



UNIVERSIDAD TÉCNICA DE BABAHOYO

FACULTAD DE ADMINISTRACIÓN, FINANZAS E INFORMÁTICA

PROCESO DE TITULACIÓN

MAYO – OCTUBRE 2018

PROPUESTA TECNOLÓGICA

INGENIERÍA EN SISTEMAS

PREVIO A LA OBTENCIÓN DEL TÍTULO DE INGENIERA EN SISTEMAS

TEMA:

**SISTEMA INFORMÁTICO PARA EL COBRO DE LOS PERMISOS DEL CUERPO DE
BOMBEROS DEL CANTÓN BABA.**

EGRESADO:

DAVID JAVIER CEREZO ESMERALDA

TUTOR:

ING. MIGUEL ÁNGEL ZÚÑIGA SÁNCHEZ

AÑO 2018

ÍNDICE GENERAL

TABLA DE CONTENIDO

ÍNDICE GENERAL.....	II
INTRODUCCIÓN.....	1
CAPÍTULO I	3
1. Diagnóstico de Necesidades y Requerimientos	3
1.1. Ámbito de Aplicación	3
1.1.1. Descripción del Contexto y Hechos de Interés.....	3
1.2. Establecimiento de Requerimientos.	4
1.3. Justificación del Requerimiento a Satisfacer.	9
CAPÍTULO II	10
2. Desarrollo del Prototipo.....	10
2.1. Definición del Prototipo Tecnológico.....	10
2.2. Fundamentación Teórica del Prototipo.....	10
2.2.1. Sistema Informático Web.	10
2.2.2. Alojamiento de Registro.	10
2.2.3. Gestión de Datos.	11
2.2.4. Tramite.....	11
2.2.5. Permiso de Bomberos.....	11
2.2.6. Requerimientos.....	11
2.2.7. Requerimientos Operacionales, Funcionales y de Rendimiento por:	12
2.2.8. Diagrama de Casos de Uso.....	14
2.2.9. Definición de Caso de Uso de los Requerimientos.....	21
2.2.10. Prototipo.....	25
2.2.11. Metodología Rad.....	27
2.2.12. UML (Lenguaje Unificado de Modelado)	28
2.2.13. Método Análisis Estructurado.....	29
2.2.14. Php.....	30
2.2.15. Servidor Wamp.....	31
2.2.16. Mysql	32
2.3. Objetivos del Prototipo	34

2.3.1.	General	34
2.3.2.	Específicos.....	34
2.4.	Diseño del Prototipo.	35
2.4.1.	Diagrama Entidad Relación.....	35
2.4.2.	Diagrama de Secuencia.....	36
2.4.3.	Diagramas de Actividad.	37
2.4.4.	Diccionario De Datos.	39
2.5.	Ejecución y/o Ensamblaje del Prototipo.....	48
2.5.1.	Plan de Instalación.....	48
2.5.2.	Plan de Ejecución.....	50
CAPÍTULO III		54
3.	Evaluación Del Prototipo.....	54
3.1.	Plan de Evaluación	54
3.1.1.	Prueba de Caja Blanca.	55
3.1.2.	Prueba de Caja Negra.	57
3.2.	Resultados de la Evaluación	60
CONCLUSIONES.....		62
RECOMENDACIONES		63
BIBLIOGRAFÍA		64

INTRODUCCIÓN

La presente investigación tiene que ver con la elaboración y desarrollo de un sistema informático, para el cobro de los permisos del cuerpo de bomberos del Cantón Baba. Este sistema va a manejar información sobre documentos y datos importantes donde se ven involucrados, software, hardware y el personal que lo va a manipular. La razón por la cual se va a desarrollar la solución informática es la siguiente; falta de control de la información, prevenir la falsificación de los permisos, datos incorrectos, pérdida de tiempo y desperdicio de recursos.

La finalidad de resolver la problemática es para que el personal que manipula este tipo de documentación, tenga información en tiempo oportuno, facilidad de gestión de este trámite, agilidad y rapidez a la hora de presentar un informe y además incentivar a que más procesos se automaticen y estar a la par a como está desarrollada la tecnología en estos tiempos.

El presente proyecto está dividido en tres capítulos. El capítulo número uno: se va a desarrollar el diagnóstico de necesidades y requerimientos, además se va a detallar en la zona donde va a influir y ejecutar el prototipo, aspectos relevantes, la razón por la que se va a realizar el prototipo, los requerimientos; lo que se necesita para poder realizar la solución informática.

En el capítulo número dos: se detalla el desarrollo del prototipo; se define en qué lenguaje se va a desarrollar y qué funciones va a hacer. También se muestra la teoría que respalda la realización del prototipo, los objetivos, el diseño del prototipo, detalles que muestran la ejecución y ensamblaje del prototipo.

En el capítulo número tres: aquí se muestra la evaluación del prototipo; se hace un análisis del rendimiento, estabilidad, eficiencia y manejabilidad del sistema, también se muestran los resultados de la evaluación del prototipo. Y al final se presentan las conclusiones y recomendaciones.

El prototipo que se va a realizar tiene como finalidad de agilizar la obtención del permiso que otorga el cuerpo de bomberos, para así mejorar la eficiencia de dicha institución, por el lado del personal que se encarga de realizar este trámite va a mejorar sustancialmente la manera de controlar y verificar información, también le va a permitir realizar informes más ágilmente; ahorrar tiempo y recursos.

CAPÍTULO I

1. Diagnóstico de Necesidades y Requerimientos

1.1. Ámbito de Aplicación

1.1.1. Descripción del Contexto y Hechos de Interés.

En este mundo globalizado y dominado por los cambios tecnológicos que día a día van avanzando es así, el caso de las instituciones y empresas. Hoy en día utilizan un sinnúmero de soluciones informáticas para mejorar la rapidez con la que se realizan los procesos y el manejo de la información por parte de los usuarios, estas soluciones o más bien llamados sistemas informáticos, son de gran ayuda para mejorar el tiempo de respuesta en que se realizan dichos procesos.

Los sistemas Web son muy factibles para las instituciones y empresas que quieren poner servicios más accesibles a todos, ya que se conoce que solo basta con tener una conexión a internet para acceder a cualquier página Web, así que esto mejora notablemente la cobertura del sistema. Éste es el caso del cuerpo de bomberos del cantón Baba que se ha propuesto mejorar el servicio de otorgamiento de un permiso muy indispensable para cualquier vehículo, empresa o negocio que necesite el permiso de funcionamiento de bomberos para poder funcionar legalmente.

El cuerpo de bomberos de Baba está dirigido por el Cnel. Juan Cercado, el jefe de compañía Gustavo Bajaña que a su vez tiene a cargo a los departamentos de: secretaría, tesorería y personal operativo. La secretaría es dirigida por la Lic. Mildred Olvera; la tesorería está a cargo de la Ing. Anabel Tello, y el personal operativo está conformado por los Bomberos: Pedro Bravo, Darwin Vera, Eduardo Zurita, Darío Bajaña y los choferes: Pedro Santur y Miguel Aguilar.

En el departamento que se basa la investigación es el departamento de tesorería, ahí se realizan el cobro del permiso del cuerpo de Bomberos.

El proceso empieza de la siguiente forma: un usuario que necesite el permiso de bomberos, primero tiene que hacer una solicitud dirigida al Jefe Cnel. Juan Cercado y ahí se le entregan una serie de requisitos que debe reunir para poderle hacer la inspección, una vez hecha la inspección se le evalúa el tipo de establecimiento y según eso se debe realizar un deposito a la cuenta del cuerpo de bombero, ya realizado este pago se verifica y luego se le hace el talonario y se le entrega al usuario el original, una copia se queda en los archivos de la tesorería junto con los demás requisitos. También surgen problemas al realizar reportes anuales porque se demora una considerable cantidad de tiempo al realizarlos y perdida de eficiencia.

En otras instituciones de Bomberos ya existe un sistema similar este es el caso del Cuerpo de Bomberos del Distrito Metropolitano de Quito que en su página web tiene este servicio ya en línea, estando a la vanguardia y acorde a los avances tecnológicos y de llegar de manera más practica a todos los usuarios que quieran realizar este trámite.

1.2. Establecimiento de Requerimientos.

El cuerpo de bomberos del cantón Baba presenta la siguiente estructura organizacional basada en funciones y delegaciones de responsables de cada departamento, según se muestra en la Figura 1.

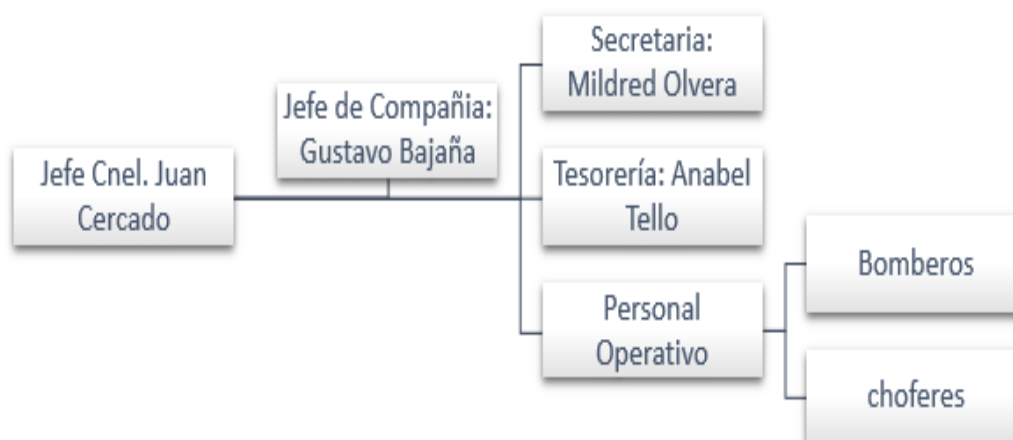


Figura 1. Organigrama del Cuerpo de Bomberos del Cantón Baba.

Fuente: Cuerpo de Bomberos de Baba.

La información que administra el Cuerpo de Baba son datos personales de las personas que realizan los trámites, y demás documentos que debe de adjuntar para que se haga efectivo el permiso, también se contienen las fichas de las inspecciones que se realizan antes de entregar el permiso.

El Cuerpo de bomberos de Baba y su departamento de tesorería se encarga:

- Recibir la solicitud para la petición de sacar el permiso.
- Evaluar y coordinar en que tiempo se puede realizar la inspección.
- Almacenar la información obtenida de la evaluación.
- Entregar la lista de los demás requisitos que debe cumplir el usuario.
- Verificar que cumplan y que todos los requisitos estén debidamente completos.
- Generar el comprobante del permiso de Bomberos.
- Almacenar en un folder todos los trámites realizados.
- Generar reportes ya sea anual o mensual según sea el caso.

El Cuerpo de Bomberos de Baba está constituido por un jefe absoluto, un jefe operativo, la secretaria, la tesorería y el personal operativo (bomberos y choferes) ellos realizan diferentes funciones en la institución la misma que se detalla a continuación en la siguiente Tabla 1.

Tabla 1. *Funciones del Personal del Cuerpo de Bomberos de Baba.*

Apellido y Nombre	Cargo	Función
		Dirigir y coordinar todo las operaciones de todo el personal.
		Gestionar el cronograma de actividades a realizar el personal administrativo.
Cnel. Cercado Juan	Jefe del Cuerpo de Bomberos de Baba	Rendir cuentas a la ciudadanía, sobre las actividades e intervenciones de rescate realizadas por la institución.
		Encargado de vigilar el correcto comportamiento de todo el personal.
Bajaña Gustavo	Jefe de Compañía	

		Asumir el mando cuando el jefe principal no este.
Olvera Mildred	Secretaria	Encargada de elaborar los informes sobre las actividades que realice la institución. Elabora los memos dirigido al personal si este es el caso.
Ing. Tello Anabel	Tesorera	Gestiona y realiza el cobro de los valores del permiso de bomberos. Elabora los reportes anuales o mensuales según sea el caso.
Bravo Pedro Vera Darwin Bravo Mariana Zurita Eduardo Bajaña Dario	Bombero (Personal Operativo)	Realizan la labor de acudir a cualquier emergencia a la que se los llame. Cumplir las disposiciones dispuestas por el Jefe.
		Encargados de conducir los vehículos de

Santur Pedro	Choferes (Personal	Emergencia	de	la
Aguilar Miguel	Operativo)	Institución;	el	carro
		motobomba	y	la
		ambulancia.		

Fuente: Cuerpo de Bomberos de Baba.

El cuerpo de Bomberos de Baba cuenta con las herramientas informáticas: (Word, Excel, Power Point, Nitro Pro 10), internet (correos personales, Zona Wi-Fi), navegadores (Internet Explorer, Mozilla Firefox, Google Chrome), Sistema operativo (Windows 7 Ultimate), componentes físicos (Pc, Cámaras en circuito cerrado, Sistema de Radio Frecuencia, Impresora Multifunciones, router, Flash Memory).

El siguiente SI se va a realizar utilizando la herramienta de desarrollo notepad++ con el lenguaje de programación PHP, el motor de base de datos es MySQL.

Características principales del SI:

- Usa el lenguaje unificado de modelado (UML).
- Utiliza el método de análisis estructurado de las herramientas entidad-relacion.
- Facilidad de Manipular SI.
- Lenguaje de programación PHP.
- Editor de texto Notepad++.
- Gestor de base de datos phpMyAdmin.
- Servidor web Apache Http.
- Se puede ejecutar en los distintos navegadores como Mozilla Firefox, Internet Explorer, Opera, Google Chrome, etc.

- En el servidor Wamp que se alojará y se mostrará las páginas web del SI.
- Protección al ingresar al SI.
- Gestión (guardar, eliminar, modificar y consultar) de información acerca de los permisos.

Al tener el sistema que gestione todo tipo de información requerida por el Cuerpo de bomberos de Baba se mejorará en la eficiencia de la obtención de los datos necesarios.

1.3. Justificación del Requerimiento a Satisfacer.

Se decide llevar a cabo el desarrollo de un Sistema Informático para el cobro del permiso de Cuerpo de Bomberos de Baba, debido a que no se cuenta con un sistemas para la realización de este trámite, y esto hace que se demore en la obtención del mismo.

Para el departamento de tesorería es algo complicado despachar el permiso de bomberos, ya que primero se deben recibir las solicitudes ordenarlas según como han ido llegando y es también difícil obtener datos, si es el caso de realizar una verificación de información, porque se tendría que buscar de forma manual dicha información y eso resulta bastante tedioso.

El Sistema Informático Web sería muy útil y ayudaría al departamento de tesorería en realizar de una forma más ágil y eficaz la manipulación de la información, ya sea para una verificación, búsqueda, informe, etc.

Con la realización del Sistema Informático del cuerpo de Bomberos de Baba se impulsa a que más instituciones y empresas se animen a desarrollar soluciones informáticas para realizar procesos más rápidos, y mejorar en el rendimiento optimización de recursos entre otras cosas.

CAPÍTULO II

2. Desarrollo del Prototipo

2.1. Definición del Prototipo Tecnológico.

Desarrollo del prototipo del Sistema Informático web para el cobro de los permisos del cuerpo de bomberos del cantón baba.

2.2. Fundamentación Teórica del Prototipo.

2.2.1. Sistema Informático Web.

Los sistemas informáticos también llamados (SI), son utilizados para guardar y almacenar información y están compuesto por hardware, software y los seres humanos.

Los sistemas informáticos web son enfocados en la red, los usuarios pueden acceder a través de internet, con la ayuda de los distintos servidores que componen el sistema informático web, a ellos se accede por medio de los navegadores web.

2.2.2. Alojamiento de Registro.

Tipo de alojamiento de los registros: hay varios tipos distintos de alojamiento de los registros que también hay que tener en cuenta y son importantes cuando se quiere seleccionar un sistema de almacenamiento. (CHICANO, 2014)

- Alojamiento tradicional: el alojamiento tradicional de datos se utiliza cuando la organización dispone de en sus instalaciones de equipos destinados al almacenamiento.

La organización autogestión el almacenamiento de sus registros. (CHICANO, 2014)

- Alojamiento web o “web hosting”: el alojamiento web es un tipo de almacenamiento en los que los datos y registros se encuentran almacenado en internet (páginas web, servidores, etc.) y se puede acceder a ellos de modo virtual desde cualquier equipo o dispositivo. Ese tipo de alojamiento puede ser gratuito aunque no es lo habitual y los servicios ofrecidos están bastante limitados. Suelen ser alojamientos de pago en los que se alquila espacio de almacenamiento en un disco virtual o en un sitio web. (CHICANO, 2014).

2.2.3. Gestión de Datos.

Definición de los datos, seguridad e integridad de los datos, recuperación manipulación de los datos y concurrencia de los datos.

Es la agrupación de recursos y personas que se encargan de procesar la información. Implica la recolección, acumulación, ejecución, y recuperación de información. (ARRIGHI, 2004)

2.2.4. Tramite.

Es la realización de una gestión en alguna institución ya sea pública o privada, para obtención de algún certificado del cumplimiento de un requisito.

2.2.5. Permiso de Bomberos.

Un permiso de bomberos es un documento que emite el cuerpo de bomberos de cada ciudad, para el funcionamiento de cualquier local, tienda, negocio, empresa, etc. Y este se debe de renovar anualmente.

2.2.6. Requerimientos.

Un requerimiento determina lo que el sistema debe hacer, que funciones tiene que realizar y propiedades necesarias y deseables. La captura de los requerimientos tiene como principal objetivo

llegar a un entendimiento profundo y no debe hacer el sistema que se está desarrollando. La validación de requerimientos permite comprobar que estos definen el sistema que el cliente busca, asegurando que son completos, exactos consistentes. (VILLADA, 2015)

2.2.7. Requerimientos Operacionales, Funcionales y de Rendimiento por:

2.2.7.1. *Ingreso al Sistema Informático Web.*

2.2.7.1.1. *Ingreso Primario (administrador).*

A este ingreso solo puede acceder el personal autorizado por el Cuerpo de Bomberos del Cantón Baba, debido que podrá gestionar (insertar, eliminar, buscar, modificar, etc.) todo lo que tengan que ver con las peticiones de los usuarios para la obtención del permiso del cuerpo de bomberos; para poder ingresar debe escribir en el login del sistema la palabra ADMINISTRADOR y luego la clave.

2.2.7.1.2. *Ingreso Secundario (contribuyente).*

Este permitirá ingresar a cualquier usuario que quiera realizar el trámite para obtener el permiso de cuerpo de bomberos; para poder ingresar debe de escribir en el login del sistema el usuario y la clave que ya le haya sido asignada por el sistema y si es un usuario que ingresa por primera vez debe de registrarse.

2.2.7.1.3. *Ingreso Secundario (recaudador).*

Este Ingreso permitirá a la persona encargada de cobrar y llevar el control del sistema, para poder ingresar al Sistema debe de escribir en el login la palabra Recaudador y la clave que haya sido asignada una vez que me hay registrado previamente o que el administrador le haya creado el usuario y la contraseña.

2.2.7.2. Ingreso de Datos.

2.2.7.2.1. Ingreso de Datos del Usuario.

Para ingresar los primeros datos de los usuarios el administrador debe de ingresar a las pestañas crear usuario y poner toda la información y asignarle los permisos que según el usuario debe tener.

2.2.7.2.2. Ingreso de los Locales Comerciales.

El acceso para poder ingresar estos datos primero tenemos que acceder como usuario recaudador para poder llenar todos estos campos aquí puede (eliminar, actualizar, y crear uno nuevo).

2.2.7.2.3. Ingreso de las Cuentas por Cobrar.

El acceso para llenar este campo lo debe de hacer usuario recaudador una vez que se haya hecho la petición de obtener el permiso de cuerpo de bomberos.

2.2.7.3. Modificar Datos.

El usuario recaudador realiza la modificación de datos se realiza ya sea en los (Usuario, Locales Comerciales, Cuentas por Cobrar, etc.). Se debe escoger en editar y hacer los cambios, las correcciones que sean necesarios de cambiar.

2.2.7.4. Eliminar Datos.

El usuario recaudador podrá eliminar datos (Usuarios, Locales Comerciales, Cuentas por Cobrar, etc.). Se debe escoger la opción eliminar. Lo que sea necesario de eliminar.

2.2.7.5. *Búsqueda de Datos.*

El usuario recaudador podrá buscar datos (Locales Comerciales, Cuentas por Cobrar, etc.). Se utiliza la opción de búsqueda.

2.2.8. **Diagrama de Casos de Uso**

Un caso de uso es una descripción de las acciones de un sistema desde el punto de vista del usuario. Es una herramienta valiosa dado que es una técnica de aciertos y errores para obtener los requerimientos del sistema, justamente desde el punto de vista del usuario. (CEVALLOS, 2015)

Los diagramas de caso de uso modelan la funcionalidad del sistema usando actores y casos de uso. Los casos de uso son servicios o funciones provistas por el sistema para sus usuarios. (CEVALLOS, 2015)

2.2.8.1. *Símbolos de los Casos de Uso*

- Sistema: El rectángulo representa los límites del sistema que contiene los casos de uso.

Los actores se ubican fuera de los límites del Sistema. (CEVALLOS, 2015)

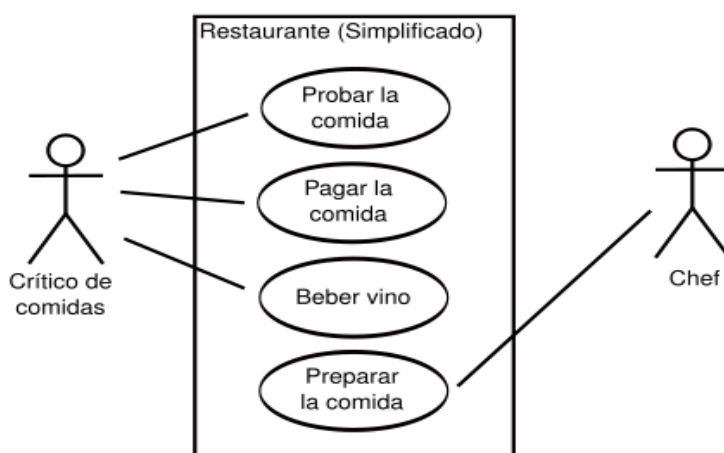


Figura 2: Sistema de Casos de Uso
Fuente: (CEVALLOS, 2015)

- Caso de uso: Se representan con óvalos. La etiqueta en el óvalo indica la función del sistema. (CEVALLOS, 2015)



Figura 3: Función del Sistema
Fuente: (CEVALLOS, 2015)

- Actor: Un diagrama de caso de uso contiene los símbolos del actor y del caso de uso, junto con líneas conectoras. Los actores son similares a las entidades externas; existen fuera del sistema. El término actor se refiere a un rol específico de un usuario del sistema. (CEVALLOS, 2015)

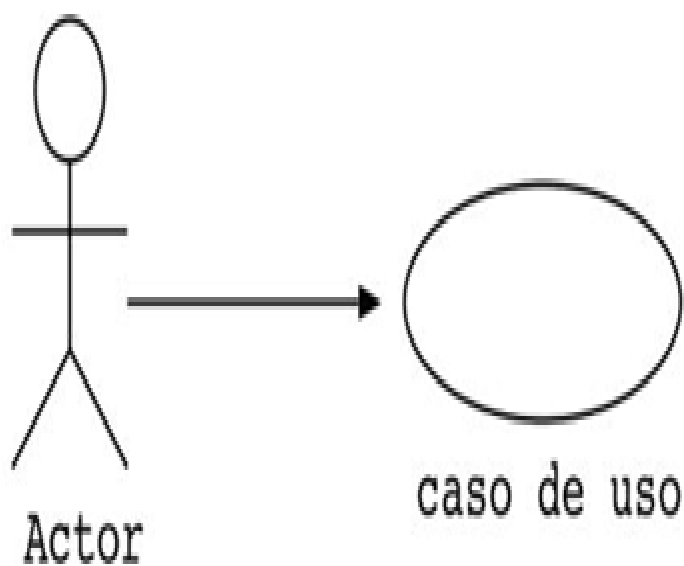


Figura 4: Actor que inicia el caso de uso
Fuente: (CEVALLOS, 2015)

Por ejemplo:

Un actor puede ser un empleado, pero también puede ser un cliente en la tienda de la empresa. Incluso cuando es la misma persona en el mundo real, se representa como dos símbolos distintos en un diagrama de caso de uso, ya que la persona interactúa con el sistema en distintos roles. (CEVALLOS, 2015)

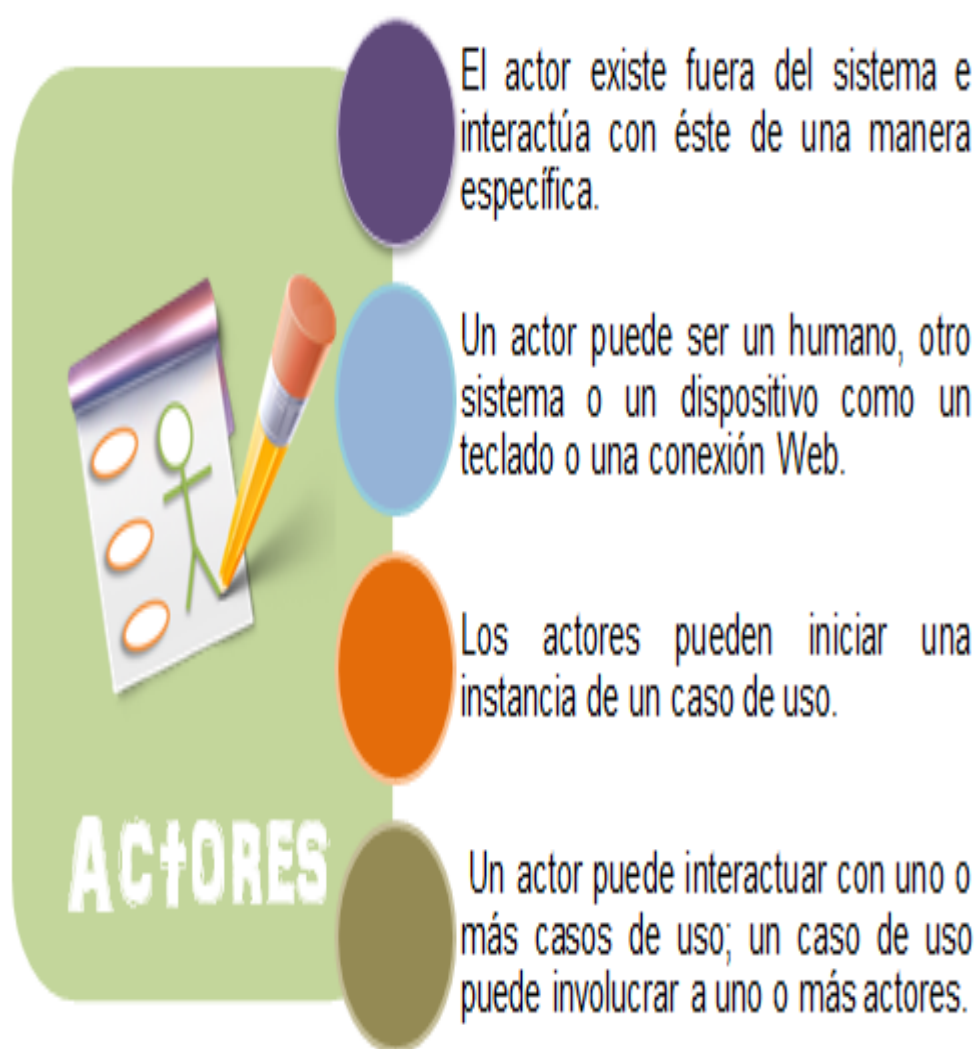


Figura 5: Características de los actores
Fuente: (CEVALLOS, 2015)

2.2.8.2. Relaciones

Las relaciones entre un actor y un caso de uso, se dibujan con una línea simple. Para relaciones entre casos de uso, se utilizan flechas etiquetadas “incluir” o “extender.” Una relación “incluir” indica que un caso de uso es necesitado por otro para poder cumplir una tarea. Una relación “extender” indica opciones alternativas para un cierto caso de uso. (CEVALLOS, 2015)

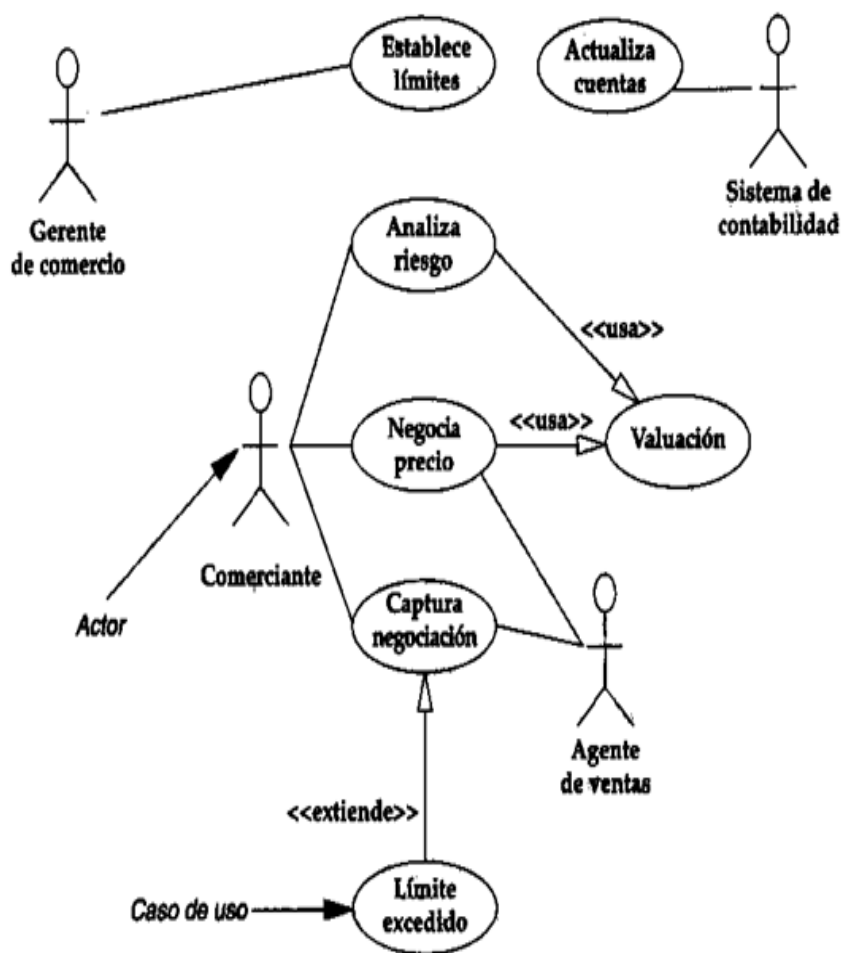


Figura 6: Ejemplo de caso de uso
Fuente: (CEVALLOS, 2015)

2.2.8.3. Relaciones de los Casos de Uso

Las relaciones activas se conocen como relaciones de comportamiento y se utilizan principalmente en los diagramas de casos de uso. Hay cuatro tipos básicos de relaciones de comportamiento: comunica, incluye, extiende y generaliza. (CEVALLOS, 2015)


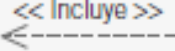
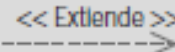

Relación	Símbolo	Significado
Comunica		Para conectar un actor con un caso de uso se utiliza una línea sin puntas de flecha.
Incluye		Un caso de uso contiene un comportamiento común para más de un caso de uso. La flecha apunta al caso de uso común.
Extiende		Un caso de uso distinto maneja las excepciones del caso de uso básico. La flecha apunta del caso de uso extendido al básico.
Generaliza		Una "cosa" de UML es más general que otra "cosa". La flecha apunta a la "cosa" general.

Figura 7: Relaciones entre los casos de uso
Fuente: (CEVALLOS, 2015)

2.2.8.4. Documentación de los casos de uso

Existen dos formas principales de documentar un caso de uso:

- Un diagrama en UML (CEVALLOS, 2015)
- Un documento detallado (CEVALLOS, 2015)

Documentar casos de usos no es una tarea fácil que se pueda dominar de un día para otro, requiere de tiempo, disciplina y experiencia, sin embargo podemos definir una serie de pasos identificables para escribir los casos de uso. (CEVALLOS, 2015)

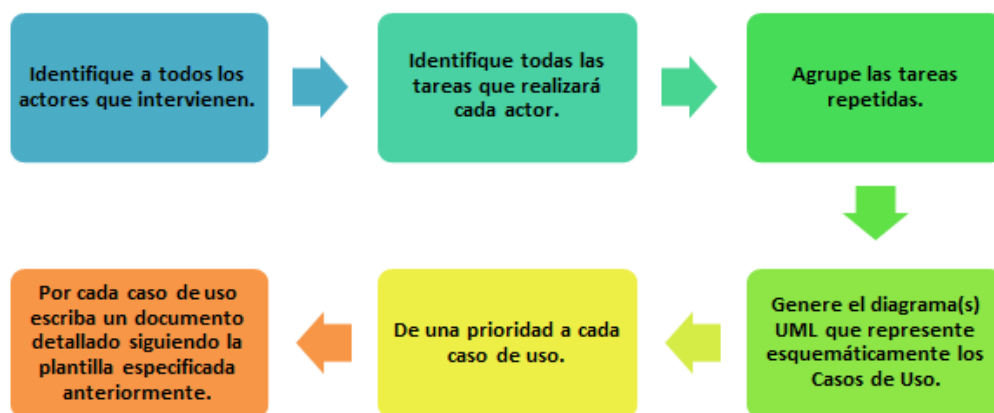


Figura 8: Pasos para la documentación de los casos de uso
Fuente: (CEVALLOS, 2015)

Formato de la documentación de caso de uso para los actores que participen:

ACTOR	CASO DE USO EN LOS QUE PARTICIPA	TIPO	DESCRIPCIÓN
Los actores son aquellos que relacionan directamente con el caso de uso, también se considera actor todo aquello que inicia un caso de uso (por ejemplo una tarea agendada) o responde a un caso de uso (un sistema externo de procesamiento en batch).	Número de casos de uso en los que participe	Primario Secundario	Breve descripción de la función de actor

Figura 9: Documentación de los actores dentro de los casos de uso
Fuente: (CEVALLOS, 2015)

Documentación de un caso de uso:

Nombre del caso de uso	Nombre del Caso de Uso	ID Único: Conf RG 003
Área	Planeación de la conferencia	
Actor(es)	Participante	
Intereados	Personal interesado en el caso de uso	
Nivel	Azul	
Descripción	Permite que el docente envíe la tarea a los estudiantes de la clase	
Evento Desencadenado	El docente utiliza el sistema para enviar la tarea a los estudiantes, para esto introduce en nombre del curso al cual se encuentran matriculados los estudiantes.	
Tipo de Desencadenador	Externo	
	Pasos Realizados (Ruta Principal)	Información Para los Pasos
	1. Se elige entre la lista de cursos asignadas a cual enviar la tarea	ID de usuario, Nombre del Curso
	2. Se asigna una fecha máxima de entrega	Fecha
	3. Se ingresa la Descripción de la tarea	Descripción
	4. Se envía la Información	Confirmación
Precondiciones	El docente ya se registró y cuenta con ID de Usuario y contraseña	
Poscondiciones	La tarea se envió con éxito a los estudiantes	
Suposiciones	El docente tiene ID de Usuario y una contraseña	
Garantía de Éxito	La tarea se envió a los estudiantes y estos recibirán la notificación del plazo de entrega	
Garantía Mínima	El docente pudo ingresar al Sistema	
Requerimientos Cumplidos	Permitir el envío de tarea	
Cuestiones Pendientes	En caso de no realizarse con éxito el envío de la tarea se muestra un mensaje de error	
Prioridad	Alta	

Figura 10: Documentación de los casos de uso
Fuente: (CEVALLOS, 2015)

2.2.9. Definición de Caso de Uso de los Requerimientos.

2.2.9.1. Ingreso del Sistema Informático Web.

2.2.9.1.1. Ingreso Primario (administrador).

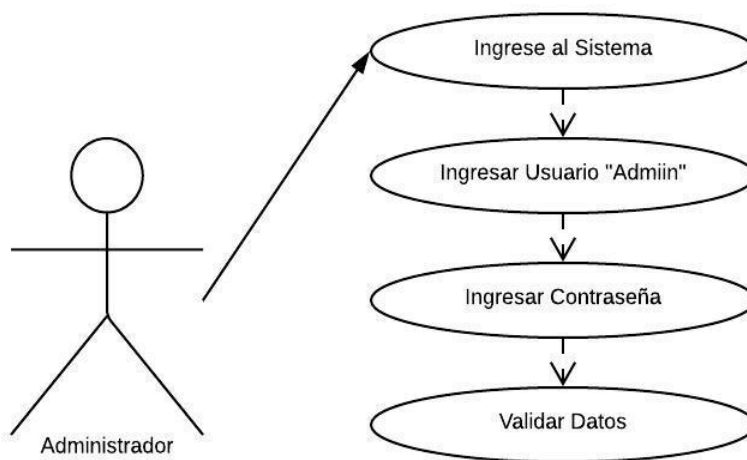


Figura 11: Ingreso Primario Administrador
Fuente: Elaborado por el autor.

2.2.9.1.2. Ingreso Secundario (contribuyente).

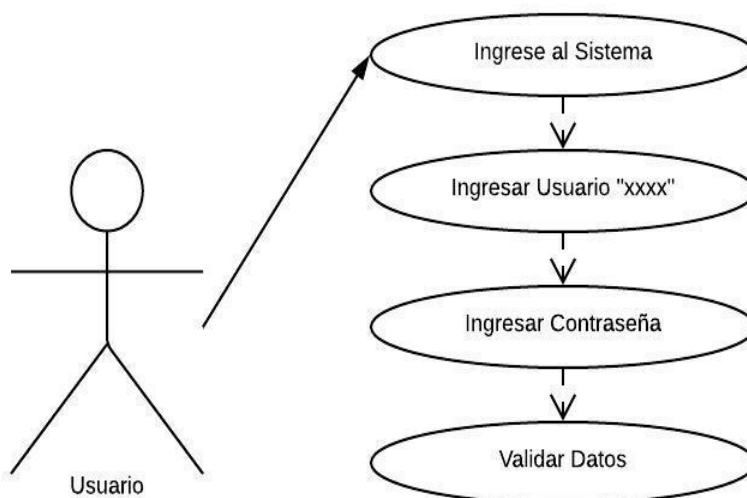


Figura 12: Ingreso Secundario Contribuyente
Fuente: Elaborado por el autor.

2.2.9.1.3. *Ingreso Secundario (recaudador).*

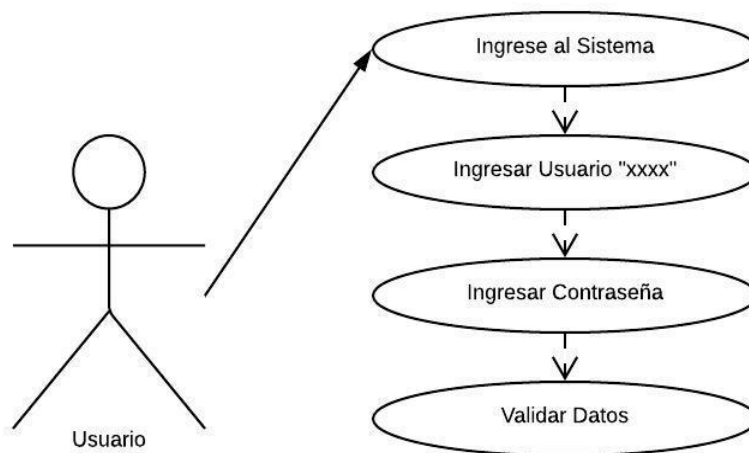


Figura 13: Ingreso Secundario Recaudador
Fuente: Elaborado por el autor.

2.2.9.2. *Ingreso de Datos.*

2.2.9.2.1. *Ingreso de Datos del Usuario.*

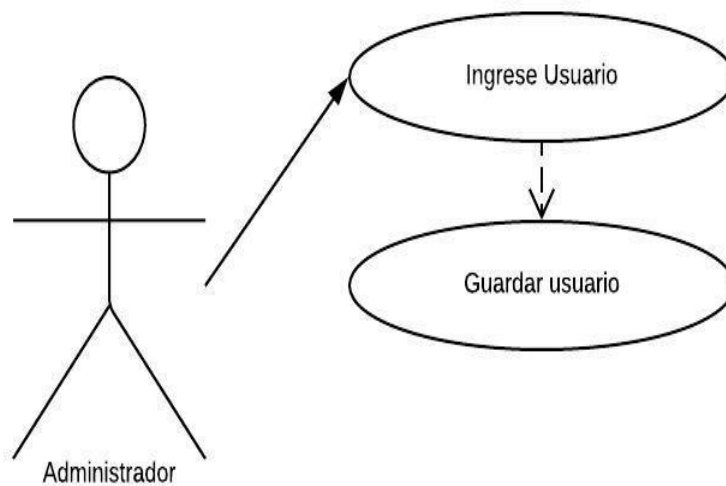


Figura 14: Ingreso datos usuarios
Fuente: Elaborado por el autor.

2.2.9.2.2. *Ingreso de los Locales Comerciales.*

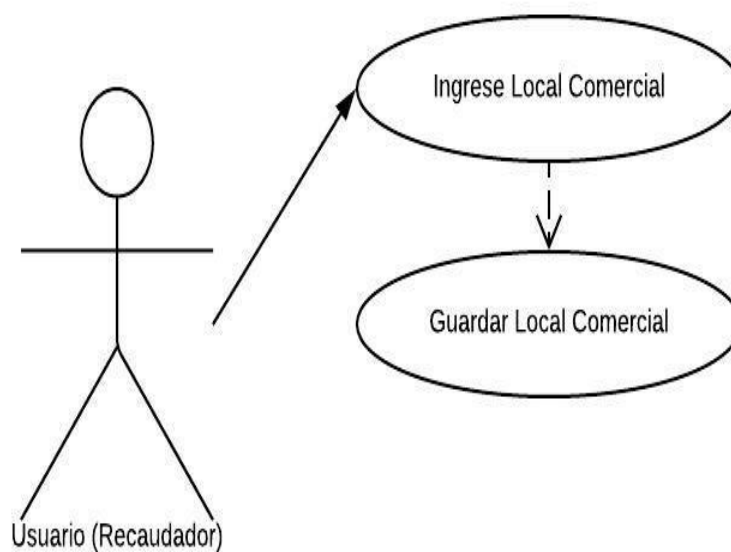


Figura 15: Ingreso locales comerciales
Fuente: Elaborado por el autor.

2.2.9.2.3. *Ingreso de las Cuentas por Cobrar.*

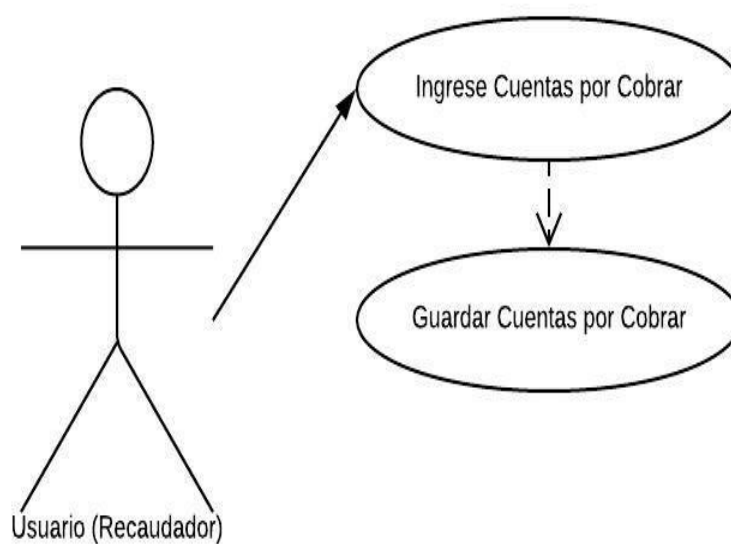


Figura 16: Ingreso cuentas por cobrar
Fuente: Elaborado por el autor.

2.2.9.3. *Modificar Dato*

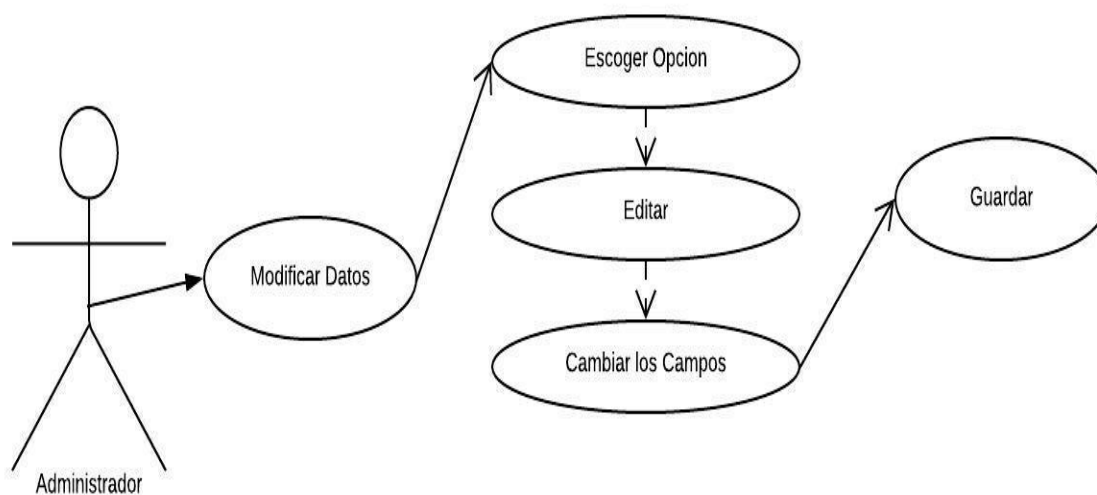


Figura 17: Modificar datos
Fuente: Elaborado por el autor.

2.2.9.4. *Eliminar Datos.*

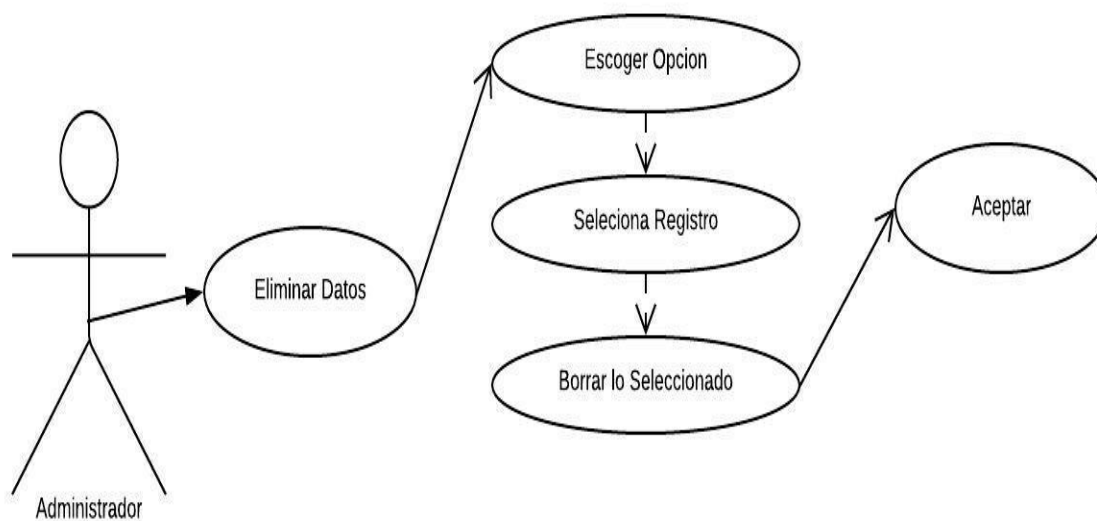


Figura 18: Eliminar datos
Fuente: Elaborado por el autor.

2.2.9.5. *Búsqueda de Datos.*

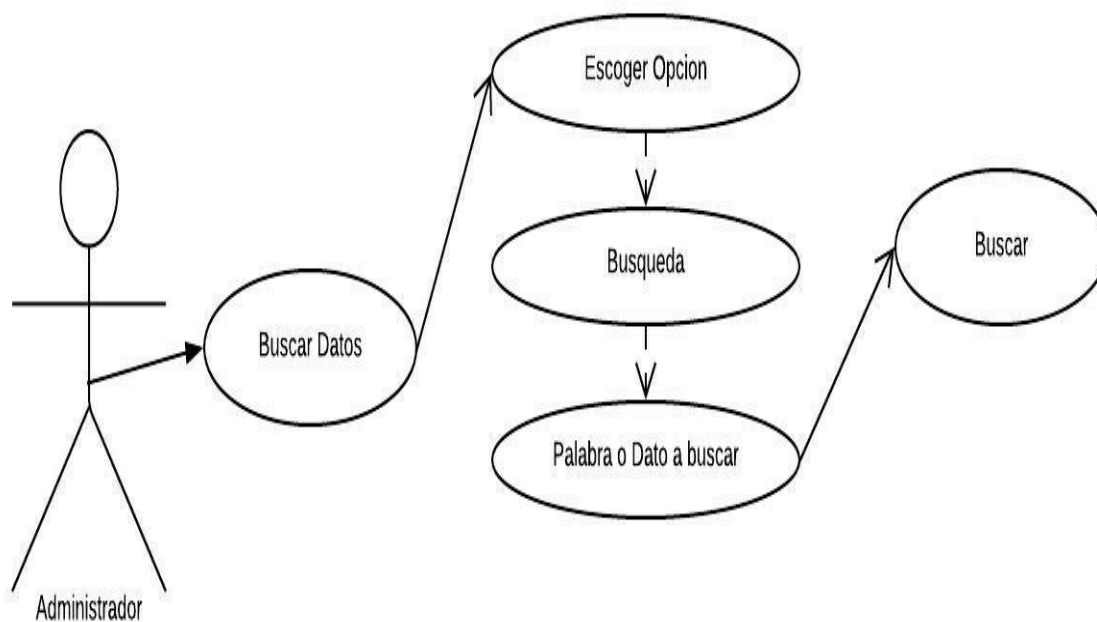


Figura 19: Buscar datos
Fuente: Elaborado por el autor.

2.2.10. **Prototipo**

Un prototipo no es el producto final, ni hay que esperar que sea perfecto ni que tenga una alta fidelidad, pero sí un alto grado de interactividad. (UNIVERSIA, 2015)

Un prototipo es una simulación del producto final. Es como una maqueta interactiva cuyo objetivo principal es probar si el flujo de interacción es el correcto o si hace falta corregirlo. (UNIVERSIA, 2015)

Los prototipos dan vida a cualquier diseño y proporcionan una gran cantidad de información sobre la interacción del usuario en varios niveles. No sólo nos permiten poner a prueba

la viabilidad y la utilidad de nuestros diseños antes de que se comience a programar, sino que también ayudan a descubrir mejoras e innovaciones inesperadas que pueden hacer nuestro proyecto aún mejor. (UNIVERSIA, 2015)

2.2.10.1. ¿Cómo se Crean los Prototipos?

Normalmente se tiende a diseñar prototipos de baja fidelidad porque, siempre y cuando un prototipo simule el flujo de interacción deseado y de la sensación de que estamos usando el producto final, será aceptable. Elementos como el diseño visual pueden quedarse en un segundo plano por el momento. Y, además, se tarda mucho menos tiempo y requiere menos esfuerzo construir un prototipo de baja fidelidad. (UNIVERSIA, 2015)

Hay varias herramientas y técnicas que se pueden utilizar para la construcción de un prototipo. Las más utilizadas y recomendadas son un papel, un lápiz y notas adhesivas para la creación de prototipos en papel; y Adobe Fireworks para la construcción de prototipos de “clickables”. Cada una de estas técnicas tiene su propio conjunto de ventajas y desventajas. (UNIVERSIA, 2015)

2.2.10.2. Papel

En el caso del prototipo de papel, hay varios bocetos que muestran los diferentes estados de la pantalla, fichas que muestran los menús desplegados y notas adhesivas que representan las ventanas. Puedes borrar y hacer cambios de acuerdo a las expectativas de los usuarios y sus comentarios. Las posibilidades son prácticamente infinitas. Pero, un prototipo de papel no ofrece una sensación de "clickabilidad", que te ofrecen los prototipos diseñados con herramientas como Adobe Fireworks, por ejemplo. (UNIVERSIA, 2015)

2.2.11. Metodología Rad

El desarrollo rápido de aplicaciones o RAD (acrónimo en inglés de rapid application development) es un proceso de desarrollo de software, desarrollado inicialmente por James Martin en 1980. El método comprende el desarrollo interactivo, la construcción de prototipos y el uso de utilidades CASE (Computer Aided Software Engineering). Tradicionalmente, el desarrollo rápido de aplicaciones tiende a englobar también la usabilidad, utilidad y la rapidez de ejecución. (OLIVA & CARRILLO, s.f.)

Hoy en día se suele utilizar para referirnos al desarrollo rápido de interfaces gráficas de usuario tales como Glade, o entornos de desarrollo integrado completos. Algunas de las plataformas más conocidas son Visual Studio, Lazarus, Gambas, Delphi, Foxpro, Anjuta, Game Maker, Velneo o Clarion. En el área de la autoría multimedia, software como Neosoft Neoboo y MediaChance Multimedia Builder proveen plataformas de desarrollo rápido de aplicaciones, dentro de ciertos límites. (OLIVA & CARRILLO, s.f.)

2.2.11.1. Fases Del RAD

- **Modelado de gestión:** el flujo de información entre las funciones de gestión se modela de forma que responda a las siguientes preguntas: ¿Qué información conduce el proceso de gestión? ¿Qué información se genera? ¿Quién la genera? ¿A dónde va la información? ¿Quién la proceso? (OLIVA & CARRILLO, s.f.)
- **Modelado de datos:** el flujo de información definido como parte de la fase de modelado de gestión se refina como un conjunto de objetos de datos necesarios para apoyar la empresa. Se definen las características (llamadas atributos) de cada uno de los objetos y las relaciones entre estos objetos. (OLIVA & CARRILLO, s.f.)

- **Modelado de proceso:** los objetos de datos definidos en la fase de modelado de datos quedan transformados para lograr el flujo de información necesario para implementar una función de gestión. Las descripciones del proceso se crean para añadir, modificar, suprimir, o recuperar un objeto de datos. Es la comunicación entre los objetos. (OLIVA & CARRILLO, s.f.)
- **Generación de aplicaciones:** El DRA asume la utilización de técnicas de cuarta generación. En lugar de crear software con lenguajes de programación de tercera generación, el proceso DRA trabaja para volver a utilizar componentes de programas ya existentes (cuando es posible) o a crear componentes reutilizables (cuando sea necesario). En todos los casos se utilizan herramientas automáticas para facilitar la construcción del software. (OLIVA & CARRILLO, s.f.)
- **Pruebas de entrega:** Como el proceso DRA enfatiza la reutilización, ya se han comprobado muchos de los componentes de los programas. Esto reduce tiempo de pruebas. Sin embargo, se deben probar todos los componentes nuevos y se deben ejercitar todas las interfaces a fondo. (OLIVA & CARRILLO, s.f.)

2.2.12. UML (Lenguaje Unificado de Modelado)

UML son las siglas de “Unified Modeling Language” o “Lenguaje Unificado de Modelado”. Se trata de un estándar que se ha adoptado a nivel internacional por numerosos organismos y empresas para crear esquemas, diagramas y documentación relativa a los desarrollos de software (programas informáticos). (KRALL, 2018)

2.2.12.1. ¿Qué es y Para Qué Sirve Uml?

El término “lenguaje” ha generado bastante confusión respecto a lo que es UML. En realidad el término lenguaje quizás no es el más apropiado, ya que no es un lenguaje propiamente dicho,

sino una serie de normas y estándares gráficos respecto a cómo se deben representar los esquemas relativos al software. Mucha gente piensa por confusión que UML es un lenguaje de programación y esta idea es errónea: UML no es un lenguaje de programación. Como decimos, UML son una serie de normas y estándares que dicen cómo se debe representar algo. (KRALL, 2018)

UML es una herramienta propia de personas que tienen conocimientos relativamente avanzados de programación y es frecuentemente usada por analistas funcionales (aquellos que definen qué debe hacer un programa sin entrar a escribir el código) y analistas-programadores (aquellos que dado un problema, lo estudian y escriben el código informático para resolverlo en un lenguaje como Java, C#, Python o cualquier otro). Por tanto si estás dando tus primeros pasos en programación, te recomendaríamos que te olvides de UML hasta que tengas unos conocimientos mínimos como uso de condicionales, bucles, y conocimiento de la programación orientada a objetos. Esto es solo una recomendación, en realidad prácticamente cualquier persona puede usar UML, incluso podría usarse para realizar esquemas o documentación de procesos que no tengan que ver con la informática. (KRALL, 2018)

El UML es popular entre programadores, pero no suele ser usado por desarrolladores de bases de datos. Una razón es sencillamente que los creadores de UML no se enfocaron en las bases de datos. A pesar de ello, el UML es efectivo para el modelado de alto nivel de datos conceptuales y se puede usar en diferentes tipos de diagramas UML. Puedes encontrar información sobre la multidimensionalidad de un modelo de clases orientado a objetos en una base de datos relacional en este artículo sobre Modelado de bases de datos en UML. (Lucid Software Inc., 2018)

2.2.13. Método Análisis Estructurado

El análisis estructurado es el método más usado para el modelado de requisitos, utiliza el modelo de datos y el modelo de flujos para crear la base de un adecuado modelo de análisis.

Utilizando el diagrama entidad-relación, el ingeniero del software crea una representación de todos los objetos de datos que son importantes para el sistema. Los sistemas de datos y flujo de control son la base de representación de la transformación de datos y control. Después, crean un modelo de comportamiento usando el diagrama de transición de estados y un modelo de contenido de los datos con un diccionario de datos. (PRESSMAN, 2001)

2.2.14. Php

Lenguaje de programación, interpretado, diseñado originalmente para la creación de Páginas web dinámicas. Es usado principalmente en interpretación del lado del servidor (server-side scripting) pero actualmente puede ser utilizado desde una interfaz de línea de comandos o en la creación de otros tipos de programas incluyendo aplicaciones con interfaz gráfica usando las bibliotecas Qt o GTK+. (ECURED, 2018)

PHP es un Acrónimo recursivo que significa *PHP Hypertext Pre-processor* (inicialmente PHP Tools, o, *Personal Home Page Tools*). Fue creado originalmente por Rasmus Lerdorf en 1994; sin embargo la implementación principal de PHP es producida ahora por The PHP Group y sirve como el estándar *de facto* para PHP al no haber una especificación formal. Publicado bajo la PHP License, la Free Software Foundation considera esta licencia como software libre. (ECURED, 2018)

2.2.14.1. Lenguaje de Programación Php

Es un lenguaje de programación de dominio específico, es decir, su alcance se extiende a un campo de actuación que es el desarrollo web, aunque existan variantes como PHP-GTK. Su

propósito principal es de implementar soluciones web veloces, simples y eficientes. Sus principales características: (ARIAS, 2017)

- Velocidad y robustez. (ARIAS, 2017)
- Estructurado y orientado a objetos. (ARIAS, 2017)
- Portabilidad - independencia de plataforma. (ARIAS, 2017)
- Mecanografiado dinámico. (ARIAS, 2017)
- Sintaxis similar a C/C++ y Perl. (ARIAS, 2017)
- Open-source. (ARIAS, 2017)

2.2.15. Servidor Wamp

WAMP es un acrónimo que describe un sistema de infraestructura de internet que usa las siguientes herramientas: **Windows** como sistema operativo; **Apache** como servidor web; **MySQL** como gestor de base de datos; **PHP, Perl o Python** como lenguajes de programación.

Esta herramienta te permite crear aplicaciones de manera local y probarlas para luego poderlas subir a internet.

2.2.15.1. Notepad++

Notepad++ es un programa para editar código fuente de cualquier lenguaje de programación. Como tiene soporte para una gran cantidad de lenguajes, interesará no sólo a los desarrolladores de webs, sino en general a toda la comunidad de programadores. (ALVAREZ, 2008)

Es de estos editores que ofrecen ayudas muy útiles para "tirar líneas de código", como resaltado de colores, posibilidad de editar varios documentos a la vez, menús contextuales, auto-completar código, etc. Todo un regalo para los programadores, ya que además es gratuito. (ALVAREZ, 2008)

2.2.16. Mysql

MySQL es un motor de base de datos relacional (RDBMS) de código abierto, originalmente creado por la empresa MySQL AB y actualmente de propiedad de Oracle Corporation. (RAMÍREZ, 2018)

2.2.16.1. *¿Qué es y Para Qué Sirve Mysql?*

MySQL es una de las bases de datos más populares en el mundo, especialmente para desarrollos web. Fue desarrollada originalmente en los lenguajes C y C++. (RAMÍREZ, 2018)

Como administrador de bases de datos relacional, MySQL almacena los datos en forma de tablas estructuradas, con campos, índices, llaves foráneas e integridad referencial en la actualización o borrado en cascada. Usa el lenguaje SQL para las diferentes transacciones de datos. (RAMÍREZ, 2018)

Se puede instalar prácticamente en cualquier sistema operativo: Unix, Linux, Windows, MacOS; solo o acompañado de otros programas del entorno de programación como el servidor web Apache, PHP y Phpmyadmin; en las distribuciones WAMP, LAMP o XAMPP. (RAMÍREZ, 2018)

Algunas características de MySQL. (RAMÍREZ, 2018)

- Autenticación de usuarios con permisos específicos para ciertas bases de datos, atadas a las direcciones IP de origen. (RAMÍREZ, 2018)

- Gestión de memoria y cache para una cantidad determinada de consultas o transacciones simultáneas. (RAMÍREZ, 2018)
- Conectores para integración en ambientes PHP, Perl, Python, ODBC. (RAMÍREZ, 2018)
- Replicación transaccional en línea de la base de datos, a otra base paralela. (RAMÍREZ, 2018)
- Monitoreo de usuarios, transacciones, uso de memoria y de procesos. (RAMÍREZ, 2018)
- Uso de triggers o disparadores para actuar sobre creación, edición o borrado de registros. (RAMÍREZ, 2018)
- Uso automático de valores autonuméricos en las tablas. (RAMÍREZ, 2018)
- Se pueden crear vistas, procedimientos almacenados y funciones. (RAMÍREZ, 2018)
- Posee funciones de chequeo del desempeño de consultas y sus índices, cómo también de los parámetros de memoria, caché, transacciones, etc. Podemos encontrar un consejero de desempeño en Phpmyadmin. (RAMÍREZ, 2018)
- Se puede administrar desde consola, phpmyadmin, o desde programas externos como MySQLfront o Sequel (para MacOS) mediante conexión por el puerto TCP 3306. (RAMÍREZ, 2018)
- Puede manejar millones de registros en una sola tabla. (RAMÍREZ, 2018)
- Gestiona el bloqueo de tablas para evitar conflictos de transacciones simultáneas. (RAMÍREZ, 2018)
- Se pueden importar o exportar datos o la estructura misma de la base, sus tablas, índices, etc. (RAMÍREZ, 2018)

2.3. Objetivos del Prototipo

2.3.1. General

Desarrollar un sistema informático para el cobro del permiso de bomberos para el Cuerpo de Bomberos del Cantón Baba.

2.3.2. Específicos

- Realizar una aplicación sencilla y fácil de utilizar para la recaudadora del Cuerpo de Bomberos del Cantón Baba.
- Disminuir la cantidad de tiempo en que se emite el permiso de bomberos.
- Fomentar el uso de las nuevas tecnologías en el Cuerpo de Bomberos del Cantón Baba, y de la ciudadanía en general.

2.4. Diseño del Prototipo.

2.4.1. Diagrama Entidad Relación

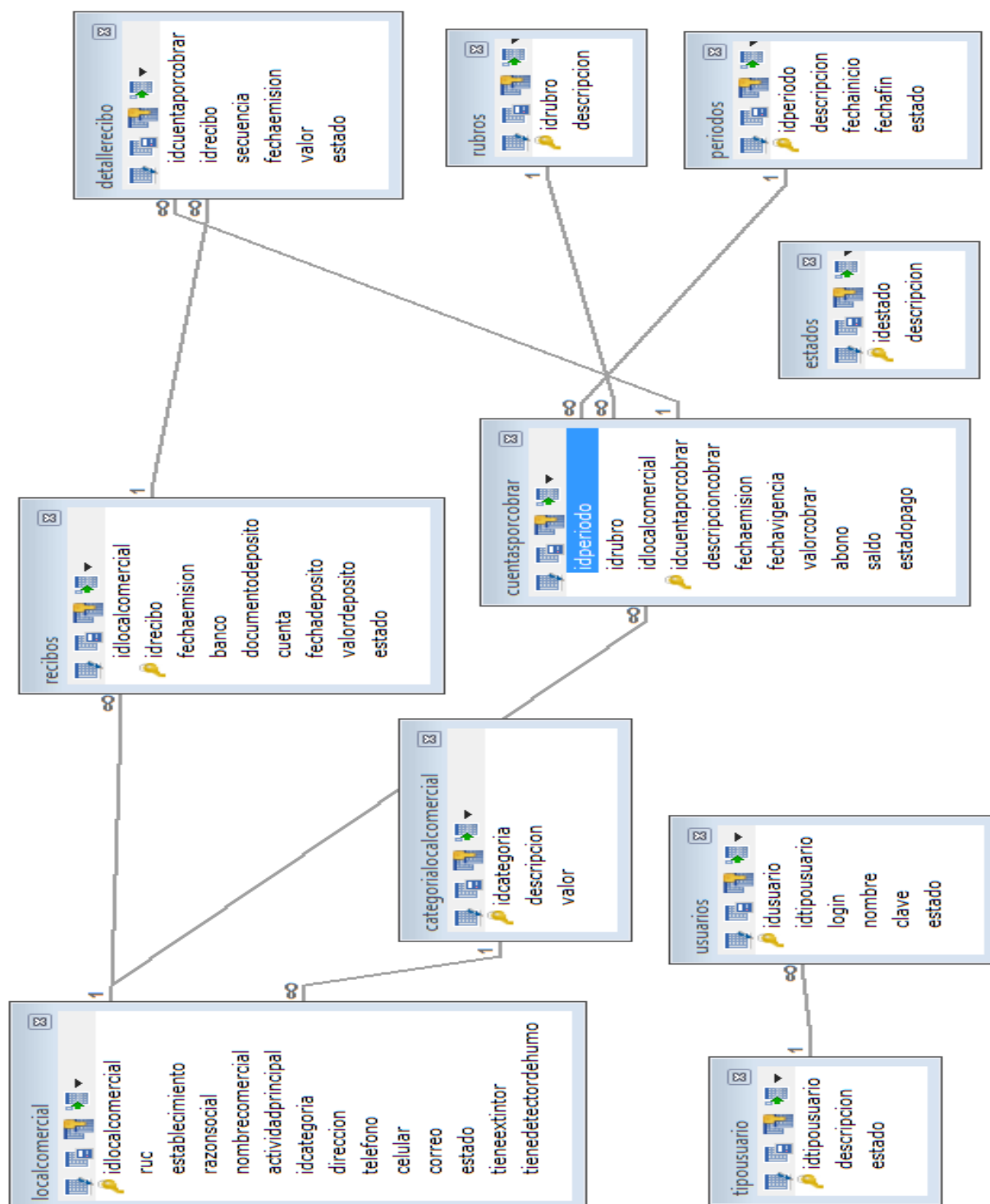


Figura 20: Diagrama entidad - relación
Fuente: Elaborado por el autor.

2.4.2. Diagrama de Secuencia

DIAGRAMA DE SECUENCIA

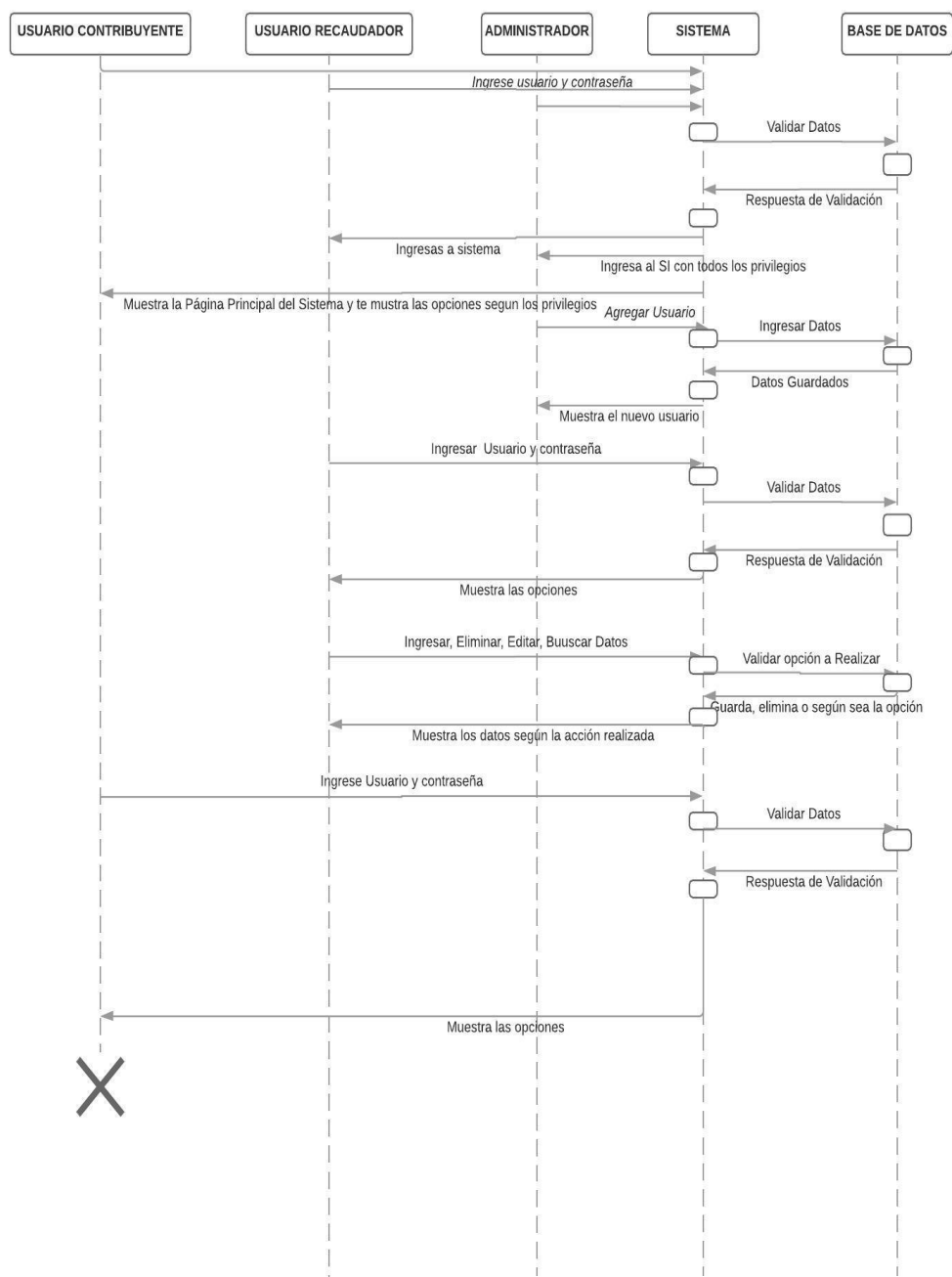


Figura 21: Diagrama de secuencia
Fuente: Elaborado por el autor.

2.4.3. Diagramas de Actividad.

Ingreso al Sistema Informático

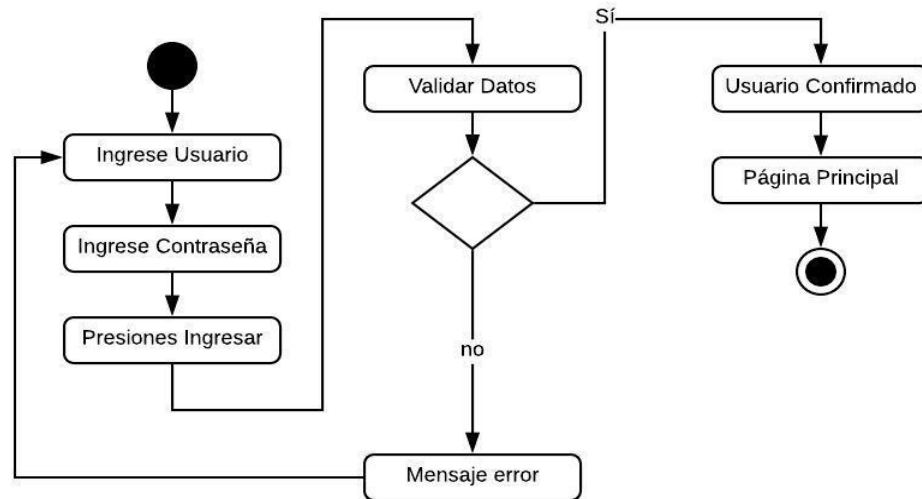


Figura 22: Ingreso al SI.
Fuente: Elaborado por el autor.

Ingreso del Administrador de un nuevo Usuario

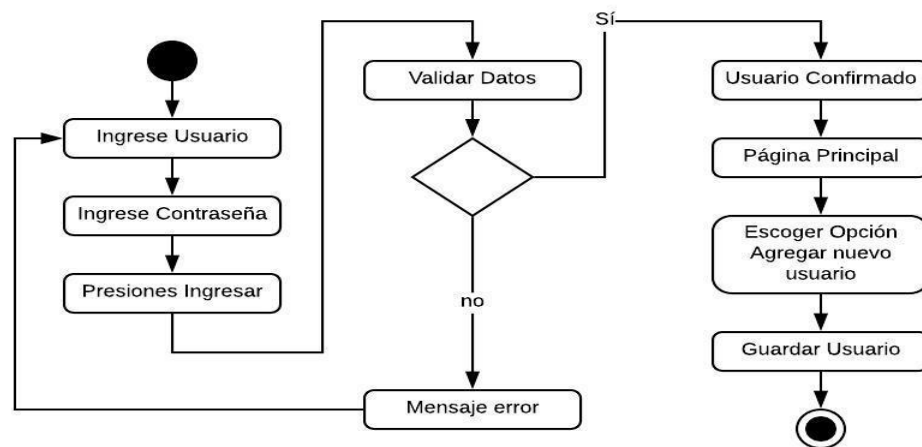


Figura 23: Ingreso al SI como administrador.
Fuente: Elaborado por el autor.

Ingreso de usuario recaudador y acciones que puede realizar

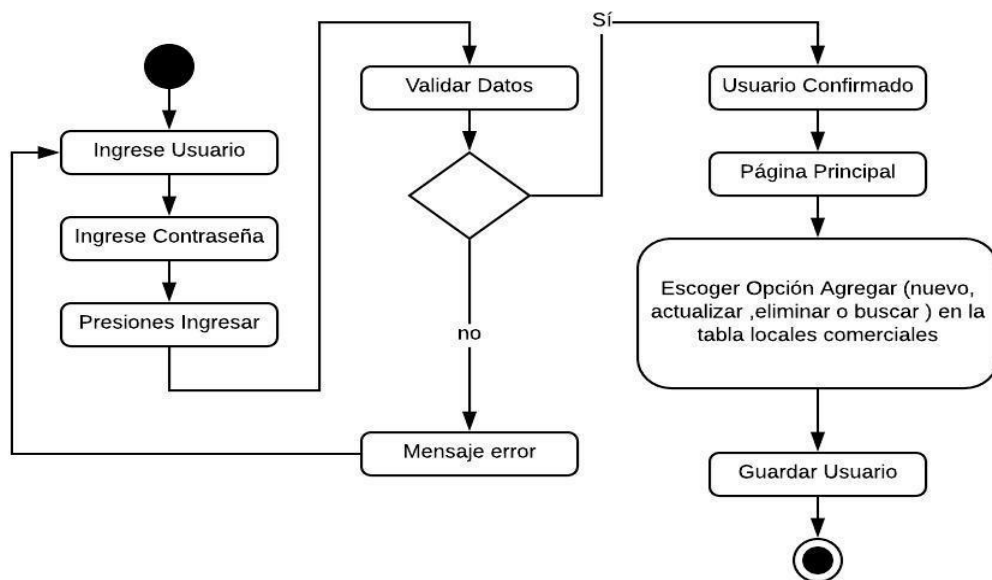


Figura 24: Ingreso al SI como usuario recaudador.
Fuente: Elaborado por el autor.

Ingreso de usuario contribuyente y acciones que puede realizar

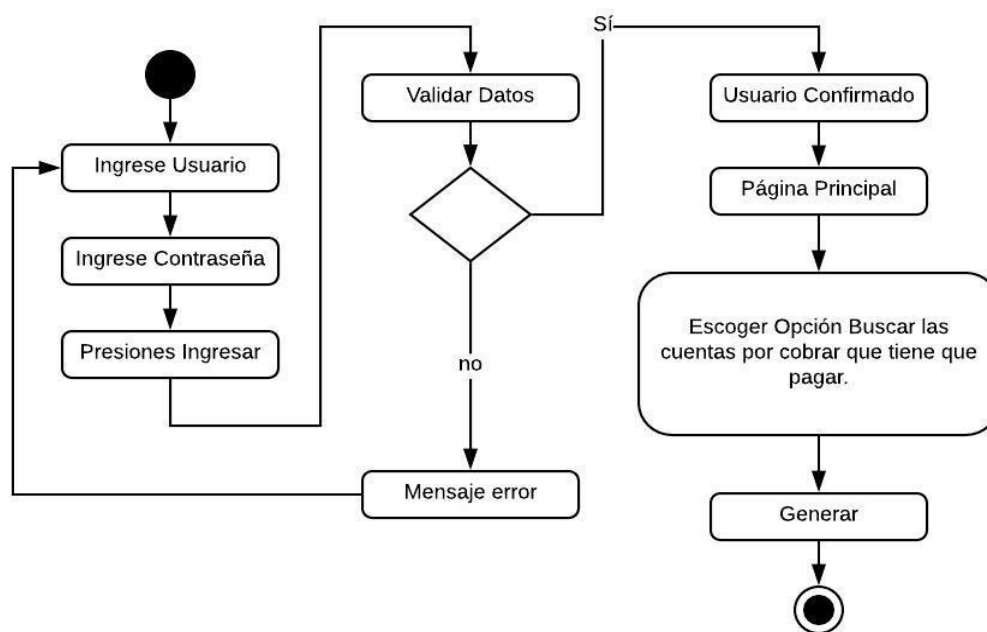


Figura 25: Ingreso al SI como usuario contribuyente.
Fuente: Elaborado por el autor.

2.4.4. Diccionario De Datos.

Aquí se muestran todas las tablas y los campos se detallan uno a uno la descripción. Enlaces entre tablas, tipo de datos, etc.

Tabla 2. Descripción de la tabla *tipousuario*.

Columna	Tipo	Nulo	Predeterminado	Enlaces a	Comentarios
idtipodeusuario (PRIMARY KEY)	int(20)	NOT NULL			Identifica el tipo de usuario.
descripcion	varchar(20)	NOT NULL			Descripción del tipo de usuario.
estado	char(1)	NOT NULL	DEFAULT 'A'		Estado del usuario.

Fuente: Elaborado por el autor.

Tabla 3. Descripción de la tabla *usuarios*.

Columna	Tipo	Nulo	Predetermi nado	Enlaces a	Comentarios
idusuario (PRIMARY KEY)	int(11)	NOT NULL	AUTO_INC REMENT		Identifica el usuario.
idtipodeusuario (FOREIGN KEY)	int(20)	NOT NULL		tipousuario-> idtipodeusuario	Identifica el tipo de usuario.

login	varchar (20)	NOT NULL			
nombre	varchar (80)	NOT NULL			Descripción del nombre del usuario.
clave	varchar (20)	NOT NULL			Clave del usuario.

Fuente: Elaborado por el autor.

Tabla 4. Descripción de la tabla estados.

Columna	Tipo	Nulo	Predeterminado	Enlaces a	Comentarios
Idestado (PRIMARY KEY)	char(1)	NOT NULL			Identifica el estado.
descripcion	varchar(10)	NOT NULL			Descripción del estado.

Fuente: Elaborado por el autor.

Tabla 5. Descripción de la tabla categorialocalcomercial.

Columna	Tipo	Nulo	Predeterminado	Enlaces a	Comentarios
idcategoria (PRIMARY KEY)	int(20)	NOT NULL	AUTO_INCREMENT	a	Identifica la clave principal de la categoría.

descripción	varchar(50)	NOT NULL			Descripción de categoría local comercial.
valor	double	NOT NULL	DEFAULT '0'		Valor que identifica la categoría del local comercial.

Fuente: Elaborado por el autor.

Tabla 6. Descripción de la tabla localcomercial.

Columna	Tipo	Nulo	Predeter minado	Enlaces a	Comentarios
idlocalcomercial (PRIMARY KEY)	int(11)	NOT NULL			Identifica el local comercial.
idcategoria (FOREIGN KEY)	int(20)	NOT NULL		categoria- alcomercial- >idcategoria	Identifica la clave principal de la categoría.
ruc	varchar (13)	NOT NULL			Identifica el RUC del local comercial.
establecimiento	varchar (3)	NOT NULL			Número del establecimiento.
razonsocial	varchar (80)	NOT NULL			Nombre del titular del comercial.

nombrecomercial	varchar (80)	NOT NULL			Nombre del comercial.
actividadprincipal	varchar (200)	NOT NULL			Descripción a que se dedica el comercial.
direccion	varchar (100)	NOT NULL			Dirección del comercial.
telefono	varchar (10)	DEFAULT T NULL			Teléfono del comercial.
celular	varchar (10)	DEFAULT T NULL			Celular del comercial.
correo	varchar (80)	DEFAULT T NULL			Correo del comercial.
estado	char(1)	NOT NULL	DEFAULT LT 'A'		Estado del comercial.
tieneextintor	char(1)	NOT NULL	DEFAULT LT 'S'		Tiene extintor s/n
tienedetectordehumo	char(1)	NOT NULL	DEFAULT LT 'S'		Tiene detector de humo s/n

Fuente: Elaborado por el autor.

Tabla 7. Descripción de la tabla recibos.

Columna	Tipo	Nulo	Predeter minado	Enlaces a	Comentarios
idrecibo (PRIMARY KEY)	decimal (10,0)	NOT NULL			Identifica el recibo.
idlocalcomercial (FOREIGN KEY)	int(11)	NOT NULL		localcomercial -> idlocalcomercial	Identifica el local comercial.
fechaemision	date	NOT NULL			Fecha en la que se emite el recibo.
banco	varchar (50)	DEFAUL T NULL			Banco donde se emite el recibo.
documentodeposito	varchar (10)	DEFAUL T NULL			Documento del depósito.
cuenta	varchar (10)	DEFAUL T NULL			Número de cuenta.
fechadeposito	varchar (10)	DEFAUL T NULL			Fecha en la que se hace el depósito.
valordeposito	double	NOT NULL	DEFAU LT '0'		Valor del depósito.
estado	char(1)	NOT NULL	DEFAU LT 'A'		Estado del recibo.

Fuente: Elaborado por el autor.

Tabla 8. Descripción de la tabla periodos.

Columna	Tipo	Nulo	Predeterminado	Enlaces a	Comentarios
idperiodo (PRIMARY KEY)	int(11)	NOT NULL	AUTO_INCREMENT		Identifica el periodo.
descripcion	varchar(80)	NOT NULL			Descripción del periodo.
fechainicio	date	NOT NULL			Fecha de inicio del periodo
fechafin	date	NOT NULL			Fecha de fin del periodo
estado	char(1)	NOT NULL	DEFAULT 'A'		Estado del periodo.

Fuente: Elaborado por el autor.

Tabla 9. Descripción de la tabla rubros.

Columna	Tipo	Nulo	Predeterminado	Enlaces a	Comentarios
idrubro (PRIMARY KEY)	int(11)	NOT NULL	AUTO_INCREMENT	a	Identifica la clave principal del rubro.

descripcion	varchar(50)	NOT NULL			Descripción del rubro.
-------------	-------------	-------------	--	--	------------------------

Fuente: Elaborado por el autor.

Tabla 10. Descripción de la tabla *cuentasporcobrar*.

Columna	Tipo	Nulo	Predeterminado	Enlaces a	Comentarios
idcuentaporcobrar (PRIMARY KEY)	decimal(10, 0)	NOT NULL			Identifica la clave principal de las cuentas por cobrar.
Idperiodo (FOREIGN KEY)	int(11)	NOT NULL		Periodos-> Idperiodo	Identifica el periodo.
Idrubro (FOREIGN KEY)	int(11)	NOT NULL		Rubros-> Idrubro	Identifica la clave principal del rubro.
Idlocalcomercial (FOREIGN KEY)	int(11)	NOT NULL		Localcomercia l-> Idlocalcomerci al	Identifica el local comercial.
descripcioncobrar	varchar(50)	NOT NULL	DEFAULT NULL		Descripción de las cuentas por cobrar.

fechaemision	date	NOT NULL			Fecha de emisión de las cuentas por cobrar.
fechavigencia	date	NOT NULL			Fecha de vigencia de las cuentas por cobrar.
valorcobrar	double	NOT NULL			Volar a cobrar de las cuentas por cobrar.
abono	double	NOT NULL	DEFAULT '0'		Cantidad abonada en las cuentas por cobrar.
saldo	double	NOT NULL			Saldo de las cuentas por cobrar.
estadopago	char(1)	NOT NULL	DEFAULT 'P'		Estado de las cuentas por cobrar.

Fuente: Elaborado por el autor.

Tabla 11. Descripción de la tabla detallerecibo.

Columna	Tipo	Nulo	Predeterminado	Enlaces a	Comentarios
idcuentaporcobrar (FOREIGN KEY)	decimal(10, 0)	NOT NULL		Cuentasporc obrar- >idcuentapor cobrar	Identifica la clave principal de las cuentas por cobrar.
idrecibo (FOREIGN KEY)	decimal(10, 0)	NOT NULL		Recibos- >idrecibo	Identifica el recibo.
secuencia	int(20)	NOT NULL			Identifica la secuencia
fechaemision	date	NOT NULL			Fecha en que se emite
valor	double	NOT NULL	DEFAULT '0'		Valor a pagar.
estado	char(1)	NOT NULL	DEFAULT 'A'		Estado del detalle recibo.

Fuente: Elaborado por el autor.

2.5. Ejecución y/o Ensamblaje del Prototipo.

2.5.1. Plan de Instalación.

Para la instalación del Sistema Informático cuando lo tenemos de forma local no necesitamos algún sistema operativo específico solo basta tener instalado el servidor Wamp y si lo hemos desarrollado en otro ordenador se debe pegar el proyecto en la carpeta C:\wamp\www.

Instalación del Wamp server primero tienes que descargarlo de la página oficial en español. <http://www.wampserver.es/>.

Una vez descargada la aplicación la ejecutamos y hacemos todos los pasos de la instalación.

Primero ejecutamos el archivo que se ha descargado de la página oficial de Wamp server

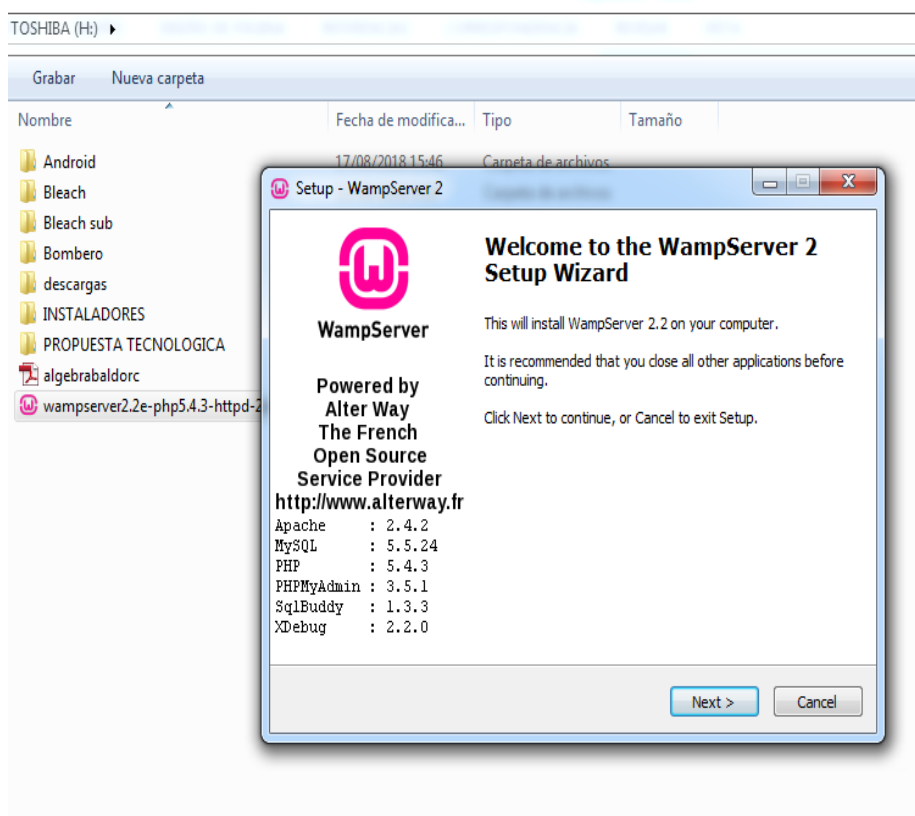


Figura 26: Instalación del Wamp server.
Fuente: Elaborado por el autor.

Una vez instalado el Wamp server iniciamos los servicios.

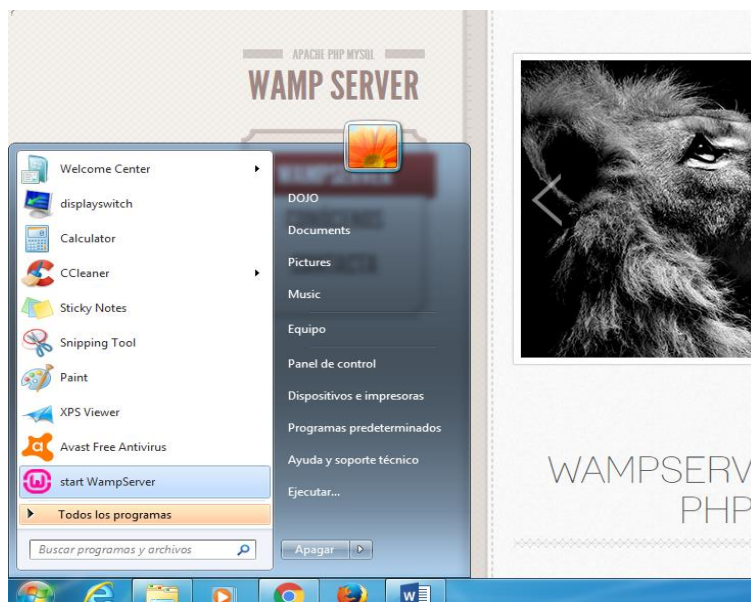


Figura 27: Inicio de servicios de Wamp server.

Fuente: Elaborado por el autor.

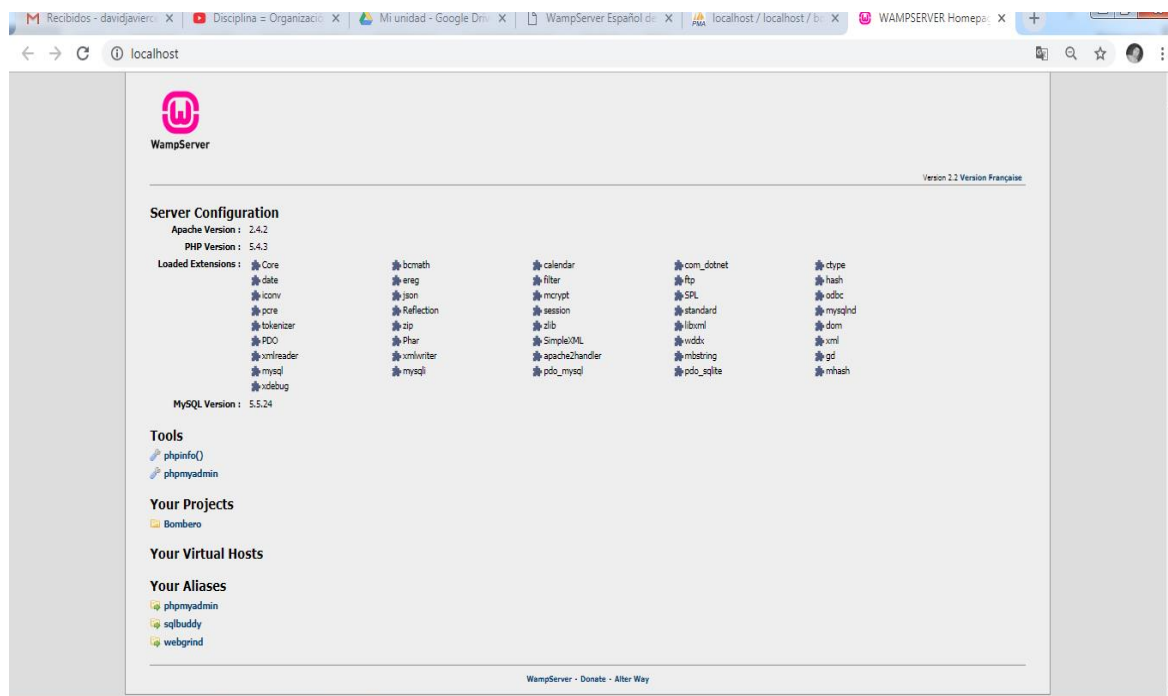


Figura 28: Página principal de Wamp server.

Fuente: Elaborado por el autor.

Luego se extrae la base de datos instalada y copiada el proyecto en la carpeta

C:\wamp\www.

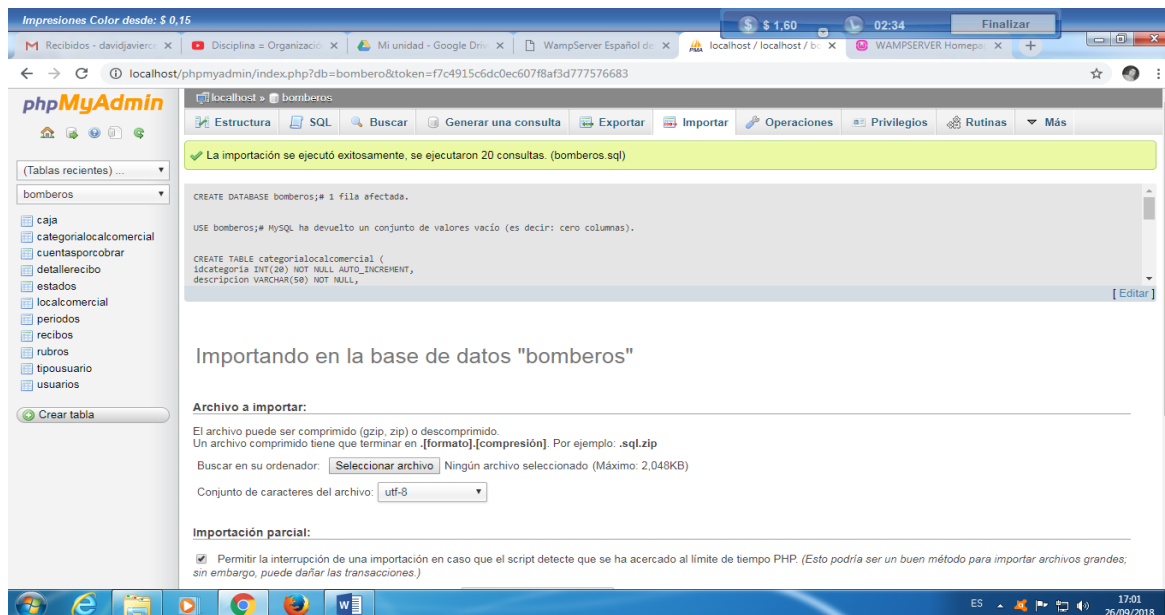


Figura 29: Importar la base de datos.

Fuente: Elaborado por el autor.

2.5.2. Plan de Ejecución.

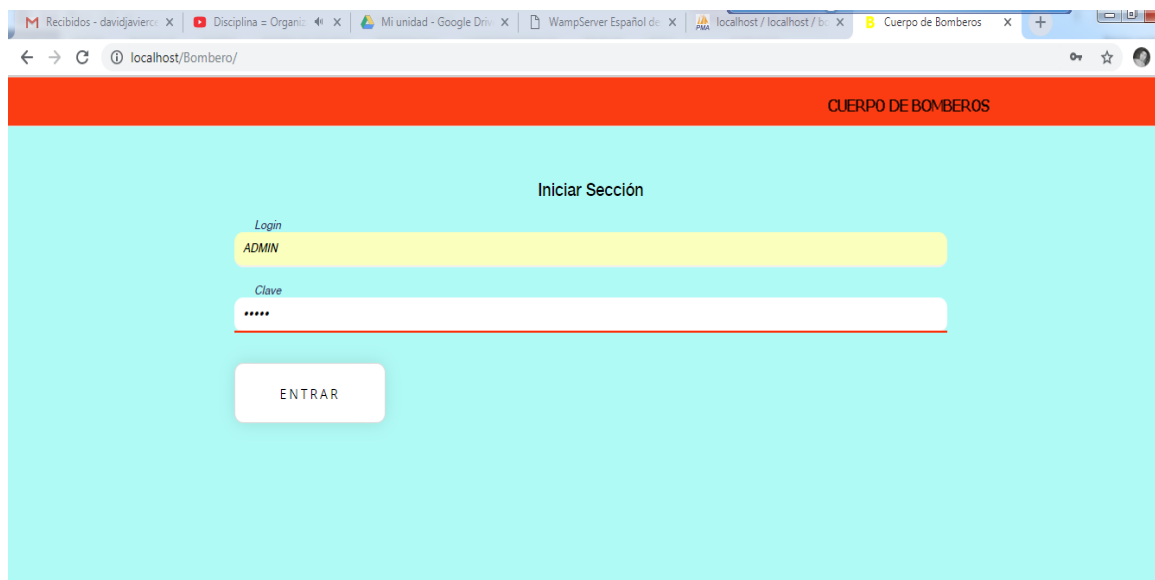


Figura 30: Página del login del Sistema.

Fuente: Elaborado por el autor.

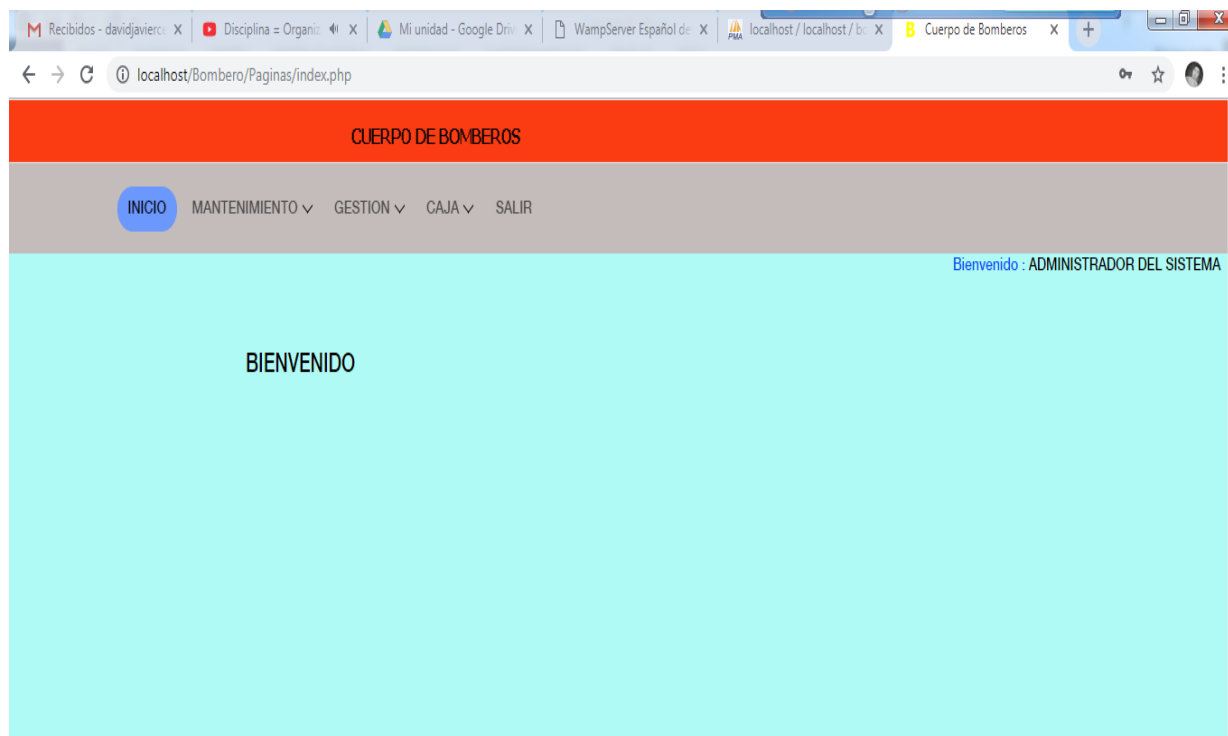


Figura 31: Ingreso al Sistema como Administrador.
Fuente: Elaborado por el autor.



Figura 32: Ejecución del SI, menú consultar usuario (Administrador).
Fuente: Elaborado por el autor.

CUERPO DE BOMBEROS

INICIO MANTENIMIENTO ▾ GESTION ▾ CAJA ▾ SALIR

Bienvenido : ADMINISTRADOR DEL SISTEMA

Nuevo registro

Periodo
ENERO A DICIEMBRE DEL 2018

Rubro
PREVENCION DE INCENDIOS

Razón Social
CEREZO ESMERALDA DAVID JAVIER

Fecha Emisión
26/09/2018

Fecha Vigencia
26/09/2018

Valor
54

GUARDAR

Figura 33: Ejecución del SI, creando la cuenta por cobrar (Administrador).
Fuente: Elaborado por el autor.

CUERPO DE BOMBEROS

INICIO PAGOS DE RECIBOS DESCARGAR RECIBOS SALIR

Bienvenido : ANABEL XXXX XXXX

Recibo de Caja

Fecha Emisión	2018-09-26	N° Recibo	0000001
RUC	1207862978001		
Razon Social	CEREZO ESMERALDA DAVID JAVIER		
Local Comercial	BLUE.COM®		

Código	Descripción de la cuenta	Valor
1	PREVENCION DE INCENDIOS	80

Banco Depósito
Banco del Pichincha

N° de Cuenta
055555555

Fecha Depósito
26/09/2018

N° de Documento
066666666


Valor a Pagar 80

PAGAR

Figura 34: Ejecución del SI, pagando la cuenta por cobrar (Recaudador).
Fuente: Elaborado por el autor.

ACTIVIDADES DE TITULACIÓN x Saiuto | Sistema Académico Int... x Cuerpo de Bomberos x Recibo-Descargado.php x localhost / localhost / bombero... x

localhost/Bombero/Paginas/Recibo-Descargado.php?id=1

 **Sistema de recaudación (Cuerpo de Bomberos Municipal de Baba)**

Recibo de caja

Recibo No	1	Fecha emision	2018-10-02
Establecimiento	1		
Razon social	CEREZO ESMERALDA DAVID JAVIER		
Codigo	Descripcion de la cuenta	Valor	
1	PREVENCION DE INCENDIOS	50	
2	MULTAS	28.5	
Banco deposito	Banco Pichincha	No de cuenta	6258746866
Fecha deposito	2018-10-15	No de documento	000000001
		Valor a pagar	78.5

Figura 35: Ejecución del SI, Recibo de pago de permiso de bomberos (Contribuyente).
Fuente: Elaborado por el autor.

CAPÍTULO III

3. Evaluación Del Prototipo

3.1. Plan de Evaluación

Tabla 12.

Cronograma de Actividades del Desarrollo del Prototipo.	
Fecha	Actividad
15/08/2018	Instalación del Notepad++, Wamp server.
16/08/2018	Realización de la lluvia de ideas.
17/08/2018	Construir la base de datos en un borrador, utilizando los requerimientos del Cuerpo de bomberos del Cantón Baba.
18/08/2018	Realizar la identificación de las tablas adecuadas del sistema.
20/08/2018	Buscar las relaciones entre las tablas del sistema.
21/08/2018	Reunión con el tutor para definir el modelado correcto del sistema.
22/08/2018	Realización del script de la base de datos.
23/08/2018	Normalizar y comprobar las relaciones estén correctas entre las tablas de la base de datos.
24/08/2018	Construir la conexión entre la base de datos con el sistema.
25/08/2018	Ver tutoriales de sistemas informáticos realizados en PHP y MySQL.
27/08/2018	Realización del login del sistema.
28/08/2018	Diseño del menú Inicio del sistema.
29/08/2018	Diseño de la páginas del sistema.

30/08/2018	Diseño de la páginas del sistema.
31/08/2018	Diseño de la páginas del sistema.
03/09/2018	Diseño de la páginas del sistema.
04/09/2018	Diseño de la páginas del sistema.
05/09/2018	Diseño de la páginas del sistema.
06/09/2018	Diseño de la páginas del sistema.
07/09/2018	Diseño de la páginas del sistema.
10/09/2018	Realización de la conexión entre todas las páginas del sistema.
11/09/2018	Prueba de registros en algunas tablas.
12/09/2018	Validación de los campos en cada una de las cajas de texto en las páginas.
13/09/2018	Pruebas de prototipo entando como administrador, recaudador y contribuyente respectivamente.
14/09/2018	Pruebas de caja blanca.
15/09/2018	Pruebas de caja negra.
20/09/2018	Pruebas de funcionamiento del sistema en otro ordenador.

Fuente: Elaborado por el autor.

3.1.1. Prueba de Caja Blanca.

La prueba de caja blanca del software se basa en el minucioso examen de los detalles procedimentales. Se comprueban los caminos lógicos del software proponiendo casos de prueba que ejerciten conjuntos específicos de condiciones y/o bucles. Se puede examinar el «estado del programa» en varios puntos para determinar si el estado real coincide con el esperado o mencionado. (PRESSMAN, 2001)

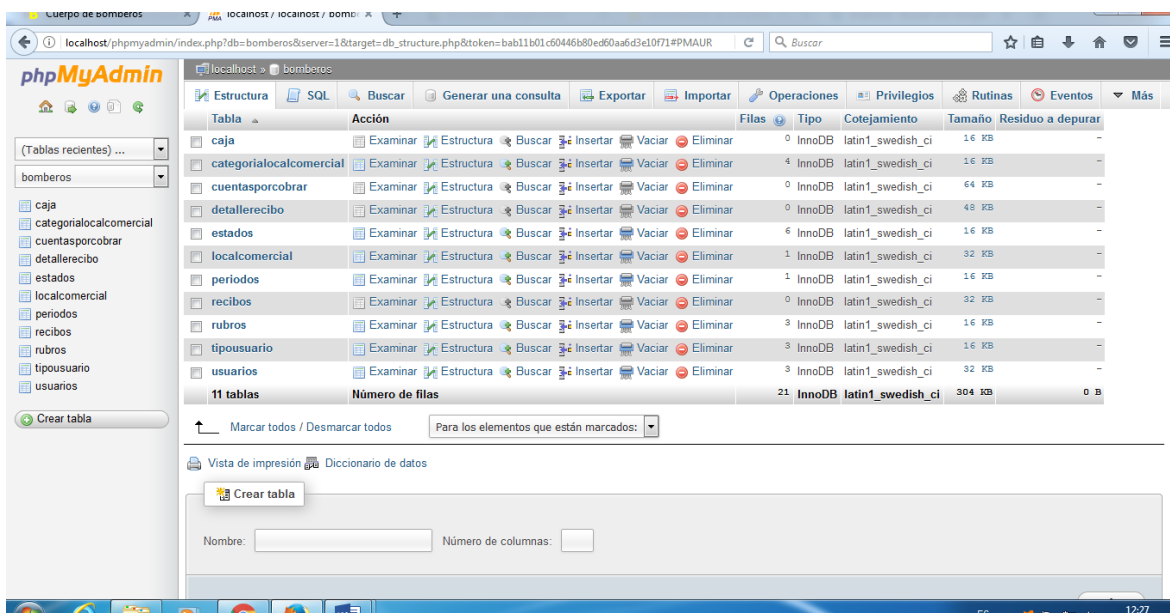


Figura 36: Validación de Datos en el Inicio de Sesión
Fuente: Elaborado por el autor.

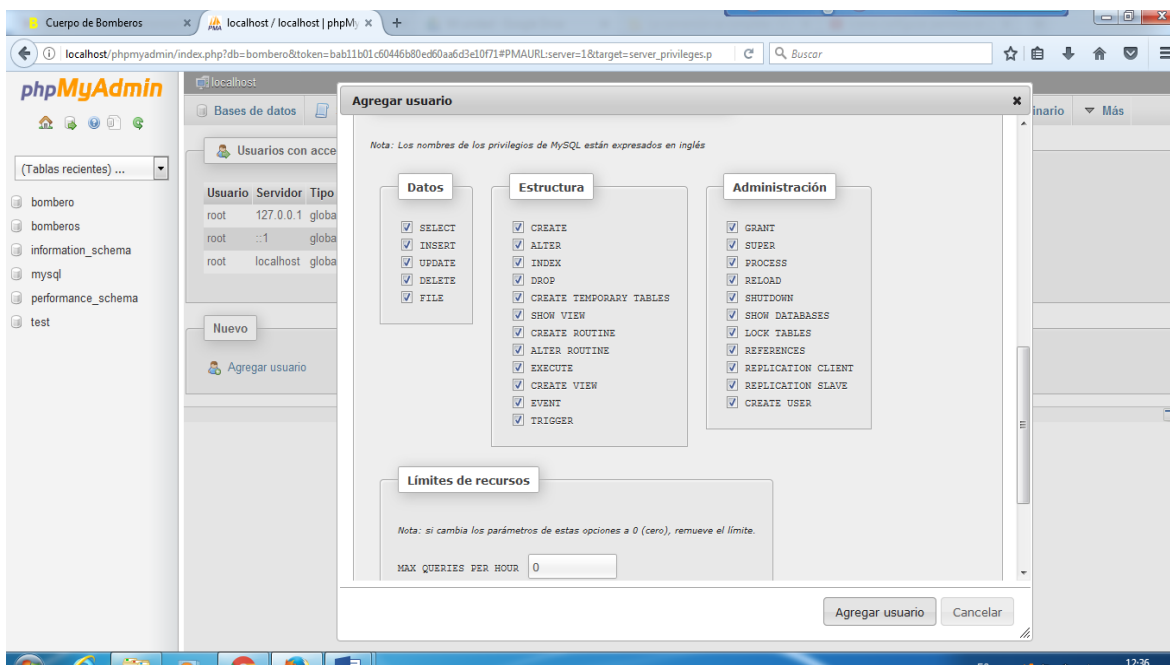


Figura 37: Permisos para el inicio de gestión del SI.
Fuente: Elaborado por el autor.

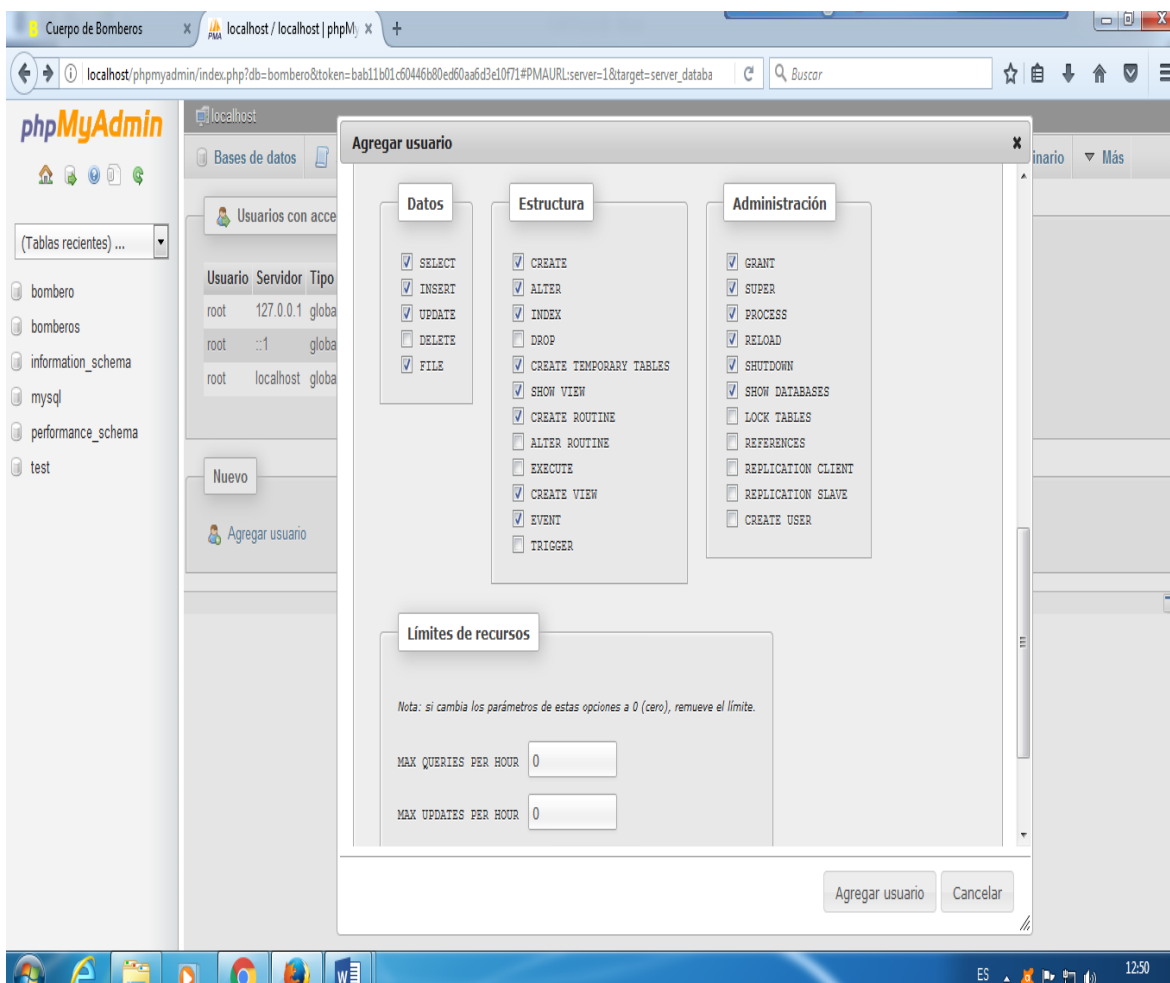


Figura 38: Permisos para el inicio de gestión del SI (usuarios).
Fuente: Elaborado por el autor.

3.1.2. Prueba de Caja Negra.

Cuando se considera el software de computadora, la prueba de caja negra se refiere a las pruebas que se llevan a cabo sobre la interfaz del software. O sea, los casos de prueba pretenden demostrar que las funciones del software son operativas, que la entrada se acepta de forma adecuada y que se produce un resultado correcto, así como que la integridad de la información externa (por ejemplo, archivos de datos) se mantiene. Una prueba de caja negra examina algunos aspectos del modelo fundamental del sistema sin tener mucho en cuenta la estructura lógica interna del software. (PRESSMAN, 2001)

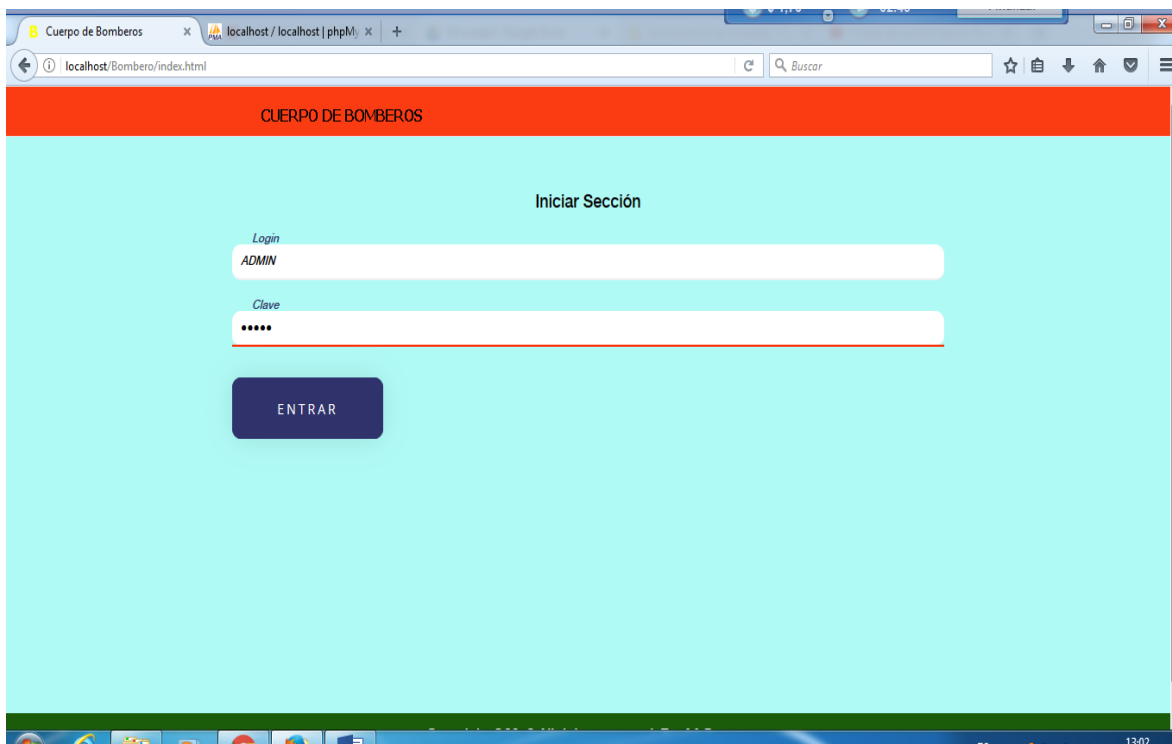


Figura 39: Verificación de conexión exitosa del SI y la Base de Datos.
Fuente: Elaborado por el autor.

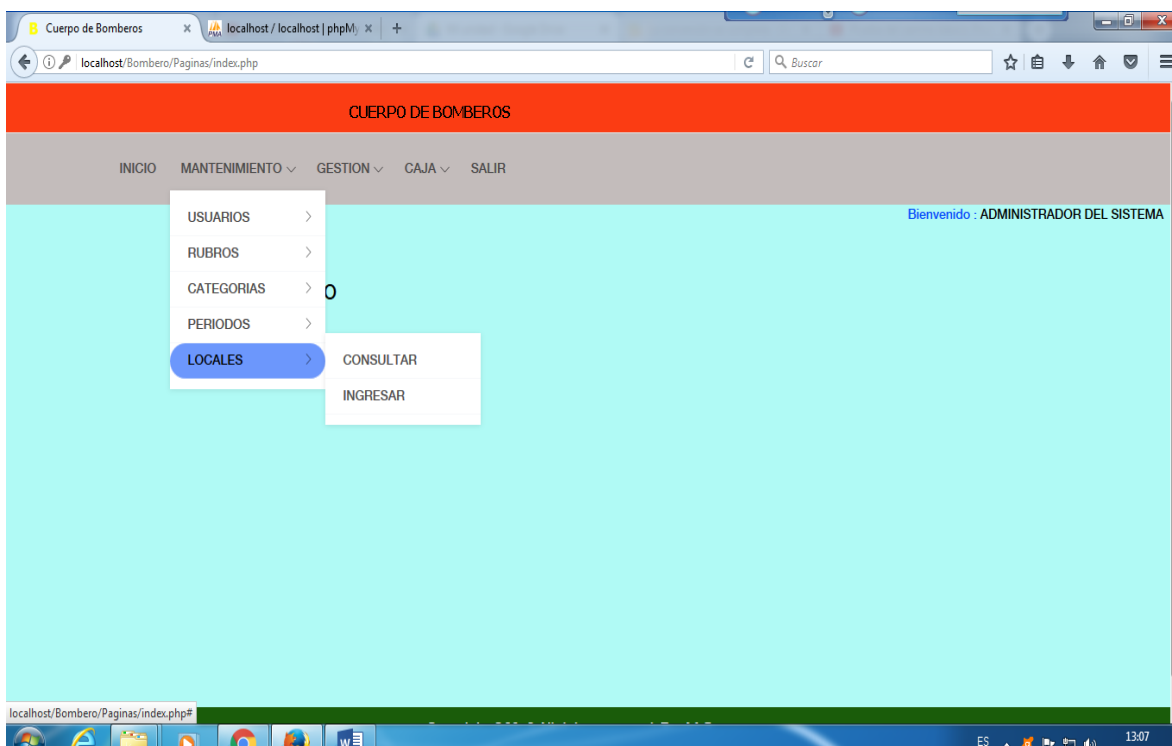


Figura 40: Inicio de Sesión al SI como Administrador.
Fuente: Elaborado por el autor.

The screenshot shows a web browser window with the URL `localhost/Bombero/Paginas/Locales-Agregar.php`. The form is titled "Cedula o Huc" and contains several input fields: "Establecimiento", "Nombre Comercial", "Actividad Comercial", "Categoria del Comercial" (a dropdown menu with "MICRO EMPRESA" selected), "Dirección", "Telefono", "Celular", "Correo", "Tiene Extintor" (a dropdown menu with "SI" selected), and "Tiene Detector de Humo" (a dropdown menu with "SI" selected). A "GUARDAR" button is located at the bottom of the form. A tooltip "Rellene este campo." is visible over the first empty input field. The footer of the page reads "Copyright ©2018 All rights reserved David Cerezo."

Figura 41: Validación de las cajas de texto en el SI.
Fuente: Elaborado por el autor.

The screenshot shows a web browser window with the URL `localhost/Bombero/Paginas/Cuentas-Agregar.php`. The page has a red header "CUERPO DE BOMBEROS" and a navigation menu with "INICIO", "MANTENIMIENTO", "GESTION", "CAJA", and "SALIR". The user is logged in as "ADMINISTRADOR DEL SISTEMA". The form is titled "Nuevo registro" and contains several input fields: "Periodo" (a dropdown menu with "ENERO A DICIEMBRE DEL 2018" selected), "Rubro" (a dropdown menu with "PREVENCION DE INCENDIOS" selected), "Razón Social" (a dropdown menu with "CEREZO ESMERALDA DAVID JAVIER" selected), "Fecha Emisión", "Fecha Vigencia", and an empty text field with a tooltip "Rellene este campo." A "GUARDAR" button is located at the bottom of the form.

Figura 42: Validación de las cajas de texto en el SI.
Fuente: Elaborado por el autor.

3.2. Resultados de la Evaluación

El sistema informático desarrollado e implementado por El Cuerpo de Bomberos del Cantón Baba, ha obtenido resultados muy buenos por parte de los usuarios que laboran en el departamento tesorería por su sencillez a la hora de manipularlo, por estar bien detallado y fácil de usar.

Se comprobó que el sistema cumple con los factores de eficacia, eficiencia, y además que mejora notablemente en antes la forma de hacer el trámite y el después, como se muestra en las siguiente figuras.

Figura 43: Comprobante de Pagar el permiso del Cuerpo de Bomberos.
Fuente: Cuerpo de Bomberos de Baba.


 Sistema de recaudación (Cuerpo de Bomberos Municipal de Baba)			
Recibo de caja			
Recibo No	1	Fecha emision	2018-10-02
Establecimiento	1		
Razon social	CEREZO ESMERALDA DAVID JAVIER		
Codigo	Descripcion de la cuenta	Valor	
1	PREVENCION DE INCENDIOS	50	
2	MULTAS	28.5	
Banco deposito	Banco Pichincha	No de cuenta	6258746866
Fecha deposito	2018-10-15	No de documento	0000000001
Valor a pagar			78.5

Figura 44: Comprobante de Pagar el permiso del Cuerpo de Bomberos generado por el SI.
Fuente: Cuerpo de Bomberos de Baba.

CONCLUSIONES

- El sistema informático obtuvo resultados eficientes por el Cuerpo de Bomberos del Cantón Baba y cumplió con las expectativas esperadas, para la persona que trabaja en la tesorería ha generado buenos comentarios sobre la forma que ayudara a agilizar el trámite.
- El sistema informático está diseñado en base a la metodología RAD y la metodología estructurada se tomó algunas herramientas de gran popularidad en el campo del desarrollo de software. Se utilizó UML para visualizar, construir, construir y detallar la documentación del sistema.
- Para desarrollar el sistema informático se utilizó el Notepad++ para crear el código PHP, y crear todo el sistema desde cero y realizar cada una de las páginas que han sido necesarias, para que cumpla con todas las funciones.
- Al crear este sistema se pretende que la realización del cobro del permiso de bomberos se agilice.

RECOMENDACIONES

- Para el correcto funcionamiento del sistema se debe contratar un Hosting para que la accesibilidad de los usuarios contribuyentes sea las 24 horas del día en caso de que pierdan el comprobante, lo puedan descargar con facilidad.
- Se recomienda tener un directorio ya estructurado para que se guarde la información diaria de todos los comprobantes que se generan.
- También es necesario que se haga diariamente una copia de seguridad por si alguna vez hay algún contratiempo u otros factores.
- Si se va a manipular la base de datos, hay que tener cuidado de no borrar alguna tabla, campo o relación, lo aconsejable es tratar de solo utilizar el sistema informático, por lo que este permite apreciar mejor la información.
- Realizar una correcta capacitación en el uso del sistema para evitar que se escriba información equivocada en los campos y evitar problemas en el sistema.

BIBLIOGRAFÍA

- ALVAREZ, M. (8 de ABRIL de 2008). *https://desarrolloweb.com*. Obtenido de <https://desarrolloweb.com/articulos/notepad.html>
- ARIAS, M. (2017). *Aprende Programación Web con PHP y MySQL: 2ª Edición*. IT Campus Academy.
- ARRIGHI, H. (2004). *www.epidemiolog.net*. Obtenido de <http://www.epidemiolog.net/es/endesarrollo/GestionY analisisDeDatos.pdf>
- CEVALLOS, K. (4 de JUNIO de 2015). *https://ingsoftwarekarlacevallos.wordpress.com*. Obtenido de <https://ingsoftwarekarlacevallos.wordpress.com/2015/06/04/uml-casos-de-uso/>
- CHICANO, E. (2014). *Gestión de servicios en el sistema informático. IFCT0109*. MÁLAGA: IC EDITORIAL.
- ECURED. (2018). *https://www.ecured.cu*. Obtenido de <https://www.ecured.cu/PHP>
- KRALL, C. (2018). *https://www.aprenderaprogramar.com*. Obtenido de https://www.aprenderaprogramar.com/index.php?option=com_content&view=article&id=688:i que-es-y-para-que-sirve-uml-versiones-de-uml-lenguaje-unificado-de-modelado-tipos-de-diagramas-uml&catid=46&Itemid=163
- Lucid Software Inc. (2018). *https://www.lucidchart.com*. Obtenido de <https://www.lucidchart.com/pages/es/qu%C3%A9-es-el-lenguaje-unificado-de-modelado-uml>

- OLIVA, K., & CARRILLO, M. (s.f.). *http://metodologiarad.weebly.com*. Obtenido de <http://metodologiarad.weebly.com>
- PRESSMAN, R. (2001). *INGENIERÍA DEL SOFTWARE (Quinta ed.)*. MADRID: The McGraw-Hill.
- RAMÍREZ, F. (6 de MARZO de 2018). *https://fireosoft.com.co*. Obtenido de <https://fireosoft.com.co/blogs/que-es-y-para-que-sirve-mysql/>
- UNIVERSIA. (2015). *http://noticias.universia.es*. Obtenido de <http://noticias.universia.es/consejos-profesionales/noticia/2015/09/29/1131645/prototipo-sirve.pdf>
- VILLADA, J. (2015). Desarrollo y optimización de componentes software para tareas administrativas de sistemas. IFCT0609. IC EDITORIAL.