

UNIVERSIDAD NACIONAL DE SAN CRISTÓBAL DE HUAMANGA

FACULTAD DE INGENIERÍA DE MINAS, GEOLOGÍA Y CIVIL

ESCUELA PROFESIONAL DE INGENIERÍA DE SISTEMAS



“Diseño de un sistema de información para el Área de Catastro Urbanístico de la Municipalidad Distrital de Hualla, Ayacucho, 2021”

Tesis presentada por : Bach. Luis Cancho Cuba

Para optar el título profesional de : Ingeniero Informático

Tipo de investigación : Observacional, transversal y descriptivo

Área de investigación : Ingeniería de Software

Asesor : Dr. Ing. Manuel Avelino Lagos Barzola

Ayacucho, junio de 2022



ACTA DE SUSTENTACIÓN DE TESIS

ACTA N° 054-2022-FIMGC

En la ciudad de Ayacucho, en cumplimiento a la RESOLUCIÓN DECANAL N° 204-2022-FIMGC-D, siendo el primer día del mes de agosto del 2022, a horas 8:00 a.m.; se reunieron los jurados del acto de sustentación, en el Auditorium virtual google meet del Campus Universitario de la Universidad Nacional de San Cristóbal de Huamanga.

Siendo el Jurado de la sustentación de tesis compuesto por el presidente el **Dr. Ing. Efraín Elías PORRAS FLORES**, Jurado - Asesor el **Dr. Ing. Manuel Avelino LAGOS BARZOLA**, Jurado el **Mg. Ing. Christian LEZAMA CUELLAR**, Jurado el **Mg. Ing. Hubner JANAMPA PATILLA** y secretario del proceso el **Ing. Richar Ruben JORGE BERROCAL**, con el objetivo de recepcionar la sustentación de la tesis denominada “DISEÑO DE UN SISTEMA DE INFORMACIÓN PARA EL ÁREA DE CATASTRO URBANÍSTICO DE LA MUNICIPALIDAD DISTRITAL DE HUALLA, AYACUCHO, 2021”, sustentado por el Sr. Luis CANCHO CUBA, Bachiller en Ex-escuela de Ingeniería Informática.

El Jurado luego de haber recepcionado la sustentación de la tesis y realizado las preguntas, el sustentante al haber dado respuesta a las preguntas, y el Jurado haber deliberado; califica con la nota aprobatoria de 15 (QUINCE).

En fe de lo cual, se firma la presente acta, por los miembros integrantes del proceso de sustentación.



Dr. Efraín Elías Porras Flores
DECANO

Firmado digitalmente
por Dr. Ing. Efraín Elías
Porras Flores
Fecha: 2022.08.05
10:39:09 -09'00'

Dr. Ing. Efraín Elías PORRAS FLORES
presidente

Mg. Ing. Hubner JANAMPA PATILLA
Jurado

Dr. Ing. Manuel Avelino LAGOS BARZOLA
Jurado Asesor

Mg. Ing. Christian LEZAMA CUELLAR
Jurado

Ing. Richar Ruben JORGE BERROCAL
Secretario del Proceso

c.c.:
Bach. Luis CANCHO CUBA
Jurados (4)
Archivo



UNSCH

FACULTAD DE
INGENIERÍA
DE MINAS, GEOLOGÍA Y CIVIL

“Año del Fortalecimiento de la Soberanía Nacional”

CONSTANCIA DE ORIGINALIDAD DE TRABAJO DE INVESTIGACIÓN

CONSTANCIA N° 051-2022-FIMGC

El que suscribe; responsable verificador de originalidad de trabajos de tesis de pregrado en segunda instancia para las **Escuelas Profesionales** de la **Facultad de Ingeniería de Minas, Geología y Civil**; en cumplimiento a la Resolución de Consejo Universitario N° 039-2021-UNSCH-CU, Reglamento de Originalidad de Trabajos de Investigación de la UNSCH y Resolución Decanal N° 158-2021-FIMGC-UNSCH-D, deja constancia que Sr./Srta.

Apellidos y Nombres : Luis Cancho Cuba
Escuela Profesional : INGENIERO INFORMÁTICO
Título de la Tesis : “Diseño de un sistema de información para el Área de Catastro Urbanístico de la Municipalidad Distrital de Hualla, Ayacucho, 2021”
Evaluación de la Originalidad : **28 %** Índice de Similitud
Identificador de la entrega : 1875196314

Por tanto, según los Artículos 12, 13 y 17 del Reglamento de Originalidad de Trabajos de Investigación, es **PROCEDENTE** otorgar la **Constancia de Originalidad** para los fines que crea conveniente.

Ayacucho, 25 de julio del 2022

Firmado digitalmente
por LEZAMA CUELLAR
CHRISTIAN

Mg. Ing. Christian LEZAMA CUELLAR
Verificador de Originalidad de Trabajos de Tesis de Pregrado
de la FIMGC

(X) Con depósito para Sustentación y Tramite de Titulo

“Diseño de un sistema de información para el Área de Catastro Urbanístico de la Municipalidad Distrital de Hualla, Ayacucho, 2021”

por Luis Cancho Cuba

Fecha de entrega: 25-jul-2022 06:22p.m. (UTC-0500)

Identificador de la entrega: 1875196314

Nombre del archivo: Tesis_Luis_Cancho_Cuba_EPIS.pdf (3.93M)

Total de palabras: 28948

Total de caracteres: 176608

“Diseño de un sistema de información para el Área de Catastro Urbanístico de la Municipalidad Distrital de Hualla, Ayacucho, 2021”

INFORME DE ORIGINALIDAD

28%

INDICE DE SIMILITUD

25%

FUENTES DE INTERNET

2%

PUBLICACIONES

18%

TRABAJOS DEL ESTUDIANTE

FUENTES PRIMARIAS

1	Submitted to Universidad Nacional de San Cristóbal de Huamanga Trabajo del estudiante	12%
2	repositorio.unsch.edu.pe Fuente de Internet	7%
3	cybertesis.urp.edu.pe Fuente de Internet	1%
4	comercio-electronico162.blogspot.com Fuente de Internet	1%
5	repositorio.espe.edu.ec Fuente de Internet	1%
6	zulia.colciencias.gov.co:8081 Fuente de Internet	1%
7	Submitted to unjbg Trabajo del estudiante	<1%
8	baixardoc.com Fuente de Internet	<1%

9	281.wikispaces.com Fuente de Internet	<1 %
10	www.repositorioacademico.usmp.edu.pe Fuente de Internet	<1 %
11	pt.scribd.com Fuente de Internet	<1 %
12	es.slideshare.net Fuente de Internet	<1 %
13	repositorio.unan.edu.ni Fuente de Internet	<1 %
14	inba.info Fuente de Internet	<1 %
15	1library.co Fuente de Internet	<1 %
16	ctscafe.pe Fuente de Internet	<1 %
17	ninive.ismm.edu.cu Fuente de Internet	<1 %
18	plone.org Fuente de Internet	<1 %
19	issuu.com Fuente de Internet	<1 %
20	itamod.blogspot.com Fuente de Internet	<1 %

21	www.bibliotecadigital.usb.edu.co Fuente de Internet	<1 %
22	www.mindmeister.com Fuente de Internet	<1 %
23	hdl.handle.net Fuente de Internet	<1 %
24	docplayer.es Fuente de Internet	<1 %
25	Submitted to tec Trabajo del estudiante	<1 %
26	repositorio.utc.edu.ec Fuente de Internet	<1 %
27	www.mimdes.gob.pe Fuente de Internet	<1 %
28	repositorio.ucv.edu.pe Fuente de Internet	<1 %
29	bibdigital.epn.edu.ec Fuente de Internet	<1 %
30	repositorio.urp.edu.pe Fuente de Internet	<1 %
31	sistemaacademicogrupo5.wordpress.com Fuente de Internet	<1 %
32	www.monografias.com Fuente de Internet	<1 %

33

www.sigasa.com

Fuente de Internet

<1 %

34

icc2.act.uji.es

Fuente de Internet

<1 %

35

Yersi-Luis Huaman-Romani, Edwin Roque-Tito, Luis Bautista-Lopez, Marco-Dennis Gutierrez-Aguilar. "Level of Depression of College Students with Binary Logistic Regression Model Approximation in Covid-19 times", 2021 IEEE 1st International Conference on Advanced Learning Technologies on Education & Research (ICALTER), 2021

Publicación

<1 %

36

dspace.uclv.edu.cu

Fuente de Internet

<1 %

37

repositorio.upeu.edu.pe

Fuente de Internet

<1 %

38

www.arquitectura.com.ar

Fuente de Internet

<1 %

39

ri.biblioteca.udo.edu.ve

Fuente de Internet

<1 %

40

www.slideshare.net

Fuente de Internet

<1 %

Excluir citas Activo

Excluir bibliografía Activo

Excluir coincidencias < 30 words

Dedicatoria

A todos los compañeros que como en mi condición de egresado de la carrera de Ingeniería Informática se han alejado de manera no intencional del hermoso mundo de la Ingeniería de Software y más específicamente el desarrollo de programas de computador, pero que hacen todos los días sus intentos de volver al camino.

Agradecimiento

A mi alma máter, la Universidad Nacional de San Cristóbal de Huamanga, que dentro de sus aulas he aprendido de una profesión y que hoy en día sirve de soporte para mi aspecto personal y familiar.

A mis docentes de aquel entonces la Escuela Profesional de Ingeniería Informática, quienes supieron desarrollar en mí las capacidades que hoy en día me permiten desenvolverme en el campo laboral.

A mi madre, esposa e hijas, quienes continuamente me apoyaron y estuvieron al tanto que terminara este tan postergado proyecto.

Resumen

Muchas municipalidades distritales y específicamente la Municipalidad Distrital de Hualla, hoy en día aún se encuentran lejos de la reducción de la brecha digital que les permitiría la automatización y eficiencia de sus procesos en beneficio de su desarrollo como institución, así como a sus ciudadanos. El estudio presenta el análisis y diseño de un sistema de información para el Área de Catastro Urbanístico de la Municipalidad Distrital de Hualla, sistema que de llegarse a implementar permitirá la gestión del catastro urbano dentro de la municipalidad. El estudio presenta los resultados de la fase de exploración, fase de planificación y fase de iteración del análisis y diseño del sistema de información, aplicando el Proceso Ágil de Desarrollo de Software Programación Extrema. Se presentan los artefactos historias de usuario, arquitectura técnica inicial, el plan de alto nivel, el plan de versiones, la arquitectura técnica final, las tareas de ingeniería desarrolladas al detalle, propuestas iniciales de las interfaces gráficas de usuario, las tarjetas Clase Responsabilidad Colaboración (CRC) y el diseño lógico de la base de datos.

Palabras clave: análisis de software, diseño de software, catastro urbano, municipalidad.

Abstract

Many district municipalities, and specifically the District Municipality of Hualla, today are still far from reducing the digital gap that would allow them to automate and make their processes efficient for the benefit of their development as an institution, as well as their citizens. The study presents the analysis and design of an information system for the Urban Cadastre Area of the District Municipality of Hualla, a system that, if implemented, will allow the management of the urban cadastre within the municipality. The study presents the results of the exploration phase, planning phase and iteration phase of the analysis and design of the information system, applying the Agile Software Development Process Extreme Programming. The artifacts of user stories, initial technical architecture, the high-level plan, the version plan, the final technical architecture, the engineering tasks developed in detail, initial proposals of the graphical user interfaces, the Class Responsibility Collaboration cards are presented. (CRC) and the logical design of the database.

Keywords: software analysis, software design, urban cadastre, municipality.

Contenido

Página de aprobación o conformidad	ii
Dedicatoria	iii
Agradecimiento	iv
Resumen.....	v
Abstract	vi
Contenido	vii
Lista de tablas	x
Lista de figuras.....	xii
Introducción	xiii
Capítulo I.....	14
Planteamiento del problema	14
1.1 Diagnóstico y enunciado del problema.....	14
1.2 Formulación del problema	14
1.2.1 Problema principal	14
1.2.2 Problemas secundarios	14
1.3 Objetivos de la investigación	15
1.3.1 Objetivo principal	15
1.3.2 Objetivos secundarios.....	15
1.4 Hipótesis de la investigación.....	16
Capítulo II	17
Marco Teórico.....	17
2.1 Antecedentes de la Investigación	17
2.2 Marco teórico	17
2.2.1 Software.....	17
2.2.2 Lenguaje Unificado de Modelado (UML).....	18
2.2.3 Sistema de información	18
2.2.3.1 Actividades de un sistema de información	19
2.2.3.2 Datos de entrada y salida	20
2.2.3.3 Diseño de la base de datos.....	20
2.2.3.4 Diagrama de componentes	20
2.2.3.5 Diagrama de despliegue	20

2.2.3.6	Interfaces de Usuario.....	21
2.2.4	Programación Orientada A Objetos	21
2.2.4.1	Objeto	22
2.2.4.2	Mensaje.....	22
2.2.4.3	Clase.....	22
2.2.5	Programación Extrema (XP)	23
2.2.5.1	Los valores de XP.....	24
2.2.5.2	Actividades de XP.....	26
2.2.5.3	Prácticas básicas de XP.....	27
2.2.5.4	Procesos XP.....	30
2.2.5.5	Ciclo de Vida de XP	31
2.2.5.6	Actores y responsabilidades de XP	33
2.2.5.7	Artefactos de XP.....	34
	Capítulo III	39
	Material y métodos.....	39
3.1	Tipo investigación	39
3.2	Nivel de investigación	40
3.3	Diseño de la investigación	40
3.4	Población y muestra.....	41
3.4.1	Población.....	41
3.4.2	Muestra	41
3.5	Variables e indicadores	41
3.5.1	Definición conceptual de las variables.....	41
3.5.2	Definición operacional de los variables.....	42
3.6	Técnicas e instrumentos para el tratamiento de datos e información.....	42
3.6.1	Técnicas para recolectar información.....	42
3.6.2	Instrumentos para recolectar información	42
3.6.2.1	Guía de entrevista.....	42
3.6.3	Herramientas para el tratamiento de datos e información	43
3.6.4	Técnicas para aplicar el proceso ágil Programación Extrema	44
	Capitulo IV.....	48
	Resultados y discusión	48
4.1	Resultados de la aplicación del Proceso Ágil de Programación Extrema	48

4.1.1	Fase de exploración	48
4.1.1.1	Historias de usuario	48
4.1.1.2	Arquitectura técnica inicial.....	49
4.1.1.3	Plan de alto nivel.....	49
4.1.2	Fase de planificación.....	50
4.1.2.1	Historias de usuario	50
4.1.2.2	Plan de versiones	52
4.1.3	Fase de iteración.....	53
4.1.3.1	Arquitectura técnica final	54
4.1.3.2	Plan de iteración	55
4.1.3.3	Plan de iteración por fechas de desarrollo	56
4.1.3.4	Tareas de ingeniería	57
4.1.3.4.1	Historia de Usuario: Autenticar usuario.....	58
4.1.3.4.2	Historia de Usuario: Gestionar contribuyente	59
4.1.3.4.3	Historia de Usuario: Gestionar predio	63
4.1.3.4.4	Historia de Usuario: Gestionar declaración jurada de predio.....	67
4.1.3.4.5	Historia de Usuario: Gestionar cuentas de usuario	70
4.1.3.5	Interfaz gráfica de usuario.....	72
4.1.3.6	Tarjetas clase responsabilidad colaboración (CRC).....	76
4.1.3.7	Diseño lógico de la base de datos	79
4.2	Discusión de resultados	80
	Capitulo V	81
	Conclusiones y Recomendaciones	81
	Conclusiones.....	81
	Recomendaciones	82
	Lista de abreviaturas.....	85
	Glosario	86
	Anexo B	90
	Anexo c.....	91

Lista de tablas

Tabla 1 Historia de usuario	35
Tabla 2 Prueba de aceptación	36
Tabla 3 Tarea de Ingeniería.....	37
Tabla 4 Tarjeta CRC.....	37
Tabla 5 Instrumentos tipo Software utilizado para el tratamiento de los datos	43
Tabla 6 Fase de exploración	44
Tabla 7 Fase de planeación	45
Tabla 8 Fase de iteración.....	46
Tabla 9 Historias de Usuario	48
Tabla 10 Plan de alto nivel.....	49
Tabla 11 Historia de Usuario Autenticar Usuario.....	50
Tabla 12 Historia de Usuario Gestionar Contribuyente	50
Tabla 13 Historia de Usuario Gestionar Predio	51
Tabla 14 Historia de Usuario Gestionar Ubicación.....	51
Tabla 15 Historia de Usuario Gestionar declaración jurada de impuesto predial	52
Tabla 16 Historia de Usuario Gestionar cuentas de usuario Error! Marcador no definido.	
Tabla 17 Historia de Usuario Gestionar roles de usuario Error! Marcador no definido.	
Tabla 18 Plan de Versiones.....	53
Tabla 19 Plan de Iteración (Primera)	55
Tabla 20 Plan de iteración por fechas de desarrollo	56
Tabla 21 Tarea de Ingeniería Autenticar usuario	58
Tabla 22 Tarea de Ingeniería Registrar contribuyente.....	59
Tabla 23 Tarea de Ingeniería Buscar Contribuyente	60
Tabla 24 Tarea de Ingeniería Modificar contribuyente	61
Tabla 25 Tarea de Ingeniería Eliminar Contribuyente.....	62
Tabla 26 Tarea de Ingeniería Registrar predio	63
Tabla 27 Tarea de Ingeniería Buscar Predio	64
Tabla 28 Tarea de Ingeniería Modificar predio	65
Tabla 29 Tarea de Ingeniería Eliminar Predio.....	66
Tabla 30 Tarea de Ingeniería Generar Declaración Jurada de Predio	67

Tabla 31 Tarea de Ingeniería Buscar Declaración Jurada de predio	68
Tabla 32 Tarea de Ingeniería Imprimir Declaración Jurada de predio	69
Tabla 33 Tarea de Ingeniería Registrar Cuenta de usuario	70
Tabla 34 Tarea de Ingeniería Modificar Cuenta de Usuario.....	71
Tabla 35 Clase Contribuyente	77
Tabla 36 Clase Predio.....	77
Tabla 37 Clase Ubicación.....	78
Tabla 38 Clase Declaración Jurada	78
Tabla 39 Clase Usuario.....	78
Tabla 40 Matriz de Operacionalización	87

Lista de figuras

Figura 1 Valores de XP	25
Figura 2 Prácticas de XP.....	28
Figura 3 Ciclo de Vida Programación Extrema.....	31
Figura 4 Arquitectura Técnica Inicial	49
Figura 5 Arquitectura técnica final (Diagrama de componentes)	54
Figura 6 Arquitectura técnica final (Diagrama de despliegue)	55
Figura 7 Interfaz principal del sistema de información.....	72
Figura 8 Interfaz de autenticación del usuario para el sistema	72
Figura 9 Interfaz para gestión de contribuyentes.....	73
Figura 10 Interfaz para registro de contribuyentes.....	73
Figura 11 Interfaz para gestión de cuentas de predios.....	74
Figura 12 Interfaz para registro de predios	74
Figura 13 Interfaz para gestión de declaraciones juradas de predio	75
Figura 14 Interfaz para registro de declaraciones juradas de predio.....	75
Figura 15 Interfaz para gestión de cuentas de usuario.....	76
Figura 16 Interfaz para registro de cuentas de usuario	76
Figura 17 Diseño lógico de la base de datos	79

Introducción

Hoy en día, la implementación de las tecnologías de información y comunicaciones por parte de las organizaciones públicas y privadas es una necesidad imperiosa, puesto estos elementos finalmente desencadenan en una mejor gestión de los procesos de estas organizaciones.

Sin embargo, muchas entidades al no tener dentro de sus órganos funcionales una unidad de gestión de TIC's se ven limitados en cuanto al uso de la tecnología de información; se puede indicar que esta brecha permanece latente debido también al carente presupuesto asignadas a las municipalidades distritales.

En el caso de la Municipalidad Distrital de Hualla, también se encuentra caracterizada por esta realidad, en consecuencia, en esta oportunidad se da inicio a la implementación de un sistema de información para el área de catastro urbanístico.

Los objetivos del estudio son determinar los resultados del análisis del sistema de información para catastro urbanístico, así como también determinar los resultados del diseño de dicho sistema de información.

Capítulo I

Planteamiento del problema

1.1 Diagnóstico y enunciado del problema

La Municipalidad Distrital de Hualla, viene desarrollando sus procesos de gestión de predios manualmente, en ese sentido la información que se procesa en esta área es gestionada en medios físicos y digitales de manera desorganizada.

La gestión de la información según esta modalidad es deficiente porque conduce a la duplicación de información, errores constantes en los datos, pérdida de información valiosa, acceso ineficiente a los datos e información, porque no se puede acceder a la información de manera rápida y oportuna, es decir en tiempo real.

Así mismo, se ha observado que para que un contribuyente tenga información oficial sobre su predio, esta puede tener una demora sustancial, ya que no había iniciativas en cuanto a la implementación de un sistema de información para lo referente al catastro urbano.

1.2 Formulación del problema

1.2.1 Problema principal

¿Cuál es el análisis y diseño pertinente de un sistema de información para el Área de Catastro Urbanístico de la Municipalidad Distrital de Hualla, Ayacucho, 2021?

1.2.2 Problemas secundarios

- a) ¿Cuáles son las historias de usuario para el desarrollo del sistema de información para el Área de Catastro Urbanístico de la Municipalidad Distrital de Hualla, Ayacucho, 2021?
- b) ¿Cuáles son los datos de entrada y salida para el sistema de información

para el Área de Catastro Urbanístico de la Municipalidad Distrital de Hualla, Ayacucho, 2021?

- c) ¿Cuáles son las tareas de ingeniería para el desarrollo del sistema de información para el Área de Catastro Urbanístico de la Municipalidad Distrital de Hualla, Ayacucho, 2021?
- d) ¿Cuál es el diagrama de componentes para el desarrollo del sistema de información para el Área de Catastro Urbanístico de la Municipalidad Distrital de Hualla, Ayacucho, 2021?
- e) ¿Cuál es el diagrama de despliegue para el desarrollo del sistema de información para el Área de Catastro Urbanístico de la Municipalidad Distrital de Hualla, Ayacucho, 2021?
- f) ¿Cuál es el diseño de la base de datos para el desarrollo del sistema de información para el Área de Catastro Urbanístico de la Municipalidad Distrital de Hualla, Ayacucho, 2021?
- g) ¿Cuáles son los prototipos de interfaces de usuario para el desarrollo del sistema de información para el Área de Catastro Urbanístico de la Municipalidad Distrital de Hualla, Ayacucho, 2021?

1.3 Objetivos de la investigación

1.3.1 Objetivo principal

Determinar el análisis y diseño pertinente de un sistema de información para el Área de Catastro Urbanístico de la Municipalidad Distrital de Hualla, Ayacucho, 2021.

1.3.2 Objetivos secundarios

- a) Determinar las historias de usuario para el desarrollo del sistema de información para el Área de Catastro Urbanístico de la Municipalidad Distrital de Hualla, Ayacucho, 2021.
- b) Determinar los datos de entrada y salida para el sistema de información para el Área de Catastro Urbanístico de la Municipalidad Distrital de

Hualla, Ayacucho, 2021.

- c) Identificar y describir las tareas de ingeniería para el desarrollo del sistema de información para el Área de Catastro Urbanístico de la Municipalidad Distrital de Hualla, Ayacucho, 2021.
- d) Establecer el diagrama de componentes para el desarrollo del sistema de información para el Área de Catastro Urbanístico de la Municipalidad Distrital de Hualla, Ayacucho, 2021.
- e) Determinar diagrama de despliegue para el desarrollo del sistema de información para el Área de Catastro Urbanístico de la Municipalidad Distrital de Hualla, Ayacucho, 2021.
- f) Determinar el diseño de la base de datos para el desarrollo del sistema de información para el Área de Catastro Urbanístico de la Municipalidad Distrital de Hualla, Ayacucho, 2021.
- g) Establecer los prototipos de interfaces de usuario para el desarrollo del sistema de información para el Área de Catastro Urbanístico de la Municipalidad Distrital de Hualla, Ayacucho, 2021.

1.4 Hipótesis de la investigación

“No, no en todas las investigaciones cuantitativas plantean hipótesis. El hecho de que formulemos o no hipótesis depende de un factor esencial: el alcance inicial del estudio. Las investigaciones cuantitativas que formulan hipótesis son aquellas cuyo planteamiento define que su alcance será correlacional o explicativo, o las que tienen un alcance descriptivo, pero que intentan pronosticar una cifra o un hecho” (Hernández, Baptista y Collado, 2014).

Capítulo II

Marco Teórico

2.1 Antecedentes de la Investigación

Bautista (2019), en su tesis de pregrado titulado como "Sistema de información geográfico de catastro y rentas para la Municipalidad Provincial de Huaraz" realiza un estudio de nivel descriptivo, utilizando el proceso de desarrollo de software RUP, donde concluye que "el sistema de información geográfica para Catastro y Rentas de la Municipalidad provincial de Huaraz mejora la información catastral en la Municipalidad Provincial de Huaraz".

Por otro lado, Benites(2019) en su tesis de grado titulado como "Sistema de información vía Web para mejorar el control de ficha catastral rural en el departamento la Libertad", busca mejorar el control de ficha catastral rural en el departamento la Libertad mediante la implementación de un sistema de información vía Web, llegando a las conclusiones siguientes: "Se logro reducir el tiempo promedio en el registro de la ficha catastral rural en un 54.79% con un decremento de 175.76 segundos, el registro de la identificación del propietario en un 73.05% con un decremento de 318.55 segundos, el tiempo promedio en la búsqueda de la ficha catastral rural en un 85.28% con un decremento de 233.03 segundos y el tiempo promedio en la obtención de reporte de la ficha catastral rural en un 90.93% con un decremento de 437.11 segundos".

2.2 Marco teórico

2.2.1 Software

"Es el conjunto de programas, procedimientos, reglas, documentación y datos asociados que forman parte de las operaciones de un sistema informático". (IEEE, 2011).

“El software es el conjunto de programas, documentos, procedimientos y rutinas asociados con el funcionamiento de una computadora; Tiene las funciones de administrar los recursos o medios de la computadora, proporcionando herramientas para utilizar de manera eficiente estos recursos, actuando como enlace entre el usuario y la computadora. Se clasifican en software de sistema, software de aplicación, software de usuario final” (Ibáñez y García, 2009).

2.2.2 Lenguaje Unificado de Modelado (UML)

Para (Ferré y Sánchez, sf) “es un lenguaje que permite modelar, construir y documentar los elementos que forman un sistema software orientado a objetos. Se ha convertido en el estándar de facto de la industria, debido a que ha sido concebido por los autores de los tres métodos más usados de orientación a objetos: Grady Booch, Ivar Jacobson y Jim Rumbaugh”.

Funciones:

- a) Visualizar: UML permite expresar de una forma gráfica un sistema de forma que otro lo puede entender.
- b) Especificar: UML permite especificar cuáles son las características de un sistema antes de su construcción.
- c) Construir: permite guiar la realización de un sistema similar a los "planos".
- d) Documentar: capturar conocimiento respecto a un sistema a lo largo de todo el proceso de su ciclo de vida.

2.2.3 Sistema de información

“Un sistema de información es un sistema que recopila, almacena, procesa y distribuye conjuntos de información entre los diferentes elementos que componen una organización y entre la propia organización y su entorno”. (Pastore, 2002).

“Es un conjunto de componentes que interactúan entre sí para lograr un objetivo común. Aunque existe una gran variedad de sistemas, la mayoría de ellos se pueden representar mediante un modelo compuesto por cinco bloques básicos: elementos de entrada, elementos de salida, sección de transformación, mecanismos de control y objetivos” (Fernández, 2006).

2.2.3.1 Actividades de un sistema de información

a. Entrada de información

“Es el proceso mediante el cual el Sistema de Información adquiere los datos necesarios para procesar la información” (Pastor, 2002).

“Las entradas pueden ser manuales o automáticas. Los manuales son aquellos que son proporcionados directamente por el usuario, mientras que los automáticos son datos o información que proviene o se toma de otros sistemas o módulos” (Stair, 2000).

b. Almacenamiento de información

“Archivar es una de las actividades o habilidades más importantes que tiene una computadora, ya que a través de esta propiedad el sistema puede recordar la información guardada en la sección o proceso anterior” (Pastor, 2002).

Esta información generalmente se almacena en estructuras de información llamadas archivos.

c. Procesamiento de información

“Es la capacidad del Sistema de Información para realizar cálculos según una secuencia predeterminada de operaciones” (Fernández, 2006).

“El procesamiento de la información permite transformar los datos fuente en información que puede ser utilizada para la toma de decisiones, lo que permite, entre otras cosas, al decisor generar una proyección financiera a partir de los datos contenidos en la misma. un estado de resultados o un

balance general para un año base” (Stair, 2000).

d. Salida de información

“Un sistema de información es el sistema de personas, registros de datos y actividades que procesa datos e información en una organización determinada, incluidos los manuales de proceso o los procesos automatizados” (Pastor, 2002).

2.2.3.2 Datos de entrada y salida

Los datos de entrada son aquellos datos que serán procesados, en tanto los datos de salida son derivados de los datos entrantes. Una operación de entrada y salida de datos puede definirse al conjunto de acciones para transferir de un conjunto de datos y generar otros datos con mayor valor.

2.2.3.3 Diseño de la base de datos

El diseño de la base de datos es el proceso mediante el cual se modela un conjunto de datos, partiendo de un dominio del cual se obtendrá un modelo conceptual, luego un modelo lógico, al cual se le debe aplicar normalización, y finalmente obtener un modelo físico y poder implementarlo; cada nivel de creación se caracteriza por la realización de un esquema representativo de la base de datos.

2.2.3.4 Diagrama de componentes

Es una ilustración de todos los componentes de software que se usarán para construir un sistema; los componentes son agregaciones de alto nivel de las piezas de software más pequeñas y proveen un enfoque de construcción de bloques de “caja negra” para la elaboración de software.

2.2.3.5 Diagrama de despliegue

Ilustra la arquitectura física (estática) del hardware y el software en el sistema, la misma que se muestra mediante la distribución de sus dispositivos

(nodos); este diagrama permite identificar los nodos que se utilizarán el sistema, además representa la arquitectura física de la red, así como la distribución del software y finalmente ofrece una visión global del proyecto.

2.2.3.6 Interfaces de Usuario

Es el espacio mediante el cual se controla, interactúan o comunica el usuario con una aplicación de software o dispositivo de hardware. Este control o comunicación puede incluir controles gráficos, teclados o mouse.

2.2.4 Programación Orientada A Objetos

Izquierdo (2010), explica que "Es importante aclarar la diferencia que existe entre programación orientada a objetos y un lenguaje orientado a objetos.

La programación orientada a objetos es una filosofía, un modelo de programación, con su teoría y su metodología. En cambio, un lenguaje orientado a objetos es un lenguaje de programación que permite el diseño de aplicaciones orientada a objetos".

"El diseño orientado a objetos consiste en averiguar cuáles son los objetos de un sistema, las clases en que se pueden agrupar y las relaciones entre objetos". (Durán, 2007).

Según Rueda (2005), "La programación orientada a objetos es realmente un nuevo estilo de programación, el cual, básicamente consiste en definir clases y de poner dichas clases a comunicarse entre sí".

Martínez y Quiroga (2001), explican que "La programación orientada a objetos podría definirse como una filosofía para programar encapsulando datos y código para formar objetos, que interactúan para obtener resultados esperados".

2.2.4.1 Objeto

Durán (2007), señala que “Un objeto es cualquier cosa tangible o intangible que se pueda imaginar, definida frente al exterior mediante unos atributos y las operaciones que permiten modificar dichos atributos”.

Para Izquierdo (2010), “Un objeto no es más que un conjunto de variables (o datos) y métodos (o funciones) relacionados entre sí. Los objetos en programación se usan para modelar objetos o entidades del mundo real”.

Según Álvarez (2005), “Los objetos son ejemplares de una clase cualquiera. Cuando creamos un ejemplar tenemos que especificar la clase a partir de la cual se creará. Esta acción de crear un objeto a partir de una clase se llama instanciar”.

Según Martínez y Quiroga (2001), “Un objeto es un paquete que contiene datos y el código en forma de subrutinas que operarán sobre los datos del propio objeto. Los datos que guarda un objeto se llaman atributos del objeto, mientras que a las subrutinas que operan sobre los datos se le conoce como métodos”.

2.2.4.2 Mensaje

Izquierdo (2010), explica que “Un objeto por sí solo no es muy útil. En general, un objeto aparece como un componente más de una programación que contiene otros muchos objetos. Es precisamente haciendo uso de esta interacción como los programadores consiguen una funcionalidad de mayor orden y modelar comportamientos mucho más complejos”.

2.2.4.3 Clase

Izquierdo (2010), sostiene que “Una clase es una plantilla que define las variables y los métodos que son comunes para todos los objetos de un cierto tipo”.

Por otro lado, Durán (2007), señala que “Una clase es una plantilla que permite definir un conjunto de objetos”.

Según Álvarez (2005), “Las clases son declaraciones de objetos, también se

podrían definir como abstracciones de objetos. Esto quiere decir que la definición de un objeto es la clase”.

2.2.5 Programación Extrema (XP)

“XP es una metodología ágil para pequeños o medianos equipos, desarrollando software cuando los requerimientos son ambiguos o rápidamente cambiantes.” (Beck, 1999).

Weitzendfeld (2005), explica que “La programación extrema (XP, eXtreme Programming) es un modelo de proceso de software que toma los principios y prácticas aceptadas, y las lleva a niveles extremos. Tiene como objetivo reducir el riesgo en el ciclo de vida del software mediante grupos de desarrollo pequeños. Considera que la mejor manera de tratar la falta de requisitos estables en un sistema, es mediante la agilidad de un grupo pequeño de desarrollo”.

Según Don Wells (2009):

“XP es una metodología ágil 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. XP se basa en realimentación continua entre el cliente y el equipo de desarrollo, comunicación fluida entre todos los participantes, simplicidad en las soluciones implementadas y coraje para enfrentar los cambios. XP se define como especialmente adecuada para proyectos con requisitos imprecisos y muy cambiantes, y donde existe un alto riesgo técnico”.

Así mismo refiere que:

“Este método es típicamente atribuido a Kent Beck, Ron Jeffries y

Ward Cunningham. El objetivo de XP son grupos pequeños y medianos de construcción de software en donde los requisitos aún son muy ambiguos, cambian rápidamente o son de alto riesgo. Xp busca la satisfacción del cliente tratando de mantener durante todo el tiempo su confianza en el producto. Además, sugiere que el lugar de trabajo sea una sala amplia, si es posible sin divisiones (en el centro los programadores, en la periferia los equipos individuales). Una ventaja del espacio abierto es el incremento en la comunicación y el proporcionar una agenda dinámica en el entorno de cada proyecto”.

Respecto a los principios de este proceso ágil de desarrollo de software, el autor precisa que:

“XP se basa en cuatro principios: simplicidad, comunicación, retroalimentación y valor. Además, orientada por pruebas y refactorización, se diseñan e implementan las pruebas antes de programar la funcionalidad, el programador crea sus propios tests de unidad. Los objetivos de XP son muy simples: la satisfacción del cliente. Esta metodología trata de dar al cliente el software que él necesita y cuando lo necesita. Por tanto, debemos responder muy rápido a las necesidades del cliente, incluso cuando los cambios sean al final del ciclo de la programación”.

2.2.5.1 Los valores de XP

Según Don Wells (2009),
“XP no es en realidad un conjunto de normas, sino más bien una manera de trabajar en armonía con sus valores personales y corporativos”.

Figura 1

Valores de XP



a) Simplicidad

Realiza solo lo necesario y requerido. Esto podrá maximizar el valor creado por las inversiones realizadas hasta el momento.

El valor de la simplicidad no significa necesariamente que algo sea simple o fácil, pero si significa que primero te enfrentes a un problema "¿Qué es lo más simple que todavía puede funcionar?" y empezar desde allí. La simplicidad debe estar relacionada con el contexto del problema y el equipo que lo resuelve.

b) Comunicación

Todos son parte del equipo y nos comunicamos cara a cara todos los días. Desde los requisitos hasta el código, todos trabajan juntos. La comunicación dentro del equipo de trabajo es fundamental. Cuando ocurre un problema, muchas veces alguien ya sabe la solución, pero debido a la falta de comunicación, ese conocimiento no llega a quienes pueden hacer cambios. Incluso si el problema es inesperado y nuevo, la comunicación dentro del equipo puede conducir a la solución más efectiva y evitar que vuelva a ocurrir.

c) Feedback

Se demuestra el software tempranamente y con frecuencia luego escuchar

cuidadosamente y hacer los cambios necesarios.

El desarrollo, los requisitos del sistema, la arquitectura y muchos otros detalles están sujetos a cambios durante la vida del proyecto, por lo que hay poca dirección para permanecer constante. Los caminos iniciados antes de que varíe la experiencia suelen cambiar inmediatamente. Los cambios son inevitables, sin embargo, se necesita una retroalimentación para corregirlos.

d) Respeto

El respeto se muestra de muchas maneras diferentes. Los miembros del equipo se consideran entre ellos porque los desarrolladores no pueden hacer cambios que puedan causar demoras en el proyecto en marcha. Todos respetan el trabajo de otros, ya que siempre se esfuerzan por obtener productos de alta calidad y buscan el mejor diseño o el más eficiente para sus soluciones a través de la mejora del código. Los miembros del equipo respetan el trabajo de los demás al no degradar a los demás, mejorando la autoestima del grupo y aumentando la tasa de producción del equipo.

e) Coraje

Di la verdad sobre tu progreso y estimaciones. Queremos tener éxito, por lo que no documentamos excusas para el fracaso. No le tenemos miedo a nada porque nadie trabaja solo. Nos adaptamos a los cambios a medida que ocurren.

2.2.5.2 Actividades de XP

a) Codificar

Necesitamos sistematizar e incorporar nuestras ideas a través del código. En programación, el código expresa la interpretación de un problema, por lo que puede ser usado para comunicarse, compartir ideas, aprender y mejorar.

b) Hacer Pruebas

No hay características de software que no se pueden probar con una simple prueba. Las pruebas le brindan la oportunidad de saber si lo que ha implementado es realmente lo que pretendía. Las pruebas muestran que nuestro trabajo está funcionando. Si no puede pensar en ninguna prueba que pueda causar un error en su sistema, está completamente acabado.

c) Escuchar

Los programadores no lo sabemos todo, y sobre todo muchas cosas que la gente de negocios encuentra particularmente interesantes.

Se necesita escuchar a los clientes sobre sus problemas comerciales escuchando activamente lo fácil y difícil que es conseguirlos, y la retroalimentación entre ambos nos ayuda a todos a entender el problema.

d) Diseñar

El diseño crea la estructura que organiza la lógica del sistema. El sistema bien diseñado le permite realizar cambios y escalar en un solo lugar. El diseño debe ser simple. Si parte del sistema es complejo, es recomendable dividirlo en varias partes.

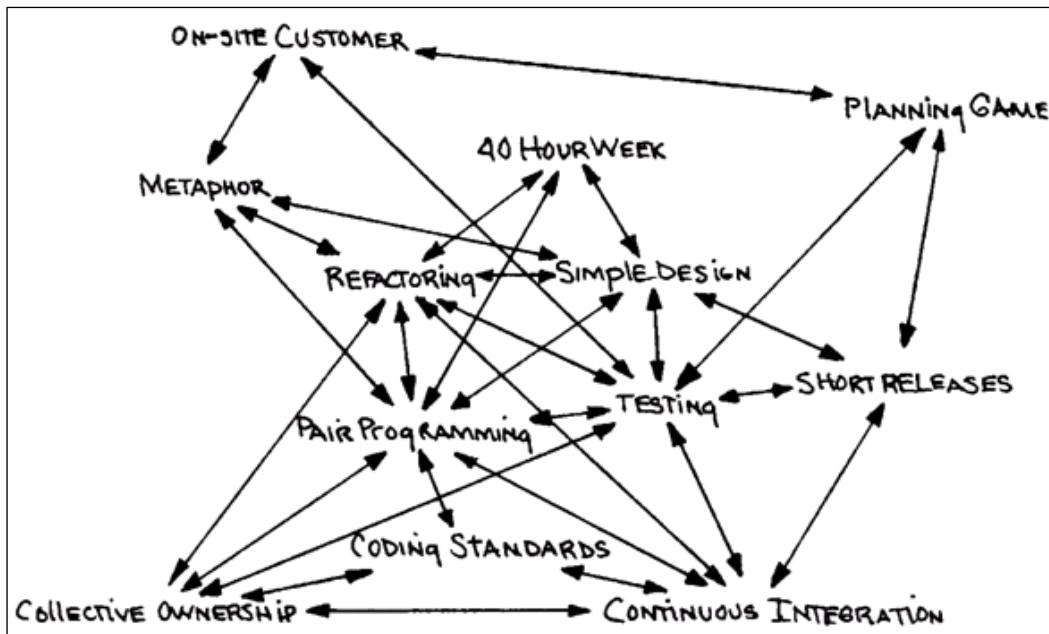
2.2.5.3 Prácticas básicas de XP

Cubel (2004), nos dice:

“para evaluar XP hay que mirar la gran foto, es decir, todo el conjunto de prácticas”.

Figura 2

Prácticas de XP



Nota: Fuente (Cube, 2004)

a) El juego de la planificación (Planning Game)

Los alcances de una próxima versión del sistema están definidos por prioridades y estimaciones técnicas.

El objetivo del PLanning Game es maximizar el valor del sistema a desarrollarse. La estrategia es incorporar las características más importantes a la producción lo antes posible. El elemento clave es la tarjeta de historias, los jugadores son los desarrolladores y clientes; y las movidas son exploración, selección y actualización.

b) Versiones pequeñas (Short Releases)

El producto software que se va desarrollando se pone rápidamente en producción. Las nuevas versiones se crean periódicamente, agregando características que se consideran valiosas para el cliente en cada iteración.

c) Metáfora del sistema (Metaphor)

Cada Proyecto es guiado por una breve historia sobre cómo funciona un sistema en general, reemplazar la arquitectura y ser un lenguaje común que todos (clientes y programadores) puedan entender, y está sujeto a cambios en cualquier momento.

d) Diseño Simple (Simple Designs)

El sistema será lo más sencillo posible, el diseño quedará reflejado en la tarjeta CRC (clase-responsabilidad-colaboración) y no se implementará ninguna funcionalidad innecesaria. Con esta técnica, pueden filtrar las clases encontradas durante el análisis para determinar qué clases realmente necesita para el sistema.

e) Pruebas Continuas (Testing)

Los casos de prueba son desarrollados antes de la fase de implementación. Los programadores establecen pruebas unitarias y los clientes especifican pruebas funcionales.

f) Refactorización (Refactoring)

Puede rediseñar el software sin alterar su comportamiento, como eliminar el código duplicado, simplificando las funciones y mejorar continuamente su código. Si su código se complica, necesita volver a cambiar el diseño a algo más sencillo.

g) Programación por parejas (Pair Programming)

Dos desarrolladores desarrollan el software en la misma computadora. "Una sola maquina con un teclado y un mouse".

h) Posesión Colectiva del código (Collective Code Ownership)

Nadie es dueño del código. El programador puede alterar cualquier parte del software según su evaluación, basado en estándares de desarrollo y se

excluyen los comentarios. La prueba siempre debe ser 100% funcional para que pueda integrarse permanentemente en todo el código.

i) Integración Continua (Continuous Integration)

Las modificaciones se añaden al código base reiteradas veces al día. Todos los casos de prueba deben pasarse antes y después de la integración. Se dispone de una máquina para la integración y se realiza una prueba funcional involucrando al cliente.

j) Semana laboral de 40 horas (40 – Hour Week)

Cada Trabajador labora no más de 40 horas a la semana. Si necesita trabajar horas extras, no lo haga durante dos semanas consecutivas.

k) Cliente en el Sitio (On Site Customer)

El equipo de desarrollo siempre tiene acceso al cliente y puede usarlo para hacer preguntas, establecer prioridades y más. Esto no siempre se logra. Los clientes muy jóvenes no sirven y los clientes muy antiguos no están disponibles. "Lo ideal es un cliente Analista".

l) Estándares de codificación (Coding Standard)

Todo el código debe escribirse de acuerdo con estándares de codificación.

2.2.5.4 Procesos XP

- a. El cliente fija el valor de negocio a implementar.
- b. El programador calcula el esfuerzo necesario para su implementación.
- c. El cliente selecciona qué construir, de acuerdo con sus prioridades y las restricciones de tiempo.
- d. El programador construye ese valor de negocio.
- e. Vuelve al paso 1.

2.2.5.5 Ciclo de Vida de XP

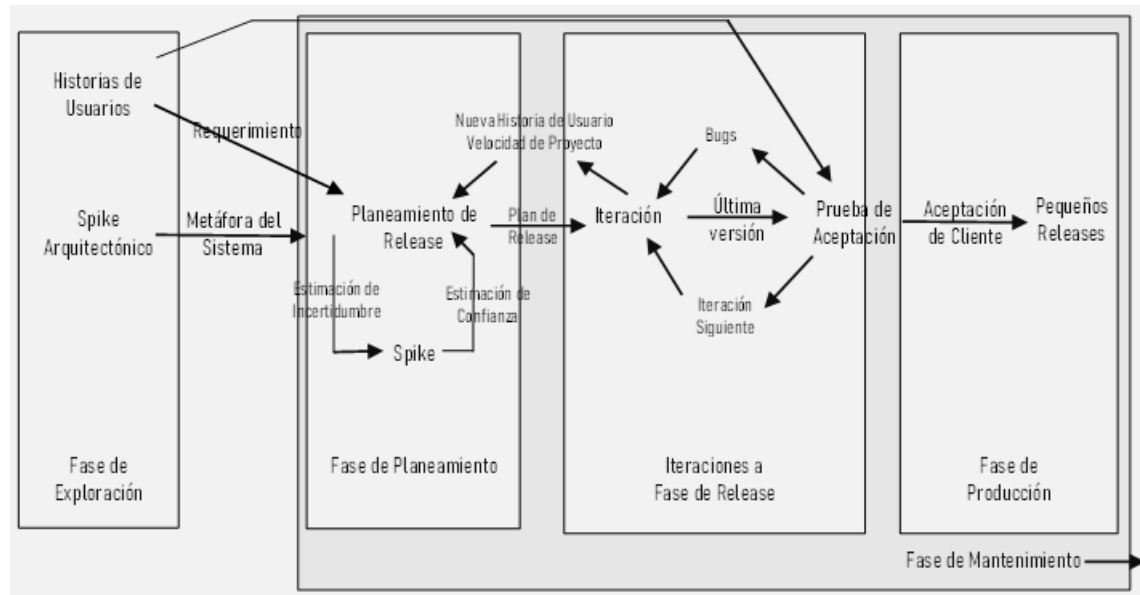
El ciclo de vida de XP enfatiza la naturaleza interactiva de las etapas de desarrollo. Una interacción de desarrollo es el periodo de tiempo durante el cual se realiza un conjunto particular de funciones y, en el caso de XP, corresponde a un conjunto de historias de usuarios.

Cuanto más rápido se entrega el desarrollo al cliente, más comentarios se reciben, lo que se espera que mejore la calidad del producto a largo plazo, por lo que las iteraciones son relativamente cortas.

La siguiente figura muestra las fases en las que se divide el ciclo de vida de Xp.

Figura 3

Ciclo de Vida Programación Extrema



Nota: Fuente (Hurtado, 2005)

Hurtado (2005), describe cada una de las fases en la que se subdivide el ciclo de vida de XP.

a) Fase de Exploración

En esta etapa, el cliente describe la historia de usuario asociada con el

primer lanzamiento del software. Así mismo, los miembros del equipo están habituados con las tecnologías manejadas en el desarrollo del proyecto.

La creación de prototipos prueba la tecnología y explora las posibilidades de la arquitectura del sistema. La fase de la exploración puede durar de semanas a meses, según las competencias del desarrollador y sus destrezas en cuanto al uso de la tecnología.

b) Fase de Planeamiento

Los programadores y los clientes acuerdan priorizar la historia del usuario y el alcance de la primera versión del sistema. El programador estima el esfuerzo requerido para cada historia, a partir del cual se determina el cronograma. La fase de planeamiento toma un par de días. Se deben incluir varias iteraciones para lograr un release.

c) Fase de Desarrollo o Iteración

En cada iteración, el cliente decide qué historia de usuario crear. Estos se dividen en tareas que serán desarrolladas en la próxima versión. Idealmente, el cliente debería realizar una prueba funcional al final de cada iteración para asegurarse de que todo funciona correctamente. Si hay un error, se solucionará en la próxima iteración.

d) Fase de Producción

Se requieren pruebas adicionales y validación del rendimiento del sistema antes del lanzamiento a los clientes. Todavía puede encontrar nuevos cambios en esta fase y decidir si incluirlos en la versión actual. Las ideas y sugerencias pospuestas se documentarán en implementaciones posteriores, como la fase de mantenimiento.

e) Fase de Mantenimiento

Se necesita mucho esfuerzo para lograr la demanda de nuestros clientes. Como resultado, la velocidad de desarrollo puede disminuir después de que el

sistema entre en producción. Durante la fase de mantenimiento, es posible que deba contratar nuevos empleados y cambiar la estructura de su equipo.

f) Fase de Muerte

Hurtado (2005) precisa que "Es cuando el cliente se queda sin historias para incluir en el sistema. Esto requiere cumplir con los requisitos del cliente en otros aspectos, como el rendimiento y la confiabilidad del sistema. Se crea la documentación final del sistema y no se realizan más cambios en la arquitectura. La terminación del proyecto también ocurre si el sistema no produce los beneficios que el cliente espera, o si no hay un presupuesto de mantenimiento".

2.2.5.6 Actores y responsabilidades de XP

XP cumple diferentes roles y responsabilidades para diferentes tareas y propósitos en el proceso.

Según (Extreme Programming – XP, s/f):

Programador: "considerado el más importante miembro del equipo ya que escribe las pruebas unitarias y el código del sistema".

Coach: "Es responsable del proceso global y se encarga de guiar a los miembros del equipo para seguir el proceso correctamente".

Tester: "ayuda al cliente a escribir las pruebas funcionales. Ejecuta pruebas regularmente, difunde los resultados en el equipo y es responsable de las herramientas de soporte para pruebas".

Gestor (Big Boss): "es el dueño de la tienda y el vínculo entre clientes y programadores. Su labor esencial es la coordinación".

Cliente (Customer): "es quien escribe las historias de usuario y las pruebas funcionales para validar su implementación. Asigna la prioridad a las historias

de usuario y decide cuáles se implementan en cada iteración centrándose en aportar el mayor valor de negocio”.

Tracker: “es el encargado de seguimiento. Proporciona retroalimentación al equipo. Debe verificar el grado de acierto entre las estimaciones realizadas y el tiempo real dedicado, comunicando los resultados para mejorar futuras estimaciones”.

Manager: “se encarga de agendar las reuniones, se asegura de que el proceso de juntas sea seguido, registra los resultados de las reuniones para futuros reportes para el Tracker. Asiste a las reuniones y trae información importante, mantiene al equipo feliz y productivo”.

2.2.5.7 Artefactos de XP

a) Historias de Usuario

“Historias de usuario tienen el mismo propósito como casos de uso, pero no son el mismo. Se utilizan para crear las estimaciones de tiempo para la reunión de planificación de entregas. También se utilizan en lugar de un documento de requisitos grande. Las historias de usuarios están escritas por los clientes como las cosas que el sistema necesita para hacer por ellos. Son similares a los escenarios de uso, excepto que no se limitan a describir una interfaz de usuario” (Don Wells, 2009).

Proporcionan una breve descripción de cómo funciona el sistema, usan términos no técnicos del cliente, se crean para cada función clave del sistema, se usan para hacer estimaciones de tiempo y son voluminosos para los planes de lanzamiento. Reemplazan los documentos de requisitos y presiden la creación prueba de aceptación.

Tabla 1

Historia de usuario

Historia de Usuario	
Número:	Usuario:
Nombre de la historia:	
Prioridad en Negocio: (Alta / Media / Baja)	Riesgo en Desarrollo: (Alto / Medio / Bajo)
Esfuerzo:	Iteración Asignada:
Programador Responsable:	
Descripción:	
 Observaciones:	

Nota: Fuente (Canós, 2003).

Las Historias de Usuario tienen tres aspectos:

- i. **Tarjeta.** – Sección donde se registra la información referente a las historias de usuarios.
- ii. **Conversación.** – Los desarrolladores y stakeholders amplían los detalles de las historias de usuario.
- iii. **Pruebas de Aceptación.** – Utilizada para corroborar si las historias de usuario se implementaron de acuerdo al requerimiento del cliente de manera correcta.

Tabla 2

Prueba de aceptación

Caso de Prueba de Aceptación	
Código:	Historia de Usuario (Nro. y Nombre):
Nombre:	
Descripción:	
Condiciones de Ejecución:	
Entrada / Pasos de ejecución:	
Resultado Esperado:	
Evaluación de la Prueba:	

Nota: Fuente (Canós, 2003).

b) Task Card (Tarea de Ingeniería)

Las task card son utilizadas para detallar las actividades que permitirán el cumplimiento de las historias de usuarios; pueden determinarse tareas de desarrollo, corrección, mejora y diseño. Se define la fecha de inicio y fin de la tarea, se designa al desarrollador responsable de ejecutarla y se describe en detalle en qué consiste dicha tarea.

Tabla 3

Tarea de Ingeniería

Tarea de Ingeniería	
Número Tarea:	Historia de Usuario (Nro. y Nombre):
Nombre Tarea:	
Tipo de Tarea:	Puntos Estimados:
Desarrollo / Corrección / Mejora / Otra (especificar)	
Fecha Inicio:	Fecha Fin:
Programador Responsable:	
Descripción:	

Nota: Fuente (Canós, 2003).

c) Tarjetas CRC (Clase, Responsabilidad, Colaborador)

En (Extreme Programming – XP, s/f) se indican que “estas tarjetas se dividen en tres secciones que contienen la información del nombre de la clase, sus responsabilidades y sus colaboradores”.

Tabla 4

Tarjeta CRC

Nombre de la clase.	
Responsabilidades	Colaboradores

Una clase es un conjunto de objetos que tienen atributos y comportamientos comunes; podría concebirse como un sustantivo concreto o abstracto. Las responsabilidades de una clase son las tareas que debe realizar. Los colaboradores de una clase son otras clases que apoyan a cumplir sus responsabilidades.

Capítulo III

Material y métodos

3.1 Tipo investigación

Según (Supo, 2015) “los estudios no intervencionistas denominados también como estudios observacionales, en los que el investigador no toma ninguna medida para cambiar los resultados de la medición; incluso si el investigador hace las mediciones; así, los datos encontrados y la información registrada reflejan el estado natural de las unidades de investigación”.

En otro sentido en palabras de Tinco y Sáenz (1999), la investigación observacional tiene como objetivo una comprensión integral del fenómeno que se estudia en el contexto de la investigación, la caracterización basada en la observación y la descripción y análisis de eventos de interés. En general, este tipo de investigación es ágil y flexible, describiendo, analizando y concluyendo.

Por otro lado, respecto a los estudios transaccionales o transversales, (Hernández et al., 2014) indican que estos recopilan datos en un solo momento. Su objetivo es describir variables, analizar su incidencia y su correlación en un determinado momento.

Al respecto (Hernández M. , 2000), precisa que la investigación transaccional nos proporciona una visión general del estado de una o más variables en uno o más grupos de personas o índices en un momento dado.

Así mismo Supo (2015) precisó “que las variables analíticas son las variables que aparecen en la descripción del estudio. Nuestro estudio es descriptivo cuando sólo existe una variable analítica. Pero si tenemos más de una variable analítica, nuestro estudio es analítico”.

Por las cuestiones expuestas, indicamos que nuestro estudio es observacional, transaccional y descriptivo.

3.2 Nivel de investigación

De acuerdo a (Supo, 2015) "los niveles descriptivos describen fenómenos sociales o clínicos en un contexto geográfico y temporal particular. Los términos clave son tiempo y geografía, porque los eventos descubiertos deben enmarcarse en un espacio o tiempo geográfico".

Según (Hernández, Fernández, & Baptista, 2006) "los estudios descriptivos tienen como objetivo determinar los atributos, características y perfiles de la persona, grupo, comunidad, proceso, objeto o cualquier otro fenómeno que se analiza. Es decir, sólo tienen por objeto medir o recopilar información, de forma independiente o en conjunto, sobre los conceptos o variables a los que se refieren, es decir, su propósito no es indicar su relevancia o relaciones".

En este sentido nuestro estudio es de nivel descriptivo.

3.3 Diseño de la investigación

(Hernández, 2010) precisa al respecto que "la investigación no experimental es una investigación realizada sin manipular intencionalmente las variables independientes, para ver el efecto sobre las otras variables denominadas dependientes. Lo que se hace es simplemente observar los fenómenos a medida que ocurren en su contexto natural y luego analizarlas".

Por su parte Carrasco (2005), en el mismo sentido indica que los diseños no experimentales transversales son aquellos "diseños se emplean para analizar y conocer las características, rasgos, propiedades y cualidades de un hecho o fenómeno de la realidad en un momento determinado de tiempo".

Por lo señalado anteriormente, nuestro estudio contempla un diseño no

experimental del tipo transversal, puesto que no manipulamos ninguna variable y tomamos la información en un determinado momento.

3.4 Población y muestra

3.4.1 Población

La población estuvo compuesta por todos los procesos del área de catastro urbanístico de la Municipalidad Distrital de Hualla.

3.4.2 Muestra

Se consideró el proceso de gestión de catastro urbano de la Municipalidad Distrital de Hualla. El tipo de muestreo fue por juicio de experto.

3.5 Variables e indicadores

3.5.1 Definición conceptual de las variables

Primera variable de estudio

Análisis de sistema de información. Es el proceso de comprensión y definición de que servicios se requieren del sistema y de identificación de las restricciones de funcionamiento y desarrollo de este.

Segunda variable de estudio

Diseño de sistema de información informático. El diseño especifica la solución a un problema o "cómo el software tiene que hacer su función". Los aspectos que se deben considerar son: la arquitectura, las bases de datos, la especificación de cada requisito y las interfaces de usuario.

3.5.2 Definición operacional de los variables

Primera variable de estudio

Análisis de sistema de información

Variables descriptivas

- Historias de usuario
- Datos de entrada y salida

Segunda variable de estudio

Diseño de sistema de información informático

Variables descriptivas

- Tareas de ingeniería
- Diagrama de componentes
- Diagrama de despliegue
- Diseño de la base de datos
- Interfaces de Usuario

3.6 Técnicas e instrumentos para el tratamiento de datos e información

3.6.1 Técnicas para recolectar información

Para la recolección de datos se utilizó la técnica de entrevista.

3.6.2 Instrumentos para recolectar información

El instrumento para la recolección de información fue:

3.6.2.1 Guía de entrevista

Se ha diseñado el instrumento "guía de entrevista", lo cual se muestra en el Anexo B, para la recolección de datos del personal especializado de área de

catastro.

3.6.3 Herramientas para el tratamiento de datos e información

Las tecnologías utilizadas para el trabajo fueron:

Tabla 5

Instrumentos tipo Software utilizado para el tratamiento de los datos

Software	Desarrollador	Descripción
Windows 10	Microsoft	Sistema operativo de soporte para la ejecución de otros programas para ordenador.
Microsoft Office 2016	Microsoft	Paquete de software que brinda herramientas para el procesamiento de textos, hojas de cálculo, soporte para presentaciones entre otros.

3.6.4 Técnicas para aplicar el proceso ágil Programación Extrema

Tabla 6

Fase de exploración

Tarea	Artefacto	Técnica	Responsables
Escribir historias de usuario	Historia de usuario	Describir brevemente las historias de usuario con la regla del negocio (lo que el sistema debe hacer). Dividir historias de usuario grandes.	Cliente
Probar tecnologías utilizar	Arquitectura a técnica inicial	Explorar posibilidades de uso de tecnologías. Probar el rendimiento de las tecnologías Definir las tecnologías a usar	Cliente Programador Entrenador
Estimar esfuerzo para historia de usuario	Plan de alto nivel	Conocer previamente la historia de usuario Estimar esfuerzo (semana) para desarrollar la historia de usuario	Programador

Fuente. Tomado de "comparación de dos procesos de Desarrollo de Software Usando los Métodos ICONIX y XP" Porras, 2010.

Tabla 7*Fase de planeación*

Tarea	Artefacto	Técnica	Responsable
Reescribir las historias de usuario	Historia de usuario	<p>Describir detalladamente la historia de usuario con la regla de negocio</p> <ul style="list-style-type: none"> • Introducir nuevos requisitos del software. • Definir prioridad para cada historia de usuario por necesidad del negocio. • Utilizar técnicas de elaboración del plan de alto nivel. • Estimar y asignar esfuerzo (semana) para cada historia de usuario en función a tiempo para planear, diseñar, implementar y probar. 	Cliente
Formular el plan de versiones	Plan de versión (una iteración)	<ul style="list-style-type: none"> • Estimar y asignar riesgo a cada historia de usuario en función a las situaciones que afectan la estimación del esfuerzo. • Actualizar tarjeta de historia de usuario 	Programador

Fuente. Tomado de "comparación de dos procesos de Desarrollo de Software Usando los Métodos ICONIX y XP" Porras, 2010.

Tabla 8*Fase de iteración*

Tarea	Artefacto	Técnica	Responsables
Definir la arquitectura técnica	Arquitectura técnica	<ul style="list-style-type: none"> • Actualizar la arquitectura técnica inicial. • Usar características del negocio. • Utilizar arquitectura por capas. • Integrar frameworks. 	Cliente Programador Entrenador
Escribir tareas de ingeniería	Tarea de ingeniería	<p>Dividir cada historia de usuario en tareas, describir usando reglas del negocio cada tarea de ingeniería.</p> <p>Estimar y asignar esfuerzo para desarrollar una tarea de ingeniería</p> <p>Asignar una tarea de ingeniería al programador.</p> <ul style="list-style-type: none"> • Utilizar el plan de versión • Actualizar el plan con tareas de ingeniería de la siguiente iteración 	Cliente programador Programador Entrenador Programador
Formular el plan de iteraciones	Plan de iteraciones	<ul style="list-style-type: none"> • Actualizar el cuándo fallo prueba de aceptación • Actualizar el plan con tareas no concluidas • Actualizar las tarjetas de tarea de ingeniería 	Programador Entrenador
Interfaz preliminar	GUI	<p>Diseñar la interfaz preliminar relacionada a cada historia de usuario.</p> <ul style="list-style-type: none"> • Diseñar para una tarea de ingeniería de forma simple • Rediseñar por falla de prueba de aceptación una tarea. 	Cliente Programador
Escribir tarjetas CRC para cada tarea de ingeniería	Tarjeta CRC	<ul style="list-style-type: none"> • Identificar responsabilidades. • Identificar colaboración • Identificar Atributos 	Cliente Programador

Modelar procesos	Diagrama de modelado de procesos	<ul style="list-style-type: none"> Definir las actividades y flujo de trabajo dependiendo del requerimiento de la organización 	Analista de procesos
Implementar código para la clase entidad	Código fuente	<ul style="list-style-type: none"> Escribir código fuente o generar con una herramienta usando tarjetas CRC 	Programador
Crear pruebas unitarias para las clases control	Prueba unitaria	<ul style="list-style-type: none"> Escribir código fuente para una prueba unitaria, usando una herramienta 	Programador
Implementar código fuente	Código fuente	<ul style="list-style-type: none"> Codificar una tarea de ingeniería Hacer refactoring Mover programadores Ejecutar el módulo de cada prueba unitaria 	Programador Supervisor
Ejecutar pruebas unitarias	Código fuente	<ul style="list-style-type: none"> Modificar código fuente si la prueba unitaria muestra resultado incorrecto 	Programador
Realizar integración continúa	Código fuente	<ul style="list-style-type: none"> Integrar las tareas para una historia de usuario Mantener sistema integrado todo el tiempo 	Programador
Realizar pruebas de integración para una historia de usuario	Reporte de pruebas de integración	<ul style="list-style-type: none"> Integrar continuamente al concluir las tareas de una historia de usuario Verificar que las pruebas de integración pasan al 100% 	Programador
Ejecutar pruebas de aceptación	Reporte de pruebas de aceptación	<ul style="list-style-type: none"> Correr la última versión de una iteración Utilizar los casos de prueba de aceptación 	Cliente encargado de pruebas

Fuente. Tomado de "comparación de dos procesos de Desarrollo de Software Usando los Métodos ICONIX y XP" Porras, 2010.

Capítulo IV

Resultados y discusión

4.1 Resultados de la aplicación del Proceso Ágil de Programación Extrema

Utilizando las Técnicas para aplicar el proceso ágil Programación Extrema, mostrada en el capítulo III, sección 3.6.4, se logró determinar los artefactos del análisis y diseño del Sistema de Catastro Urbanístico.

4.1.1 Fase de exploración

Se presenta a continuación los artefactos: historias de usuario, arquitectura inicial y el plan de alto nivel.

4.1.1.1 Historias de usuario

En esta sección se enumera las historias de usuario indispensables que describen los distintos requerimientos para el diseño del Sistema de Catastro Urbanístico.

Tabla 9

Historias de Usuario

N°	Historia de Usuario	Descripción
1	Autenticar Usuario	El sistema debe permitir que un usuario inicie sesión ingresando su usuario y contraseña respectivamente.
2	Gestionar contribuyente	El sistema debe permitir registrar, modificar, buscar, o eliminar los datos de un contribuyente.
3	Gestionar predio	El sistema debe permitir registrar, modificar, buscar, o eliminar los datos de

		un predio.
4	Gestionar declaración jurada de predio	El sistema debe permitir generar la declaración jurada referido a un predio.
5	Gestionar cuentas de usuario	El sistema debe permitir registrar, modificar, buscar, o eliminar los datos de un usuario del sistema.

4.1.1.2 Arquitectura técnica inicial

Figura 4

Arquitectura Técnica Inicial



4.1.1.3 Plan de alto nivel

Tabla 10

Plan de alto nivel

N°	Historia de Usuario	Esfuerzo (en días)
1	Autenticar usuario	5
2	Gestionar contribuyente	12
3	Gestionar predio	13
4	Gestionar declaración jurada de predio	10
5	Gestionar cuentas de usuario	6

4.1.2 Fase de planificación

Según el procedimiento para la fase de planificación, se desarrollan los artefactos como las historias de usuario y plan de versiones, que se muestran a continuación.

4.1.2.1 Historias de usuario

En esta sección se describen de manera detallada las historias de usuario de acuerdo con lo demandado por los requerimientos del usuario.

Tabla 11

Historia de Usuario Autenticar Usuario

Historia de Usuario	
Número: 1	Usuario: Todos
Nombre de historia: Autenticar Usuario	
Prioridad en negocio: Media	Riesgo en desarrollo: Media
Puntos estimados: 1	Iteración asignada: 1
Descripción: Como usuario (técnico de catastro urbanístico) el sistema de información debe permitirme iniciar sesión con mis datos de acceso como nombre de usuario y contraseña, y poder hacer uso del sistema.	
Observaciones: Ninguna.	

Tabla 12

Historia de Usuario Gestionar Contribuyente

Historia de Usuario	
Número: 2	Usuario: Todos
Nombre de historia: Gestionar Contribuyente	
Prioridad en negocio: Alta	Riesgo en desarrollo: Media
Puntos estimados: 2	Iteración asignada: 1
Descripción: Como usuario (técnico de catastro urbanístico) el sistema de información debe permitirme registrar, modificar, eliminar y buscar un contribuyente, así también generar un reporte de los contribuyentes registrados.	

Observaciones: Ninguna.

Tabla 13

Historia de Usuario Gestionar Predio

Historia de Usuario	
Número: 3	Usuario: Todos
Nombre de historia: Gestionar Predio	
Prioridad en negocio: Alta	Riesgo en desarrollo: Media
Puntos estimados: 2	Iteración asignada: 1
Descripción: Como usuario (técnico de catastro urbanístico) el sistema de información debe permitirme registrar, modificar, eliminar y buscar un predio, así también generar un reporte de los predios registrados.	
Observaciones: Ninguna.	

Tabla 14

Historia de Usuario Gestionar declaración jurada de impuesto predial

Historia de Usuario	
Número: 4	Usuario: Administrador
Nombre de historia: Gestionar declaración jurada de impuesto predial	
Prioridad en negocio: Alta	Riesgo en desarrollo: Media
Puntos estimados: 2	Iteración asignada: 1
Descripción: Como usuario (técnico de catastro urbanístico) el sistema de información debe permitirme registrar, modificar, eliminar y buscar una declaración jurada de impuesto predial.	
Observaciones: Ninguna.	

Tabla 15

Historia de Usuario Gestionar Cuentas de Usuario

Historia de Usuario	
Número: 5	Usuario: Administrador
Nombre de historia: Gestionar cuenta de usuario	
Prioridad en negocio: Alta	Riesgo en desarrollo: Media
Puntos estimados: 1	Iteración asignada: 1
Descripción: Como usuario administrador el sistema de información debe permitirme registrar, modificar, eliminar y buscar un usuario para el sistema.	
Observaciones: Ninguna.	

4.1.2.2 Plan de versiones

Se precisan las prioridades para cada historia de usuario por urgencia del negocio, el riesgo asociado de acuerdo a la complejidad del requerimiento, el esfuerzo en días, esto de acuerdo también a la complejidad y experiencia del desarrollador y finalmente se asigna el número de iteraciones en el que se logrará completar las historias de usuarios.

Tabla 16*Plan de Versiones*

N°	Historia de Usuario	Prioridad	Riesgo	Esfuerzo (en días)	Iteración
1	Autenticar usuario	Medio	Bajo	5	1
2	Gestionar contribuyente	Alto	Medio	12	1
3	Gestionar predio	Alto	Medio	13	1
4	Gestionar declaración jurada de predio	Alto	Medio	10	1
5	Gestionar cuentas de usuario	Medio	Bajo	6	1

4.1.3 Fase de iteración

Determinamos en esta etapa los siguientes artefactos: arquitectura técnica final en el que se considera, tareas de ingeniería, plan de iteración, Interfaz gráfica de usuario, tarjetas CRC, base de datos lógica.

4.1.3.1 Arquitectura técnica final

Figura 5

Arquitectura técnica final (Diagrama de componentes)

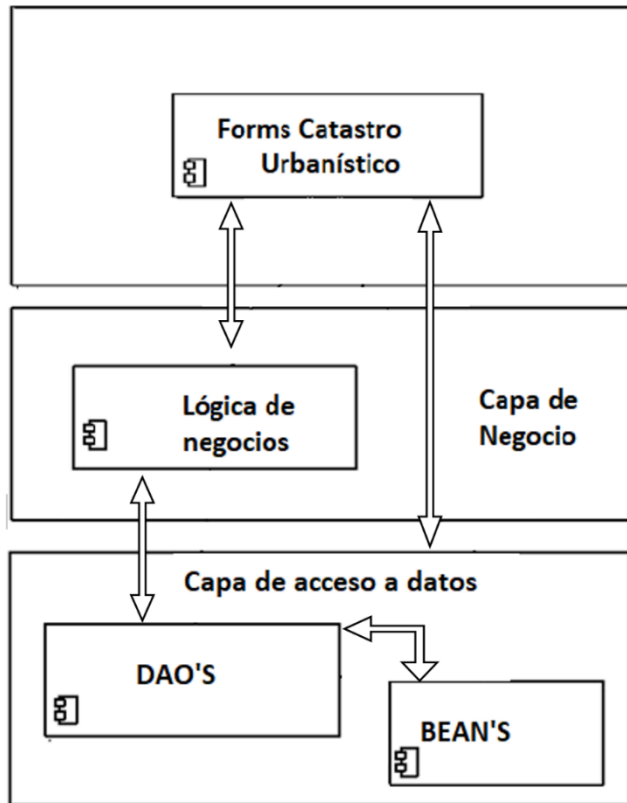
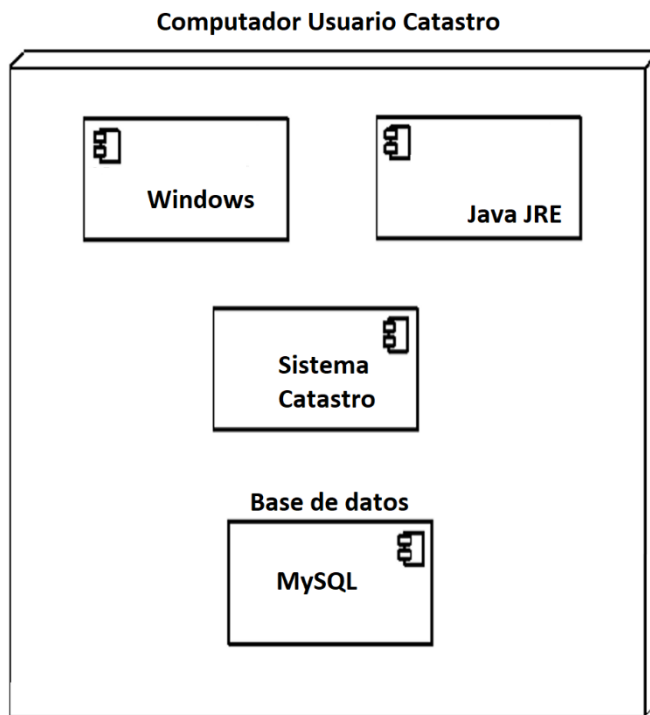


Figura 6

Arquitectura técnica final (Diagrama de despliegue)



4.1.3.2 Plan de iteración

Se muestra en seguida la organización de las distintas tareas de ingeniería para cada historia de usuario que deberían desarrollarse en una iteración.

Tabla 17

Plan de Iteración (Primera)

N°	Historia de Usuario	Tarea de Ingeniería
1	Autenticar usuario	Autenticar usuario
2	Gestionar contribuyente	Registrar contribuyente Modificar Contribuyente Eliminar Contribuyente Buscar contribuyente
3	Gestionar predio	Registrar Predio Modificar predio

		Eliminar predio
		Buscar predio
4	Gestionar declaración de predio	Registrar declaración jurada de predio
		Buscar declaración jurada de predio
		Imprimir declaración jurada de predio
5	Gestionar cuentas de usuario	Registrar cuenta de usuario
		Modificar cuenta de usuario

4.1.3.3 Plan de iteración por fechas de desarrollo

Tabla 18

Plan de iteración por fechas de desarrollo

N°	Historia de Usuario	Tarea de Ingeniería	Fecha		Programador
			Inicio	Fin	
1	Autenticar usuario	Autenticar usuario			
2	Gestionar contribuyente	Registrar contribuyente			
		Modificar Contribuyente			
		Eliminar Contribuyente			
		Buscar contribuyente			
3	Gestionar predio	Registrar Predio			
		Modificar predio			
		Eliminar predio			

			Buscar predio
4	Gestionar declaración jurada predio	de	Registrar declaración jurada de predio Buscar declaración jurada de predio Imprimir declaración jurada de predio
5	Gestionar cuentas usuario	de	Registrar cuenta de usuario Modificar cuenta de usuario

4.1.3.4 Tareas de ingeniería

En esta sección se describen las tareas de ingeniería del tipo desarrollo, las que deberán dar cumplimiento a las Historias de Usuario demandadas por los usuarios del sistema de información.

4.1.3.4.1 Historia de Usuario: Autenticar usuario

Tabla 19

Tarea de Ingeniería Autenticar usuario

Tarea de Ingeniería	
Número de tarea: 1	Número de historia de usuario: 1
Nombre de tarea: Autenticar usuario	
Tipo de tarea: Desarrollo	Puntos estimados: 1
Fecha Inicio:	Fecha fin:
Programador responsable:	
Descripción:	
<p>El usuario en la ventana de inicio de sesión ingresa el nombre de usuario y contraseña, luego hace clic en el botón "acceder".</p> <p>El sistema verifica que los datos proporcionados por el usuario sean los registrados previamente en la base de datos, en el caso que los datos correspondan, el sistema muestra la ventana principal al usuario; en caso contrario muestra un mensaje indicando que los datos ingresados son incorrectos.</p>	
Observaciones:	
<p>Si el usuario no ingresa uno de los datos necesarios, ya sea el nombre de usuario o la contraseña, el sistema mostrará un mensaje indicando qué dato faltó ingresar.</p> <p>Los datos de los usuarios están registrados previamente en la base de datos.</p>	

4.1.3.4.2 Historia de Usuario: Gestionar contribuyente

Tabla 20

Tarea de Ingeniería Registrar contribuyente

Tarea de Ingeniería	
Número de tarea: 2	Número de historia de usuario: 2
Nombre de tarea: Registrar Contribuyente	
Tipo de tarea: Desarrollo	Puntos estimados: 1
Fecha Inicio:	Fecha fin:
Programador responsable:	
Descripción:	
El usuario selecciona la opción "contribuyente" de la ventana principal.	
El sistema muestra la ventana "gestión de contribuyentes", con una lista de los últimos diez contribuyentes registrados.	
El usuario (personal de catastro urbanístico), hace clic en el botón "registrar", el sistema muestra la ventana "Registro Contribuyente", con los campos obligatorios: DNI, apellido paterno, apellido materno, nombres, dirección, teléfono, tipo de declarante, y la condición de domicilio.	
El usuario ingresa los campos solicitados y hace clic en el botón "guardar", el sistema valida los datos ingresados, registra el contribuyente en la base de datos y muestra el mensaje "Los datos del contribuyente se registraron correctamente" y limpia el formulario de los datos Contribuyente registrado.	
Observaciones:	
Si el usuario no ingresa todos los datos solicitados: el sistema muestra un mensaje de "dato requerido" en cada campo que se encuentra vacío.	

Tabla 21*Tarea de Ingeniería Buscar Contribuyente*

Tarea de Ingeniería	
Número de tarea: 5	Número de historia de usuario: 2
Nombre de tarea: Buscar Contribuyente	
Tipo de tarea: Desarrollo	Puntos estimados: 1
Fecha Inicio:	Fecha fin:
Programador responsable:	
Descripción:	
El usuario selecciona la opción "contribuyente" de la ventana principal.	
El sistema muestra la ventana "gestión de contribuyentes", con una lista de los últimos diez contribuyentes registrados.	
El usuario (personal de catastro urbanístico), ingresa el DNI o nombres y apellidos del contribuyente en la caja de texto de búsqueda y hace clic en el botón "buscar".	
El sistema realiza una búsqueda en la base de datos de la información del contribuyente ingresado, mostrando un mensaje de "contribuyente encontrado" y mostrando todos los datos del registro del contribuyente encontrado.	
Observaciones:	
En caso no existiera ninguna coincidencia en la base de datos, del contribuyente buscado, el sistema muestra un mensaje de "No hay registros del contribuyente".	

Tabla 22

Tarea de Ingeniería Modificar contribuyente

Tarea de Ingeniería	
Número de tarea: 3	Número de historia de usuario: 2
Nombre de tarea: Modificar contribuyente	
Tipo de tarea: Desarrollo	Puntos estimados: 1
Fecha Inicio:	Fecha fin:
Programador responsable:	
Descripción:	
El usuario selecciona la opción "contribuyente" de la ventana principal.	
El sistema muestra la ventana "gestión de contribuyentes"	
El usuario ingresa un dato como (dni ó nombres y apellidos) del contribuyente que desea modificar en la caja de texto buscar y hace clic en el botón buscar.	
El sistema muestra una lista de todos los contribuyentes de la base de datos que coinciden con el dato de búsqueda ingresado.	
El usuario hace clic en el registro del contribuyente que desea editar.	
El sistema muestra la ventana "contribuyente" con la información del contribuyente y los campos habilitados para la modificación (Apellido Paterno, Apellido Materno, Nombres, tipo de Tipo de Declarante y la condición de domicilio), además los botones de guardar y cancelar.	
El usuario modifica los campos que desea y hace clic en guardar.	
El sistema valida la información y guarda los cambios, finalmente muestra mensaje de confirmación "registro de contribuyente modificado satisfactoriamente".	
Observaciones:	
En caso no existiera ninguna coincidencia del contribuyente buscado, el sistema muestra un mensaje de "No hay registros del contribuyente".	

Tabla 23

Tarea de Ingeniería Eliminar Contribuyente

Tarea de Ingeniería	
Número de tarea: 4	Número de historia de usuario: 2
Nombre de tarea: Eliminar Contribuyente	
Tipo de tarea: Desarrollo	Puntos estimados: 1
Fecha Inicio:	Fecha fin:
Programador responsable:	
Descripción:	
<p>El usuario ingresa al menú "Contribuyente" y realiza la búsqueda ingresando en la caja de texto de búsqueda el DNI o nombres del contribuyente y haciendo clic en buscar.</p> <p>La aplicación devuelve la lista de contribuyentes registrados en la base de datos.</p> <p>El usuario selecciona el contribuyente que desea eliminar. Luego hace clic en el botón "eliminar".</p> <p>El sistema muestra un mensaje de confirmación si se desea eliminar el contribuyente seleccionado, el usuario hace clic en el botón "confirmar", finalmente el sistema muestra un mensaje "Contribuyente eliminado" y elimina el registro del contribuyente de la base de datos.</p>	
Observaciones:	
<p>Si el contribuyente buscado no se encuentra en la base de datos, el sistema muestra un mensaje "No hay registros del contribuyente".</p>	

4.1.3.4.3 Historia de Usuario: Gestionar predio

Tabla 24

Tarea de Ingeniería Registrar predio

Tarea de Ingeniería	
Número de tarea: 6	Número de historia de usuario: 3
Nombre de tarea: Registrar Predio	
Tipo de tarea: Desarrollo	Puntos estimados: 1
Fecha Inicio:	Fecha fin:
Programador responsable:	
Descripción:	
El usuario selecciona la opción "predio" de la ventana principal.	
El sistema muestra la ventana "gestión de predios", con una lista de los últimos diez predios registrados.	
El usuario (personal de catastro urbanístico), hace clic en el botón "registrar", el sistema muestra la ventana "Registro de Predio" con el campo de búsqueda del contribuyente; campos del contribuyente: Nombres y apellidos; campos sobre las características del predio: código de registro, tipo de predio, la forma de adquisición, uso del predio, año de adquisición, área del predio, valor del predio; campos sobre la ubicación del predio: distrito, código de zona, tipo de zona, nombre de zona, código de vía, tipo de vía, nombre de vía, manzana y lote.	
El usuario ingresa en la caja de texto de búsqueda del contribuyente, el DNI o nombres y apellidos de éste, y hace clic en el botón "buscar"	
El sistema muestra los nombres y apellidos del contribuyente previamente registrado en la base de datos.	
El sistema genera el código de registro del predio.	
El usuario ingresa los datos de las características del predio, así como sus datos de ubicación y hace clic en el botón "guardar", el sistema valida los datos ingresados, registra los datos del predio en la base de datos y muestra el mensaje "Los datos	

del predio se registraron correctamente" y limpia el formulario de los datos del Predio registrado.

Observaciones:

Si el usuario no ingresa los datos obligatorios, el sistema muestra un mensaje de "dato requerido" en cada campo obligatorio.

Tabla 25

Tarea de Ingeniería Buscar Predio

Tarea de Ingeniería	
Número de tarea: 9	Número de historia de usuario: 3
Nombre de tarea: Buscar Predio	
Tipo de tarea: Desarrollo	Puntos estimados: 1
Fecha Inicio:	Fecha fin:
Programador responsable:	
Descripción:	
El usuario selecciona la opción "predio" de la ventana principal.	
El sistema muestra la ventana "gestión de predios", con una lista de los últimos diez predios registrados.	
El usuario (personal de catastro urbanístico) ingresa el DNI o nombres y apellidos del contribuyente dueño del predio a buscar en la caja de texto de búsqueda y hace clic en el botón "buscar".	
El sistema realiza una búsqueda en la base de datos y muestra una lista de los predios registrados por el contribuyente.	
Observaciones:	
En caso no existiera predios registrados por el contribuyente, el sistema muestra un mensaje de "No hay predios registrados por el contribuyente".	

Tabla 26*Tarea de Ingeniería Modificar predio*

Tarea de Ingeniería	
Número de tarea: 7	Número de historia de usuario: 3
Nombre de tarea: Modificar predio	
Tipo de tarea: Desarrollo	Puntos estimados: 1
Fecha Inicio:	Fecha fin:
Programador responsable:	
Descripción:	
El usuario selecciona la opción "predio" de la ventana principal.	
El sistema muestra la ventana "gestión de predios", con una lista de los últimos diez predios registrados.	
El usuario (personal de catastro urbanístico) ingresa el DNI o nombres y apellidos del contribuyente dueño del predio a buscar en la caja de texto de búsqueda y hace clic en el botón "buscar".	
El sistema realiza una búsqueda en la base de datos y muestra una lista de los predios registrados por el contribuyente.	
El usuario hace clic en el predio que desea modificar.	
El sistema muestra la ventana "Predio" con la información del predio y los campos habilitados para la modificación, además los botones de "guardar" y "cancelar".	
El usuario modifica los campos que desea modificar del predio seleccionado y hace clic en "guardar".	
El sistema valida la información y guarda los cambios, finalmente muestra mensaje de confirmación "Registro de predio modificado satisfactoriamente".	
Observaciones:	
En caso el usuario no ingrese todos los campos obligatorios, el sistema mostrará un mensaje "Dato obligatorio" para cada campo obligatorio sin rellenar.	

Tabla 27*Tarea de Ingeniería Eliminar Predio*

Tarea de Ingeniería	
Número de tarea: 8	Número de historia de usuario: 3
Nombre de tarea: Eliminar Predio	
Tipo de tarea: Desarrollo	Puntos estimados: 1
Fecha Inicio:	Fecha fin:
Programador responsable:	
Descripción:	
El usuario selecciona la opción "predio" de la ventana principal.	
El sistema muestra la ventana "gestión de predios", con una lista de los últimos diez predios registrados.	
El usuario (personal de catastro urbanístico) ingresa el DNI o nombres y apellidos del contribuyente dueño del predio a eliminar en la caja de texto de búsqueda y hace clic en el botón "buscar".	
El sistema realiza una búsqueda en la base de datos y muestra una lista de los predios registrados por el contribuyente.	
El usuario selecciona el predio que desea eliminar. Luego hace clic en el botón "eliminar".	
El sistema muestra un mensaje de confirmación si se desea eliminar el predio seleccionado, el usuario hace clic en el botón "confirmar", finalmente el sistema muestra un mensaje "Predio eliminado" y elimina el registro del predio de la base de datos.	
Observaciones:	
En caso no existiera predios registrados por el contribuyente, el sistema muestra un mensaje de "No hay predios registrados por el contribuyente".	

4.1.3.4.4 Historia de Usuario: Gestionar declaración jurada de predio

Tabla 28

Tarea de Ingeniería Generar Declaración Jurada de Predio

Tarea de Ingeniería	
Número de tarea: 10	Número de historia de usuario: 4
Nombre de tarea: Registrar Declaración Jurada de Predio	
Tipo de tarea: Desarrollo	Puntos estimados: 1
Fecha Inicio:	Fecha fin:
Programador responsable:	
Descripción:	
El usuario selecciona la opción "declaración jurada" de la ventana principal.	
El sistema muestra la ventana "gestión de declaración jurada de predio", con una lista de las últimas diez declaraciones juradas generadas.	
El usuario hace clic en el botón "registrar"	
El usuario (personal de catastro urbanístico) ingresa el DNI o nombres y apellidos del contribuyente dueño del predio a buscar en la caja de texto de búsqueda y hace clic en el botón "buscar".	
El sistema muestra los nombres y apellidos del contribuyente.	
El sistema realiza una búsqueda en la base de datos y muestra una lista de los predios registrados por el contribuyente en el combo predios del contribuyente.	
El usuario selecciona uno de los predios registrados por el contribuyente, del cual desea generar la declaración jurada.	
El sistema genera la fecha y número para la declaración jurada, carga cada uno de los datos del predio seleccionado.	
El usuario finalmente hace clic en el botón "guardar".	
El sistema guarda la declaración jurada del predio en la base de datos.	
Observaciones:	

En caso no existiera datos del contribuyente, el sistema muestra un mensaje de "No hay contribuyentes con el dato de búsqueda ingresado".

En caso no existiera predios registrados por el contribuyente, el sistema muestra un mensaje de "No hay predios registrados por el contribuyente".

Tabla 29

Tarea de Ingeniería Buscar Declaración Jurada de predio

Tarea de Ingeniería

Número de tarea: 11

Número de historia de usuario: 4

Nombre de tarea: Buscar Declaración Jurada de predio

Tipo de tarea: Desarrollo

Puntos estimados: 1

Fecha Inicio:

Fecha fin:

Programador responsable:

Descripción:

El usuario selecciona la opción "Declaración jurada" de la ventana principal.

El sistema muestra la ventana "Gestión de declaración jurada del precio", con una lista de las últimas diez declaraciones juradas registradas.

El usuario (personal de catastro urbanístico) ingresa el DNI o nombres y apellidos del contribuyente dueño del predio a buscar en la caja de texto de búsqueda y hace clic en el botón "buscar".

El sistema realiza una búsqueda en la base de datos y muestra una lista de las declaraciones juradas registradas para el contribuyente.

Observaciones:

En caso no existiera declaraciones juradas registradas para el contribuyente, el sistema muestra un mensaje de "No hay declaraciones juradas registradas para el contribuyente".

Tabla 30

Tarea de Ingeniería Imprimir Declaración Jurada de predio

Tarea de Ingeniería	
Número de tarea: 12	Número de historia de usuario: 4
Nombre de tarea: Imprimir Declaración Jurada de predio	
Tipo de tarea: Desarrollo	Puntos estimados: 1
Fecha Inicio:	Fecha fin:
Programador responsable:	
Descripción:	
El usuario selecciona la opción "Declaración jurada" de la ventana principal.	
El sistema muestra la ventana "Gestión de declaración jurada del precio", con una lista de las últimas diez declaraciones juradas registradas.	
El usuario (personal de catastro urbanístico) ingresa el DNI o nombres y apellidos del contribuyente dueño del predio a buscar en la caja de texto de búsqueda y hace clic en el botón "buscar".	
El sistema realiza una búsqueda en la base de datos y muestra una lista de las declaraciones juradas registradas para el contribuyente.	
El usuario selecciona la declaración jurada del cual desea realizar la impresión.	
El sistema carga la ventana "Impresión declaración jurada de predio" con todos los datos de la declaración jurada y hace clic en el botón "Imprimir".	
El sistema imprime la declaración jurada correspondiente por la impresora de turno.	
Observaciones:	
En caso no existiera declaraciones juradas registradas para el contribuyente, el sistema muestra un mensaje de "No hay declaraciones juradas registradas para el contribuyente".	

4.1.3.4.5 Historia de Usuario: Gestionar cuentas de usuario

Tabla 31

Tarea de Ingeniería Registrar Cuenta de usuario

Tarea de Ingeniería	
Número de tarea: 13	Número de historia de usuario: 5
Nombre de tarea: Registrar Cuenta de Usuario	
Tipo de tarea: Desarrollo	Puntos estimados: 1
Fecha Inicio:	Fecha fin:
Programador responsable:	
Descripción:	
<p>El usuario (Administrador) selecciona la opción "Sistema" de la ventana principal. El sistema muestra la ventana "Gestión de usuarios", con una lista de los usuarios registrados.</p> <p>El usuario(administrador), hace clic en el botón "registrar", el sistema muestra la ventana "Registro de usuario" con los campos obligatorios nombre de usuario y contraseña.</p> <p>El usuario ingresa los datos necesarios y hace clic en el botón "guardar", el sistema valida los datos ingresados, registra los datos del nuevo usuario en la base de datos y muestra el mensaje "Los datos del usuario se registraron correctamente" y limpia el formulario de los datos del usuario registrado.</p>	
Observaciones:	
<p>Si el usuario no ingresa los datos obligatorios, el sistema muestra un mensaje de "dato requerido" en cada campo obligatorio.</p>	

Tabla 32*Tarea de Ingeniería Modificar Cuenta de Usuario*

Tarea de Ingeniería	
Número de tarea: 14	Número de historia de usuario: 5
Nombre de tarea: Modificar Cuenta de Usuario	
Tipo de tarea: Desarrollo	Puntos estimados: 1
Fecha Inicio:	Fecha fin:
Programador responsable:	
Descripción:	
El usuario (Administrador) selecciona la opción "Sistema" de la ventana principal.	
El sistema muestra la ventana "Gestión de usuarios", con una lista de los usuarios registrados.	
El usuario(administrador), selecciona el usuario que desea modificar y hace clic en el botón "modificar"	
El sistema muestra la ventana "Modificar Cuenta de Usuario" con la información de la cuenta de usuario y los campos habilitados para la modificación, además los botones de "guardar" y "cancelar".	
El usuario modifica los campos que desea modificar de la cuenta de usuario seleccionado y hace clic en "guardar".	
El sistema valida la información y guarda los cambios, finalmente muestra mensaje de confirmación "Cuenta de usuario modificada satisfactoriamente".	
Observaciones:	
En caso el usuario no ingrese todos los campos obligatorios, el sistema mostrará un mensaje "Dato obligatorio" para cada campo obligatorio sin rellenar.	

4.1.3.5 Interfaz gráfica de usuario

Figura 7

Interfaz principal del sistema de información



Figura 8

Interfaz de autenticación del usuario para el sistema

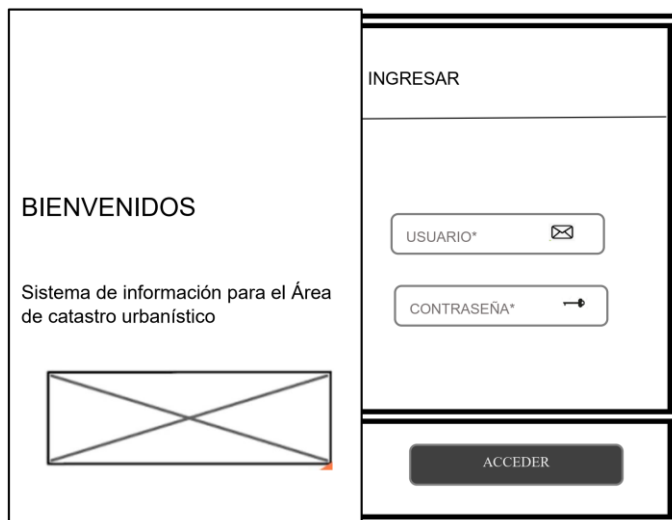


Figura 9

Interfaz para gestión de contribuyentes

	DNI	APELLIDO PATERNO	APELLIDO MATERNO	NOMBRES	GÉNERO	CORREO	TIPO DE DECLARANTE	CONDICIÓN DE DOMICILIO
<input type="checkbox"/>	XXX	YYY	VVV	ZZZ	F	Xxx	Mmm	Nnn
<input type="checkbox"/>	XXX	YYY	VVV	ZZZ	F	Xxx	Mmm	Nnn
<input type="checkbox"/>	XXX	YYY	VVV	ZZZ	F	Xxxx	Mmm	Nnn
<input type="checkbox"/>	XXX	YYY	VVV	ZZZ	F	Xxx	Mmm	Nnn
<input type="checkbox"/>	XXX	YYY	VVV	ZZZ	F	Xxx	Mmm	Nnn
<input type="checkbox"/>	XXX	YYY	VVV	ZZZ	F	Xxx	Mmm	Nnn
<input type="checkbox"/>	XXX	YYY	VVV	ZZZ	F	Xxx	Mmm	Nnn
<input type="checkbox"/>	XXX	YYY	VVV	ZZZ	F	Xxx	Mmm	Nnn

Figura 10

Interfaz para registro de contribuyentes

REGISTRO DE CONTRIBUYENTE

DNI CONTRIBUTUYENTE

APELLIDO PATERNO

APELLIDO MATERNO

NOMBRES

TIPO DE DECLARANTE ▼

CONDICIÓN DE DOMICILIO ▼

DOMICILIO

TELÉFONO

GUARDAR NUEVO SALIR

Figura 11

Interfaz para gestión de cuentas de predios

CÓDIGO	TIPO	FORMA DE ADQUISICIÓN	USO	AÑO DE ADQUISICIÓN	ÁREA DEL PREDIO	VALOR	DISTRITO	CÓDIGO DE ZONA	VÍA	MANZANA	
<input type="checkbox"/>	XXXX	YYYY	ZZZ	AAAA	BBBB	QQQQ	CCCC	PPPP	FFFF	MMM	HHHH
<input type="checkbox"/>	XXXX	YYYY	ZZZ	AAAA	BBBB	QQQQ	CCCC	PPPP	FFFF	MMM	HHHH
<input type="checkbox"/>	XXXX	YYYY	ZZZ	AAAA	BBBB	QQQQ	CCCC	PPPP	FFFF	MMM	HHHH
<input type="checkbox"/>	XXXX	YYYY	ZZZ	AAAA	BBBB	QQQQ	CCCC	PPPP	FFFF	MMM	HHHH
<input type="checkbox"/>	XXXX	YYYY	ZZZ	AAAA	BBBB	QQQQ	CCCC	PPPP	FFFF	MMM	HHHH
<input type="checkbox"/>	XXXX	YYYY	ZZZ	AAAA	BBBB	QQQQ	CCCC	PPPP	FFFF	MMM	HHHH
<input type="checkbox"/>	XXXX	YYYY	ZZZ	AAAA	BBBB	QQQQ	CCCC	PPPP	FFFF	MMM	HHHH

Figura 12

Interfaz para registro de predios

REGISTRO DE PREDIO

DNI Contribuyente:

Apellidos y nombres del Contribuyente

Código de registro

CARACTERÍSTICAS DEL PREDIO

Tipo de predio:

Forma de adquisición:

Uso del predio:

Año de adquisición:

Área del predio (m2):

Valor del predio:

DATOS DE UBICACIÓN DEL PREDIO

Distrito:

Código zona:

Tipo de zona:

Nombre de zona:

Código de vía:

Tipo de vía:

Nombre de vía:

Manzana:

Lote:

Figura 13

Interfaz para gestión de declaraciones juradas de predio

FECHA	CÓDIGO	CÓDIGO PREDIO	TIPO PREDIO	FORMA DE ADQUISICIÓN	USO	AÑO DE ADQUISICIÓN	ÁREA DEL PREDIO	VALOR	DISTRITO	
<input type="checkbox"/>	XXXX	YYYY	ZZZ	AAAA	BBBB	QQQQ	CCCC	PPPP	FFFF	MMM
<input type="checkbox"/>	XXXX	YYYY	ZZZ	AAAA	BBBB	QQQQ	CCCC	PPPP	FFFF	MMM
<input type="checkbox"/>	XXXX	YYYY	ZZZ	AAAA	BBBB	QQQQ	CCCC	PPPP	FFFF	MMM
<input type="checkbox"/>	XXXX	YYYY	ZZZ	AAAA	BBBB	QQQQ	CCCC	PPPP	FFFF	MMM
<input type="checkbox"/>	XXXX	YYYY	ZZZ	AAAA	BBBB	QQQQ	CCCC	PPPP	FFFF	MMM
<input type="checkbox"/>	XXXX	YYYY	ZZZ	AAAA	BBBB	QQQQ	CCCC	PPPP	FFFF	MMM
<input type="checkbox"/>	XXXX	YYYY	ZZZ	AAAA	BBBB	QQQQ	CCCC	PPPP	FFFF	MMM

Figura 14

Interfaz para registro de declaraciones juradas de predio

REGISTRO DE DECLARACIONES JURADAS

DNI CONTRIBUYENTE

Apellido y Nombres del Contribuyente:

SELECCIONE PREDIO REGISTRADO ▼

CODIGO DECLARACIÓN JURADA: XXX FECHA: XX/XX/XX HORA:

TIPO DE PREDIO

FORMA DE ADQUISICIÓN

USO DEL PREDIO

AÑO DE ADQUISICIÓN

ÁREA DEL PREDIO (m2)

VALOR DEL PREDIO

DISTRITO

CÓDIGO ZONA

TIPO DE ZONA

NOMBRE DE ZONA

CÓDIGO DE VÍA

TIPO DE VÍA

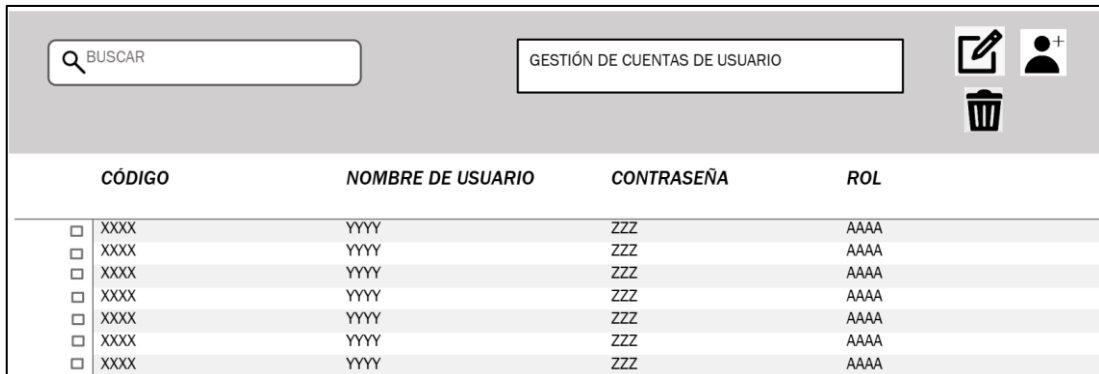
NOMBRE DE VÍA

MANZANA

LOTE

Figura 15

Interfaz para gestión de cuentas de usuario



	CÓDIGO	NOMBRE DE USUARIO	CONTRASEÑA	ROL
<input type="checkbox"/>	XXXX	YYYY	ZZZ	AAAA
<input type="checkbox"/>	XXXX	YYYY	ZZZ	AAAA
<input type="checkbox"/>	XXXX	YYYY	ZZZ	AAAA
<input type="checkbox"/>	XXXX	YYYY	ZZZ	AAAA
<input type="checkbox"/>	XXXX	YYYY	ZZZ	AAAA
<input type="checkbox"/>	XXXX	YYYY	ZZZ	AAAA
<input type="checkbox"/>	XXXX	YYYY	ZZZ	AAAA


Figura 16

Interfaz para registro de cuentas de usuario



NUEVA CUENTA DE USUARIO

NOMBRE 

CONTRASEÑA 

ROL

GUARDAR SALIR

4.1.3.6 Tarjetas clase responsabilidad colaboración (CRC)

En esta sección se hacen un intento para la elaboración de las tarjetas CRC, cada tarjeta expresa una entidad importante del sistema u objeto del mundo real dentro de un contexto determinado, debemos comprender que un conjunto de objetos de atributos y responsabilidades en común definen lo que se denomina Clase. Se determinan los atributos o características de estas clases, las responsabilidades de ésta y los colaboradores con los cuales tendría que interactuar durante la fase de desarrollo.

Tabla 33*Clase Contribuyente*

Nombre de clase: Contribuyente	
Atributos	Responsabilidades:
Dni	Registrar contribuyente
Apellido paterno	Buscar contribuyente
Apellido materno	Modificar contribuyente
Nombres	Eliminar contribuyente
Tipo de declarante	
Condición de domicilio	
	Colaboradores:
	Usuario
	Predio
	Declaración Jurada

Tabla 34*Clase Predio*

Nombre de clase: Predio	
Atributos	Responsabilidades:
Nombre contribuyente	Registrar predio
Apellido contribuyente	Buscar predio
Código de registro	Modificar predio
Tipo de predio	Eliminar predio
Forma de adquisición	
Uso del predio	
Año de adquisición	
Área del predio	Colaboradores:
Valor del predio	Contribuyente
	Declaración Jurada

Tabla 35*Clase Ubicación*

Nombre de clase: Ubicación	
Atributos: Distrito Código de zona Tipo de zona Nombre de zona Código de vía Tipo de vía Nombre de vía Lote Manzana	Responsabilidades: Registra la dirección del predio Colaboradores: Contribuyente Predio

Tabla 36*Clase Declaración Jurada*

Nombre de clase: Declaración Jurada	
Atributos: Número Fecha DNI Contribuyente DatosPredio	Responsabilidades: Registra Declaración Jurada Buscar Declaración Jurada Imprimir Declaración Jurada Colaboradores: Predio Contribuyente

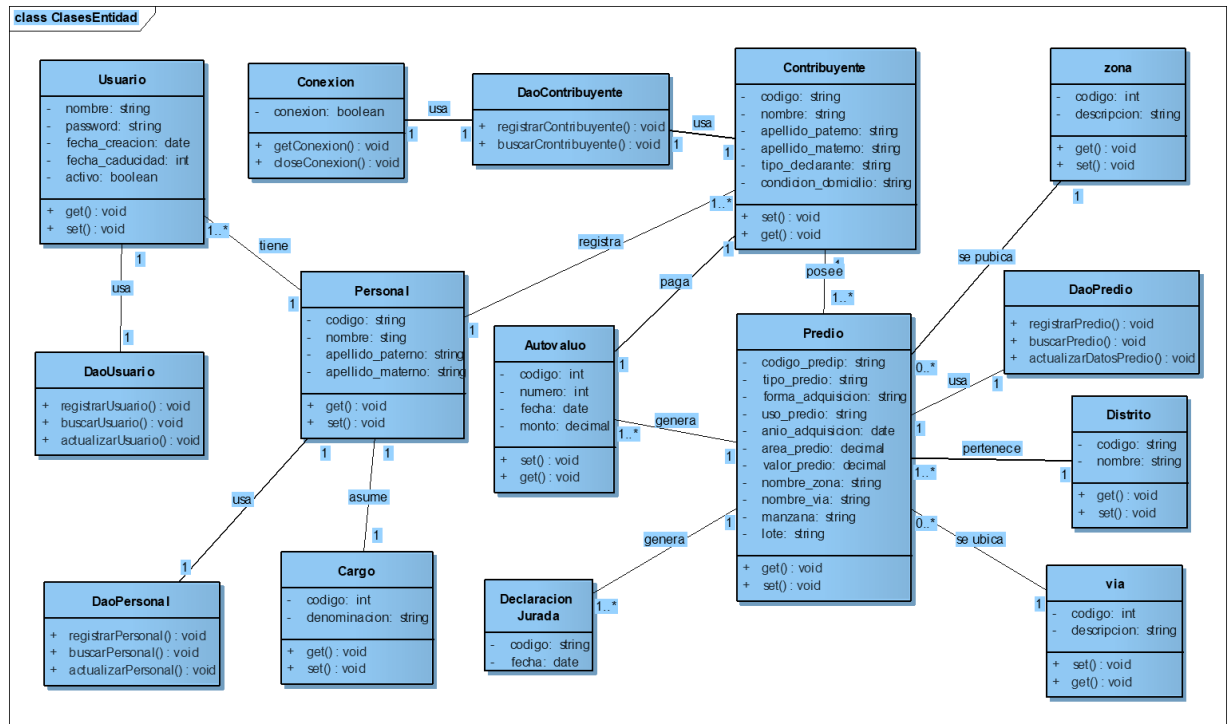
Tabla 37*Clase Usuario*

Nombre de clase: Usuario	
Atributos: NombreUsuario Contraseña Estado	Responsabilidades: Registrar usuario Listar usuario Autenticar usuario Colaboradores:

4.1.3.7 Diseño lógico de la base de datos

Figura 17

Diseño lógico de la base de datos



4.2 Discusión de resultados

En base a las entrevistas con el personal de la municipalidad se encontró que se requería iniciar con la elaboración de un sistema de información para el área de catastro urbanístico.

Si bien es cierto el catastro urbano es el inventario de todos los bienes que posee una localidad y su objetivo es lograr una correcta identificación física, jurídica, fiscal y económica, en este punto se tiene que precisar que se tuvieron inconvenientes para ver el tema fiscal y económico, puesto que no se tuvo predisposición de apoyo por el personal encargado debido a la rotación y situaciones que escaparon de las manos del tesista.

A pesar de las limitaciones de tiempo y temas técnicos del tesista se logra la determinación de los resultados del análisis y diseño del sistema de información para el área de catastro urbanístico de la municipalidad.

Capítulo V

Conclusiones y Recomendaciones

Conclusiones

- a) Se determinó las historias de usuario para el desarrollo del sistema de información para el Área de Catastro Urbanístico de la Municipalidad Distrital de Hualla, como se puede observar en las tablas 11, 12, 13, 14 y 15.
- b) Se determinó los datos de entrada y salida para el sistema de información para el Área de Catastro Urbanístico de la Municipalidad Distrital de Hualla, como se puede observar en las descripciones de las tareas de ingeniería descritas en las tablas 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31 y 32.
- c) Se identificaron y describieron cada una de las tareas de ingeniería para el desarrollo del sistema de información para el Área de Catastro Urbanístico de la Municipalidad Distrital de Hualla, cuyos resultados se muestran en las tablas 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31 y 32.
- d) Se evaluó y describió gráficamente el diagrama de componentes para el desarrollo del sistema de información para el Área de Catastro Urbanístico de la Municipalidad Distrital de Hualla, cuyo resultado se muestra en la figura 5.
- e) Se evaluó y describió gráficamente el diagrama de despliegue para el desarrollo del sistema de información para el Área de Catastro Urbanístico de la Municipalidad Distrital de Hualla, cuyo resultado se muestra en la figura 6.
- f) Se determinó el diseño de la base de datos para el desarrollo del sistema de información para el Área de Catastro Urbanístico de la Municipalidad Distrital de Hualla, Ayacucho, cuyo resultado se muestra en la figura 17.
- g) Se evaluaron y describieron gráficamente los prototipos de interfaces de

usuario para el desarrollo del sistema de información para el Área de Catastro Urbanístico de la Municipalidad Distrital de Hualla, cuyos resultados se muestran las figuras 7, 8, 9, 10, 11, 12, 13, 14, 15, 15 y 16.

Recomendaciones

- a) Se recomienda terminar con la fase de la implementación del sistema de información para el área de Catastro Urbanístico, puesto que el presente trabajo sólo abarcó la fase del análisis y diseño del sistema de información.
- b) Se recomienda precisar las cuestiones fiscales y económicos del catastro urbano, para que los resultados del análisis y diseño gocen de completitud.

Referencias bibliográficas

- (S/f-b). Rae.es. Recuperado el 29 de junio de 2022, de <https://dle.rae.es/>
- Bautista, M. (2019). Sistema de información geográfico de catastro y rentas para la Municipalidad Provincial de Huaraz. Huaraz.
- Beck, K., (1999). Extreme Programming Explained: Embrace Change. Pub. Co; 1 edición.
- Benites, A. (2019). "Sistema de información vía web para mejorar el control de ficha catastral rural en el departamento la Libertad", Trujillo.
- Bernal, C. A. (2010). Metodología de la investigación (Vol. Tercera edición). Colombia: PEARSON EDUCACIÓN.
- Calero, S. (2003). Una explicación de la programación extrema (XP). Madrid: <http://www.apolosoftware.com/>
- Carrasco, S. (2005). Metodología de la Investigación Científica. Lima: San Marcos.
- Extreme Programming - XP. (s/f). Usal.es. Recuperado el 29 de junio de 2022, de <https://viewnext.usal.es/blog/extreme-programming-xp>
- Ferré, X. y Sánchez, M. (sf). Desarrollo Orientado a Objetos con UML. Facultad de Informática. UPM
- Hernandez , R., Fernández, C., y Baptista, P. (2014). Metodología de la investigación (Sexta ed.). Mc Graw Hill Education.
- Hernandez, R., Fernández, Collado, y Baptista, P. (2014). El proceso ágil de desarrollo de software de la investigación. (6º Ed.), Distrito Federal, México: Editorial McGraw-Hill.
- Ibáñez y García, (2009). Informática I (1ª Ed.). México, México DF: Cengagen Learning Editores.
- IEEE 488, (2011). Standard Digital Interface for Programmable Instrumentation
- Pastor, J., (2002). Concepto de sistemas de información en las organizaciones. Barcelona, España: UOC Papers.
- Porras, E. (2010). Comparación de dos procesos de desarrollo de software

usando los métodos ICONIX y XP, caso: comercialización de la Tara de la Región Ayacucho. Tesis de Maestría. Universidad Nacional de Ingeniería, Lima, Perú.

Pressman, R. (2010). Ingeniería de Software: Un enfoque Práctico. España: McGraw Hill.

Sommerville, I. (2005). Ingeniería de software (7ª Ed.). Madrid, España: Pearson Educación.

Stair, R. y Reynolds, G. (1999). Principios de Sistemas de Información: Enfoque administrativo (4ª ed.). Madrid, España: Thomson.

Supo, J. (2015). Taxonomía de la investigación: El arte de clasificar aplicado a la investigación científica.

Weitzenfeld, A. (2005). Ingeniería de Software Orientado a Objetos con UML, Java e Internet. México, D.F., México: Thomson.

Lista de abreviaturas

- CRC:** Clase - Responsabilidad – Colaborador
- POO:** Programación Orientada a Objetos
- OSI:** Organización Internacional de Normalización
- HTTP:** Protocolo de Transferencia de Hipertexto
- TCP:** Protocolo de Control de Transmisión
- API:** Interfaz de programación de aplicaciones
- SGBD:** Sistema de gestión de bases de datos
- XP:** Programación Extrema
- TI:** Tecnología de La Información
- HTML:** Lenguaje de Marcas de Hipertexto
- GUI:** Interfaz Gráfica de Usuario
- UML:** Lenguaje de Modelado Unificado

Glosario

Clase

Una clase es un conjunto de objetos con propiedades y comportamientos en común.

GUI

Son las interfaces de un sistema informático, con las cuales interactuará el usuario del sistema.

CRC

Herramienta utilizada para modelar una clase de un sistema informática. Describe el nombre de la clase, sus responsabilidades y clases colaboradoras.

Contribuyente

"Persona obligada por ley al pago de un impuesto", APA(2022).

Predio

"Herencia, hacienda, tierra o posesión inmueble", APA(2022).

Catastro

"Censo y padrón estadístico de las fincas rústicas y urbanas", APA(2022).

Anexos
Anexo A

Tabla 38

Matriz de Operacionalización

Variable de estudio	Variables descriptivas	Item	Instrumento
Análisis de sistema de información	Historias de usuario	<ul style="list-style-type: none"> • ¿Cómo se realiza el registro de predio? • ¿Cómo se realiza el registro de un contribuyente? • ¿Qué se requiere que haga el sistema informático para el área de catastro urbanístico? • ¿Quiénes utilizarán el sistema de catastro urbanístico? • ¿Qué cargos tienen los personales del área de catastro urbanístico? 	Guía de entrevista
	Datos de entrada Y salida	<ul style="list-style-type: none"> • ¿Qué datos se registran de un contribuyente? • ¿Qué datos se registran de un predio? 	Guía de entrevista

		<ul style="list-style-type: none"> • ¿Hay datos que se tienen que validar previamente al registro de un contribuyente o un predio? • ¿Qué datos se requieren mostrar a los usuarios de catastro? • ¿Qué datos se requieren mostrar a los contribuyentes de catastro? • ¿Qué información se registra del personal de catastro urbanístico? • ¿Cuáles son las tareas de ingeniería que darán el cumplimiento de las historias de usuario? • ¿Qué tipos de tareas de ingeniería serán críticos o necesarios? 	Análisis documental
Diseño de sistema de información	Diagrama de componentes	<ul style="list-style-type: none"> • ¿Cuál es la descripción de las tareas de ingeniería? • ¿Cuáles son los elementos que debe considerar el diagrama de componentes? • ¿Cuál es el diagrama de componentes inicial y final del sistema informático de catastro? • ¿Cómo se relacionan los elementos del diagrama de componentes? 	Análisis documental

Diagrama de despliegue	<ul style="list-style-type: none"> • ¿Cuáles son los elementos que debe considerar el diagrama de despliegue? • ¿Cuál es el diagrama de despliegue final del sistema informático de catastro? • ¿Cómo se relaciona los elementos del diagrama de despliegue? 	Análisis documental
Diseño de base de datos	<ul style="list-style-type: none"> • ¿Cuáles son las entidades más importantes del diseño de la base de datos del sistema de catastro? • ¿Cuál es la relación entre las entidades del diseño de la base de datos del sistema de catastro? 	Análisis documental
Interfaces de usuario	<ul style="list-style-type: none"> • ¿Cuáles son las interfaces gráficas de usuario necesarias para el sistema de catastro? • ¿Qué elementos debiera tener las interfaces gráficas de usuario para el sistema de catastro? • ¿Cuál es el diseño simple que deben tener las interfaces gráficas de usuario para el sistema de catastro? 	Análisis documental

Anexo B

Guía de entrevista

"Sistema informático para el área de Catastro urbanístico de la Municipalidad
Distrital de Hualla"

Fecha:

Nombre del entrevistado:

Objetivo: Determinar los requisitos funcionales para determinar el análisis y diseño del sistema informático para el área de catastro urbanístico.

Preguntas:

1. ¿Cómo se realiza el registro de predio?
2. ¿Cómo se realiza el registro de un contribuyente?
3. ¿Qué se requiere que haga el sistema informático para el área de catastro urbanístico?
4. ¿Quiénes utilizarán el sistema de catastro urbanístico?
5. ¿Qué cargos tienen los personales del área de catastro urbanístico?
6. ¿Qué datos se registran de un contribuyente?
7. ¿Qué datos se registran de un predio?
8. ¿Hay datos que se tienen que validar previamente al registro de un contribuyente o un predio?
9. ¿Qué datos se requieren mostrar a los usuarios de catastro?
10. ¿Qué datos se requieren mostrar a los contribuyentes de catastro?
11. ¿Qué información se registra del personal de catastro urbanístico?

