



UNIVERSIDAD TÉCNICA DE BABAHOYO

**FACULTAD DE ADMINISTRACIÓN, FINANZAS E
INFORMÁTICA**

PROCESO DE TITULACIÓN

MAYO – OCTUBRE 2018

PROPUESTA TECNOLÓGICA DE GRADO O DE FIN DE CARRERA

PRUEBA PRÁCTICA

PREVIO A LA OBTENCIÓN DEL TÍTULO DE INGENIERO EN SISTEMAS

TEMA:

Sistema Web para el control de trámites de la Unidad de Tránsito
Municipal del Cantón Las Naves.

EGRESADO:

Jorge Isrrael Diaz Montoya

TUTOR:

Ing. Fredy Maximiliano Jordán Cordones.

AÑO 2018

ÍNDICE

ÍNDICE	II
ÍNDICE DE GRÁFICOS.	III
ÍNDICE DE TABLAS.	III
ÍNDICE DE ILUSTRACIONES.	IV
Introducción.....	1
Planteamiento del problema.	2
Delimitación.	3
Justificación.	4
CAPÍTULO I.....	5
1. Diagnóstico de necesidades y requerimientos.	5
1.1. Ámbito de la aplicación: descripción del contexto y hechos de interés.	5
1.2. Establecimiento de requerimientos.	6
1.3. Especificaciones técnicas.....	22
CAPÍTULO II.....	23
2. Desarrollo del prototipo.....	23
2.1. Definición del prototipo.....	23
2.2. Fundamentación teórica del prototipo.	24
2.3. Objetivos de la propuesta.....	35
2.4. Diseño del prototipo.	35
2.5. Ejecución y/o ensamblaje del prototipo.....	65
CAPÍTULO III	81
3. Evaluación del prototipo.....	81
3.1. Plan de evaluación.....	81
3.2. Resultados de evaluación.....	84
3.3. Análisis de resultados.	84
4. Conclusiones y recomendaciones.	85
4.1. Conclusiones.....	85
4.2. Recomendaciones.	86
Referencias	87

ÍNDICE DE GRÁFICOS.

Gráfico 1: Diagrama de clases	38
Gráfico 2: Diagrama de caso de uso del sistema.	39
Gráfico 3: Diagrama de caso de uso de inicio de sesión.	40
Gráfico 4: Diagrama de caso de uso de configuración del perfil de usuario.....	41
Gráfico 5: Diagrama de caso de uso del administrador.	42
Gráfico 6: Diagrama de caso de uso del digitador.....	43
Gráfico 7: Diagrama de caso de uso del asistente administrativo.	44
Gráfico 8: Diagrama de actividades de inicio de sesión.	45
Gráfico 9: Diagrama de actividades de ingreso de usuarios.	46
Gráfico 10: Diagrama de actividades para editar usuarios.....	46
Gráfico 11: Diagrama de actividades para ingreso de trámites de clientes.....	47
Gráfico 12: Diagrama de secuencia para autenticar usuarios.....	47
Gráfico 13: Diagrama de secuencia para crear usuarios.	48
Gráfico 14: Diagrama de secuencia para ingreso de trámites.	49
Gráfico 15: Diagrama de secuencia para control de requerimientos ciudadanos de clientes.....	50
Gráfico 16: Modelo conceptual de la base de datos	51
Gráfico 17: Diagrama entidad relación.....	52
Gráfico 18: Diagrama relacional de la base de datos.....	53

ÍNDICE DE TABLAS.

Tabla 1: Colores.....	54
Tabla 2: Folders.	54
Tabla 3: roles	55
Tabla 4: tipo_licencias	55
Tabla 5: tipo_documentos	56
Tabla 6: licencias	56
Tabla 7: servicios	57
Tabla 8: users.....	57
Tabla 9: clientes	58
Tabla 10: revisiones	59

Tabla 11: transferencia_dominios.....	60
Tabla 12: cambio_caracteristicas.....	61
Tabla 13: renovacion_matriculas.....	61
Tabla 14: cambio_servicios.....	62
Tabla 15: requerimientos_ciudadanos	63
Tabla 16: detalles_requerimientos.....	63
Tabla 17: Funcionalidad y factibilidad de uso	81
Tabla 18: Estabilidad del sistema	82
Tabla 19: Compatibilidad del sistema.....	83
Tabla 20: Interoperabilidad del sistema	83
Tabla 21: Resultados de evaluación.....	84

ÍNDICE DE ILUSTRACIONES.

Ilustración 1: Interfaz del phpMyAdmin.....	64
Ilustración 2: Interfaz del Visual Studio Code.	65
Ilustración 3: Interfaz acceso al sistema.....	65
Ilustración 4: Roles de usuarios.....	66
Ilustración 5: Usuarios del sistema	66
Ilustración 6: Búsqueda de usuarios.	67
Ilustración 7: Registro de usuarios.....	67
Ilustración 8: Inactividad de usuario.....	68
Ilustración 9: Usuario desactivado.....	68
Ilustración 10: Configuración de perfil.	69
Ilustración 11: Actualizando información personal del usuario.	69
Ilustración 12: Consulta de todos los clientes ingresados.....	70
Ilustración 13: Registro de datos válidos de clientes.	70
Ilustración 14: Actualización de clientes.	71
Ilustración 15: Listado de los servicios de la unidad de la unidad de tránsito municipal	71
Ilustración 16: Lista de las revisiones vehiculares realizadas en la unidad de tránsito municipal.	72

Ilustración 17: Listado de revisiones vehiculares por fechas especificadas por el usuario.	72
Ilustración 18: Ingreso de que el cliente posee sus documentos necesarios para la revisión vehicular.....	73
Ilustración 19: Revisión vehicular más renovación de matrícula.....	73
Ilustración 20: Crear cambio de servicios de clientes que ya posean una revisión vehicular.....	74
Ilustración 21: Generar informes de la cantidad de trámites que ha realizado un cliente.	74
Ilustración 22: Informe general de los trámites vehiculares de un cliente.....	75
Ilustración 23: Generar informe general de trámites de revisión vehicular.....	75
Ilustración 24: Informe por fecha de trámites de revisión vehicular.....	76
Ilustración 25: Generar informe de trámites de revisión vehicular de un cliente en una fecha especificada.....	76
Ilustración 26: Informe de revisión vehicular de un cliente en una fecha especificada anteriormente.....	77
Ilustración 27: Gráficos estadísticos donde se muestra la cantidad de trámites de revisión, cambio de servicios y transferencia de dominio realizadas en el año actual. ..	77
Ilustración 28: Gráfico de pastel de los cambios de servicios que se han realizado en los últimos meses.....	78
Ilustración 29: Panel para el registro de nuevos requerimientos que los moradores o ciudadanos tengan en el cantón las Naves	78
Ilustración 30: Selección de archivo de proyecto o solicitud, donde el cliente expresa lo que se requiere en el cantón o los inconvenientes que causan molestias.....	79
Ilustración 31: Panel para añadir los inconvenientes del sector, especificando su calle y dirección.....	79
Ilustración 32: Consulta de los requerimientos realizados con sus respectivos proyectos.	80

Introducción.

Los trámites vehiculares siempre han sido un requerimiento esencial para aquellas personas que poseen algún medio de transporte terrestre, de manera que esta información debe permanecer segura en el lugar donde se realizó dicho trámite, y si en algún futuro un cliente desea corroborar información con respecto a sus documentos, pueda acercarse y mediante algún parámetro de búsqueda que el cliente pueda facilitar, le puedan generar una respuesta de forma rápida con la información que él requiere.

En esta Unidad de tránsito existe el inconveniente de que, si un trámite es realizado, y el cliente hace la entrega de todos sus documentos respectivos, estos son almacenados en carpetas físicas, donde los funcionarios del departamento de archivos tienen que realizar búsquedas de forma manual de todos los documentos cuando estos sean requeridos, ya sea porque alguna persona se los solicitó, o por que desean contabilizar la cantidad de trámites que se han realizado.

Ya que la tecnología avanza, es de suma importancia actualizarse y hacer uso de la misma, para que mediante un sistema de información sea capaz de llevar el control de estos procesos, de manera que los documentos que son archivados en folders, puedan ser corroborados de forma digital, para que los funcionarios de la Unidad de tránsito obtengan de forma rápida y eficiente la información requerida.

Como en esta propuesta tecnológica, se va a llevar a cabo el desarrollo de un sistema informático para el control de trámites, encuadra perfectamente a la línea de investigación, Ingeniería de Software y Programación.

Planteamiento del problema.

La cantidad de trámites que se lleva a cabo dentro de la Unidad de tránsito municipal de las Naves diariamente es de vital importancia, pues representa el bien más importante de esta institución como es la información, por lo cual es necesario que exista una constancia digital de la entrega de los documentos respectivos de cada cliente en dicha organización, y facilitar su búsqueda al momento de que estos sean requeridos.

La falta de responsabilidad e incluso en ocasiones de ética por parte de los funcionarios en la unidad de tránsito, puede conllevar a que tengan desorganizado o hasta incluso llegar a perder documentos de los diferentes trámites, que se hayan realizado o que estén por llevarse a cabo.

Además que cuando se desee realizar alguna búsqueda, cada trámite tiene un código el cual lo identifica, además de pertenecer a un tipo de folder, de manera que si algún empleado desea buscar algún documento tendrá que revisar en todos los folders que tenga a su disposición, trayendo consigo problemas para el cliente, ya que tendrá que esperar hasta que encuentren su respectivo trámite.

Por otra parte las auditorias que realizan los funcionarios externos a la Unidad de tránsito de las Naves como son los miembros del municipio, siempre califican la forma de trabajar de los empleados de los diferentes departamentos, y una de ellas es donde tienen el respaldo de la información de todos los documentos tramitados por los clientes, entonces es necesario que cuenten con un sistema de información en donde conste que los documentos han sido entregados a la unidad de tránsito, para que puedan ser aprobados por esta evaluación.

La unidad de tránsito ha buscado ya soluciones por otros medios como archivar dicha información en documentos de Excel, que muchas veces también tienden a ser un problema, ya que por A o B motivo la máquina se infecta de virus malicioso o llega a suceder algún suceso inesperado, los datos pueden llegar a perderse.

Es por ello que aprovechando que hoy en día tenemos a nuestra disposición tecnologías que podemos utilizar para gestionar y automatizar procesos de diferente tipo, se ha optado por el desarrollo de esta propuesta tecnológica que será de gran utilidad para esta unidad de tránsito en el cantón Las Naves, más que todo en el ahorro de tiempo que obtendrán cuando deseen buscar u obtener información detallada de los diferentes trámites realizados en fechas que el usuario pueda especificar, o en su defecto búsquedas de acuerdo al cliente, o al número de adhesivo que posee cada trámite.

Esta propuesta tecnológica mejora también el control de todas las peticiones o requerimiento que realicen los ciudadanos a la unidad de tránsito, de manera que ya no tengan que contar con papeles en físico únicamente cuando un cliente realice una solicitud, si no que sea ingresado al sistema junto con lo que la persona está solicitando y la fecha en la cual se realizó dicho proceso.

Delimitación.

Localización del problema objeto de estudio.

La investigación se llevó a cabo en la Unidad de Tránsito Municipal perteneciente a la provincia de Bolívar en el cantón Las Naves.

Tiempo.

La investigación se realizó en el periodo 2017, donde se determinó que la Unidad de tránsito no cuenta con ningún sistema informático para llevar un control de todos sus procesos realizados.

Justificación.

En la entrevista que se realizó con el director de la Unidad de tránsito de las Naves, el Tecnólogo Benito López, me dio a conocer que el problema mayor es la búsqueda de los diferentes documentos tramitados de los clientes, ya que cada documento está en un folder de diferente color, y además cada folder posee un sin número documentos, por lo tanto para mejorar esto, requieren de una aplicación que tenga como constancia el registro de ese trámite, además de conocer en qué folder se encuentra, la cantidad de documentos que posee cada trámite, y un conteo de todos los trámites realizados.

Con la aplicación informática SIUT se gestionará todos esos procesos, con el fin, salvaguardar la información de forma digital, y agilizar las búsquedas de los trámites, además de generar informes de cuantos trámites se han llevado a cabo en diferentes fechas y de acuerdo a un parámetro de búsqueda.

Otra de las razones por la cual es viable este sistema informático es por la necesidad de controlar el rol que tiene cada funcionario, es decir que un empleado no pueda realizar tareas administrativas que no le pertenezcan en los distintos departamentos de la Unidad de Tránsito, de manera que cada miembro tendrá un usuario y una clave, y por supuesto un rol asignado, para que de esta manera, lleve a cabo los diferentes procesos que estén dentro de sus labores.

CAPÍTULO I

1. Diagnóstico de necesidades y requerimientos.

1.1. Ámbito de la aplicación: descripción del contexto y hechos de interés.

La Unidad de tránsito municipal de las Naves cuenta con una gran cantidad de información, de todos los trámites que se han llevado a cabo dentro de la organización, como por ejemplo: la revisión vehicular, la transferencia de dominio, cambio de servicios, y los requerimientos ciudadanos (es decir todas las peticiones realizadas por los clientes a la institución, como por ejemplo: una señalización de tránsito, un semáforo, etc).

Por otra parte las auditorias que realizan los funcionarios externos a la Unidad de tránsito de las Naves como son los miembros del municipio, siempre califican la forma de trabajar de los empleados de los diferentes departamentos, y una de ellas es donde tienen el respaldo de la información de todos los documentos tramitados por los clientes, entonces es necesario que cuenten con un sistema de información en donde conste que los documentos han sido entregados a la unidad de tránsito, para que puedan ser aprobados por esta evaluación.

Además las búsquedas que realizan los funcionarios de la empresa, la realizan de forma manual, es decir contabilizando uno por uno cada trámite, para luego generar un informe de dicha información.

1.2. Establecimiento de requerimientos.

Control de cambios.

Registro del control de cambios en el documento SRS (Software Requirements Specifications) del SIUT (Sistema de Información de la unidad de tránsito – WEB).

Numero Revisión	Descripción.	Autor.	Fecha (d/m/a).	Versión que genera.
1.0	Versión inicial de la aplicación.	Jorge Isrrael Diaz Montoya	11/06/2018	V1

Elaborado por: (Diaz Montoya Jorge, 2018)

Propósito.

El presente documento tiene como propósito definir las especificaciones funcionales, no funcionales para la implementación de una aplicación WEB que permitirá administrar y consultar la información de los documentos que se hayan entregado de los diferentes trámites en la Unidad de tránsito de las Naves.

Alcance.

Diseño, desarrollo e implementación de la aplicación SIUT (Sistema de Información de la Unidad de Tránsito).

El SIUT será una aplicación que funcionará en un entorno WEB que permitirá administrar y consultar la información de los documentos tramitados en la Unidad de tránsito, además llevar un control de la cantidad de trámites que se han realizado. Esta aplicación dará apoyo a los siguientes procesos:

- Administrar usuarios de la aplicación.
- Consulta de roles y permisos de usuarios.
- Administración de perfiles de usuarios.
- Administración de clientes

- Administración de revisión vehicular.
- Administración de transferencia de dominios.
- Administración de cambio de servicios.
- Administración de requerimientos ciudadanos.
- Generar informes de la cantidad de trámites realizados por fechas.
- Generación de gráficos estadísticos de barra.

El SIUT permitirá una flexibilidad de acceso a la información de los trámites y de los diferentes requerimientos ciudadanos, y de igual manera una actualización dinámica de la misma.

Permitirá ingreso de clientes, y trámites vehiculares a través de un ordenador con una conexión a Internet.

Personal involucrado.

Nombre	Jorge Israel Diaz Montoya.
Rol	Programador, tester, diseñador, encargado del despliegue de la aplicación.
Categoría profesional	Estudiante de ingeniería en sistemas.
Responsabilidades	Codificar la aplicación, diseñar el front-end y desplegarla en un hosting.
Información de contacto	jdiazm470@utb.edu.ec

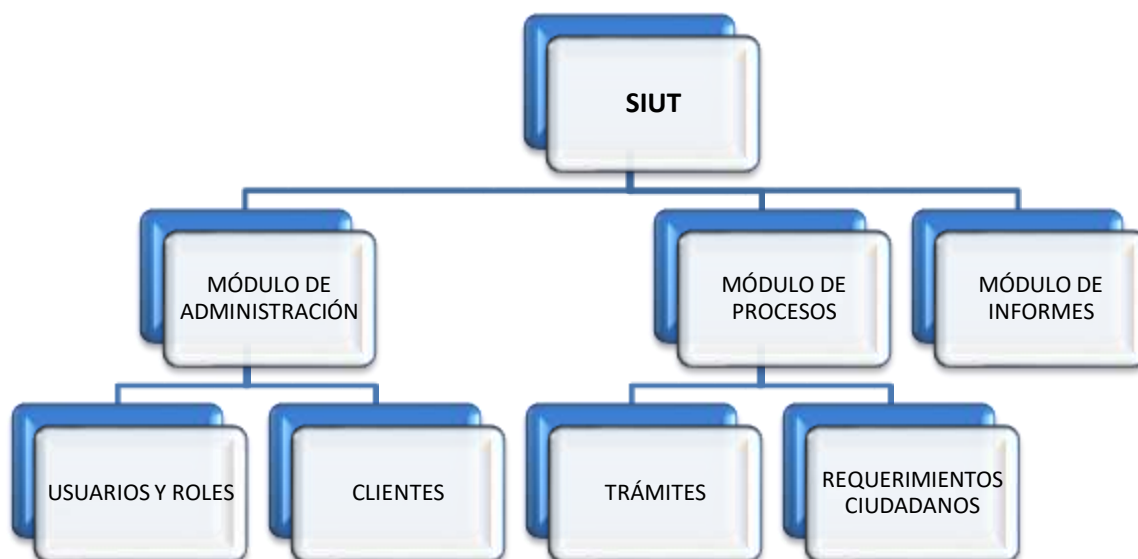
Elaborado por: (Diaz Montoya Jorge, 2018)

Descripción general.

Perspectiva del producto.

La aplicación web podrá ser ejecutada desde un navegador donde el usuario deberá autenticarse, para luego poder realizar los diferentes procesos de la unidad de tránsito.

Funcionalidad del producto.



Elaborado por: (Diaz Montoya Jorge, 2018)

Características de los usuarios.

Tipo de usuario	Director de la unidad de tránsito.
Formación	Tecnólogo en mecánica general.
Habilidades	Manejo de equipos y sistemas informáticos.
Actividades	Gestionar usuarios, roles, y generar informes de cantidad de trámites realizados.

Elaborado por: (Diaz Montoya Jorge, 2018)

Tipo de usuario	Digitador de la unidad de tránsito.
Formación	Administración.
Habilidades	Manejo de equipos y sistemas informáticos.
Actividades	Ingreso clientes y documentos de trámites vehiculares.

Elaborado por: (Diaz Montoya Jorge, 2018)

Tipo de usuario	Asistente administrativo.
Formación	Conocimientos en computación básica.
Habilidades	Manejo de equipos y sistemas informáticos.
Actividades	Ingreso de clientes y sus requerimientos.

Elaborado por: (Díaz Montoya Jorge, 2018)

Restricciones.

La aplicación será desarrollada en laravel 5.4, un framework open source que garantiza una seguridad en sistemas construidos con php, además que nos ofrece un patrón diseño MVC, que nos permite tener un código escalable, y fácil de mantener, el motor de base de datos es mysql, por motivos de despliegue de la aplicación en un hosting de pago, ya que la mayoría de los hosting nos provee un SGDB mysql.

Suposiciones y dependencias.

Ninguno.

Evolución previsible del sistema.

Crear un subdominio en un hosting, configuración de storage y base de datos en el internet.

Requisitos específicos.

R1: Permitir la autenticación de usuarios.

R2: Configuración de perfiles de usuario.

R3: Permitir la gestión (crear, editar, habilitar, deshabilitar, listar) usuarios, clientes, trámites, y requerimientos ciudadanos.

R4: Ingresar la constancia de trámites de matriculación vehicular de los clientes.

R5: Realizar ingresos de peticiones de los ciudadanos.

R6: Emitir informes del total de trámites realizados.

R7: Emitir recibos para los clientes.

Requisitos comunes de los interfaces.

Interfaces de usuario.

Las interfaces de usuario están relacionadas con el front-end de la aplicación con el cuál usuario tendrá que interactuar. La entrada de los datos será realizada a través de los periféricos de entradas del computador.

Las interfaces de usuario ayudaran al usuario final trabajando en una página cuya navegación por las diferentes opciones será dinámica y en tiempo real sin recargar la página gracias a la filosofía de las SPA (Single page application), cuya programación de componentes se basa en la reactividad. Las vistas contienen:

- Formularios de ingreso.
- Alertas de errores.
- Alertas informativas.
- Tablas responsive.
- Componentes reactivos.

A continuación, observamos un wireframe de la interfaz de usuario para autenticarse, el usuario deberá tener una cuenta de usuario para acceder, que le será proporcionada por el director o administrador de la aplicación.

En caso de ocurrir un error se mostrará un error indicando que las credenciales no coinciden.

Interfaz de comunicación.

La interfaz de comunicación entre la base de datos Mysql y el sistema informático desarrollado en LARAVEL se lo realiza mediante el ORM Eloquent.

Requisitos funcionales.

Requisito funcional 1.

Número	1
Nombre	Permitir la autenticación de usuarios
Fuente	Relación: Usuario; Columnas: usuario o correo y clave.
Prioridad	Alta y Esencial

Elaborado por: (Diaz Montoya Jorge, 2018)

Introducción.

El sistema debe permitir el ingreso del correo o el nombre de usuario y la clave de acceso, para poder acceder a las diferentes secciones del sistema.

Entrada.

Nombre de usuario, correo, contraseña, por defecto el sistema detectará que rol de usuario es: (Director, Digitador, Asistente administrativo).

Procesos

Los usuarios con el rol de administrador o director, asistente administrativo, y digitador deberán ingresar sus credenciales de autenticación, y presionar enter o dar click en el botón entrar.

Salidas

Salidas para: El Administrador/Director.

Mostrar arreglo de mensajes con errores de campos vacíos.

Mensaje de información por enviar credenciales incorrectas.

Requisito funcional 2.

Número	2
Nombre	Permitir el ingreso de la revisión vehicular
Fuente	Relaciones: Clientes, Revisión vehicular
Prioridad	Alta y Esencial

Elaborado por: (Diaz Montoya Jorge, 2018)

Introducción.

El sistema debe permitir, la gestión de todos los clientes, y la constancia de que se ha entregado todos los documentos de revisión vehicular.

Entradas

Datos personales nombres, apellidos, cédula, constancia de poseer: (papeleta de votación, matrícula, revisión e identificación vehicular, formulario de revisión), registrar la fecha de transacción, cantidad de hojas que tiene cada trámite de revisión vehicular, el número de adhesivo, y una observación donde el usuario podrá especificar si la matricula ha caducado y se le otorgará una nueva.

Procesos

Para cumplir con este requerimiento, el usuario deberá autenticarse, solo los usuarios con el rol de digitador podrán ingresar este trámite, y lo primero que deberá es ingresar el cliente, y en caso de ya existir el cliente, entonces solo seleccionarlo de acuerdo a sus datos personales, como por ejemplo la cédula, y añadir su trámite de revisión vehicular.

Salidas

Salidas para el: Usuario digitador y director.

Aviso de información por enviar datos en blanco.

Cuadros de diálogo de error en caso de ingresar el formato incorrecto de los datos.

Requisito funcional 3.

Número	3
Nombre	Permitir crear transferencia de dominios, activarlas, desactivarlas, modificarlas.
Fuente	Relaciones: Clientes, Transferencia de dominio
Prioridad	Alta y esencial

Elaborado por: (Diaz Montoya Jorge, 2018)

Introducción.

La aplicación debe permitir gestionar los procesos de los clientes con sus respectivos trámites de transferencia de dominio.

Entradas

Nombres, apellidos, cédula, constancia de poseer: (matricula original, contrato de compra y venta, pago de rodaje, pago de recargas, pago de duplicado de matrícula, pago de traspaso, revisión vehicular), registro de cambio de características, observación de citaciones al contrato, y la fecha de transacción, por cada trámite ingresar la cantidad de hojas que conforman el documento, e ingreso del número de adhesivo.

Procesos

Para cumplir con este requerimiento, el usuario digitador deberá autenticarse, para poder ingresar el trámite, y el cliente, y en caso de ya existir el cliente, entonces solo seleccionarlo de acuerdo a sus datos personales, como por ejemplo la cédula, o algunos de sus datos personales y a continuación añadir el trámite de la transferencia.

Salidas

Salidas para: El usuario digitador, director.

Alertas de error por enviar datos vacíos.

Mensaje de error en casos de ingresar incorrectamente el formato de los datos de transferencia de dominio.

Salida de error en caso de ingresar un número de adhesivo ya existente.

Requisito funcional 4.

Número	4
Nombre	Ingresos de los cambios de servicios de un cliente.
Fuente	Relaciones: Clientes, Trámites de cambio de servicios.
Prioridad	Alta y Esencial

Elaborado por: (Diaz Montoya Jorge, 2018)

Introducción.

La aplicación debe permitir gestionar todos los clientes con sus respectivos trámites de cambio de servicios.

Entradas

Num. Adhesivo, Nombres, apellidos, cédula, dirección, documentos del trámite para cambio de servicios.

Procesos

Para cumplir con este requerimiento, el usuario digitador deberá autenticarse y posteriormente ingresar los datos del cliente, como la cédulas y sus nombres, y la constancia de entrega de los documentos: (Permiso, de operación, Resolución del SRI del cambio de servicio), la licencia del clientes profesional (C, D, E), y en caso de no tener licencia el cliente deberá presentar un contrato de trabajo.

Salidas

Salidas para: Usuario administrador o digitador.

Alertas de error por no llenar todos los campos.

Mensajes de excepciones en casos de ingresar incorrectamente el formato de los datos.

Avisos por cualquier error que se presenta durante la ejecución de alguna transacción.

Requisito funcional 5

Número	5
Nombre	Ingreso de requerimientos de clientes y sus requerimientos ciudadanos.
Fuente	Relaciones: Clientes, requerimientos ciudadanos.
Prioridad	Alta y esencial.

Elaborado por: (Diaz Montoya Jorge, 2018)

Introducción

En este requisito se desarrollará los módulos respectivos para el ingreso de clientes, todos sus datos personales, y sus peticiones a realizar, si el cliente posee documentos, estos deberán ser escaneados y luego subirlos al sistema.

Entradas

Cédula, fecha, nombres y apellidos del cliente, documentos del cliente en digital.

Procesos

Para cumplir este requerimiento el usuario con el rol asistente administrativo deberá ingresar los datos del cliente, y a continuación sus peticiones que desea que la unidad de tránsito solvente, luego de la petición se le entregará un recibo que el sistema debe generar, al cliente con la petición que ha realizado.

Salidas

Las salidas van dirigidas a: cliente.

Recibo: al cliente se le entregará un recibo donde se especifique los datos de cliente y sus requerimientos.

Requisito funcional 6

Número de requisito	RF6
Nombre de requisito	Emitir informes de: Trámites de revisión vehicular realizados de acuerdo a una fecha específica, y de los clientes.
Fuente del requisito	BD Tabla: Clientes, Revisión vehicular.
Prioridad del requisito	Alta/Esencial.

Elaborado por: (Diaz Montoya Jorge, 2018)

Introducción

El sistema deberá emitir informes de la cantidad de trámites de revisión vehicular realizados en cualquier fecha que el usuario especifique junto con el cliente que realizó dicho trámite.

Entradas

Cédula, número de adhesivo del trámite, fecha, nombres y apellidos del cliente.

Procesos

Para cumplir este requerimiento se mostrará en un componente web de Vue JS la opción para imprimir el informe de los trámites que corresponden a revisión vehicular, donde el usuario deberá especificar el filtro por el cuál desea generar el informe, ya sea por la cédula, nombres de clientes, la fecha o el número de adhesivo del trámite realizado, de cualquier modo, si selecciona una fecha obtendrá todos los trámites de revisión que se registraron en esa fecha, y por supuesto con la cantidad de trámite que se han llevado a cabo.

Salidas

Las salidas van dirigidas a: Administrador (Director), Digitador.

Mensaje de aviso: operaciones efectuadas correctamente.

Mensaje de error: en casos de ingresar incorrectamente los datos es decir que el formato de los datos sea incorrecto.

Informe en pdf de todos los trámites de revisión vehicular.

Requisito funcional 7

Número de requisito	RF7
Nombre de requisito	Emitir informes de: Trámites de transferencia de dominio.
Fuente del requisito	BD Tabla: Clientes, transferencia de dominio.
Prioridad del requisito	Alta/Esencial

Elaborado por: (Diaz Montoya Jorge, 2018)

Introducción

El sistema debe emitir informes de la cantidad de trámites de transferencia de dominios en función de un parámetro que podrá ser al igual que el de revisión vehicular, la fecha en la que se realizó el trámite.

Entradas

Cédula, número de adhesivo del trámite, fecha, nombres y apellidos del cliente.

Procesos

Para cumplir con este requerimiento el usuario autenticado deberá entrar al componente de los trámites de transferencia de dominio y dar click en imprimir en caso que desea imprimir todos los trámites, o especificar una fecha para mostrar la cantidad de trámites que coincidan con ese criterio de búsqueda.

Salidas

Las salidas van dirigidas a: Administrador (Director), Digitador.

Mensaje de aviso: operaciones efectuadas correctamente.

Mensaje de error: en casos de ingresar incorrectamente los datos es decir que el formato de los datos sea incorrecto.

Informe en pdf de las transferencias de dominio.

Requisito funcional 8

Número	8
Nombre	Emitir informes de: Trámites de cambio de servicios.
Fuente	Relaciones: Clientes, cambio de servicios.
Prioridad del requisito	Alta y esencial

Elaborado por: (Diaz Montoya Jorge, 2018)

Introducción

El sistema debe emitir informes de la cantidad de trámites de cambio de servicios que se ha realizado.

Entradas

Cédula, número de adhesivo del trámite, fecha, nombres y apellidos del cliente.

Procesos

Para cumplir con este requerimiento el usuario autenticado deberá entrar al componente de los trámites de cambio de servicios y luego dar click en imprimir para mostrar en un pdf todos los cambios de servicios ingresado, además de poder hacer búsquedas específicas ya sea por el número de adhesivo, la fecha de la transacción, o datos personales del cliente.

Salidas

Las salidas van dirigidas a: Administrador (Director), Digitador.

Informe en pdf de los cambios de servicios.

Requisito funcional 9

Número de requisito	9
Nombre de requisito	Dashboard de gráficos estadísticos.
Fuente del requisito	Relaciones: Clientes, revisión vehicular, transferencia de dominio, cambio de servicios, requerimientos ciudadanos.
Prioridad del requisito	Media y deseado

Elaborado por: (Diaz Montoya Jorge, 2018)

Introducción

En la parte principal del sistema en un componente Vue JS se mostrará todos los gráficos estadísticos donde se visualizará la cantidad de trámites, que se han realizado, en los últimos meses.

Entradas

Navegar al home o inicio de la single page application.

Procesos

Para este requerimiento es necesario que el usuario navegue al inicio de la aplicación donde se encontrará con un dashboard con todos los gráficos estadísticos de los trámites realizados en las últimas fechas, como por ejemplo cuantos trámites se ha realizado cada mes.

Salidas

Las salidas van dirigidas a: Administrador (Director), Digitador, asistente administrativo.

Gráficos de barra estadísticos.

Requisito funcional 10

Número de requisito	10
Nombre de requisito	Gestión de perfiles de usuarios
Fuente del requisito	Relaciones: Usuarios, Color.
Prioridad del requisito	Baja u opcional

Elaborado por: (Diaz Montoya Jorge, 2018)

Introducción

El sistema debe ofrecer al usuario la posibilidad de que configure su perfil, ya sea añadir una foto de perfil, o de portada, o cambiar el color del menú de navegación.

Entradas

Nombres, apellidos, correo, contraseña, foto, portada, nombre de usuario.

Procesos

Para cumplir con este requerimiento el usuario debería acceder a la opción de perfil, donde se mostrará su información personal, y luego dará click en un botón para editar el perfil, y podrá modificar su información personal.

Salidas

Las salidas van dirigidas a: Usuarios Administrador (Director), Digitador, asistente administrativo.

Información personal del usuario.

Requisitos no funcionales.

Requisitos de rendimiento.

Cantidad de usuarios simultáneos:

Pueden existir la cantidad de usuarios que sea necesario haciendo peticiones al sistema.

Transacciones en un tiempo determinado:

No hay una cantidad exacta sobre la cantidad de trámites a efectuar de forma diaria, ya que eso depende de los ciudadanos que se acerquen a la Unidad de tránsito a realizar algún trámite, o de la cantidad de documentos que se suban a la plataforma.

El servidor de base de datos, deberá tener un respaldo apropiado de la información.

Seguridad.

- El uso de claves de acceso, permitirá que solo las personas autorizadas puedan ingresar.
- Asignación de roles a los funcionarios de acuerdo a su ocupación.

Fiabilidad.

La aplicación está controlando los procesos que ha dado a conocer el director de tránsito de la unidad de tránsito municipal, y está apto para responder a incidentes.

Disponibilidad.

La aplicación estará disponible todas las 24 horas del día, sin embargo, esta será utilizada para funciones administrativas durante el día por lo miembros de la Unidad de tránsito.

Mantenibilidad.

La aplicación posee características que permiten al desarrollador hacer cambios, agregar cosas nuevas, o borrar funcionalidades innecesarias. El mantenimiento se llevará a cabo cada 6 meses, las 2 primeras versiones sin cargos adicionales, pasada de estas 2 versiones tendrá un costo económico.

Portabilidad.

Una de las ventajas de utilizar herramientas y lenguajes basados en software libre es, la portabilidad, el soporte debido a la gran cantidad de comunidades open source que existen.

- EL 80% de la aplicación está desarrollado con laravel y un 20% con Vue Js para el frontend y el dinamismo en el lado del cliente.
- Gracias a la programación basada en componentes de VueJs podemos tener cada pantalla como un componente de la web, que es reactivo y no requiere refrescar la página para actualizar su contenido.
- La persistencia es portable gracias al gestor de base de datos Mysql, es decir que puedo tenerlo en Windows y Linux.

1.3. Especificaciones técnicas.

- **Software:**

- Frontend: Html, CSS3, Javascript, Materialize css Sweet Alert, VUE JS.
- Backend: Laravel, Php, Mysql.
- SO: Windows o Linux.

- **Hardware:**

- Procesador Intel Core I3 4790k 4 Nucleos Velocidad 3.80g.
- Memoria Ram 8 Gb (2x4gb) Ddr3 667 Mhz.
- Disco duro 1TB libres (mínimo).

CAPÍTULO II

2. Desarrollo del prototipo.

2.1. Definición del prototipo.

Sistema web para el control de trámites de la unidad de tránsito del cantón las Naves, es un prototipo que permitirá automatizar todos los procesos internos que realizan, los cuales fueron recaudados en un diálogo y/o entrevista que se realizó con el director de la unidad de tránsito.

El presente sistema informático funcionará en el internet, para el cuál se lo desplegará a futuro en un hosting, en un subdominio del Gad Las Naves.

Para el desarrollo de este sistema informático se utilizará 2 frameworks uno para el frontend y otro para el backend, para este último haremos uso de laravel 5.4, un poderoso marco de trabajo escrito en el lenguaje de programación PHP, actualmente en su versión 5.6 pero para el desarrollo se hará uso de una versión inferior ya que el hosting nos provee una versión de php 5 y laravel 5.6 requiere php 7 como mínimo para funcionar correctamente.

Para el frontend se hará uso del framework Vue JS, un marco de trabajo escrito en Javascript, que nos permite construir aplicaciones dinámicas de lado del cliente, y que funcionen en una misma página sin que recarguen el navegador al momento de realizar la navegación, gracias a la programación reactiva basada en componentes, que es la filosofía que utiliza Vue JS, podemos lograr que al momento de que el usuario ingrese un nuevo trámite, se envíen estos datos a una Api RestFul construida en Laravel, y no se tenga que recargar la página, si no únicamente actualizar el componente que se ve afectado al momento de realizar la transacción, reduciendo el

tiempo de carga que el usuario tiene que esperar para que el servidor responda a la petición Http realizada en el cliente.

Para la parte de almacenamiento de los datos se hará uso del sistema gestor de base de datos Mysql, ya que la contratación del hosting que posee el GAD Las Naves nos provee únicamente el servicio de este SGBD.

2.2. Fundamentación teórica del prototipo.

Los sistemas web no dependen de otro complemento, además del navegador en el cliente para su funcionamiento, y por supuesto de una conexión a internet para acceder al sitio a través de una URL, a cualquier hora que el usuario desee.

Además que los sistemas al no depender de configuraciones extras en el equipo cliente, garantiza una mayor seguridad al sistema, ya que de este modo el usuario no tendrá acceso a archivos de instalación del sistema, ni configuraciones que tenga que realizar para acceder al mismo.

El sistema informático SIUT solucionará el problema de la mala administración de documentos tramitados en la Unidad de Tránsito de las Naves, y el usuario solo requerirá de un navegador y conexión a internet para poder acceder a la aplicación.

2.2.1. Metodologías.

La metodología de desarrollo de software tiene que ver con la forma en cómo se va llevando a cabo todos los avances y progresos de la aplicación durante un tiempo determinado, es decir debemos determinar cuál es la forma ágil que más nos sea conveniente para efectuar dicho proceso.

También involucra la diagramación, que para nuestro caso se hará uso de los diagramas UML, ya que nos permite conocer a través de los diagramas de clase,

cuáles son las entidades que participan en el sistema, o también los diagramas de caso de uso, en donde podemos detallar los usuarios que interactúan con el sistema y sus respectivos procesos.

Modelo de ciclo de vida de software.

El modelo de ciclo de vida de software a utilizarse será **Cascada**, porque este modelo nos ofrece diferentes fases por las cuales la aplicación debe pasar, y por el motivo de que no podemos seguir desarrollando si no hemos culminado una de las fases de desarrollo de software.

Durante el transcurso del tiempo dedicado para el desarrollo de la aplicación se realizó un diálogo con el director de la unidad de tránsito el Tecnólogo Benito López, quien dio a conocer todas las especificaciones del software y lo que desean automatizar, facilitándonos así la toma de requerimientos.

Para la parte de los diseños necesitareé prototipos visuales, para entregarle al director lo que voy a desarrollar, en el caso del front-end haré uso wireframes y mackups para las interfaces visuales de la aplicación.

Por otra parte, para los diferentes procesos que se llevará a cabo durante todo el desarrollo de la aplicación, haré uso de los diagramas UML, que me permitirán detallar los usuarios que interactúan con la aplicación, los diferentes módulos que tendrá mi aplicación al momento de codificarla.

Para el desarrollo de la aplicación se hará uso del framework laravel para el backend y del framework Vue JS para el front-end, los cuáles me permiten crear aplicaciones modernas, sofisticadas, escalables, y fácil de mantener.

Para las pruebas o test, laravel nos ofrece módulos para hacer pruebas unitarias en el código fuente, para verificar que no tenemos errores de sintaxis en el mismo, además se realizarán pruebas tanto en el backend como en el front-end, para el caso de este último utilizaremos webpack, que es un compilador de código css y js, para revisar la indentación del código en el front-end, revisar errores de sintaxis, entre otros errores de lógica en cuanto al funcionamiento de la aplicación.

Para la implementación del software, se desplegará la aplicación en un hosting de pago que posee el municipio del cantón Las Naves, para ello se creará un subdominio, y un usuario ftp por el cuál podré acceder a la configuración del cpanel para desplegar la aplicación desarrollada en laravel.

En cuanto al mantenimiento, hay que recalcar que la aplicación tendrá una versión inicial, que en este caso es la 1.0, y tendrá un mantenimiento de 6 meses, cuyos 2 primeros serán totalmente gratis a partir del tercer mantenimiento ya tendrá cargos adicionales.

Arquitectura.

Como cualquier plataforma web, la arquitectura está orientada a una comunicación bidireccional entre equipos clientes y servidores, permitiendo que cualquier equipo cliente autorizado, pueda comunicarse desde cualquier dispositivo con la aplicación.

La arquitectura de programación o también denominado patrón de diseño, que se empleará es el MVC (modelo, vista controlador), por defecto el framework laravel trabaja con este patrón de diseño, además de que está construido bajo una filosofía orientada a objetos haciendo que el código sea reutilizable.

En los modelos se especificará todas las reglas de negocio de nuestro sistema de acuerdo a los requerimientos previamente especificados en la etapa de análisis, como son las reglas de los usuarios, roles, trámites, permisos, requerimientos ciudadanos y configuración de perfiles, con sus respectivas relaciones entre las diferentes entidades.

Las vistas de la aplicación serán renderizadas por medio de la reactividad de cada componente creado en Vue JS, con la intención de mostrar una interfaz amigable, con un tiempo de carga muy pequeño, y vistas que puedan interactuar en tiempo real con el usuario, de manera que los funcionarios de la unidad de tránsito puedan editar, desactivar, activar, o dar de un alta a un documento tramitado sin necesidad de que recarguen el sitio para visualizar los cambios, pues cada componente posee su propio rendering y no depende del navegador, de esta manera evitamos que la aplicación sufra algún colapso al momento de estar en un ambiente de producción, debido al bajo performance o mala optimización del sistema.

Los controladores, que serán desarrollados a través del framework laravel de lado del servidor, van a contener toda lógica de programación, como, por ejemplo:

El controlador de los usuarios, donde se crearán los métodos para hacer un post, get, update, delete, que deberá resolver el Api Rest, al momento de que el frontend haga la petición por medio del protocolo Http.

El controlador de los clientes, de igual manera dicho controlador permitirá elaborar los métodos para crear, desactivar, activar, editar a los clientes, y por medio de una url se podrá acceder a cada método de dicho controlador, al momento de realizar la petición.

El controladores de trámites: Todos los controladores para los diferentes procesos que se va a realizar en el sistema, deberán ser capaces de dar ciertos permisos a los usuarios, para que el usuario administrador/director sea el único que tenga control total del sistema, mientras que los demás funcionarios tendrán roles de cosas más específicas de acuerdo a las transacciones que tengan que llevar a cabo.

Single Page Application (SPA).

La mayoría de las aplicaciones web hoy en día, tienen un único punto entrada, y para realizar la navegación, lo único que cambia es el componente que se desea mostrar, evitando refrescar el navegador. Para este sistema informático se hará uso de las SPA con el fin de obtener un alto performance en el cliente, evitando que la aplicación se cuelgue al momento de hacer las peticiones http a nuestro Api Restful.

Con el framework Vue JS podemos lograr construir SPA de una forma fácil, rápida, sin mucho esfuerzo, gracias al router que maneja Vue, podemos crear navegación y a los Vue component, podemos crear componentes para la gestión de los diferentes trámites que realiza la Unidad de tránsito de las Naves.

Las single page application nos ofrecen una experiencia de usuario muy agradable, gracias a que el código no es complejo, los datos son muy livianos y la transferencia y procesamiento de los mismos es sumamente rápido, de manera que podemos establecer una comunicación cliente servidor muy fluida. (Alberto Blanch, 2017)

Las SPA nos libran bastante de inconvenientes al momento de realizar la optimización del sistema para ponerlo en un ambiente de producción, ya que, al no depender del navegador para actualizar el contenido de la página, estamos evitando el uso de ese recurso que nos proveen todos los browsers de lado del cliente, pues para

actualizar el contenido hacemos uso del rendering del componente, que nos ofrece el framework Vue JS.

Beneficios y desventajas.

Al realizar cualquier aplicación web, se debe tener en cuenta que los aspectos son muchos, pues por lo general siempre es complicado cumplir con las expectativas del usuario, con respecto al funcionamiento del sistema, esto por lo general depende mucho de la toma de requerimientos y el análisis que se haga en dicha etapa.

También tenemos la parte del desarrollo en si del sistema, cuya etapa requiere el uso de una buena metodología y un buen patrón de diseño, para optimizar el código, reutilizarlo, y que sea fácil de mantener.

Para nuestro caso el patrón de diseño a utilizar es el MVC, pero al igual que cualquier arquitectura de programación tiene sus ventajas y desventajas:

- Ventajas.
 - Separa la lógica de negocio de la parte visual de la aplicación.
 - Es más fácil el manejo de los errores.
 - Permite crear código escalable y fácil de mantener.
 - Se puede reutilizar el código.
- Desventajas.
 - Gran cantidad de archivos, por cada capa, modelos vistas, y controladores.
 - La separación de la lógica de negocio del frontend de la aplicación genera una complejidad mayor en el sistema, a nivel de código fuente, de manera que si un programador no conoce esta arquitectura de programación, tendrá que auto prepararse.

- Requiere buenos conocimientos en POO y buen manejo del lenguaje en el cuál se valla a trabajar con esta arquitectura.

Tecnologías.

El sistema informático será web, debido a los distintos puntos de conexión a internet que poseen los equipos de cómputo en la Unidad de Tránsito, impidiendo así que la aplicación sea de escritorio para su funcionamiento en red. Además que la aplicación de escritorio sería dependiente de ciertas configuraciones y ficheros que por lo general van a permanecer en un equipo (no servidor) principal, dejando así vulnerable el sistema, a que cualquier usuario pueda acceder a estas configuraciones y corromper algún archivo importante.

Las tecnologías para el desarrollo del sistema web SIUT son:

Html.

Html es un lenguaje marcado de hipertexto, que nos permite especificar el orden del contenido de nuestra web, utilizando las etiquetas, y además ciertas filosofías como por ejemplo la semántica, que no es más que hacer uso de las etiquetas con nombres descriptivos de acuerdo al contenido que deseamos mostrar, en nuestro sitio web. (Herrera, 2015)

Desde luego para el desarrollo del sistema web SIUT haremos uso de Html, para realizar la maquetación de la web, con las diferentes secciones, artículos, y contenido que la componen.

Css.

Los archivos Css son hojas de estilos que mediante un lenguaje que entiende el navegador, especificamos el diseño y la apariencia que deseamos mostrar en los archivos de nuestro sitio web. (Moreno, 2018)

JavaScript.

Javascript es un lenguaje de programación que entienden los navegadores, con el cual podemos realizar tareas complejas en una página web, como por ejemplo, mostrar contenido en tiempo real, al momento de que se hace la petición, mostrar mapas, etc. (Perilla, 2018)

Para el caso de nuestro sitio web, desde luego utilizaremos javascript, para las validaciones que se deben realizar en el lado del frontend de nuestro sistema, como por ejemplo, que los campos no estén vacíos, o cuando hagamos una petición y los datos que enviamos sean incorrectos, el servidor nos devuelve un error, y ese error lo capturamos con programación en el lenguaje javascript y lo mostramos al usuario mediante algún tipo de alerta.

Axios.

Esta es una librería Javascript, que nos permite crear programación asíncrona basada en promesas, básicamente Axios nos permite realizar peticiones Http como si estuviéramos usando Ajax. (Caules, 2016)

Para nuestro sistema informático, se hará uso de Axios para las peticiones http que realicemos a nuestra Api Restful, por motivo de que axios, nos ahorra mucho código a la hora hacer peticiones como un get, post, put, delete, y además de su sencillez, y la poca complejidad que posee esta librería, razón por la cual, podemos programar de una manera rápida y sencilla en el cliente.

Vue JS.

Es un framework progresivo con un parecido a la librería jQuery, que nos permite construir aplicaciones web, de lado del front-end, se enfoca en únicamente en la capa de vistas, en donde cada sección de la aplicación es un componente, que a su vez puede tener componentes hijos, dando así a conocer una filosofía

comúnmente conocida como programación basada en componentes, con la cuál es posible desarrollar una SPA o una SR (Server Rendering) (Azaustre, 2017)

En el sistema SIUT, Vue JS nos permitirá construir todos los componentes clientes con los que el usuario interactuará para manejar el sistema, además de poder controlar las rutas de lado del front-end, y que la navegación sea sobre la misma página, evitando así que el navegador se recargue al momento de navegar por las diferentes opciones del menú.

SPA.

De sus siglas en inglés single page application, las SPA son aplicaciones que corren que poseen un único punto de entrada, y en donde todas las páginas se muestran en una sola sin tener que recargar el navegador. (Alvarez, 2016)

Desde luego gracias al framework Vue JS podemos crear aplicaciones SPA cuyos componentes funcionan en una misma página, mejorando así el performance de la aplicación y rendimiento, y reduciendo los tiempos de carga de la misma.

Heroku.

Heroku es una de las plataformas como servicio, más utilizadas a día de hoy, ya que el enfoque que posee para realizar despliegue de aplicaciones, ya sea en ambientes de pruebas, o aplicaciones empresariales, es bastante sencillo y eficiente. Solo necesitas especificar el lenguaje de backend que requieres, y la base de datos que vas a usar para tu proyecto, y listo, puedes despreocuparte del despliegue, y centrarte en el desarrollo del sistema. (Celis, 2017)

Laravel.

Laravel es un framework muy potente creado en el 2011 y que posee una gran variedad de tecnologías similares al mismo, como por ejemplo: Django, Ruby on rails, entre otros. (García, 2015)

El framework laravel trabaja de una forma frontal hacia la ruta y de la ruta hacia el controlador. Aquí solo existe un archivo como punto de entrada de la aplicación, este archivo es el index.php y está en la carpeta public, y que por supuesto es el directorio al cual se debe poder acceder desde la web, cuando la aplicación este en producción. (Laguna, 2016)

Básicamente lo que nos ofrece laravel es la posibilidad de crear aplicaciones modernas y robustas. Actualmente el framework está en su versión 5.6, esta versión requiere de PHP 7 como mínimo, sin embargo la mayoría de servicios de hosting nos proveen una versión de PHP limitada de hasta 5.6, es por esto que se ha optado por utilizar la versión 5.4 del framework que requiere de PHP 5, ya que es la versión más común en la mayoría de hosting, y así no tener inconvenientes al momento de querer realizar un despliegue.

Con laravel gestionaremos la autenticación de usuarios, manejo de rutas del lado del backend y la creación del servicio Restful.

ORM.

Anteriormente para establecer una conexión a una base de datos y ejecutar diferentes transacciones, debíamos hacer uso de un lenguaje de programación, como puede ser Java o PHP, y de objetos especializados (Generalmente de tipo Connection), con los cuáles se podía ejecutar sentencias SQL hacía la base de datos, sin embargo nos trae algunos inconvenientes a la hora de desarrollar, pues teníamos que preparar las consultas o Query de bases de datos, antes de ejecutarlas

a través de los objetos en la aplicación, independientemente del lenguaje en el cuál se esté desarrollando. (Alarcón, 2018)

Con los ORM, cuyas siglas en inglés significan Object Relational Mapping, no es necesario escribir código SQL, pues los ORM ya lo hacen por nosotros, al momento de mapear la base de datos, y que luego sea posible acceder a la data mediante objetos desde la aplicación, sin la necesidad de ejecutar Query.

Laravel por defecto integra un ORM, que nos permite mapear toda la base de datos, de manera que no nos tengamos que preocupar por la persistencia, si no únicamente por la desarrollo de la aplicación, o para el caso del sistema informático SIUT, las transacciones que realizará nuestra Api Restful.

Eloquent.

Eloquent es un ORM que viene integrado por defecto en el framework laravel, que nos permite manejar de forma sencilla todas transacciones de base de datos de nuestro sistema informático sin tener que usar una sola sentencia de código SQL. (Ochoa, 2015)

Para el desarrollo del SIUT, necesitaremos obviamente una base de datos, y para ellos eloquent nos viene de maravilla, puesto que podemos ejecutar transacciones sin lanzar SQL desde la aplicación, agilizando así nuestro desarrollo, además de que nos brinda una capa extra de seguridad, pues nunca mostramos esas sentencias en el código backend en el servidor, si no que dejamos que el ORM encapsule todos esos objetos y se comuniquen con nuestra base de datos.

Apache.

Es un servidor web HTTP de código abierto y multiplataforma muy robusto, que nos permite correr aplicaciones web, creadas en tecnologías como PHP, o simplemente páginas estáticas. (Cases, 2014)

Para el desarrollo del sistema SIUT se hará uso de apache, para correr nuestra aplicación en laravel ya en un ambiente de producción, ya que en local, laravel posee un servidor interno conocido como artisan, pero esto por supuesto no es recomendable usar al momento de desplegar nuestra aplicación, pues artisan fue únicamente implementado en el framework para ambientes de desarrollo.

Api.

Proviene de las siglas Application Programming Interface, y nos permite establecer comunicación entre 2 o más aplicaciones, que pueden estar en un mismo equipo. (Cruz, 2018)

2.3. Objetivos de la propuesta.

2.3.1. Objetivo general.

Desarrollar un sistema informático que permita llevar un control de todos los trámites realizados en la Unidad de tránsito de Las Naves.

2.3.2. Objetivos específicos.

- Realizar el respectivo diagnóstico de los requerimientos de software en la Unidad de tránsito de Las Naves.
- Diseñar los diagramas UML que nos permitirá conocer los procesos, actores y entidades que tendrá nuestro sistema, y así poder llevar a cabo el desarrollo del mismo.
- Realizar un plan de evaluación del sistema informático SIUT.

2.4. Diseño del prototipo.

Los usuarios del sistema serán los mismos funcionarios de la Unidad de Tránsito de Las Naves, quienes deberán ingresar la información de los clientes, y el registro del control de documentos que ya han sido tramitados, de acuerdo al rol que el usuario tenga asignado.

El sistema web mejorará la calidad de atención al cliente, y disminuirá el tiempo de respuesta para obtener información con respecto a la cantidad de documentos que se han tramitados, en una fecha determinada por el usuario.

Se utilizará las tecnologías adecuadas para que el sistema web tenga una interfaz amigable y sea fácil su uso, por otra parte los datos serán enviados de lado del frontend a una Api Restful, que será el servicio encargado de recibir esos datos y enviarlos a almacenar en la base de datos.

2.4.1. Análisis.

En el diálogo establecido con el director de tránsito Benito López, dio a conocer los siguientes requerimientos:

Gestión de usuarios: El sistema deberá permitir añadir usuarios, y asignarles un rol, los únicos roles disponibles serán, digitador, director, asistente administrativo, cabe recalcar que el director es el mismo usuario administrador.

Configuración de perfiles: Es un requisito no tan importante, sin embargo el usuario debe poder cambiar su información personal, como nombres, contraseñas, nombres de usuario o correo, fotos de perfil y de portada.

Ingreso de clientes: Los usuarios independientemente al rol que pertenezcan pueden ingresar clientes, es decir sus datos personales, como la cédula y sus nombres.

Ingreso de trámites: Los usuarios con el rol de digitador podrán ingresar los trámites, que son revisión vehicular, transferencia de dominio, y cambio de servicios, cada trámite debe tener un número de adhesivo el cuál lo identificará, y la fecha de la transacción de ese trámite.

Ingreso de requerimientos ciudadanos: los usuarios con el rol de asistente administrativo deberán estar autenticados para poder ingresar un nuevo requerimiento, y el cliente deberá presentar su documentación, la misma que será escaneada y subida a la plataforma como un archivo pdf junto con los datos personales del cliente.

Se requiere generar informes de la cantidad de trámites realizados ya sea para revisión vehicular, transferencia de dominio y cambio de servicios, de acuerdo a un criterio de búsqueda.

En el dashboard del inicio de la aplicación se mostrará unos gráficos estadísticos de barra donde se visualizará la cantidad de trámites realizados en los últimos meses.

2.4.2. Diseño. UML.

A continuación se muestra el modelado UML del proyecto.

El UML (Lenguaje de Modelado Unificado) es una especificación orientada a objetos y que está conformada por diferentes diagramas, para la etapa de diseño, y posteriormente llegar a la etapa de desarrollo. (Fernández, 2018)

Además del diagrama conceptual, entidad relación, y modelo relacional de la base de datos del sistema.

Diagrama de clases.

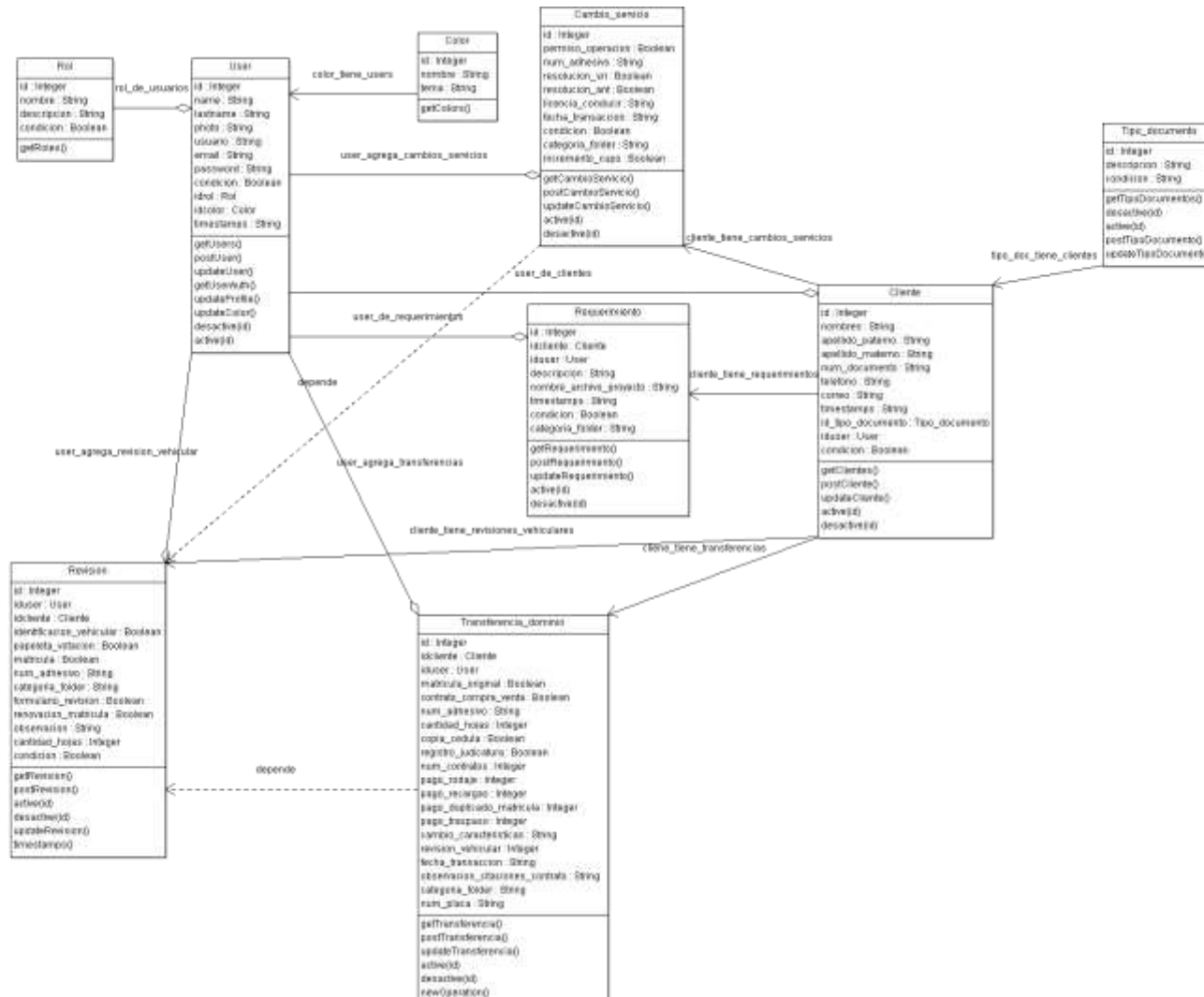


Gráfico 1: Diagrama de clases. Desarrollado por: Jorge Isrrael Diaz Montoya.

Diagrama de caso de uso

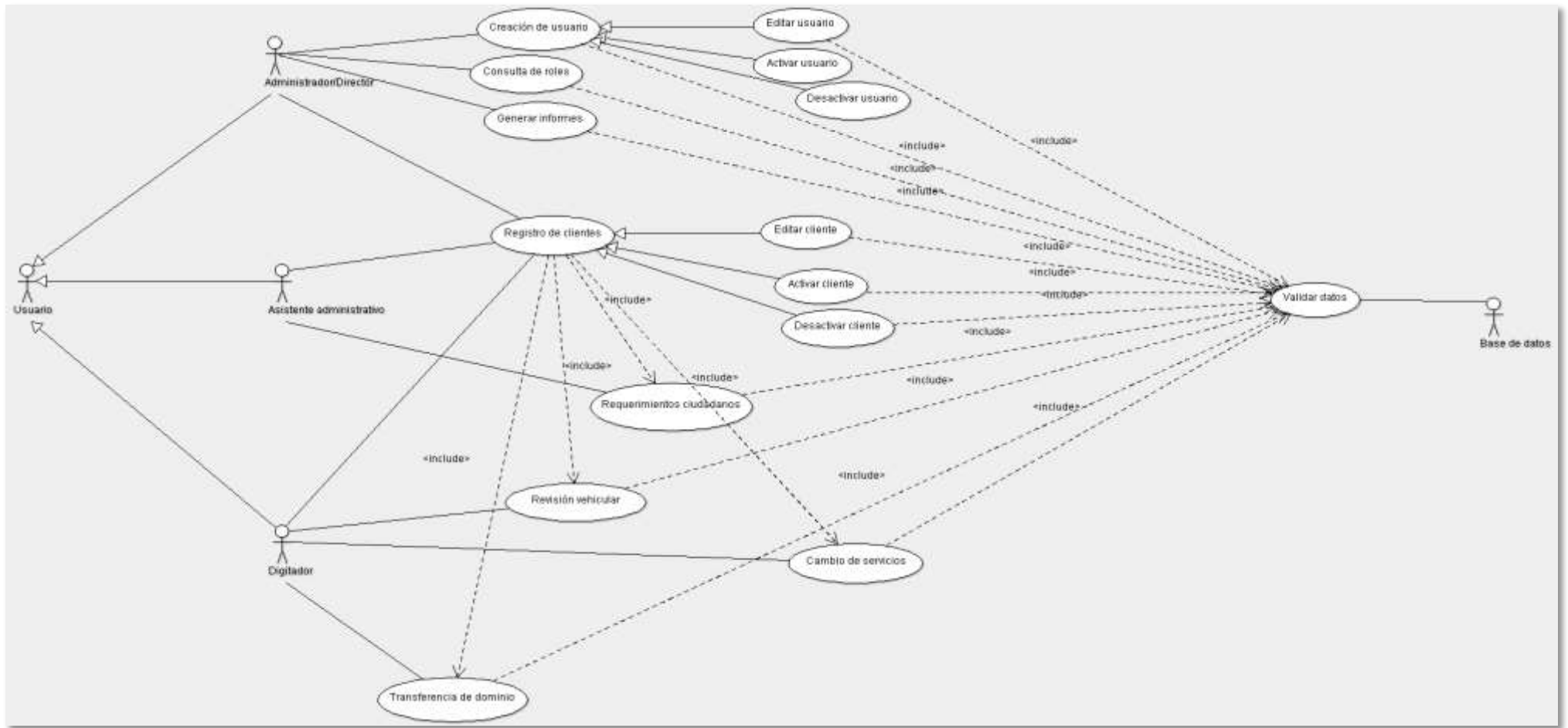


Gráfico 2: Diagrama de caso de uso del sistema. Desarrollado por: Jorge Isrrael Diaz Montoya.

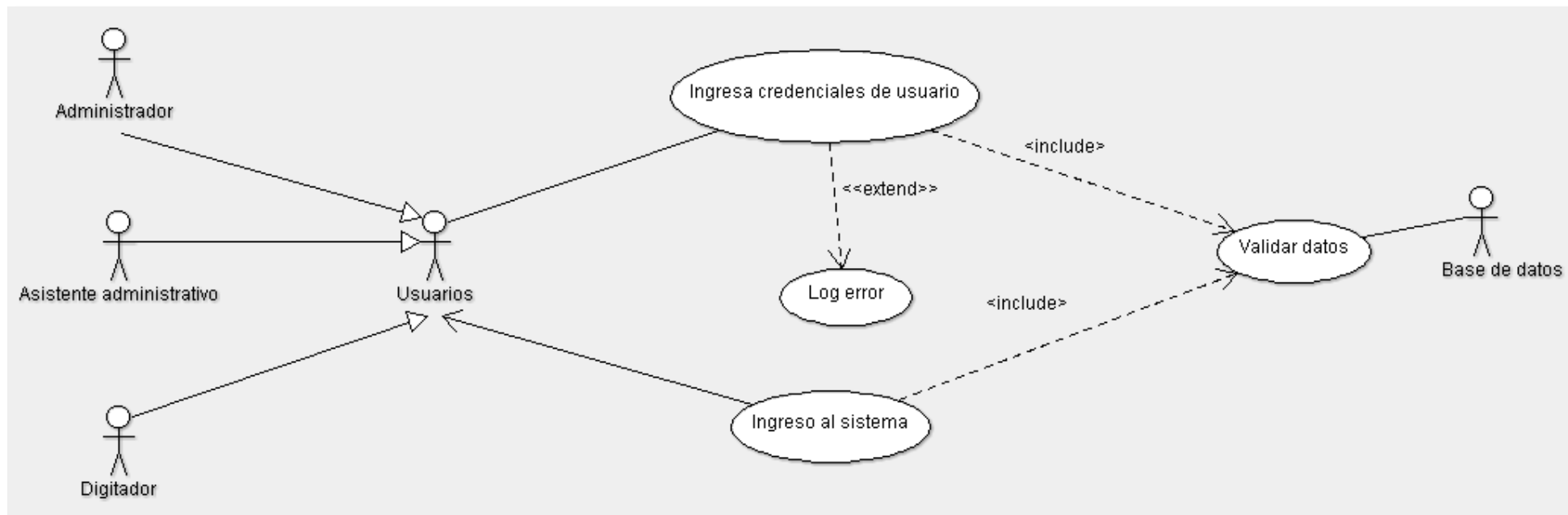


Gráfico 3: Diagrama de caso de uso de inicio de sesión. Desarrollado por: Jorge Isrrael Diaz Montoya.

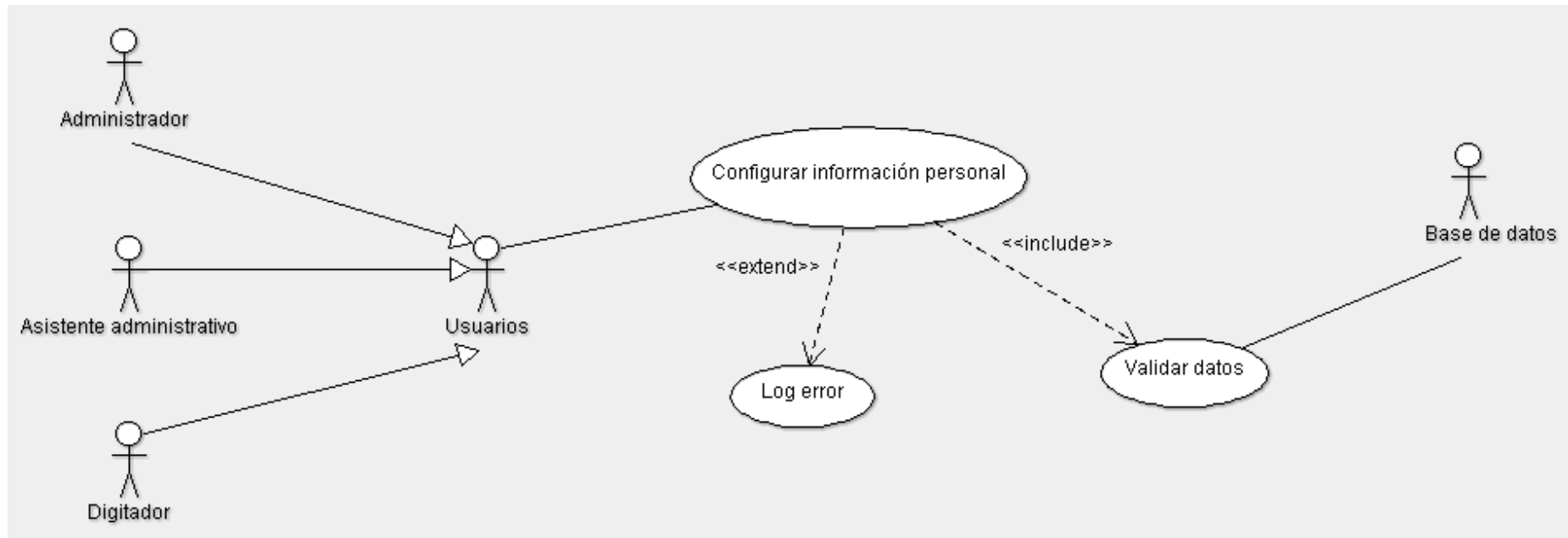


Gráfico 4: Diagrama de caso de uso de configuración del perfil de usuario. Desarrollado por: Jorge Isrrael Diaz Montoya.

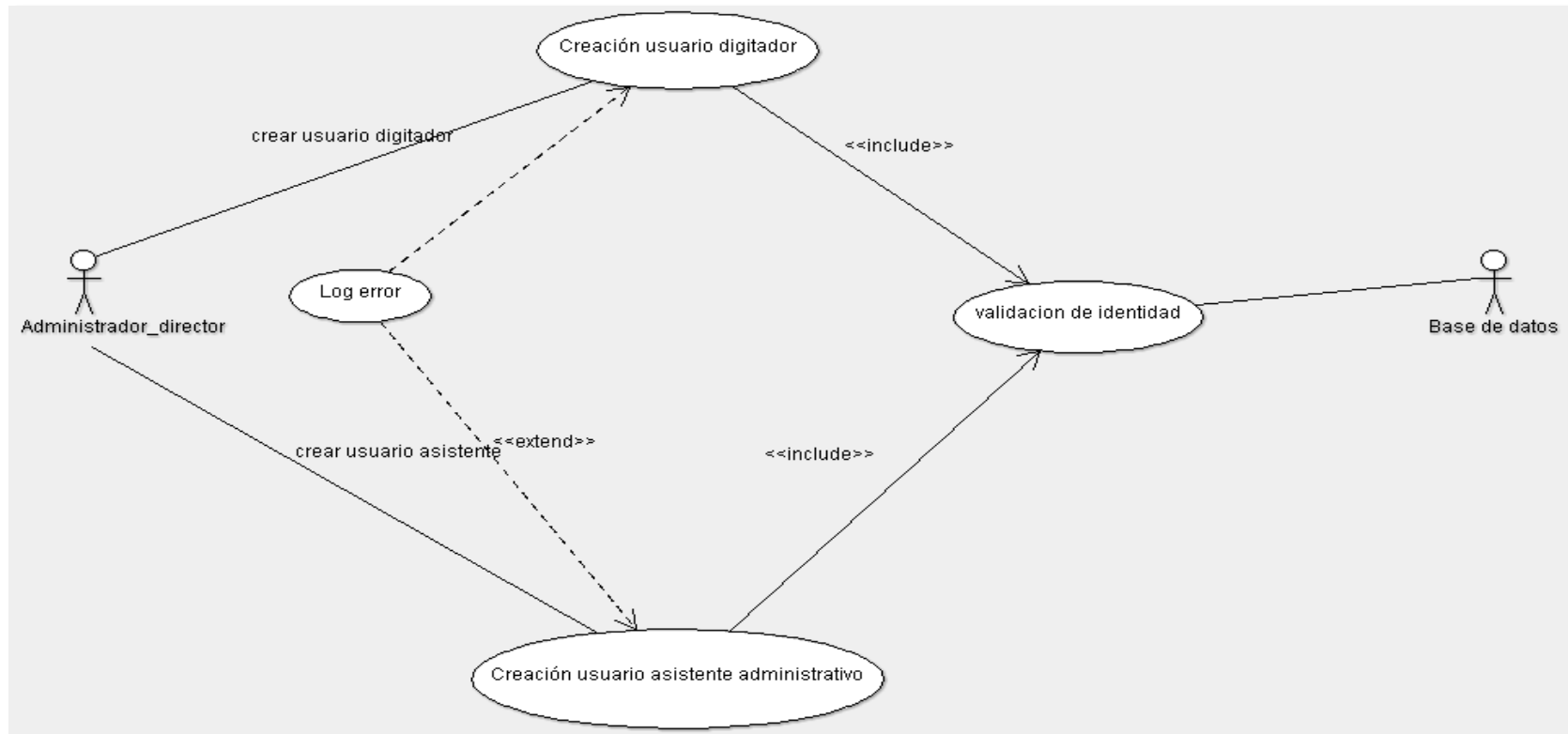


Gráfico 5: Diagrama de caso de uso del administrador. Desarrollado por: Jorge Isrrael Diaz Montoya.

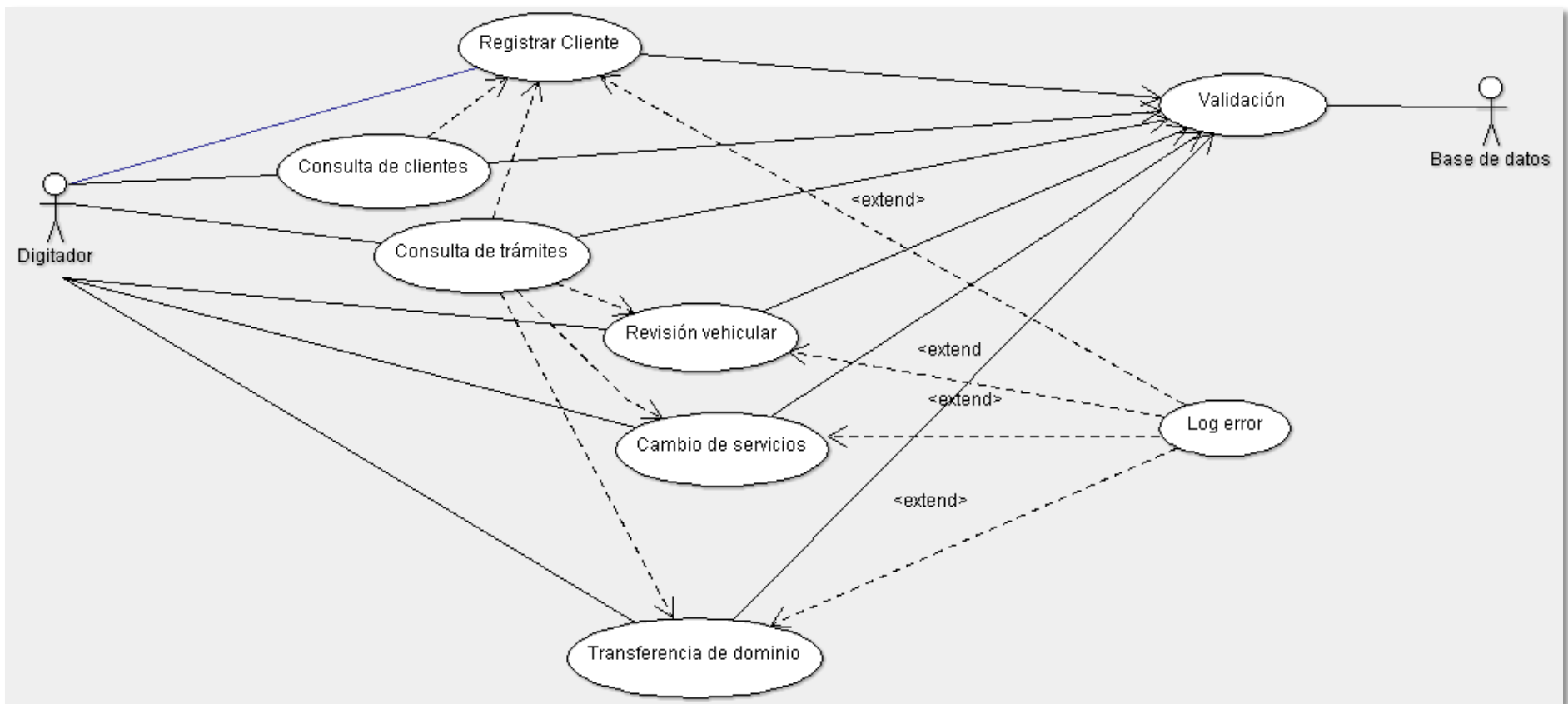


Gráfico 6: Diagrama de caso de uso del digitador. Desarrollado por: Jorge Isrrael Diaz Montoya.

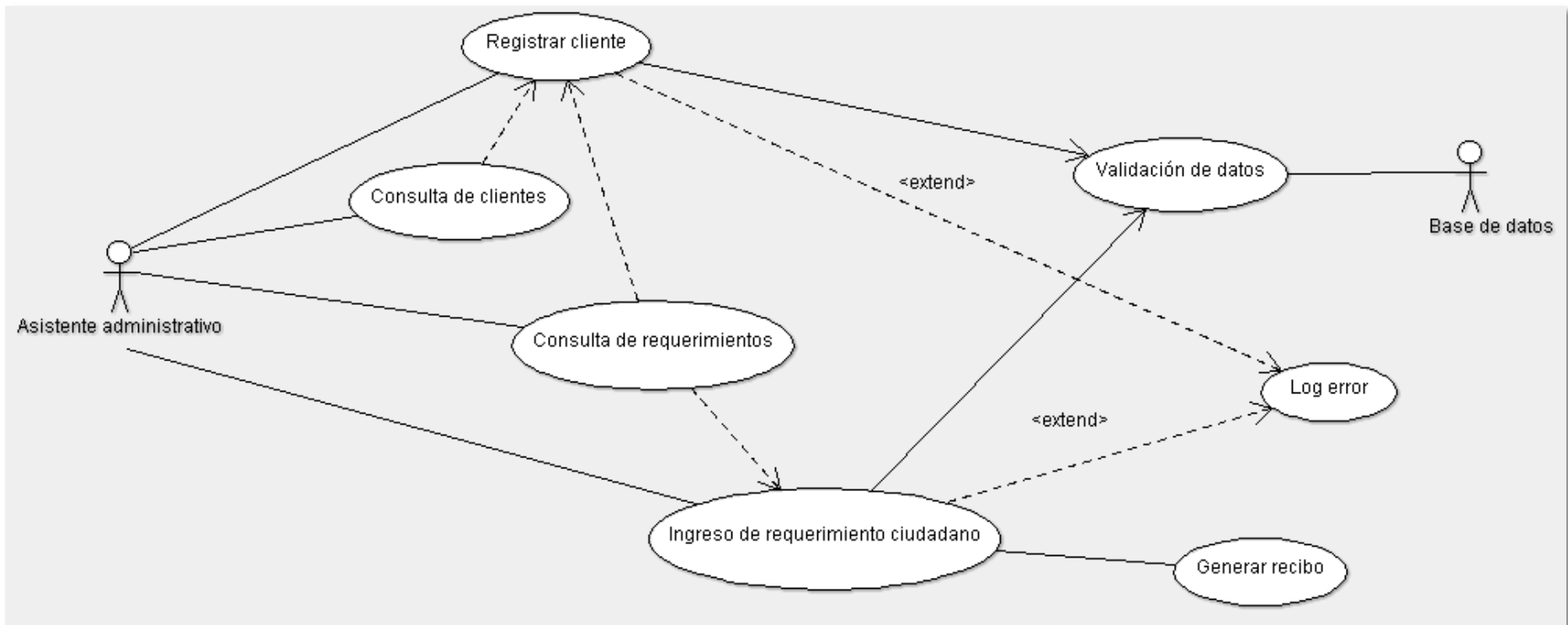


Gráfico 7: Diagrama de caso de uso del asistente administrativo. Desarrollado por: Jorge Isrrael Diaz Montoya.

Diagrama de actividades.

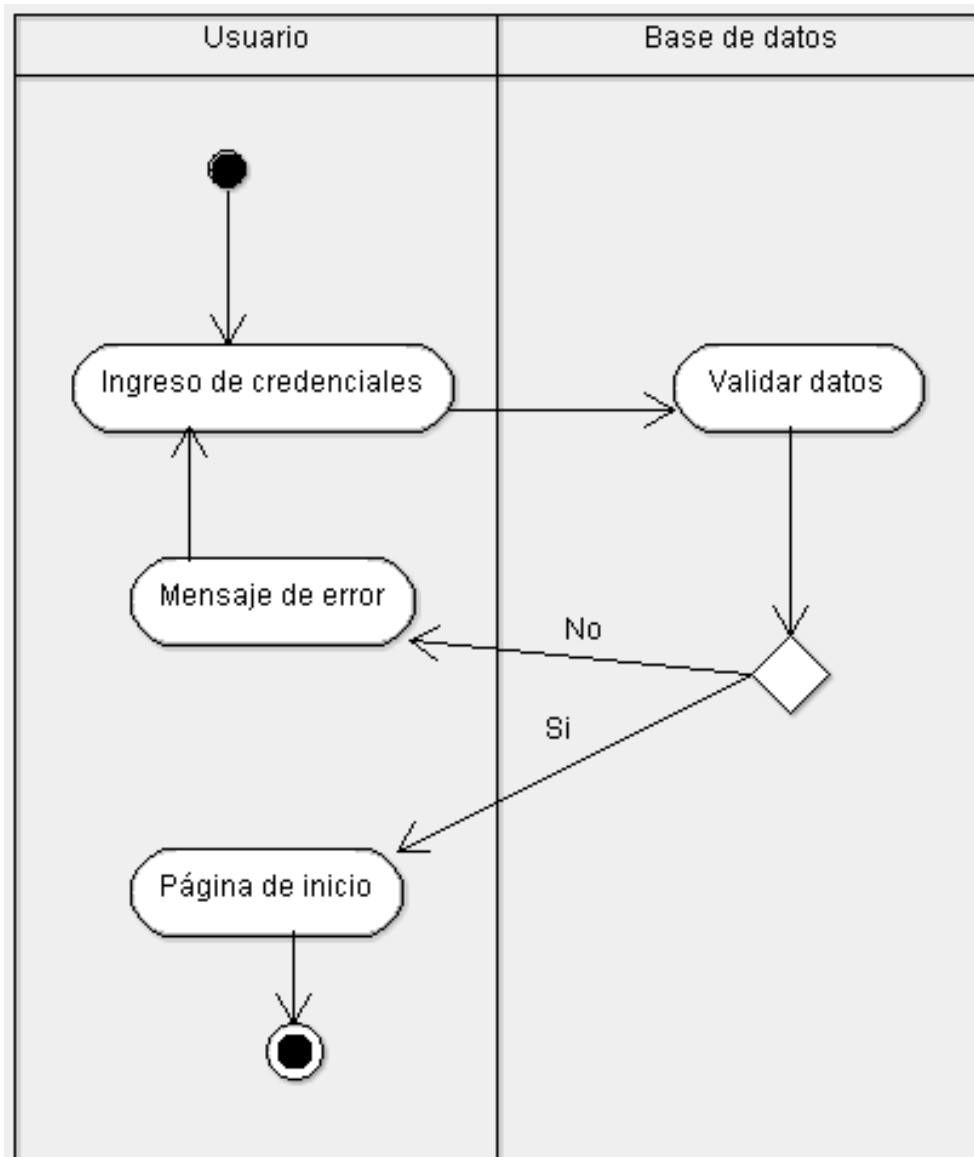


Gráfico 8: Diagrama de actividades de inicio de sesión.

Desarrollado por: Jorge Isrrael Diaz Montoya.

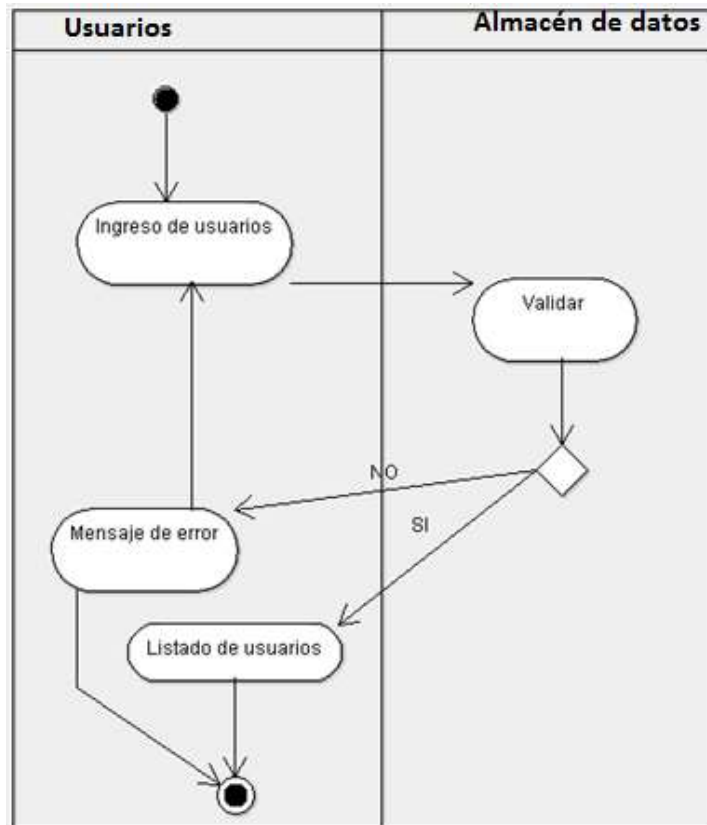


Gráfico 9: Diagrama de actividades de ingreso de usuarios.
Desarrollado por: Jorge Isrrael Diaz Montoya.

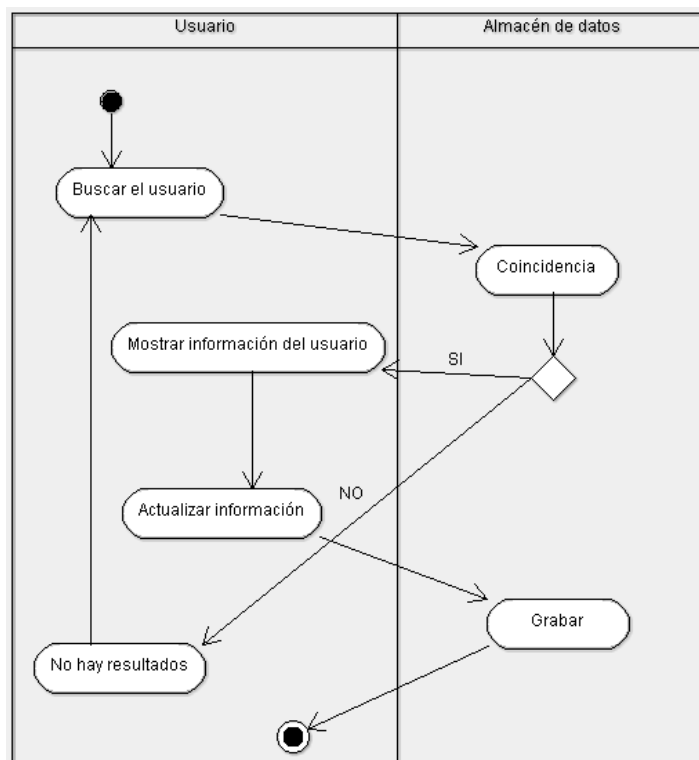


Gráfico 10: Diagrama de actividades para editar usuarios.
Desarrollado por: Jorge Isrrael Diaz Montoya.

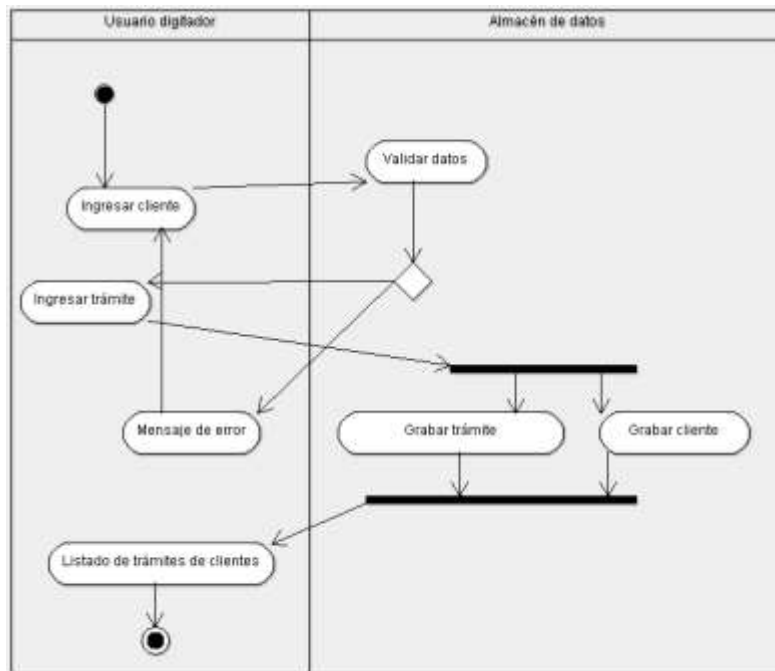


Gráfico 11: Diagrama de actividades para ingreso de trámites de clientes.
Desarrollado por: Jorge Isrrael Diaz Montoya.

Diagrama de secuencias.

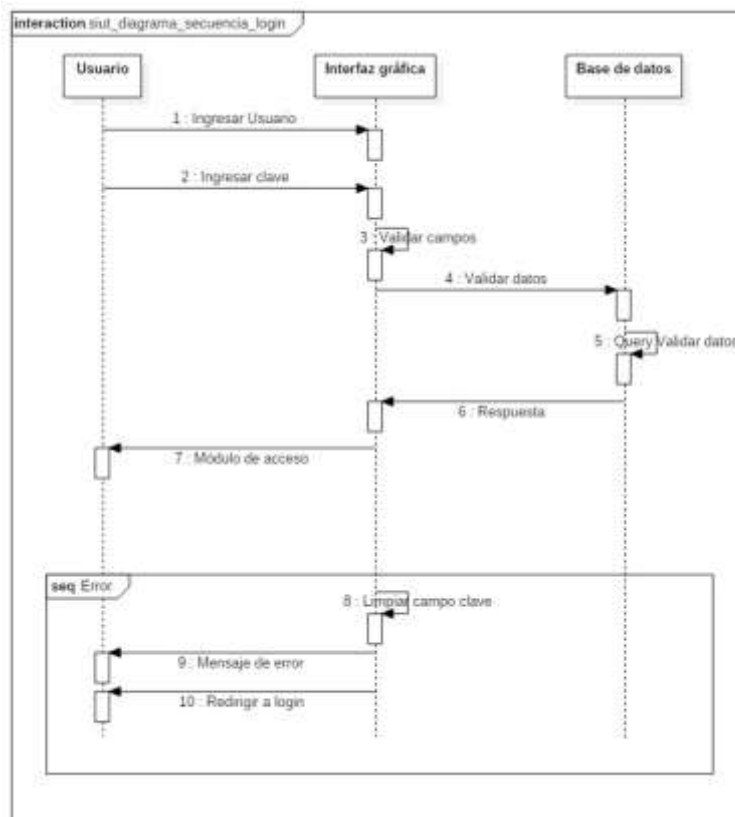


Gráfico 12: Diagrama de secuencia para autenticar usuarios.
Desarrollado por: Jorge Isrrael Diaz Montoya.

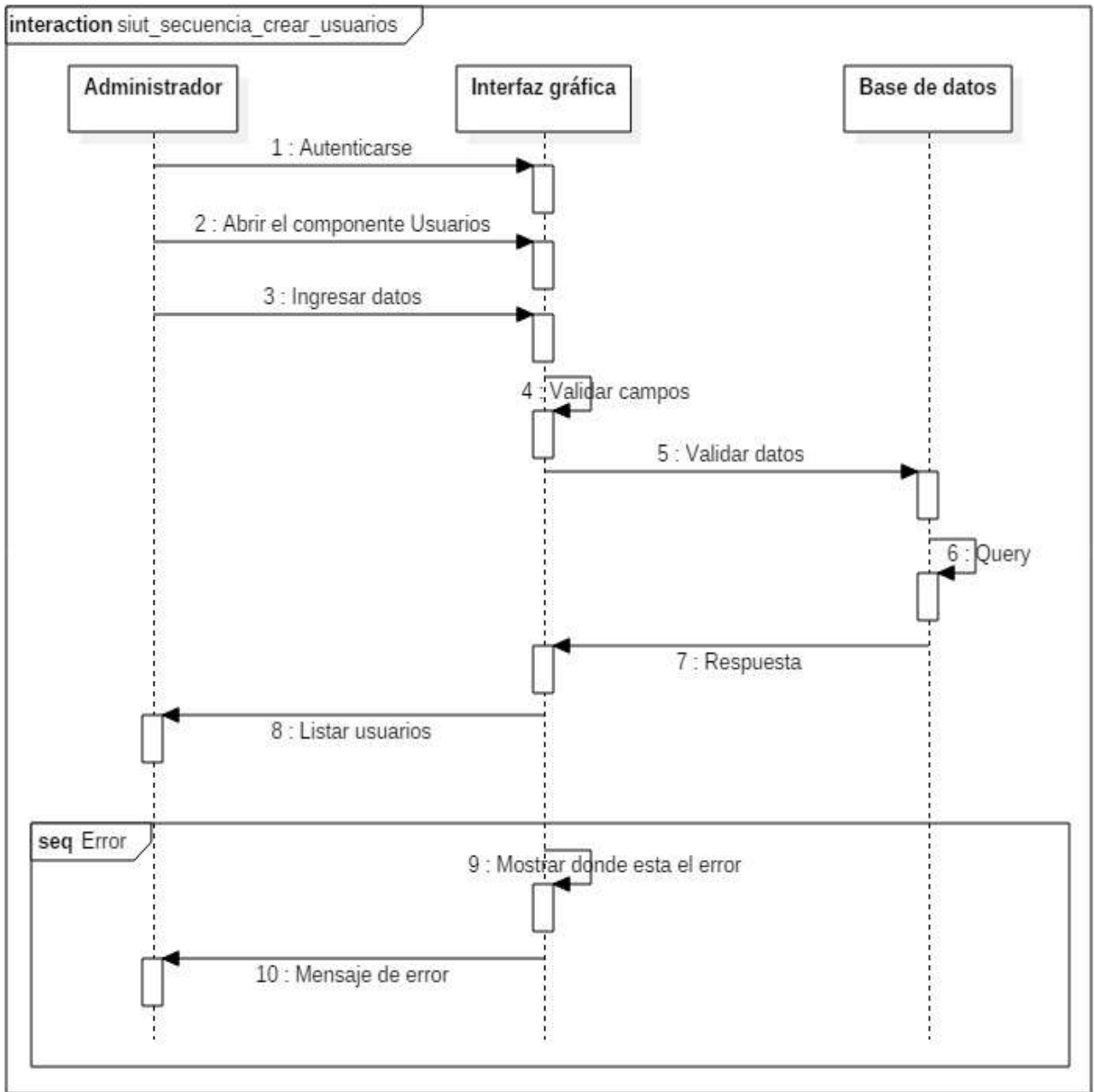


Gráfico 13: Diagrama de secuencia para crear usuarios. Desarrollado por: Jorge Isrrael Diaz Montoya.

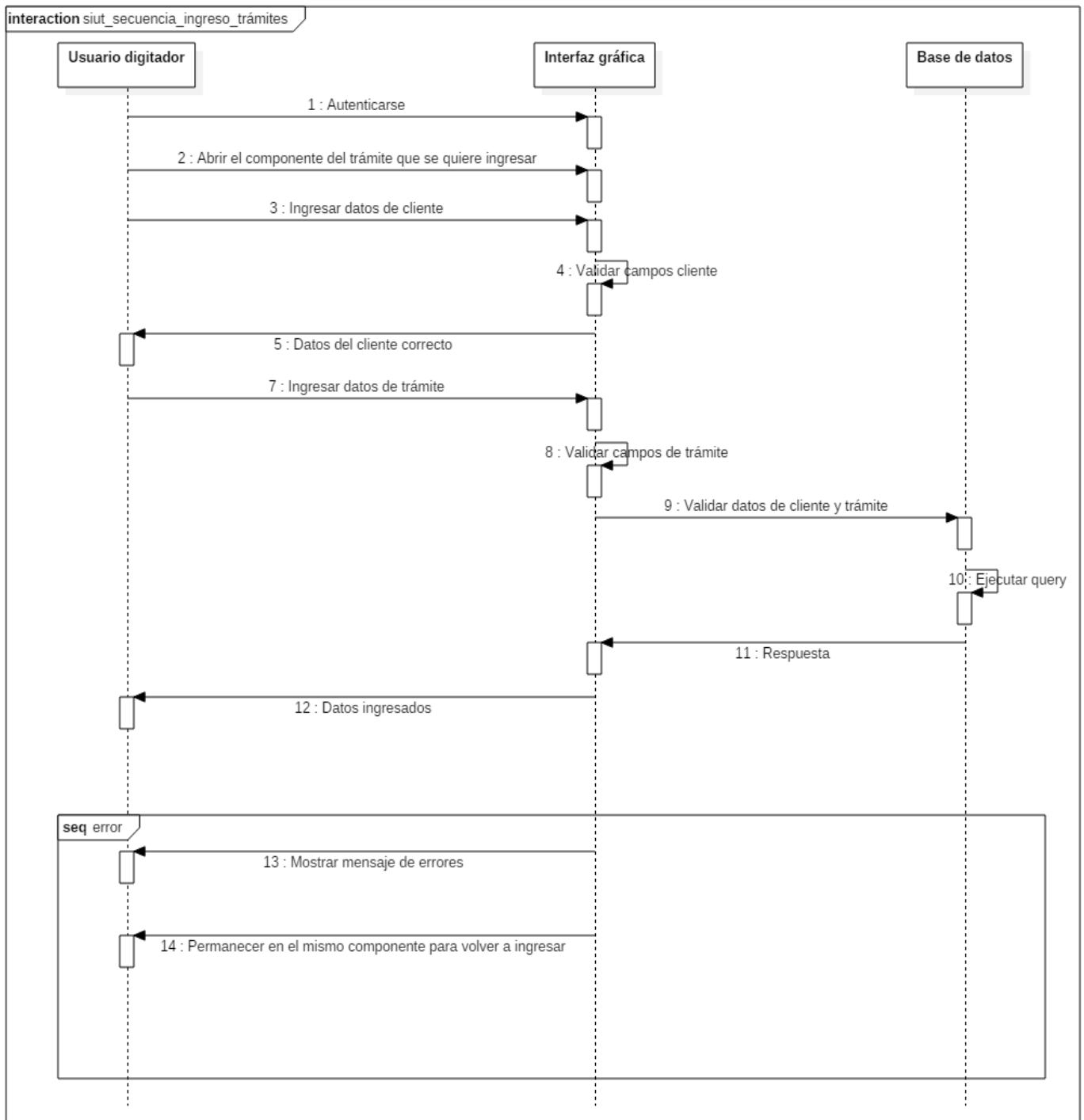


Gráfico 14: Diagrama de secuencia para ingreso de trámites. Desarrollado por: Jorge Isrrael Diaz Montoya.

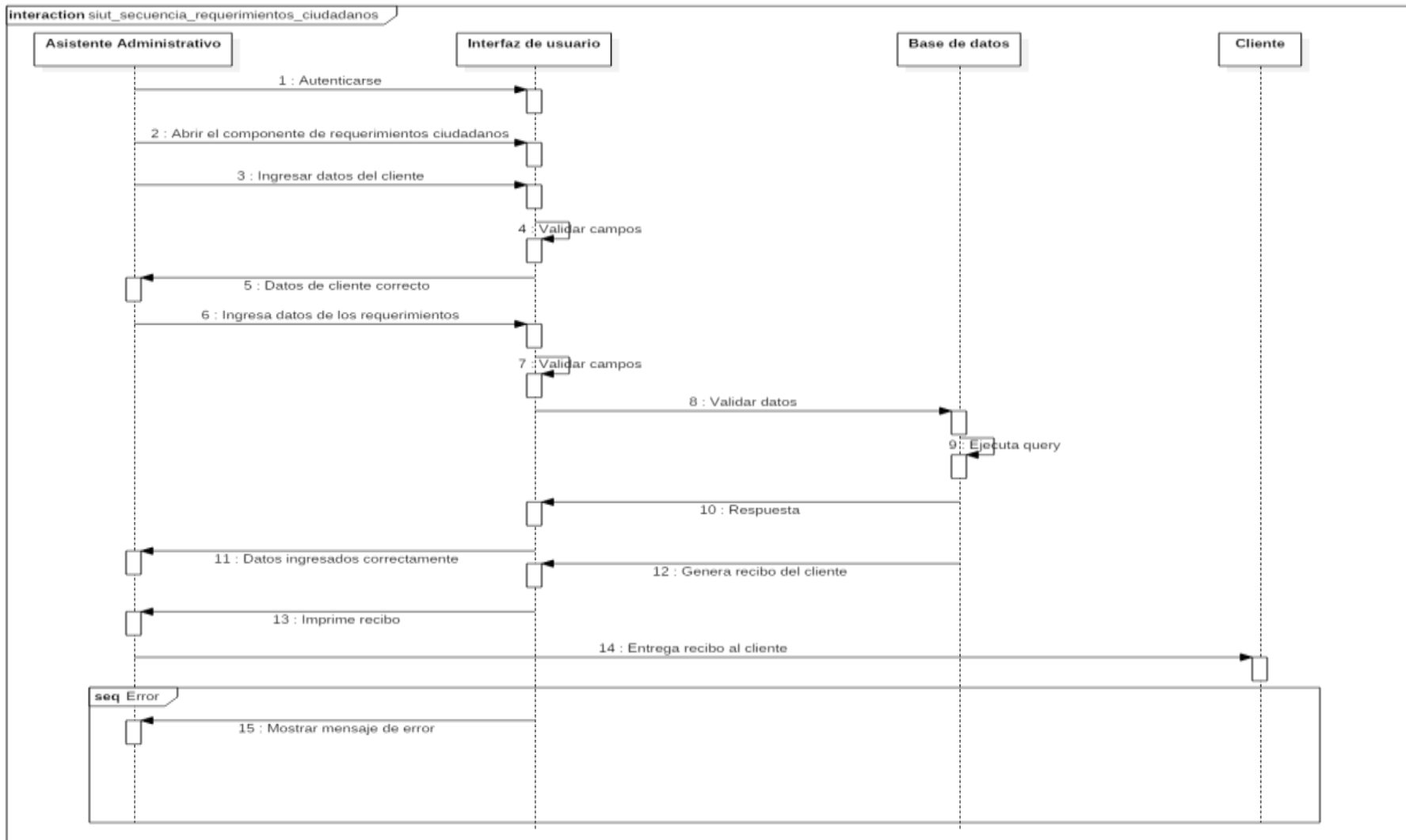


Gráfico 15: Diagrama de secuencia para control de requerimientos ciudadanos de clientes. Desarrollado por: Jorge Isrrael Diaz Montoya.

Modelo conceptual.

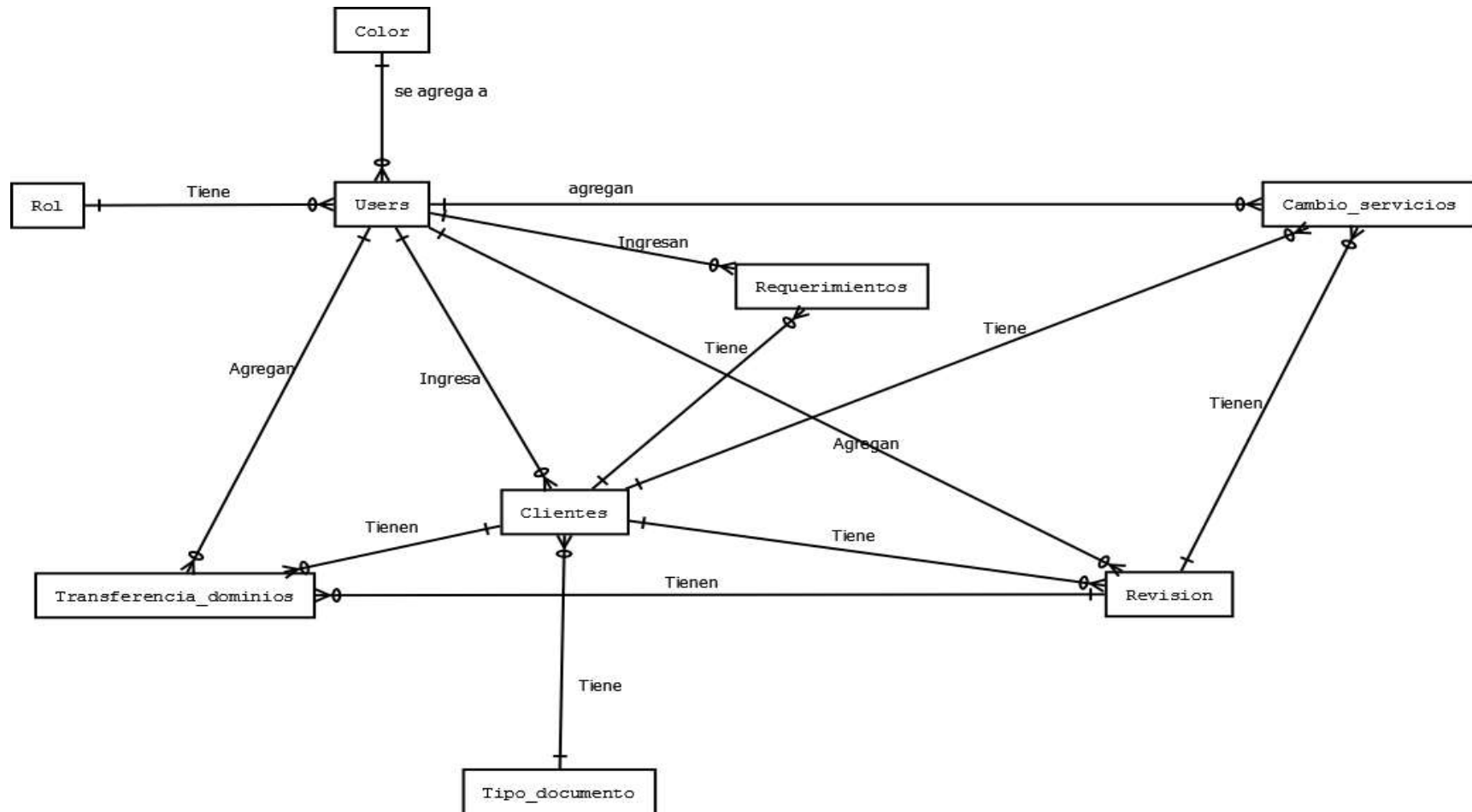


Gráfico 16: Modelo conceptual de la base de datos. Desarrollado por: Jorge Isrrael Diaz Montoya.

Modelo Entidad relación.

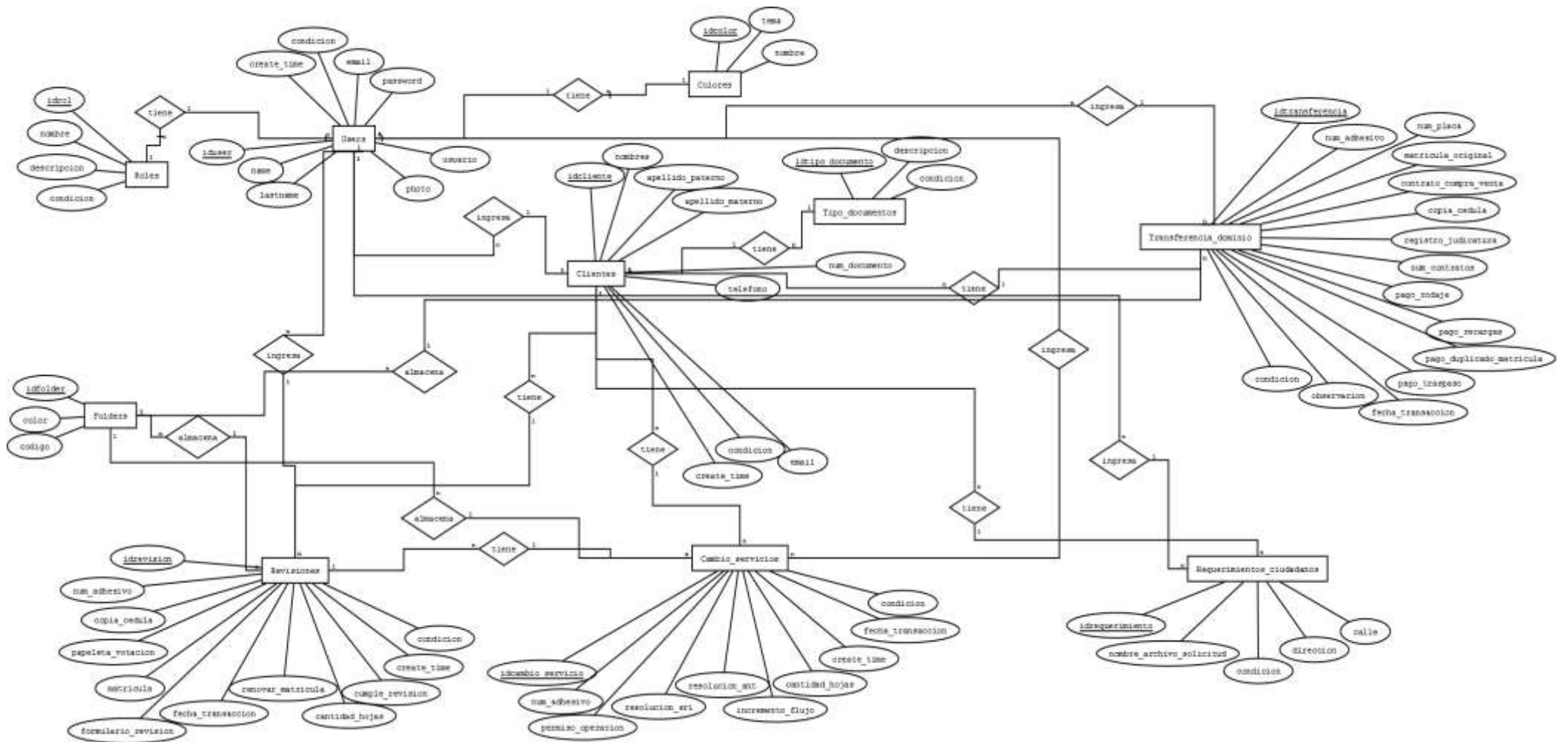


Gráfico 17: Diagrama entidad relación. Desarrollado por: Jorge Isrrael Diaz Montoya.

Modelo relacional de base de datos.

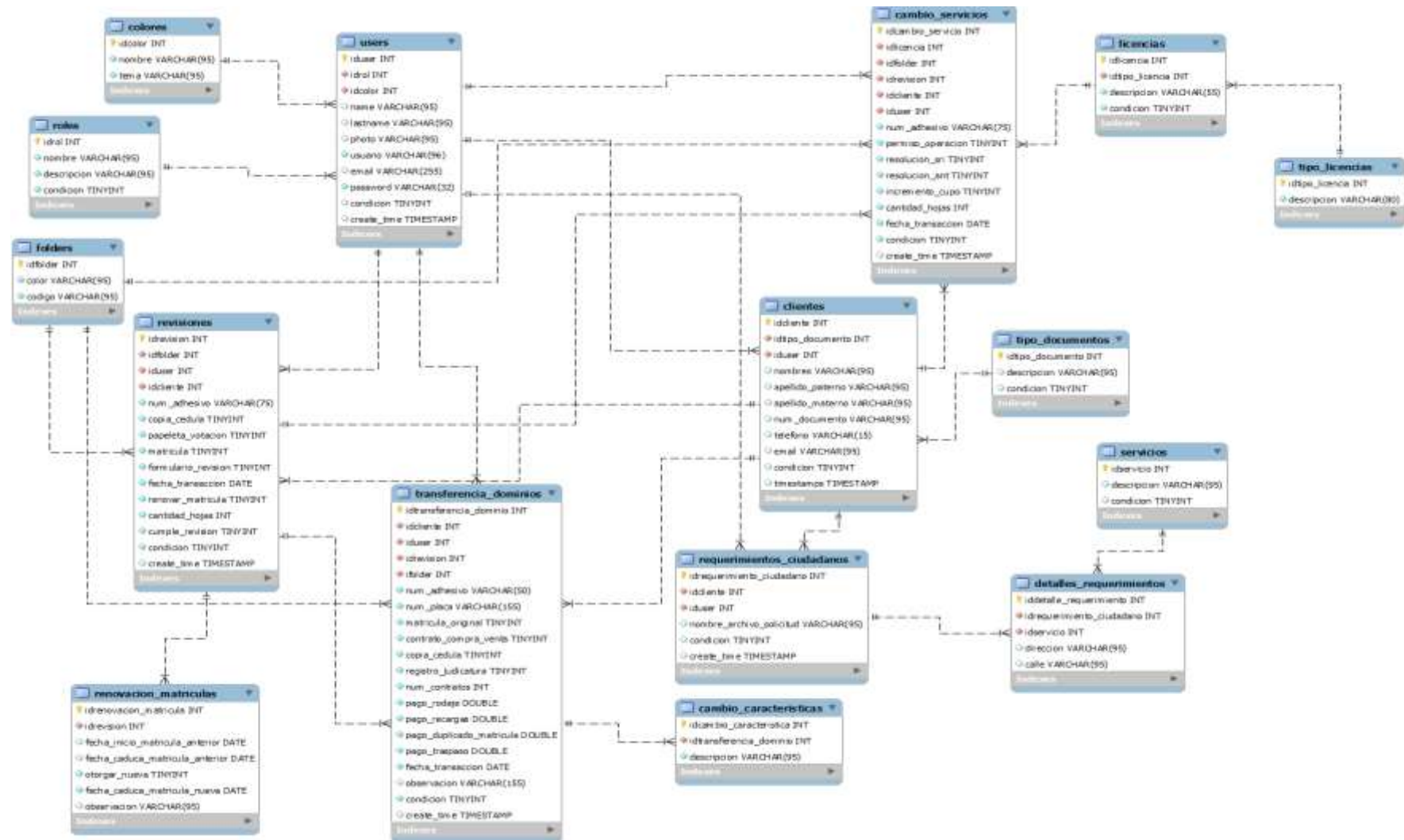


Gráfico 18: Diagrama relacional de la base de datos. Desarrollado por: Jorge Isrrael Diaz Montoya.

Diccionario de datos.

Tabla colores

Columna	Tipo de dato	PK	NN	AI	Por defecto
idcolor	INT	✓	✓	✓	
nombre	VARCHAR(95)		✓		
tema	VARCHAR(95)		✓		

Tabla 1: Colores. Desarrollado por: Jorge Isrrael Diaz Montoya.

Tabla de folders.

Columna	Tipo de dato	PK	NN	AI	Por defecto
idfolder	INT	✓	✓	✓	
color	VARCHAR(95)		✓		
codigo	VARCHAR(95)		✓		

Tabla 2: Folders. Desarrollado por: Jorge Isrrael Diaz Montoya.

Tabla de roles.

Columnas	Tipos de datos	PK	NN	AI	Por defecto
idrol	INT	✓	✓	✓	
nombre	VARCHAR(95)		✓		
descripción_rol	VARCHAR(95)		✓		
Condición_rol	TINYINT		✓		

Tabla 3: roles. Desarrollado por: Jorge Isrrael Diaz Montoya.

Tabla tipos de licencias.

tipo_licencias						
Columnas	Tipos de datos	PK	NN	UQ	BIN	Por defecto
idtipo_licencia	INT			✓	✓	✓
descripción_tipo_licencia	VARCHAR(80)				✓	

Tabla 4: tipo_licencias. Desarrollado por: Jorge Isrrael Diaz Montoya.

Tipos de documentos.

Columnas	Tipos de datos	PK	NN	AI	Por defecto
idtipo_documento	INT	✓	✓	✓	
descripción_tipo_documento	VARCHAR(95)				
condición_tipo_documento	TINYINT				

Tabla 5: tipo_documentos. Desarrollado por: Jorge Isrrael Diaz Montoya.

Tabla licencias.

licencias					
Columnas	Tipos de datos	PK	NN	AI	Por defecto
idlicencia	INT	✓	✓	✓	
idtipo_licencia	INT		✓		
descripción_licencia	VARCHAR(55)		✓		
condición_licencia	TINYINT		✓		

Tabla 6: licencias. Desarrollado por: Jorge Isrrael Diaz Montoya.

Tabla de servicios.

servicios					
Columnas	Tipos de datos	PK	NN	AI	Por defecto
idservicio	INT	✓	✓	✓	
descripción_servicio	VARCHAR(95)				
condición_servicio	TINYINT				

Tabla 7: servicios. Desarrollado por: Jorge Isrrael Diaz Montoya.

Tabla users.

users					
Columnas	Tipos de datos	PK	NN	AI	Por defecto
iduser	INT	✓	✓	✓	
idrol	INT		✓		
idcolor	INT		✓		
name	VARCHAR(95)				
lastname	VARCHAR(95)				
photo	VARCHAR(95)				
usuario	VARCHAR(96)		✓		
email	VARCHAR(255)				
password	VARCHAR(32)		✓		
condición_user	TINYINT				
create_time	TIMESTAMP				CURRENT_TIMESTAMP

Tabla 8: users. Desarrollado por: Jorge Isrrael Diaz Montoya.

Tabla de clientes.

clientes					
Columnas	Tipos de datos	PK	NN	AI	Por defecto
idcliente	INT	✓	✓	✓	
idtipo_documento	INT		✓		
Iduser	INT		✓		
nombres	VARCHAR(95)				
apellido_paterno	VARCHAR(95)				
apellido_materno	VARCHAR(95)				
num_documento	VARCHAR(95)				
telefono	VARCHAR(15)				
Email	VARCHAR(95)				
condición_cliente	TINYINT				
timestamps	TIMESTAMP				

Tabla 9: clientes. Desarrollado por: Jorge Isrrael Diaz Montoya.

Tabla de revisiones.

revisiones						
Columnas	Tipos de datos	PK	NN	AI	Por defecto	
idrevision	INT	✓	✓	✓		
Idfolder	INT		✓			
Iduser	INT		✓			
Idcliente	INT		✓			
num_adhesivo	VARCHAR(75)		✓			
copia_cedula	TINYINT		✓			
papeleta_votacion	TINYINT		✓			
matricula	TINYINT		✓			
formulario_revision	TINYINT		✓			
fecha_transaccion	DATE		✓			
renovar_matricula	TINYINT		✓			
cantidad_hojas	INT		✓			
cumple_revision	TINYINT		✓			
condición_revision	TINYINT		✓			
create_time	TIMESTAMP					

Tabla 10: revisiones. Desarrollado por: Jorge Isrrael Diaz Montoya.

Tabla de transferencia de dominios.

transferencia_dominios					
Columnas	Tipos de datos	PK	NN	AI	Por defecto
idtransferencia_dominio	INT	✓	✓	✓	
Idcliente	INT		✓		
Iduser	INT		✓		
Idrevision	INT		✓		
Ifolder	INT		✓		
num_adhesivo	VARCHAR(50)		✓		
num_placa	VARCHAR(155)		✓		
matricula_original	TINYINT		✓		
contrato_compra_venta	TINYINT		✓		
copia_cedula	TINYINT		✓		
registro_judicatura	TINYINT		✓		
num_contratos	INT		✓		
pago_rodaje	DOUBLE		✓		
pago_recargas	DOUBLE		✓		
pago_duplicado_matricula	DOUBLE		✓		
pago_traspaso	DOUBLE		✓		
fecha_transaccion	DATE		✓		
observación_transferencia	VARCHAR(155)				
condición_transferencia	TINYINT		✓		
create_time	TIMESTAMP				

Tabla 11: transferencia_dominios. Desarrollado por: Jorge Israel Diaz Montoya.

Tabla de cambio de características.

Columnas	Tipos de datos	PK	NN	AI	Por defecto
idcambio_caracteristica	INT	✓	✓	✓	
idtransferencia_dominio	INT		✓		
descripción_cambio_caracteristica	VARCHAR(95)		✓		

Tabla 12: cambio_caracteristicas. Desarrollado por: Jorge Isrrael Diaz Montoya.

Tabla de renovación de matrículas.

renovacion_matriculas					
Columnas	Tipos de datos	PK	NN	AI	Por defecto
idrenovacion_matricula	INT	✓	✓	✓	
Idrevision	INT		✓		
fecha_inicio_matricula_anterior	DATE				
fecha_caduca_matricula_anterior	DATE				
otorgar_nueva	TINYINT		✓		
fecha_caduca_matricula_nueva	DATE		✓		
observación_renovacion	VARCHAR(95)				

Tabla 13: renovacion_matriculas. Desarrollado por: Jorge Isrrael Diaz Montoya.

Tabla de cambio de servicios.

cambio_servicios					
Columnas	Tipos de datos	PK	NN	AI	Por defecto
idcambio_servicio	INT	✓	✓	✓	
idlicencia	INT		✓		
Idfolder	INT		✓		
idrevision	INT		✓		
idcliente	INT		✓		
Iduser	INT		✓		
num_adhesivo	VARCHAR(75)		✓		
permiso_operacion	TINYINT		✓		
resolucion_sri	TINYINT		✓		
resolucion_ant	TINYINT		✓		
incremento_cupo	TINYINT		✓		
cantidad_hojas	INT		✓		
fecha_transaccion	DATE		✓		
condición_cambio_servicio	TINYINT		✓		
create_time	TIMESTAMP				

Tabla 14: cambio_servicios. Desarrollado por: Jorge Isrrael Diaz Montoya.

Tabla de requerimientos ciudadanos.

Columnas	Tipos de datos	PK	NN	AI	Por defecto
idrequerimiento_ciudadano	INT	✓	✓	✓	
Idcliente	INT		✓		
Iduser	INT		✓		
nombre_archivo_solicitud	VARCHAR(95)				
condición_requerimiento	TINYINT				
create_time	TIMESTAMP				

Tabla 15: requerimientos_ciudadanos. Desarrollado por: Jorge Isrrael Diaz Montoya.

Tabla de detalle de requerimientos.

Columnas	Tipos de datos	PK	NN	AI	Por defecto
iddetalle_requerimiento	INT	✓	✓	✓	
idrequerimiento_ciudadano	INT		✓		
Idservicio	INT		✓		
Dirección	VARCHAR(95)				
Calle	VARCHAR(95)				

Tabla 16: detalles_requerimientos. Desarrollado por: Jorge Isrrael Diaz Montoya.

2.4.3. Desarrollo

Las tecnologías web que se han utilizado para el desarrollo del sistema son:

PhpMyAdmin:

El phpmyAdmin es una herramienta que se instala por defecto con el programa xampp, es muy útil y sencillo su uso, ya que a través del mismo podemos correr la base de datos en Mysql.

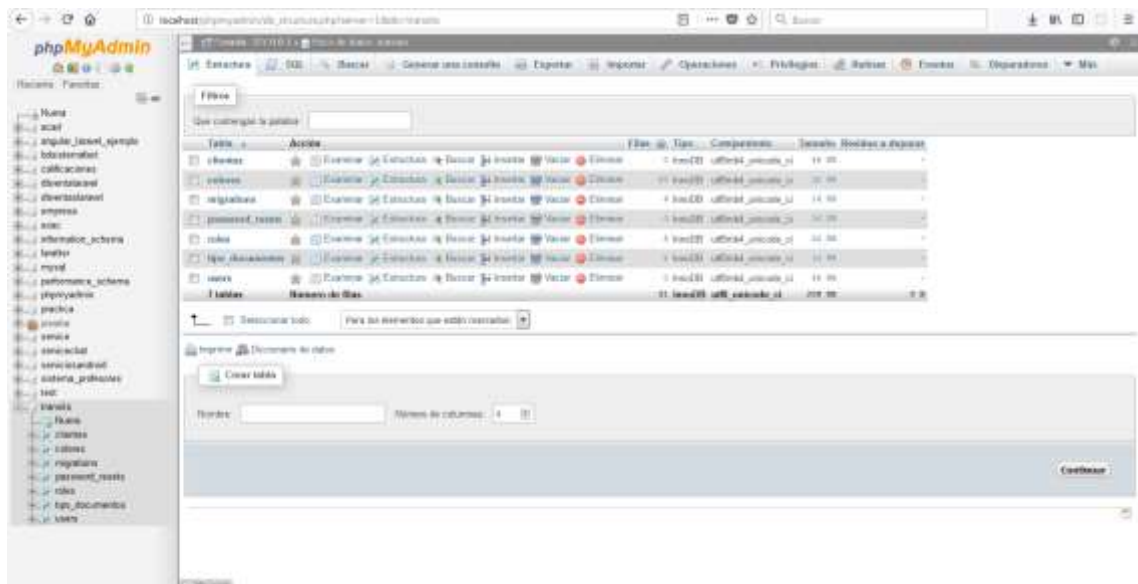
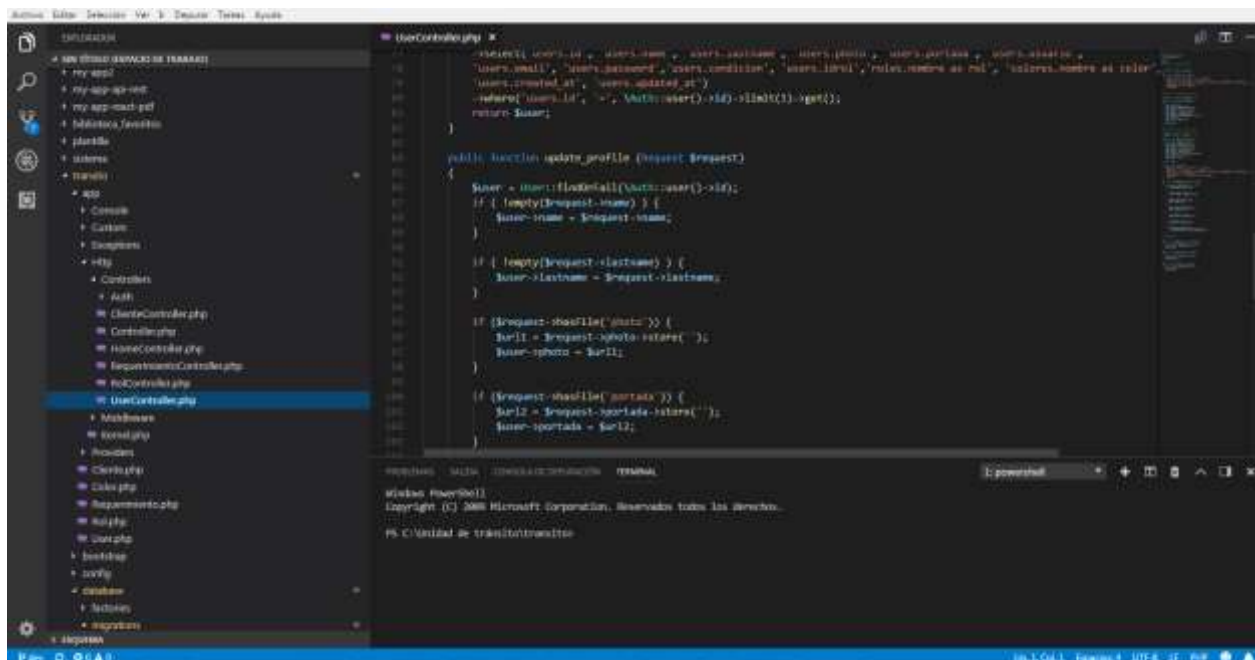


Ilustración 1: Interfaz del phpMyAdmin.

Visual Studio Code:

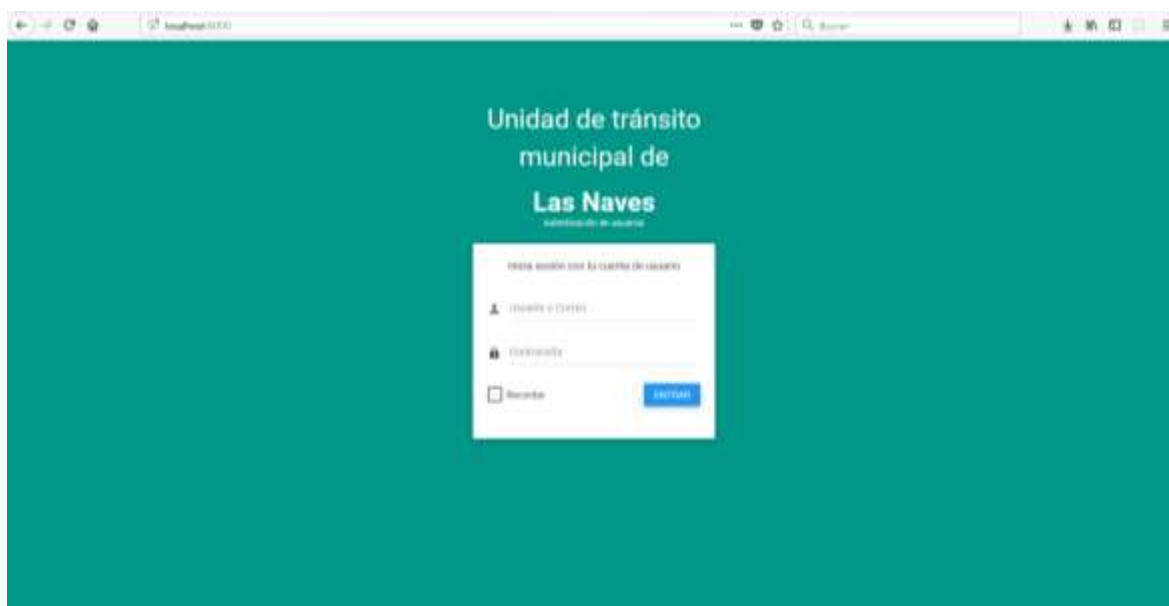
Visual Studio code es un editor de texto, que nos facilita el desarrollo de aplicaciones, en diferentes lenguajes de programación, marcado de hipertexto y de estilos, ya que nos ofrece un gran variedad de atajos con el teclado y autocompletar sintaxis, si se tratase de un IDE (Entorno de desarrollo integrado).



*Ilustración 2: Interfaz del Visual Studio Code.
Desarrollado por: Jorge Isrrael Diaz Montoya.*

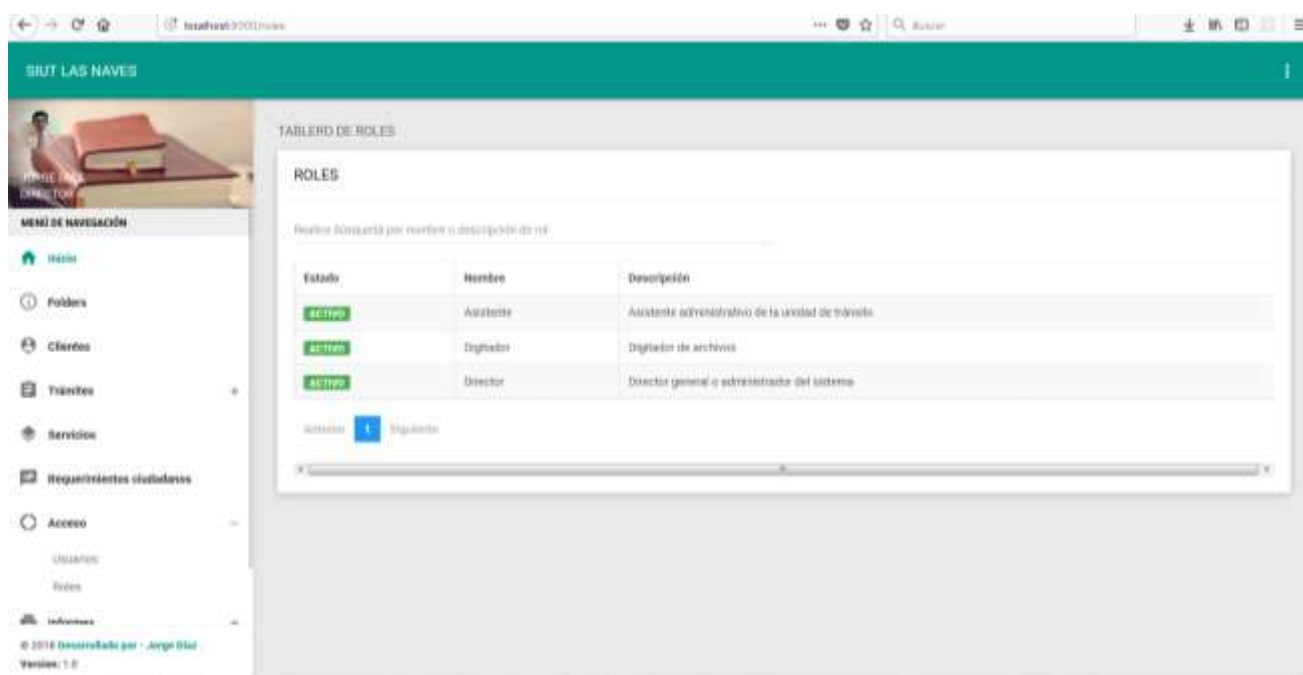
2.5. Ejecución y/o ensamblaje del prototipo.

Acceso de usuarios al sistema.



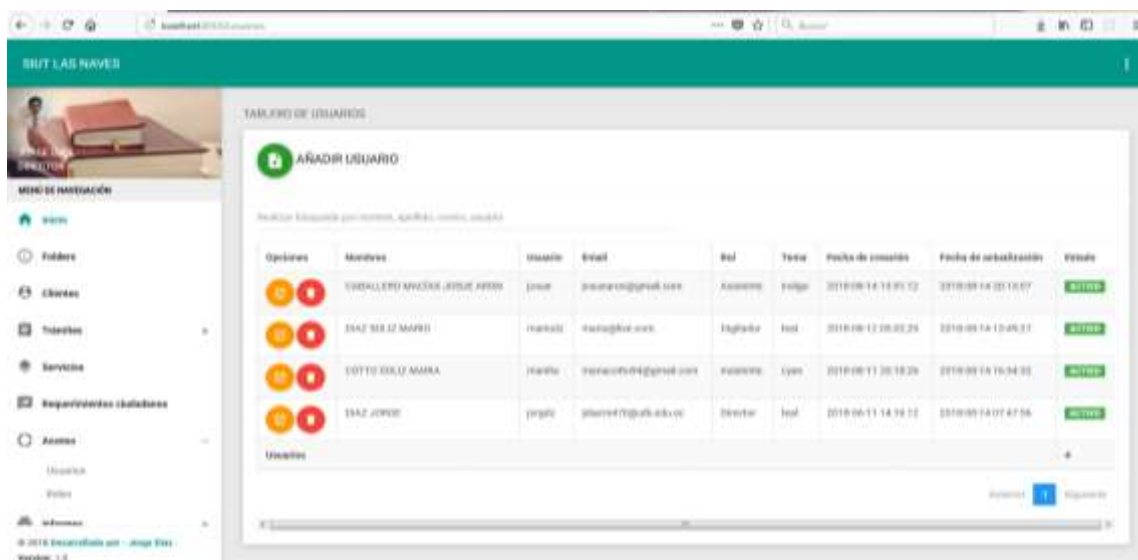
*Ilustración 3: Interfaz acceso al sistema.
Desarrollado por: Jorge Isrrael Diaz Montoya.*

Consulta de roles permitidos para los usuarios.



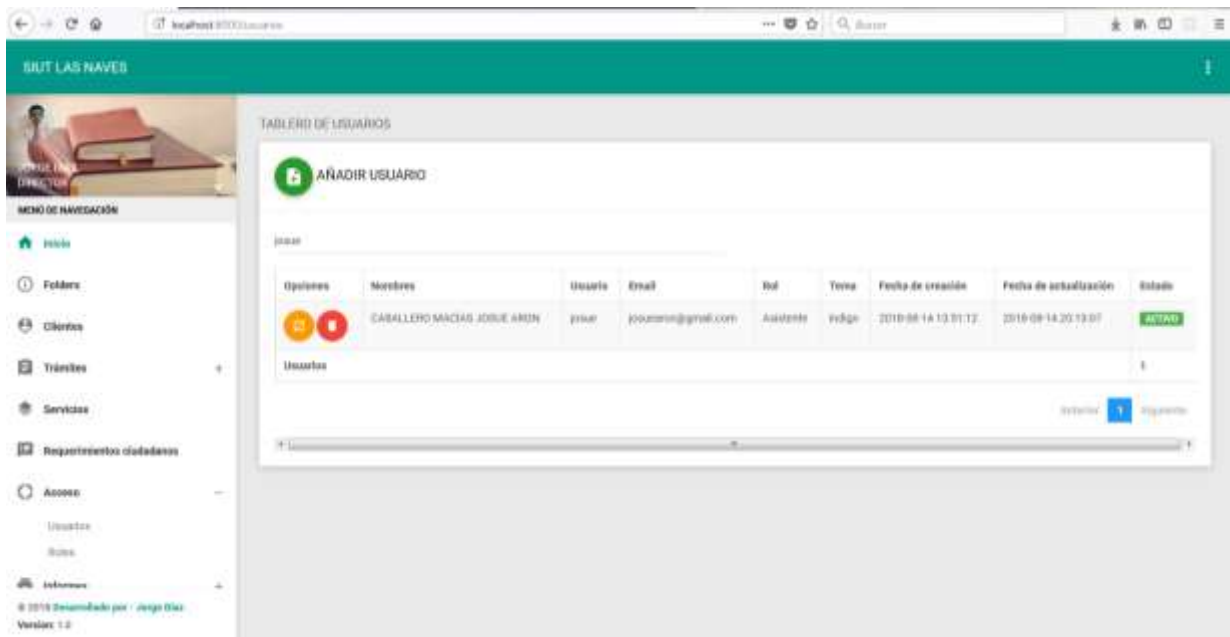
*Ilustración 4: Roles de usuarios.
Desarrollado por: Jorge Isrrael Diaz Montoya.*

Consulta de usuarios activos e inactivos en el sistema.



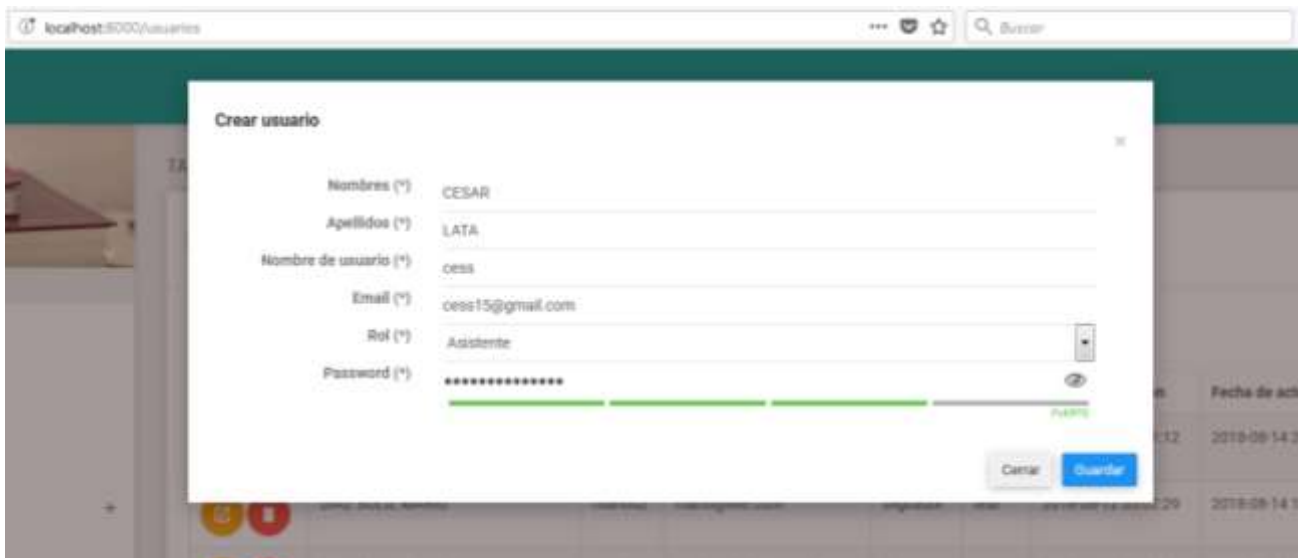
*Ilustración 5: Usuarios del sistema
Desarrollado por: Jorge Isrrael Diaz Montoya.*

Búsqueda de usuarios por nombres.



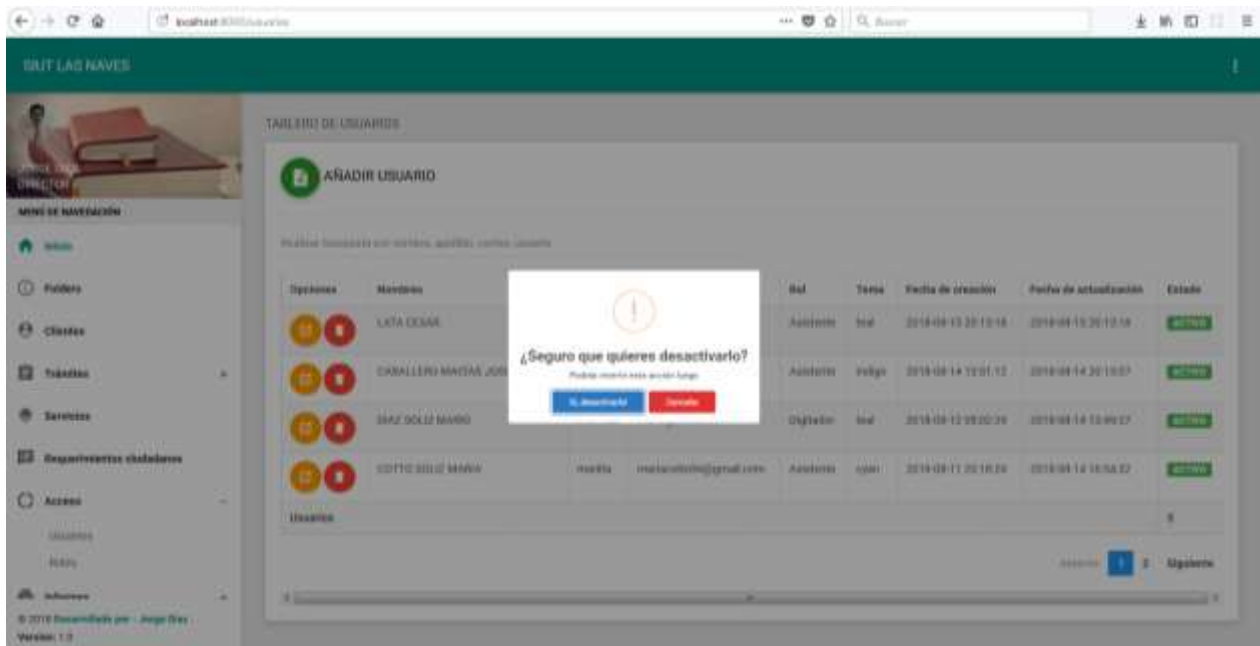
*Ilustración 6: Búsqueda de usuarios.
Desarrollado por: Jorge Isrrael Diaz Montoya.*

Ingreso de un nuevo usuario.



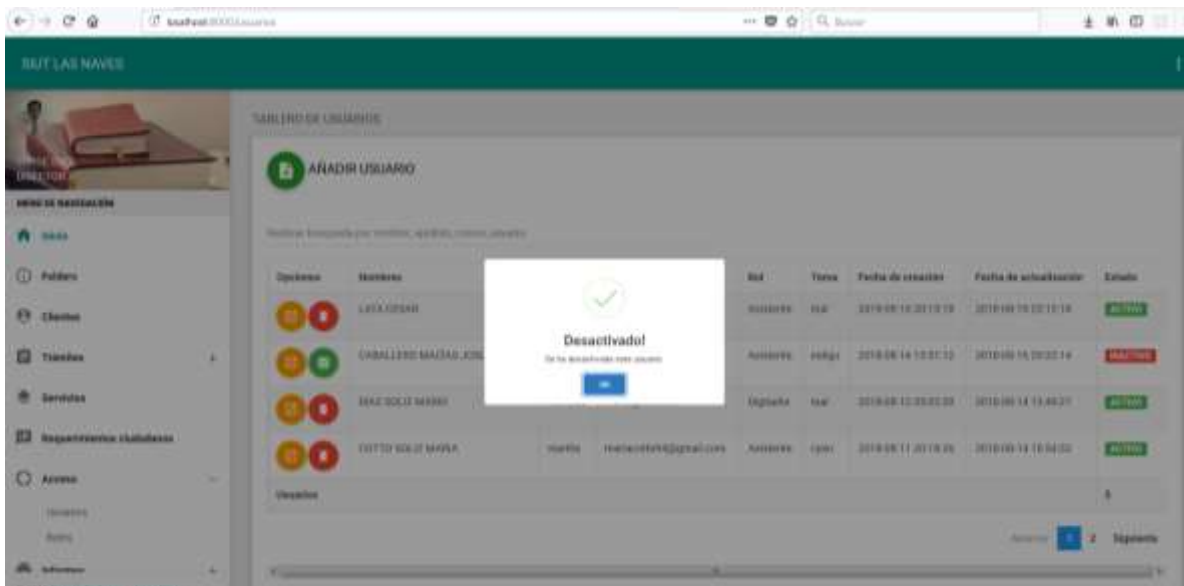
*Ilustración 7: Registro de usuarios
Desarrollado por: Jorge Isrrael Diaz Montoya.*

Desactivando un usuario para que no pueda acceder al sistema.



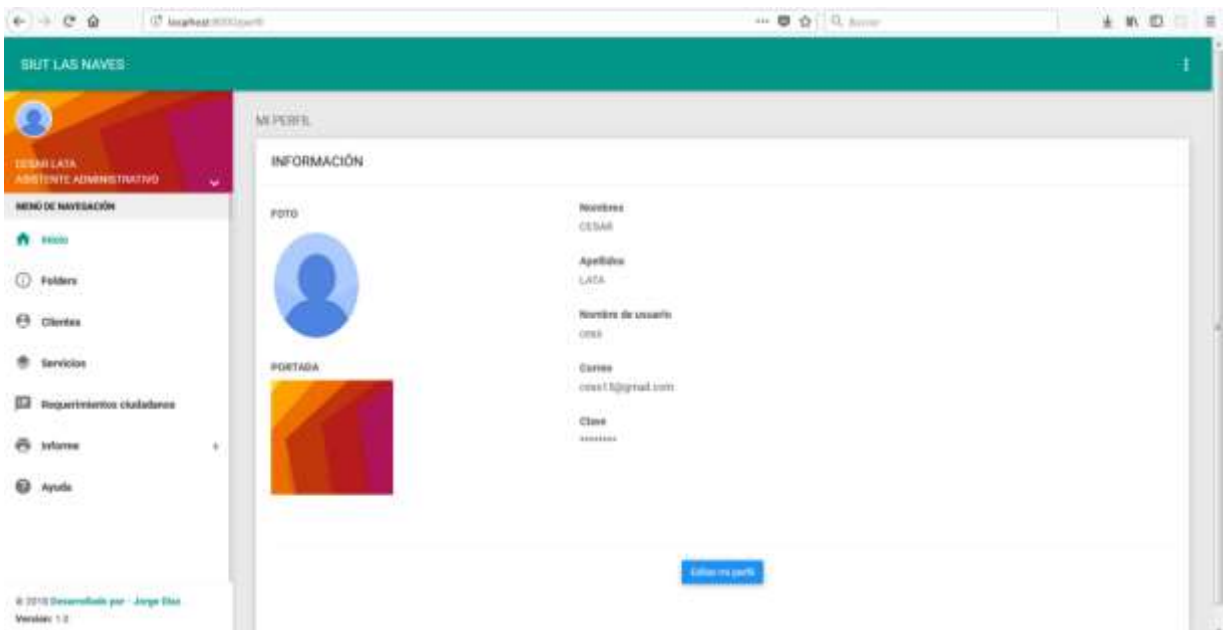
*Ilustración 8: Inactividad de usuario.
Desarrollado por: Jorge Isrrael Diaz Montoya.*

Usuario desactivado con éxito.

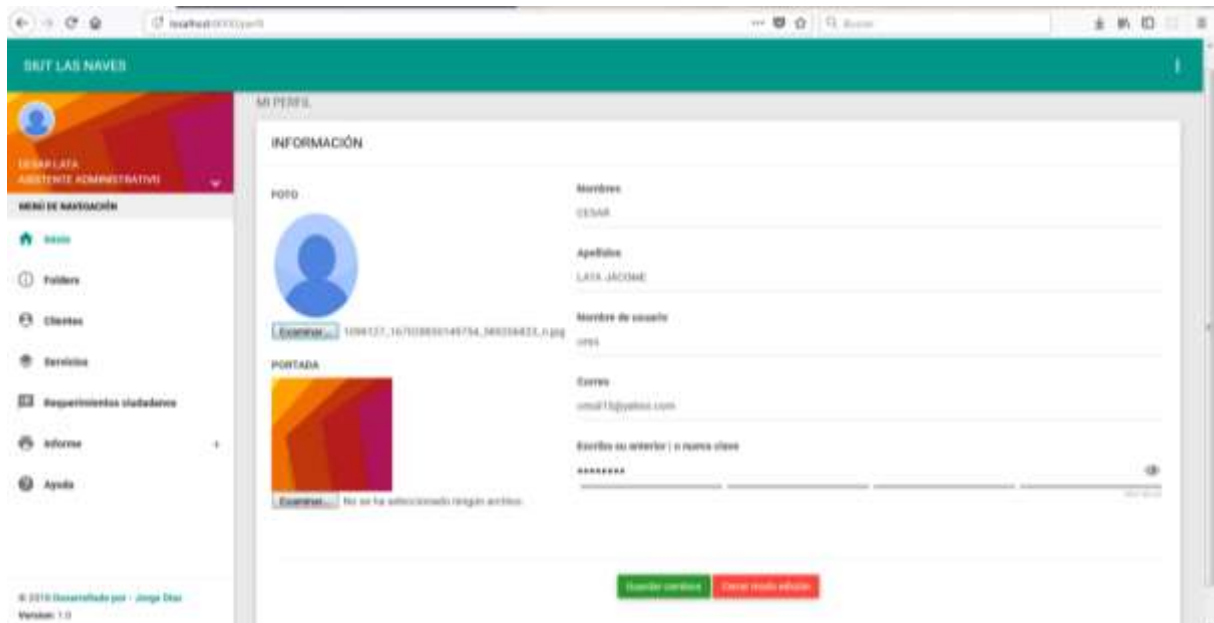


*Ilustración 9: Usuario desactivado.
Desarrollado por: Jorge Isrrael Diaz Montoya.*

Configuración de perfil de usuario.

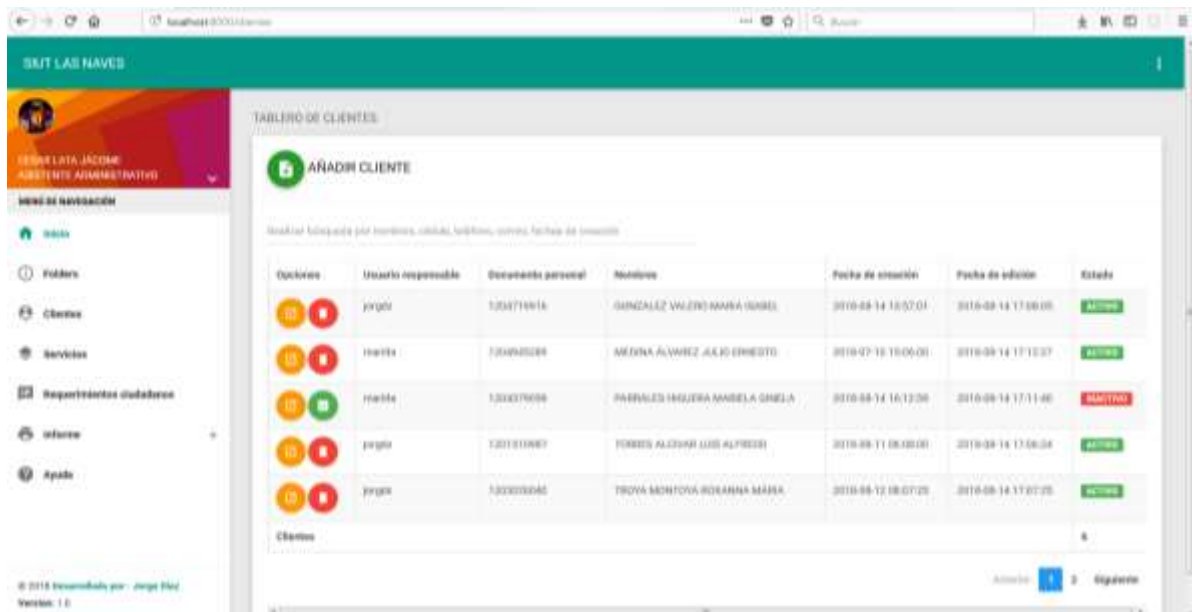


*Ilustración 10: Configuración de perfil.
Desarrollado por: Jorge Isrrael Diaz Montoya.*



*Ilustración 11: Actualizando información personal del usuario.
Desarrollador por: Jorge Isrrael Diaz Montoya.*

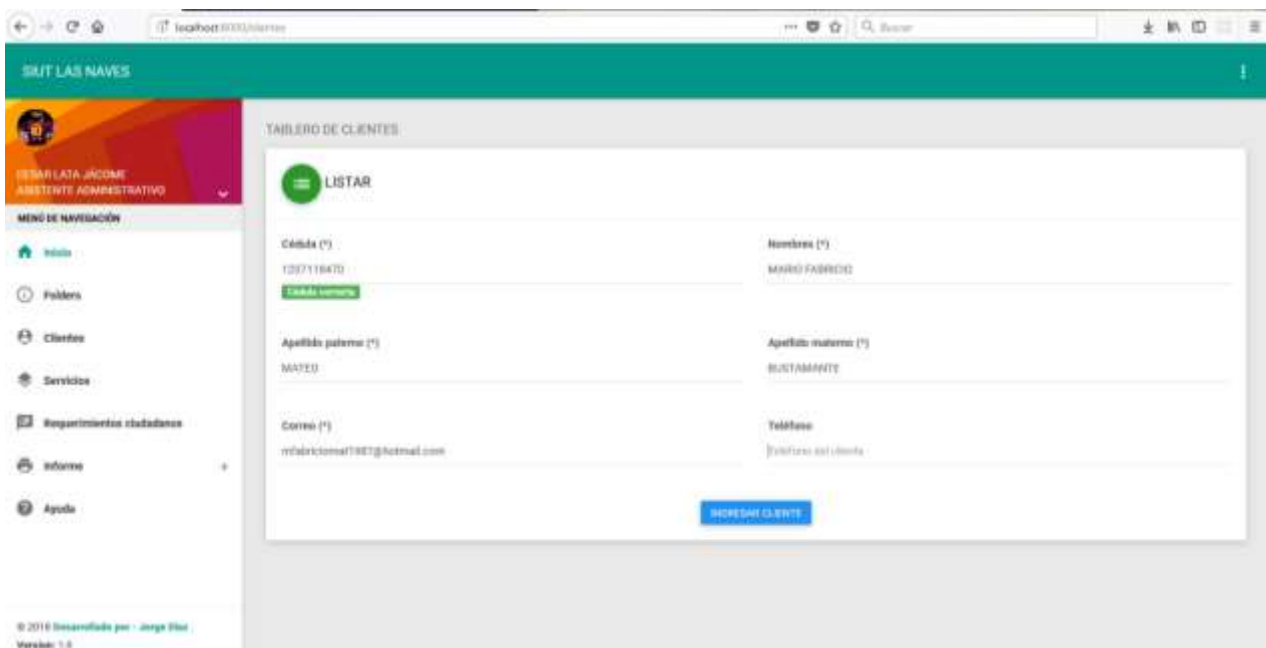
Listado de clientes.



Opciones	Usuario responsable	Documento personal	Nombre	Fecha de creación	Fecha de edición	Estado
	yrgr	1204719416	GONZALEZ VAZQUEZ MARIA ISABEL	2016-08-14 13:57:01	2016-08-14 17:06:05	ACTIVO
	heh4	120490208	MEDINA ALVAREZ JULIO ERNESTO	2016-07-19 18:06:00	2016-08-14 17:12:37	ACTIVO
	heh4	1200176008	PARRALES INJENIERA MARBELA GABRIELA	2016-08-14 16:17:36	2016-08-14 17:11:40	ACTIVO
	yrgr	1201210967	FERRER ALONSO JUAN ALFREDO	2016-08-11 08:00:00	2016-08-14 17:08:04	ACTIVO
	yrgr	1200000045	TROYA MONTOYA ROSANNA MARIA	2016-08-12 08:07:25	2016-08-14 17:07:25	ACTIVO

Ilustración 12: Consulta de todos los clientes ingresados.
Desarrollado por: Jorge Israel Diaz Montoya.

Ingreso de nuevos clientes.



LISTAR

Cédula (*)
1207118470

Nombre (*)
MIRKO FABRICIO

Apellido paterno (*)
MATEO

Apellido materno (*)
BUSTAMANTE

Correo (*)
mifabriciomat1987@hotmail.com

Teléfono

Ilustración 13: Registro de datos válidos de clientes.
Desarrollado por: Jorge Israel Diaz Montoya.

Listado de revisiones vehiculares ingresadas.

SIUT LAS NAVES

JORGE ISRAEL MONTOYA ADMINISTRADOR

MENÚ DE NAVEGACIÓN

- Inicio
- Folders
- Cientes
- Trámites
 - Revisión vehicular
 - Transferencia de dominio
 - Cambio de servicios
- Servicios
- Requerimientos ciudadanos
- Acceso

© 2018 Desarrollado por - Jorge Illaz
Version: 1.0

TABLERO DE REVISIÓN

AÑADIR REVISIÓN VEHICULAR

Realizar búsqueda por cédula, cliente, folder o fecha de transacción

Opciones	Num.adhesivo	Usuario	Cédula	Folder	Fecha de transacción	Copia de cédula	Votación	Matrícula	Formulario	Matrícula renovada	Cart. hojas	Revisión completa	Fecha de creación	Fe est
	NPA0008	marita	1207118470	NEGRO	2018-08-21						9		2018-08-21 18:59:49	20 18
	NPA0007	jorgdr	1709438232	GRIS	2018-08-21						12		2018-08-20 21:05:23	20 21
	NPA0006	jorgdr	1204719916	NEGRO	2018-08-20						10		2018-08-20 19:37:37	20 18

Ilustración 16: Lista de las revisiones vehiculares realizadas en la unidad de tránsito municipal. Desarrollado por: Jorge Isrrael Diaz Montoya.

Mostrar revisiones vehiculares por fechas.

SIUT LAS NAVES

JORGE ISRAEL MONTOYA ADMINISTRADOR

MENÚ DE NAVEGACIÓN

- Inicio
- Folders
- Cientes
- Trámites
 - Revisión vehicular
 - Transferencia de dominio
 - Cambio de servicios
- Servicios
- Requerimientos ciudadanos
- Acceso

© 2018 Desarrollado por - Jorge Illaz
Version: 1.0

TABLERO DE REVISIÓN

AÑADIR REVISIÓN VEHICULAR

Realizar búsqueda por cédula, cliente, folder o fecha de transacción

2018-08-21

< 2018 Aug >

Opciones	Num.adhesivo	Usuario	Cédula	Folder	Fecha de transacción	Copia de cédula	Revisión completa	Fecha de creación	Fe est
	NPA0008	marita	1207118470	NEGRO	2018-08-21			2018-08-21 18:59:49	20 18
	NPA0007	jorgdr	1709438232	GRIS	2018-08-21			2018-08-20 21:05:23	20 21

Revisiones vehiculares realizadas el 2018-08-21: 2

Atención Equipos

Ilustración 17: Listado de revisiones vehiculares por fechas especificadas por el usuario. Desarrollado por: Jorge Isrrael Diaz Montoya.

Ingreso de que el cliente posee los documentos necesarios para la revisión.

SIUT LAS NAVES

TABLERO DE REVISIÓN

LISTAR REVISIONES

INGRESO DE NUEVA REVISIÓN VEHICULAR

Clientes (*)
17042003 COELLO VERA DOMINGOS VICENTE

Folders (*)
FOLIO ROJO

Num. Adhesivo (*)
N/ADEEN

Fecha de transacción
2018-08-22

Num. hojas del trámite
8

Copias de cédula

Vinculación

Matrícula

Finc. Revisión

Revisión completa

Renovación

Guardar

© 2018 Desarrollado por - Jorge Elías
Versión: 1.0

*Ilustración 18: Ingreso de que el cliente posee sus documentos necesarios para la revisión vehicular.
Desarrollado por: Jorge Isrrael Diaz Montoya.*

Ingreso de revisión vehicular del cliente más la renovación de matrícula.

SIUT LAS NAVES

INGRESO DE NUEVA REVISIÓN VEHICULAR

Clientes (*)
1207118470 DIAZ MONTOYA JORGE ISRAEL

Folders (*)
FOLIO CELESTE

Num. Adhesivo (*)
Número de adhesivos del trámite

Fecha de transacción
2018-08-22

Num. hojas del trámite
8

Copias de cédula

Vinculación

Matrícula

Finc. Revisión

Revisión completa

Renovación

Fecha de inicio de la matrícula anterior
2018-08-22

Fecha de caducidad de la matrícula anterior
2018-08-22

Fecha de caducidad de la matrícula nuevo
2023-08-22

Otorgar nueva ?

Observación
Alguna observación que deba tener ?

Guardar

© 2018 Desarrollado por - Jorge Elías
Versión: 1.0

*Ilustración 19: Revisión vehicular más renovación de matrícula.
Desarrollado por: Jorge Isrrael Diaz Montoya.*

Ingreso de cambios de servicios del cliente que ya ha realizado una anterior revisión vehicular.

The screenshot shows the 'LISTAR CAMBIO DE SERVICIOS' form. The form includes the following fields and options:

- Adhensivo (*):** Numero de adhesivo: 1709435232 COELLO VERA DIOMEDES VICENTE
- Clients (*):** N° hojas del trámite: N°40857 COELLO VERA DIOMEDES VICENTE 2018-01
- Revisión vehicular (*):** N°40857 COELLO VERA DIOMEDES VICENTE 2018-01
- Licencia (*):** Dropdown menu with options: SELECCIONE LA LICENCIA, A NO PROFESIONAL, B NO PROFESIONAL, F NO PROFESIONAL, A1 PROFESIONAL, C PROFESIONAL (highlighted), C1 PROFESIONAL, D PROFESIONAL, D1 PROFESIONAL, E PROFESIONAL, E1 PROFESIONAL, G PROFESIONAL.
- Folder (*):** SELECCIONE UN FOLDER
- Fecha de transacción:** 2018-08-22
- Res. ANT:** Toggle switch (off)
- Inc. Cupó:** Toggle switch (off)

© 2018 Desarrollado por - Jorge Diaz
Version: 1.0

Ilustración 20: Crear cambio de servicios de clientes que ya posean una revisión vehicular.
Desarrollador por: Jorge Isrrael Diaz Montoya.

Generar informes de trámites vehiculares de un cliente.

The screenshot shows the 'TRÁMITES VEHICULARES' report form. The form includes the following fields and options:

- Trámite (*):** Revisión vehicular
- Num. Adhensivo o Cédula:** 1709435232
- Fecha de transacción:**

© 2018 Desarrollado por - Jorge Diaz
Version: 1.0

Ilustración 21: Generar informes de la cantidad de trámites que ha realizado un cliente.
Desarrollador por: Jorge Isrrael Diaz Montoya.

Informe general de un cliente y sus trámites de revisión vehicular realizados.

Fecha: 22-05-2018 Hora: 23:01:34

Revisión vehicular

Cédula	Cliente	Num. Adhesivo	Cumple revisión	Num. Hojas	Fecha de transacción
170435232	DIONEES VICENTE COELLO VERA	NP4005	NO	8	2018-05-22
170435232	DIONEES VICENTE COELLO VERA	NP4007	SI	12	2018-05-21
170435232	DIONEES VICENTE COELLO VERA	NP4006	SI	10	2018-05-12
170435232	DIONEES VICENTE COELLO VERA	NP4003	NO	12	2018-07-09
170435232	DIONEES VICENTE COELLO VERA	NP4001	NO	12	2018-05-19

Total de revisiones: 5

*Ilustración 22: Informe general de los trámites vehiculares de un cliente.
Desarrollado por: Jorge Isrrael Diaz Montoya.*

Generar informe general de los trámites de revisión vehicular.

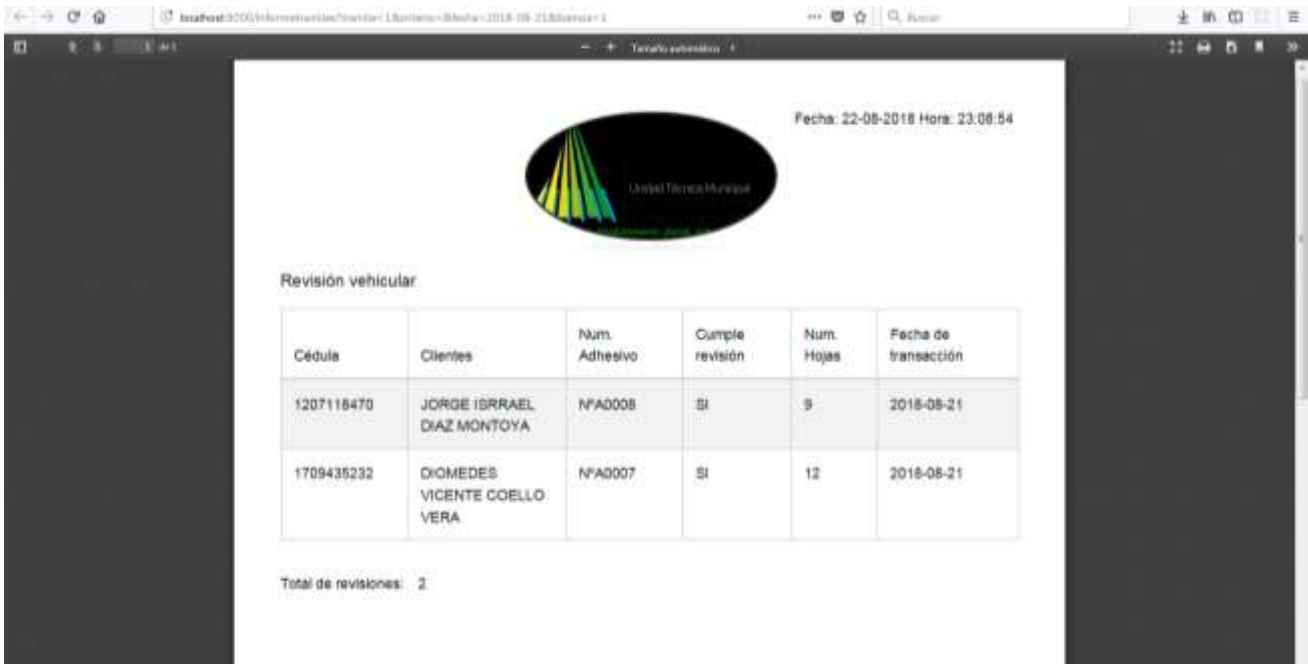
Fecha: 22-05-2018 Hora: 23:03:01

Revisión vehicular

Cédula	Cliente	Num. Adhesivo	Cumple revisión	Num. Hojas	Fecha de transacción
170435232	DIONEES VICENTE COELLO VERA	NP4005	NO	8	2018-05-22
120718475	JORGE ISRAEL DIAZ MONTOYA	NP4008	SI	8	2018-05-21
170435232	DIONEES VICENTE COELLO VERA	NP4007	SI	12	2018-05-21
120718475	JORGE ISRAEL DIAZ MONTOYA	NP4009	NO	11	2018-05-20
170435232	DIONEES VICENTE COELLO VERA	NP4006	SI	10	2018-05-12
120718475	JORGE ISRAEL DIAZ MONTOYA	NP4004	NO	14	2018-05-14
170435232	DIONEES VICENTE COELLO VERA	NP4003	NO	12	2018-07-09

*Ilustración 23: Generar informe general de trámites de revisión vehicular.
Desarrollador por: Jorge Isrrael Diaz Montoya.*

Informe obtenido por fecha, de los trámites de revisión vehicular.



Fecha: 22-08-2018 Hora: 23:08:54

Unidad Técnica Municipal

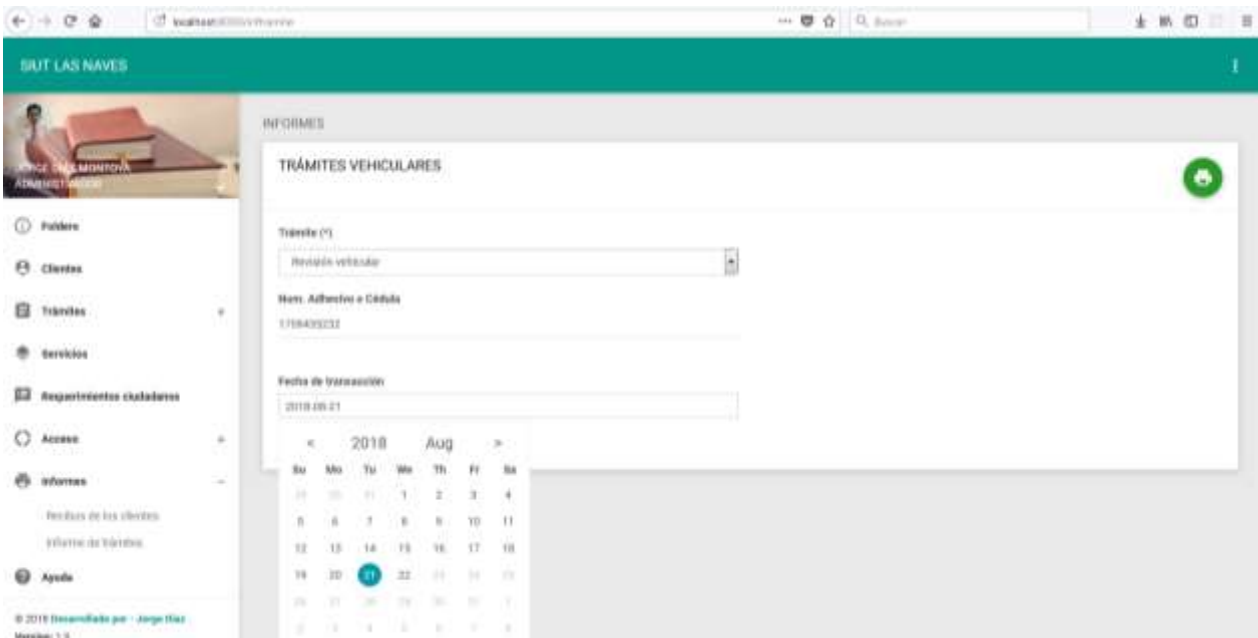
Revisión vehicular:

Cédula	Clientes	Num. Adhesivo	Cumple revisión	Num. Hojas	Fecha de transacción
1207118470	JORGE ISRAEL DIAZ MONTOYA	NFA0008	SI	9	2018-08-21
1709435232	DIOMEDES VICENTE COELLO VERA	NFA0007	SI	12	2018-08-21

Total de revisiones: 2

Ilustración 24: Informe por fecha de trámites de revisión vehicular. Desarrollado por: Jorge Isrrael Diaz Montoya.

Generar informe de trámites de revisión vehicular de un cliente, en una fecha específica.



SIT LAS NAVES

INFORMES

TRÁMITES VEHICULARES

Trámite (%)
Revisión vehicular

Num. Adhesivo e Cédula
1709435232

Fecha de transacción
2018-08-21

2018 Aug

Su Mo Tu We Th Fr Sa

21 22 23 24 25 26 27

28 29 30 31

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20

© 2018 Desarrollado por: Jorge Diaz
Versión: 1.3

Ilustración 25: Generar informe de trámites de revisión vehicular de un cliente en una fecha especificada. Desarrollador por: Jorge Isrrael Diaz Montoya.

Informe de revisión vehicular de un cliente en una fecha determinada.



Fecha: 22-08-2018 Hora: 23:10:39

Revisión vehicular

Cédula	Clientes	Num. Adhesivo	Cumple revisión	Num. Hojas	Fecha de transacción
1709435232	DICMEDES VICENTE COELLO VERA	N°A0007	SI	12	2018-08-21

Total de revisiones: 1

Ilustración 26: Informe de revisión vehicular de un cliente en una fecha especificada anteriormente.
Desarrollado por: Jorge Isrrael Diaz Montoya.

Gráficos estadísticos de la cantidad de trámites que se ha realizado en los últimos meses.

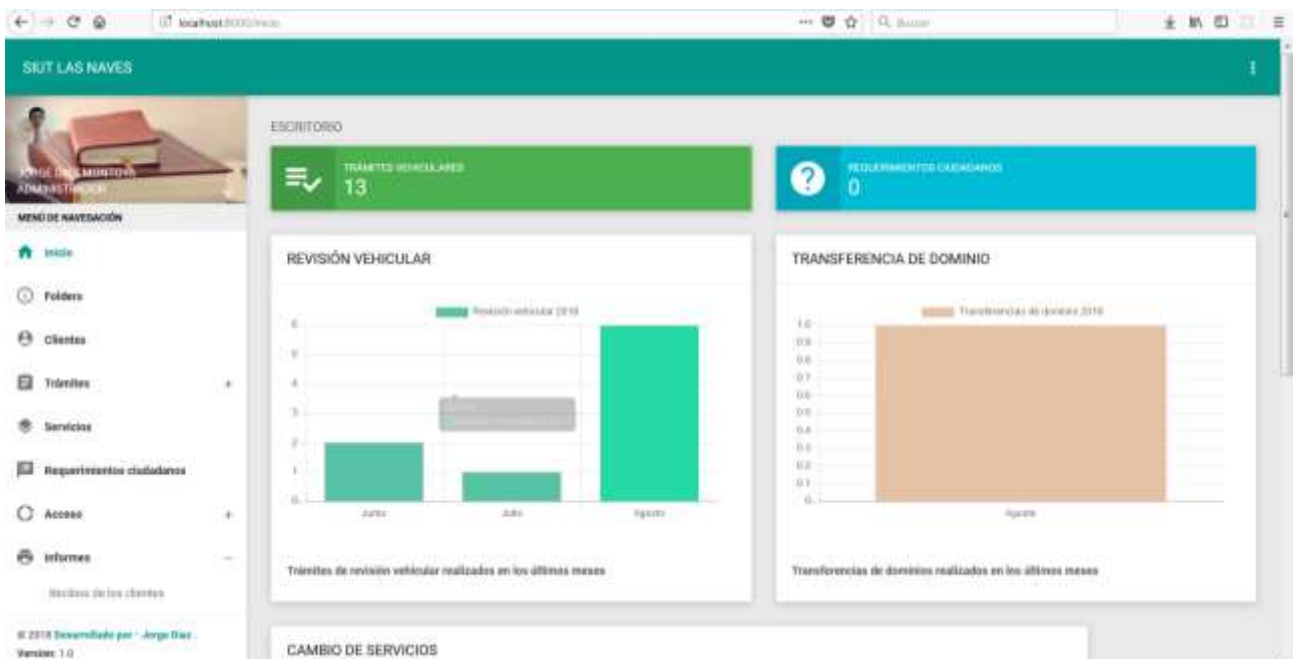


Ilustración 27: Gráficos estadísticos donde se muestra la cantidad de trámites de revisión, cambio de servicios y transferencia de dominio realizadas en el año actual.
Desarrollado por: Jorge Isrrael Diaz Montoya.

Gráfico de pastel de los cambios de servicios que se han realizado.

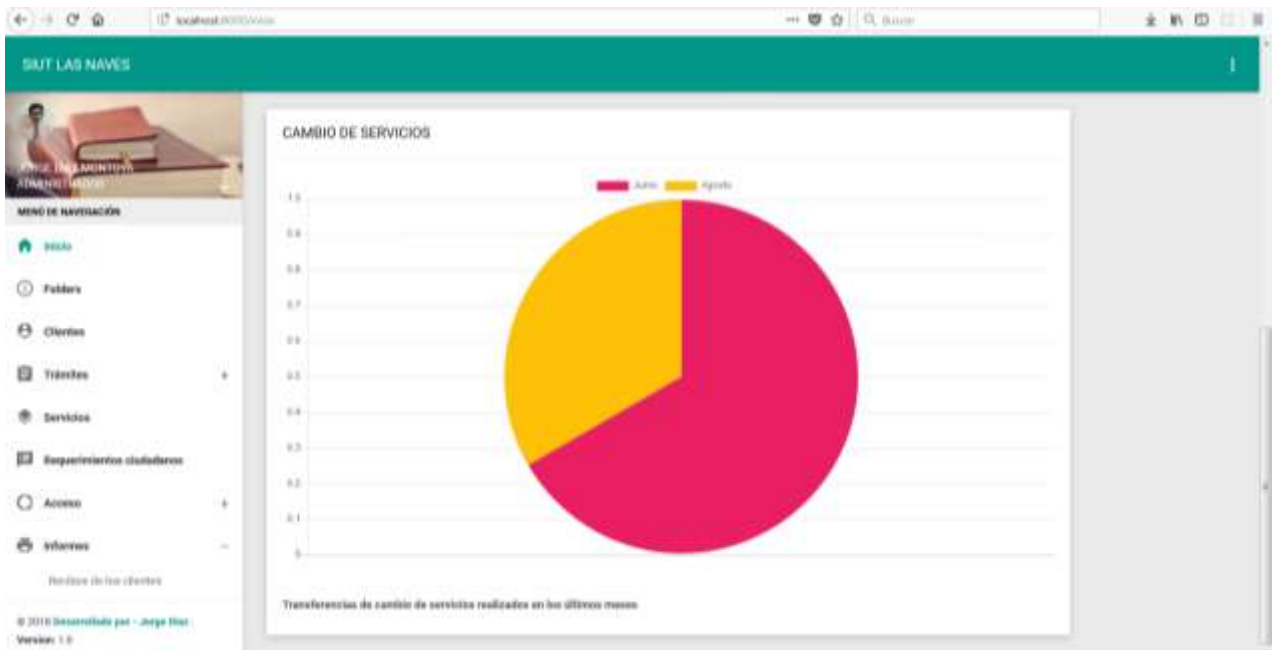


Ilustración 28: Gráfico de pastel de los cambios de servicios que se han realizado en los últimos meses. Desarrollado por: Jorge Isrrael Diaz Montoya.

Panel de registro de requerimiento ciudadanos.

Panel de registro de nuevos requerimientos ciudadanos. El panel está dividido en secciones: 'LISTAR REQUERIMIENTOS CIUDADANOS' y 'NUEVO REQUERIMIENTO CIUDADANO'. El formulario de registro incluye campos para: Cliente (*), Proyecto o solicitud (*), Servicios a los ciudadanos (*), Calle (*), Dirección (*), Nombre, ID, Descripción del requerimiento, Calle y Dirección. El campo 'Proyecto o solicitud (*)' tiene un botón 'Ejemplar' y el texto 'no se ha seleccionado ningún activo'. El campo 'Dirección (*)' tiene un botón 'Agregar' y el texto 'Desarrolla la dirección exacta del lugar'. El campo 'Servicios a los ciudadanos (*)' tiene un botón 'SELECCIONAR'. El campo 'Calle (*)' tiene el texto 'Calle donde está el problema'. El campo 'Dirección (*)' tiene el texto 'Desarrolla la dirección exacta del lugar'. El campo 'Nombre' tiene el texto 'No hay requerimientos agregados'. El campo 'ID' tiene el texto 'No hay requerimientos agregados'. El campo 'Descripción del requerimiento' tiene el texto 'No hay requerimientos agregados'. El campo 'Calle' tiene el texto 'No hay requerimientos agregados'. El campo 'Dirección' tiene el texto 'No hay requerimientos agregados'. El botón 'REGISTRAR' está ubicado en la parte inferior del formulario. El panel está ubicado en una interfaz de usuario con un menú de navegación a la izquierda que incluye opciones como Inicio, Pedidos, Clientes, Servicios, Requerimientos ciudadanos y Ayuda. El encabezado de la interfaz es 'SIT LAS NAVES'.

Ilustración 29: Panel para el registro de nuevos requerimientos que los moradores o ciudadanos tengan en el cantón las Naves. Desarrollado por: Jorge Isrrael Diaz Montoya.

Seleccionando archivo de proyecto, o solicitud del cliente, especificando los inconvenientes.

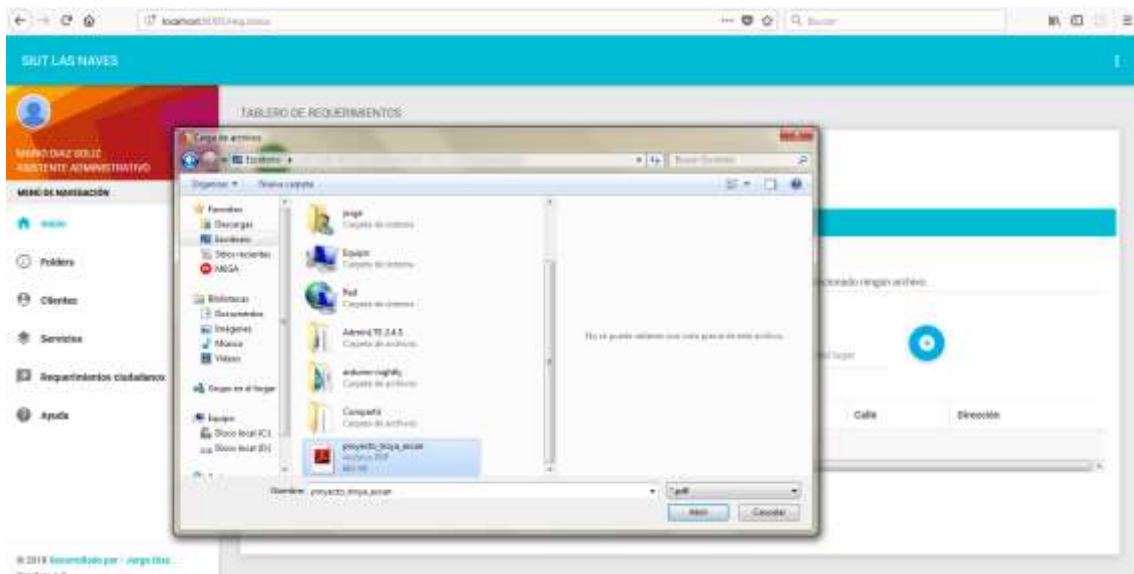


Ilustración 30: Selección de archivo de proyecto o solicitud, donde el cliente expresa lo que se requiere en el cantón o los inconvenientes que causan molestias.

Desarrollado por: Jorge Isrrael Diaz Montoya.

Añadiendo los inconvenientes con su respectiva dirección.

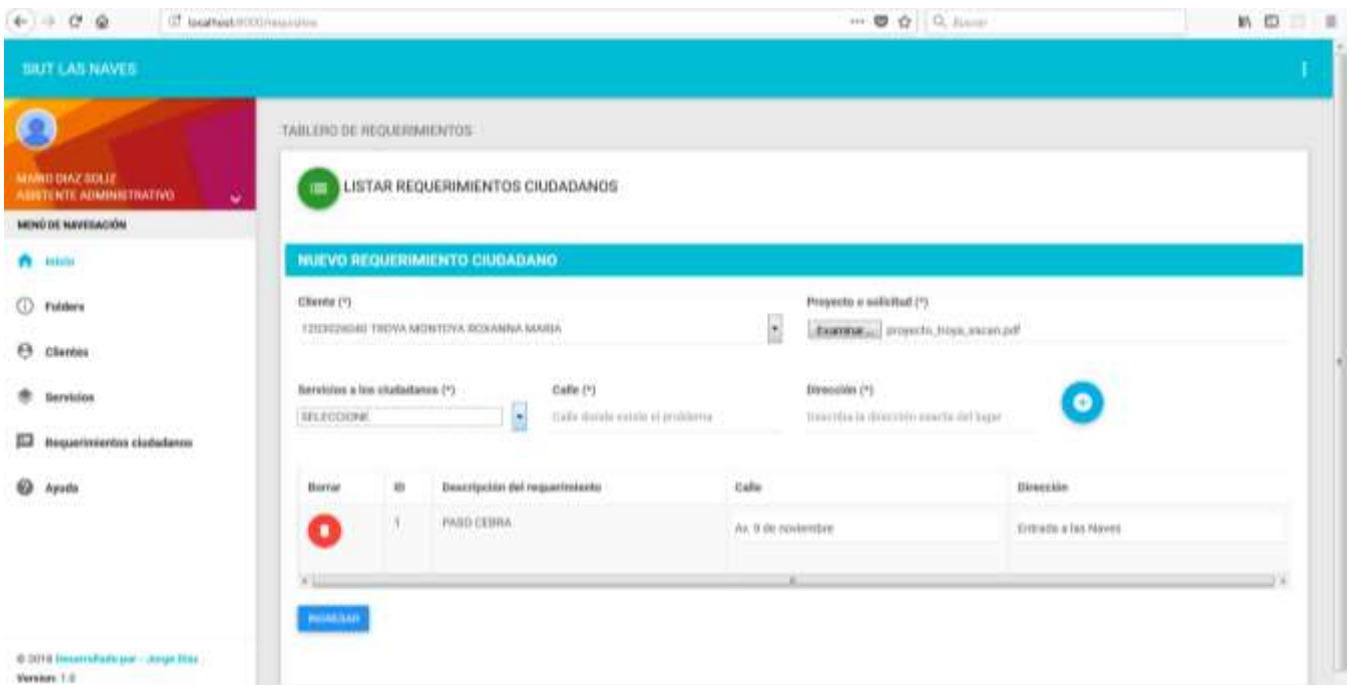


Ilustración 31: Panel para añadir los inconvenientes del sector, especificando su calle y dirección.

Desarrollado por: Jorge Isrrael Diaz Montoya.

Listado de todos los requerimientos realizados.

The screenshot shows a web application interface for managing requirements. On the left is a navigation menu with items like 'Inicio', 'Folders', 'Clientes', 'Servicios', 'Requerimientos ciudadanos', and 'Ayuda'. The main area is titled 'TABLERO DE REQUERIMIENTOS' and features a 'ANADIR REQUERIMIENTO' button. A modal window with a green checkmark and the text 'Correcto' is displayed over the table, indicating a successful action. The table below lists requirements with columns for 'Opciones', 'Proyecto', 'Fecha de creación', 'Fecha de edición', and 'Estado'. The 'Estado' column shows 'ACTIVO' for all entries.

Opciones	Proyecto	Fecha de creación	Fecha de edición	Estado
[Red, Green, Blue icons]	[Project icon]	2018-09-03 01:27:57	2018-09-03 01:27:57	ACTIVO
[Red, Green, Blue icons]	[Project icon]	2018-09-01 17:22:43	2018-09-01 17:22:43	ACTIVO
[Red, Green, Blue icons]	[Project icon]	2018-09-01 17:20:30	2018-09-01 17:20:30	ACTIVO
[Red, Green, Blue icons]	[Project icon]	2018-09-30 19:28:18	2018-09-30 19:28:18	ACTIVO

*Ilustración 32: Consulta de los requerimientos realizados con sus respectivos proyectos.
Desarrollado por: Jorge Isrrael Diaz Montoya.*

CAPÍTULO III

3. Evaluación del prototipo.

3.1. Plan de evaluación.

Para llevar a cabo la evaluación del sistema, se optó por el método de evaluación experto, el cual consiste en que 1 o más profesionales en el área del desarrollo de software, hagan pruebas en cuanto a la funcionalidad de la aplicación, y realicen las respectivas observaciones, ya sea que exista algo que corregir, o algún detalle que agregar para mejorar la experiencia de usuario en el sistema.

3.1.1. Funcionalidad y uso.

Tabla 17: Funcionalidad y factibilidad de uso

Destinatario	Guía experto.	Fecha	Actividades Desarrolladas	Observaciones	Cambios con el Sistema.	%
Usuario	Consultor Josué Caballero.	10/09/2018	Consultas de la información ingresada.	Peticiones realizadas de manera redundante en el código fuente.	Se optó por devolver el JSON en una respuesta.	85%
		10/09/2018	Mensajes de errores.	Msg. Correctos	Ninguno	100%
		10/09/2018	Búsqueda de información específica.	Validar los campos de fechas	Cambio del data del html por Vue Data	90%
		10/09/2018	Visualización del sistema desde dispositivos móviles.	Unas ventanas modales no se veían completas en teléfonos.	Ninguno	95%
		10/09/2018	Validación de los datos al realizar un ingreso.	Ninguna	Ninguna	100%
		10/09/2018	Pruebas de performance del sistema desplegado en internet.	El sistema comporta de manera correcta.	Ninguna	100%

Desarrollado por: Jorge Isrrael Diaz Montoya.

3.1.2. Estabilidad.

Tabla 18: Estabilidad del sistema

Destinatario	Guía experto	Fecha	Actividades desarrolladas	Observaciones	Cambios en el sistema	%
Servicio restful.	Consultor Josué Caballero.	10/09/2018	Peticiones simultáneas al servicio Restful.	Ninguna	Ninguna	100 %
Servicio restful.		10/09/2018	Tiempo de carga de archivos pdf.	El único inconveniente es que cuando el usuario desea ver el archivo en la web, sale dentro del servidor de dropbox, lo cual hace que el tiempo de carga sea un poquito mayor. Sin embargo el archivo se logra visualizar sin problemas a cabo de unos minutos.	A futuro se usará un servicio diferente para almacenar los archivos pdf y no tener estos inconvenientes.	80%
Sistema web		11/09/2018	Pruebas de concurrencia del sistema desplegado en internet.	.El sistema puede llegar a dejar de funcionar con un exceso de peticiones desde el cliente.	Ninguna	90 %
Sistema web.		12/09/2018	Pruebas de funcionalidad en general	Ninguna	Ninguna	90 %

Desarrollador por: Jorge Isrrael Diaz Montoya.

3.1.3. Compatibilidad.

Tabla 19: Compatibilidad del sistema

Destinatario	Guía experto	Fecha	Actividades desarrolladas	Observaciones	Cambios en el Sistema	%
Aplicación web	Consultor Josué Caballero.	11/09/2018	Funcionalidad en diferentes navegadores.	En los navegadores antiguos no funcionaban ciertos componentes	Ninguno	95%
Aplicación web		11/09/2018	Adaptación del sistema a diferentes dispositivos.	Algunas modales no aparecían completas.	Ninguna	90%
Aplicación web		11/09/2018	Pruebas de velocidad en la navegación y consulta de datos.	Ninguna	Ninguna	100 %

Desarrollador por: Jorge Isrrael Diaz Montoya.

3.1.4. Interoperabilidad.

Tabla 20: Interoperabilidad del sistema

Destinatario	Guía experto.	Fecha	Actividades desarrolladas	Observaciones	Cambios en el sistema	%
Sistema en general	Consultor Josué Caballero.	11/09/2018	Pruebas de comunicación entre la Api Restful y la aplicación SPA.	La comunicación con axios funciona de manera correcta.	Ninguna	100 %
Servicio restful, Base de Datos		11/09/2018	Generar respuestas JSON	Ninguna	Ninguna	100 %
Servicio restful.		12/09/2018	Validación de datos en las Api's restful.	Las validaciones están funcionando correctamente a nivel de frontend haciendo uso de expresiones de regulares y backend, usando los helpers de laravel.	Ninguno	100 %

Desarrollador por: Jorge Isrrael Diaz Montoya.

3.2. Resultados de evaluación.

Tabla 21: Resultados de evaluación.

Plan de evaluación	Aceptación	Rechazo
Funcionalidad	95%	5%
Estabilidad	90%	10%
Compatibilidad	95%	5%
Interoperabilidad	100%	0%
Resultados finales	95%	5%

Desarrollador por: Jorge Isrrael Diaz Montoya.

3.3. Análisis de resultados.

Después de haberse realizado la evaluación del sistema, podemos observar en la *tabla 21*, que los resultados obtenidos son bastante favorables, ya que a pesar de haber presentado alguno que otro inconveniente con la visualización de archivos pdf en el navegador, o que las ventanas modales, no se ven completas en los dispositivos móviles, hemos obtenido un porcentaje con un 5% de rechazo, lo cual es bastante bueno, y más que todo porque son problemas que no influyen con el funcionamiento del sistema en sí, sino más bien con aspectos de preferencia visual por parte de los usuarios.

4. Conclusiones y recomendaciones.

4.1. Conclusiones.

Hoy en día la información es considerada como el bien más importante de una empresa, por tanto es necesario que las instituciones y/o organizaciones hagan uso de sistemas de información para agilizar los procesos internos que realizan.

La unidad de tránsito municipal realiza un gran cantidad de procesos, que por supuesto mucho de ellos no estaban contemplado dentro de los requerimiento del sistema SIUT, sin embargo, gracias a esta aplicación informática, ellos podrán tener un control sobre que trámites se han realizado, en qué fecha, y que cantidad de trámites se han efectuado.

Con las tecnologías y metodologías de desarrollo ágil que existen hoy en día es posible llevar a cabo el desarrollo de muchos sistemas, con diferentes propósitos, y gracias a Laravel, que nos ofrece un conjunto de herramientas, para validar, generar respuestas JSON, sistema de autenticación, etc, ha sido posible construir un backend para el sistema SIUT, con una arquitectura cliente servidor, separando la lógica de negocio de la aplicación cliente, patrón de diseño (MVC), etc.

El despliegue en un ambiente de preproducción, se lo ha realizado en una plataforma como servicio llamada heroku, y que luego la URL está siendo redireccionada desde el subdominio de las Naves, (siut.lasnaves.gob.ec), debido a que el hosting de la Unidad de tránsito por el momento posee ciertos servicios deshabilitados que el framework laravel requiere para funcionar correctamente.

Para el almacenamiento de archivos en la nube se utilizó dropbox por el momento, ya que a futuro para mayor cantidad de archivos que se vallan a

almacenar, se contratará un servicio en Digital Ocean por \$5 dólares mensuales, y de esta manera adquirir un espacio en disco de hasta 250GB.

4.2. Recomendaciones.

Los archivos escaneados para los requerimientos ciudadanos, deben ser obligatoriamente pdf, y no exceder el tamaño máximo establecido en la aplicación que es de aproximadamente 1Mb, ya que son archivos que en la mayoría de los casos no contienen imágenes ni tanto contenido, por lo que sería un desperdicio de espacio de almacenamiento en dropbox.

El servicio donde se despliegue la aplicación, ya sea el hosting, o en caso de querer usar heroku, debe tener asociada una tarjeta de crédito, con la cual puedan hacer el cobro respectivo, de manera que el sistema no deje de funcionar por falta de pago.

Referencias

- Alarcón, J. M. (26 de Febrero de 2018). *Que es un ORM*. Recuperado el 14 de Julio de 2018, de <https://www.campusmvp.es/recursos/post/que-es-un-orm.aspx>
- Alberto Blanch, H. G. (10 de Abril de 2017). *Single Page Application*. Recuperado el 13 de Julio de 2018, de <https://www.arsys.es/blog/programacion/disenio-web/spa-unica-pagina/>
- Alvarez, M. A. (29 de Noviembre de 2016). *Que es una SPA*. Recuperado el 14 de Julio de 2018, de <https://desarrolloweb.com/articulos/que-es-una-spa.html>
- Azaustre, C. (16 de Agosto de 2017). *Vue Js y sus funcionalidades más interesantes*. Recuperado el 20 de 08 de 2018, de <https://carlosazaustre.es/que-es-lo-que-me-gusta-de-vue-js/>
- Cases, E. F. (11 de Junio de 2014). *Apache HTTP Server*. Recuperado el 14 de Julio de 2018, de <http://www.ibrugor.com/blog/apache-http-server-que-es-como-funciona-y-para-que-sirve/>
- Caulés, C. Á. (21 de Diciembre de 2016). *Axios js Librería de promesas*. Recuperado el 14 de Julio de 2018, de <https://www.arquitecturajava.com/axios-js-una-libreria-de-promesas/>
- Celis, R. (Diciembre de 2017). *Qué es heroku y para que me sirve*. Recuperado el 15 de Septiembre de 2018, de <https://platzi.com/blog/que-es-heroku-y-para-que-me-sirve/>
- Cruz, L. (16 de Enero de 2018). *¿Qué es restful Web Service y que es REST?* Recuperado el 20 de 08 de 2018, de <https://www.codigonaranja.com/2018/restful-web-service/>
- Fernández, L. A. (2018). *UML, modelado de proyectos*. Recuperado el 13 de Julio de 2018, de <https://msdn.microsoft.com/es-es/library/bb972214.aspx>
- García, J. B. (11 de Diciembre de 2015). *Que es laravel*. Recuperado el 14 de Julio de 2018, de <https://www.arsys.es/blog/programacion/que-es-laravel/>
- Herrera, J. F. (2015). *Que es HTML*. Recuperado el 13 de Julio de 2018, de <https://codigofacilito.com/articulos/que-es-html>
- Laguna, D. R. (2016). *The Laravel Framework Version 5 for Beginners*.
- Moreno, F. (7 de Junio de 2018). *Qué es Css*. Recuperado el 12 de 08 de 2018, de https://developer.mozilla.org/es/docs/Web/CSS/Como_iniciar/Que_es_CSS
- Ochoa, J. (15 de Mayo de 2015). *Aprende a usar Eloquent*. Recuperado el 14 de Julio de 2018, de <https://styde.net/aprende-a-usar-eloquent-el-orm-de-laravel/>

Perilla, A. (1 de Septiembre de 2018). *¿Qué es Javascript?* Recuperado el 20 de 08 de 2018, de https://developer.mozilla.org/es/docs/Learn/JavaScript/First_steps/Qu%C3%A9_es_JavaScript

ANEXOS

Árbol de problemas.



