.

Técnicamente, JavaScript es un lenguaje de programación interpretado, por lo que no es necesario compilar los programas para ejecutarlos. En otras palabras, los programas escritos con JavaScript se pueden probar directamente en cualquier navegador sin necesidad de procesos intermedios. **(Eguiluz Perez, 2008)** .

- **AJAX**

AJAX, acrónimo de Asynchronous JavaScript And XML (JavaScript asíncrono y XML), es una técnica de desarrollo web para crear aplicaciones interactivas. Estas aplicaciones se ejecutan en el cliente, es decir, en el navegador de los usuarios mientras se mantiene la comunicación asíncrona con el servidor en segundo plano. De esta forma es posible realizar cambios sobre las páginas sin necesidad de recargarlas, mejorando la interactividad, velocidad y usabilidad en las aplicaciones **(Gauchat, 2012)**.

Ajax es una tecnología asíncrona, en el sentido de que los datos adicionales se solicitan al servidor y se cargan en segundo plano sin interferir con la visualización ni el comportamiento de la página. JavaScript es el lenguaje interpretado en el que normalmente se efectúan las funciones de llamada de Ajax mientras que el acceso a los datos se realiza mediante XMLHttpRequest, objeto disponible en los navegadores actuales. En cualquier caso, no es necesario que el contenido asíncrono esté formateado en XML.

Ajax es una técnica válida para múltiples plataformas y utilizable en muchos sistemas operativos y navegadores, dado que está basado en estándares abiertos como JavaScript y Document Object Model (DOM).

Las tecnologías que forman AJAX son:

- ✓ XHTML y CSS, para crear una presentación basada en estándares.
- ✓ DOM, para la interacción y manipulación dinámica de la presentación.

- ✓ XML, XSLT y JSON, para el intercambio y la manipulación de información. XMLHttpRequest, para el intercambio asíncrono de información.
- ✓ JavaScript, para unir todas las demás tecnologías.

La característica fundamental de AJAX es permitir actualizar parte de una página con información que se encuentra en el servidor sin tener que refrescar completamente la página. De modo similar podemos enviar información al servidor **(Gauchat, 2012)**.

- **JQUERY**

Es una biblioteca de JavaScript, creada inicialmente por John Resig, que permite simplificar la manera de interactuar con los documentos HTML, manipular el árbol DOM, manejar eventos, desarrollar animaciones y agregar interacción con la técnica AJAX a páginas web. Fue presentada el 14 de enero de 2006 en el BarCamp NYC. JQuery es la biblioteca de JavaScript más utilizada **(Gauchat, 2012)**.

JQuery es software libre y de código abierto, posee un doble licenciamiento bajo la Licencia MIT y la Licencia Pública General de GNU v2, permitiendo su uso en proyectos libres y privados. jQuery, al igual que otras bibliotecas, ofrece una serie de funcionalidades basadas en JavaScript que de otra manera requerirían de mucho más código, es decir, con las funciones propias de esta biblioteca se logran grandes resultados en menos tiempo y espacio **(Chaffer & Swedberg, 2009)**.

2.2.14. framework

El concepto framework se emplea en muchos ámbitos del desarrollo de sistemas software, no solo en el ámbito de aplicaciones Web. Podemos encontrar frameworks para el desarrollo de aplicaciones médicas, de visión por computador, para el desarrollo de juegos, y para cualquier ámbito que pueda ocurrírse nos. En general, con el término framework, nos estamos refiriendo a una estructura software compuesta de componentes personalizables e intercambiables para el desarrollo de una aplicación. En otras palabras, un framework se puede considerar como una aplicación genérica incompleta y configurable a la que podemos añadirle las últimas piezas para construir una aplicación concreta (J.Gutierrez, 2010).

2.2.14.1. bootstrap

Bootstrap, es un framework originalmente creado por Twitter, que permite crear interfaces web con CSS y JavaScript, cuya particularidad es la de adaptar la interfaz del sitio web al tamaño del dispositivo en que se visualice. Es decir, el sitio web se adapta automáticamente al tamaño de una PC, una Tablet u otro dispositivo. Esta técnica de diseño y desarrollo se conoce como “**responsive design**” o diseño adaptativo (Solis, 2015).

El beneficio de usar responsive design en un sitio web, es principalmente que el sitio web se adapta automáticamente al dispositivo desde donde se acceda. Lo que se usa con más frecuencia, y que a mi opinión personal me gusta más, es el uso de *media queries*, que es un módulo de CSS3 que permite la representación de contenido para adaptarse a condiciones como la resolución de la pantalla y si trabajás las

dimensiones de tu contenido en porcentajes, puedes tener una web muy fluida capaz de adaptarse a casi cualquier tamaño de forma automática.

Pero si no quieres nada que ver con los media queries, otra muy buena opción es el uso del framework de Bootstrap, que como te dije te ayudará a desarrollar tus sitios adaptativos.

Aun ofreciendo todas las posibilidades que ofrece Bootstrap a la hora de crear interfaces web, los diseños creados con Bootstrap son simples, limpios e intuitivos, esto les da agilidad a la hora de cargar y al adaptarse a otros dispositivos. El Framework trae varios elementos con estilos predefinidos fáciles de configurar: Botones, Menús desplegables, Formularios incluyendo todos sus elementos e integración jQuery para ofrecer ventanas y tooltips dinámicos. **(Solis, 2015)**

Ventajas de usar Bootstrap

La más genérica es que permite simplificar el proceso de maquetación, sirviéndonos de guía para aplicar las buenas prácticas y los diferentes estándares. Aquí van unos cuantos pros más:

- Puedes tener una web bien organizada de forma visual rápidamente: la curva de aprendizaje hace que su manejo sea asequible y rápido si ya sabes maquetar.
- Permite utilizar muchos elementos web: desde iconos a desplegables, combinando HTML5, CSS y Javascript.
- Sea lo que sea que creemos, el diseño será adaptable, no importa el dispositivo, la escala o resolución.

- El grid system: maquetar por columnas nunca fue tan fácil. Además, son muy configurables.
- Se integra muy bien con las principales librerías Javascript.
- El haber sido creado por Twitter nos da ciertas garantías: está muy pensado y hay mucho trabajo ya hecho. Por lo tanto, hay una comunidad muy activa creando, arreglando cosas, ofreciendo plugins y mucho más.
- Cuenta con implementaciones externas para WordPress, Drupal, etc.
- Nos permite usar Less, para enriquecer aún más los estilos de la web.

2.2.14.2. codeigniter

Codeigniter es un programa o aplicación web desarrollada en PHP para la creación de cualquier tipo de aplicación web bajo PHP. Es un producto de código libre, libre de uso para cualquier aplicación.

Como cualquier otro framework, Codeigniter contiene una serie de librerías que sirven para el desarrollo de aplicaciones web y además propone una manera de desarrollarlas que debemos seguir para obtener provecho de la aplicación. Esto es, marca una manera específica de codificar las páginas web y clasificar sus diferentes scripts, que sirve para que el código esté organizado y sea más fácil de crear y mantener. Codeigniter implementa el proceso de desarrollo llamado Model View Controller (MVC), que es un estándar de programación de aplicaciones, utilizado tanto para hacer sitios web como programas tradicionales. Este sistema tiene sus características, que veremos en artículos siguientes **(Solis, 2015)**.

2.3. Definición de Términos Básicos

- **Administrador**

Es la persona o equipo de personas profesionales responsables del control y manejo del sistema de base de datos, generalmente tienen experiencia en DBMS, diseño de bases de datos, sistemas operativos, comunicación de datos, hardware y programación.

- **Automatización**

La automatización de tareas es, en Informática, el conjunto de métodos que sirven para realizar tareas repetitivas en un ordenador.

- **Campo**

Es un conjunto de datos del mismo tipo, que es representado por una columna.

- **Jurado**

Tiene la función de calificar (valorar algo o atribuir cierta cualidad o denominación).

- **Participante**

Participante es ampliamente usada en nuestro idioma para dar cuenta de aquel individuo u organización que participa en alguna actividad, acción, tarea, entre otras alternativas

- **Entidad**

La entidad es cualquier objeto, real o abstracto, que existe en un contexto determinado o puede llegar a existir y del cual deseamos guardar información.

- **Implementación**

En desarrollo de Sistemas Informáticos, la implementación es la etapa donde efectivamente se programa el sistema.

- **Información**

Es un conjunto de datos que tienen un sentido semántico y que nos permite deducir la incertidumbre y que aumenta el conocimiento de algo.

- **Informática**

Ciencia del tratamiento automático de la información mediante un computador (llamado también ordenador o computadora).

- **Interfaz de Sistema**

Son las especificaciones funcionales del sistema, los cuales son representados mediante pantallas y/o menús, que permitirán al usuario interactuar con el sistema.

- **Internet**

Es una red internacional de redes de computadoras, que permite compartir información con gran parte del mundo.

- **página web**

Es un documento electrónico que contiene información específica de un tema en particular y que es almacenado en algún sistema de cómputo que se encuentre conectado a la red mundial de información denominada Internet, de tal forma que este documento pueda ser consultado por cualesquiera personas que se conecte a esta red mundial de comunicaciones y que cuente con los permisos apropiados para hacerlo.

- **Servidor**

Es cualquier recurso de cómputo dedicado a responder los requerimientos del cliente.

- **Sistema**

Conjunto de componentes interrelacionados e interactuantes para llevar a cabo una misión conjunta. Admite ciertos elementos de entrada y produce ciertos elementos de salida en un proceso organizado.

- **Sitio web**

Es un conjunto de archivos, principalmente HTML e imágenes, que constituyen el contenido al que el navegador tiene acceso.

2.4. Municipalidad distrital de Platería

El Distrito peruano de Platería es uno de los 15 distritos de la Provincia de Puno en el Departamento de Puno, bajo la administración del Gobierno regional de Puno, Perú.

La villa de Plateria es el centro urbano más importante a nivel de distrito, cuya tipología es administrativa, de servicios básicos, financiera, turística y cultural.

Demografía

La población estimada en el año 2007 es de 8, 268 habitantes **(INEI, 2007)**.

Etimología

Plateria: Se remonta de versiones de tiempos inmemoriales de Coloniaje donde los Españoles Explotaron diversas minas como las de Laykakota, Tee Kollo y otros donde establecieron una fundición en Plateria a donde llegaron los minerales, como el oro, cobre, plata y fabricar de estos valiosas minerales alhajas y objetos de orfebrería, el lugar denominado Pila Patja son los vestigios de esta existencia donde se encontraron restos que afirman esta versión, en tanto, esta teoría más aproximada originaria la denominación de Platería que hoy surge como una tierra antigua, brava de contornos prominentes, con matices de viento silban entre las cumbres andinas bajo la mirada altiva del hombre Lupaka aymara.

Ubicación

Ubicado en el altiplano a una altura de 3 826 msnm, muy cerca del Lago Titicaca. Actualmente tiene una extensión de 240,63 km².

Figura N° 15 Ubicación del Distrito de Plateria



Fuente: (wikipedia, es.wikipedia.org, s.f.)

CAPITULO III

MATERIALES Y METODOS

3.1. Tipo de investigación

El tipo de estudio que se desarrollara en la investigación es descriptivo, porque está dirigido a determinar cómo es, cómo está la situación de las variables o estudios en una población concreta, la frecuencia con que ocurre el hecho, en quiénes, dónde y cuándo se estará presentando y se tratara de obtener toda la información necesaria del área de registro civil de la municipalidad distrital de Plateria. Esta investigación brindará las bases cognoscitivas para otros estudios.

El método utilizado en esta investigación será el método probabilístico ya que el universo en estudio es finito.

Tabla N° 2 Población de usuario área de registro civil (Mes de enero 2018)

Usuarios	Número de usuarios
Partidas de nacimiento	31
Partidas de matrimonio	1
Partidas de defunciones	8
Población total	40

FUENTE: Elaboración propia

3.2. Población y Muestra del Estudio

Población

La población estuvo definida por todos los usuarios de registro civil que son un promedio de 40 personas atendidas por mes; es por esta razón que el estudio se realizó, entre varones y mujeres que se encuentran registrados en las actas de nacimientos, defunciones y matrimonios de la municipalidad distrital de Plateria

Muestra

Se tomó el método muestral no probabilístico a criterio del investigador en este sentido la muestra está conformado por 33 registros de actas de nacimientos, 2 defunciones y 1 de matrimonio.

Calculo de tamaño de la muestra

$$n = \frac{N * Z_{\alpha/2}^2 * p * q}{N * E^2 + Z_{\alpha/2}^2 * p * q}$$

Donde:

n : Tamaño de la muestra

N : Tamaño de la población

$Z_{\alpha/2}^2$: Valor de Z correspondiente al nivel de confianza.

E : Error máximo permisible (E = 0.05).

p : Probabilidad de éxito

q : Probabilidad de fracaso

$$n = \frac{40 \cdot 1.96^2 \cdot 0.5 \cdot 0.5}{39 \cdot 0.05^2 - 1.96^2 \cdot 0.5 \cdot 0.5}$$

$$n = 36.22$$

$$n \cong 36$$

3.3. Operacionalización de Variables

Tabla N° 3 Operacionalización de variables independientes

Variable Independiente	Dimensión	Indicadores	Escala o Medición
Sistema web	Diagrama UML	Diseño de la interface del sistema	- Muy Bueno - Bueno - Regular
		Amigabilidad del interface del sistema	- Amigable - Poco Amigable - No es Amigable
		Accesibilidad al sistema	- Fácil - Medianamente fácil - Deficil
		Accesibilidad al sistema	- Eficiente - Medianamente Eficiente - Deficiente
		Accesibilidad al sistema	- Confiable - Poco Confiable - No es Confiable

FUENTE: Elaboración propia

Tabla N° 4 Operacionalización de variables dependientes

Variable Dependiente	Dimención	Indicadores	Escala o Medición
Mejora en la Atención en el Tiempo	Reduce el tiempo para la atención en los trámitantes	Tiempo de Atención	A. Más de 5 minutos.
Trámitantes de los de las actas de nacimientos, matrmonio y defunción			B. 4 minutos.
			C. 3 minutos.
			D. Menos de 2 minutos

FUENTE: Elaboración propia

3.4. Método de Recopilación de Datos

La recolección de datos para el presente trabajo de investigación se obtuvo a través de una encuesta a los usuarios del registro civil y donde tiempo X_1 representa la muestra que se obtuvo mediante una encuesta aplicada en el mes de enero del 2018; X_2 representa la muestra que se obtuvo mediante una encuesta aplicada en el mes de marzo del 2018 vale decir después del desarrollo del sistema.

Figura N° 16 Diseño de investigación



FUENTE: Elaboración propia

La información que se necesita fue obtenida por medio de encuestas, las cuales fueron divididas al grupo constituido por la muestra

para este proyecto de investigación. El instrumento que se utilizó en la investigación es el cuestionario, el cual es un documento en donde están consignadas una serie de preguntas que están dirigidas a obtener información específica, el diseño de la encuesta es:

- Encuesta de satisfacción del usuario

3.4.1. Prueba de Hipótesis

La contrastación de la hipótesis se realizó mediante la prueba de la media poblacional, que nos permite aceptar o rechazar la hipótesis. Para esto se realizó una prueba por cada indicador las cuales se emplean las siguientes afirmaciones:

$H_0: \mu_x \leq \mu_y$: Desarrollo de un Sistema de Registro Civil no reduce el tiempo de demora de servicio de atención hacia las unidades orgánicas de la municipalidad del distrito de Plateria.

$H_a: \mu_x > \mu_y$: Desarrollo de un Sistema de Registro Civil reduce el tiempo de demora de servicio de atención hacia las unidades orgánicas de la municipalidad del distrito de Plateria.

i. Nivel de significancia

El nivel de significancia (α) escogido para la prueba de hipótesis será de 5%. Es decir $\alpha = 0.05$ (Nivel de significancia)

ii. Prueba estadística

$$Z_c = \frac{\bar{D}}{s_D / \sqrt{n}} \sim N(0; 1)$$

$$\bar{D} = \frac{\sum_{i=1}^n D_i}{n} ; \quad D_i = x_1 - x_2 ; \quad S_D^2 = \frac{\sum_{i=1}^n (D_i - \bar{D})^2}{n - 1}$$

Donde:

Z_c : Z calculada.

\bar{D} : Promedio diferencial de la diferencia de los resultados después y antes de la implementación del sistema.

n : Tamaño de la muestra del grupo de estudio.

S_D : Desviación estándar de la diferencia (antes y después).

X_1 : Tiempo de atención antes del desarrollo del sistema.

X_2 : Tiempo de atención después del desarrollo del sistema.

$$\bar{D} = 9.74 ; \quad S_D = 10.53 ; \quad Z_c = \frac{9.72}{10.83 / \sqrt{39}} = 5.30$$

$$Z_c = 5.30$$

iii. Regla de decisión

Si $Z_c = 5.30$ y $Z_t = 1.645$ entonces $Z_c > Z_t$ se rechaza la H_0 y se acepta la H_a .

iv. Conclusión

Se concluye que se rechaza la hipótesis nula (H_0) y se acepta la hipótesis alterna (H_a), por lo tanto se prueba la validez de la hipótesis con un nivel de error de 5% (=0.05) siendo una solución al problema en cuanto

al desarrollo del sistema Registro Civil para la municipalidad distrital de Plateria, se mejora el tiempo de atención a los usuarios del registro civil.

3.5. Arquitectura de Software

La estructuración del sistema Registro Civil para la municipalidad distrital de Plateria, idealmente, se creó en etapas del desarrollo. Esta estructuración representó un diseño de alto nivel del sistema que tuvo dos propósitos primarios: satisfacer los atributos de calidad (desempeño, seguridad, modificabilidad), y servir como guía en el desarrollo. Al igual que en la ingeniería civil, las decisiones críticas relativas al diseño general de un sistema de software complejo deben de hacerse desde un principio. El no crear este diseño desde etapas tempranas del desarrollo puede limitar severamente el que el producto final satisfaga las necesidades de los usuarios de la municipalidad.

3.5.1. Ciclo de Desarrollo de la Arquitectura

Dentro de un proyecto de desarrollo, e independientemente de la metodología que se utilice, se puede hablar de “desarrollo de la arquitectura de software”. Este desarrollo, que precede a la construcción del sistema, está dividido en las siguientes etapas: requerimientos, diseño, documentación y evaluación.

A continuación, se describen dichas etapas.

Requerimientos. La etapa de requerimientos se enfoca en la captura, documentación y priorización de requerimientos que influyen la arquitectura. Como se mencionó anteriormente, los atributos de calidad

juegan un papel preponderante dentro de estos requerimientos, así que esta etapa hace énfasis en ellos

Diseño. La etapa de diseño es la etapa central en relación con la arquitectura y probablemente la más compleja. Durante esta etapa se definen las estructuras que componen la arquitectura. La creación de estas estructuras se hace en base a patrones de diseño, tácticas de diseño y elecciones tecnológicas.

Documentación. Una vez creado el diseño de la arquitectura, es necesario poder comunicarlo a otros involucrados dentro del desarrollo. La comunicación exitosa del diseño muchas veces depende de que dicho diseño sea documentado de forma apropiada. La documentación de una arquitectura involucra la representación de varias de sus estructuras que son representadas a través de distintas vistas

Evaluación. Dado que la arquitectura de software juega un papel crítico en el desarrollo, es conveniente evaluar el diseño una vez que este ha sido documentado con el fin de identificar posibles problemas y riesgos. La ventaja de evaluar el diseño es que es una actividad que se puede realizar de manera temprana (aún antes de codificar), y que el costo de corrección de los defectos identificados a través de la evaluación es mucho menor al costo que tendría el corregir estos defectos una vez que el sistema ha sido construido.

El rol del arquitecto

Las actividades descritas anteriormente requieren de habilidades

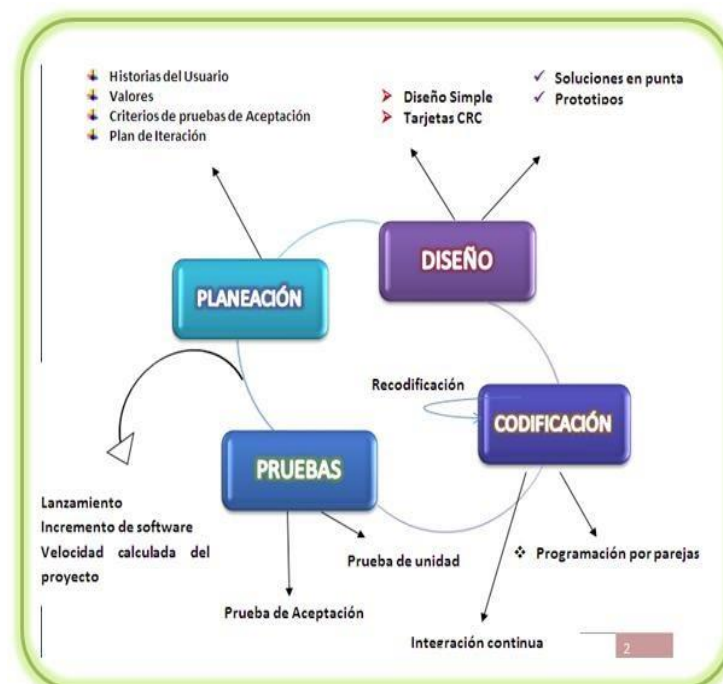
particulares que son la responsabilidad del arquitecto de software. El arquitecto es un líder técnico que debe conocer los principios relacionados con la arquitectura de software, tener un amplio conocimiento respecto a la tecnología, y tener excelentes habilidades de comunicación escrita y oral (Cervantes, 2009)

3.6. Desarrollo del Sistema

A. Metodología de desarrollo

Para el desarrollo del proyecto se utilizó la metodología ágil, metodología programación extrema (XP), pues esta se adapta a las necesidades del proyecto, de poner énfasis para el producto final, además de su flexibilidad y principalmente por su agilidad en el desarrollo de las aplicaciones.

Figura N° 17 Programación Extrema XP



FUENTE: (wikipedia, es.wikipedia.org, s.f.)

B. Programación extrema

➤ **Las historias del usuario**

Para esta etapa el personal que labora describió brevemente las características para su requerimiento del sistema que debe poseer; las funcionalidades que debe cumplir como mínimo. El objetivo de esta etapa es proporcionar los requisitos del sistema y también se llevó a cabo el modelado sin duda utilizando como herramienta el software UML Enterprise Architect versión 3.1.1.

➤ **Roles**

- El cliente
- Tutor/Entrenador (coach)
- Encargado de seguimiento (Tracker)
- Encargado de pruebas (Tester)
- Consultor
- Gestor (Big Boss)

➤ **Ciclo de desarrollo XP**

Para el ciclo de desarrollo del sistema se tuvo en cuenta los siguientes pasos:

1. El usuario fue quien redactó las historias que deben implementarse en cada módulo y a la vez establecido las prioridades del caso.
2. El programador estimó el esfuerzo necesario para su implementación de

la primera propuesta del usuario.

3. El programador desarrollo las historias de prioridad o de mayor riesgo.

La condición del sistema se realizó con el uso del lenguaje de programación como PHP, MySQL, JavaScript; CSS3 y Bootstrap.

4. Se realizó diversas pruebas a cada módulo implementado ara asegurar el ciclo de desarrollo en el proceso anterior, se retomó al primer proceso y así sucesivamente con las demás historias hasta concluirse con el desarrollo del sistema.

➤ **Faces del desarrollo XP**

- **Exploración:** En esta fase se plantearon las historias del usuario de mayor interés, se realizaron los diagramas de UML con la finalidad de elaborar la primera entrega del producto
- **Planificación de la entrega:** En esta fase se estableció la prioridad de cada historia de usuario, ayudando a la estimación del esfuerzo y la planificación adecuada junto con el cliente y la base al primer prototipo de acuerdo a un cronograma de entrega establecido.
- **Iteraciones:** En esta fase se consideró el número de iteraciones necesarias sobre el sistema antes de ser entregado, sobre un plan de entrega defindo.
- **Producción:** Esta fase fuel la más importante, ya que se realizó la codificación, las pruebas, la revisión del requerimiento del sistema y la forma de decisiones en cuanto a la inclusión de nuevas

características sobre la iteración actual.

- **Mantenimiento:** En esta fase de desarrollo, se mantuvo el sistema en funcionamiento al mismo tiempo se producían nuevas iteraciones de forma paralela, mediante tareas de soporte y cliente.
- **Muerte del proyecto:** Esta fase es cuando el cliente ya no tenía más historias para ser incluidas en el sistema, puesto que el proyecto ya ha sido concluido en su totalidad y se ha logrado satisfacer sus necesidades de rendimiento y confiabilidad.

➤ **Practicas XP**

- **La planificación:** En esta práctica, primero se planificó cual era la prioridad fundamental que se requería para el desarrollo del sistema; entonces el cliente y el programador decidieron que historias eran más importantes y que debieron ser implementadas en primera instancia y además se estimó el tiempo de construcción de cada historia.
- **Entregas pequeñas:** Se procuró que las entregas deben de ser lo más pequeñas posibles, conteniendo siempre los requerimientos de negocio más importantes para el cliente. De esta manera el cliente va obteniendo funcionalidades del sistema en forma gradual hasta finalizar el proyecto. En cada entrega los programadores obtienen retroalimentación del cliente determinando si lo implementado es lo que en realidad necesita.
- **Metáfora:** la metáfora proporcionó al equipo una imagen del

sistema, la cual ellos utilizaron para describir la estructura en forma simple y sencilla. Las ventajas de su aplicación es lo que hizo más fácil la comprensión del sistema y ayudó a mantener el diseño simple.

- **Diseño simple:** El diseño fue creado en forma progresiva, sin prever las necesidades del futuro. Al tener un diseño simple capaz de mantener las características actuales del sistema, este pudo adaptarse mejor a un entorno cambiante cuando surgió nuevos requerimientos o cambiaron los ya establecidos.
- **Pruebas:** En esta práctica, las pruebas del código implementado se llevó a prueba al mismo tiempo de programarse un cierto bloque de código.
- **Programación por parejas:** En esta práctica, no fue posible realizar una programación en pareja ya que el código del sistema fue implementado en su mayor dimensión solamente por el investigador del presente trabajo.
- **Prioridad colectiva del código:** En esta práctica, el investigador pudo modificar en cualquier momento del desarrollo del sistema, una parte o todo el código de un determinado módulo o historia.
- **Integración continua:** la integración del código se realizó en forma continua. La ventaja de la investigación continua fue obtener retroalimentación lo más rápido posible. Además de ser más sencilla que las integraciones que se realizan luego de varias semanas de

programación.

- **Semana de cuarenta horas:** En esta práctica, la programación de código fuente se realizó en plazo de cuatro meses; de los cuales solo como máximo 8 horas diarias y 5 días a la semana.
- **Estándares de programación:** Esta práctica. La programación se realizó de acuerdo a los estándares de la programación en entorno web, evitándose así errores y molestias a la hora de mudarse de un sistema operativo.

3.7. Material Experimental

Los materiales y herramientas utilizados para el desarrollo del trabajo de investigación fueron los siguientes:

3.7.1. Software

El funcionamiento del desarrollo del sistema Registro Civil para la municipalidad distrital de Plateria se realizó bajo los Sistemas Operativos de Microsoft Windows Windows 7, Windows 8, Windows 8.1 y Windows 10.

Herramientas para el desarrollo del sistema:

- ✓ Xampp
- ✓ PHP y MySQL.
- ✓ Workbeanch
- ✓ JavaScript, Ajax y CSS3.
- ✓ Sublime Tex3
- ✓ HTML5.

- ✓ Enterprise Architect
- ✓ Star UML-The Open Source UML/MDA Platform 5.0.2.1570
- ✓ Navegadores: Internet Explorer 11, Mozilla Firefox 23.0.1 y Google Chrome 45.0.1500.95 m.

3.7.2. Hardware

En cuanto al Hardware se utilizó una computadora Intel Core i5 y una computadora portátil Hp Core i5.

CAPITULO IV

RESULTADOS Y DISCUSION

4.1. Resultados

4.1.1. Ámbito del Problema

El sistema Registro Civil para la municipalidad distrital de Plateria, mediante la automatización para el trámite documentario en la unidad de trámite documentario se mejora la atención a las unidades orgánicas y así mismo a la población de Plateria.

Lo expuesto anteriormente fueron las evidencias para la implementación de un Sistema de Información para brindar una atención más eficiente y oportuna.

4.1.2. Especificación de Requerimientos del Sistema

- Realizar el registro de los usuarios al sistema de información a través del Sitio Web.

- Registrar y validar y garantizar la calidad de la información obtenida de los participantes mediante encuestas realizadas después de la implementación del sistema.
- Evaluar los resultados de los participantes para minimizar el tiempo de atención.
- Realizar las ponderaciones de las calificaciones, y finalmente presentará una calificación final.
- Realizar el mantenimiento y las actualizaciones de los intereses.

Identificación y descripción de historias de usuarios y módulos del sistema

Para poder especificar los requerimientos se hizo uso de las Historias de Usuarios, como también el uso de los Diagramas de caso de Uso que constituyen una técnica en el desarrollo de la Programación Extrema (XP).

4.1.3. Elaboración de diagramas de caso de uso

4.1.3.1. Análisis de requisitos del sistema

El desarrollo del software no consiste únicamente en la codificación, aunque sea una de las fases más importantes, sino que existen múltiples tareas que deben ser desarrolladas si se desea un sistema robusto y de fácil mantenimiento y con garantías de finalización en el tiempo y coste estimados.

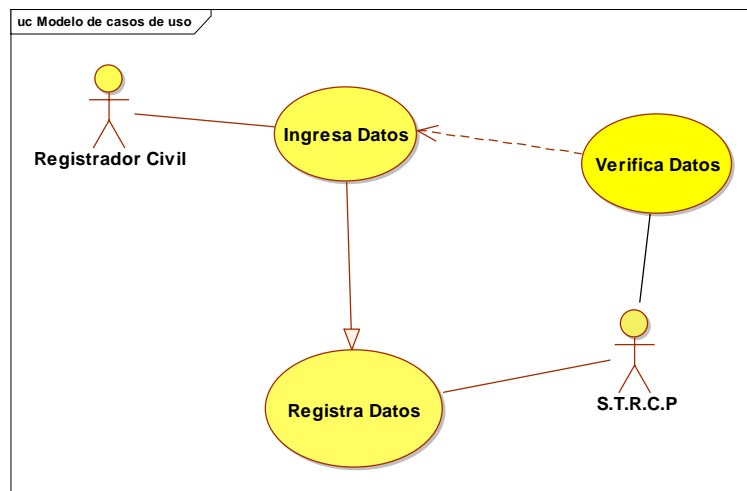
Para determinar las tareas es necesario asegurar un proceso de desarrollo software, adaptándolo según se estime conveniente al proyecto

en el cual se aplique.

Una de estas tareas es permitir conocer en todo momento el estado en el que se encuentra el sistema durante el desarrollo.

4.1.3.2. Diagrama de casos de uso del negocio

Figura N° 18 Acciones del Administrador del Sistema



FUENTE: Software Enterprise Architect UML

- En la Figura N 18 el diagrama muestra los diferentes tipos de usuarios así como administrador, trámite y dependencia y el rol que tiene dentro del sistema, de manera que se controla el trabajo de cada uno de usuario, dando privilegio a los mismos según tipos de usuario.

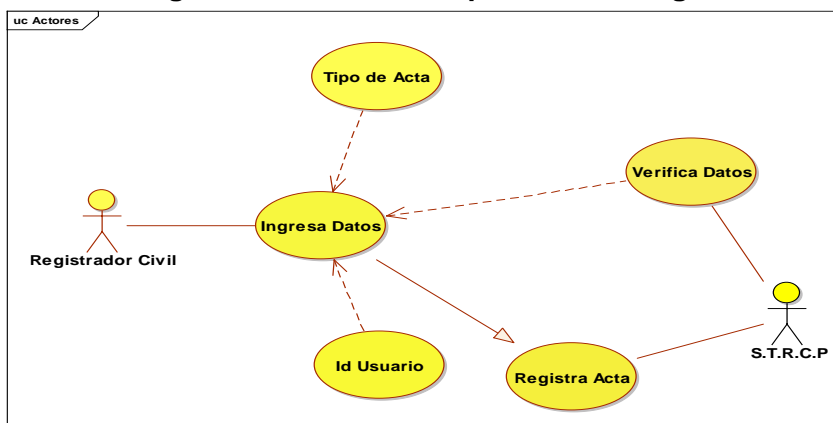
Tabla N° 5 Acciones Generales del Administrador

1. Caso de Uso	Login, Registro, Validación, Resultado, Editar-Eliminar.
2. Descripción	Este caso de uso permite que el administrador pueda dar mantenimiento, generar reportes, evaluar y gestionar seguridad
3. Actores	Registrador civil
4. Precondiciones	Antes de que este caso de uso pueda comenzar, tiene que ser llamado por alguno de las opciones correspondientes a sus privilegios.
5. Requerimientos Especiales	El sistema necesita tener una base de datos.
6. Flujo del Evento	
Evento disparador: El caso de uso se inicia cuando el administrador accede al sistema web a través de un navegador web.	
<ul style="list-style-type: none"> • Flujo Básico <<Control total>> • El sistema muestra las opciones disponibles según los privilegios del administrador. <ol style="list-style-type: none"> 1. El Administrador accede al sistema web. 2. El sistema muestra la pantalla inicial. 3. El sistema muestra el control total de los participantes. 4. El administrador podrá ver todos los resultados obtenidos y editar como modificar. 5. Finaliza el caso de uso. 	
7. Poscondiciones	
El administrador tendrá el control total del REGISTRO CIVIL.	

Fuente: Elaboración propia

4.1.3.3. DIAGRAMA DE CASO DE USO DEL REQUERIMIENTO

Figura N° 19 Diagrama de Uso del Requerimiento Registrar Actas



FUENTE: software Enterprise Architect UML

En la Figura N° 19 el diagrama muestra el requerimiento para registrar el trámite de documentos en cada dependencia que realiza el trámite respectivo en el sistema.

Tabla N° 6 Acciones generales del usuario

1. Caso de Uso	Registro, Derivar, Recepcionar , Archivar
8. Descripción	Este caso de uso permite en cada dependencia se registre y derive los documentos a la unidad correspondiente con el trámite.
9. Actores	Usuario de dependencia
10. Precondiciones	Antes de que este caso de uso pueda comenzar, tiene que ser llamado por alguno de las opciones correspondientes a sus privilegios.
11. Requerimientos Especiales	El sistema necesita tener una base de usuarios.
12. Flujo del Evento	
Evento disparador: El caso de uso se inicia cuando el usuario acceso al sistema web a través de un navegador web.	
<ul style="list-style-type: none"> • Flujo Básico <<Login - Tramite>> • El sistema muestra las opciones disponibles según los privilegios del usuario. <ol style="list-style-type: none"> 1. El usuario de dependencia accede al sistema web. 2. El sistema muestra la pantalla inicial. 3. El sistema muestra un formulario para la atención de documentos de trámite. 4. El sistema muestra un formulario para el registro de nuevos documentos de trámite. 5. El sistema muestra un formulario para el envío de documentos. 6. El sistema muestra un formulario para el para archivar si el documento ya está atendido. 7. Finaliza el caso de uso. 	
13. Poscondiciones	
El usuario de dependencia tramita documentos y registra actas de nacimiento actas de defunción y actas de matrimonio	

Fuente: Elaboración propia

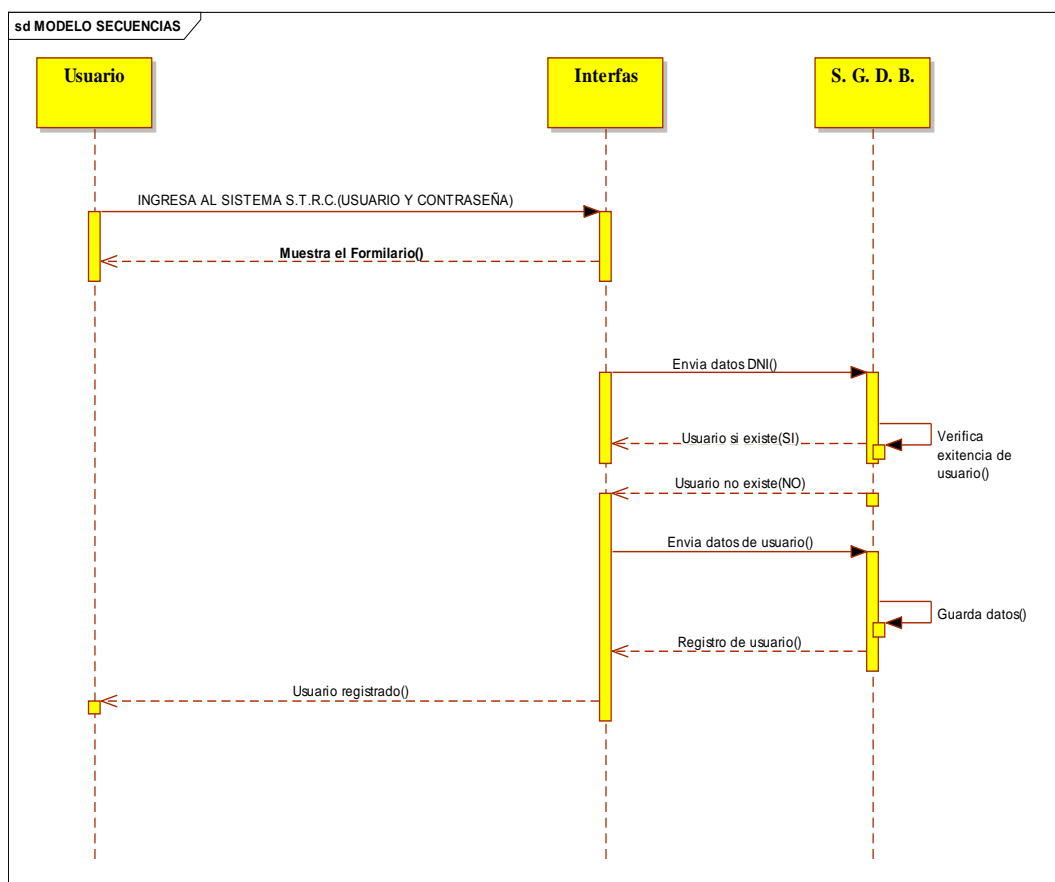
4.1.4. Elaboración de diagrama de secuencia

4.1.4.1. Ingreso de usuarios registrados al sistema

En el Figura N° 23 el diagrama se muestra la secuencia de interacción entre los objetos y los mensajes para realizar el proceso de crear nuevos usuarios en el sistema.

El usuario solicita al personal de mayor rango o al administrador del sistema para la creación de nuevos usuarios al sistema. El nuevo usuario luego ingresa y luego el sistema evalúa si son correcto los datos del usuario.

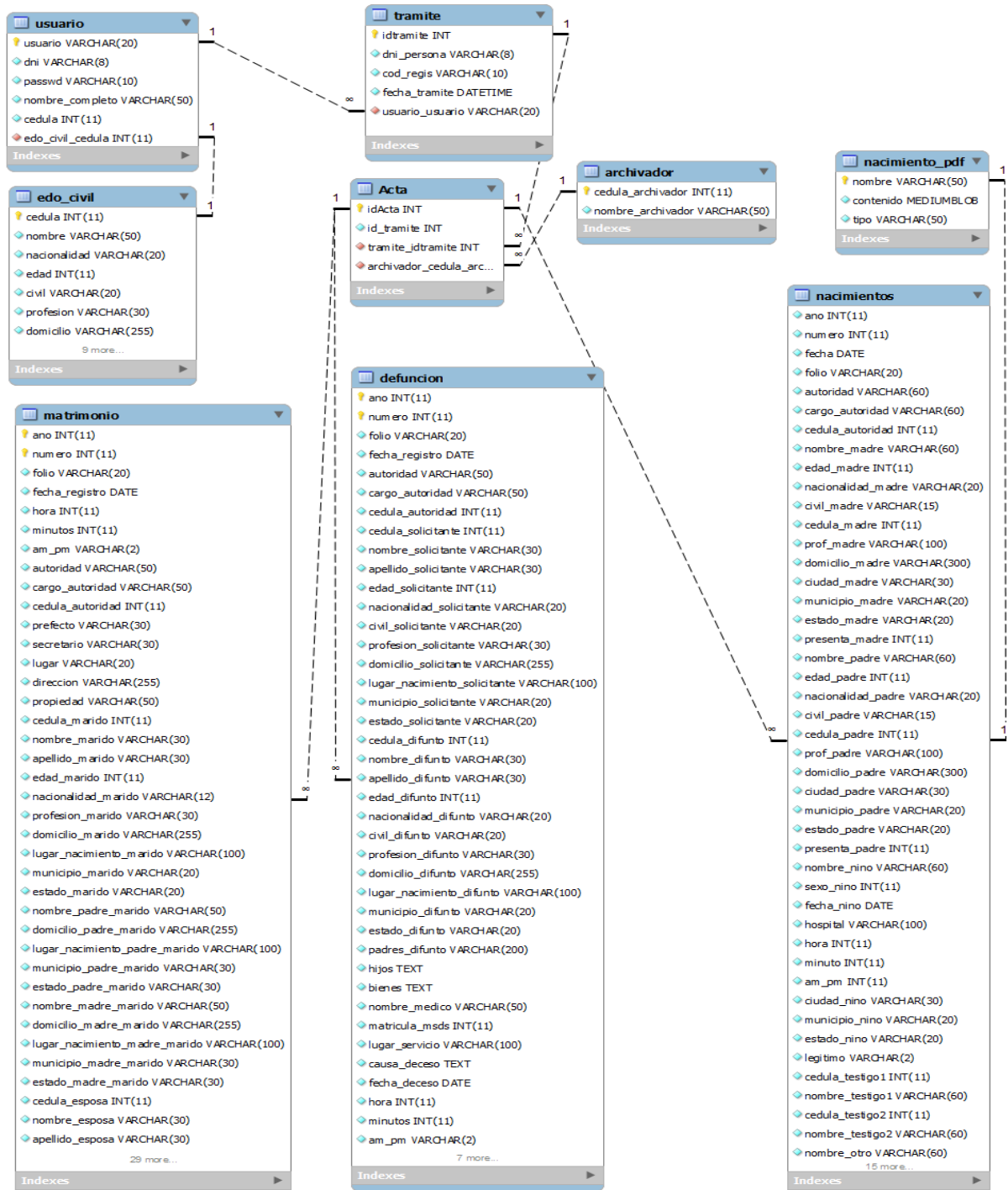
Figura N° 20 Diagrama de Secuencia Ingreso de Usuarios



FUENTE: software Enterprise Architect UML

4.1.5. Diseño

Figura N° 21 Entidad Relación de la base de datos

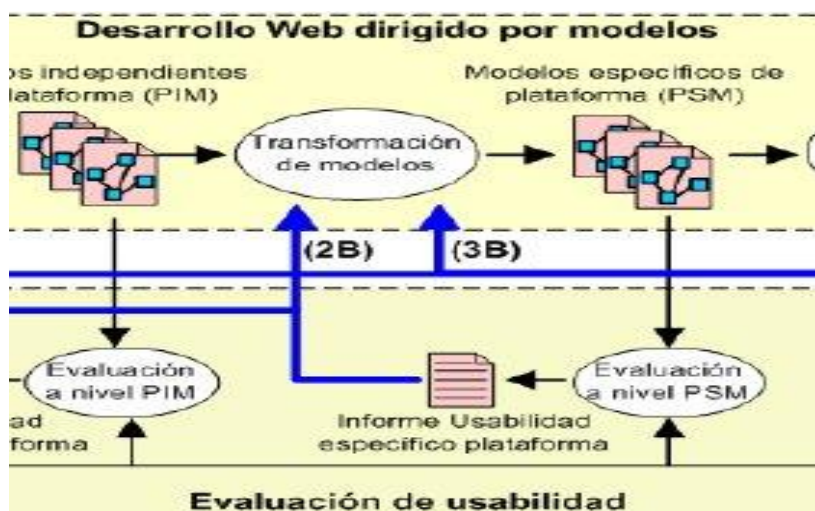


FUENTE: MySQL WorkBench

Implementación y desarrollo web

El desarrollo del software se realizó con el lenguaje de programación PHP y el lenguaje de marcado HTML en el editor de código Sublime Text3, divididos en dos áreas, una es la parte del código y el otro es donde muestra la interfaz la cual está siendo diseñada en la parte del código donde se hace las modificaciones lógicas del software en PHP o en HTML y en la parte del interfaz se puede hacer también modificaciones en mayor cantidad de código puede ser insertado botones, áreas de texto, creación de tablas y todo lo necesario para la creación del software.

Figura N° 22 Modelo desarrollo web



Fuente: www.proyectprogram/desarrollosweb/

4.1.6. Pruebas

4.1.6.1. métricas orientadas a la función

Son medidas indirectas del software y del proceso por el cual se desarrolla. Las métricas orientadas a la función se centran en la funcionalidad o utilidad del programa

Los puntos de función que obtienen utilizando una función empírica

basado en medidas cuantitativas del dominio de información del software y valoraciones subjetivos de la complejidad del software. Los puntos de función se calcularon de acuerdo a la siguiente tabla.

Tabla N° 7 Valoraciones del dominio de información del desarrollo de un sistema de Registro Civil para la municipalidad distrital de Plateria

PARAMENTROS DE MEDICION	CUENTA	SIMPLE	MEDIA	COMPLEJA	TOTAL
Número de entradas de usuario	4	3			12
Número de salidas de usuario	3	3			9
Número de consultas del usuario	3		4		12
Número de archivos	2	7			14
Número de interfaces externas	6		8		48
Cuenta Total					95

FUENTE: Elaboración propia

Para calcular los puntos de fusión se utilizó la siguiente relación:

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

Donde:

Fi : Es el valor de los factores de calidad del desarrollo del software .

Para encontrar los valores de ajuste de complejidad del desarrollo de un sistema de registro civil para la municipalidad distrital de Plateria.

Tabla N° 8 Valoraciones de ajuste de la complejidad del desarrollo de un registro civil para la municipalidad distrital de Plateria

FACTORES DE CALIDAD	VALOR
¿Requiere el sistema copias de seguridad y de recuperación fiables?	3
¿Se requiere de comunicación de datos?	2
¿Existen funciones de procesamiento distribuido?	5
¿ Es crítico el rendimiento?	1
¿ Se ejecutará el sistema en un entorno operativo existente y fuertemente utilizado?	4
¿ Requiere el sistema entrada de datos interactiva?	2
¿ Requiere la entrada de datos interactiva que las transacciones de entrada se lleven a cabo sobre múltiples pantallas u operaciones?	2
¿ Se actualizan los archivos maestros de forma interactiva?	2
¿ Son complejas las entradas, salidas, archivos o las peticiones?	2
¿ Es complejo el procesamiento interno?	3
¿Se ha diseñado el código para ser reutilizable?	3
¿ Están incluidas en el diseño la conversión y la instalación?	3
¿Se ha diseñado el sistema para soportar múltiples instalaciones en diferentes organizaciones?	3
¿Se ha diseñado la aplicación para facilitar los cambios y para ser fácilmente utilizada por el usuario?	3
Total (Suma)	39

FUENTE: Elaboración propia

Por consiguiente:

$$PF = 95 * [0.65 + (0.01*39)]$$

$$PF = 98.98 \cong 99$$

- El tiempo estimado que se tardó el desarrollo e implementación del desarrollo de un registro civil para la municipalidad distrital de Plateria, fue en un promedio de 3 meses y distribuías en 8 horas diarias aproximadamente. Entonces la productividad media del sistema es de

$PF/persona_mes, 99/3 = 33$ punto de fusión al mes (persona/mes).

- Según la tarifa al laborar estimada en nuevos soles es de S/. 1800.00 persona/mes, el coste por PF fue de $1800/33 = 54.54$
- El costo estimado del sistema de registro civil para la municipalidad distrital de Plateria fue de $99*54.54 = S/. 5399$ y el esfuerzo estimado de una persona/mes

El uso de métricas de punto de fusión, permitió asegurar el cumplimiento de la entrega del software en el tiempo establecido a la municipalidad distrital de Plateria; además permitió estimar el esfuerzo y el costo del Sistema implementado.

4.1.6.2. métricas de calidad de software

Para la evaluación del nivel de calidad del producto de software del sistema de registro civil para la municipalidad distrital de Plateria, se aplicó los indicadores de calidad del Estándar ISO – 9126, que ofrece una ficha de evaluación en el cual se obtuvieron las respuestas emitidas por los usuarios del sistema.

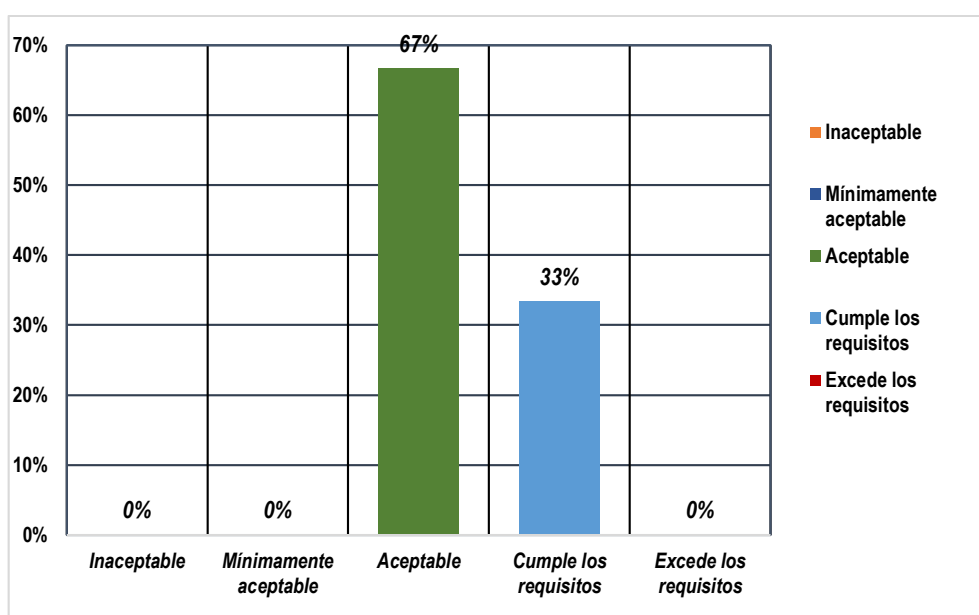
La calidad del producto de software del sistema de registro civil para la municipalidad distrital de Plateria, para lo cual se comprobó llenando la ficha de evaluación en se encuentra en el Anexo N° 4 Y Anexo N° 5 el resultado final se obtuvo de esta evaluación se muestra en la Tabla N° 7.

Tabla N° 9 Resultados de la Validación de la Calidad del Producto de Software del Sistema

Clasificación	Intervalo	Nº	%
Inaceptable	[27 – 54 >	0	0%
Mínimamente aceptable	[54 – 81 >	0	0%
Aceptable	[81 – 95 >	2	67%
Cumple los requisitos	[95 - 122 >	1	33%
Excede los requisitos	[122 - 135]	0	0%
Total		3	

Fuente: Elaborado por el autor.

Figura N° 23 Resultados de la Validación de la Calidad del Producto de Software del Sistema



Fuente: Elaborado por el autor.

En el Tabla N° 9 y la Figura N° 23 se presenta la medición de la calidad de software mediante el Estándar ISO-9126, que considera 5 factores principales, según el número de indicadores de calidad se puede obtener un puntaje entre 27 y 135 puntos, teniendo intervalos que ubican al producto en los intervalos de “Inaceptable”, “Mínimamente Aceptable”, “Aceptable”, “Cumple los Requisitos” y “Excede los Requisitos”.

Se puede apreciar que en un 67% se tiene el calificativo de Aceptable, el cual indica que el software cumple con realizar las tareas

básicas, pero que aun faltando algunos analizar algunos puntos de vista y criterios para ser considerado como óptimo. Se observa también que en un 33% obtiene el calificativo de “Cumple los requisitos”, esto nos indica que el sistema cumple todos los requisitos planteados por los usuarios, en cuanto a los factores e indicadores propuestos por el estándar. Se observa también que un 0% del total se obtiene el calificativo de “Excede los Requisitos”, lo cual nos muestra que con el sistema se puede hallar proceso que mejoran y exceden los requisitos básicos del sistema, por ejemplo, exceso de interfaces a la hora de realizar un proceso.

Por tanto, en un 100% el sistema aprueba el Estándar ISO-9126 de calidad de software, ubicándose mayoritariamente en al escala de “Cumple los Requisitos”.

CAPÍTULO V

CONCLUSIONES

- 1) Se ha diseñado el aplicativo usando tecnologías de desarrollo basado en Xampp, con MySQL.
- 2) Se ha modelado la base de datos con el software MySQL Workbench, como herramienta de diseño del diagrama.
- 3) La interfaz del Sistema web es ergonómica de fácil acceso e intuitiva utilizando representaciones visuales y gráficas, ventanas, cuadros de diálogo, botones y comandos.
- 4) La seguridad del sistema en el control de la información, dio como resultado que el 100% aprueba el Estándar ISO-9126 de calidad de software, ubicándose mayoritariamente en al escala de “Cumple los Requisitos”

CAPÍTULO VI

RECOMENDACIONES

- Se recomienda integrar el software a una interfaz web una vez incrementada la información de los usuarios de la municipalidad, para que ellos puedan acceder desde cualquier interfaz de comunicación ya sea computador, Tablet o Smartphone de modo que el alcance del sistema de información se incrementa y es posible darle un sentido de distribuido para los usuarios y cualquier trámite que deseen realizar esto incrementa el alcance y la satisfacción de los usuario y ciudadanos que dependen de la Municipalidad.
- Incrementar fuertemente las capacitaciones en TIC (Tecnologías de información y Comunicaciones) para el personal y personal que pueda guiar a los usuarios y prestatarios de los servicios de trámite del o los sistemas de información, muchos de los procesos se quedan sin aplicación útil del contenido por el desconocimiento de los usuarios. Préstese especial atención en las TICs para un mejor desempeño de los trabajadores de la Municipalidad.
- Se recomienda integrar el software a una interfaz web una vez

incrementada la información de los usuarios de la municipalidad, para que ellos puedan acceder desde cualquier interfaz de comunicación ya sea computador, Tablet o Smartphone de modo que el alcance del sistema de información se incrementa y es posible darle un sentido de distribuido para los usuarios y cualquier trámite que deseen realizar esto incrementa el alcance y la satisfacción de los usuario y ciudadanos que dependen de la Municipalidad.

- Para el desarrollo del software de pequeñas magnitudes tanto en costo y documentación, se recomienda la utilización de metodologías ágiles tal como la Programación Extrema(XP)
- Se recomienda la adquisición de un equipo exclusivamente para el almacenamiento y la administración de la base de datos (DATA)

CAPÍTULO VII

REFERENCIAS BIBLIOGRAFÍAS

Aubry, C. (2017). *html5 y css3: revolución en el diseño de sus sitios web*.
barcelona: 3ª ed.

Baez. (2012). "*sistemas web*".

Baiker, A. (2016). introducción a xampp y mysql. *wikipedia*,
<https://es.wikipedia.org/wiki/xampp>.

Bartolomé Sintés , M. (2014). *qué es xampp*. obtenido de wikipedia:
<https://es.wikipedia.org/wiki/xampp>

Blanco Arteaga, C. (2005). "*sistema automatizado para el control de envío y recepción de la correspondencia del banco provincial de caracas*".
caracas.

Borjas Giraldo, G. (2013). "*análisis, diseño e implementación de un sistema de información para la administración de horarios y rutas en empresas de transporte público*". lima: puc.

- Carlos. (s.f.). *el lenguaje unificado de modelado (uml)*. obtenido de
[http://profesores.fi-b.unam.mx:](http://profesores.fi-b.unam.mx) <http://profesores.fi-b.unam.mx/carlos/aydoo/uml.html>
- Carrera Jimenez, D. (2009). *análisis y diseño de un sistema de trámite de*. lima.
- Castillo Barriga, E. (2005). *desarrollo de sistema de información orientado a objetos para la gestión del registro civil de la municipalidad provincial de chucuito juli - 2003*. puno.
- Cervantes, H. (2009). *arquitectura de software*.
- Chaffer, J., & Swedberg, K. (2009). *aprende jquery v. 1.3*. madrid: anaya.
- Chaparro Lopez, P. A. (2005). *"sistema de informacion para administración de proyectos de grado"*. bogota: siap.
- Coaquira Pinto, W. (2015). *rediseño de procesos de negocio aplicando la tecnología workflow para el proceso de trámite documentario de la unidad de gestión educativa local puno - 2014*. puno.
- Cobo, A., Gómez, P., Pérez, D., & Rocha, R. (2005). *php y mysql tecnologías para el desarrollo de aplicaciones web*. españa: díaz de santos.
- Cox Sanmiguel, L. (2012). *"sistemas de información, en la gestión y control de hojas de seguridad electrónica en la dirección general de registro civil del ecuador"*. quito.
- Eguiluz Perez, J. (2008). *concepto javascript*.

- Flores Paredes , A., & Roque Choque, M. (2015). *desarrollo de un sistema de registro único y gestión documentaria de los juzgados de paz del distrito judicial de puno - 2015*. puno.
- Font Aranda, O. (2013). *"implementación de un sistema de gestión documental en la universidad central "marta abreu" de las villas cuba: facultad de ciencias de la información y de educación"*. granada.
- Fresno Fernandez, V. (2012). *lenguajes de programación y procesadores*. editorial universitaria ramon areces.
- Gauchat, J. (2012). *el gran libro de html5, css3 y javascript*. españa: marcombro.
- Gilfillan, I. (2003). *la biblia de mysql*. ee.uu.: anaya.
- Hayamares Huaman, L. (2012). *implementación de un sistema de información para el área de registro civil, en la municipalidad distrital de pueblo nuevo*. chincha.
- Heurtel, O. (1995). *php y mysql*.
- inei. (2007). *instituto nacional de estadística e informática*. obtenido de <http://censos.inei.gob.pe>: <http://censos.inei.gob.pe/cpv2007/tabulados>
- Gutierrez, J. (2010). *el concepto framework en el ámbito del desarrollo de sistemas software*.
- Kendall, K., & Kendall, J. (2011). *análisis y diseño de sistemas*. méxico: pearson.
- Largo Garcia, C., & Marin Masco, E. (2005). *guia técnica para evaluación de software*.

- M. Stair, R., & W. Reynolds, G. (2010). *principios de sistemas de información un enfoque administrativo*. méxico: pearson educación.
- Martinez. (2001). *los sistemas de información*.
- Matzukawa. (2002). *"modelado de software uml"*.
- Meza Cardama , L. (2012). *"análisis y diseño e implementación del sistema informático detramite documentario de registro civil vía web para lamunicipalidad distrital de punchana de iquitos, del mes deseptiembre a diciembre del 2012"*. iquitos.
- Narciso, E. L. (2011). *estadística,matematica y computación*. obtenido de
<http://reyesestadística.blogspot.pe>:
<http://reyesestadística.blogspot.pe/2011/07/muestreo-simple-aleatorio.html>
- Nevado Cabello, V. (2010). *introduccion a las bases de datos relacionales*. madrid: vision libros.
- Oppel, A. (2009). *fundamentos de bases de datos*. méxico: mcgraw-hill.
- Penades, L. (2006). *"programación extrema xp"*.
- Pérez A, O. (2011). *cuatro enfoques metodológicos para el desarrollo de software rup – msf – xp - scrum*. bogotá.
- Pressman, R. (2010). *ingeniería del software un enfoque práctico*. méxico: mcgraw-hill.

Ralph M., S., & George W., R. (2010). *principios de sistemas de información*.
méxico: cengage learning editores.

Sabana. (2006). "*introducción base de datos*".

Sciencedirect. (s.f.). *instrumento de variables*. obtenido de
<https://www.sciencedirect.com/>: <https://www.sciencedirect.com/>

Solis, J. (2015). *bootstrap*. costa rica.

Tejada Coaguila, J. (2016). "*diseño e implementacion de prototipo de control de
flujo y temperatura para sistema "hvac" aplicando control multivariable
cascada*". arequipa.

Tomasino. (2005).

Welling, Luke ; Thomson, Laura. (2009). *php y mysql*. ee.uu.: anaya multimedia.

Wiki. (s.f.). *editores de codigos*. obtenido de https://desarrolloweb.com:
<https://desarrolloweb.com/wiki/editor-de-codigo.html>

Wikipedia. (s.f.). *es.wikipedia.org*. obtenido de
https://es.wikipedia.org/wiki/base_de_datos/

Wikipedia. (s.f.). *los lenguajes de programación*. obtenido de
https://es.wikipedia.org:
https://es.wikipedia.org/wiki/lenguaje_de_programaci%c3%b3n

ANEXOS

ANEXO 1



UNIVERSIDAD NACIONAL DEL ALTIPLANO – PUNO
FACULTAD DE INGENIERÍA ESTADÍSTICA E INFORMÁTICA
ESCUELA PROFESIONAL DE INGENIERÍA ESTADÍSTICA E INFORMÁTICA



ENCUESTA PARA LAS PERSONAS QUE OPERAN EL SISTEMA REGISTRO CIVIL PARA LA
MUNICIPALIDAD DISTRITAL DE PLATERIA – 2017

Nombre del Encuestado: _____
Localidad: _____ Dirección: _____
CARGO: _____
OFICINA: _____ Fecha: ____ / ____ / ____

A continuación se le presenta 6 preguntas, le solicito que frente a ellas exprese su opinión personal considerando que no existe preguntas correctas ni incorrectas, marcando con un aspa (X) una de las alternativas que cree conveniente de acuerdo a su criterio y experiencia durante su permanencia en la institución.

1. ¿Cómo considera Ud. El diseño de la interfaz del sistema registro civil para la municipalidad distrital de Plateria?
 - a) Muy Bueno
 - b) Poco Amigable
 - c) No es Amigable
2. ¿Cómo considera Ud. ¿La interacción del sistema registro civil para la municipalidad distrital de Plateria?
 - a) Muy Fácil.
 - b) Fácil.
 - c) Difícil
3. ¿Cómo considera Ud. Los servicios que ofrece el sistema registro civil para la municipalidad distrital de Plateria?
 - a) Eficiente.
 - b) Medianamente eficiente
 - c) Deficiente
4. ¿el ingreso de información y/o registro al sistema es?
 - a) Muy fácil
 - b) Fácil
 - c) Complicado
5. ¿Cree Ud. Que el registro de información al sistema es confiable?
 - a) Si
 - b) No
6. ¿considera Ud. Que los reportes que emite es satisfactorio?
 - a) Muy bueno
 - b) Bueno
 - c) Regular

ANEXO 2



UNIVERSIDAD NACIONAL DEL ALTIPLANO – PUNO
FACULTAD DE INGENIERÍA ESTADÍSTICA E INFORMÁTICA
ESCUELA PROFESIONAL DE INGENIERÍA ESTADÍSTICA E INFORMÁTICA



ENCUESTA PARA LAS PERSONAS QUE TRAMITAN SUS ACTAS DE NACMIENTOS PATRMONIO Y DEFUNCON EN LA MUNICIPALIDAD DE PLATERIA – 2018

Nombre del Encuestado: _____

Localidad: _____ Dirección: _____

CARGO: _____

OFICINA: _____ Fecha: ____/____/____

A continuación se le presenta 4 preguntas, le solicito que frente a ellas exprese su opinión personal considerando que no existe preguntas correctas ni incorrectas, marcando con un aspa (X) una de las alternativas que cree conveniente de acuerdo a su criterio y experiencia durante su permanencia en la institución.

1. ¿Cómo considera Ud. El tiempo de atención actualmente en la municipalidad distrital de Plateria?
 - a) Muy rápido.
 - b) Rápido.
 - c) Medianamente Rápido.
 - d) Lento.
 - e) Muy lento

2. ¿Cómo considera Ud. la clasificación de los expedientes previamente registrados en la municipalidad distrital de Plateria?
 - a) Muy bueno.
 - b) Bueno.
 - c) Regular.
 - d) Malo
 - e) Muy malo

3. ¿Cómo considera Ud. el proceso de seguimiento de un expediente previamente registrado documental para la municipalidad distrital de Plateria?
 - a) Muy bueno.
 - b) Bueno
 - c) Regular
 - d) Malo.
 - e) Muy malo

4. ¿Cuánto tiempo ha demorado desde el momento de su llegada hasta que le presten el servicio de atención después de desarrollar el sistema? Especifique en minutos.....

ANEXO 3

FICHA DE EVALUACIÓN DE LA CALIDAD DEL PRODUCTO ESTANDAR ISO – 9126

INDICADORES	PUNTUACIÓN				
	1	2	3	4	5
1. FUNCIONALIDAD					
Adecuación: La capacidad del producto software para proporcionar un conjunto apropiado de funciones para tareas específicas y objetivos de los usuarios.					
Exactitud: La capacidad del producto software para proporcionar los resultados o efectos correctos y con el grado de precisión acordado.					
Interoperabilidad: La capacidad del producto software para interactuar con uno o mas sistemas especificados.					
Seguridad: Referido a la capacidad del producto software para proteger la información y los datos.					
Conformidad: La capacidad del producto software para adaptarse a los estándares, convenciones o regulaciones en leyes y prescripciones relativos a la funcionalidad.					
2. FIABILIDAD					
Madurez: La capacidad del producto software para evitar fallos provocados por errores en el software.					
Tolerancia a Fallos: La capacidad del producto software para mantener un nivel de rendimiento determinado en caso de defectos en el software o incumplimiento de su interfaz.					
Recuperabilidad: La capacidad de producto software para restablecer un determinado nivel de rendimiento y recuperar los datos afectados directamente en caso de ocurrir en fallo.					
Conformidad: La capacidad del producto software para adaptarse a estándares, convenciones y regulaciones referidas a la fiabilidad.					
3. USABILIDAD					
Comprensibilidad: La capacidad del producto software para permitir al usuario que entienda si el software es adecuado, y como debe utilizarse para determinadas tareas y bajo ciertas condiciones de uso.					
Facilidad de Aprendizaje: La capacidad del producto software para permitir al usuario aprender su aplicación.					
Atracción: La capacidad del producto software para atraer al usuario.					
Conformidad: La capacidad del producto software para adaptarse a estándares, convenciones, guías de estilo y regulaciones relacionadas con la usabilidad.					
Operabilidad: La capacidad del producto software para permitir que el usuario lo opere y lo controle.					

INDICADORES	PUNTUACIÓN				
	1	2	3	4	5
4. EFICIENCIA					
Comportamiento Temporal: La capacidad del producto software para proporcionar tiempos de respuesta y de procesamiento apropiados cuando realiza sus funciones bajo condiciones determinadas.					
Utilización de Recursos: La capacidad del producto software para utilizar cantidades y tipos de recursos apropiados cuando el software realiza su función bajo determinadas condiciones.					
Conformidad: La capacidad del producto software para adaptarse a estándares o convenciones relacionadas con la eficiencia.					
5. MANTENIBILIDAD					
Analizabilidad: Capacidad del producto software de diagnosticar sus deficiencias o causas de fallos, o de identificar las partes que deben ser modificadas.					
Cambiabilidad: Capacidad del producto software de permitir implementar una modificación especificada. La implementación incluye los cambios en el diseño, el código y la documentación.					
Estabilidad: Capacidad de producto software de evitar los efectos inesperados de las modificaciones.					
Facilidad de Prueba: Capacidad del producto software de permitir validar las partes modificadas.					
Conformidad: Capacidad del producto software de cumplir los estándares o convenciones relativas a la mantenibilidad.	1.1.1	1.1.2	1.1.3	1.1.4	1.1.5
6. PORTABILIDAD					
Adaptabilidad: La capacidad del producto software para ser adaptado para ambientes determinados sin realizar acciones o aplicar medios, más que los proporcionados para este propósito para el software considerado.					
Facilidad de Instalación: La capacidad del producto software para ser instalado en un ambiente determinado.					
Coexistencia: La capacidad del producto software para coexistir con otro software independiente en un ambiente común compartiendo recursos.					
Reemplazabilidad: La capacidad del producto software para ser utilizado en lugar de otro producto de software para el mismo propósito en el mismo ambiente.					
Conformidad: La capacidad del producto software para adaptarse a estándares relacionados con la portabilidad.					
SUB TOTALES					
TOTAL					

ANEXO 4

**TABLA DE VALORES DE PUNTUACIÓN DE LA CALIDAD DEL PRODUCTO DE SOFTWARE
ESTÁNDAR ISO - 9126**

Indicador Cualitativo	Valor
Deficiente	1
Malo	2
Regular	3
Bueno	4
Muy bueno	5

Clasificación	Intervalo	Nº
Inaceptable	[27 – 54 >	
Mínimamente aceptable	[54 – 81 >	
Aceptable	[81 – 95 >	
Cumple los requisitos	[95 - 122 >	
Excede los requisitos	[122 - 135]	
Total		

ANEXO 5

1. RESULTADOS DEL CUESTIONARIO CLIENTE – SERVIDOR

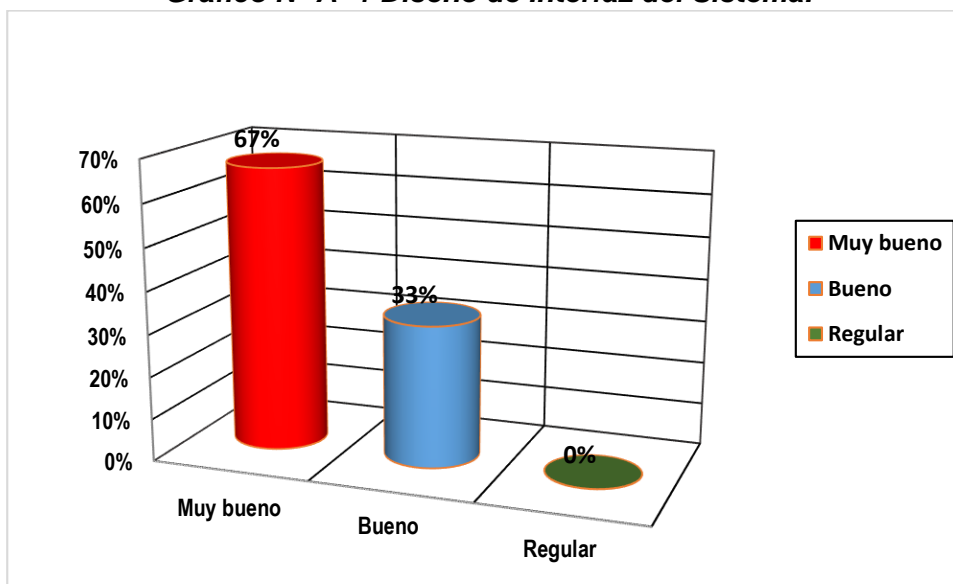
1.1. Resultados del diseño de la Interfaz del Sistema registro civil para la municipalidad distrital de Plateria.

Tabla N° A- 1 Diseño de Interfaz del Sistema.

Diseño_de_la_interfaz	Frecuencia	Porcentaje acumulado
Muy bueno	2	67%
Bueno	1	33%
Regular	0	0%
Total	3	100%

Fuente: Encuesta aplicada a los usuarios del sistema

Gráfico N° A- 1 Diseño de Interfaz del Sistema.



Fuente: Elaborada por el desarrollador del trabajo de investigador

De acuerdo a las encuestas realizadas al personal encargado que labora en la unidad de registro civil se puede observar sobre el diseño de interfaz que el 67% de los encargados de las dependencias afirman que el sistema es Muy bueno y el 33% de los encargados de las dependencias afirman que el sistema es bueno

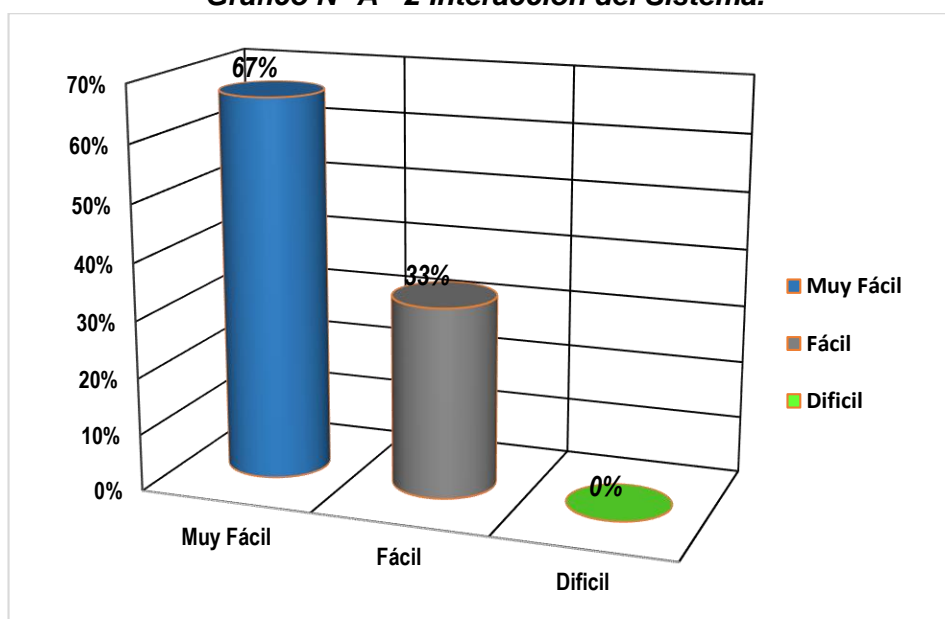
1.2. Resultados sobre la interacción de Sistema registro civil para la municipalidad distrital de Plateria.

Tabla N° A - 2 Interacción del Sistema.

Interacción_con_el_Sistema	Frecuencia	Porcentaje acumulado
Muy Fácil	2	67%
Fácil	1	33%
Difícil	0	0%
Total	3	100%

Fuente: Encuesta aplicada a los usuarios del sistema

Gráfico N° A - 2 Interacción del Sistema.



Fuente: Elaborada por el desarrollador del trabajo de investigador

De acuerdo a las encuestas realizadas al personal encargado que labora en la unidad de registro civil; se puede observar sobre la interacción con el software que el 67% de los encargados de las dependencias afirman que el sistema es Muy fácil y el 33% de los encargados de las dependencias afirman que el sistema es fácil

ANEXO 6

RESULTADOS DE LAS ENCUESTAS REALIZADAS A LOS USUARIOS Y SUS RESPECTIVOS

CÁLCULOS

Tiempo	x2	x1	D_i	$D_i - \bar{D}$	$(D_i - \bar{D})^2$
1	3	12	9	-0,722222222	0,521604938
2	5	10	5	-4,722222222	22,29938272
3	5	15	10	0,277777778	0,077160494
4	3	15	12	2,277777778	5,188271605
5	5	15	10	0,277777778	0,077160494
6	5	10	5	-4,722222222	22,29938272
7	4	15	11	1,277777778	1,632716049
8	4	15	11	1,277777778	1,632716049
9	5	20	15	5,277777778	27,85493827
10	5	15	10	0,277777778	0,077160494
11	5	10	5	-4,722222222	22,29938272
12	3	10	7	-2,722222222	7,410493827
13	5	10	5	-4,722222222	22,29938272
14	3	15	12	2,277777778	5,188271605
15	4	12	8	-1,722222222	2,966049383
16	3	15	12	2,277777778	5,188271605
17	5	15	10	0,277777778	0,077160494
18	5	15	10	0,277777778	0,077160494
19	3	10	7	-2,722222222	7,410493827
20	4	10	6	-3,722222222	13,85493827
21	3	15	12	2,277777778	5,188271605
22	3	10	7	-2,722222222	7,410493827
23	3	13	10	0,277777778	0,077160494
24	4	20	16	6,277777778	39,41049383
25	5	17	12	2,277777778	5,188271605
26	4	15	11	1,277777778	1,632716049
27	3	15	12	2,277777778	5,188271605
28	5	20	15	5,277777778	27,85493827
29	5	10	5	-4,722222222	22,29938272
30	4	20	16	6,277777778	39,41049383
31	4	13	9	-0,722222222	0,521604938
32	3	8	5	-4,722222222	22,29938272
33	3	15	12	2,277777778	5,188271605
34	3	15	12	2,277777778	5,188271605
35	4	15	11	1,277777778	1,632716049
36	5	10	5	-4,722222222	22,29938272
suma	145	495	350		379,2222222
Promedio	4,027777778	13,75	9,722222222		

Fuente: Elaboración propia

ANEXO 7

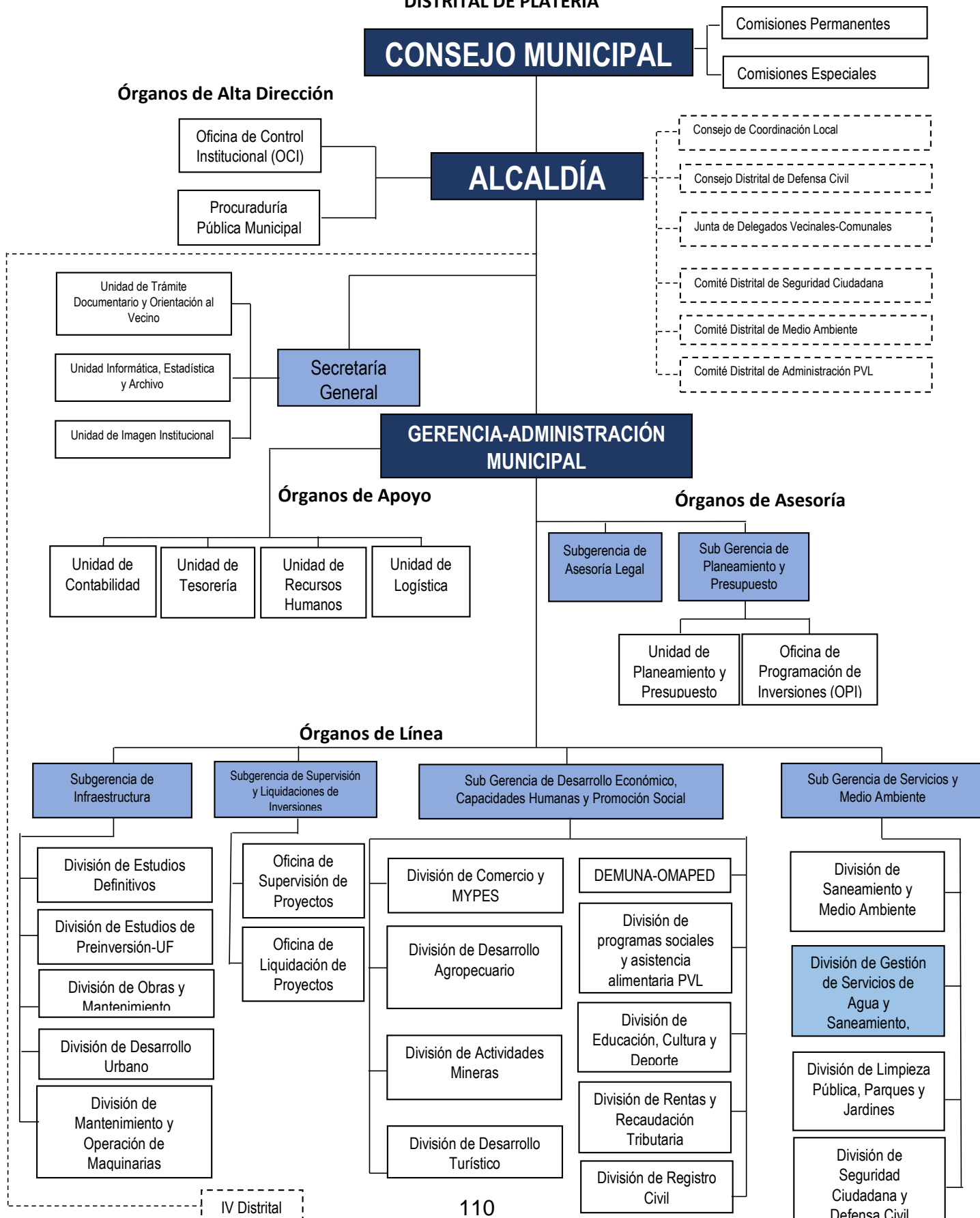
RESULTADOS DE LA EVALUACIÓN DE LA CALIDAD DEL PRODUCTO DE SOFTWARE DEL SISTEMA DE REGISTRO CIVIL PARA LA MUNICIPALIDAD DISTRITAL DE PLATERIA

INDICADORES	PUNTUACIÓN														
	Usuario 1					Usuario 2					Usuario 3				
	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5
1. FUNCIONALIDAD:															
Adecuación:				x					x					x	
Exactitud:					x				x						x
Interoperabilidad:					x				x						x
Seguridad:					x				x						x
Conformidad:					x					x					x
2. FIABILIDAD:															
Madurez:					x				x						x
Tolerancia a fallos:					x				x						x
Recuperabilidad:					x				x						x
Conformidad:			x					x							x
3. USABILIDAD:															
Comprensibilidad				x					x					x	
Facilidad de Aprendizaje:					x				x						x
Atracción:					x				x						x
Conformidad:					x				x						x
Operabilidad:					x				x						x
4. EFICIENCIA:															
Comportamiento					x				x						x
Utilización de Recursos:					x				x						x
Conformidad:					x										x
5. MANTENIBILIDAD:															
Analizabilidad:					x				x						x
Cambiabilidad:					x										x
Estabilidad:					x										x
Facilidad de Prueba:					x				x						x
Conformidad:					x				x						x
6. PORTABILIDAD:															
Adaptabilidad:					x				x						x
Facilidad de Instalación:					x				x						x
Coexistencia:					x				x						x
Remplazabilidad:					x				x						x
Conformidad:									x						x
SUB TOTALES	0	0	3	20	100	0	4	33	40	5	0	8	18	28	50
TOTAL	123					82					104				
PROMEDIO	103														

Fuente: Elaboración propia

ANEXO 8

ORGANIGRAMA ESTRUCTURAL DE LA MUNICIPALIDAD DISTRIITAL DE PLATERIA



IV Distrital

ANEXO 9

UNIVERSIDAD NACIONAL DEL ALTIPLANO – PUNO
FACULTAD DE INGENIERÍA ESTADÍSTICA E INFORMÁTICA
ESCUELA PROFESIONAL DE INGENIERÍA ESTADÍSTICA E INFORMÁTICA



MANUAL DEL USUARIO

**DESARROLLO DE UN SISTEMA DE REGISTRO CIVIL PARA LA
MUNICIPALIDAD DISTRITAL DE PLATERIA - 2017**

Elaborado por Bach: ELIZABETH AROPAZA ARPASI

Plateria, marzo del 2018.

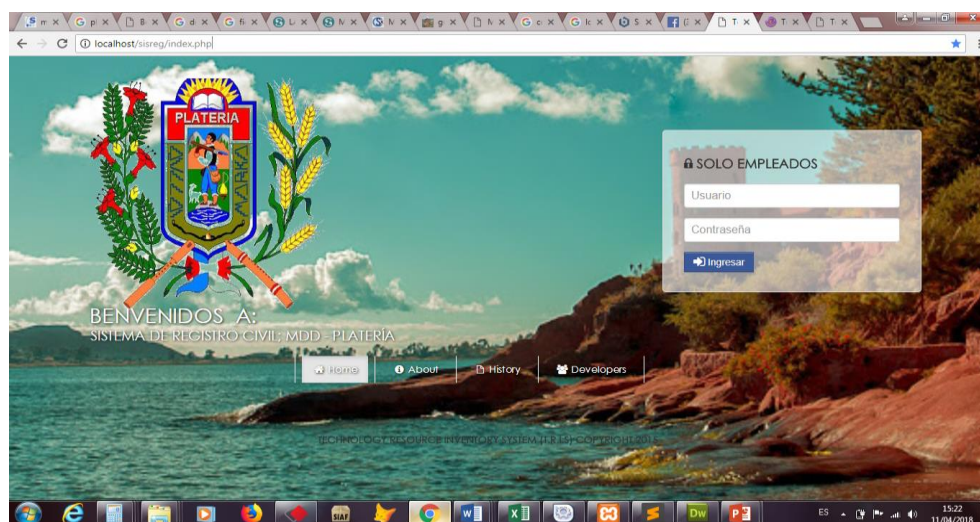
PRESENTACIÓN

En el presente documento se presenta el manual de usuario del SISTEMA DE REGISTRO CIVIL PARA LA MUNICIPALIDAD DISTRITAL DE PLATERIA que tiene como objetivo proporcionar una guía de práctica a los encargados de dicha área, para el manejo y dominio del sistema también se detalla cada una de las opciones del menú principal así como las instrucciones necesarias y las acciones a realizar en cada pantalla.

1. INGRESO AL SISTEMA

1.1. Panel Principal del Sistema

Aquí se muestra la primera pantalla del sistema donde se puede mostrar el ingreso para loguearse al sistema, así como consta de dos opciones como: **USUARIOS**, **NUEVO** **USUARIO**.



1.2. Ingreso al Sistema Condición Administrador

Del panel de principal escogemos la opción ADMINISTRADOR se muestra la pantalla para el ingreso del administrador al Sistema.



1.3. Ventana para Registro de nuevo usuario

Del panel de principal escogemos la opción NUEVO USUARIO se muestra la pantalla para el registro de nuevo usuario al Sistema.



1.4. Ventana para principal

Aquí se muestra la pantalla para la atención para registrar actas de nacimientos defunción y actas de matrimonio.

