



**UNIVERSIDAD DE HOLGUÍN**

Facultad de Informática y Matemática

**Sistema para la gestión de los expedientes en el departamento administrativo de la Vivienda de la Dirección Provincial de Holguín.**

**Trabajo de Diploma para optar por el Título de Ingeniero en Informática.**

**Autor: Rosa Elena Proenza Gómez.**

**Tutores: Ing. Dalilis Escobar Rivera.**

**Ing. Javier Fernández Taño.**

**Holguín, 2014**

## DECLARACIÓN DE AUTORÍA

Declaro que soy el único autor de este trabajo y autorizo a la Facultad de Informática y Matemática de la Universidad de Holguín para que hagan el uso que estimen pertinente con el mismo.

Para que así conste firmo la presente a los \_\_\_\_ días del mes de \_\_\_\_\_ del año 2014.

---

Rosa Elena Proenza Gómez

Autor

---

Ing. Dalilis Escobar Rivera

Tutor

---

Ing. Javier Fernández Taño

Tutor

## OPINIÓN DE LOS TUTORES

Título del Trabajo de Diploma: Sistema para la gestión de los expedientes en el departamento administrativo de la Vivienda de la Dirección Provincial de Holguín.

Autor del Trabajo de Diploma: Rosa Elena Proenza Gómez.

---

---

---

---

---

---

---

Por todo lo anteriormente expuesto se considera que la estudiante está lista para ejercer como Ingeniera Informática y se le propone la calificación de \_\_\_\_\_ puntos.

Para que así conste firman la presente a los \_\_\_\_\_ días del mes de \_\_\_\_\_ del año 2014.

\_\_\_\_\_  
Tutor: Ing. Dalilis Escobar Rivera

\_\_\_\_\_  
Tutor: Ing. Javier Fernández Taño

## AGRADECIMIENTOS

*A mis tutores Dalilis y Javier  
Por el apoyo, confianza y ayuda brindada  
en todo momento.*

*A mis Padres  
Por ayudarme siempre y estar ahí cada vez que los necesité...*

*A mi Primo Migue  
Por toda su ayuda y apoyo de tu secretaria.*

*A mi Esposo Ariel  
Por su amor y apoyo incondicional.*

*A mis amigos  
A mis amigos Héctor, Pablo, Annia y Marianela,  
por su amistad y toda su ayuda.*

*Al Departamento de Administración de la vivienda  
Por su amistad, por su ayuda y tiempo dedicado*

*A todos los que de una forma u otra me brindaron su colaboración,*

*Muchas Gracias.*



UNIVERSIDAD DE HOLGUÍN  
OSCAR LUCERO MOYA

---

## DEDICATORIA

*A mi familia*

*A mis padres, que siempre me  
Impulsaron a seguir adelante, a la memoria de mi  
abuela Elda por su cariño y dedicación,  
a mi niña del alma y a mi esposo por todo su amor.*

## RESUMEN

En la Dirección Provincial de la Vivienda de Holguín, específicamente en el departamento de administración se necesita el desarrollo de un sistema que permita llevar un mejor control del proceso de gestión de los expedientes de administración de la vivienda. Actualmente toda la información se realiza manualmente lo que trae como consecuencia que sea poco confiable y exista cúmulo de trabajo diario; incurriéndose en demoras y errores que afectan la eficiencia y confiabilidad de la información que se procesa. Partiendo de estas deficiencias en la presente investigación se planteó como objetivo desarrollar un producto informático que favorezca la gestión de información referente a los expedientes de la vivienda y su control en el Departamento de Administración de la Dirección Provincial de la Vivienda en Holguín, que sea capaz de brindar seguridad e integridad a los datos; evitando la duplicidad de la información y garantizando el rápido acceso a la documentación, para que el proceso pueda tener la calidad requerida. Para el desarrollo de la solución propuesta se emplea la Metodología de Desarrollo de Software Iconix, como Gestor de Bases de Datos MySQL, lenguaje de programación PHP, con el Framework Codeigniter y como servidor *Web Apache*. Este conjunto de herramientas permitió obtener una aplicación *Web* con un ambiente amigable, robusto y confiable.

## **ABSTRACT**

In the Provincial Direction of the Housing of Holguín, specifically in the administration department the development of a system is needed that allows to take a better control of the process of administration of the files of administration of the housing. At the moment the whole information is carried out what results in that is not very reliable manually and exist heap working newspaper; being incurred in delays and errors that affect the efficiency and dependability of the information that it is processed. Leaving of these deficiencies in the present investigation thought about as objective to develop a computer product that favors the administration of information with respect to the files of the housing and their control in the Department of Administration of the Provincial Direction of the Housing in Holguín that is able to offer security and integrity to the data; avoiding the duplicity of the information and guaranteeing the quick access to the documentation, so that the process can have the required quality. For the development of the proposed solution the Methodology of Development of Software Iconix is used, as Agent of Databases MySQL, programming language PHP, with the Framework Codeigniter and I eat servant Apache Web. This group of tools allowed to obtaining an application web with a friendly, robust and reliable atmosphere.

## ÍNDICE

INTRODUCCIÓN.....	1
CAPÍTULO I. FUNDAMENTOS TEÓRICOS.....	7
1.1 Resoluciones para la administración de la vivienda: 4-91, 58-87, 71- 58.....	7
1.2 Gestión de los Expedientes de Administración de la Vivienda.....	8
1.3 Caracterización de la entidad objeto de estudio.....	10
1.4 Sistemas informáticos relacionados con la gestión de los expedientes de la vivienda .	11
1.5 Tendencias y Tecnologías actuales .....	12
1.5.1 Arquitectura Cliente/Servidor.....	12
1.5.2 Aplicaciones web.....	13
1.5.3 Servidores de Aplicaciones Web .....	14
1.5.4 Lenguajes de Programación.....	16
1.5.5 Framework.....	17
1.5.6 Sistemas Gestores de Bases de Datos .....	20
1.6 Metodologías de desarrollo de Software .....	22
1.7 Conclusiones del capítulo .....	25
CAPÍTULO II. DESCRIPCIÓN DE LA SOLUCIÓN PROPUESTA .....	26
2.1 Modelo del Dominio.....	26
2.1.2 Requerimientos Funcionales .....	26
2.1.3 Definición de los principales objetos del modelo del dominio .....	33
2.2 Modelación de los casos de uso .....	36
2.3 Análisis y Diseño Preliminar .....	41
2.3.1 Descripción Textual de los Principales Casos de Uso del Sistema .....	41
2.3.2 Diagramas de Robustez .....	44



---

2.4 Prototipación rápida .....	46
2.5 Modelo de despliegue .....	48
2.6 Diseño detallado.....	49
2.6.1 Diagrama de secuencia .....	49
2.6.2 Diseño de la base de datos .....	51
2.7 Implementación .....	51
2.7.1 Estándar de codificación.....	52
2.8 Estudio de factibilidad .....	52
2.9 Valoración de sostenibilidad.....	56
2.10 Valoración de la Propuesta de Solución.....	59
2.11 Conclusiones del capítulo .....	60
CONCLUSIONES.....	62
RECOMENDACIONES.....	63
GLOSARIO DE TÉRMINOS .....	64
REFERENCIAS BIBLIOGRÁFICAS.....	67
ANEXOS.....	I

## ÍNDICE DE FIGURAS

Figura 1. Flujograma de Información de la Dirección Provincial Vivienda .....	9
Figura 2. Organigrama de la Dirección Provincial Vivienda .....	11
Figura 3. Arquitectura Cliente - Servidor .....	13
Figura 4. Esquema de trabajo RUP .....	23
Figura 5. Esquema de XP .....	23
Figura 6. Esquema de trabajo de ICONIX .....	24
Figura 7. Modelo del Dominio .....	36
Figura 8. Actores del sistema .....	37
Figura 9. Diagrama de paquetes .....	38
Figura 10. Diagrama de casos de uso del paquete seguridad .....	39
Figura 11. Diagrama de casos de uso del paquete Administración del sistema .....	40
Figura 12. Diagrama de casos de uso del paquete gestión de expedientes .....	41
Figura 13. Diagrama de robustez del caso de uso crear expediente de cuartería .....	45
Figura 14. Diagrama de robustez del caso de uso modificar organismo .....	45
Figura 15. Diagrama de robustez del caso de uso ver trazas del sistema .....	46
Figura 16. Interfaz autenticación .....	47
Figura 17. Interfaz principal del sistema .....	47
Figura 18. Interfaz expediente de ilegalidad .....	48
Figura 19. Modelo de Despliegue .....	49
Figura 20. Diagrama secuencia del caso de uso crear expediente de cuartería .....	50
Figura 21. Diagrama secuencia del caso de uso modificar organismo .....	50
Figura 22. Diagrama de secuencia del caso de uso ver trazas del sistema .....	51

## ÍNDICE DE TABLAS

Tabla 1. Glosario de Términos.....	35
Tabla 2. Actores del sistema.....	38
Tabla 3. Descripción textual del caso de uso crear expediente de cuartería .....	42
Tabla 4. Descripción textual del caso de uso modificar organismo .....	43
Tabla 5. Descripción textual del caso de uso ver trazas del sistema .....	44
Tabla 6. Factores de complejidad técnica.....	54
Tabla 7. Cálculo del TCF .....	55
Tabla 8. Factor ambiental .....	55
Tabla 9. Cálculo del ECF .....	55
Tabla 10. Estimación del esfuerzo y costo.....	56

## INTRODUCCIÓN

En su alegato “La Historia me Absolverá”, el líder de la Revolución cubana Fidel Castro, condenó el régimen de explotación que imperaba en Cuba; al presentar el sombrío espectáculo de las viviendas precarias que ocupaba gran parte del pueblo y la extorsión que significaba el negocio de los alquileres de espacios habitacionales. Proclamó que lo ideal en la ciudad es que cada familia viva en su propia casa o apartamento [1].

La Ley No.48, del 27 de diciembre de 1984, Ley General de la Vivienda, autorizó la transferencia de la propiedad a los usufructuarios onerosos y ocupantes legítimos de sus viviendas y la entrega en propiedad de las viviendas que en lo sucesivo se asignen por el Estado para esos fines. Legalizó situaciones de hecho en la ocupación de la vivienda que hasta entonces no habían tenido solución jurídica; habiéndose desarrollado con ese propósito en todo el país un amplio proceso que convirtió en propietarios a más de setecientas cincuenta mil familias, quedando constituido el Instituto Nacional de la Vivienda [1].

El 6 de febrero de 1985 se constituye en la Provincia Holguín la Dirección Provincial de la Vivienda (DPV), la cual estaba compuesta por los departamentos Jurídicos, Inversiones y la actividad de Planificación y Conservación, además de otras áreas de apoyo.

En 1998 en un proceso de fusión y creación se constituye mediante resolución no. 32 de fecha 16 de febrero de 1998 del Ministro de Economía y Planificación la Unidad Presupuestada denominada Dirección Provincial de la Vivienda de Holguín situada en calle Fomento # 237 e/ Aguilera y Arias, municipio y provincia de Holguín [2].

Entre los departamentos que la componen se encuentra el de Administración de la Vivienda, donde se lleva a cabo el proceso de gestión de los distintos expedientes de: cuarterías, comunidades, edificios multifamiliares, viviendas vinculadas y medios básicos, las ilegalidades en los edificios multifamiliares y en viviendas vinculadas o medios básicos y los ceses de las viviendas vinculadas por arribar al término establecido de 20 años.

La gestión de los expedientes de Administración de la Vivienda se realiza de forma manual, lo que trae como consecuencia que la información sea poco confiable, exista cúmulo de trabajo diario y no fluyan las informaciones con rapidez.

A partir de todo lo antes mencionado se evidencian limitaciones en el proceso de gestión de los expedientes del departamento de Administración de la Vivienda:

- ✓ La conformación de los expedientes de administración de la vivienda se realiza de forma manual lo que trae consigo que su gestión y control se convierta en una tarea compleja, no siendo este el método más óptimo; dado el gran volumen de trámites que se realizan, incurriendo en gastos excesivos de materiales de oficina.
- ✓ Las viviendas vinculadas cuando llegan al tiempo establecido de 20 años se le debe comenzar a realizar el expediente de cese, lo que provoca que todos los días se realicen búsquedas complejas en los archivos para obtener los expedientes de las viviendas que serán cesadas.
- ✓ La probabilidad de incurrir en errores en los datos que se envían para el Instituto de la Vivienda como la totalidad de edificios, cuarterías, comunidades, viviendas vinculadas y medios básicos con que cuenta la provincia trimestralmente es alta, debido al elevado flujo de información que se procesa y a la no adecuada organización de la misma.
- ✓ La pérdida y deterioro de los expedientes de administración de la vivienda, por no estar organizados, trae como consecuencia que el servicio prestado al cliente no se haga con la eficiencia necesaria.
- ✓ El cierre estadístico que implica el cálculo del total de edificios, cuarterías, comunidades y otras operaciones matemáticas se realiza manualmente y se pasa esa información a un modelo en Excel, mecanismo que duplica el trabajo.
- ✓ Para brindarle información a los clientes interesados en sus expedientes hay que realizar búsquedas en cientos de estos en un archivo bastante extenso siendo el servicio complicado por motivos de demora en tiempo empleado.
- ✓ La dirección de la entidad, específicamente el director, tiene que esperar que el departamento le conforme cada mes un balance de la totalidad de edificios, cuarterías, comunidades, etc. Por ende no se puede conocer el estado de esta información de inmediato, lo que imposibilita el apoyo y mejora en la toma de decisiones por parte de la dirección de la empresa.

Estas limitaciones conducen a la formulación del **problema científico** siguiente: ¿Cómo favorecer la gestión de información referente a los expedientes de la vivienda y su control en el Departamento de Administración de la Dirección Provincial de la Vivienda en Holguín?

Este problema se enmarca en el **objeto de estudio**: proceso de gestión de información referente a los expedientes de la vivienda y su control.

Para dar solución al problema se propone como **objetivo**: desarrollar un producto informático que favorezca la gestión de información referente a los expedientes de la vivienda y su control en el Departamento de Administración de la Dirección Provincial de la Vivienda en Holguín.

El objetivo delimita el **campo de acción**: informatización de la gestión de información referente a los expedientes de la vivienda y su control en el Departamento de Administración de la Dirección Provincial de la Vivienda en Holguín.

Para guiar la investigación se plantearon las **preguntas científicas** siguientes:

- ✓ ¿Qué fundamentos teóricos sustentan el desarrollo de un producto informático que favorezca el proceso de gestión de información referente a los expedientes de la vivienda y su control?
- ✓ ¿Cómo desarrollar un producto informático que beneficie la gestión de información referente a los expedientes de la vivienda y su control en el Departamento de Administración de la Dirección Provincial de la Vivienda en Holguín?
- ✓ ¿Resultará sostenible en el tiempo el producto informático propuesto?
- ✓ ¿Cómo valorar el producto informático propuesto?

La investigación transcurre a través del cumplimiento de las **tareas científicas** siguientes:

1. Estudiar el estado actual del proceso de gestión de información referente a los expedientes de la vivienda y su control en el Departamento de Administración de la Dirección Provincial de la Vivienda en Holguín.
2. Determinar los fundamentos teóricos que sustentan la elaboración de un producto informático para la gestión de información referente a los expedientes de la vivienda y su control.

3. Realizar el análisis y diseño del producto informático para la gestión de información referente a los expedientes de la vivienda y su control.
4. Valorar la sostenibilidad del producto informático propuesto.
5. Implementar el producto informático.
6. Valorar la solución propuesta a través del criterio de expertos.

#### **Métodos del nivel teóricos utilizados**

**Análisis y síntesis:** se utilizó en el análisis y estudio de la información referente a los expedientes del departamento de administración de la vivienda, permitió la elaboración de los fundamentos teóricos, ayudó a una mejor descomposición de cada requerimiento para lograr un mejor entendimiento.

**Histórico - Lógico:** este método permitió conocer con mayor profundidad el proceso que se realiza con los expedientes del departamento de administración de la vivienda, además ayudó a la comprensión de las regulaciones legales que rigen la entidad y se desarrollaron investigaciones sobre las herramientas informáticas para conocer los antecedentes y las tendencias actuales.

**Método de modelación:** este método se utilizó durante el proceso de elaboración del sistema, se empleó la metodología de ingeniería de software Iconix, la cual está basada en un conjunto de modelos que describen las perspectivas del proceso de desarrollo en sentido general y así lograr que el sistema se pueda modelar de una forma menos abstracta y con mayor comprensión.

#### **Métodos del nivel empíricos utilizados**

**Entrevista:** permitió la obtención de la información necesaria referente a la gestión de los expedientes que se generan en el departamento de Administración de la Vivienda y de otras informaciones de interés. Posibilitó la definición de las necesidades y funcionalidades del sistema teniendo en cuenta los criterios emitidos para lograr adquirir un producto informático que cumpla las expectativas.

**Revisión de documentos:** este método se utilizó fundamentalmente para recopilar y comprender la información documentada y el funcionamiento de los requerimientos del sistema.

**Observación:** se empleó en el análisis, diseño e implementación del sistema. Éste facilitó que se tuviera una mejor seguridad y exactitud en las decisiones tomadas y los resultados obtenidos acerca del producto, además lograr la información directa sobre el proceso de desarrollo y objeto que está siendo investigado.

**Encuesta:** se realizaron encuestas a expertos para evaluar los resultados del producto informático propuesto aplicándose el criterio de experto.

### **Métodos Estadísticos**

Se empleó para el tratamiento de las encuestas.

Este trabajo está compuesto por una introducción, dos capítulos, conclusiones, recomendaciones, bibliografía y anexos.

### **Capítulo I** Fundamentos Teóricos:

En este capítulo se exponen los fundamentos teóricos en que se apoya la investigación y la caracterización de la entidad. Se describen las principales tendencias y tecnologías actuales imprescindibles para dar cumplimiento al objetivo trazado, también se hace un estudio para seleccionar la metodología de desarrollo de software.

### **Capítulo II** Descripción de la solución propuesta:

Este capítulo contiene el diseño y construcción de la propuesta, se detalla el uso de la metodología seleccionada para el desarrollo del producto informático en cada una de sus fases, mostrando sus modelos y diagramas. Se refleja, además la valoración de sostenibilidad del producto informático propuesto; así como los resultados obtenidos de su evaluación.

## **CAPÍTULO I. FUNDAMENTOS TEÓRICOS**

En el presente capítulo se realiza un análisis de los fundamentos que guían la investigación, con el objetivo de establecer y comprender mejor los elementos teóricos que se relacionan con la gestión de los expedientes de Administración de la Vivienda. Se realiza un estudio de las tendencias y tecnologías actuales especialmente las escogidas para el desarrollo de la aplicación, así como una breve valoración de las metodologías de desarrollo de software con énfasis en la seleccionada.

### **1.1 Resoluciones para la administración de la vivienda: 4-91, 58-87, 71- 58**

Con el fin de regular el proceso que se realiza en el sistema de la vivienda y en específico en el departamento de administración, se crean las resoluciones: 4-91 Resolución general de los edificios multifamiliares, 58-87 Resolución de viviendas vinculadas y medios básicos y 71- 58 Resolución de cese, dichas resoluciones son editadas y revisadas periódicamente asegurando la calidad del proceso.

La Resolución general de edificios multifamiliares (4-91), tiene como objeto, establecer el régimen jurídico especial a que están sujetos los edificios multifamiliares, así como los deberes y derechos de los propietarios y arrendatarios de las viviendas que conforman dichos edificios. Sus disposiciones se proponen controlar, proteger y conservar el uso y destino de las áreas, elementos o equipos comunes de la edificación, además de establecer la calificación que presentan y las ilegalidades que pueden cometerse en los edificios las cuales deben ser debidamente controladas para su posterior erradicación [3].

La Resolución de viviendas vinculadas y medios básicos (58-87) tiene por objetivo definir el concepto de vivienda vinculada y vivienda medio básico, así como regular los requisitos que estas deben tener y su calificación de acuerdo a sus características. También deben de controlar las ilegalidades que pueden cometerse igualmente que en los edificios multifamiliares [4].

La resolución de cese (71- 58) regula el procedimiento para la conformación del expediente de cese de las viviendas vinculadas que arriban al término legal establecido, o sea 20 años vividos, el cual le asiste a los arrendatarios el derecho a la transferencia de la propiedad de la misma [5].

## **1.2 Gestión de los Expedientes de Administración de la Vivienda**

La gestión de los expedientes de Administración de la Vivienda: edificios multifamiliares, viviendas vinculadas, medios básicos, cuarterías, comunidades, ilegalidades en edificios, ilegalidades en viviendas vinculadas y medios básicos y los ceses a las viviendas vinculadas; no es más que el proceso que lleva el departamento Administración de la Vivienda el cual está clasificado como un proceso estratégico en la entidad y tiene una estructura compuesta por un especialista principal y cuatro especialistas B en viviendas, edificios multifamiliares y comunidades.

Este proceso comienza con la información primaria aportada por los técnicos de las Direcciones Municipales de la Vivienda (DMV) los cuales recogen la descripción del inmueble, el lugar donde está asentado y las características que tiene, luego es enviada esta información a la Dirección Provincial, específicamente al departamento de Administración de la Vivienda. Esta es revisada y confirmada por los especialistas para luego comenzar con la confección de los expedientes donde se recoge un registro primario con la descripción del inmueble y un documento con la clasificación que estos presentan por sus características en el cual estará reflejado la tipología, estado técnico, la categoría, el municipio y reparto al que pertenecen.

A los expedientes de edificios multifamiliares, viviendas vinculadas y medios básicos se les realiza un expediente de ilegalidad en caso de que en el inmueble se detecten violaciones en cuanto a lo establecido en las resoluciones de la vivienda donde se recogen cada una de las ilegalidades que estos presentan.

Las viviendas vinculadas cuando arriban al término legal establecido, o sea que los arrendatarios tengan 20 años vividos, se debe comenzar a realizar el expediente de cese. El departamento de Administración de la Vivienda, un mes antes de terminado el año habilita un registro con relación nominal de las viviendas que serán cesadas, los arrendatarios que arriban al término legal establecido al año siguiente y la fecha en que cumple los 20 años. Mediante este registro se realiza la búsqueda de las viviendas que serán cesadas, para luego proceder a la conformación del expediente de cese el cual recoge entre otros datos los motivos del cese, la fecha de pago de la vivienda y el número de resolución de cese; una vez

terminado el expediente los arrendatario de la viviendas cesadas pueden comenzar con los trámites para la transferencia de propiedad [6].

Todo lo descrito anteriormente es regulado por las resoluciones de la vivienda específicamente: 4-91 Resolución general de los edificios multifamiliares, 58-87 Resolución de viviendas vinculadas y medios básicos y 71- 58 Resolución de cese.

Para la realización de este proceso en la Dirección Provincial de la Vivienda de Holguín es necesario contar con una estrategia de comunicación de la información que se define en la entidad como sigue:

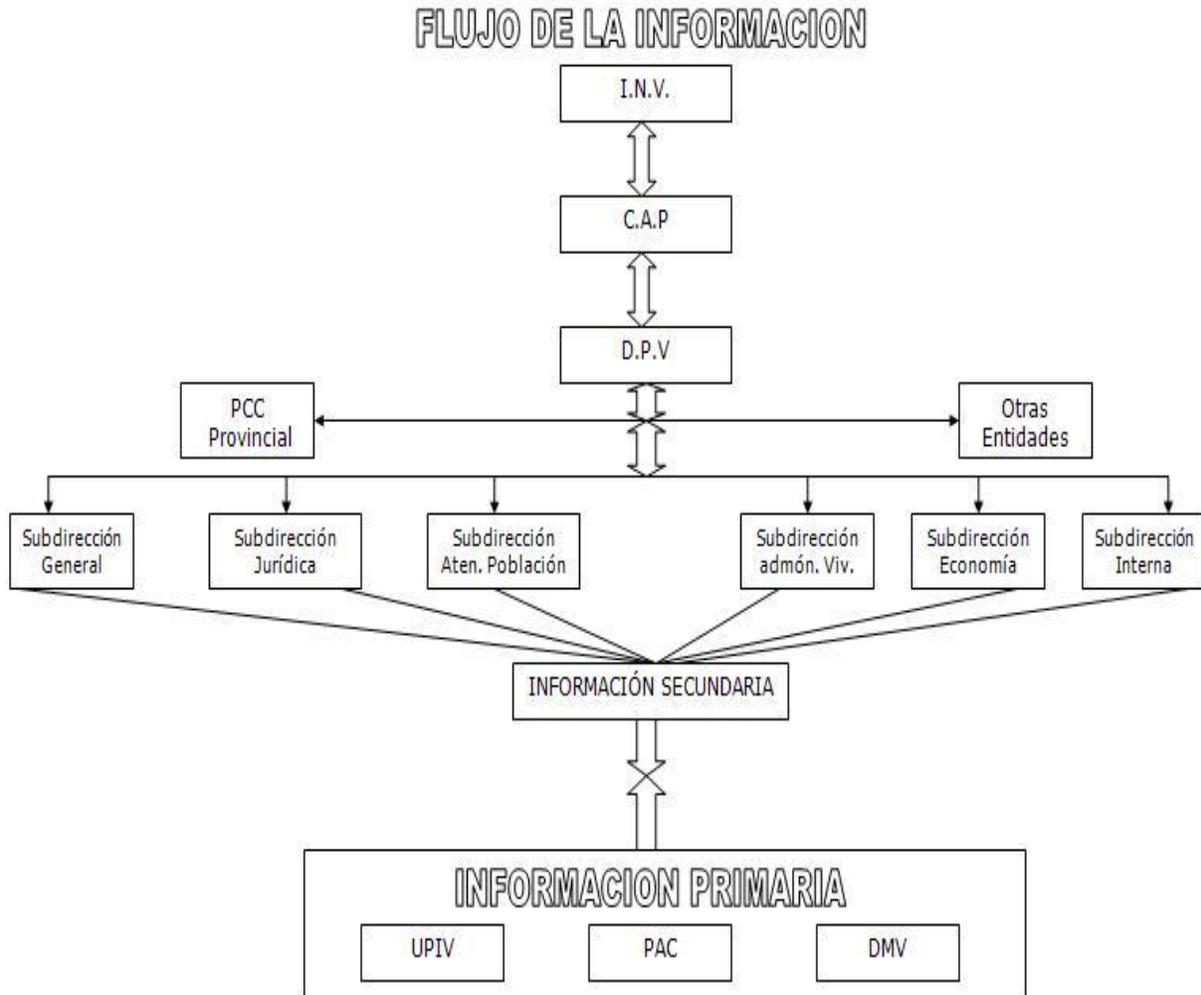


Figura 1. Flujograma de Información de la Dirección Provincial Vivienda

### **1.3 Caracterización de la entidad objeto de estudio**

La Dirección Provincial de la Vivienda es la entidad rectora en la formulación y aplicación de la actividad legal e inversionista y sus urbanizaciones en la Provincia Holguín, trabaja con una alta presencia en el escenario político social. Tiene como misión ejercer la rectoría en la aplicación de la política de la vivienda en los trámites jurídicos, en el proceso técnico inversionista para los programas constructivos de conservación y rehabilitación y sus urbanizaciones, en el balance de recursos que demanda su ejecución y en el control del fondo habitacional de la provincia; con el objetivo de elevar la satisfacción de las necesidades habitacionales de la sociedad, sobre bases económicas y ecológicas sustentables, con elevado compromiso social en el marco de la Ley General de la Vivienda y sus normas complementarias [7].

Su objeto social, es promover, controlar y garantizar el cumplimiento de la Ley General de la Vivienda y demás disposiciones legales sobre esta materia, dirige y controla la estrategia para la recuperación del fondo habitacional, evitando la pérdida de la capacidad habitacional existente, así como la erradicación de condiciones habitacionales precarias. Dirige y controla el proceso inversionista para la construcción y rehabilitación de viviendas en correspondencia con el desarrollo socioeconómico del País y los territorios; logrando la integridad de las inversiones con un mayor control y efectividad de los recursos materiales y financieros [7].

Para darle cumplimiento a su objeto social la Dirección Provincial de la Vivienda de Holguín cuenta con una estructura organizativa para su centro de Dirección compuesta por: una Dirección, una Sub Dirección General, cinco Sub Direcciones y ocho Departamentos resultando de la siguiente manera:

**DIRECCION PROVINCIAL DE LA VIVIENDA**

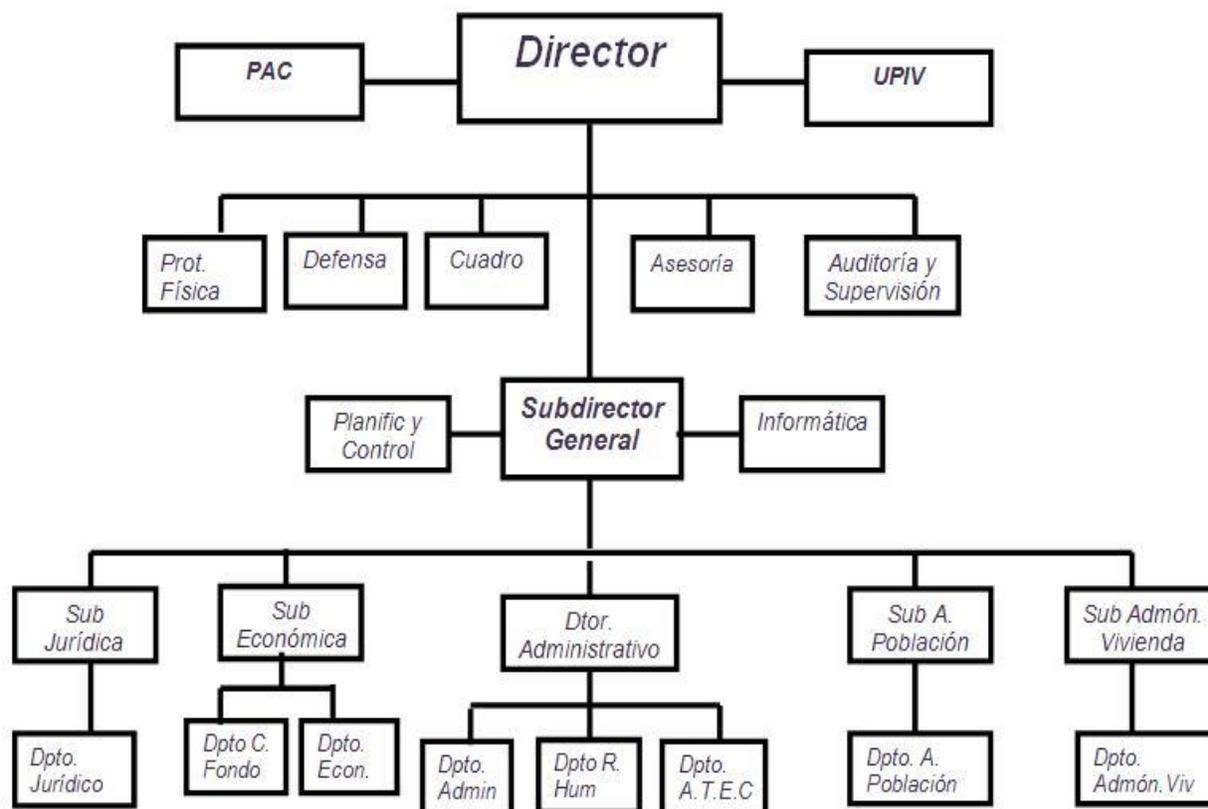


Figura 2. Organigrama de la Dirección Provincial Vivienda

#### 1.4 Sistemas informáticos relacionados con la gestión de los expedientes de la vivienda

Las búsquedas de aplicaciones relacionadas con la gestión de expedientes de la vivienda se dividió en dos fases, en la primera se exploró en la intranet nacional; para conocer sobre aplicaciones existentes que pudieran favorecer el trabajo de los especialistas y técnicos en la labor del proceso de gestión de los expedientes en el departamento administrativo de la Vivienda de la Dirección Provincial de Holguín.

La investigación arrojó los siguientes resultados: hasta el momento no se ha encontrado una aplicación informática que gestione dicho proceso. En conclusión, actualmente no se cuenta con un sistema eficiente que gestione y agilice el trabajo con los expedientes en el departamento de Administración de la Vivienda de la Dirección Provincial.

En la segunda fase las búsquedas se centraron en Internet utilizando como criterio de búsqueda, aplicaciones relacionadas que facilitan y agilizan el trabajo de la vivienda. La investigación dio el siguiente resultado: hasta el momento no se ha encontrado ninguna que se relacione a la actividad que realiza el departamento de Administración de la Vivienda en la Dirección Provincial de Holguín.

Por último se visitó la entidad Desoft la cual hasta el momento no han implementado un sistema relacionado con la gestión de expedientes de la vivienda para la Dirección Provincial. Por todo lo antes expuesto se evidencia la necesidad de un sistema informático que apoye el proceso de los expedientes de la vivienda en el departamento de administración por lo que se propone continuar con la investigación para la creación del Sistema para la gestión de los expedientes en el departamento administrativo de la Vivienda de la Dirección Provincial de Holguín.

### **1.5 Tendencias y Tecnologías actuales**

Para el desarrollo del sistema propuesto, se realizó un análisis de las tendencias y tecnologías actuales para realizar un producto que se ajuste a las necesidades de los usuarios y con la calidad requerida. A continuación se exponen los resultados de dicho estudio, con un detallado análisis de aquellas herramientas utilizadas para la implementación de la propuesta.

#### **1.5.1 Arquitectura Cliente/Servidor**

En el mundo de TCP/IP las comunicaciones entre computadoras se rigen básicamente por lo que se llama modelo Cliente-Servidor, éste es un modelo que intenta proveer usabilidad, flexibilidad y escalabilidad en las comunicaciones. Desde el punto de vista funcional, se puede definir la computación Cliente/Servidor como una arquitectura distribuida que permite a los usuarios finales obtener acceso a la información en forma transparente aún en entornos multiplataforma [8].

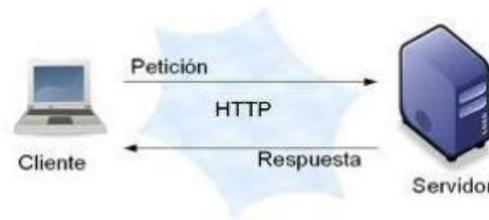


Figura 3. Arquitectura Cliente - Servidor

Entre las principales características de la arquitectura cliente servidor se encuentran [9]:

- ✓ El servidor presenta a todos sus clientes una interfaz única y bien definida.
- ✓ Es multiplataforma. La plataforma del sistema operativo del cliente y del servidor no son siempre la misma.
- ✓ El acceso a los datos es centralizado, lo que evita tener información duplicada o no sincronizada en todos los clientes, además se puede obtener de forma inmediata.
- ✓ Permite el uso tanto de *hardware* como de *software* de diversos fabricantes, lo cual contribuye considerablemente a la reducción de costos y favorece la flexibilidad en la implantación y actualización de soluciones.
- ✓ Las reglas de la organización y las reglas de seguridad se pueden definir una sola vez en el servidor para todos los usuarios.
- ✓ Escalabilidad: se puede aumentar la capacidad de clientes y servidores por separado.

Dentro del desarrollo de aplicaciones informáticas una de las más utilizadas es la arquitectura Cliente/Servidor, no solo por las características anteriormente mencionadas, también porque permite la utilización de los servidores con los que cuenta la entidad. La arquitectura reduce los costos de *hardware*, además proporciona la posibilidad de conectar clientes y servidores independientemente de sus plataformas todo esto conllevó a la aplicación de este modelo en el producto informático que se propone.

### 1.5.2 Aplicaciones Web

Las aplicaciones *Web* son aquellas que los usuarios pueden utilizar accediendo a un servidor *Web* a través de Internet o de una intranet mediante un navegador. En otras palabras, es una aplicación *software* que se codifica en un lenguaje soportado por los navegadores *Web* y en la que se confía la ejecución de la aplicación al navegador [10].

Entre las principales ventajas de las aplicaciones *Web* se encuentran [11]:

- ✓ Ahorra tiempo: Se pueden realizar tareas sencillas sin necesidad de descargar ni instalar ningún programa.
- ✓ No hay problemas de compatibilidad: Basta tener un navegador actualizado para poder utilizarlas.
- ✓ No ocupan espacio en el disco duro.
- ✓ Actualizaciones inmediatas: Como el *software* lo gestiona el propio desarrollador, cuando se conecta está usando siempre la última versión que haya lanzado.
- ✓ Consumo de recursos bajo: Dado que toda (o gran parte) de la aplicación no se encuentra en el ordenador del cliente, muchas de las tareas que realiza el *software* no consumen recursos porque se realizan desde el otro ordenador, el servidor.
- ✓ Multiplataforma: Se pueden usar desde cualquier sistema operativo porque sólo es necesario tener un navegador.
- ✓ La disponibilidad suele ser alta porque el servicio se ofrece desde múltiples localizaciones para asegurar la continuidad del mismo.

Las aplicaciones *Web* generan dinámicamente una serie de páginas en un formato estándar, como HTML o XHTML, soportados por los navegadores *Web* comunes. Se utilizan lenguajes interpretados en el lado del cliente tales como JavaScript, Java, para añadir elementos dinámicos a la interfaz de usuario. Cada página *Web* en particular se envía al cliente como un documento estático, pero la secuencia de páginas ofrece al usuario una experiencia interactiva. Durante la sesión, el navegador *Web* interpreta y muestra en pantalla las páginas, actuando como cliente para cualquier aplicación *Web* [12].

Una aplicación *Web* esta normalmente estructurada como una aplicación de tres-capas [13]:

- ✓ El navegador *Web* ofrece la primera capa
- ✓ El motor capaz de usar alguna tecnología web dinámica por ejemplo PHP, Java, ASP, CGI, Python constituye la capa intermedia.
- ✓ La base de datos constituye la tercera y última capa.

### 1.5.3 Servidores de Aplicaciones Web

Los servidores *Web* son aquellos *software* que normalmente están alojados en un servidor. Estos se ejecutan continuamente en un computador, manteniéndose a la espera de

peticiones de ejecución que le realizará un cliente o un usuario de Internet. El servidor *Web* se encarga de contestar a estas peticiones de forma adecuada, entregando como resultado una página *Web* o información de todo tipo de acuerdo a los comandos solicitados [13].

Una de las ventajas de usar aplicaciones *Web* es que los servidores *Web* se encuentran y ejecutan en el servidor y no en la máquina del cliente, por lo tanto este no necesita ninguna capacidad adicional. Entre los servidores más utilizados en la actualidad se encuentran: *Apache*, *Sun Application Server*, *Internet Information Server (IIS)* [14].

### **Servidor Apache**

Apache es un servidor de páginas *Web* HTTP de código abierto, es uno de los servidores más utilizados en el mundo, desarrollado y mantenido por una comunidad abierta de desarrolladores. Es el encargado de aceptar las peticiones de páginas o recursos en general que provienen de los visitantes que acceden al sitio *Web* y gestionar su entrega o denegación, de acuerdo a las políticas de seguridad establecidas [15].

Entre sus principales ventajas se encuentran [16]:

- ✓ Es personalizable, la arquitectura modular de Apache permite construir un servidor hecho a la medida.
- ✓ Tiene un formato simple, los archivos de configuración de Apache están en ASCII, por lo que pueden ser editados tan solo con un editor de texto.
- ✓ Corre sobre múltiples plataformas y sistemas operativos.
- ✓ Es gratis (gracias a sus características open-source).
- ✓ Provee una muy buena base para la seguridad del sistema, gracias a los módulos de Autenticación, Autorización y Control de Acceso al Servidor *Web*.
- ✓ Trabaja en conjunto con gran cantidad de lenguajes de programación interpretados como PHP (Lenguaje Personal Home Page), Perl, Java, JSP (Java Server Pages) y otros lenguajes de script, que son el complemento ideal para los sitios *Web* dinámicos.

Se escogió como servidor *Web* el Apache por las ventajas que brinda, además por ser este una solución sencilla y rápida para tener sitios *Web* funcionando al 100% de forma gratuita.

#### 1.5.4 Lenguajes de Programación

Un lenguaje de programación puede ser utilizado para controlar el comportamiento de una máquina, particularmente una computadora. Consiste en un conjunto de símbolos y reglas sintácticas y semánticas que definen su estructura y el significado de sus elementos y expresiones. Una de la característica relevante de los lenguajes de programación es que puedan tener un conjunto común de instrucciones que puedan ser comprendidas para realizar la construcción del programa [17].

Existen numerosos lenguajes de programación empleados para el desarrollo de aplicaciones *Web*, entre los cuales se encuentran: *PHP, Perl, Ruby, Python, HTML, XML*.

##### Lenguaje Personal Home Page (PHP)

PHP es un lenguaje interpretado de alto nivel embebido en páginas HTML y ejecutado en el servidor. Escribir un interfaz vía *Web* para una base de datos es una tarea simple en PHP, una de sus principales características es la de soportar una gran cantidad de bases de datos, además de no necesitar ser compilado para ejecutarse y tener la libertad de elegir el sistema operativo y el servidor *Web* [18].

Entre sus principales ventajas se encuentran [18]:

- ✓ Es un lenguaje multiplataforma.
- ✓ Fácil de aprender.
- ✓ Capacidad de conexión con la mayoría de los manejadores de base de datos: *MySQL, PostgreSQL, Oracle, MS SQL Server*.
- ✓ Posee una amplia documentación en su sitio *Web* oficial la cual incluye descripción y ejemplos de cada una de sus funciones.
- ✓ Es libre, no se está forzado a pagar actualizaciones anuales para tener una versión que funcione.
- ✓ Dispone de una amplia gama de librerías.

Las desventajas de PHP son:

- ✓ Se necesita instalar un servidor *Web*.
- ✓ La legibilidad del código puede verse afectada al mezclar sentencias HTML y PHP.

## Seguridad

PHP es un poderoso lenguaje e intérprete, ya sea incluido como parte de un servidor *Web* en forma de módulo o ejecutado como un binario de interfaz de entrada común (Common Gateway Interface CGI) separado es capaz de acceder a archivos, ejecutar comandos y abrir conexiones de red en el servidor. PHP está diseñado específicamente para ser un lenguaje seguro para escribir programas CGI y con la selección correcta de opciones de configuración y ejecución, siguiendo algunas prácticas correctas de programación [19].

En los últimos años la utilización de PHP se ha generalizado; dadas sus ventajas para el desarrollo de aplicaciones *Web*, por lo que se consideró una de las tecnologías con las que desarrollar esta propuesta, además de ser un lenguaje de programación excelente para el desarrollo de páginas *Web*.

### 1.5.5 Framework

En el desarrollo de *software*, un *framework* o infraestructura digital, tiene una estructura conceptual y tecnológica de soporte definido, normalmente con artefactos o módulos de *software* concretos, que puede servir de base para la organización y desarrollo de *software*. Típicamente, puede incluir soporte de programas, bibliotecas y un lenguaje interpretado, entre otras herramientas, para así ayudar a desarrollar y unir los diferentes componentes de un proyecto, además promueve buenas prácticas de desarrollo como el uso de patrones [20].

Las principales ventajas de la utilización de un *framework* son [20]:

- ✓ El desarrollo rápido de aplicaciones. Los componentes incluidos en un *framework* constituyen una capa que libera al programador de la escritura de código de bajo nivel.
- ✓ La reutilización de componentes *software* al por mayor. Los *frameworks* son los paradigmas de la reutilización.
- ✓ El uso y la programación de componentes que siguen una política de diseño uniforme. Un *framework* orientado a objetos logra que los componentes sean clases que pertenezcan a una gran jerarquía de clases, lo que resulta en bibliotecas más fáciles de aprender a usar.

## Frameworks para PHP

PHP cuenta con numerosos *frameworks*, cada uno ofrece una amplia gama de funciones que agregan gran versatilidad a las aplicaciones. El uso de *frameworks* de desarrollo para la construcción de aplicaciones en lenguaje PHP es común, ya que brinda una base para la programación de las aplicaciones. Algunos de los más populares son: *CakePHP*, *Zend*, *Symfony*, *Codeigniter*, entre otros.

**CakePHP:** Es un *framework* para el desarrollo aplicaciones *Web* escrito en PHP de código abierto, en ocasiones combinado con algún componente de Zend. Implementa el patrón Modelo-Vista-Controlador (MVC). Posee una infraestructura que tiene como finalidad permitir el desarrollo de aplicaciones *Web* de manera ágil y estructurada [21].

**Zend:** Es un *framework open source* desarrollado por Zend. Es muy conocido y uno de los más utilizados profesionalmente, implementa el patrón Modelo-Vista-Controlador (MVC), es orientado a objetos. Se recomienda su uso en equipos de desarrollo donde trabajen varias personas en proyectos grandes. Su curva de aprendizaje es sumamente larga, además es un *framework* grande que usa gran cantidad de memoria [22].

**Symfony:** Es un *framework* diseñado para optimizar el desarrollo de aplicaciones *Web* a través de diversas características clave. Separa las reglas de negocio de la aplicación, la lógica del servidor y las vistas de presentación. Contiene una gran variedad de herramientas y clases para conseguir acortar el tiempo de desarrollo de aplicaciones *Web* complejas, está desarrollado completamente con PHP 5, se puede utilizar en múltiples plataformas y su configuración se realiza normalmente a través de líneas de comando. Su curva de aprendizaje es alta y se recomienda para cualquier tipo de proyecto, en especial para el desarrollo de aplicaciones grandes [23].

## Framework Codeigniter

**Codeigniter:** Es un *framework* para desarrollo de aplicaciones *Web* usando PHP. Su objetivo es permitir desarrollar proyectos mucho más rápido que lo que se harían si se escribiera el código desde cero, suministrándole un rico conjunto de bibliotecas para tareas comunes, una interfaz sencilla y una estructura lógica para acceder a esas bibliotecas. *CodeIgniter* permite enfocarse creativamente en el proyecto al minimizar la cantidad de código necesaria para

una tarea dada. Es compatible con varios gestores de bases de datos como son: *MySQL (4.1+)*, *Postgres*, *Oracle*, *SQLite* [24].

Está basado en el patrón de desarrollo Modelo-Vista-Controlador (MVC), el cual tiene un enfoque de *software* que separa la lógica de la aplicación de la presentación. En la práctica, le permite a sus páginas *Web* contener mínimo código ya que la presentación está separada del código PHP [24].

- ✓ El Modelo representa sus estructuras de datos. Típicamente sus clases del modelo contendrán funciones que los ayudarán a devolver, insertar y actualizar información de su base de datos.
- ✓ La Vista es la información que se presenta al usuario. Una vista será normalmente una página *Web*, pero en *CodeIgniter*, una vista también puede ser un fragmento de página como el encabezado o pie de página.
- ✓ El Controlador sirve como un intermediario entre el modelo, la vista y cualquier otro recurso necesario para procesar la solicitud HTTP y generar una página *Web*.

Entre sus principales ventajas se encuentran [24]:

- ✓ Es liviano. El tamaño de las versiones apenas supera los 2.1 MB. No carga sus componentes y rutinas globalmente, solo lo hace cuando lo requiere.
- ✓ Es rápido.
- ✓ Las URLs generadas por *CodeIgniter* son claras y amigables.
- ✓ Tiene una gama completa de bibliotecas.
- ✓ Es extensible.
- ✓ Su curva de aprendizaje es rápida.
- ✓ Está completamente documentado.

*CodeIgniter* es un *framework* de PHP que tiene versatilidad, es capaz de trabajar la mayoría de los entornos o servidores. Tiene facilidad de instalación y su configuración se realiza con apenas la edición de un archivo. Es bastante ligero, pues sus versiones no sobrepasan los 2,1 MB, lo que permite que el servidor no se sobrecargue interpretando o ejecutando grandes porciones de código. Su documentación es fácil de seguir y de asimilar porque está escrita en modo de tutorial; lo cual facilita mucho la referencia rápida [24].

Para la confección de la aplicación se necesita un *framework* que sea compatible y que permita ordenar el código controlando la aplicación. Por todo lo expuesto, por sus ventajas y por ser el *framework* más ajustado a la infraestructura tecnológica con que cuenta la entidad se escogió para el desarrollo de la aplicación *Web a CodeIgniter*. Además este *framework* posee una amplia y clara documentación.

### **1.5.6 Sistemas Gestores de Bases de Datos**

Un sistema gestor de bases de datos o SGBD (aunque se suele utilizar más a menudo las siglas DBMS procedentes del inglés, Data Base Management System) es el *software* que facilita los procesos de definición de la base de datos donde se especifican tipos de datos, estructuras y restricciones, también permite la construcción de las bases de datos y su manipulación donde se consulta y actualiza el diseño, además de ser donde se generan los informes. En estos sistemas se proporciona un conjunto coordinado de programas, procedimientos y lenguajes que permiten a los distintos usuarios realizar sus tareas habituales con los datos, garantizando además la seguridad de los mismos [25].

Los SGBD poseen ciertas características entre las cuales se encuentran: la integridad de los datos para garantizar su validez, redundancia mínima para evitar la aparición de información repetida, las restricciones de seguridad y confidencialidad para permitir o denegar accesos restringidos según el usuario, el respaldo de información y recuperación para proporcionar una forma eficiente de realizar copias de respaldo de la información almacenada en ellos [25].

Los gestores de base de datos han evolucionado de manera considerable, facilitándose la manipulación de grandes volúmenes de datos. Entre los sistemas de base de datos más utilizados actualmente se encuentran: *PostgreSQL, Oracle, MySQL*.

#### **PostgreSQL**

Es un sistema de base de datos relacional orientada a objetos. Es un proyecto de código libre, soporta casi toda la sintaxis SQL y posibilita muchas características tales como: consultas complejas, integridad referencial, vistas, integridad transaccional, es una de las bases de datos más potentes y robustas del mercado. Maneja muy bien grandes cantidades de datos y su rendimiento es muy bueno ante el acceso múltiple de usuarios al sistema [26].

## Oracle

Es un sistema de base de datos relacional que fue desarrollado por Oracle corporation se considera como uno de los sistemas de base de datos más completos. Surge a finales de los 70, soporta unos 17 idiomas, corre automáticamente en más de 80 arquitecturas de *hardware* y *software* distinto sin tener la necesidad de cambiar una sola línea de código. Además incluye cuatro generaciones de desarrollo de aplicación, herramientas de reportes y utilitarios. Este producto informático a pesar de ser uno de los más vendidos a nivel mundial por su gran potencia, no es tan extendido como otros SGBD, porque tiene un elevado precio y esto hace que sólo se vea en empresas generalmente grandes [17].

## MySQL

Es un sistema de base de datos relacional, multihilo y multiusuario. Desarrollado por la compañía MySQL AB, como *software* libre en un esquema de licenciamiento dual; por un lado GNU Licencia General Publica (GPL) para cualquier uso compatible y por otro, para aquellas empresas que quieran incorporarlo en productos privativos deben comprar a la empresa una licencia específica que les permita este uso. Está desarrollado en su mayor parte en ANSI C. Entre sus principales ventajas se destaca que puede ejecutarse en máquinas con escasos recursos sin ningún problema debido a su bajo consumo [27].

Entre sus principales ventajas se encuentran [17]:

- ✓ Velocidad al realizar las operaciones, lo que le hace uno de los gestores con mejor rendimiento.
- ✓ Bajo costo en requerimientos para la elaboración de bases de datos, ya que debido a su bajo consumo puede ser ejecutado en una máquina con escasos recursos sin ningún problema.
- ✓ Facilidad de configuración e instalación.
- ✓ Se integra muy bien en el desarrollo de aplicaciones *Web* donde el lenguaje de programación es PHP y el servidor *Web* Apache.
- ✓ Es multiplataforma.

MySQL es un sistema gestor de bases de datos relacionales rápido, sólido y flexible. Es idóneo para la creación de bases de datos con acceso desde páginas *Web* dinámicas,

posibilitando realizar múltiples y rápidas consultas. Es muy utilizado en aplicaciones *Web*. Su popularidad como aplicación *Web* está muy ligada a PHP, que a menudo aparece en combinación con MySQL [17].

Tomándose en cuenta lo anteriormente expuesto acerca de los diferentes SGBD, se decidió escoger MySQL por las ventajas que nos brinda, por su velocidad a la hora de realizar las operaciones, por su rendimiento y bajo consumo, además es el manejador de bases de datos más ajustado a los requerimientos del sistema.

### **1.6 Metodologías de desarrollo de Software**

Las Metodologías de desarrollo de *software* surgen debido a la necesidad de emplear una serie de procedimientos y técnicas a la hora de desarrollar un producto de *software*, con el propósito de brindarle una guía al desarrollador. Debido a que no todos los sistemas que se desarrollan tienen la misma complejidad existen una gran variedad de metodologías para la creación de los mismos, están las Metodologías tradicionales orientadas al control de los procesos, estableciendo rigurosamente las actividades a desarrollar, herramientas a utilizar y notaciones que se usarán y las Metodologías ágiles orientadas a la interacción con el cliente y el desarrollo incremental del *software*, mostrando versiones parcialmente funcionales del *software* al cliente en intervalos cortos de tiempo, para que pueda evaluar y sugerir cambios en el producto según se va desarrollando [28].

A continuación se describen algunas de las metodologías existentes para el desarrollo de *software*:

#### **Metodología RUP**

La metodología tradicional RUP (Rational Unified Process), es un proceso formal que provee un acercamiento disciplinado para asignar tareas y responsabilidades dentro de una organización de desarrollo. Su objetivo es asegurar la producción de *software* de alta calidad que satisfaga los requerimientos de los usuarios finales.

RUP hace uso de cuatro etapas en su metodología descritas de la siguiente manera: una fase de concepción, una fase de elaboración, una fase de construcción y una de transición. Dentro de cada una de ellas se realiza varias iteraciones dependiendo de la dimensión del proyecto [29].

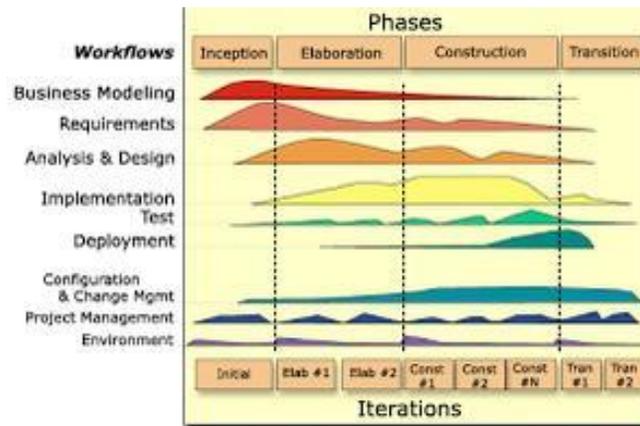


Figura 4. Esquema de trabajo RUP

Es una metodología apropiada para proyectos grandes, dado que requiere un equipo de trabajo capaz de administrar un proceso complejo en varias etapas. Una de sus limitaciones es que en proyectos pequeños, es posible que no se puedan cubrir los costos de dedicación del equipo de profesionales necesarios [30].

### Metodología XP

Es una metodología ágil centrada en potenciar las relaciones interpersonales como clave para el éxito en desarrollo de *software*, convirtiendo al cliente y desarrollador en un equipo y propiciando un buen clima de trabajo. XP (eXtremeProgramming), se define como especialmente adecuada para proyectos con requisitos imprecisos y muy cambiantes donde existe un alto riesgo técnico y es recomendado emplearla solo en proyectos a corto plazo [31].

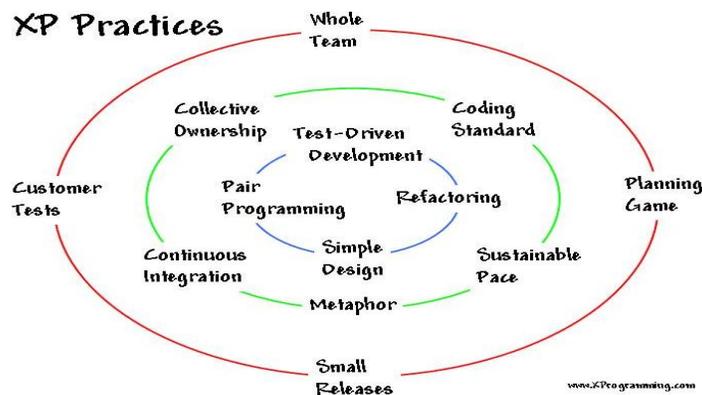


Figura 5. Esquema de XP

## ICONIX

Es una metodología de desarrollo del *software* que se halla a medio camino entre la complejidad de RUP y la simplicidad de XP, es una metodología simplificada en comparación a otras más tradicionales, la cual unifica un conjunto de métodos de orientación a objetos con el objetivo de tener un control estricto sobre todo el ciclo de vida del producto a realizar. La misma cuenta con una secuencia de pasos que se deben seguir y determina claramente las actividades a desarrollar en cada etapa del ciclo de vida del proyecto que la utilice [31].

Iconix se guía a través de casos de uso y sigue un ciclo de vida iterativo e incremental. El objetivo es que a partir de los casos de uso se obtenga el sistema final. Presenta claramente las actividades de cada fase y exhibe una secuencia de pasos que deben ser seguidos, ofrece trazabilidad, que no es más que la capacidad de seguir una relación entre los diferentes artefactos del *software* producidos. La metodología brinda un uso dinámico del UML porque utiliza solo algunos diagramas, al contrario de RUP [28].

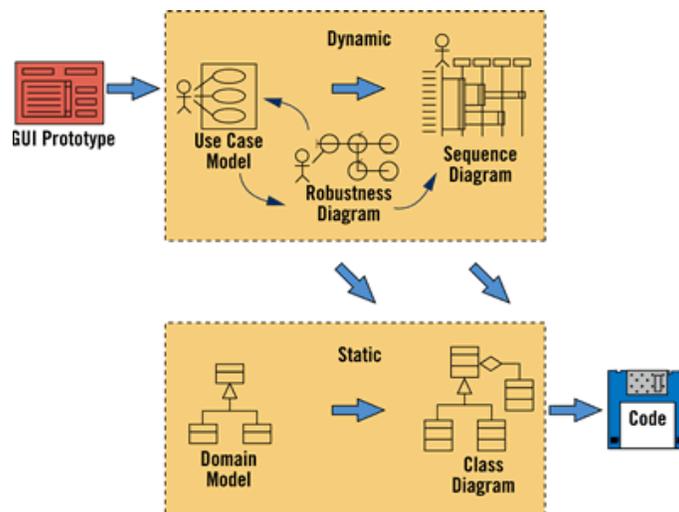


Figura 6. Esquema de trabajo de ICONIX

La metodología ICONIX, está basada en el desarrollo de sistemas a partir del análisis y la documentación. Busca tener una retroactividad con el cliente, en la mitad de los procedimientos, comenzando con un prototipo en donde el analista y el cliente definirán pantallas, funcionalidades, todo lo que se espera obtener del programa. Se definen los modelos de casos de uso, de secuencia y de robustez, con la finalidad de conseguir un buen

sistema. Esta metodología favorece la participación de los usuarios finales y la documentación de todo el proceso [31].

Cada metodología posee características peculiares que facilitan la selección de una de éstas a partir de las necesidades de la organización y las características específicas del proyecto a desarrollar. Se escogió la metodología Iconix para el desarrollo del *software* por ser la más apropiada, ya que facilita en gran manera el proceso de diseño de un *software* y provee los artefactos y herramientas necesarias para cada etapa del desarrollo.

### **1.7 Conclusiones del capítulo**

Se realizó un estudio sobre las bases teóricas que fundamentan la presente investigación y las tecnologías para el desarrollo de aplicaciones empresariales con arquitectura fundamentalmente *Web*, destacándose aquellas que se adaptan mejor a las necesidades de la aplicación que se desea implementar.

- ✓ El estudio realizado dio como resultado la necesidad de realizar una aplicación *Web* que controle y agilice el proceso de gestión de los expedientes en el departamento administrativo de la Vivienda de la Dirección Provincial de Holguín.
- ✓ Se seleccionaron como herramientas para el desarrollo de la aplicación el lenguaje de programación PHP, el *framework CodeIgniter*, empleando *MySQL* como gestor de base de datos y el servidor *Web Apache*. La metodología de desarrollo seleccionada **ICONIX**.
- ✓ Las herramientas y tecnologías propuestas se adaptan correctamente al marco donde se desarrollará la aplicación, asegurando una correcta compatibilidad.

## CAPÍTULO II. DESCRIPCIÓN DE LA SOLUCIÓN PROPUESTA

En este capítulo utilizando como guía la metodología de desarrollo de *software* Iconix se muestra el desarrollo del sistema, donde se describe como ir desde los casos de uso hasta la codificación del *software*. Además se detalla el procedimiento de valoración de sostenibilidad realizado al producto informático y se hará una exposición de los resultados logrados con la implementación de la aplicación, a través de encuestas.

### 2.1 Modelo del Dominio

El modelo del dominio es un artefacto basado en conceptos relacionados con el problema, el cual permite ser refinado y actualizado a lo largo y ancho del proyecto, por lo que siempre refleja la comprensión actual del espacio del problema. Cuando se crea el modelo del dominio, se está creando y construyendo una representación de los objetos, glosario o un diccionario de términos, formándose las bases para la construcción de los casos de uso. Además proporciona un vocabulario común que permite la comunicación entre los usuarios y los desarrolladores [12].

El modelo del dominio está compuesto por varios pasos los cuales son:

- ✓ Listar los requerimientos del sistema.
- ✓ Extraer los sustantivos o frases sustantivas de los requerimientos.
- ✓ Realizar el diagrama del modelo de dominio.

Los requerimientos pueden dividirse en dos tipos funcionales y no funcionales. Los requerimientos funcionales son los que definen las funciones que el sistema será capaz de realizar y los no funcionales son los que tienen que ver con las características que de una u otra forma pueden limitar el sistema. La realización de una buena captura de requerimientos está ligada al nivel de comunicación establecido entre el cliente y los desarrolladores [32].

#### 2.1.2 Requerimientos Funcionales

1. El sistema debe ser capaz de gestionar la información de los usuarios que pueden interactuar con el sistema estableciendo el nivel de acceso.
  - a) El sistema debe ser capaz de permitirle al **administrador** insertar un nuevo **usuario**.

- b) El sistema debe ser capaz de permitirle al **administrador** modificar un **usuario** existente.
  - c) El sistema debe ser capaz de permitirle al **administrador** eliminar un **usuario**.
  - d) El sistema debe ser capaz de permitirle al **administrador** buscar un **usuario**.
  - e) El sistema debe ser capaz de permitirle al **administrador** ver una lista con los **usuarios** existentes.
2. El sistema debe ser capaz de gestionar la información del estado técnico.
- a) El sistema debe ser capaz de permitirle al **administrador** insertar un nuevo **estado técnico**.
  - b) El sistema debe ser capaz de permitirle al **administrador** modificar un **estado técnico** existente.
  - c) El sistema debe ser capaz de permitirle al **administrador** eliminar un **estado técnico**.
  - d) El sistema debe ser capaz de permitirle al **administrador** buscar un **estado técnico**.
  - e) El sistema debe ser capaz de permitirle al **administrador** ver una lista con los **estados técnicos** existentes.
3. El sistema debe ser capaz de gestionar la información de las tipologías.
- a) El sistema debe ser capaz de permitirle al **administrador** insertar una nueva **tipología**.
  - b) El sistema debe ser capaz de permitirle al **administrador** modificar una **tipología** existente.
  - c) El sistema debe ser capaz de permitirle al **administrador** eliminar una **tipología**.
  - d) El sistema debe ser capaz de permitirle al **administrador** buscar una **tipología**.
  - e) El sistema debe ser capaz de permitirle al **administrador** ver una lista con las **tipologías** existentes.
4. El sistema debe ser capaz de gestionar la información de las categorías.
- a) El sistema debe ser capaz de permitirle al **administrador** insertar una nueva **categoría**.

- b) El sistema debe ser capaz de permitirle al **administrador** modificar una **categoría** existente.
  - c) El sistema debe ser capaz de permitirle al **administrador** eliminar una **categoría**.
  - d) El sistema debe ser capaz de permitirle al **administrador** buscar una **categoría**.
  - e) El sistema debe ser capaz de permitirle al **administrador** ver una lista con las **categorías** existentes.
5. El sistema debe ser capaz de gestionar la información de los tipos de pago.
- a) El sistema debe ser capaz de permitirle al **administrador** insertar un nuevo **tipo de pago**.
  - b) El sistema debe ser capaz de permitirle al **administrador** modificar un **tipo de pago** existente.
  - c) El sistema debe ser capaz de permitirle al **administrador** eliminar un **tipo de pago**.
  - d) El sistema debe ser capaz de permitirle al **administrador** buscar un **tipo de pago**.
  - e) El sistema debe ser capaz de permitirle al **administrador** ver una lista con los **tipos de pagos** existentes.
6. El sistema debe ser capaz de gestionar la información de los organismos de la administración central del estado.
- a) El sistema debe ser capaz de permitirle al **administrador** insertar un nuevo **organismo**.
  - b) El sistema debe ser capaz de permitirle al **administrador** modificar un **organismo** existente.
  - c) El sistema debe ser capaz de permitirle al **administrador** eliminar un **organismo**.
  - d) El sistema debe ser capaz de permitirle al **administrador** buscar un **organismo**.
  - e) El sistema debe ser capaz de permitirle al **administrador** ver una lista con los **organismos** existentes.
7. El sistema debe ser capaz de gestionar la información de las organizaciones como personas jurídicas.

- a) El sistema debe ser capaz de permitirle al **administrador** insertar una nueva **organización**.
  - b) El sistema debe ser capaz de permitirle al **administrador** modificar una **organización** existente.
  - c) El sistema debe ser capaz de permitirle al **administrador** eliminar una **organización**.
  - d) El sistema debe ser capaz de permitirle al **administrador** buscar una **organización**.
  - e) El sistema debe ser capaz de permitirle al **administrador** ver una lista con las **organizaciones** existentes.
8. El sistema debe ser capaz de gestionar la información de los técnicos.
- a) El sistema debe ser capaz de permitirle al **administrador** insertar un nuevo **técnico**.
  - b) El sistema debe ser capaz de permitirle al **administrador** modificar el **técnico** existente.
  - c) El sistema debe ser capaz de permitirle al **administrador** eliminar un **técnico**.
  - d) El sistema debe ser capaz de permitirle al **administrador** buscar un **técnico**.
  - e) El sistema debe ser capaz de permitirle al **administrador** ver una lista con los **técnicos existentes**.
9. El sistema debe ser capaz de permitirle al **administrador** realizar una salva de la base de datos, ya sea en el propio disco duro o en un soporte externo.
10. El sistema debe ser capaz de permitirle al **administrador** visualizar una lista con las **trazas del sistema**.
11. El sistema debe ser capaz de gestionar los expedientes ya sea de viviendas vinculadas, viviendas medio básicas, comunidades, cuarterías y edificios multifamiliares.
- a) El sistema debe ser capaz de permitirle al **operador** insertar un nuevo **expediente** (vivienda vinculada, medio básico, comunidad, cuartería, edificio multifamiliar).

- b) El sistema debe ser capaz de permitirle al **operador** modificar un **expediente** (vivienda vinculada, medio básico, comunidad, cuartería, edificio multifamiliar) existente.
  - c) El sistema debe ser capaz de permitirle al **operador** eliminar un **expediente** (vivienda vinculada, medio básico, comunidad, cuartería, edificio multifamiliar).
  - d) El sistema debe ser capaz de permitirle al **operador** buscar un **expediente** (vivienda vinculada, medio básico, comunidad, cuartería, edificio multifamiliar).
  - e) El sistema debe ser capaz de permitirle al **operador** ver una lista con los **expedientes** (vivienda vinculada, medio básico, comunidad, cuartería, edificio multifamiliar).
12. El sistema debe ser capaz de gestionar los expedientes de ilegalidades que se les realizan a las viviendas vinculadas, medios básicos y edificios multifamiliares.
- a) El sistema debe ser capaz de permitirle al **operador** insertar un nuevo **expediente de ilegalidad** (vivienda vinculada, medio básico, edificio multifamiliar).
  - b) El sistema debe ser capaz de permitirle al **operador** modificar un **expediente de ilegalidad** (vivienda vinculada, medio básico, edificio multifamiliar) existente.
  - c) El sistema debe ser capaz de permitirle al **operador** eliminar un **expediente de ilegalidad** (vivienda vinculada, medio básico, edificio multifamiliar).
13. El sistema debe ser capaz de gestionar los expedientes de ceses que se les realizan a las viviendas vinculadas por arribar al tiempo establecido (20 años).
- a) El sistema debe ser capaz de permitirle al **operador** insertar un nuevo **expediente de cese**.
  - b) El sistema debe ser capaz de permitirle al **operador** modificar un **expediente de cese** existente.
  - c) El sistema debe ser capaz de permitirle al **operador** eliminar un **expediente de cese**.

14. El sistema debe ser capaz de enviar una notificación al **operador** de los expedientes de las viviendas vinculadas que serán cesadas por cumplir con el tiempo establecido (20 años) donde se muestre el número del expediente.
15. El sistema debe ser capaz de permitirle a cada **usuario**.
  - a) Autenticarse
  - b) Cambiar contraseña
  - c) Salir del sistema.
16. El sistema debe ser capaz de permitirle al **consultor** mostrar, Imprimir y exportar a (Excel) los reportes según los parámetros que escoja mensualmente, trimestralmente y anualmente por municipio y reparto.
  - a) Composición del fondo habitacional.
  - b) Composición del fondo habitacional a partir de la selección de los expedientes de administración de la vivienda.
  - c) Estado de las ilegalidades según los expedientes de administración de la vivienda.
  - d) Mostrar los ceses de las viviendas vinculadas.

### **Requerimientos no funcionales**

#### **Apariencia o interfaz externa**

Diseño sencillo e intuitivo, sugerente, que permita a personas de poca experiencia trabajar en el sistema cómodamente. Los íconos muestran acciones a realizar (insertar, editar, eliminar, etc.). Mantener un sistema de mensajes informativos a lo largo de toda la aplicación en los casos de acciones realizadas o de errores cometidos por el usuario.

El color predominante en el diseño de la Aplicación es el blanco y el gris con utilización del rojo en tonos que no molestan al usuario, para compatibilidad con el logotipo de la Dirección Provincial de la Vivienda. De manera tal que facilite la lectura de la información que se muestre, la visualización de los elementos y evite el agotamiento visual del usuario.

#### **Usabilidad**

El sistema podrá ser usado por cualquier persona que posea conocimientos básicos en el manejo de la computación y de un ambiente *Web* de manera general. Se puede acceder al

sistema desde todos los departamentos de la entidad que cuenten con conectividad. Además debe requerir de poco esfuerzo para interactuar con el sistema.

### **Portabilidad**

El *software* puede ser usado bajo los sistemas operativos Windows, Linux o Unix, debe correr sobre plataforma *Web*, codificada en PHP. Las herramientas utilizadas para el desarrollo del sistema son tecnologías de *software* libre.

### **Seguridad**

La seguridad constituye un factor de vital importancia en un sistema permitiendo la confidencialidad de los datos en la transmisión y en el almacenamiento. El *software* cuenta con un sistema de acceso por contraseña, permitiendo que cada usuario acceda a las opciones que le sean permitidas según el rol que se haya asignado, garantizando la confidencialidad de la información. Además cuenta con un registro de trazas que permite poder auditar qué acciones ha realizado el usuario.

### **Rendimiento**

El sistema debe trabajar con precisión y velocidad, efectuando el acceso a la base de datos de forma rápida evitando demoras.

### **Software**

Para el funcionamiento del sistema en el servidor se necesita el sistema operativo Windows 2003 server o superior o Linux y como servidor de base de datos MySQL y tener actualizado los Service Packs y parches.

Para el funcionamiento del sistema en los clientes se necesita Windows 98 o superior o Linux. Los clientes deben tener la capacidad de ejecutar cualquier navegador *Web*, aunque se recomienda el uso del navegador firefox 19.0.2 y tener actualizado los Service Packs y parches.

### **Hardware**

Requisitos del servidor: Características mínimas para un servidor genérico.

- ✓ Procesador Intel Pentium IV a 1.0 GHz o superior.
- ✓ Memoria RAM de 512 o superior.
- ✓ Espacio en disco de 40 GB disponible o más.

Requisitos de los clientes:

- ✓ Procesador Intel Celeron a 1.0 GHz o superior.
- ✓ Memoria RAM de 256 o superior.
- ✓ Espacio en disco de 10 GB disponible o más.

### 2.1.3 Definición de los principales objetos del modelo del dominio

Luego de listar los requerimientos del sistema se realiza una inspección gramatical de los elementos señalados, los cuales se ordenan alfabéticamente y se eliminan los términos repetidos. Terminado el proceso se procede a confeccionar el diagrama del modelo del dominio.

A continuación en la tabla 1 se explica el significado de cada entidad representada en el modelo del dominio.

<b>Objeto</b>	<b>Definición operacional</b>
<b>Técnico</b>	Persona encargada de recoger los datos primarios de los inmuebles.
<b>Estado Técnico</b>	Clasificación de los inmuebles teniendo en cuenta su estado constructivo [1].
<b>Tipología</b>	Clasificación de los inmuebles por ejemplo: I, II, III, IV [1].
<b>Categoría</b>	Clasificación de los inmuebles por sus características por ejemplo: A, B, C, D.
<b>Tipo de pago</b>	Clasificación de los pagos por ejemplo: Pago por BPA, caja, control de fondo, chequera.
<b>Organización</b>	Es un sistema para que los recursos humanos, financieros, físicos de información y otros de forma coordinada, ordenada y regulada por un conjunto de



---

	normas, logren determinados fines [33].
<b>Expediente de cese</b>	Documento oficial donde se registra los datos del cese de la vivienda vinculada. Se entiende por cese aquellas viviendas vinculadas que arriban al término legalmente establecido (20) años vividos [5].
<b>Expediente de comunidad</b>	Documento oficial donde se registra los datos de las comunidades las cuales se consideran comunidad cuando las viviendas vinculadas representan el 50% +1 de las viviendas asentadas.
<b>Expediente de cuartería</b>	Documento oficial donde se registra los datos de las cuarterías, entiéndase por cuartería, toda edificación que contando con un solo piso con varias viviendas, éstas tengan elementos comunes de servicios.
<b>Expediente de vivienda medio básico</b>	Documento oficial donde se registra los datos de las viviendas medios básicos. Consideradas como medio básico aquellas que forman parte del patrimonio de entidades, bien sea que el personal que la ocupa lo hará sólo por determinado período o porque la vivienda en cuestión tenga o deba tener determinadas facilidades para el desempeño de un cargo o función específica [4].
<b>Expediente de vivienda vinculada</b>	Documento oficial donde se registra los datos de las viviendas vinculadas. Clasificadas por ser aquéllas que se hayan destinado o se destinen, se hayan construido o se construyan, para propiciar el aseguramiento y permanencia estable y prolongada de la fuerza de

	trabajo de entidades [4].
<b>Expediente de edificio multifamiliar</b>	Documento oficial donde se registra los datos de un Edificio Multifamiliar, el cual está definido como toda edificación de varios pisos en que existan viviendas independientes, que ocupan cada una de ellas todo o parte de un piso [3].
<b>Expediente de ilegalidad</b>	Documento oficial donde se registra las violaciones cometidas en las viviendas vinculadas, medios básicos y edificios multifamiliares. Entiéndase por violación el incumplimiento en cuanto a lo establecido en las resoluciones de la vivienda.
<b>Organismo</b>	Es la organización rectora o ministerio al que se subordinan las organizaciones que existen en el país.
<b>Trazas</b>	Registro de seguridad del sistema que permite conocer las acciones de los usuarios dentro del mismo.
<b>Usuario</b>	Representan todas las personas a quien va dirigido el sistema informático y tienen acceso a entrar y salir del mismo para consultar información. A estas personas se llaman también actores del sistema.

Tabla 1. Glosario de Términos

### Diagrama del Modelo del Dominio

En la Figura 7 se muestra el diagrama del modelo del dominio con sus entidades y relaciones.

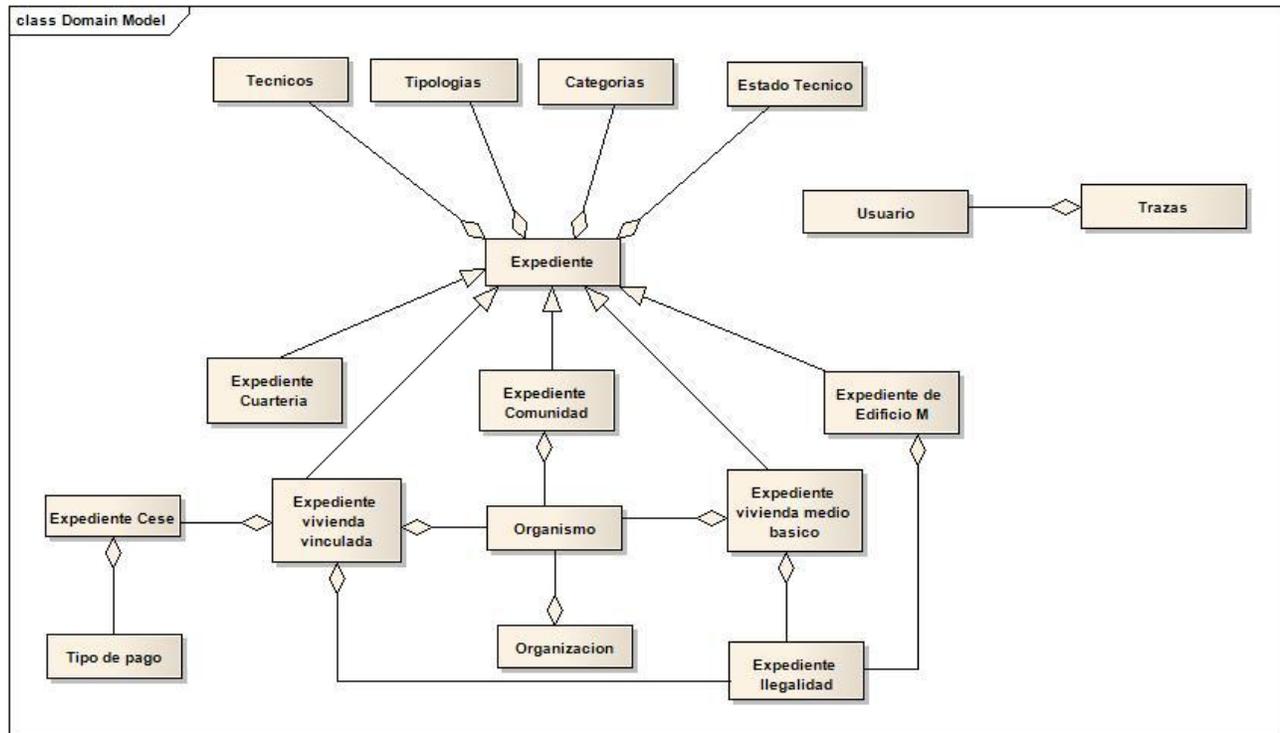


Figura 7. Modelo del Dominio

## 2.2 Modelación de los casos de uso

El modelo de casos de uso es una técnica para especificar el comportamiento de un sistema y se parte de la identificación de los requerimientos funcionales. En el modelo se describe lo que hace el sistema para cada tipo de usuario, este es de vital importancia, pues mediante él los desarrolladores se encargan de asegurar que los proyectos se enfoquen a las exigencias requeridas por los usuarios y clientes [32].

### Actores del sistema

Se puede determinar los actores del sistema examinando a los usuarios que lo utilizarán y a otros sistemas que deben interactuar con él. Cada categoría de usuarios o sistemas que interactúan se representan por tanto como actores [34].

A continuación se muestra el diagrama de actores del sistema.

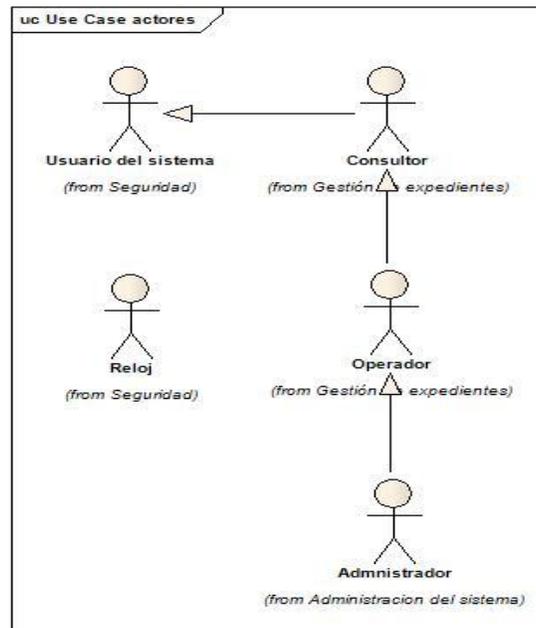


Figura 8. Actores del sistema

Los actores son llamados usuario, aunque le es dado el nombre del rol específico que desempeña en el sistema. En la tabla 2 se identificaron los actores siguientes:

Actores del Sistema	Descripción
<b>Administrador</b>	Es el responsable de la gestión de los usuarios y de la configuración necesaria para el correcto funcionamiento del sistema. Además de controlar las acciones que hace cada usuario y hacer salvadas de la base de datos.
<b>Operador</b>	Es el responsable de gestionar los diferentes expedientes con que trabaja el sistema (edificio multifamiliar, medio básico, vivienda vinculada, cuartería, comunidad, ilegalidad en vivienda vinculada y medio básico, ilegalidad en edificio multifamiliar y los ceses de las viviendas vinculadas).
<b>Consultor</b>	Es el que visualiza el estado de los datos.

<b>Reloj</b>	Es un actor ficticio que se encarga de realizar la acción de notificación de las viviendas vinculadas que deben cesar.
--------------	--

Tabla 2. Actores del sistema

Para facilitar el entendimiento y gracias a las flexibilidades brindadas por UML se han dividido los casos de uso en tres paquetes o subsistemas en la presente investigación: Seguridad, Administración del sistema y Gestión de expedientes.

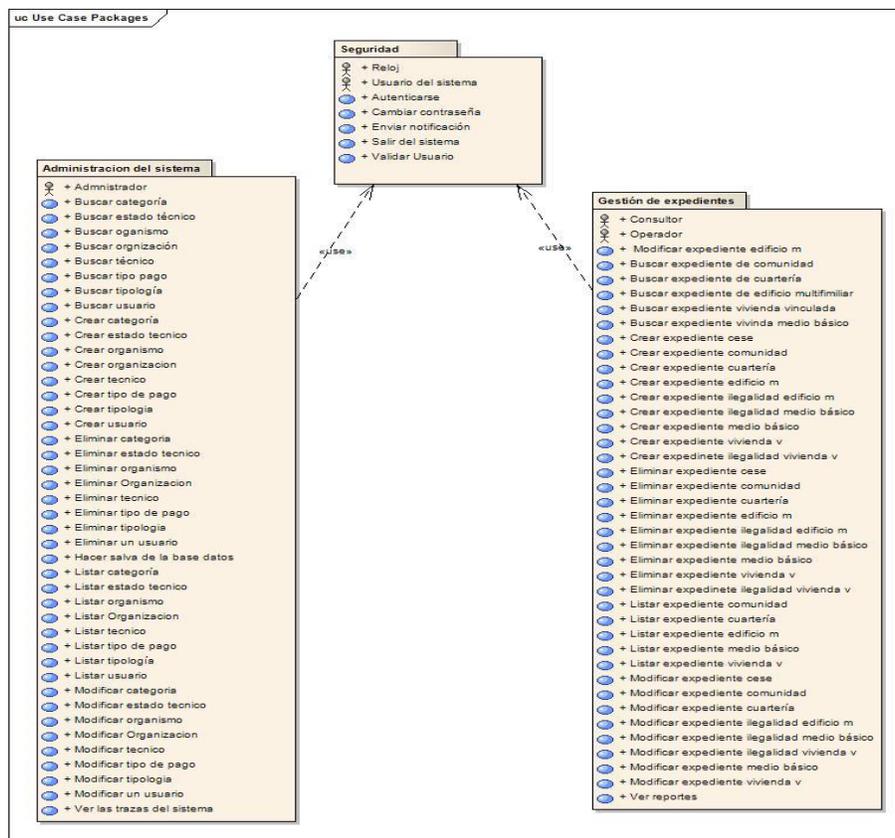


Figura 9. Diagrama de paquetes

### Casos de Uso relacionados con el paquete Seguridad.

El paquete Seguridad tiene el objetivo de gestionar los diferentes componentes de la seguridad del sistema, a través del chequeo de los permisos de los usuarios y de cada una de las acciones que se realicen en el sistema.

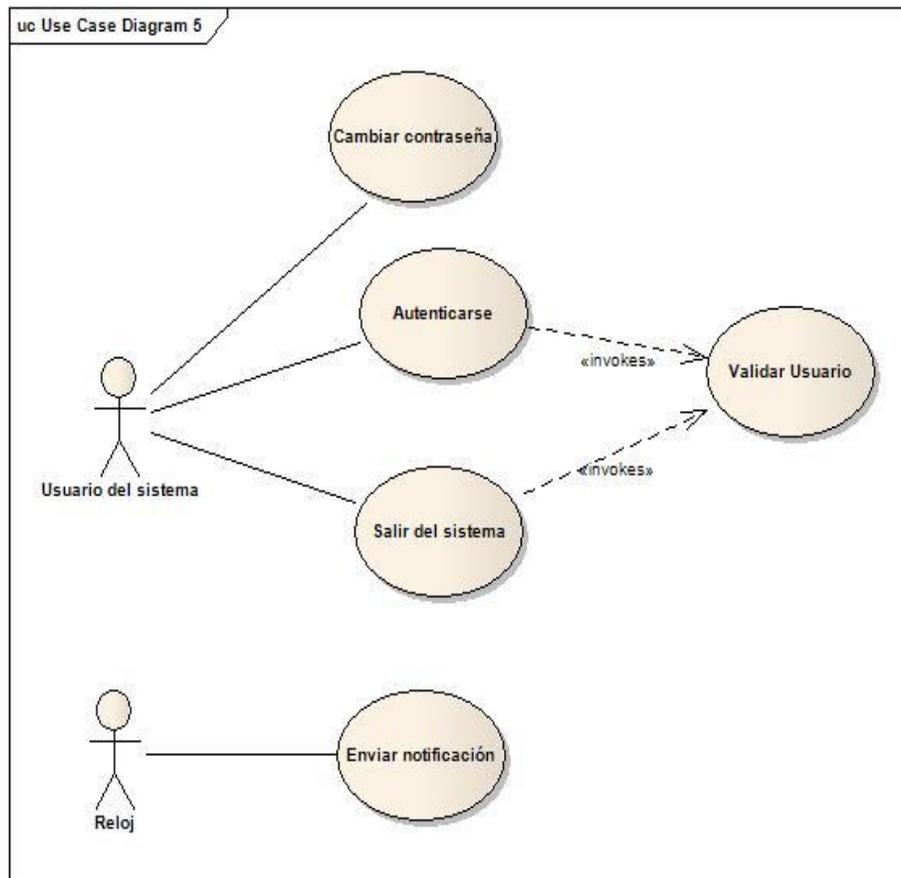


Figura 10. Diagrama de casos de uso del paquete seguridad

### **Casos de Uso relacionados con el paquete Administración del sistema.**

El paquete Administración del sistema tiene como objetivo la gestión de usuario, las trazas, la realización de la salva de la base de datos y la configuración del sistema como la tipología, categoría, estado técnico entre otros.

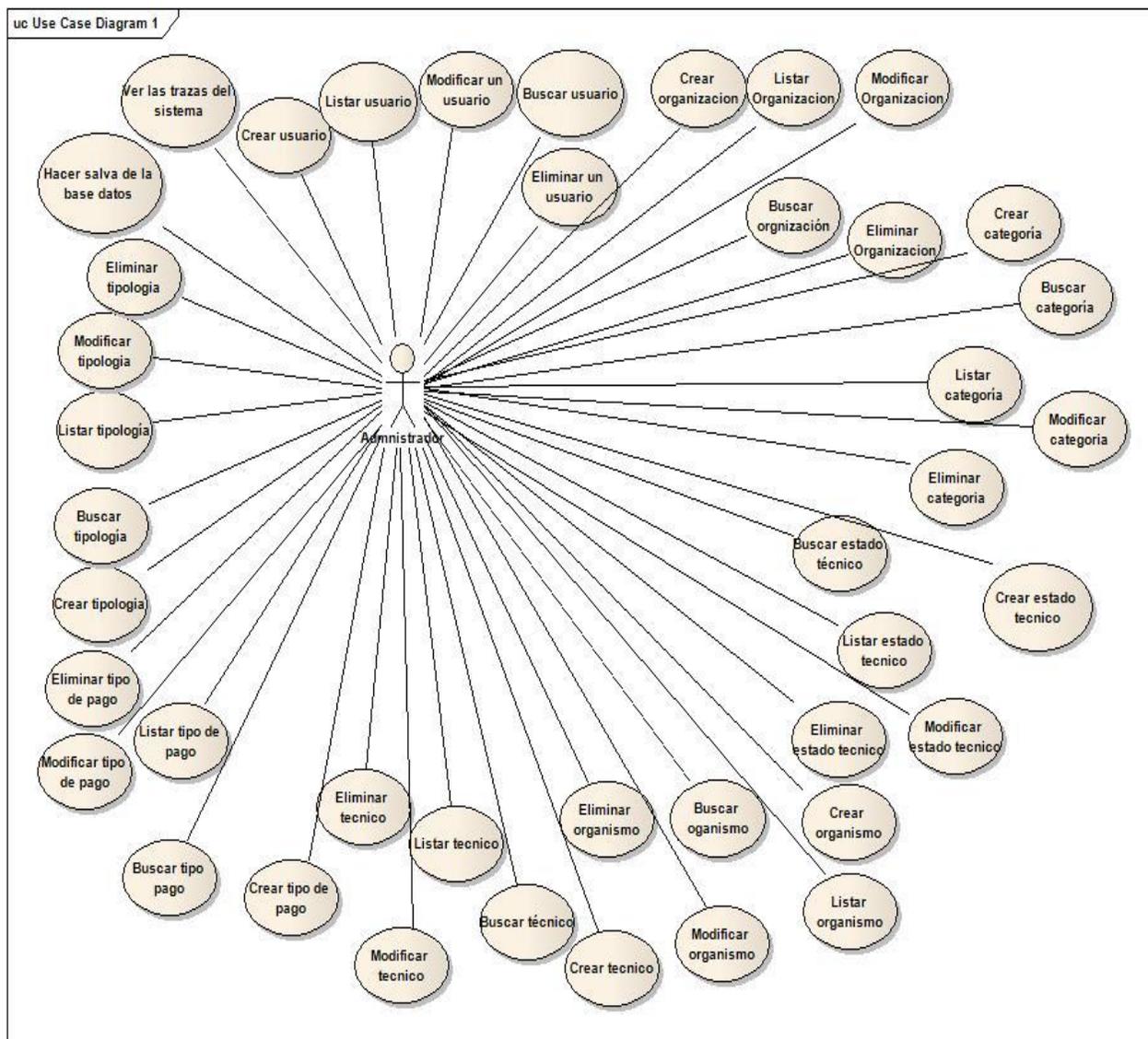


Figura 11. Diagrama de casos de uso del paquete Administración del sistema

**Casos de Uso relacionados con el paquete Gestión de expedientes.**

El paquete Gestión de expedientes tiene como objetivo la gestión de los expedientes administración de la vivienda y los reportes del sistema.

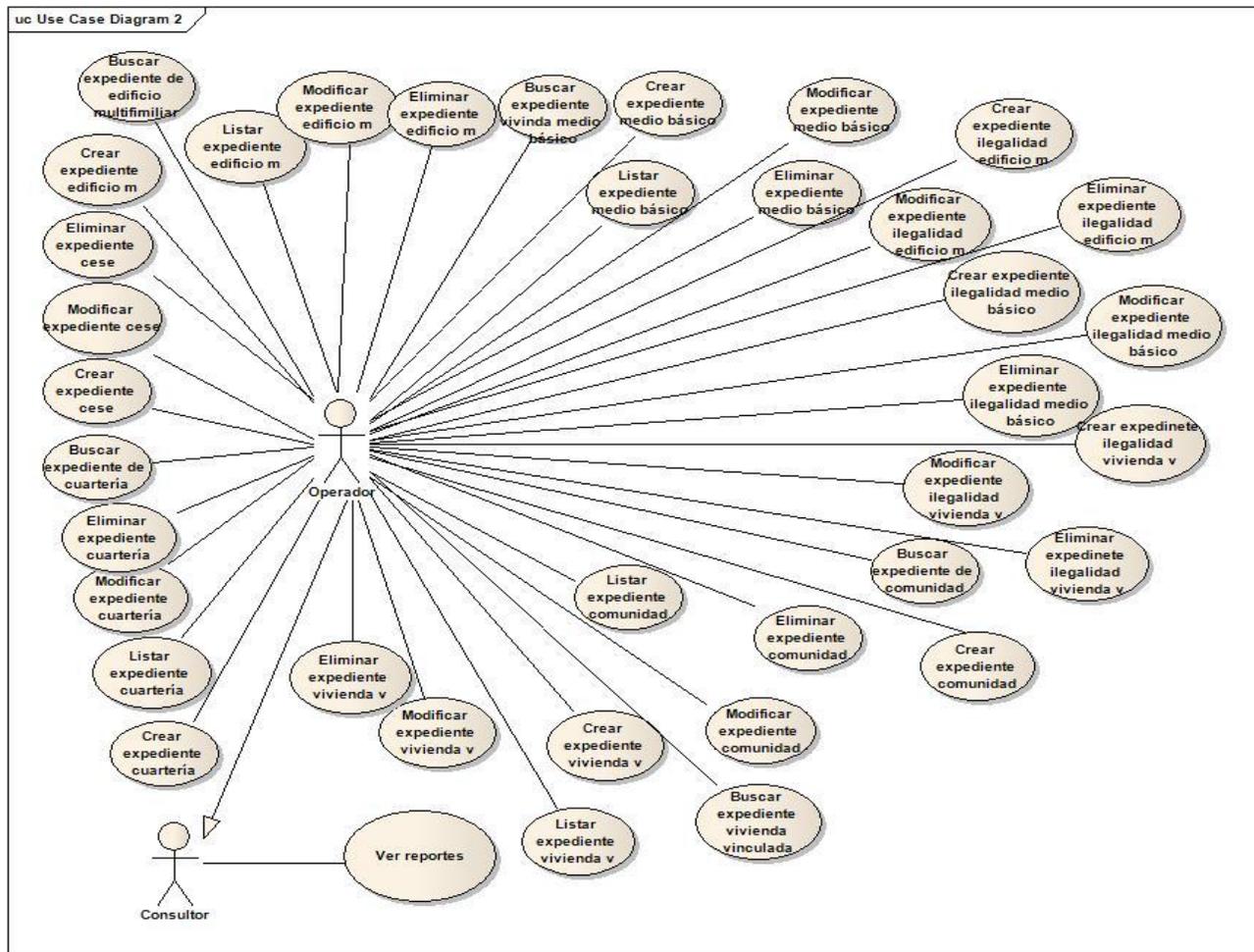


Figura 12. Diagrama de casos de uso del paquete gestión de expedientes

### 2.3 Análisis y Diseño Preliminar

En el análisis y diseño se describen los casos de uso y se realiza el diagrama de robustez. Asegurándose que exista concordancia entre los diagramas y el texto del caso de uso. Ambos deben estar completos y representando correctamente el comportamiento del sistema. También debe asegurarse que coincida el modelo del dominio con el diagrama de robustez, especialmente que todos los objetos entidad, que se muestran en los diagramas de robustez deben estar representados en el modelo del dominio [35].

#### 2.3.1 Descripción Textual de los Principales Casos de Uso del Sistema

En la descripción textual de los casos de uso se reflejan dos cursos: un curso básico que muestra el camino óptimo del proceso y un curso alternativo para describir los posibles errores

que pueden ocurrir durante las verificaciones que realiza el sistema. Estos deben guardar estrecha correspondencia con el manual de usuario y la interfaz gráfica [36]. A continuación se presenta la descripción de los casos de uso crear expediente de cuartería, modificar organismo y ver trazas del sistema en la tabla 3, 4 y 5. Las restantes descripciones se encuentran una parte en los anexos (Ver Anexo 1) y la totalidad de estas se encuentran en el archivo que se entregará adjunto a este documento.

<b>Caso de uso del sistema: Crear expediente de cuartería</b>	
<b>Curso Básico</b>	El Operador escoge la opción Cuartería dentro de la pantalla principal, el sistema valida que el usuario tenga los permisos para acceder a la pantalla, luego muestra la pantalla correspondiente a Expedientes de Cuarterías y hace clic en el ícono Registrar nuevo expediente y pasa a la pantalla Nuevo Expedientes de Cuartería. El Operador escoge el municipio, reparto, estado técnico, tipología, categoría, técnico e introduce los datos fecha, dirección, cantidad cuartos, baños, junta de vecinos, reglamento interno y hace clic en el botón Registrar expediente. El sistema valida que no haya campos vacíos, caracteres inadecuados y almacena la información mostrando un mensaje de confirmación.
<b>Cursos Alternos</b>	<p>Validar: si no tiene permisos para acceder a la pantalla solicitada el sistema muestra un mensaje de error no tiene acceso a este módulo.</p> <p>Valida datos: el administrador introduce los datos en un formato incorrecto y deja campos en blanco. El sistema muestra un mensaje de error.</p>

Tabla 3. Descripción textual del caso de uso crear expediente de cuartería

<b>Caso de uso del sistema: Modificar organismo</b>	
<b>Curso Básico</b>	<p>El Administrador escoge la opción organismo dentro de la pantalla principal , el sistema valida que el usuario tenga los permisos para acceder a la pantalla, luego muestra la pantalla correspondiente a Administración de organismo, luego hace clic en el ícono modificar organismo y pasa a la pantalla editando organismo mostrando los datos del organismo. El Administrador modifica los datos que desee cambiar y hace clic en el botón Guardar. El sistema valida que no haya campos vacíos o caracteres inadecuados y comprueba que no exista el organismo en la Base de Datos, almacena la información y muestra un mensaje de confirmación.</p>
<b>Cursos Alternos</b>	<p>Validar: el sistema muestra un mensaje de error indicando que no puede acceder al módulo solicitado.</p> <p>Valida datos: el administrador introduce los datos en un formato incorrecto, deja campos en blanco. El sistema muestra un mensaje de error.</p> <p>Organismo existente: el administrador introduce un organismo existente y el sistema muestra un mensaje de error.</p>

Tabla 4. Descripción textual del caso de uso modificar organismo

<b>Caso de uso del sistema: Ver trazas del sistema</b>	
<b>Curso Básico</b>	<p>El Administrador escoge la opción ver trazas dentro de la pantalla principal, el sistema valida que el usuario tenga los permisos para acceder a la pantalla, luego muestra la pantalla correspondiente a Reporte de trazas del sistema. El</p>

	administrador introduce la fecha inicio y escoge el usuario el sistema muestra las trazas existentes hasta el momento (en caso de que hayan trazas en la base de datos, sino muestra el mensaje de que no hay datos para mostrar).
<b>Cursos Alternos</b>	Validar: el sistema muestra un mensaje de error indicando que no puede acceder al módulo solicitado.

Tabla 5. Descripción textual del caso de uso ver trazas del sistema

### 2.3.2 Diagramas de Robustez

Según la metodología ICONIX, para llegar desde los casos de uso a un diseño detallado (y luego al código), los mismos se necesitan vincular a los objetos. El análisis de los diagramas de robustez ayuda a identificar los objetos de cada caso de uso, representando el flujo principal de acciones, flujos alternos y excepciones. Esto permite saber si las especificaciones del sistema son razonables [28].

Los objetos que conforman el diagrama de robustez son:

- ✓ Los objetos interfaces: se utilizan para describir qué usa el actor para comunicarse con el sistema.
- ✓ Los objetos controladores: son objetos de control, la mayoría de las veces representan verbos titulares para funciones del sistema. Normalmente se ubican entre las interfaces y entidades.
- ✓ Los objetos entidades: generalmente son los objetos del modelo del dominio.

A continuación se muestra en las Figuras 13, 14 y 15 los diagramas de robustez asociado a los casos de uso crear expediente de cuartería, modificar organismo y ver trazas del sistema. Los diagramas de robustez asociados a los casos de uso restantes se encuentran una parte en los anexos (Ver Anexo 2) y la totalidad de estos se encuentran en el archivo que se entregará adjunto a este documento.

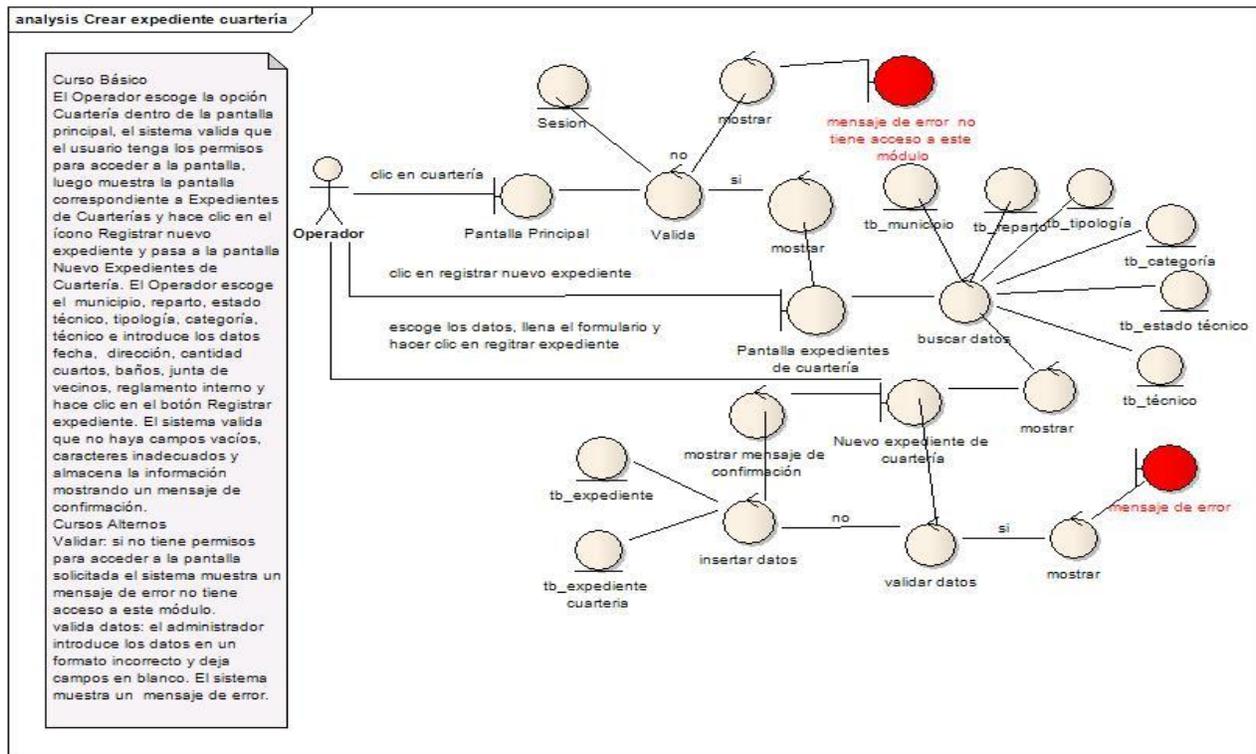


Figura 13. Diagrama de robustez del caso de uso crear expediente de cuartería

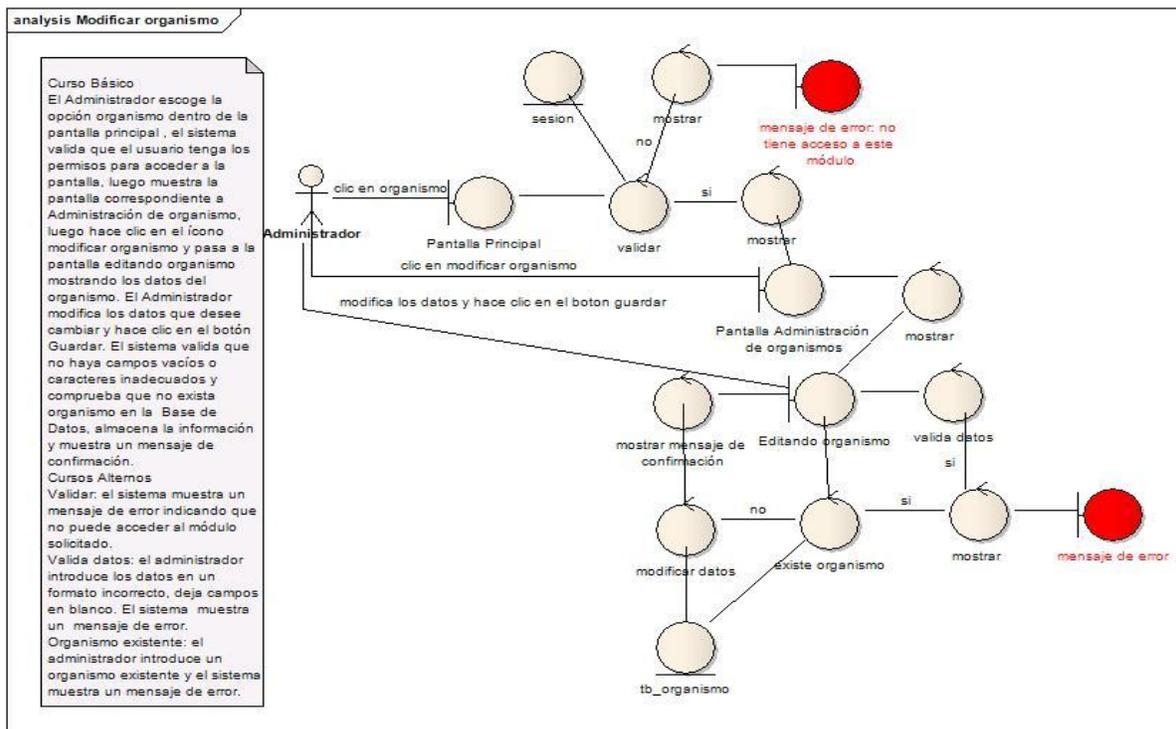


Figura 14. Diagrama de robustez del caso de uso modificar organismo

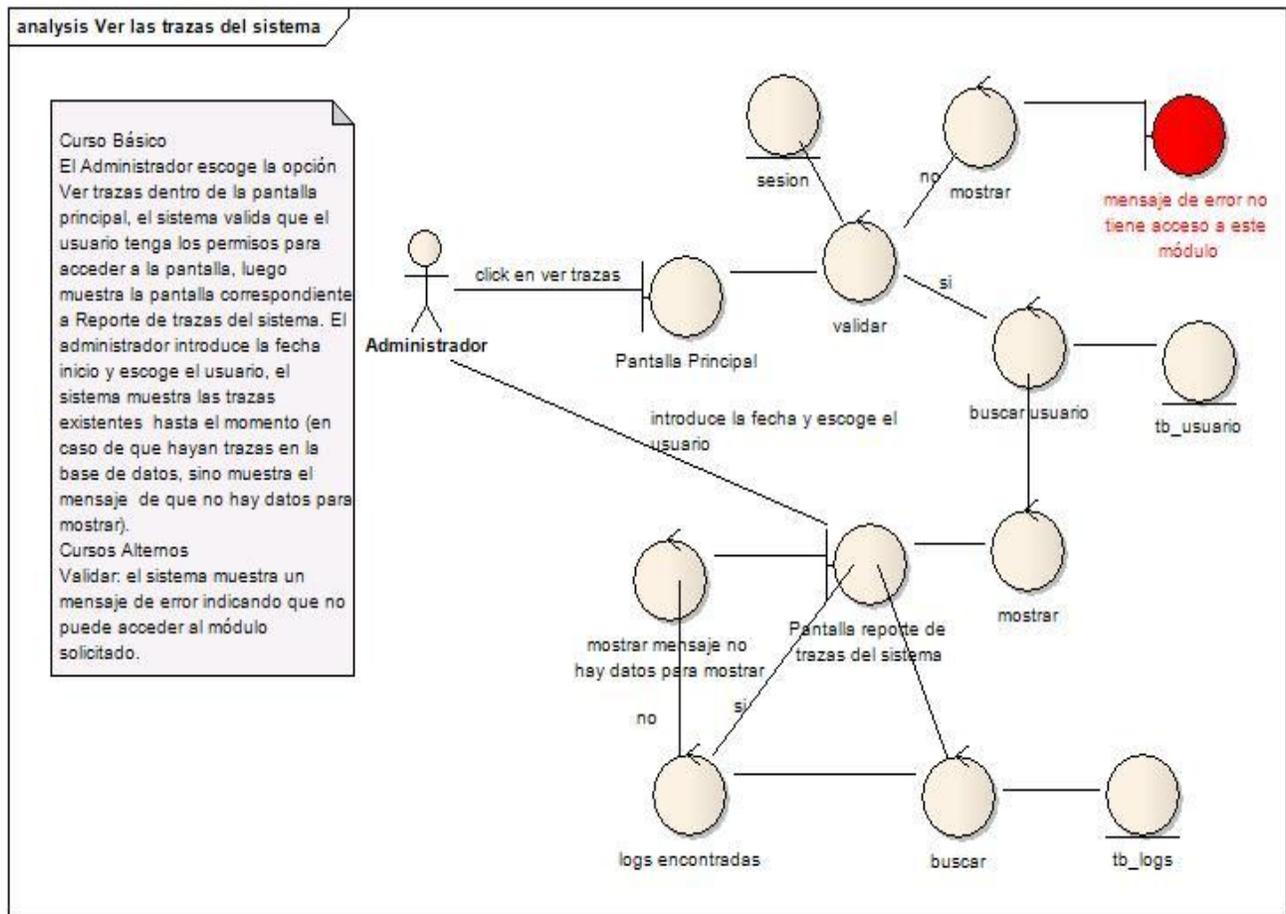


Figura 15. Diagrama de robustez del caso de uso ver trazas del sistema

## 2.4 Prototipación rápida

La Prototipación rápida se usa en Iconix para simular la interfaz final de la aplicación, para que el usuario tenga una idea clara del mismo y aporte su opinión y sugerencias. Los cambios al prototipo son planificados por los usuarios antes de la implementación y se espera que estos lo evalúen como si fuera el diseño final de la aplicación [35].

A continuación se muestran algunas de las interfaces del sistema.

La figura 16 muestra la interfaz autenticación la cual permite que una vez que el usuario introduzca su nombre y su contraseña, este pueda acceder a todas las opciones que brinda el sistema.



Figura 16. Interfaz autenticación

Después de autenticado el usuario el sistema muestra la interfaz principal la cual le da la bienvenida al usuario mostrando las funcionalidades a las cuales tiene acceso.



Figura 17. Interfaz principal del sistema

En la figura 18 se representa la interfaz expediente de ilegalidad de un edificio multifamiliar la cual muestra como el usuario registra las ilegalidades del inmueble con su descripción en el sistema.

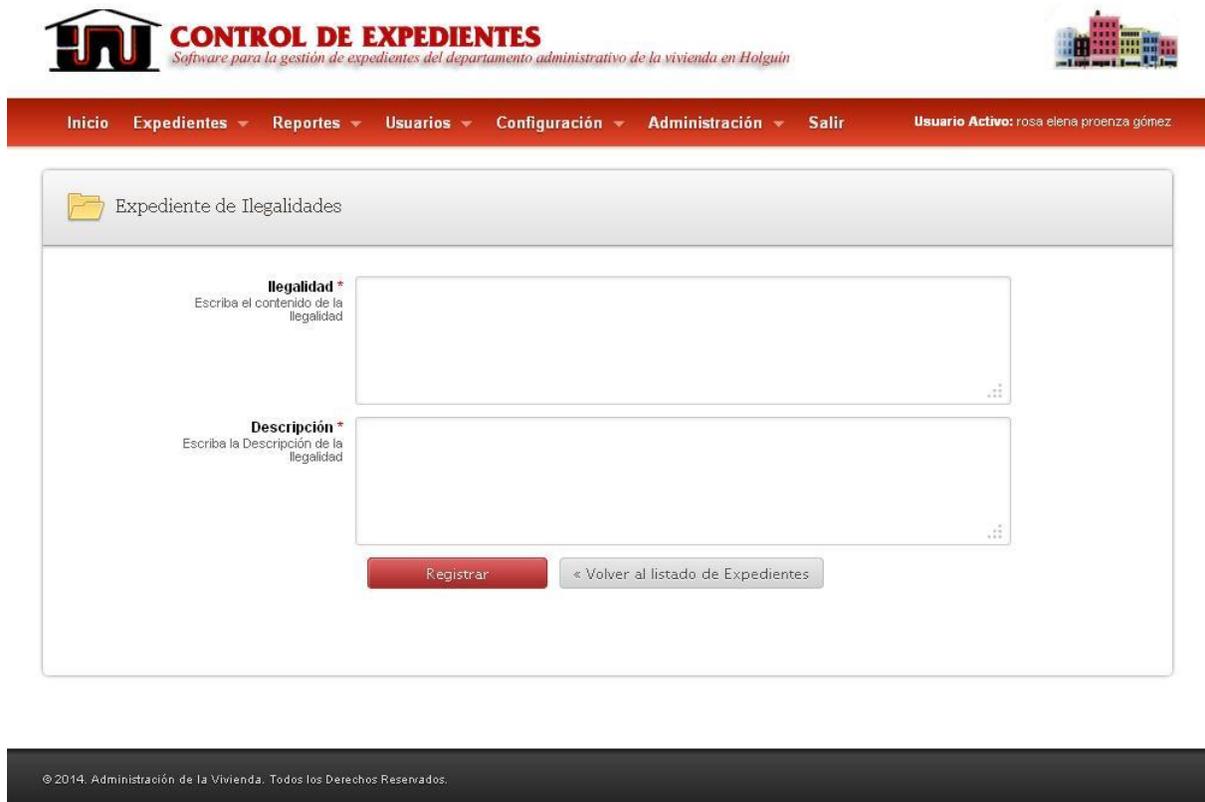


Figura 18. Interfaz expediente de ilegalidad

## 2.5 Modelo de despliegue

El modelo de despliegue (Ver figura 19) es un tipo de diagrama UML donde se describe la organización física del sistema en términos de cómo se distribuye la funcionalidad entre los nodos de cómputo; es decir, representa la correspondencia entre la arquitectura *software* y la arquitectura *hardware* mostrando las principales capas que conforman la aplicación [34].

La arquitectura de la propuesta está dispuesta de la siguiente manera:

Para desplegar el sistema se utiliza el servidor *Web* donde se aloja la aplicación y un servidor de base de datos para almacenar, consultar y actualizar la información por los usuarios. Para acceder al sistema los clientes se conectan mediante un navegador *Web*.

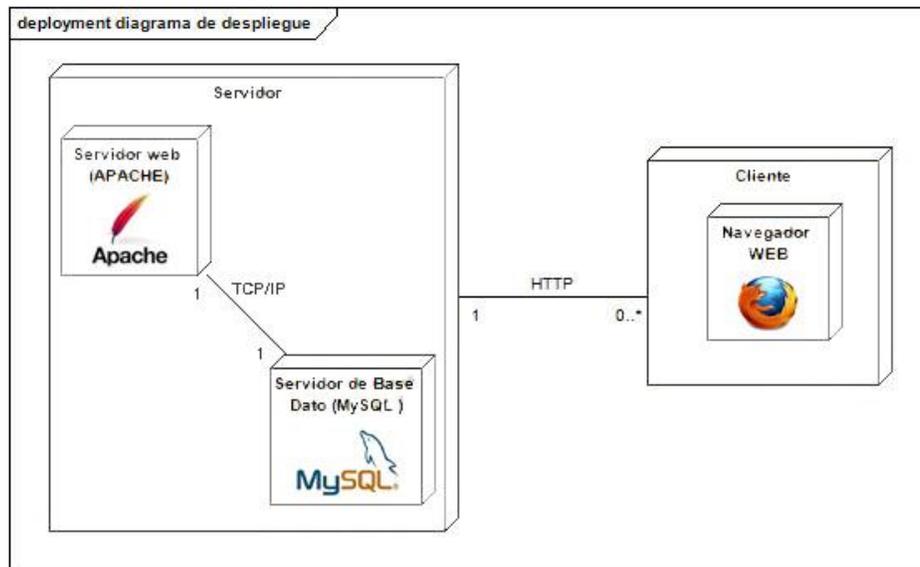


Figura 19. Modelo de Despliegue

## 2.6 Diseño detallado

Este proceso se basa en los productos del análisis para generar una especificación destinada a la solución lógica del problema. Su esencia es la elaboración de diagramas que muestran gráficamente como los objetos se comunican entre ellos a fin de cumplir los requerimientos. En el diseño de la aplicación se crean los diagramas de secuencia donde se describe el texto de los casos de uso [37].

### 2.6.1 Diagrama de secuencia

En la metodología de desarrollo de software Iconix los diagramas de secuencia abarcan el curso básico y todos los cursos alternos de cada caso de uso. El diagrama muestra la forma en que un grupo de objetos se comunican entre sí a lo largo del tiempo, este consta de las clases entidad, la interfaz y los actores que se han identificado en el diagrama de robustez. Las clases control no se ilustran en el diagrama de secuencia, estas se transforman en los mensajes entre las clases interfaz y entidad [38].

A continuación se presentan en la Figura 20, 21 y 22 los diagramas de secuencia correspondientes a los casos de uso crear expediente de cuartería, modificar organismo y ver trazas del sistema. Para ver todos los diagramas de secuencia asociados a los casos de usos restantes, se encuentran una parte en los anexos (Ver Anexo 3) y la totalidad de estos se encuentran en el archivo que se entregará adjunto a este documento.

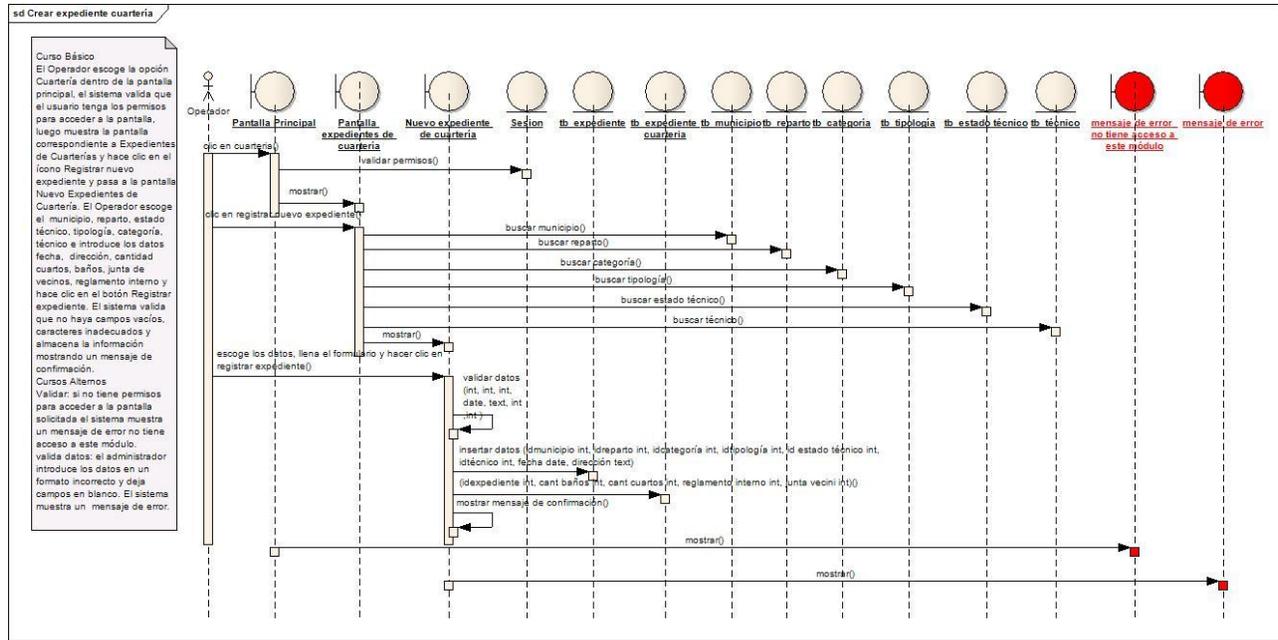


Figura 20. Diagrama secuencia del caso de uso crear expediente de cuartería

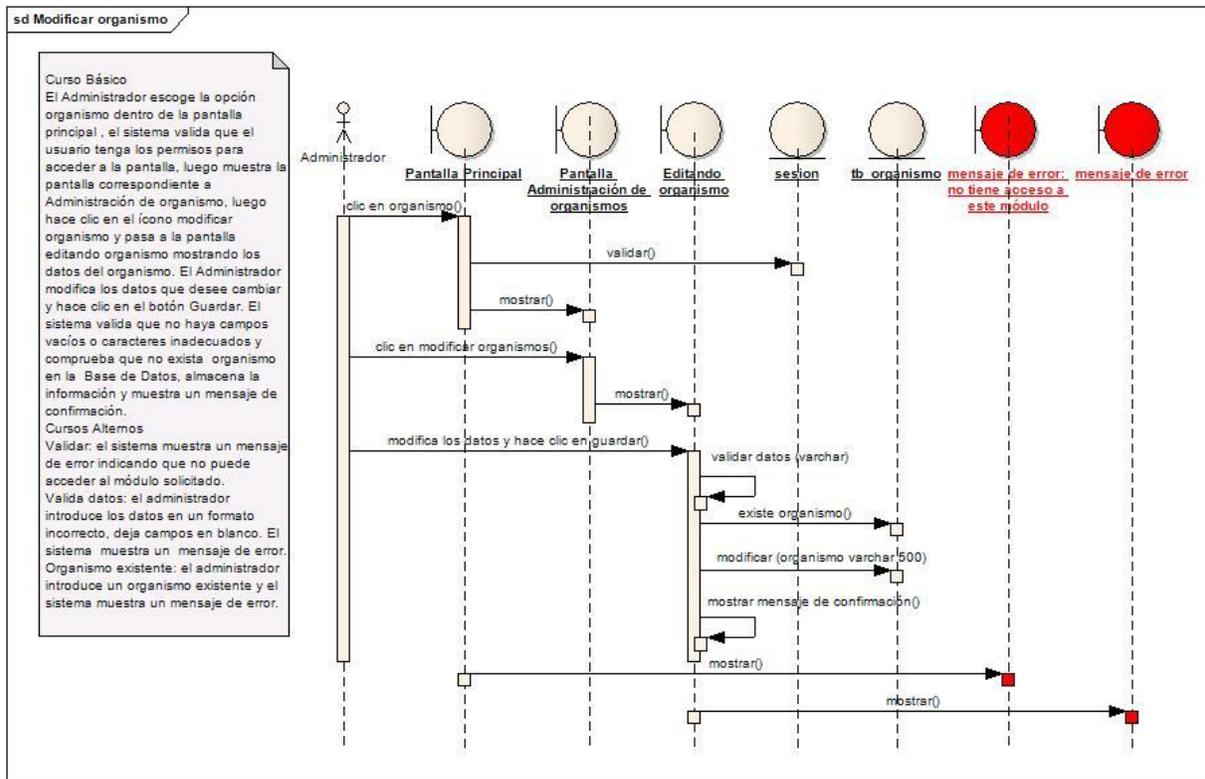


Figura 21. Diagrama secuencia del caso de uso modificar organismo

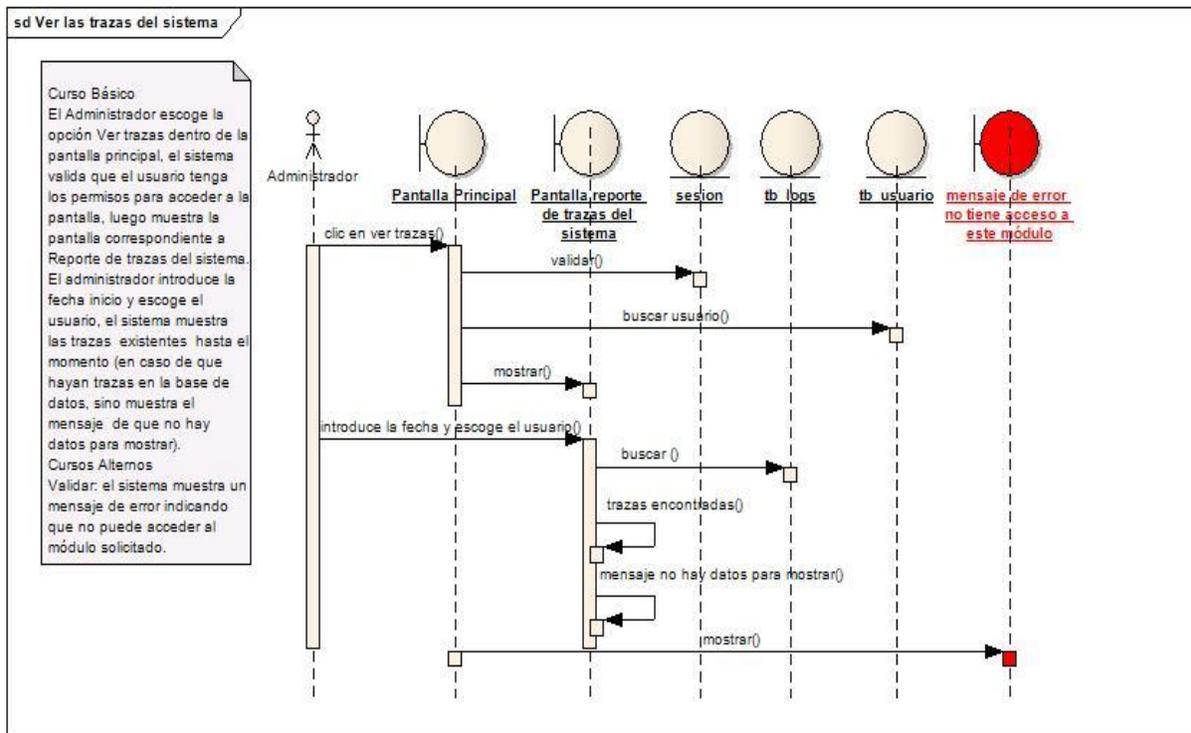


Figura 22. Diagrama de secuencia del caso de uso ver trazas del sistema

### 2.6.2 Diseño de la base de datos

El modelo de datos se ha convertido en el diagrama común para los desarrolladores de aplicaciones y desarrolladores de bases de datos. A partir del diagrama de clases como modelo lógico de la base de datos (Ver Anexo 4) se puede obtener el modelo de datos; en este modelo se define las estructuras de datos que se utilizan en el sistema y describe la estructura lógica de la información que queda almacenada en la base de datos [39]. El propio flujo de trabajo del análisis y el diseño permitió definir el modelo lógico de datos y a partir de este confeccionar el modelo de datos (Ver Anexo 5).

### 2.7 Implementación

Después de atravesar por las fases de diseño se está en condiciones para comenzar con la implementación del sistema y así lograr la obtención del producto informático propuesto. En la implementación se define la organización del código y se implementan las clases que se necesitan a lo largo del diseño [40]. Una de las ventajas que influyó en la reducción del tiempo de implementación fue la estrategia de reutilización de código.

### **2.7.1 Estándar de codificación**

En el desarrollo de la aplicación se establecieron estándares de código para facilitarle y agilizarle tanto a los desarrolladores de la aplicación como a los responsables del mantenimiento y actualización la lectura del código.

A continuación se muestran algunos de los estándares de código establecidos:

#### **Comentarios**

Los comentarios están escritos en español, siendo frases completas y comenzando con letra mayúscula.

#### **Variables**

Los nombres de las variables son escritos en minúsculas y las palabras se separan con guiones bajos (\_).

#### **Definición de clases y funciones**

En la definición de las clases y las funciones no se hace uso de abreviaturas, siendo lo más descriptivo posible y comenzando el nombre siempre con mayúscula. En el caso de que los nombres sean muy largos son separados por guiones bajos.

#### **Estructuras de control**

En las funciones se dejaron espacios en blanco entre la palabra y el paréntesis de apertura.

### **2.8 Estudio de factibilidad**

La factibilidad de un proyecto una vez obtenida su estimación está dada por la determinación de la posibilidad de hacer, según restricciones de tiempo y presupuesto, que son identificadas y aprobadas teniendo en cuenta los criterios organizativos, económicos, técnicos y de tiempo; siendo esta la valoración de disponibilidad de los recursos necesarios para llevar a cabo los objetivos o metas señalados [41].

Se hace necesario realizar un estudio de factibilidad que permita recopilar los datos fundamentales sobre el desarrollo del producto informático. Para determinar si es posible el desarrollo y la implementación del sistema propuesto.

### **Factibilidad técnica**

Como el proyecto es desarrollado por una estudiante para optar por el título de Ingeniería Informática, la Dirección Provincial de la Vivienda de Holguín, facilita todos los recursos materiales y humanos necesarios para su correcta terminación, o sea el problema se puede resolver con los medios que actualmente existen. Como resultado de este estudio se determina que en la actualidad, se cuenta con la infraestructura tecnológica para el desarrollo del sistema propuesto.

### **Factibilidad operacional**

El sistema está diseñado para que los usuarios finales una vez implantado el sistema, acepten la forma en que el mismo concibe el flujo de información en el control de los expedientes de administración de la vivienda, de manera que se reduzcan las posibilidades de resistencia al cambio. El sistema propuesto no perjudicará el trabajo que se realiza en el departamento de administración de la vivienda de la dirección provincial de Holguín, más bien lo mejora a partir de la informatización del proceso. En resumen el sistema puede ser explotado sin ningún problema de tipo operacional.

### **Factibilidad económica**

Si el sistema no es factible económicamente para la organización que lo desarrolla, entonces no puede ser elaborado y mucho menos implantado [42]. Para la determinación del costo asociado al desarrollo del proyecto, se utiliza el método de estimación por Puntos de Caso de Uso; empleado para estimar el esfuerzo requerido en el desarrollo de los primeros casos de uso de un producto informático. Los Puntos por Caso de Uso toman en cuenta la complejidad del sistema, aspectos ambientales y características técnicas como rendimiento y seguridad determinando el esfuerzo de desarrollo de un sistema así como su costo [43].

El primer paso para la estimación consiste en el cálculo de los Puntos de Caso de Uso sin ajustar (**UUCP**). Este valor se calcula a partir del factor de peso de los actores sin ajustar (**UAW**) y el factor de peso de los casos de uso sin ajustar (**UUCW**) como se muestra en la siguiente ecuación:

$$\mathbf{UUCP = UAW + UUCW}$$

El factor de peso de los actores sin ajustar y el factor de peso de los casos de uso sin ajustar se calculan mediante un análisis de la cantidad de actores y casos de uso presentes en el sistema y la complejidad de cada uno de ellos respectivamente. Las tablas que contienen esta información pueden ser consultadas en el Anexo 6.

Una vez que se tienen los Puntos de Casos de Uso sin ajustar, se debe calcular los Puntos de Casos de Uso ajustados (**UCP**) teniendo en cuenta los Puntos de Casos de Uso sin ajustar (**UUCP**), el Factor de complejidad técnica (**TCF**) y el Factor de complejidad ambiental (**ECF**) como se muestra en la siguiente ecuación:

$$\text{UCP} = \text{UUCP} \times \text{TCF} \times \text{ECF}$$

#### Factor de complejidad técnica (TCF)

Este coeficiente se calcula mediante la cuantificación de un conjunto de factores que determinan la complejidad técnica del sistema. Cada uno de los factores se cuantifica con un valor de 0 a 5 de acuerdo a la opinión del equipo de desarrollo como se muestra en la siguiente tabla.

Factor	Descripción	Peso	Valor	TCF
TCF01	Sistema Distribuido	2,00	0,00	0,00
TCF02	Objetivos de performance o tiempo de respuesta.	1,00	1,00	1,00
TCF03	Eficiencia del usuario final	1,00	2,00	2,00
TCF04	Procesamiento interno complejo	1,00	1,00	1,00
TCF05	El código debe ser reutilizable	1,00	4,00	4,00
TCF06	Facilidad de instalación	0,50	1,00	0,50
TCF07	Facilidad de uso	0,50	4,00	2,00
TCF08	Portabilidad	2,00	0,00	0,00
TCF09	Facilidad de cambio	1,00	3,00	3,00
TCF10	Concurrencia	1,00	0,00	0,00
TCF11	Incluye objetivos especiales de seguridad	1,00	3,00	3,00
TCF12	Provee acceso directo a terceras partes	1,00	5,00	5,00
TCF13	Se requieren facilidades especiales de entrenamiento a usuarios	1,00	1,00	1,00
			<b>Total:</b>	<b>22,50</b>

Tabla 6. Factores de complejidad técnica

El Factor de complejidad técnica se calcula de la siguiente manera:

Factor	Valor
Valor desajustado de TCF (UTV)	22,50
Pesos TCF (TWF)	0,01
Constante TCF (TC)	0,60
<b>TCF = TC + (UTV * TWF)</b>	<b>0,82</b>

Tabla 7. Cálculo del TCF

### Factor de complejidad ambiente (ECF)

Las habilidades y el entrenamiento del grupo involucrado en el desarrollo tienen un gran impacto en las estimaciones de tiempo. Estos factores son los que se contemplan en el cálculo del Factor de ambiente. El cálculo del mismo es similar al cálculo del Factor de complejidad técnica, es decir, se trata de un conjunto de factores que se cuantifican con valores de 0 a 5 como se muestra en la tabla 8.

Factor	Descripción	Peso	Valor	TCF
ECF01	Familiaridad con el modelo de proyecto utilizado	1,50	4,00	6,00
ECF02	Experiencia en la aplicación	0,50	4,00	2,00
ECF03	Experiencia en orientación a objetos	1,00	4,00	4,00
ECF04	Capacidad del analista líder	0,50	4,00	2,00
ECF05	Motivación	1,00	5,00	5,00
ECF06	Estabilidad de los requerimientos	2,00	2,00	4,00
ECF07	Personal part-time	-1,00	0,00	-0,00
ECF08	Dificultad del lenguaje de programación	-1,00	1,00	-1,00
			<b>Total:</b>	<b>22,00</b>

Tabla 8. Factor ambiental

El Factor de complejidad ambiental resulta de la siguiente manera:

Factor	Valor
Valor desajustado ECF (UEV)	22,00
Peso ECF (EWF)	-0,03
Constante ECF (EC)	1,40
<b>ECF = EC + (UEV * EWF)</b>	<b>0,74</b>

Tabla 9. Cálculo del ECF

Para determinar la cantidad de hombres por caso de uso se contabilizan cuántos factores de los que afectan al **ECF** están por debajo del valor medio (3), para los factores **ECF1** a **ECF6**.

Luego se contabilizan cuántos factores de los que afectan al **ECF** están por encima del valor medio (3), para los factores **ECF7** y **ECF8**. Si el total es 2 o menos, se utiliza el factor de conversión 20 horas-hombre/Punto de Casos de Uso, es decir, un Punto de Caso de Uso toma 20 horas-hombre [43]. El resultado obtenido fue de 1 por lo que se utilizó el factor de conversión 20 horas-hombre/Puntos de Casos de Uso.

Una vez calculado todos los factores que se tienen en cuenta para el Análisis de Puntos de Casos de Uso se obtuvieron los resultados siguientes:

<b>Pasos</b>	<b>Valor</b>
Fecha de Estimación	<b>13-abr-2014 14:57:59</b>
Total de Casos de Uso	<b>85</b>
UUCP	<b>535,00</b>
TCF	<b>0,82</b>
ECF	<b>0,74</b>
UCP = (UUCP * TCF * ECF)	<b>326,00</b>
CF	<b>20,00</b>
Esfuerzo en hora-hombres (E = CF * UCP)	<b>6520,00</b>
Costo Total	<b>13040,00</b>

Tabla 10. Estimación del esfuerzo y costo

Analizando los resultados mostrados en la tabla 10 se puede concluir que se emplearan 20 horas-hombres para realizar un caso de uso, obteniéndose un esfuerzo de 6520 horas-hombres, el cálculo real es de 1304 horas-hombres, disminuye en un 20% por el uso de herramientas utilizadas en la programación de la aplicación, para un costo total de 13040 pesos.

Luego de haber analizado si es posible el desarrollo y la implementación del sistema propuesto, se puede decir que se cuenta con los recursos necesarios para llevarlo a cabo y se dispone de la infraestructura tecnológica. Este podría ser explotado sin ningún problema de tipo técnico y operacional. En la factibilidad económica se obtuvo como resultado un esfuerzo de 1304 horas y un costo total de \$13040 valor ahorrado a la institución. En sentido general, en cuanto al estudio de factibilidad se puede concluir que el sistema es factible.

## **2.9 Valoración de sostenibilidad**

El análisis de sostenibilidad del producto informático se debe tener en cuenta antes de su implementación para lograr que su utilización proporcione beneficios y no dificultades. La

valoración de sostenibilidad favorece la autorregulación del proceso de diseño, con la satisfacción de la necesidad que resuelve, con un uso racional de recursos y la toma de decisiones adecuadas a las condiciones del contexto y el cliente [44].

En esta investigación se realiza la valoración a través de las cuatro dimensiones de gestión de sostenibilidad: administrativa, socio-humanista, ambiental y tecnológica.

### **Dimensión administrativa**

Desde el punto de vista administrativo el producto informático no incurre en gastos de compras de nuevas máquinas ya que la tecnología utilizada para su realización es adaptable a las características del *hardware* y la infraestructura de las redes de la entidad, además su confección se realizará en función del equipamiento técnico con que se cuenta; por lo que el consumo de electricidad asociado al funcionamiento del producto informático, no debe incrementar el ya existente. Para la determinación de los costos se utilizó la Estimación por puntos de caso de uso, cabe destacar que el diseño de este sistema informático no es con fines comerciales, tanto el diseño como la posterior elaboración del producto resultante no originarán gastos notables, puesto que se ha diseñado y será implementado como parte del trabajo de diploma.

El producto informático generará ingresos indirectos a la entidad, disminuye el derroche en materiales de oficina, principalmente papel y tinta; mejora el servicio que se presta, debido a que la puesta en funcionamiento de la aplicación está orientada a mejorar la gestión de los expedientes de la vivienda en el departamento administrativo lográndose una mejor toma de decisiones por parte de la entidad.

Las herramientas que se usaron para el desarrollo de la aplicación *Web* son libres, de modo que no es necesario pagar licencia por su uso, por lo que no se incurren en gastos monetarios referentes a la utilización de estos. Resultando en la dimensión administrativa la realización del sistema sostenible.

### **Dimensión socio-humanista**

Con respecto al impacto socio-humanista del producto informático, se puede asegurar que las condiciones de vida y en específico de trabajo se ven mejoradas: minimización del tiempo de búsqueda y del tiempo para realizar los procedimientos, confiabilidad e integridad a su

gestión de información. Este producto informático no generará empleo ni desempleo, su presencia está orientada a favorecer el trabajo de las personas que ya son empleadas por la entidad.

Para alcanzar una buena aceptación del producto informático y mitigar el rechazo al cambio, se debe capacitar al personal involucrado, además se incluye en la entrega de la aplicación *Web* el manual de usuario que apoya la utilización del sistema por el usuario. En la dimensión socio-humanista la realización del sistema es considerada sostenible.

### **Dimensión ambiental**

En la valoración de la sostenibilidad del producto informático en la dimensión ambiental se tuvieron en cuenta aspectos como: las condiciones favorables o no a las personas o la entidad y si se producen daños e impactos al medio ambiente.

El producto informático disminuye el uso de materiales de oficina, contribuye al entorno donde opera la entidad. No se contribuye al deterioro del medio ambiente por causa de contaminación por ruido, interferencia, todo lo contrario, ayudará al cuidado del mismo.

Se utilizan colores agradables a la vista manteniendo un aspecto atractivo, haciendo uso del blanco y el gris con utilización del rojo. Posee una interfaz amigable, el tipo y tamaño de texto son apropiados. Todo esto permite lograr una mayor identificación del usuario con la aplicación y evita estados de estrés de los usuarios del sistema.

El código a generar para el desarrollo del producto informático propuesto podrá ser reutilizado con vista a la implementación de otros módulos o nuevos requerimientos del sistema.

Los efectos negativos en la salud de los usuarios que puede ocasionar el sistema son: desgaste en la visión, deformación en la columna entre otras, propio del trabajo con computadores. Por lo que se recomienda una correcta postura de los usuarios en las sillas, así como tener el monitor a la altura de los ojos y utilizar protector de pantalla. Por todo lo antes expuesto en la dimensión ambiental la realización del sistema es considerada sostenible.

### **Dimensión tecnológica**

En la dimensión tecnológica se miden aspectos como el uso de la tecnología adecuada y asimilable por el usuario. Se cuenta con los servidores de red y base de datos para el correcto funcionamiento del sistema.

La capacitación de los usuarios que interactúan con el sistema es de vital importancia para la aceptación final de producto y manejo del mismo, además se adjuntará el manual de usuario junto a la aplicación *Web*.

El producto brinda facilidades para la correcta explotación por parte del usuario, con independencia del productor, pues el usuario tiene los privilegios para actualizar la información según el rol que tenga en el sistema. La aplicación puede verse afectada por la rotura del servidor de base de datos, para evitar la pérdida de la información este cuenta con la posibilidad de realizar salvallas periódicamente. Por todo lo antes expuesto en la dimensión tecnológica la realización del sistema es considerada sostenible.

### **Conclusiones de la valoración de sostenibilidad**

Al analizar el producto informático se llega a la conclusión que el mismo es sostenible desde las dimensiones administrativas, socio-humanista, ambiental y tecnológica, debido a que da solución a la necesidad social que lo originó. El producto informático es una herramienta fácil de usar y se utilizaron escasos recursos para su elaboración, ya que la entidad cuenta con los medios necesarios. Garantiza flexibilidad ante los cambios que puedan ocurrir y puede ser utilizado bajo cualquier sistema operativo que posea un navegador.

### **2.10 Valoración de la Propuesta de Solución**

Una vez concluido el sistema, se procedió a la elaboración de encuestas, a través del criterio de expertos, con el objetivo de evaluar la satisfacción de los usuarios finales y obtener valoraciones conclusivas del sistema.

Para evaluar los resultados de la aplicación *Web* Sistema para la gestión de los expedientes en el departamento administrativo de la Vivienda de la Dirección Provincial de Holguín el primer paso fue seleccionar a los expertos, para ello se realizó una encuesta (Ver Anexo 7) y a partir del procesamiento de la misma se obtuvo el coeficiente de competencia de cada uno (Ver Anexo 8), seleccionándose de 18 posibles expertos sólo los de alta competencia para

un total de 14 los cuales poseen experiencia en el tema y conocimientos informáticos. Seleccionados los expertos, se sometieron éstos a encuestas de evaluación del sistema (Ver Anexo 9), de ellos el 64% son licenciados (9), 14 % son máster (2), y el 21% son de duodécimo grado (3). Tras realizar las encuestas a los expertos se realizó el procesamiento estadístico de la información obtenida en las mismas (Ver Anexo 10). Este proceso interactivo permite a los expertos modificar sus respuestas primarias en función de los juicios aportados por otros expertos.

Los aspectos de evaluación del sistema se muestran a continuación:

1. ¿Esta aplicación *Web* favorece el control de los expedientes de administración de la vivienda en la Dirección Provincial de Holguín y agiliza este proceso?
2. ¿Cómo aprecia el diseño de las interfaces del sistema?
3. ¿Qué le parece el uso de los colores y las imágenes usadas en el sistema?
4. ¿El sistema abarca la totalidad del proceso?
5. ¿Cómo considera usted el acceso a las funcionalidades del sistema?
6. ¿Cómo evalúa el tiempo de respuesta del sistema?
7. ¿Cómo valora el uso que se le puede dar al sistema?

A partir del procesamiento estadístico de los aspectos anteriormente mencionados, se determinó, como resultado final de la evaluación, que los encuestados coinciden en que todos los aspectos son “Muy Relevantes” (Ver Anexo 11), lo cual influye positivamente en la calidad del sistema. Se puede concluir con los resultados obtenidos que el grado de satisfacción de los encuestados es alto y que el sistema cumple con las funcionalidades para las que fue creado.

### **2.11 Conclusiones del capítulo**

La aplicación de las etapas de la metodología Iconix permitió la posibilidad de dar cumplimiento a los requerimientos del sistema para el diseño e implementación del producto Informático.

- ✓ Se realizó un estudio de factibilidad utilizando el método de estimación por puntos de caso de uso que permitió determinar el esfuerzo y costo del producto informático.

- ✓ El procedimiento para la valoración de sostenibilidad aplicado al producto informático demuestra que es sostenible con respecto a las dimensiones administrativa, socio-humanista, ambiental y tecnológica.
- ✓ Las encuestas realizadas mostraron un alto nivel de satisfacción de los expertos con el sistema cumpliendo con las expectativas planteadas en la investigación.

## CONCLUSIONES

Al término de esta investigación se ha arribado a las siguientes conclusiones:

- ✓ El estudio realizado en la DPV permitió detectar las deficiencias del proceso de gestión de los expedientes de administración de la vivienda y posteriormente favorecer a partir de una aplicación *Web* la gestión y control de los mismos, dando cumplimiento al objetivo planteado en esta investigación.
- ✓ Las tecnologías asumidas para la creación del producto informático y la metodología de desarrollo de *software* ICONIX utilizada para el análisis, diseño y desarrollo de la aplicación resultaron apropiadas, quedando disponible su utilización en sistemas similares.
- ✓ El análisis de sostenibilidad desde las dimensiones administrativa, socio–humanista, ambiental y tecnológica permitió determinar que el sistema es sostenible y factible en cuanto a esfuerzo y costo.
- ✓ Se valoró el producto informático a partir del criterio de expertos, dando como resultado que el sistema desarrollado posee un alto nivel de aceptación entre los encuestados que califican al sistema como “Muy Relevante”, comprobando de esta forma que el producto informático cumple con las expectativas del cliente.

## RECOMENDACIONES

Concluida la investigación se proponen las siguientes recomendaciones:

1. Implantar el sistema en la unidad presupuestada Dirección Provincial de la Vivienda de Holguín.
2. Continuar el desarrollo del producto incorporando la creación de nuevos módulos y así extender el sistema a otras Áreas de la Entidad.
3. Generalizar el uso de la aplicación a las entidades provinciales de la vivienda del país que posean dificultades similares en el proceso de gestión de los expedientes de administración de la vivienda.

## GLOSARIO DE TÉRMINOS

**Apache:** Herramienta *software* libre para servir aplicaciones *Web* en PHP y MySQL.

**ASP:** Tecnología del lado servidor de *Microsoft* para páginas *Web* generadas. La implantación de ASP está limitada para arquitecturas basadas en tecnología *Microsoft*. Sólo tiene soporte nativo para los servidores IIS(Internet Information Server) y Personal *Web Server*, que son los dos servidores *Web* para sistemas *Microsoft*, el primero con tecnología NT y el segundo para sistemas *Windows 98* y similares. Utiliza diversos componentes ya desarrollados como algunos controles ActiveX.

**CGI:** *Common Gateway Interface* o Interfaz de entrada común, es una tecnología de la *World Wide Web* (*www*, por sus siglas en inglés) que permite a un cliente (navegador web) solicitar datos de un programa ejecutado en un servidor *Web*.

**GPL:** *General Public License* o Licencia Pública General de GNU, es la licencia más ampliamente usada en el mundo del *software* y garantiza a los usuarios finales la libertad de usar, estudiar, compartir y modificar el *software*. Su propósito es declarar que el *software* cubierto por esta licencia es *software* libre.

**GNU:** Proyecto que reside en la Fundación del *Software* Libre para la organización, control y difusión del *software* libre.

**HTML:** (*HyperText Markup Language*). Lenguaje de marcas hipertextuales, lenguaje de marcación diseñado para estructurar textos y presentarlos en forma de hipertexto, que es el formato estándar de las páginas *Web*.

**HTTP:** *Hypertext Transport Protocol*: Conjunto de especificaciones para el intercambio de ficheros (texto, gráfico, imagen, sonido, vídeo) en la *Web*.

**Internet Information Server (IIS):** Es un servidor *Web* y un conjunto de servicios para el sistema operativo *Microsoft Windows*. Los servicios de IIS proporcionan las herramientas y funciones necesarias para administrar de forma sencilla un servidor *Web* seguro.

**JSP:** *Java Server Pages* es una tecnología Java que permite a los programadores generar dinámicamente HTML, o algún otro tipo de página *Web* y permite al código Java y a algunas acciones predefinidas ser embebidas en el contenido estático.

**Java:** Es un lenguaje de programación orientado a objetos. El lenguaje en sí mismo toma mucha de su sintaxis de C y C++, pero tiene un modelo de objetos más simple y elimina herramientas de bajo nivel, que suelen inducir a muchos errores, como la manipulación directa de punteros o memoria.

**JavaScript:** Es un lenguaje de programación interpretado. Se utiliza principalmente en su forma del lado del cliente, implementado como parte de un navegador *Web* permitiendo mejoras en la interfaz de usuario y páginas *Web* dinámicas.

**MVC:** Modelo Vista Controlador. Es un patrón de arquitectura de *software* que separa los datos de una aplicación, la interfaz de usuario y la lógica de control en tres componentes distintos. El patrón MVC se ve frecuentemente en aplicaciones web, donde la vista es la página HTML y el código que provee de datos dinámicos a la página, el modelo es el Sistema de Gestión de Base de Datos y el controlador representa la Lógica de negocio.

**MS SQL Server:** Es un sistema para la gestión de bases de datos producido por *Microsoft* basado en el modelo relacional. Este constituye la alternativa de *Microsoft* a otros potentes sistemas gestores de bases de datos como son Oracle, PostgreSQL o MySQL.

**Sun Application Server:** (Sun Java System Application Server) es un servidor de aplicaciones de la plataforma Java producido por Sun Microsystems. El servidor está basado en la plataforma Java EE y es el núcleo del sistema Java Enterprise.

**Software:** Programas de sistema, utilerías o aplicaciones expresados en un lenguaje de máquina.

**SQL:** *Structured Query Language*. Lenguaje de preguntas estructurado, lenguaje que utiliza bases de datos para pedir información de las mismas.

**SQLite:** Gestor de base de datos.

**Open Source:** Código abierto. Manera de nombrar también a las aplicaciones desarrolladas bajo el amparo de un producto con licencia GPL.

**Python:** Es un lenguaje de programación de alto nivel cuya filosofía hace hincapié en una sintaxis muy limpia y que favorezca un código legible. Se trata de un lenguaje de programación que soporta orientación a objetos, programación imperativa y en menor medida, programación funcional. Es un lenguaje interpretado y multiplataforma.

**Perl:** Es un lenguaje de programación diseñado por Larry Wall. Perl toma características del lenguaje C, estructuralmente, está basado en un estilo de bloques como los del C y fue ampliamente adoptado por su destreza en el proceso de texto y no tener ninguna de las limitaciones de los otros lenguajes de script.

**Ruby:** Es un lenguaje de programación interpretado, reflexivo y orientado a objetos. Combina una sintaxis inspirada en Python y Perl con características de programación orientada a objetos. Comparte también funcionalidad con otros lenguajes de programación y su implementación oficial es distribuida bajo una licencia de *software* libre.

**UML:** *Unified Modeling Language*. Es una notación estándar para modelar objetos del mundo real como primer paso en el desarrollo de programas orientados a objetos.

**WEB (WWW):** Red de documentos HTML intercomunicados y distribuidos entre servidores del mundo entero.

**XHTML:** (*eXtensible HyperText Markup Language*). XHTML es básicamente HTML expresado como XML válido. Es más estricto a nivel técnico, pero esto permite que posteriormente sea más fácil al hacer cambios o buscar errores entre otros.

**XML:** (*eXtensible Markup Language*), es un metalenguaje extensible, no es realmente un lenguaje en particular, sino una manera de definir lenguajes para diferentes necesidades, de ahí que se le denomine metalenguaje. Este no ha nacido sólo para su aplicación en Internet, sino que se propone como un estándar para el intercambio de información estructurada entre diferentes plataformas. Se puede usar en bases de datos, editores de texto, hojas de cálculo y casi cualquier cosa imaginable.

## REFERENCIAS BIBLIOGRÁFICAS

- [1] Compilación de normas legales vigentes de la vivienda, 2006.
- [2] Diseño Estratégico, D. P. d. I. Vivienda, 2013.
- [3] Resolución No 4-91 Resolución general de edificios multifamiliares, 1988.
- [4] Resolución de viviendas vinculadas y medios básicos 58-87, 1989.
- [5] Resolución no 71-58 Cese, 2008.
- [6] MANUAL DE PROCEDIMIENTO DE LA SUB DIRECCION PROVINCIAL DE ADMINISTRACION DE LA VIVIENDA, 2009.
- [7] Expediente de Auditoria especial al Subsistema de Caja y Banco de la empresa Unidad resupuestada Dirección Provincial de la Vivienda. , D. P. d. I. V. d. Holguín, 2013.
- [8] P. Renaud. (2008, Arquitectura Cliente / Servidor Available: <http://ccia.ei.uvigo.es/docencia/SCS>
- [9] D. H. a. E. J. Robert Orfali Cliente/Servidor: Guía de supervivencia. México, D.F. , 1998.
- [10] G. M. FLORENTINO, "APLICACIONES WEB DOCUMENTACION DE PROYECTO," 2010.
- [11] P. C. Morales. (1999, Desarrollo de Aplicaciones Distribuidas basadas en Tecnologías Web. Universidad de Vigo
- [12] D. E. Rivera, "Módulo de Configuración del Sistema de Gestión Empresarial Medioambiental (GEMA), para el Sistema Integrado de Gestión (SIG) de la Oficina Territorial de Normalización de Holguín Universidad Oscar Lucero Moya," Oscar Lucero Moya Holguín, 2011.
- [13] D. C. V. TUSA, "DESARROLLO DE UN SITIO WEB PARA GESTIÓN INFORMATIVA Y CONTROL DE HISTORIAS CLÍNICAS PARA LA FUNDACIÓN. ," Quito, 2013.
- [14] netcraft. (2014, Web Server Survey. Available: <http://news.netcraft.com/>
- [15] J. O. Castilla. (2011, Ingeniería Técnica en Informática de Gestión. Available: [http://ddd.uab.cat/pub/trerecpro/2012/hdl\\_2072\\_182715/OlidCastillaJonathanR-ETIGa2010-11.pdf](http://ddd.uab.cat/pub/trerecpro/2012/hdl_2072_182715/OlidCastillaJonathanR-ETIGa2010-11.pdf)



- [16] ApacheFoundation. (2005, Servidor Apache. Available: <http://www.apache.org>.
- [17] E. O. B. LÓPEZ, "SISTEMA PARA LA ADMINISTRACIÓN DE EQUIPOS DE CÓMPUTO," UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO, MÉXICO D.F., 2012.
- [18] C. V. Mariño. (2008). Programación en PHP5 Available: [http://administraciondesistemas.pbworks.com/f/Manual\\_PHP5\\_Basico.pdf](http://administraciondesistemas.pbworks.com/f/Manual_PHP5_Basico.pdf)
- [19] Vazques. (2010). Ventajas De Usar Php. Available: [http://codigoprogramacion.com/Ventajas\\_de\\_usar\\_PHP.htm](http://codigoprogramacion.com/Ventajas_de_usar_PHP.htm)
- [20] J. J. Gutiérrez. (2009). Framework.
- [21] C. S. Foundation. (2014). CakePHP Cookbook Documentation. Available: [http://book.cakephp.org/2.0/\\_downloads/es/CakePHPCookbook.pdf](http://book.cakephp.org/2.0/_downloads/es/CakePHPCookbook.pdf)
- [22] C. M. Delgad. (2013, COMPARATIVA FRAMEWORKS Available: <http://es.scribd.com/doc/189203494/Comparativa-Framework>.
- [23] Symfony. (2011, Why should I use a framework? Available: <http://symfony.com/why-use-a-framework>
- [24] R. Ellis. (2011). CodeIgniter Guía del Usuario en Español. Available: [http://regionmoquegua.gob.pe/inia/sites/default/files/field/archivos/CodeIgniter\\_Spanish\\_UserGuide.pdf](http://regionmoquegua.gob.pe/inia/sites/default/files/field/archivos/CodeIgniter_Spanish_UserGuide.pdf)
- [25] T. G. Pérez. (2010, Sistema Gestores de Base de Datos. Available: [http://www.csi-csif.es/andalucia/modules/mod\\_ense/revista/pdf/Numero\\_30/TERESAGARZON\\_1.pdf](http://www.csi-csif.es/andalucia/modules/mod_ense/revista/pdf/Numero_30/TERESAGARZON_1.pdf)
- [26] G. D. G. PostgreSQL. (2010). PostgreSQL . Documentation.
- [27] E. Torrescano. (2013, Administración de Base de datos. Available: <http://www.slideshare.net/Ing-D-SW-TorresKhano--ME>
- [28] E. X. B. CHICA, "IMPLEMENTACIÓN DEL SERVICIO WEB 'CALIFICACIÓN DE JUGADORES EN LÍNEA' PARA LA ASOCIACIÓN DE FÚTBOL AMATEUR DE PICHINCHA (AFAP) UTILIZANDO LA METODOLOGÍA ICONIX," Quito, 2014.
- [29] A. V. Méndez. (2010, METODOLOGÍAS DE DESARROLLO DE SOFTWARE. Available: <http://www.monografias.com/trabajos-pdf4/metodologias-de-desarrollo-software/metodologias-de-desarrollo-software.pdf>

- [30] A. A. F. Nicolalde, "“APLICACIÓN WEB PARA LA ADMINISTRACIÓN ONLINE DE CITAS MÉDICAS EN EL CENTRO MÉDICO DE ORIENTACIÓN Y PLANIFICACIÓN FAMILIAR CEMOPLAF-OTAVALO; UTILIZANDO EL PATRÓN DE ARQUITECTURA MVC EN PHP”," Ibarra – Ecuador, 2013.
- [31] C. J. S. Roberth G Figueroa, Armando A Cabrera "METODOLOGÍAS TRADICIONALES VS. METODOLOGÍAS ÁGILES," Universidad Técnica Particular de Loja, Escuela de Ciencias en Computación, 2010.
- [32] M. A. Chaves. (2005, La ingeniería de requerimientos y su importancia en el desarrollo de proyectos de software. VI. Available: <http://www.redalyc.org/articulo.oa?id=66661111>
- [33] A. Simon, Definición de organización, 3ra ed., 2005.
- [34] I. J. G. B. J. Rumbaugh. (2000). El proceso unificado de desarrollo de software (Andres Otero ed.).
- [35] C. R. P. d. S. M. Oliva. (2009, Uso de Metodología ICONIX. Available: - <http://www.portalhuarpe.com.ar/Seminario09/archivos/UsodeICONIX.pdf>
- [36] M. d. I. A. S. L. Juan M Fernandez. (2004). ICONIX. Available: <http://www.roaybe.com/materiales/iconix.pdf>
- [37] J. M. F. P. Dra Maria de los angeles Sumano. (2011). Ingenierúia II Manual de Prácticas. Available: [http://www.uv.mx/personal/asumano/files/2010/07/MP\\_IS2\\_2011.pdf](http://www.uv.mx/personal/asumano/files/2010/07/MP_IS2_2011.pdf)
- [38] D. Gutierrez. (2011). UML Diagramas de Secuencia.
- [39] R. Pressman, Ingeniería de Software. Un enfoque práctico., 5ta ed., 1997
- [40] D. Rosenberg, Agile Development with ICONIX Process—People, Process, and Pragmatism, 2005.
- [41] A. Jaramillo. (2010). Desarrollo de software -Estudio de factibilidad. . Available: <http://aldrinjaramillo.googlepages.com/EstudiodeFactibilidad.doc>
- [42] A. C. A. C, "“Estudio de Factibilidad y Mercadeo.” " 2009.
- [43] M. Peralta. Estimación del esfuerzo basado en casos de usos. . Available: <http://www.itba.edu.ar/capis/webcapis/planma.html>



- [44] F. E. R. María Rita Concepción García, Sergio Cleger Tamayo (2010, Educación para la sostenibilidad en docencia de ingeniería informática. Revista Iberoamericana de Educación vol. 59(2). Available: [http://www.rieoei.org/rie\\_contenedor.php?numero=4370&titulo=Educaci%C3%83%C2%B3n%20para%20la%20sostenibilidad%20en%20docencia%20de%20ingenier%C3%83%C2%ADa%20inform%C3%83%C2%A1tica](http://www.rieoei.org/rie_contenedor.php?numero=4370&titulo=Educaci%C3%83%C2%B3n%20para%20la%20sostenibilidad%20en%20docencia%20de%20ingenier%C3%83%C2%ADa%20inform%C3%83%C2%A1tica)

## ANEXOS

### Anexo 1: Descripción Textual de los Casos de Uso

<b>Caso de uso del sistema: Crear Usuario</b>	
<b>Curso Básico</b>	El Administrador escoge la opción usuarios dentro de la pantalla principal , el sistema valida que el usuario tenga los permisos para acceder a la pantalla, luego muestra la pantalla correspondiente a Administración de usuarios, luego hace clic en el ícono de Nuevo Usuario y pasa a la pantalla nuevo usuario. El Administrador introduce los datos usuario, nombre, contraseña y escoge el tipo de usuario y hace clic en el botón Guardar. El sistema valida que no haya campos vacíos o caracteres inadecuados y comprueba que no exista el usuario en la Base de Datos, almacena la información y muestra un mensaje de confirmación.
<b>Cursos Alternos</b>	<p>Validar: el sistema muestra un mensaje de error indicando que no puede acceder al módulo solicitado.</p> <p>Valida datos: el administrador introduce los datos en un formato incorrecto, deja campos en blanco. El sistema muestra un mensaje de error.</p> <p>Existe Usuario: el administrador introduce un usuario existente y el sistema muestra un mensaje de error.</p>

Descripción textual del caso de uso Crear usuario

<b>Caso de uso del sistema: Buscar Usuario</b>	
<b>Curso Básico</b>	El Operador escoge la opción usuario dentro de la pantalla principal, el sistema valida que el usuario tenga



---

	los permisos para acceder a la pantalla, luego muestra la pantalla correspondiente a administración de usuario. El operador introduce el criterio de búsqueda y el sistema muestra los usuarios que existen para esos parámetros hasta el momento.
<b>Cursos Alternos</b>	Validar: el sistema muestra un mensaje de error indicando que no puede acceder al módulo solicitado.

Descripción textual del caso de uso Buscar usuario

<b>Caso de uso del sistema: Modificar Expediente Cuartería</b>	
<b>Curso Básico</b>	El Operador escoge la opción Cuartería dentro de la pantalla principal, el sistema valida que el usuario tenga los permisos para acceder a la pantalla, luego muestra la pantalla correspondiente a Expedientes de Cuarterías, luego hace clic en el ícono Modificar expediente de cuartería y pasa a la pantalla editando expediente de cuartería mostrando los datos de la cuartería. El Operador modifica los datos que desee cambiar y hace clic en el botón Registrar expediente. El sistema valida que no haya campos vacíos o caracteres inadecuados, almacena la información y muestra un mensaje de confirmación.
<b>Cursos Alternos</b>	Validar: si no tiene permisos para acceder a la pantalla solicitada el sistema muestra un mensaje de error no tiene acceso a este módulo.  Valida datos: el administrador introduce los datos en un formato incorrecto y deja campos en blanco. El sistema muestra un mensaje de error.

Descripción textual del caso de uso Modificar expediente de cuartería

<b>Caso de uso del sistema: Eliminar Expediente Cuartería</b>	
<b>Curso Básico</b>	El Operador escoge la opción Cuartería dentro de la pantalla principal, el sistema valida que el usuario tenga los permisos para acceder a la pantalla, luego muestra la pantalla correspondiente a Expedientes de Cuartería, luego selecciona el expediente o los expedientes que desea eliminar y hace clic en el ícono Eliminar expediente mostrándose un mensaje de si desea realmente eliminar el expediente de Cuartería el Operador hace clic en el botón Aceptar, se elimina ese expediente de la base de datos y el sistema muestra un mensaje de confirmación.
<b>Cursos Alternos</b>	El Administrador hace clic en el botón Cancelar y no se realiza la acción.  Validar: el sistema muestra un mensaje de error indicando que no puede acceder al módulo solicitado.

Descripción textual del caso de uso Eliminar expediente de cuartería

<b>Caso de uso del sistema: Listar Expediente Cuartería</b>	
<b>Curso Básico</b>	El Operador escoge la opción cuartería dentro de la pantalla principal, el sistema valida que el usuario tenga los permisos para acceder a la pantalla y verifica si hay expedientes de cuarterías insertados en la Base de Datos mostrando una lista con todos los expedientes de cuarterías existentes en la pantalla correspondiente a expediente de cuartería (en caso de que hayan expedientes de cuartería en la base de datos, sino muestra el mensaje de que no hay datos disponibles).



<b>Cursos Alternos</b>	Validar: el sistema muestra un mensaje de error indicando que no puede acceder al módulo solicitado.
------------------------	--

Descripción textual del caso de uso Visualizar expediente de cuartería

<b>Caso de uso del sistema: Buscar Organismo</b>	
<b>Curso Básico</b>	El Operador escoge la opción organismo dentro de la pantalla principal, el sistema valida que el usuario tenga los permisos para acceder a la pantalla, luego muestra la pantalla correspondiente a administración de organismo. El operador introduce el criterio de búsqueda y el sistema muestra los organismos que existen para esos parámetros hasta el momento.
<b>Cursos Alternos</b>	Validar: el sistema muestra un mensaje de error indicando que no puede acceder al módulo solicitado.

Descripción textual del caso de uso Buscar organismo

<b>Caso de uso del sistema: Listar Expediente Vivienda Vincula</b>	
<b>Curso Básico</b>	El Operador escoge la opción vivienda vinculada dentro de la pantalla principal, el sistema valida que el usuario tenga los permisos para acceder a la pantalla y verifica si hay expediente de vivienda vinculada insertados en la Base de Datos mostrando una lista con todos los expedientes de vivienda vinculada existentes en la pantalla correspondiente a expediente de vivienda vinculada (en caso de que hayan expedientes de vivienda vinculada en la base de datos, sino muestra el mensaje de que no hay datos disponibles).
<b>Cursos Alternos</b>	Validar: el sistema muestra un mensaje de error indicando que no puede acceder al módulo solicitado.

Descripción textual del caso de uso Visualizar expediente de vivienda vinculada

<b>Caso de uso del sistema: Crear Expediente Vivienda Vincula</b>	
<b>Curso Básico</b>	<p>El Operador escoge la opción vivienda vinculada de la pantalla principal, el sistema valida que el usuario tenga los permisos para acceder a la pantalla, luego muestra la pantalla correspondiente a Expedientes de Viviendas vinculada, luego hace clic en el ícono Registrar nuevo expediente y pasa a la pantalla Nuevo Expedientes de Vivienda vinculada. El Operador escoge los datos municipio, reparto, estado técnico, tipología, categoría, organismo y organización e introduce la fecha, dirección, no resolución, año de construcción, descripción del inmueble, nombre del arrendatario, FICV, FCV y hace clic en el botón Registrar expediente. El sistema valida que no haya campos vacíos, o caracteres inadecuados y almacena la información mostrando un mensaje de confirmación.</p>
<b>Cursos Alternos</b>	<p>Validar: si no tiene permisos para acceder a la pantalla solicitada el sistema muestra un mensaje de error no tiene acceso a este módulo.</p> <p>Valida datos: el administrador introduce los datos en un formato incorrecto y deja campos en blanco. El sistema muestra un mensaje de error.</p>

Descripción textual del caso de uso Crear expediente de vivienda vinculada

<b>Caso de uso del sistema: Modificar Expediente Vivienda Vincula</b>	
<b>Curso Básico</b>	<p>El Operador escoge la opción vivienda vinculada de la pantalla principal, el sistema valida que el usuario tenga</p>



	<p>los permisos para acceder a la pantalla, luego muestra la pantalla correspondiente a Expedientes de Viviendas vinculada, luego hace clic en el ícono modificar expediente de vivienda vinculada y pasa a la pantalla editando expedientes de vivienda vinculada mostrando los datos correspondientes, el Operador modifica los datos y hace clic en el botón Registrar expediente. El sistema valida que no haya campos vacíos, o caracteres inadecuados y almacena la información mostrando un mensaje de confirmación.</p>
<b>Cursos Alternos</b>	<p>Validar: el sistema muestra un mensaje de error indicando que no puede acceder al módulo solicitado.</p> <p>Valida datos: el administrador introduce los datos en un formato incorrecto, deja campos en blanco. El sistema muestra un mensaje de error.</p>

Descripción textual del caso de uso Modificar expediente de vivienda vinculada

<b>Caso de uso del sistema: Eliminar Expediente Vivienda Vincula</b>	
<b>Curso Básico</b>	<p>El Operador escoge la opción vivienda vinculada dentro de la pantalla principal, el sistema valida que el usuario tenga los permisos para acceder a la pantalla, luego muestra la pantalla correspondiente a Expedientes de vivienda vinculada , luego selecciona el expediente o los expedientes que desea eliminar y hace clic en el ícono eliminar expediente mostrándose un mensaje de si desea realmente eliminar el expediente el Operador hace clic en el botón Aceptar, se elimina ese expediente de la base de datos y el sistema muestra un mensaje de confirmación.</p>



<b>Cursos Alternos</b>	<p>El Administrador hace clic en el botón Cancelar y no se realiza la acción.</p> <p>Validar: el sistema muestra un mensaje de error indicando que no puede acceder al módulo solicitado.</p>
------------------------	---

Descripción textual del caso de uso Eliminar expediente de vivienda vinculada

<b>Caso de uso del sistema: Buscar Expediente Vivienda Vincula</b>	
<b>Curso Básico</b>	<p>El Operador escoge la opción vivienda vinculada dentro de la pantalla principal, el sistema valida que el usuario tenga los permisos para acceder a la pantalla, luego muestra la pantalla correspondiente a expediente de vivienda vinculada. El operador introduce el criterio de búsqueda y el sistema muestra las vivienda vinculadas que existen para esos parámetros hasta el momento.</p>
<b>Cursos Alternos</b>	<p>Validar: el sistema muestra un mensaje de error indicando que no puede acceder al módulo solicitado.</p>

Descripción textual del caso de uso Buscar expediente de vivienda vinculada

<b>Caso de uso del sistema: Crear expediente de ilegalidad en edificio multifamiliar</b>	
<b>Curso Básico</b>	<p>El Operador escoge la opción Edificio Multifamiliar dentro de la pantalla principal, el sistema valida que el usuario tenga los permisos para acceder a la pantalla, luego muestra la pantalla correspondiente a Expedientes de Edificio Multifamiliar y hace clic en el ícono confeccionar expediente de ilegalidades y pasa a la pantalla Expediente de ilegalidades. El Operador introduce los datos (ilegalidad y descripción) y hace clic en el botón Registrar. El sistema valida que no haya campos vacíos,</p>

	o caracteres inadecuados y almacena la información mostrando un mensaje de confirmación.
<b>Cursos Alternos</b>	<p>Validar: si no tiene permisos para acceder a la pantalla solicitada el sistema muestra un mensaje de error no tiene acceso a este módulo.</p> <p>Valida datos: el administrador introduce los datos en un formato incorrecto y deja campos en blanco. El sistema muestra un mensaje de error.</p>

Descripción textual del caso de uso Crear expediente de ilegalidad en edificio m

<b>Caso de uso del sistema: Modificar Tipología</b>	
<b>Curso Básico</b>	<p>El Administrador escoge la opción tipología dentro de la pantalla principal , el sistema valida que el usuario tenga los permisos para acceder a la pantalla, luego muestra la pantalla correspondiente a Administración de tipología, luego hace clic en el ícono modificar tipología y pasa a la pantalla editando tipología mostrando los datos de la tipología. El Administrador modifica los datos que desea cambiar y hace clic en el botón Guardar. El sistema valida que no haya campos vacíos o caracteres inadecuados y comprueba que no exista la tipología en la Base de Datos, almacena la información y muestra un mensaje de confirmación.</p>
<b>Cursos Alternos</b>	<p>Validar: el sistema muestra un mensaje de error indicando que no puede acceder al módulo solicitado.</p> <p>Valida datos: el administrador introduce los datos en un formato incorrecto, deja campos en blanco. El sistema</p>



	<p>muestra un mensaje de error.</p> <p>Existe Tipología: el administrador introduce una tipología existente y el sistema muestra un mensaje de error.</p>
--	---

Descripción textual del caso de uso Modificar tipología

<b>Caso de uso del sistema: Eliminar Tipología</b>	
<b>Curso Básico</b>	<p>El Administrador escoge la opción tipología dentro de la pantalla principal, el sistema valida que el usuario tenga los permisos para acceder a la pantalla, luego muestra la pantalla correspondiente a administración de tipología, luego selecciona la tipología o las tipología que desea eliminar y hace clic en el ícono Eliminar mostrándose un mensaje de si desea realmente eliminar la tipología, el administrador hace clic en el botón Aceptar, se elimina la tipología de la base de datos y el sistema muestra un mensaje de confirmación.</p>
<b>Cursos Alternos</b>	<p>El Administrador hace clic en el botón Cancelar y no se realiza la acción.</p> <p>Validar: el sistema muestra un mensaje de error indicando que no puede acceder al módulo solicitado.</p>

Descripción textual del caso de uso Eliminar tipología

<b>Caso de uso del sistema: Crear Expediente Cese</b>	
<b>Curso Básico</b>	<p>El Operador escoge la opción vivienda vinculada de la pantalla principal, el sistema valida que el usuario tenga los permisos para acceder a la pantalla, luego muestra la pantalla correspondiente a Expedientes de Viviendas vinculada, luego hace clic en el ícono confeccionar</p>

	<p>expediente de cese y pasa a la pantalla Expedientes de cese. El Operador escoge el tipo de pago e introduce los datos (motivo de cese, resolución de cese, mensualidad al 50 por ciento, impuesto arrendamiento, fecha de pago, importe total) y hace clic en el botón Registrar. El sistema valida que no haya campos vacíos, o caracteres inadecuados y almacena la información mostrando un mensaje de confirmación.</p>
<p><b>Cursos Alternos</b></p>	<p>Validar: el sistema muestra un mensaje de error indicando que no puede acceder al módulo solicitado.</p> <p>Valida datos: el administrador introduce los datos en un formato incorrecto, deja campos en blanco. El sistema muestra un mensaje de error.</p>

Descripción textual del caso de uso Crear expediente de cese

## Anexo 2: Diagramas de Robustez

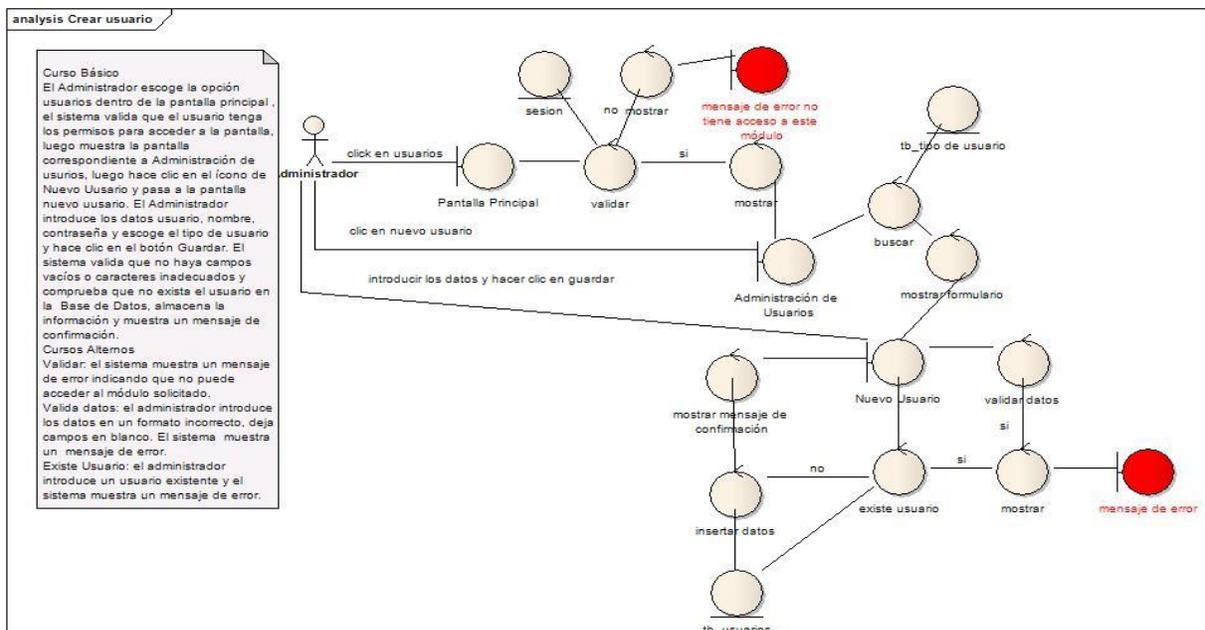


Diagrama de Robustez del caso de uso Crear usuario

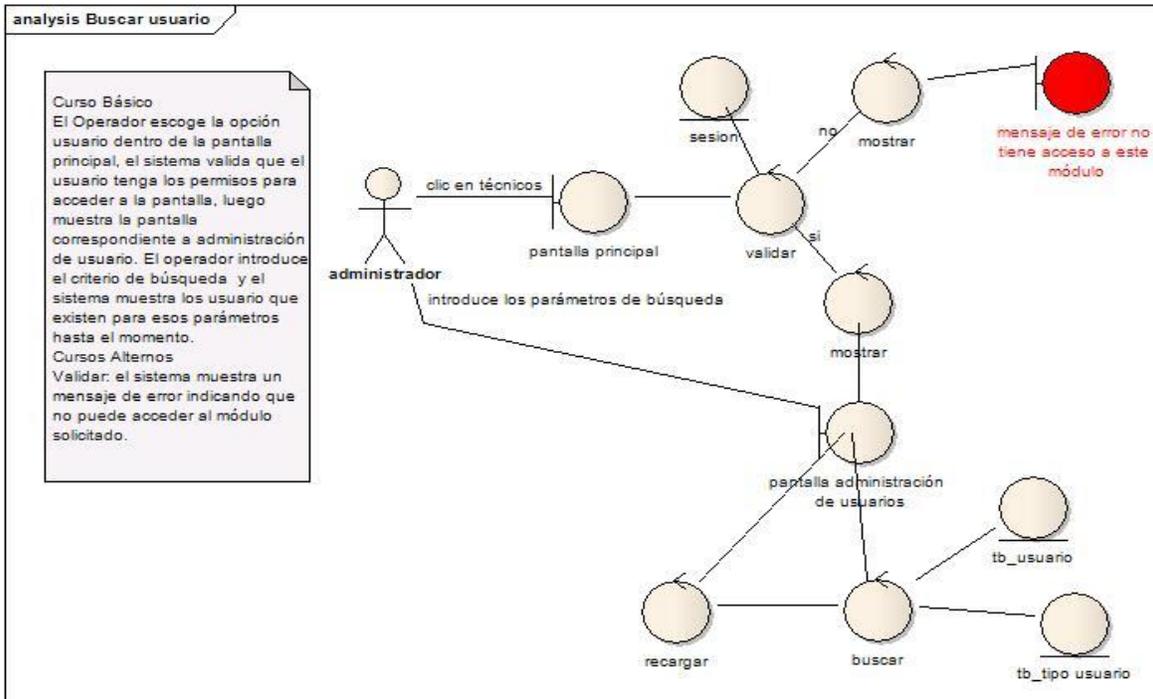


Diagrama de Robustez del caso de uso Buscar usuario

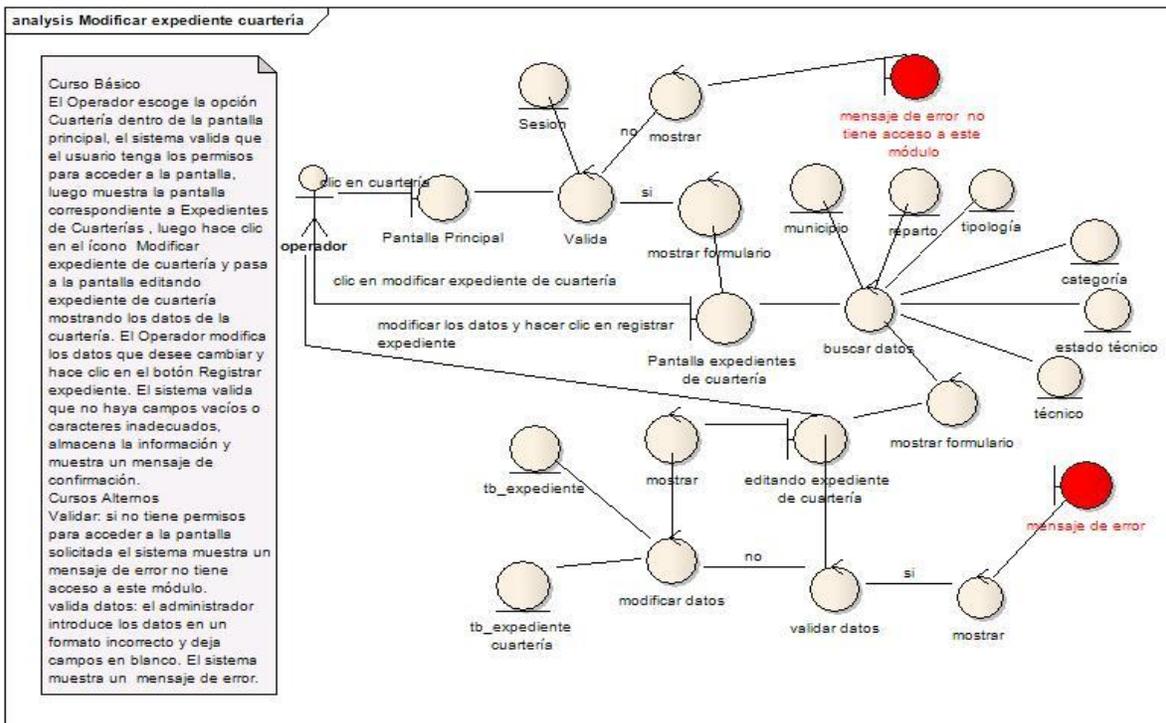


Diagrama de Robustez del caso de uso Modificar expediente de cuartería

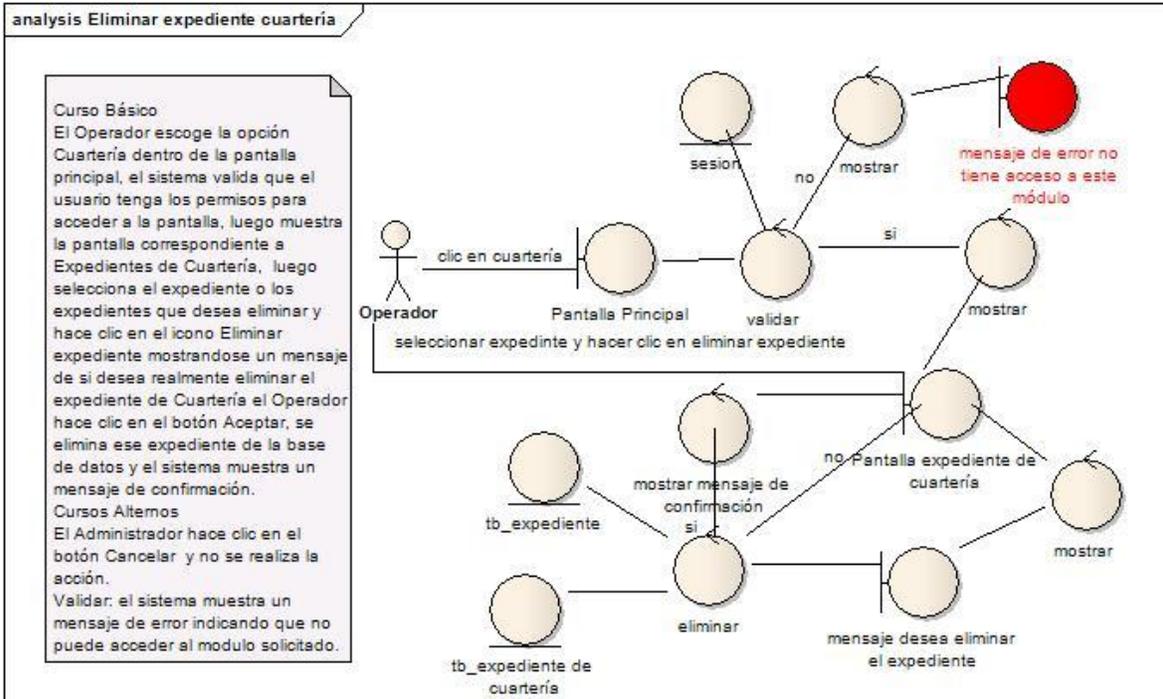


Diagrama de Robustez del caso de uso Eliminar expediente de cuartería

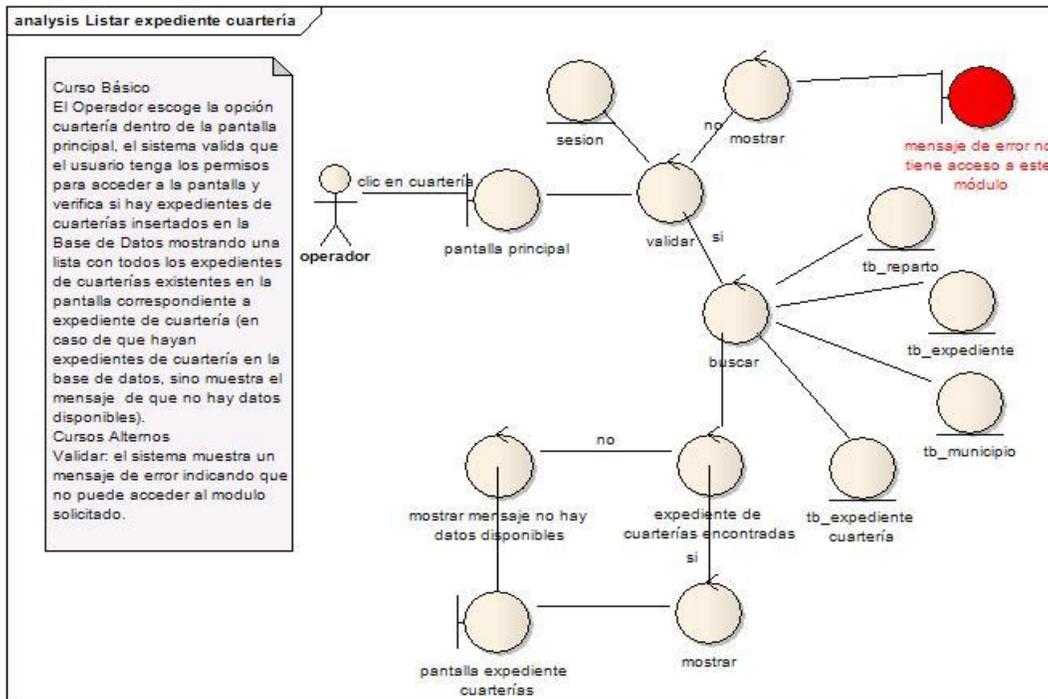


Diagrama de Robustez del caso de uso Listar expediente de cuartería

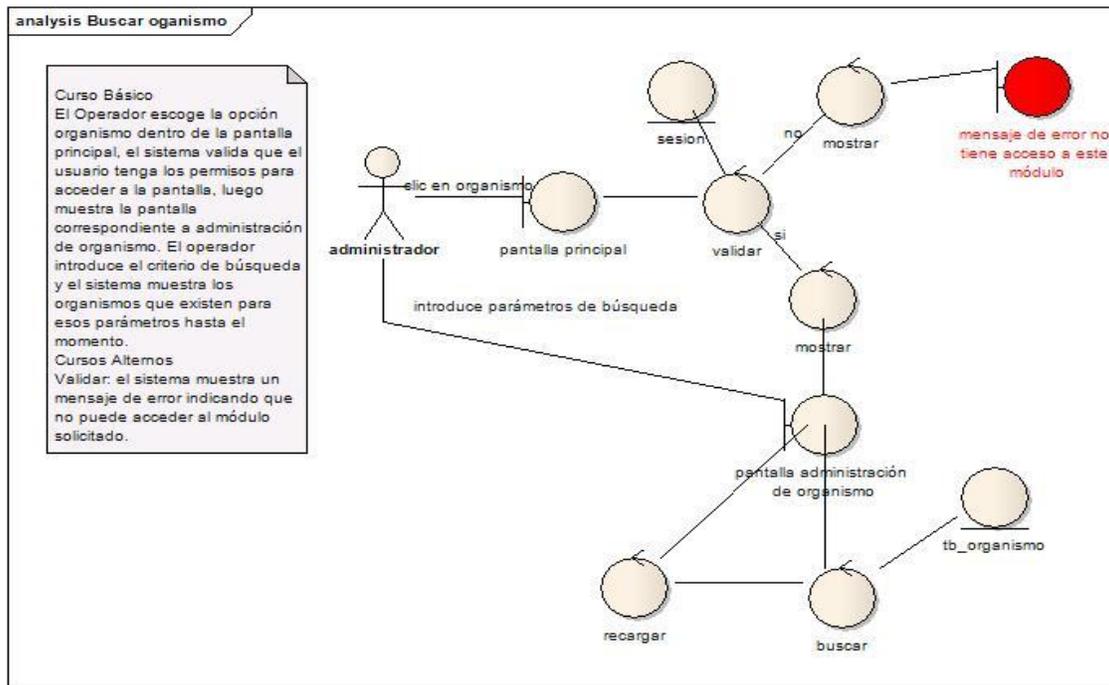


Diagrama de Robustez del caso de uso Buscar organismo

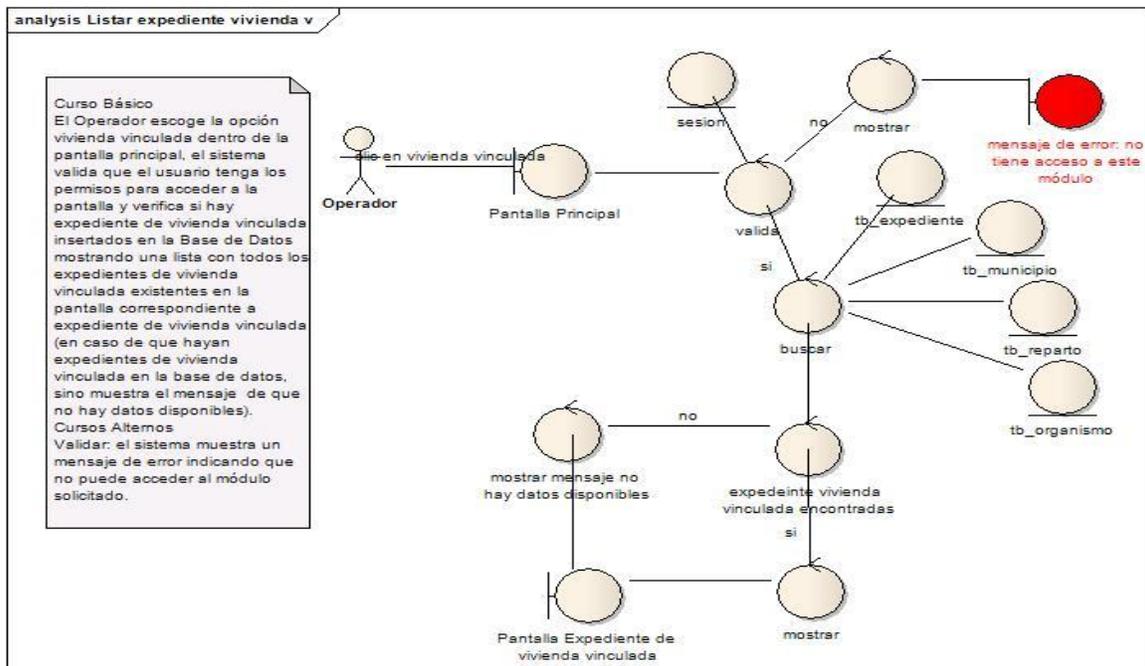


Diagrama de Robustez del caso de uso Listar expediente de vivienda v

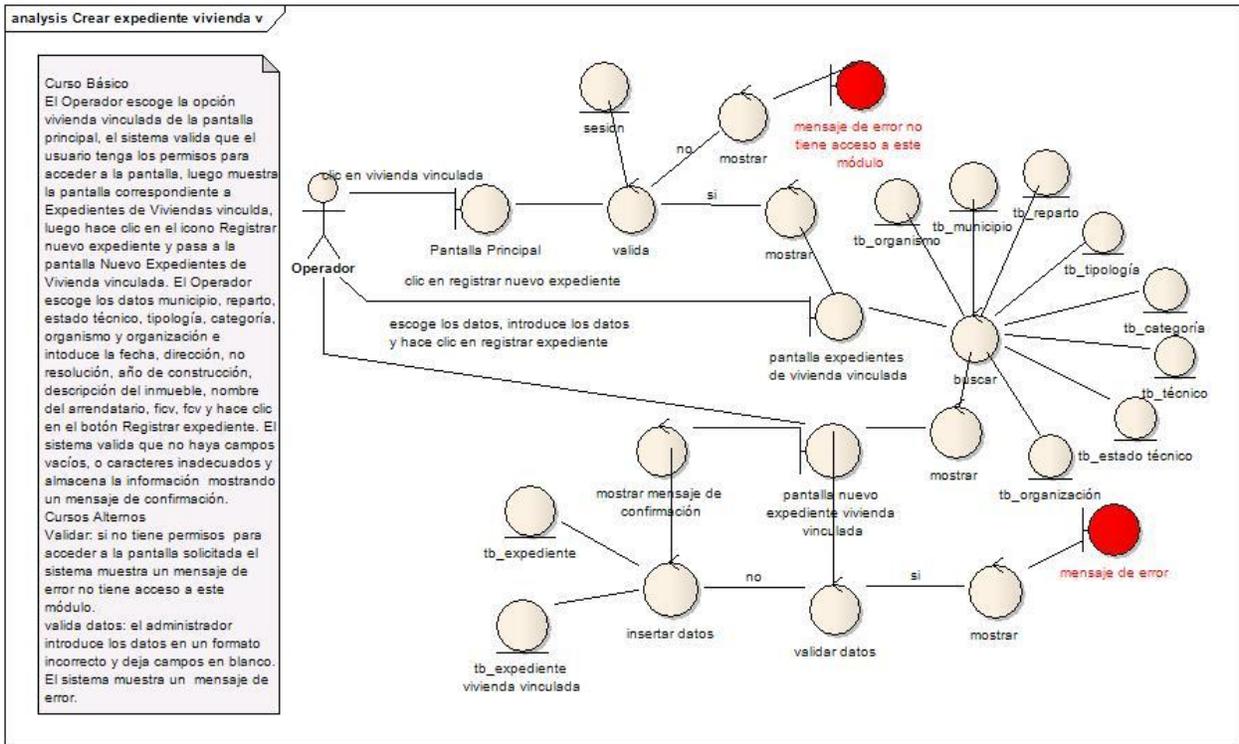


Diagrama de Robustez del caso de uso Crear expediente de vivienda v

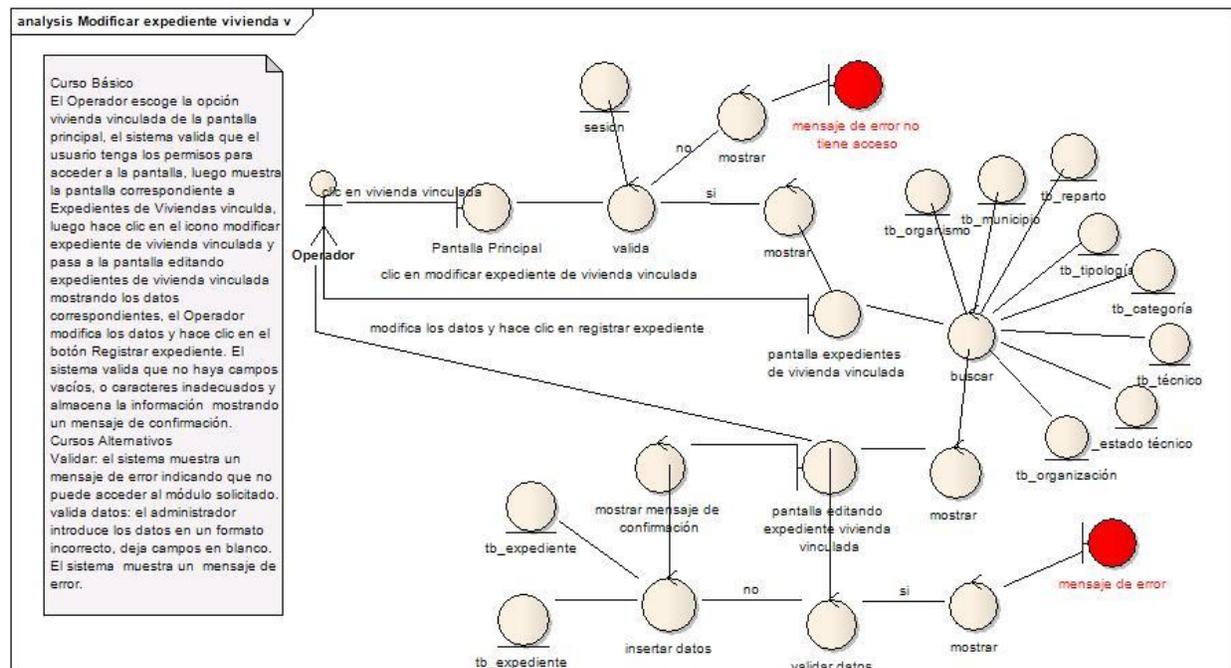


Diagrama de Robustez del caso de uso Modificar expediente de vivienda v

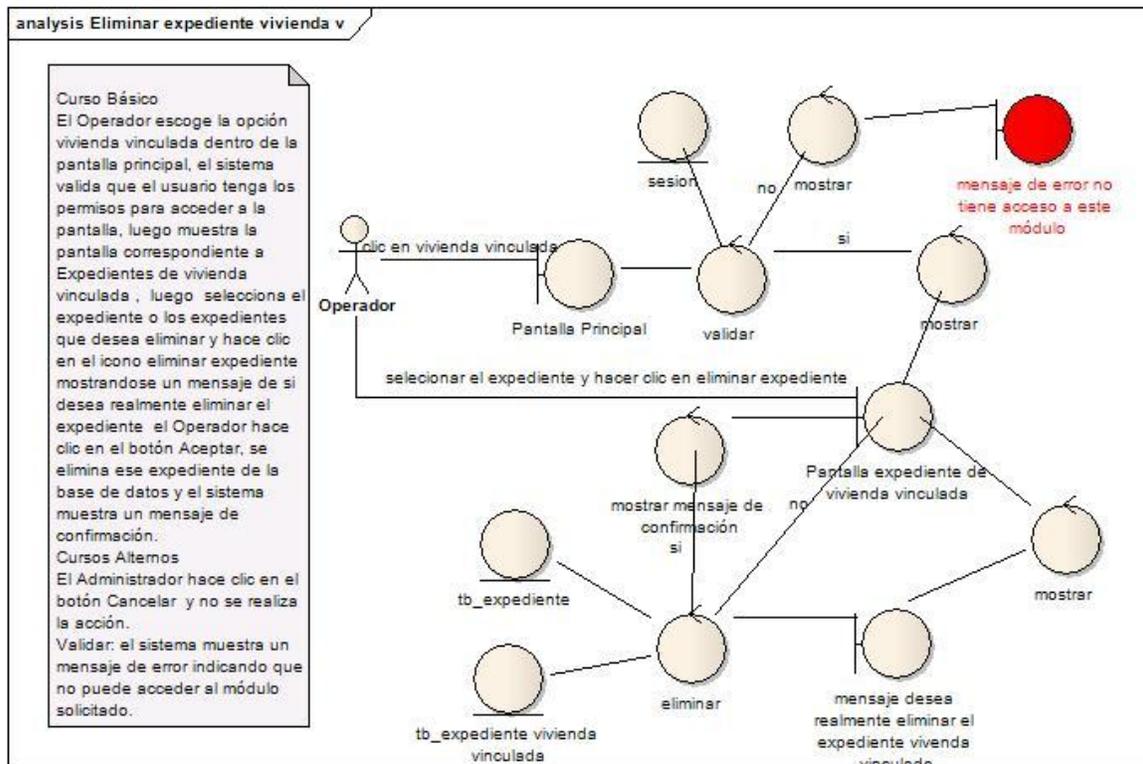


Diagrama de Robustez del caso de uso Eliminar expediente de vivienda v

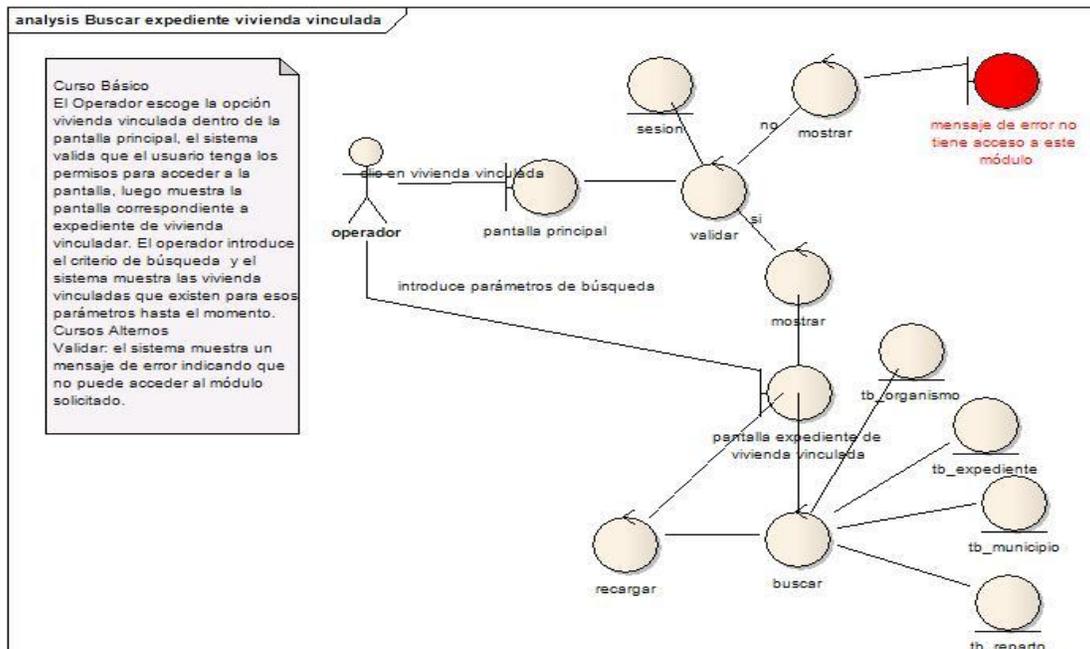


Diagrama de Robustez del caso de uso Buscar expediente de vivienda v

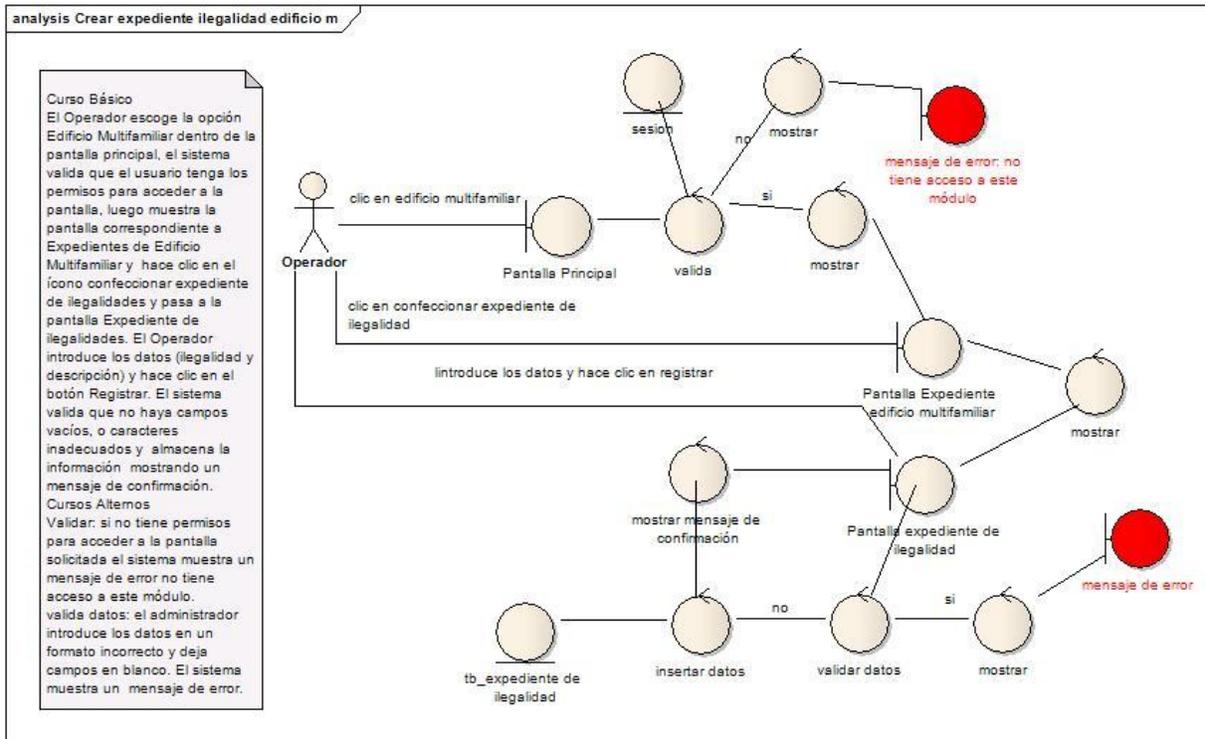


Diagrama de Robustez del caso de uso Crear expediente de ilegalidad en edificio m

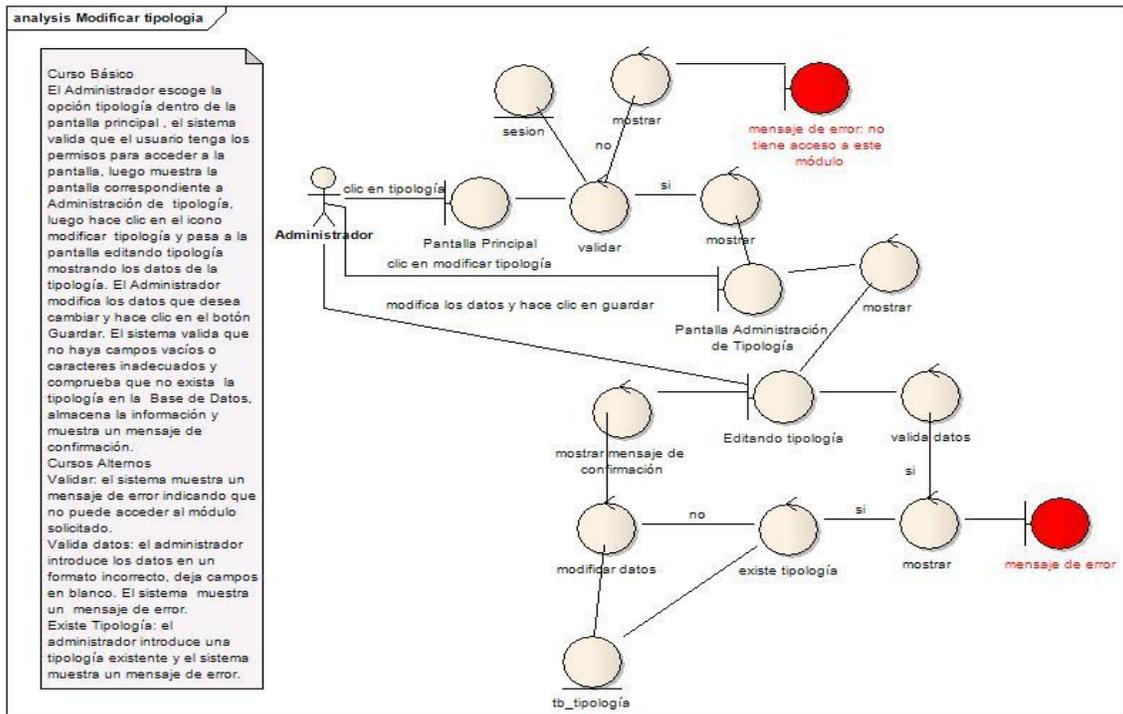


Diagrama de Robustez del caso de uso Modificar tipología

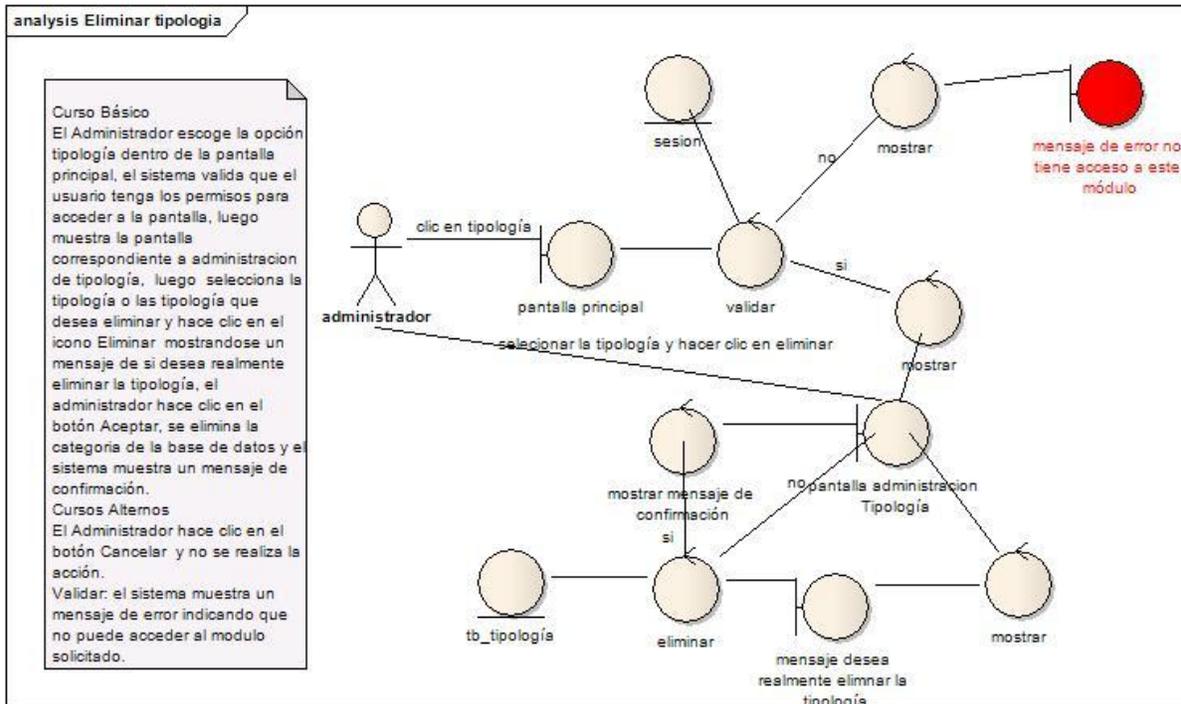


Diagrama de Robustez del caso de uso Eliminar tipología

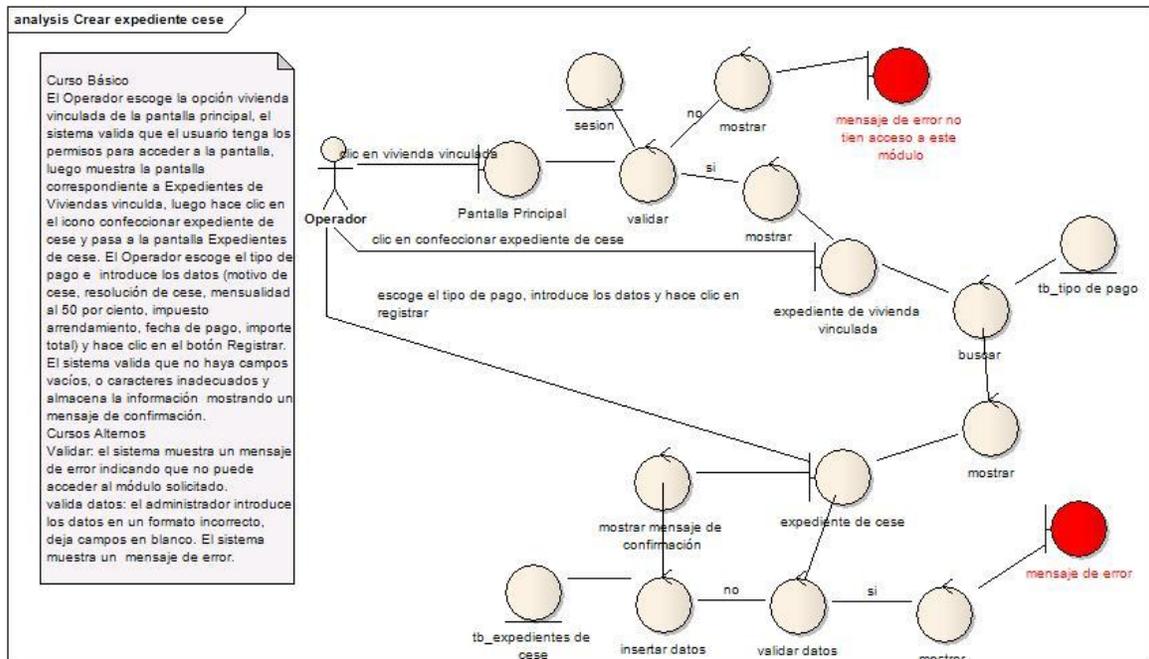


Diagrama de Robustez del caso de uso Crear expediente de cese

### Anexo 3: Diagramas de Secuencia

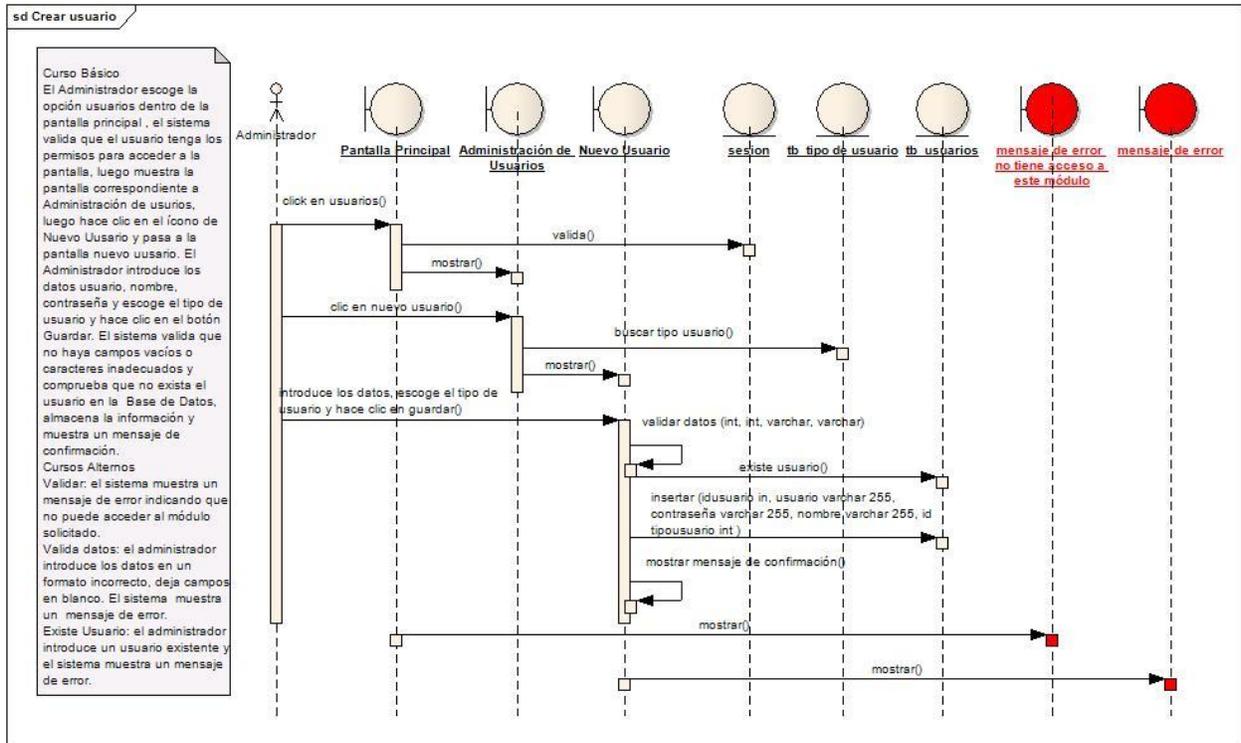


Diagrama de Secuencia del caso de uso Crear usuario

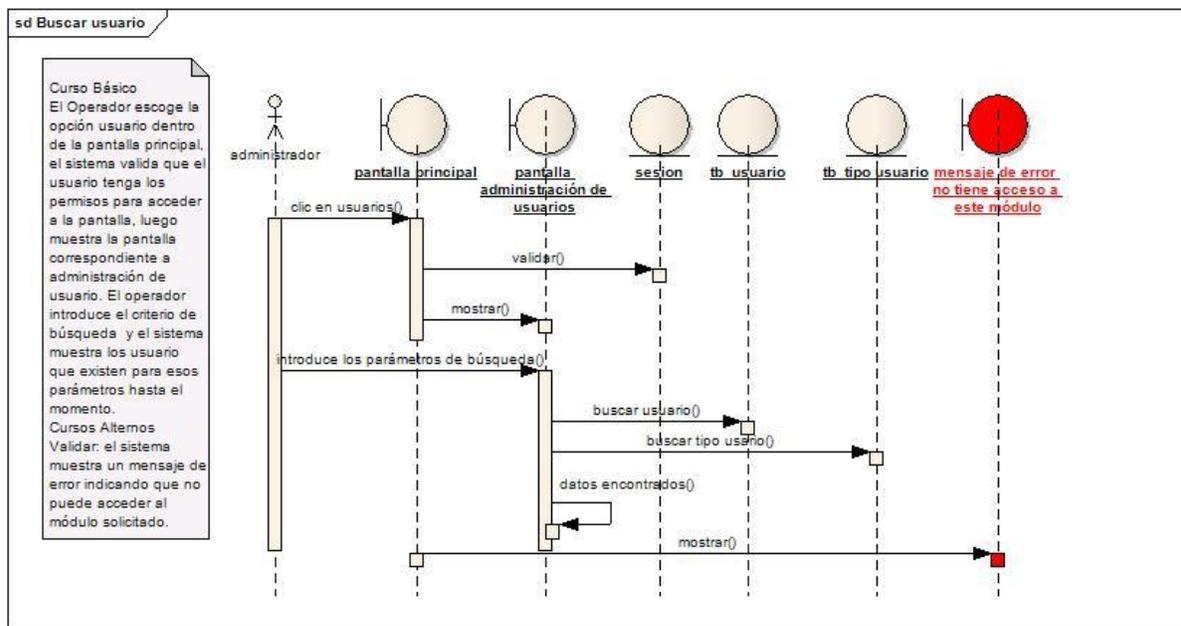


Diagrama de Secuencia del caso de uso Buscar usuario

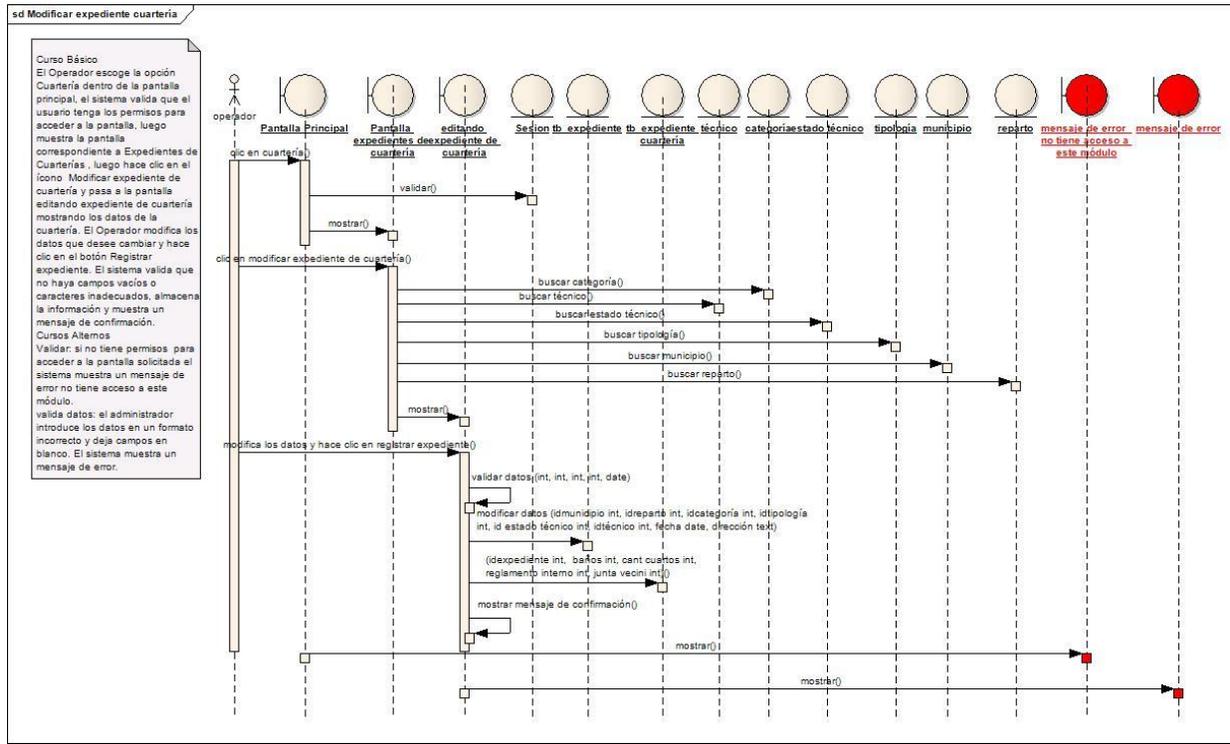


Diagrama de Secuencia del caso de uso Modificar expediente de cuartería

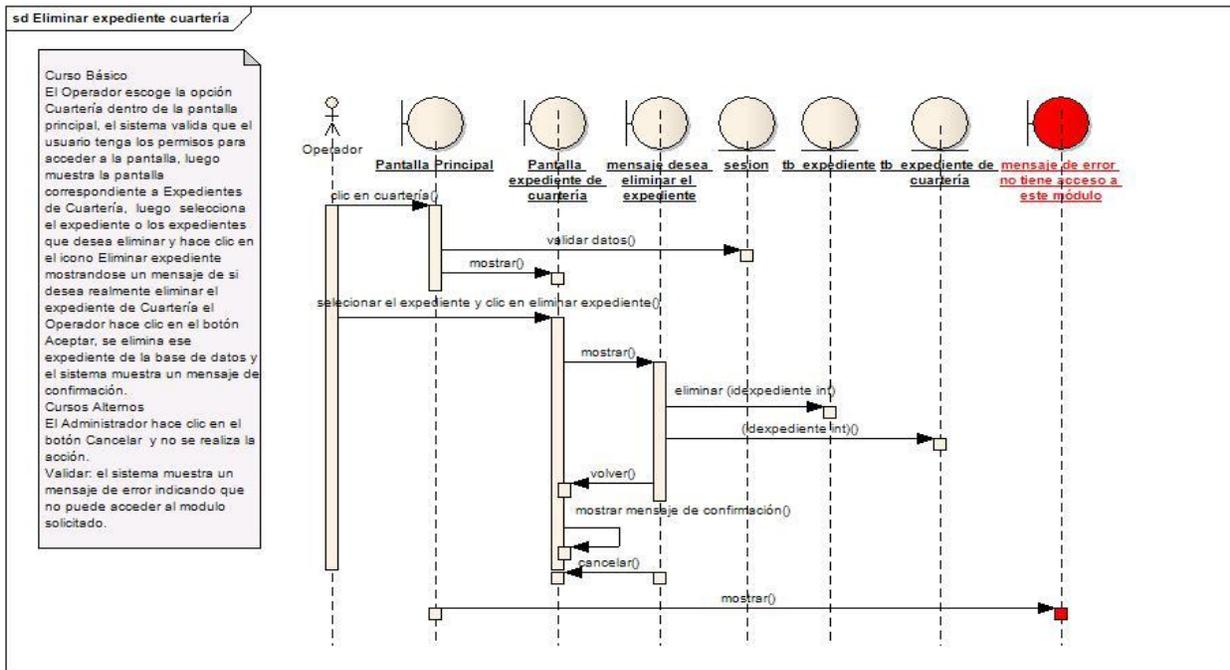


Diagrama de Secuencia del caso de uso Eliminar expediente de cuartería

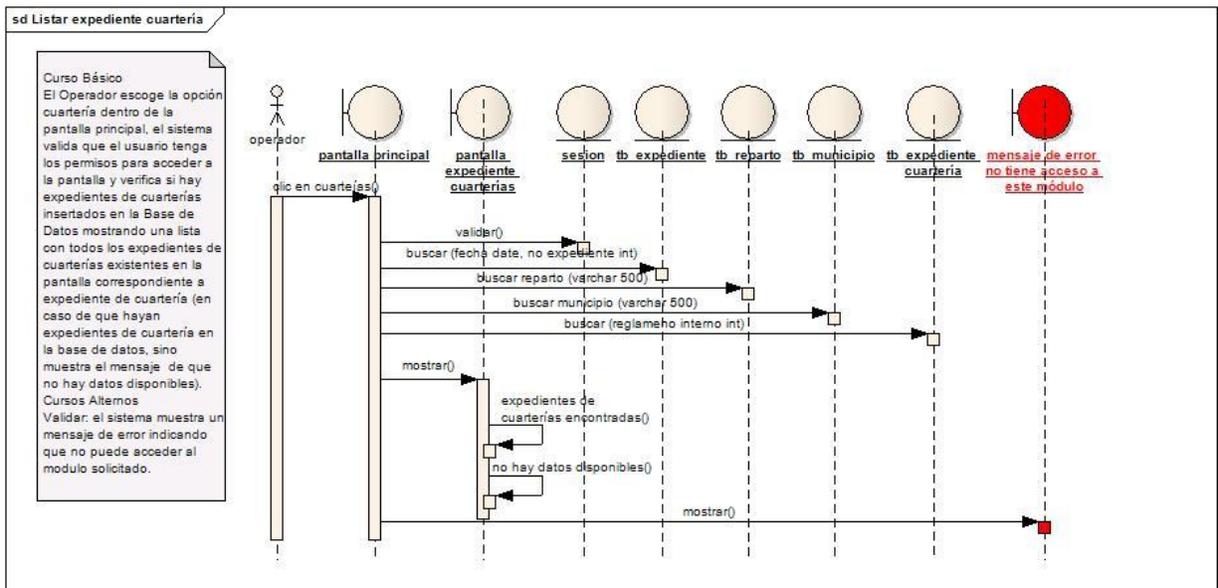


Diagrama de Secuencia del caso de uso Listar expediente de cuartería

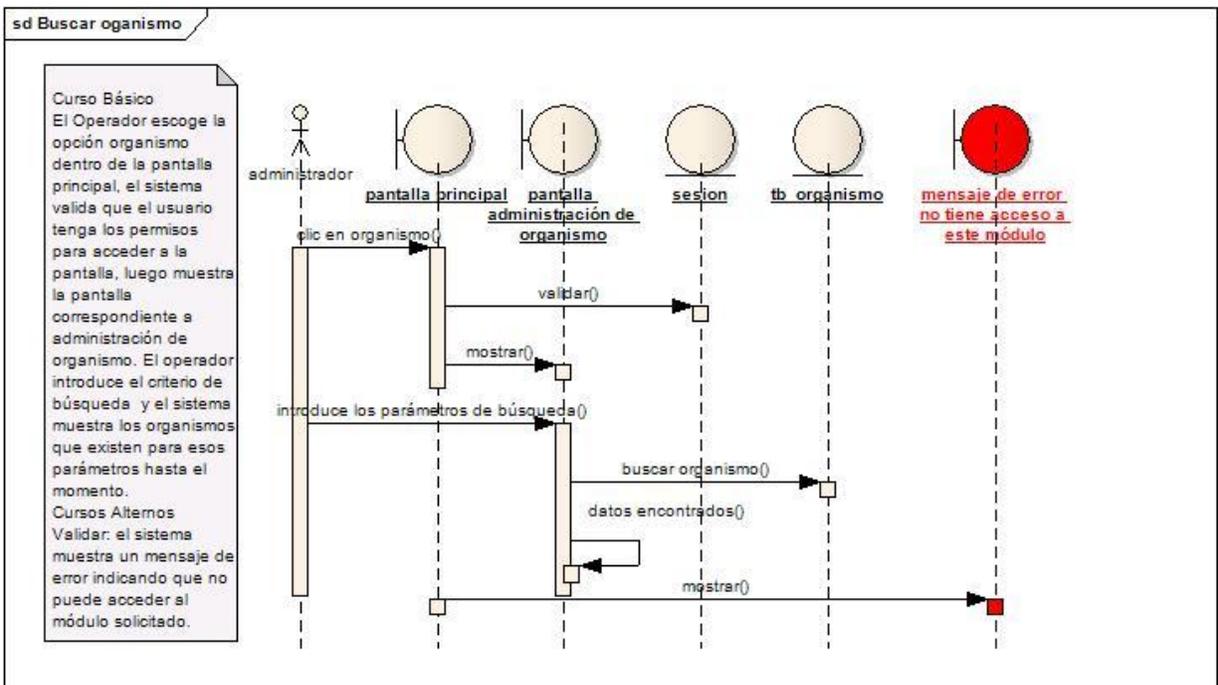


Diagrama de Secuencia del caso de uso Buscar organismo

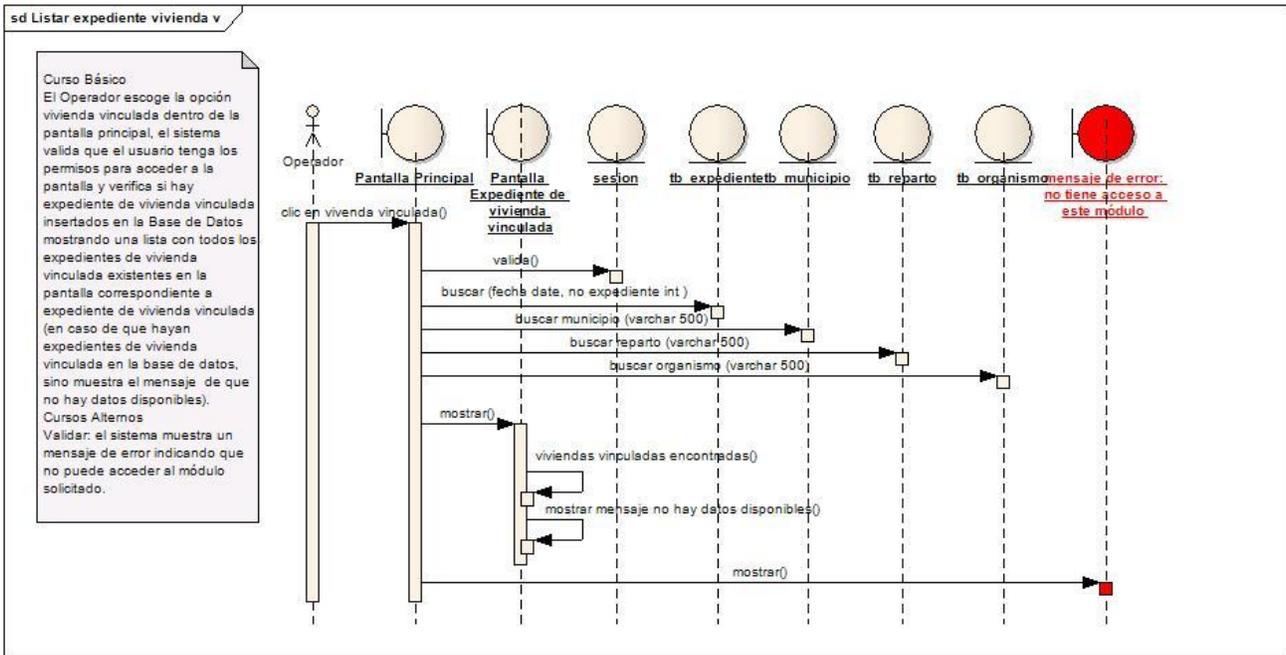


Diagrama de Secuencia del caso de uso Listar expediente de vivienda v

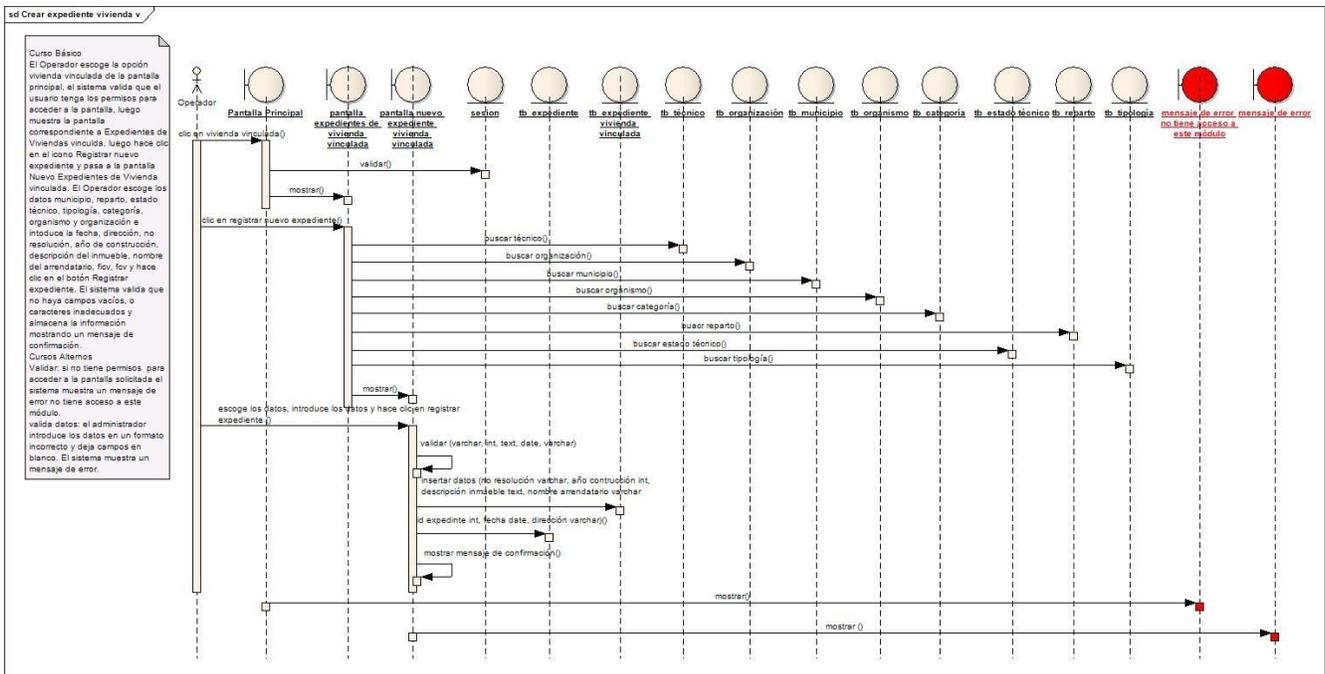


Diagrama de Secuencia del caso de uso Crear expediente de vivienda vinculada

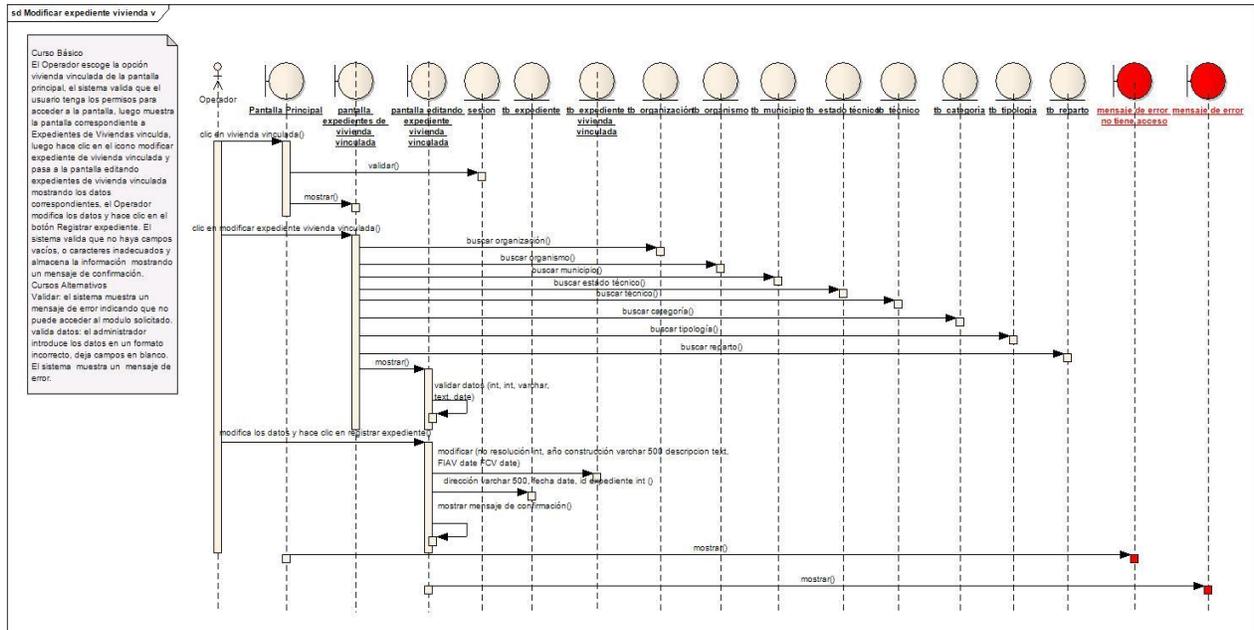


Diagrama de Secuencia del caso de uso Modificar expediente de vivienda v

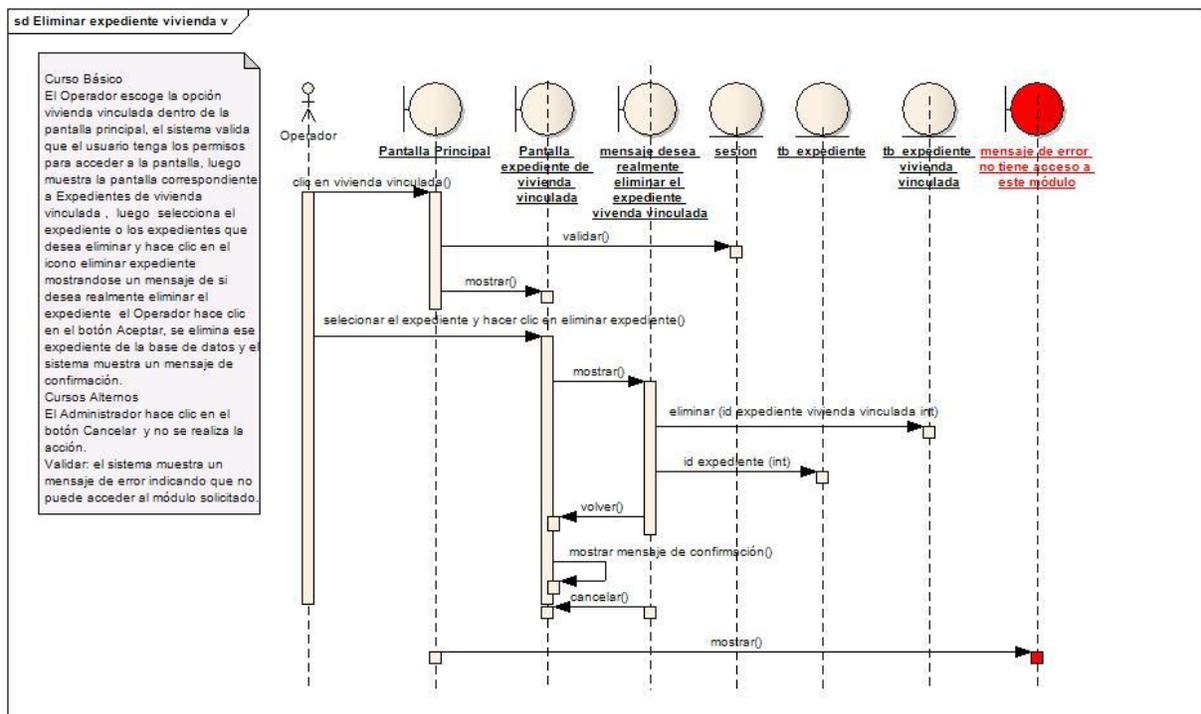


Diagrama de Secuencia del caso de uso Eliminar expediente de vivienda v

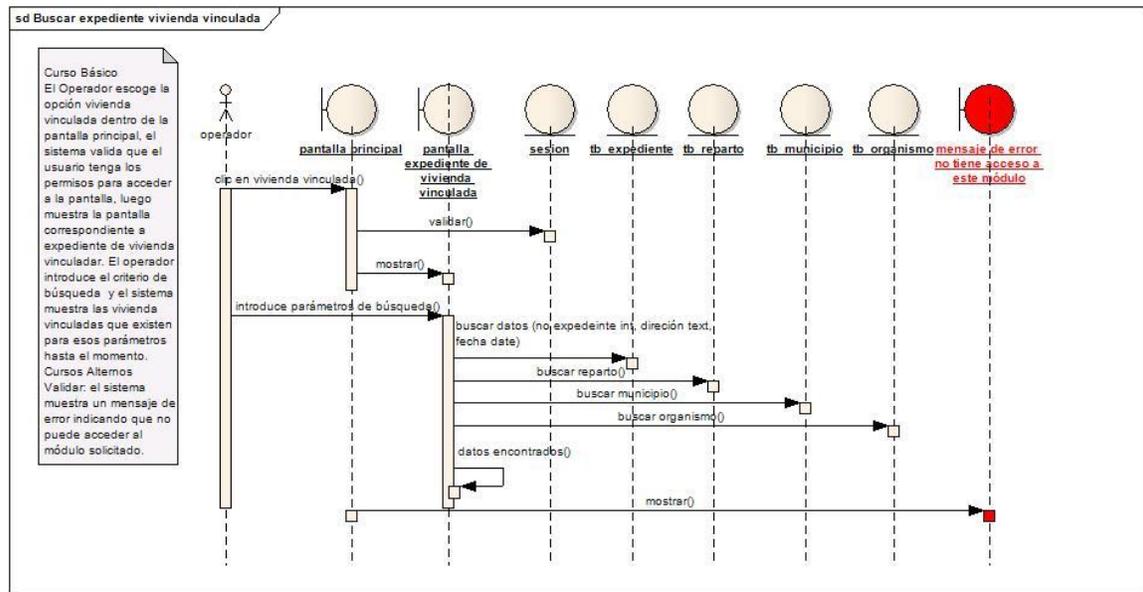


Diagrama de Secuencia del caso de uso Buscar expediente de vivienda v

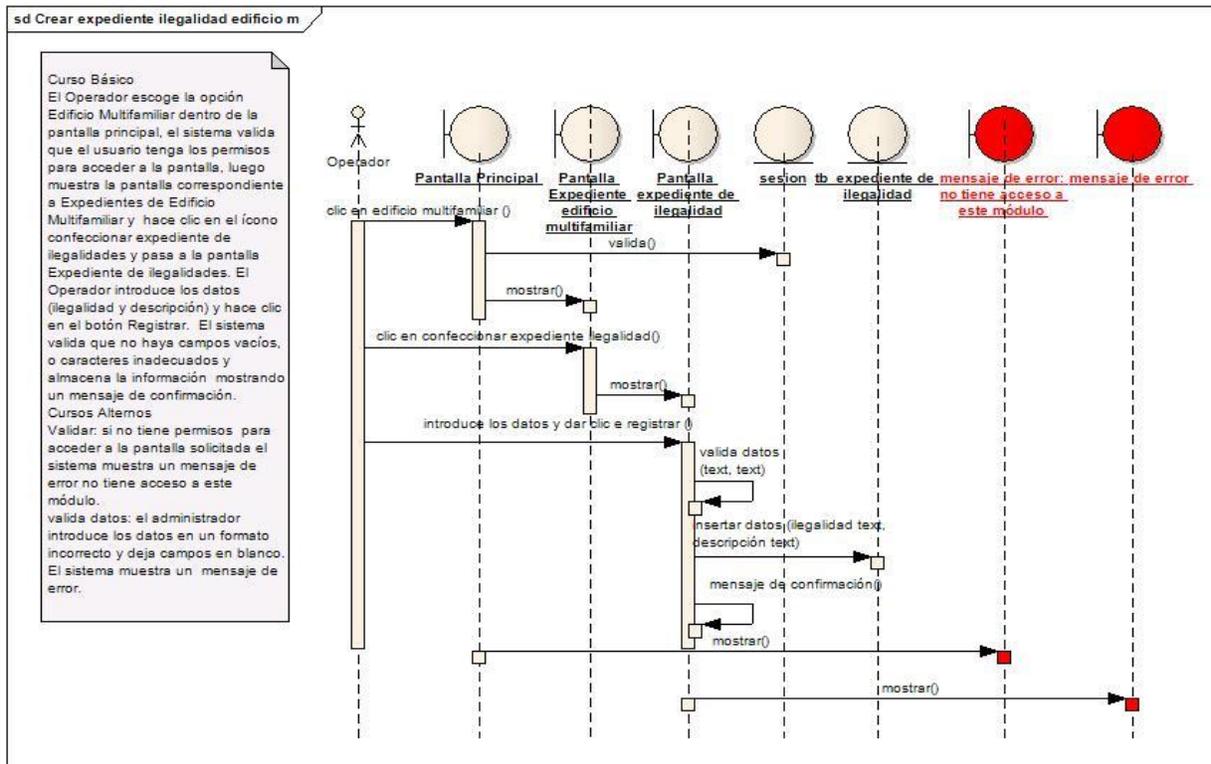


Diagrama de Secuencia del caso de uso Crear expediente de ilegalidad en edificio m

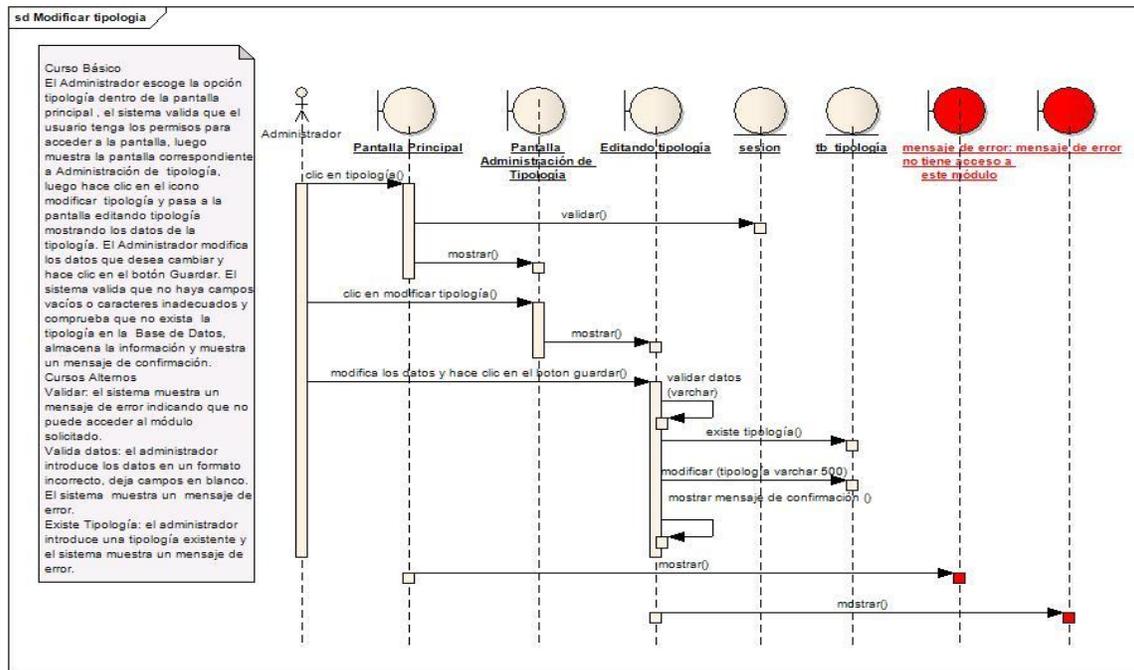


Diagrama de Secuencia del caso de uso Modificar tipología

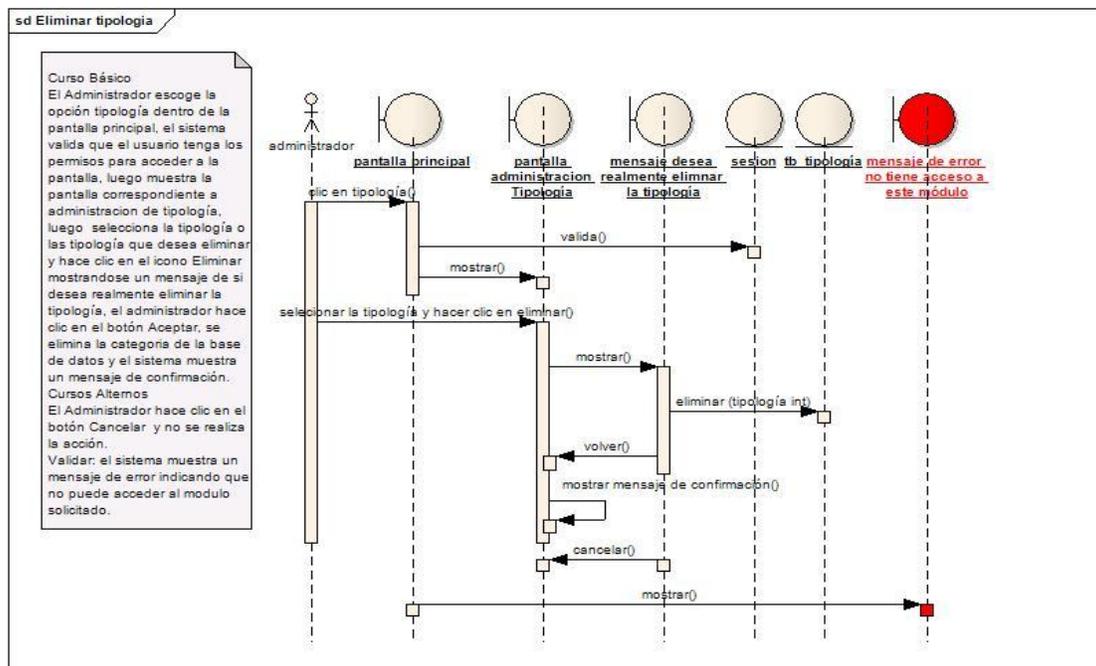


Diagrama de Secuencia del caso de uso Eliminar tipología

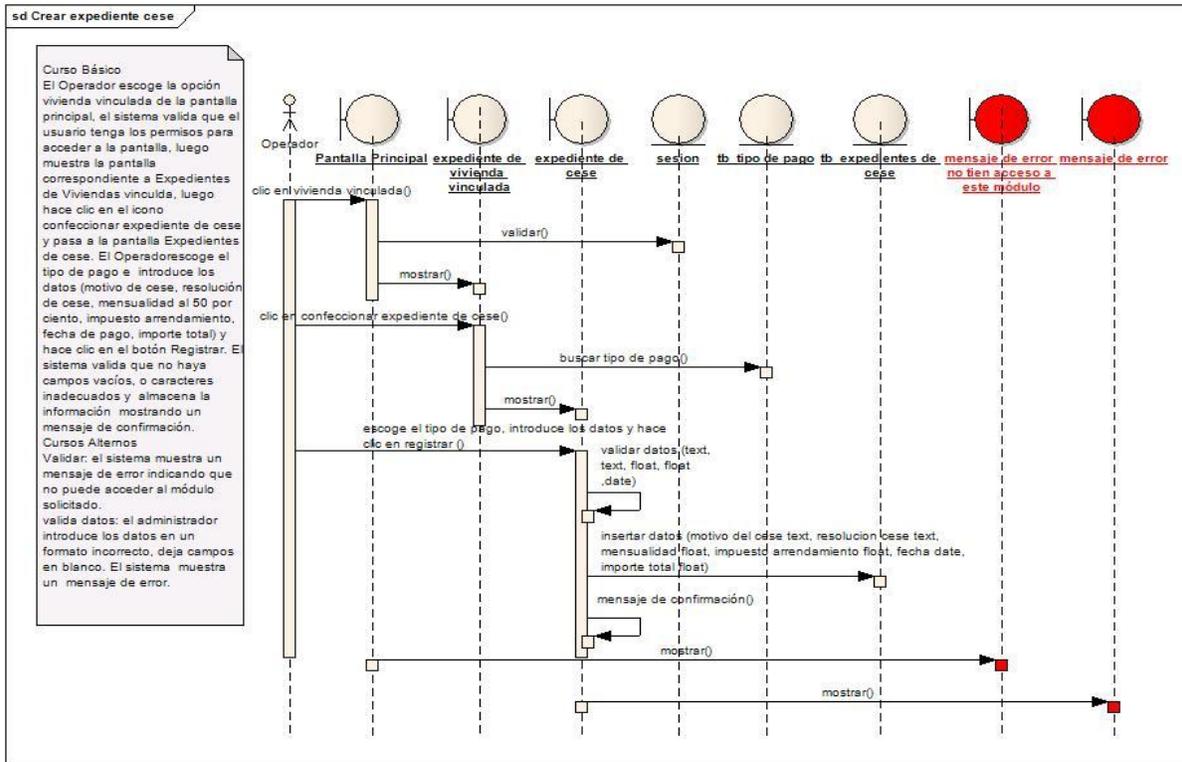
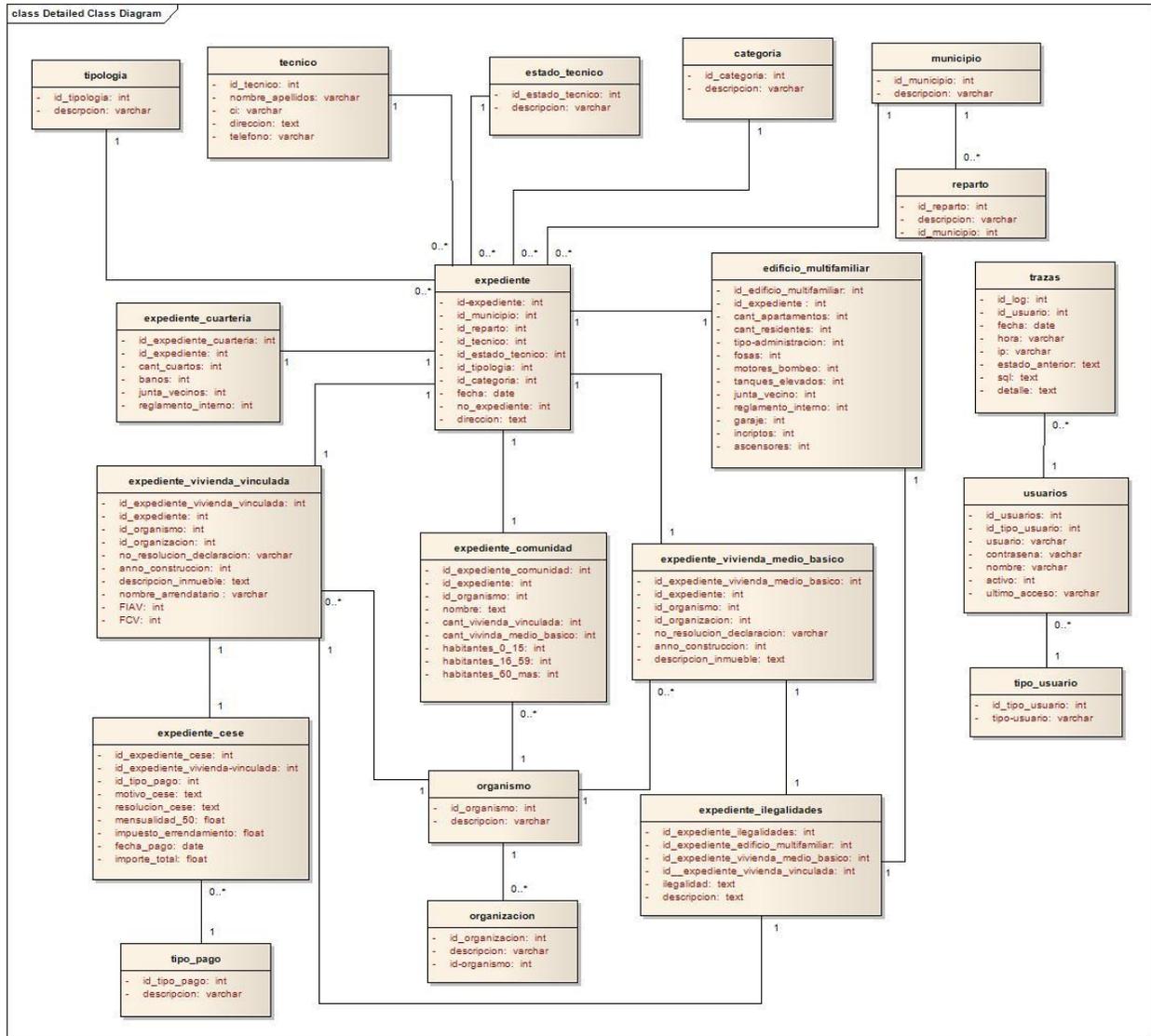
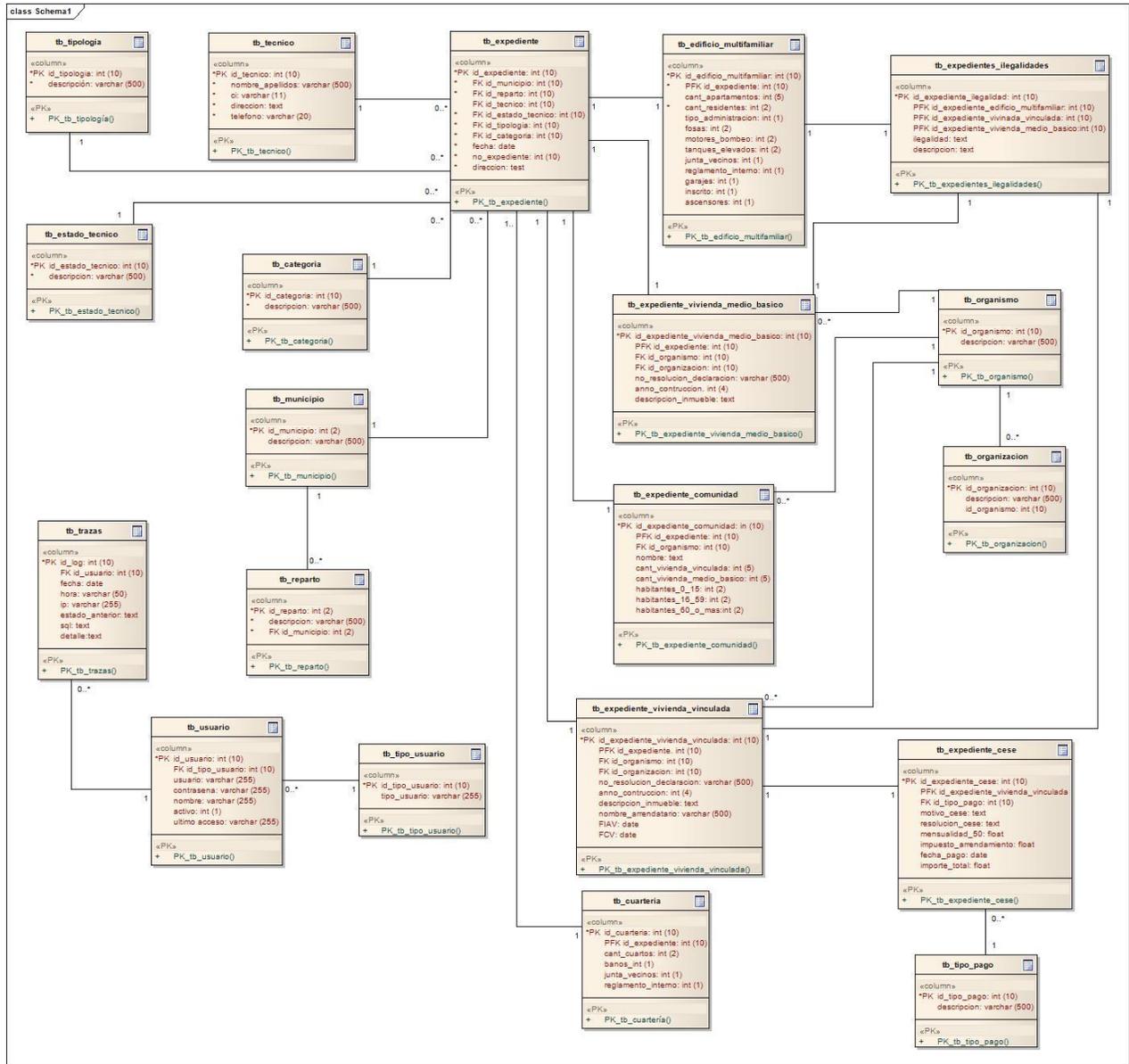


Diagrama de Secuencia del caso de uso Crear expediente de cese

## Anexo 4: Diagrama de Clases



## Anexo 5: Modelo de Datos



## Anexo 6: Factor de Peso de los casos de uso sin ajustar

Los únicos casos de uso y actores

Package	Name	Type	Complexy	Estimate Cost	Current Cost	Done %
Administración del sistema	Crear usuario	*	600,00	0	0	*
Administración del sistema	Listar estado técnico	*	600,00	0	0	*
Administración del sistema	Modificar estado técnico	*	600,00	0	0	*
Administración del sistema	Eliminar tipología	*	600,00	0	0	*
Administración del sistema	Listar categoría	*	600,00	0	0	*
Administración del sistema	Eliminar categoría	*	600,00	0	0	*
Administración del sistema	Listar tipo de pago	*	600,00	0	0	*
Administración del sistema	Eliminar tipo de pago	*	600,00	0	0	*
Administración del sistema	Listar Organización	*	600,00	0	0	*
Administración del sistema	Eliminar Organización	*	600,00	0	0	*
Administración del sistema	Modificar organismo	*	600,00	0	0	*
Administración del sistema	Listar usuario	*	600,00	0	0	*



Package	Name	Type	Comple xity	Estimat e Cost	Current Cost	Do ne %
Administración del sistema	Modificar categoría	*	600,00	0	0	*
Administración del sistema	Buscar organización	*	600,00	0	0	*
Administración del sistema	Buscar categoría	*	600,00	0	0	*
Administración del sistema	Buscar estado técnico	*	600,00	0	0	*
Administración del sistema	Buscar organismo	*	600,00	0	0	*
Administración del sistema	Buscar técnico	*	600,00	0	0	*
Administración del sistema	Buscar tipo pago	*	600,00	0	0	*
Administración del sistema	Buscar tipología	*	600,00	0	0	*
Administración del sistema	Buscar usuario	*	600,00	0	0	*
Administración del sistema	Modificar Organización	*	600,00	0	0	*
Administración del sistema	Crear tipo de pago	*	600,00	0	0	*
Administración del sistema	Modificar tipo de pago	*	600,00	0	0	*
Administración del sistema	Listar organismo	*	600,00	0	0	*



Package	Name	Type	Comple xity	Estimat e Cost	Current Cost	Do ne %
Administración del sistema	Eliminar organismo	***	1200,00	0	0	***
Administración del sistema	Modificar tipología	*	600,00	0	0	*
Administración del sistema	Eliminar un usuario	*	600,00	0	0	*
Administración del sistema	Eliminar estado técnico	*	600,00	0	0	*
Administración del sistema	Crear categoría	*	600,00	0	0	*
Administración del sistema	Crear estado técnico	*	600,00	0	0	*
Administración del sistema	Eliminar técnico	*	600,00	0	0	*
Administración del sistema	Crear tipología	***	1200,00	0	0	***
Administración del sistema	Listar tipología	*	600,00	0	0	*
Administración del sistema	Crear organismo	*	600,00	0	0	*
Administración del sistema	Modificar un usuario	*	600,00	0	0	*
Administración del sistema	Crear organización	*	600,00	0	0	*
Administración del sistema	Crear técnico	*	600,00	0	0	*



Package	Name	Type	Comple xity	Estimat e Cost	Current Cost	Do ne %
Administración del sistema	Ver las trazas del sistema	***	1200,00	0	0	***
Administración del sistema	Hacer salva de la base datos	*	600,00	0	0	*
Administración del sistema	Modificar técnico	*	600,00	0	0	*
Administración del sistema	Listar técnico	*	600,00	0	0	*
Gestión de expedientes	Modificar expediente ilegalidad vivienda v	***	1200,00	0	0	***
Gestión de expedientes	Eliminar expediente vivienda v	***	1200,00	0	0	***
Gestión de expedientes	Crear expediente cuartería	***	1200,00	0	0	***
Gestión de expedientes	Listar expediente cuartería	*	600,00	0	0	*
Gestión de expedientes	Modificar expediente cuartería	*	600,00	0	0	*
Gestión de expedientes	Eliminar expediente cuartería	*	600,00	0	0	*
Gestión de expedientes	Modificar expediente ilegalidad edificio m	***	1200,00	0	0	***
Gestión de expedientes	Crear expediente ilegalidad medio básico	***	1200,00	0	0	***



Package	Name	Type	Comple xity	Estimat e Cost	Current Cost	Do ne %
Gestión de expedientes	Modificar expediente ilegalidad medio básico	***	1200,00	0	0	***
Gestión de expedientes	Modificar expediente vivienda v	***	1200,00	0	0	***
Gestión de expedientes	Crear expediente ilegalidad vivienda v	***	1200,00	0	0	***
Gestión de expedientes	Crear expediente ilegalidad edificio m	*	600,00	0	0	*
Gestión de expedientes	Eliminar expediente ilegalidad vivienda v	***	1200,00	0	0	***
Gestión de expedientes	Ver reportes	***	1200,00	0	0	***
Gestión de expedientes	Buscar expediente de edificio multifamiliar	*	600,00	0	0	*
Gestión de expedientes	Buscar expediente vivienda medio básico	*	600,00	0	0	*
Gestión de expedientes	Buscar expediente de cuartería	*	600,00	0	0	*
Gestión de expedientes	Buscar expediente de comunidad	*	600,00	0	0	*
Gestión de expedientes	Buscar expediente vivienda vinculada	*	600,00	0	0	*
Gestión de expedientes	Eliminar expediente ilegalidad medio básico	***	1200,00	0	0	***



Package	Name	Type	Comple xity	Estimat e Cost	Current Cost	Do ne %
Gestión de expedientes	Modificar expediente medio básico	***	1200,00	0	0	***
Gestión de expedientes	Crear expediente edificio m	*	600,00	0	0	*
Gestión de expedientes	Crear expediente medio básico	*	600,00	0	0	*
Gestión de expedientes	Eliminar expediente ilegalidad edificio m	***	1200,00	0	0	***
Gestión de expedientes	Listar expediente medio básico	*	600,00	0	0	*
Gestión de expedientes	Listar expediente vivienda v	*	600,00	0	0	*
Gestión de expedientes	Eliminar expediente medio básico	***	1200,00	0	0	***
Gestión de expedientes	Crear expediente comunidad	***	1200,00	0	0	***
Gestión de expedientes	Crear expediente cese	*****	1800,00	0	0	**** **
Gestión de expedientes	Listar expediente edificio m	*	600,00	0	0	*
Gestión de expedientes	Eliminar expediente edificio m	*	600,00	0	0	*
Gestión de expedientes	Modificar expediente cese	*	600,00	0	0	*
Gestión de expedientes	Modificar expediente edificio m	*	600,00	0	0	*



Package	Name	Type	Comple xity	Estimat e Cost	Current Cost	Do ne %
Gestión de expedientes	Modificar expediente comunidad	*	600,00	0	0	*
Gestión de expedientes	Crear expediente vivienda v	*	600,00	0	0	*
Gestión de expedientes	Eliminar expediente cese	*	600,00	0	0	*
Gestión de expedientes	Listar expediente comunidad	*	600,00	0	0	*
Gestión de expedientes	Eliminar expediente comunidad	***	1200,00	0	0	***
Seguridad	Autenticarse	*	600,00	0	0	*
Seguridad	Salir del sistema	*	600,00	0	0	*
Seguridad	Validar Usuario	*	600,00	0	0	*
Seguridad	Cambiar contraseña	*	600,00	0	0	*
Seguridad	Enviar notificación	***	1200,00	0	0	***

### Anexo 7: Encuesta para determinar el coeficiente de competencia del experto

**Nombre y apellidos:** \_\_\_\_\_.

Usted ha sido seleccionado como posible experto para ser consultado respecto al grado de relevancia de las posibilidades que brinda la aplicación Web Sistema para la gestión de los expedientes en el departamento administrativo de la Vivienda de la Dirección Provincial de Holguín.

Necesitamos antes de realizarle la consulta correspondiente como parte del método empírico de investigación “encuesta a expertos”, determinar su coeficiente de competencia en este tema, a los efectos de reforzar la validez del resultado de la consulta que realizaremos. Por esta razón le rogamos que responda las siguientes preguntas de la forma más objetiva que le sea posible.

1. Marque con una cruz (X), en la tabla siguiente, el valor que se corresponde con el grado de conocimientos que usted posee sobre el tema **“diseño de Aplicaciones Web”**. Considere que la escala que le presentamos es ascendente, es decir, el conocimiento sobre el tema referido va creciendo desde 0 hasta 10.

0	1	2	3	4	5	6	7	8	9	10

2. Realice una autovaloración del grado de influencia que cada una de las fuentes que le presentamos a continuación, ha tenido en su conocimiento y criterio sobre los indicadores que se han puesto a su consideración. Para ello marque con una cruz (X), según corresponda, en MA (muy alto), A (alto), M (medio), B (bajo) o MB (muy bajo).

Fuentes de argumentación.	Grado de influencia de cada una de las fuentes.				
	MA	A	M	B	MB
Nivel de conocimientos específicos en la materia					
Experiencia de orden empírico (práctica profesional).					

Experiencia en el desarrollo de investigaciones teóricas.					
Conocimiento del estado actual del problema.					
Comprensión del problema.					

**Muchas gracias por su colaboración.**

**Anexo 8: Determinación del coeficiente de competencia**

<b>Determinación del Coeficiente de Competencia</b>				
<b>No</b>	<b>Kc</b>	<b>Ka</b>	<b>K</b>	<b>Evaluación</b>
1	0,60	0,70	0,65	Medio
2	0,50	0,75	0,63	Medio
3	0,90	1,00	0,95	Alto
4	1,00	1,00	1,00	Alto
5	0,80	0,95	0,88	Alto
6	0,80	0,80	0,80	Alto
7	1,00	0,95	0,98	Alto
8	0,70	0,90	0,80	Alto
9	0,60	0,50	0,55	Medio
10	1,00	0,85	0,93	Alto
11	0,90	0,90	0,90	Alto
12	0,70	0,95	0,83	Alto
13	0,70	0,90	0,80	Alto
14	0,90	0,95	0,93	Alto
15	0,90	0,80	0,85	Alto
16	0,70	0,55	0,63	Medio
17	0,90	0,95	0,93	Alto
18	0,90	0,75	0,83	Alto
<b>Coeficiente de Competencia General</b>			0,82	Alto

### Anexo 9: Encuesta aplicada a los expertos para la valoración del sistema

<b>ENCUESTA A EXPERTOS</b>				
Sistema para la gestión de los expedientes en el departamento administrativo de la Vivienda de la Dirección Provincial de Holguín				
Encuesta No.:		Fecha(D/M/A):		
Departamento o Institución a la que pertenece:				
Nombre y apellidos:				
Cargo actual:				
Calificación profesional, grado científico o académico:				
Duodécimo:___	Licenciado:___	Especialista:___	Máster:___	Doctor:___
Años de experiencia en el cargo:				
Años de experiencia docente y/o en la investigación:				
<p>Estimado compañero(a), la entidad Dirección Provincial de la Vivienda de Holguín, con el propósito de recopilar opiniones y sugerencias acerca del Sistema propuesto para la gestión de los expedientes en el departamento administrativo de la Vivienda, aplica esta encuesta. Le pedimos por favor que lea cuidadosamente las preguntas siguientes y exprese su criterio. Le agradecemos anticipadamente el esfuerzo que sabemos hará para responder, con la mayor fidelidad posible a su manera de pensar.</p>				
<b>Indicaciones:</b>				
<p>A continuación sometemos a su valoración una serie de elementos que se consideran importantes en la evaluación de dicho sistema. Para esta evaluación se proponen las siguientes categorías:</p>				
Muy Relevante <b>(MR)</b>	Bastante Relevante <b>(BR)</b>	Relevante <b>(R)</b>	Poco Relevante <b>(PR)</b>	No Relevante <b>(NR)</b>
<p>Marque con una cruz (X) en la celda que se corresponda a la categoría que usted otorga a cada aspecto del sistema.</p>				
Sistema para la gestión de los expedientes en el departamento administrativo de la Vivienda				



<b>Aspectos a Considerar</b>	<b>MR</b>	<b>BR</b>	<b>R</b>	<b>PR</b>	<b>NR</b>
¿Esta Aplicación Web favorece el control de los expedientes de administración de la vivienda en la Dirección Provincial de Holguín y agiliza este proceso?					
¿Cómo aprecia el diseño de las interfaces del sistema?					
¿Qué le parece el uso de los colores y las imágenes usadas en el sistema?					
¿El sistema abarca la totalidad del proceso?					
¿Cómo considera usted el acceso a las funcionalidades del sistema?					
¿Cómo evalúa el tiempo de respuesta del sistema?					
¿Cómo valora el uso que se le puede dar al sistema?					
Que otros aspectos considera usted que debería de ser incluido o eliminado en esta propuesta.					
<b>Aspectos a incluir</b>	<b>Aspectos a eliminar</b>				
¿Qué elementos novedosos encuentra en la aplicación?					
¿Cuál es su opinión con respecto a la utilización del sistema?					
Otra idea o sugerencia que usted desea hacer con relación a la propuesta que no se haya abordado en esta encuesta.					

## Anexo 10: Procesamiento de la encuesta de opinión de los usuarios aplicándose el método Delphy

<b>Tabla de Frecuencia Absoluta</b>						
<b>Fases y Momentos</b>	<b>MR</b>	<b>BR</b>	<b>R</b>	<b>PR</b>	<b>NR</b>	<b>TOTAL</b>
¿Esta Aplicación Web favorece el control de los expedientes de administración de la vivienda en la Dirección Provincial de Holguín y agiliza este proceso?	13	1				14
¿Cómo aprecia el diseño de las interfaces del sistema?	11	3				14
¿Qué le parece el uso de los colores y las imágenes usadas en el sistema?	10	1	3			14
¿El sistema abarca la totalidad del proceso?	12	2				14
¿Cómo considera usted el acceso a las funcionalidades del sistema?	12		2			14
¿Cómo evalúa el tiempo de respuesta del sistema?	10	1	3			14
¿Cómo valora el uso que se le puede dar al sistema?	11	3				14

<b>Tabla de Frecuencia Absoluta Acumulada</b>					
<b>Fases y Momentos</b>	<b>MR</b>	<b>BR</b>	<b>R</b>	<b>PR</b>	<b>NR</b>
¿Esta Aplicación Web favorece el control de los expedientes de administración de la vivienda en la Dirección Provincial de Holguín y agiliza este proceso?	13	14	14	14	14
¿Cómo aprecia el diseño de las interfaces del sistema?	11	14	14	14	14
¿Qué le parece el uso de los colores y las imágenes usadas en el sistema?	10	11	14	14	14
¿El sistema abarca la totalidad del proceso?	12	14	14	14	14



¿Cómo considera usted el acceso a las funcionalidades del sistema?	12	12	14	14	14
¿Cómo evalúa el tiempo de respuesta del sistema?	10	11	14	14	14
¿Cómo valora el uso que se le puede dar al sistema?	11	14	14	14	14

<b>Tabla del Inverso de la Frecuencia Absoluta Acumulada</b>				
<b>Fases y Momentos</b>	<b>MR</b>	<b>BR</b>	<b>R</b>	<b>PR</b>
¿Esta Aplicación Web favorece el control de los expedientes de administración de la vivienda en la Dirección Provincial de Holguín y agiliza este proceso?	0,9286	1	1	1
¿Cómo aprecia el diseño de las interfaces del sistema?	0,7857	1	1	1
¿Qué le parece el uso de los colores y las imágenes usadas en el sistema?	0,7143	0,7857	1	1
¿El sistema abarca la totalidad del proceso?	0,8571	1	1	1
¿Cómo considera usted el acceso a las funcionalidades del sistema?	0,8571	0,8571	1	1
¿Cómo evalúa el tiempo de respuesta del sistema?	0,7143	0,7857	1	1
¿Cómo valora el uso que se le puede dar al sistema?	0,9	1	1	1

<b>Tabla de Determinación de los Puntos de Corte</b>							
<b>Fases y Momentos</b>	<b>MR</b>	<b>BR</b>	<b>R</b>	<b>PR</b>	<b>Suma</b>	<b>Pro medio</b>	<b>N-Prom</b>
¿Esta Aplicación Web favorece el control de los expedientes de administración de la vivienda en la Dirección Provincial de Holguín y agiliza este proceso?	1,47	3,49	3,49	3,49	11,94	2,99	-0,43
¿Cómo aprecia el diseño de las	0,79	3,49	3,49	3,49	11,26	2,82	-0,26



interfaces del sistema?							
¿Qué le parece el uso de los colores y las imágenes usadas en el sistema?	0,57	0,79	3,49	3,49	8,34	2,09	0,47
¿El sistema abarca la totalidad del proceso?	1,07	3,49	3,49	3,49	11,54	2,89	-0,33
¿Cómo considera usted el acceso a las funcionalidades del sistema?	1,07	1,07	3,49	3,49	9,12	2,28	0,28
¿Cómo evalúa el tiempo de respuesta del sistema?	0,57	0,79	3,49	3,49	8,34	2,09	0,47
¿Cómo valora el uso que se le puede dar al sistema?	0,79	3,49	3,49	3,49	11,26	2,82	-0,26
<b>Suma</b>	6,33	16,61	24,43	24,43	71,8		
<b>Punto de corte</b>							= N (Prom . Gen.)
	0,9	2,37	3,49	3,49	10,26	2,56	



## Anexo 11: Conclusiones

<b>Conclusiones Generales</b>					
<b>Fases y Momentos</b>	<b>MR</b>	<b>BR</b>	<b>R</b>	<b>PR</b>	<b>NR</b>
¿Esta Aplicación Web favorece el control de los expedientes de administración de la vivienda en la Dirección Provincial de Holguín y agiliza este proceso?	<b>Si</b>	-	-	-	-
¿Cómo aprecia el diseño de las interfaces del sistema?	<b>Si</b>	-	-	-	-
¿Qué le parece el uso de los colores y las imágenes usadas en el sistema?	<b>Si</b>	-	-	-	-
¿El sistema abarca la totalidad del proceso?	<b>Si</b>	-	-	-	-
¿Cómo considera usted el acceso a las funcionalidades del sistema?	<b>Si</b>	-	-	-	-
¿Cómo evalúa el tiempo de respuesta del sistema?	<b>Si</b>	-	-	-	-
¿Cómo valora el uso que se le puede dar al sistema?	<b>Si</b>	-	-	-	-