

UNIVERSIDAD POLITÉCNICA DE PUEBLA
Ingeniería en Informática



**Proyecto de Estancia Práctica en
Desarrollador en Sistemas de Software y
Administrador de Redes**

“Desarrollo y Mantenimiento de Aplicativos de Software del
Área de Simuladores”

Área temática del CONACYT: VII
Ingenierías y tecnologías

Presenta:

José Javier Morales Romero

Asesor técnico

M.C. Diana Ivone Tapia López

Asesor académico

M.C. Rebeca Rodríguez Huesca

Resumen

El presente proyecto consiste en desarrollar un sitio web de control del equipo o material para Instituto Nacional de Astrofísica, Óptica y Electrónica (INAOE).

El proyecto abarca las distintas etapas de la metodología cascada: análisis, diseño, implementación, pruebas y mantenimiento que se describen con detalle. Las herramientas utilizadas en dicho proyecto son: framework Laravel versión 5.8, HTML, PostgreSQL versión 9.5.7, Bootstrap versión 4.3.1, sin embargo el framework que tiene mayor peso en el proyecto es Laravel el cual tiene muchas funcionalidades tales como: migración de datos, modelos, vistas, rutas, controladores entre otros, algunas de las ventajas que tiene son: reducción de costo, tiempo en el desarrollo y mantenimiento.

Los resultados del sitio web se obtuvieron siguiendo la metodología antes descrita.

Índice

1. Introducción	4
1.1. Descripción del problema o necesidad	4
1.2. Justificación	4
1.3. Objetivo general y específicos	4
2. Metodología y herramientas	5
2.1. Metodología empleada	5
2.2. Herramientas de desarrollo	7
3. Resultados	10
3.1. Análisis	10
3.2. Diseño	10
3.3. Implementación	13
3.4. Pruebas	27
3.5. Mantenimiento	30
4. Conclusiones y recomendaciones	31
5. Referencias bibliográficas	32

1. Introducción

Este capítulo es la descripción de la problemática y necesidad del Instituto Nacional Astrofísica, Óptica y Electrónica (INAOE), es decir por qué razón surge este trabajo. Seguidamente se establecen el objetivo general y los objetivos específicos que nos ayudara a medir el avance del proyecto.

1.1. Descripción del problema o necesidad

Actualmente el departamento de recursos materiales del Instituto Nacional de Astrofísica, Óptica y Electrónica (INAOE) lleva el control del equipo y material (herramienta de construcción y mueblería) que es otorgado a los trabajadores de manera manual a través de hojas impresas, esto ocasiona que pierda mucho tiempo para el administrador a la búsqueda del equipo y material. De acuerdo con la problemática planteada, se ve la necesidad de tener un sitio web que permita el control del equipo y material teniendo como resultado un adecuado almacenamiento de información de esta manera se obtenga mejores utilidades y beneficios para la institución.

1.2 Justificación

Desarrollar un sitio web en la actualidad es una condición laboral en la mayoría de las empresas o instituciones. Este proyecto tiene finalidad de crear un sitio web de control para el departamento de recursos materiales, el cual beneficiará las personas tener un control del equipo o material.

Para el desarrollo se utilizará un framework llamado laravel, el cual permite la migración de datos, vistas, rutas entre otros.

1.3 Objetivo General y Específicos

El objetivo general: Crear un sitio web que tenga un control de recursos materiales del Instituto Nacional De Astrofísica, Óptica y Electrónica.

Objetivos específicos:

- Analizar las necesidades específicas del sitio web.
- Diseñar estructura, arquitectura y composición de la base de datos e interfaz.
- Implementar código fuente.
- Realizar pruebas de su funcionalidad que asegure los resultados requeridos.

2. Metodología y herramientas

Este capítulo describe la metodología que se implementará en el proyecto, es decir cumpliendo cada una de las etapas. También se incluye las herramientas que nos ayudarán en el desarrollo del sitio web.

2.1 Metodología empleada

En la metodología cascada, los requerimientos tienen que estar bien definidos desde el inicio del proyecto y la probabilidad de que cambien debe ser mínima. Cabe mencionar que esto aplica, tanto al desarrollo de sistemas nuevos, como al desarrollo de modificaciones sobre un sistema existente. Pfleeger & Atlee [1] recomiendan además el uso de un modelo en cascada cuando los requerimientos están fuertemente acoplados o cuando son complejos, es decir, cuando no es sencillo separar los requerimientos para desarrollarlos uno por uno, ya que se corre el riesgo de que el desarrollo de unos no sea compatible con la de otros [2]. El modelo de cascada abarca las siguientes etapas [3] como se muestra en la figura 1:

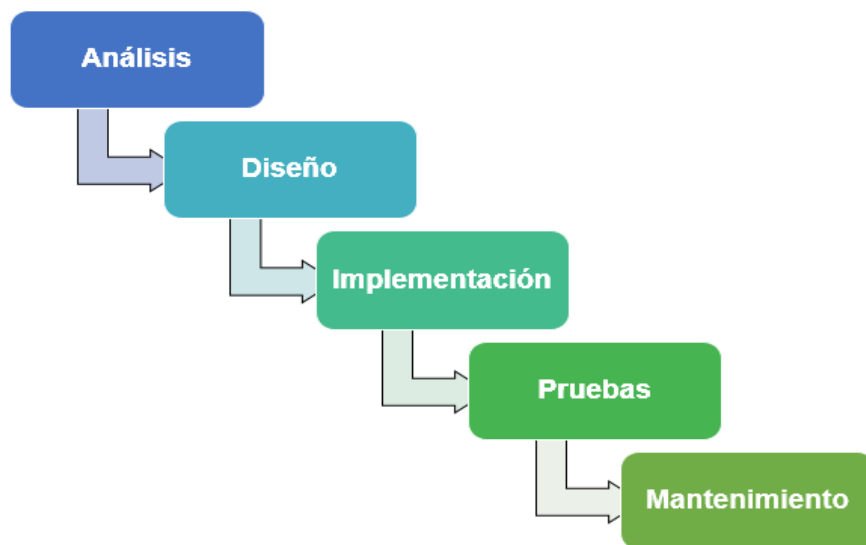


Figura 1. Etapas de la metodología cascada

- Análisis de software: el proceso de recopilación de los requisitos se centra e intensifica especialmente en el software. El ingeniero de software debe comprender el ámbito de la información del software así como la función, el rendimiento y las interfaces requeridas.

- **Diseño:** el diseño del software se enfoca en cuatro atributos distintos del programa; la estructura de los datos, la arquitectura del software, el detalle procedimental y la caracterización de la interfaz. El proceso de diseño traduce los requisitos en una representación del software con la calidad requerida antes de que comience la codificación.
- **Implementación [4]:** Es la fase de programación propiamente dicha. Aquí se desarrolla el código fuente, haciendo uso de prototipos así como pruebas y ensayos para corregir errores. Dependiendo del lenguaje de programación y su versión, se crean las librerías y componentes reutilizables dentro del mismo proyecto para hacer que la programación sea un proceso mucho más rápido.
- **Prueba:** una vez que se ha generado el código comienza la prueba del programa. La prueba se centra en la lógica interna del software y en las funciones externas, realizando pruebas que aseguren que la entrada.
- **Mantenimiento:** el software sufrirá cambios después de que se entrega al cliente. Los cambios ocurrirán debidos a que se haya encontrado errores, a que el software deba adaptarse a cambios del entorno externo (sistema operativo o dispositivos periféricos) o a que el cliente requiera ampliaciones funcionales o del rendimiento.

Características [5]:

- Es denominado ciclo de vida clásico y modelo lineal secuencial.
- Una fase no comienza hasta que la anterior ha terminado.
- Requiere disponer de unos requisitos completos y precisos al principio del desarrollo.

Ventajas [6]:

- Debido a la estructura lógica del modelo, a menudo se pueden evitar errores conceptuales.
- El modelo conduce a una extensa documentación técnica, que es un alivio para los nuevos programadores y desarrolladores y también es útil en la fase de prueba.
- El progreso del proyecto puede ser monitoreado usando metas.
- El coste total puede estimarse con relativa precisión si no hay conflictos.

Desventajas:

- Los conflictos, bugs y errores de programación a veces conducen a un aumento de los costos y a una cantidad considerable de tiempo. Lo mismo se aplica si los clientes no están satisfechos.

- Las especificaciones que se hacen inicialmente son a menudo difíciles de entender para los clientes porque son más abstractas de lo que se supone que el software debe hacer.
- La entrega del software lleva más tiempo porque los departamentos no trabajan simultáneamente y cada fase sólo puede comenzar cuando se ha completado la fase anterior.

2.2 Herramientas de desarrollo

A continuación se especificará las herramientas de desarrollo que nos ayudan a la creación del código fuente de un sitio web.

2.2.1 Laravel

Es un framework de código abierto para el desarrollo de aplicaciones web en PHP 5 que posee una sintaxis simple, expresiva y elegante. Fue creado en 2011 por Taylor Otwell, inspirándose en Ruby on Rails y Symfony, de los cuales ha adoptado sus principales ventajas.

Laravel facilita el desarrollo simplificando el trabajo con tareas comunes como la autenticación, el enrutamiento, gestión sesiones, el almacenamiento en caché, etc. [7].

Ventajas [8]:

- ✓ Reducción de costo y tiempo en el desarrollo y mantenimiento.
- ✓ Generación de url amigables lo que hace más fácil el mantenimiento de un sitio web.
- ✓ Es de código abierto.

Desventajas:

- A utilizar la consola para realizar ciertas funciones. Conocerás Composer, el administrador de dependencias de PHP y Artisan de Laravel.
- La instalación se hace mediante Composer.
- La curva inicial de aprendizaje puede ser compleja si no has utilizado un framework o no manejas bien el lenguaje.

2.2.2 PostgreSQL

Es un sistema de gestión de bases de datos que incorpora el modelo relacional para sus bases de datos y usa el lenguaje SQL como lenguaje de consulta. La base de datos relacional PostgreSQL es una de las aplicaciones de código abierto con más éxito de los últimos años, seguido por muchos desarrolladores y usuarios. Es una buena herramienta para crear una aplicación con grandes cantidades de información no trivial se puede beneficiar de él. PostgreSQL es una excelente implementación de

una base de datos relacional, con todo tipo de funcionalidades, de código abierto y de uso gratuito [9].

Ventajas [10]:

- ✓ Extensible: El código fuente está disponible de forma gratuita, para que quien necesite extender o personalizar el programa pueda hacerlo sin costo.
- ✓ Gran capacidad de almacenamiento.
- ✓ Buen sistema de seguridad mediante la gestión de usuarios, grupos de usuarios y contraseñas.
- ✓ Estabilidad y confiabilidad: No se han presentado caídas de la base de datos.

Desventajas:

- Consume más recursos.
- La sintaxis de algunos de sus comandos o sentencias no es nada intuitiva.
- En comparación con MySQL es más lento en inserciones y actualizaciones, ya que cuenta con cabeceras de intersección que no tiene MySQL.

2.2.3 HTML

HTML (HyperText Markup Language) es un lenguaje muy sencillo que permite describir hipertexto, es decir, texto presentado de forma estructurada y agradable, con enlaces (hyperlinks) que conducen a otros documentos o fuentes de información relacionadas, y con inserciones multimedia (gráficos, sonido...) La descripción se basa en especificar en el texto la estructura lógica del contenido (títulos, párrafos de texto normal, enumeraciones, definiciones, citas, etc.) así como los diferentes efectos que se quieren dar (especificar los lugares del documento donde se debe poner cursiva, negrita, o un gráfico determinado) y dejar que luego la presentación final de dicho hipertexto se realice por un programa especializado [11].

Ventajas [12]:

- ✓ Sencillo que permite describir hipertexto.
- ✓ Texto presentado de forma estructurada y agradable.
- ✓ Archivos pequeños.
- ✓ Lenguaje de fácil aprendizaje.
- ✓ Lo admiten todos los exploradores.

Desventajas:

- Lenguaje estático.
- La interpretación de cada navegador puede ser diferente.
- Guarda muchas etiquetas que pueden convertirse en “basura” y dificultan la corrección.
- El diseño es más lento.
- Las etiquetas son muy limitadas.

2.2.4 Bootstrap

Es uno de los frameworks más populares y utilizados del mercado para la creación de páginas responsive. Entre los navegadores soportados se encuentran Chrome, Firefox, Opera, Safari e Internet Explorer a partir de la versión 8 (aunque en la versión 7 también funciona correctamente). Está preparado para funcionar tanto en navegadores de PCs y portátiles con cualquier tamaño de pantalla así como para tablets y smartphones de tamaños mucho más reducido [13].

Ventajas [14]:

- ✓ Utiliza componentes y servicios creados por la comunidad web como HTML5.
- ✓ Es un conjunto de buenas prácticas que perduran en el tiempo.
- ✓ La implementación de HTML5 + CSS3.
- ✓ Herramienta sencilla y ágil para construir sitios web e interfaces.
- ✓ Además y por si fuera poco tiene un theme por defecto bastante optimizado y que puedes modificar o crear tu propio theme.

Desventajas:

- Es necesario adaptarse a su forma de trabajo, si bien su curva de aprendizaje es liviana, deberás comprender y familiarizarte con su estructura y nomenclatura.
- Un problema que he visto a menudo en muchos themes, es que el responsive funciona bien, pero a nivel de diseño en ocasiones se le escapan ciertos detalles que le quitan estética al sitio.
- Aquí empiezan los problemas, Bootstrap por defecto te trae anchos, márgenes y altos de línea, y realizar cambios específicos es por decir, un poco tedioso.

3. Resultados

En este capítulo se mostrará los resultados obtenidos a través de las distintas etapas de la metodología cascada.

3.1 Análisis

Se analizó los requisitos que necesita el sitio web, debe contener dos interfaces, una es para el administrador y otra para los usuarios.

Requisitos del Administrador:

- Insertar responsables.
- Insertar, actualizar, consultar y eliminar formularios.
- Insertar, actualizar y eliminar usuarios.
- Interfaz “responsables” debe contener un botón siguiente para que lance el interfaz formularios que tendrá tipo pase y áreas un selector.
- Interfaz usuario debe tener en tipo un selector si será administrador o usuario.
- Deber ser auto-incremental en número de folio.
- Interfaz formulario debe agregar y eliminar filas de la tabla características equipo/herramienta.

Requisitos del Usuario

- Insertar responsables, formularios.
- Consultar formularios.
- Debe ser auto-incremental el número de folio.
- Interfaz formulario debe agregar y eliminar filas de la tabla características equipo/herramienta.
- Interfaz “responsables” debe contener un botón siguiente para que lance el interfaz formularios que tendrá tipo pase y áreas un selector.

3.2 Diseño

En la Figura 2 se aprecia el diseño de modelo entidad relación para la base de datos, el diseño del interfaz administrador y usuario como se muestra en las Figuras 3 - 8.

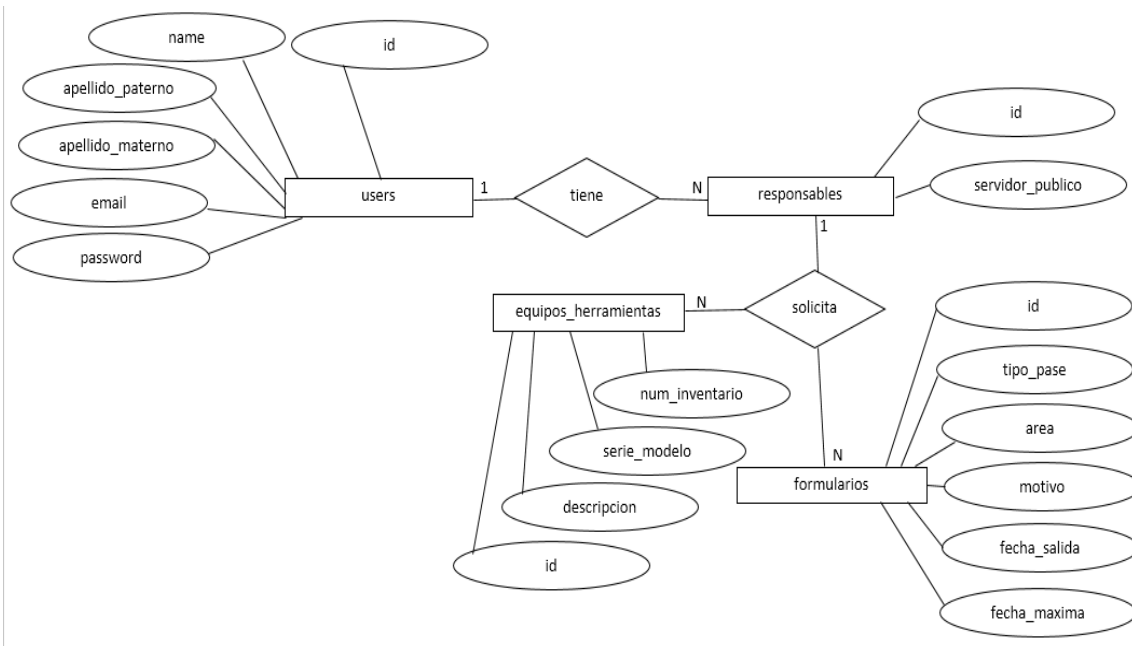


Figura 2. Modelo Entidad-Relación.

The form is titled "Login Registrar" and contains the following fields and a button:

- Nombre:
- Apellido Paterno:
- Apellido Materno:
- Correo:
- Contraseña:
- Confirma Contraseña:
-

Figura 3. Registrar Usuario.

Login Registrar

Correo:

Contraseña:

Figura 4. Login administrador y usuario

Áreas Responsbales Usuarios Administrador:

	<input type="text"/>
	<input type="text"/>

Figura 5. Página de áreas (Administrador).

Áreas Responsbales Usuarios Administrador:

	<input type="text"/>	<input type="text"/>	<input type="text"/>
	<input type="text"/>	<input type="text"/>	<input type="text"/>

Figura 6. Página de responsables (Administrador).

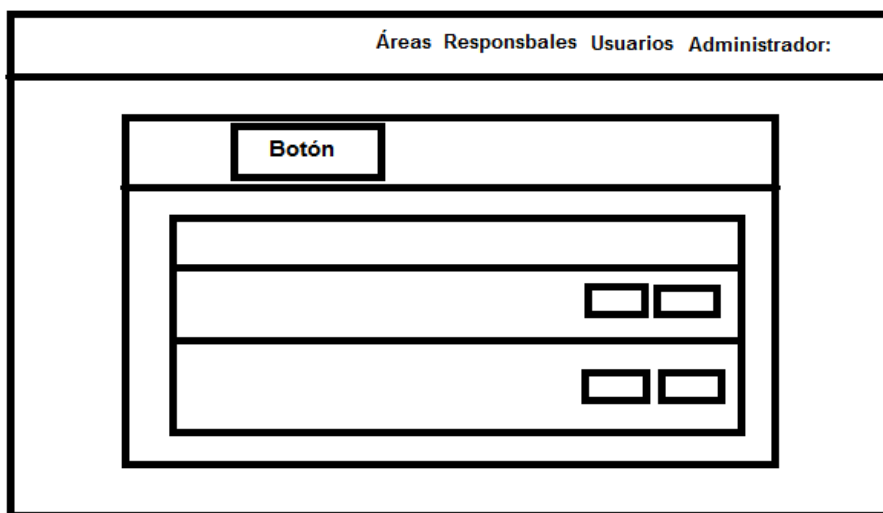


Figura 7. Página de usuarios (Administrador).

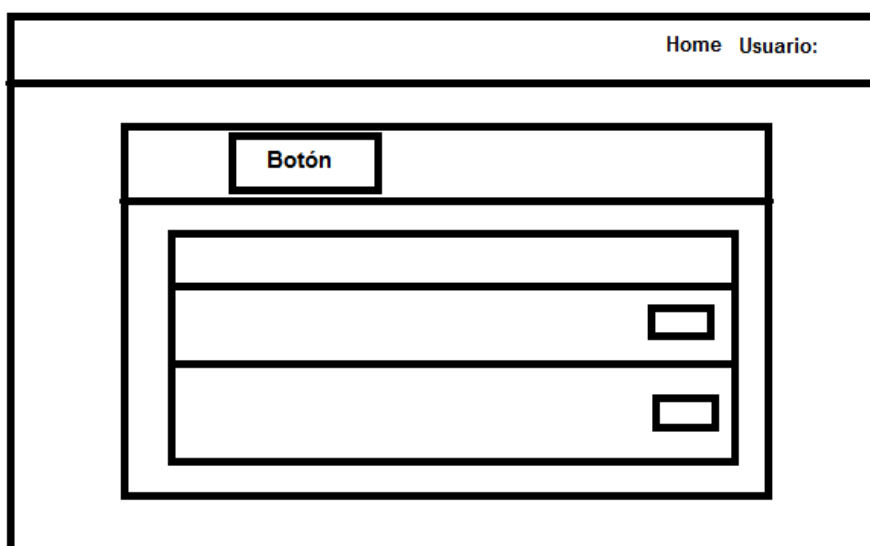


Figura 8. Home del usuario.

3.3 Implementación

En las Figuras 9 y Figura 10 se puede apreciar la interfaces registro y login del usuario ingresando los datos.

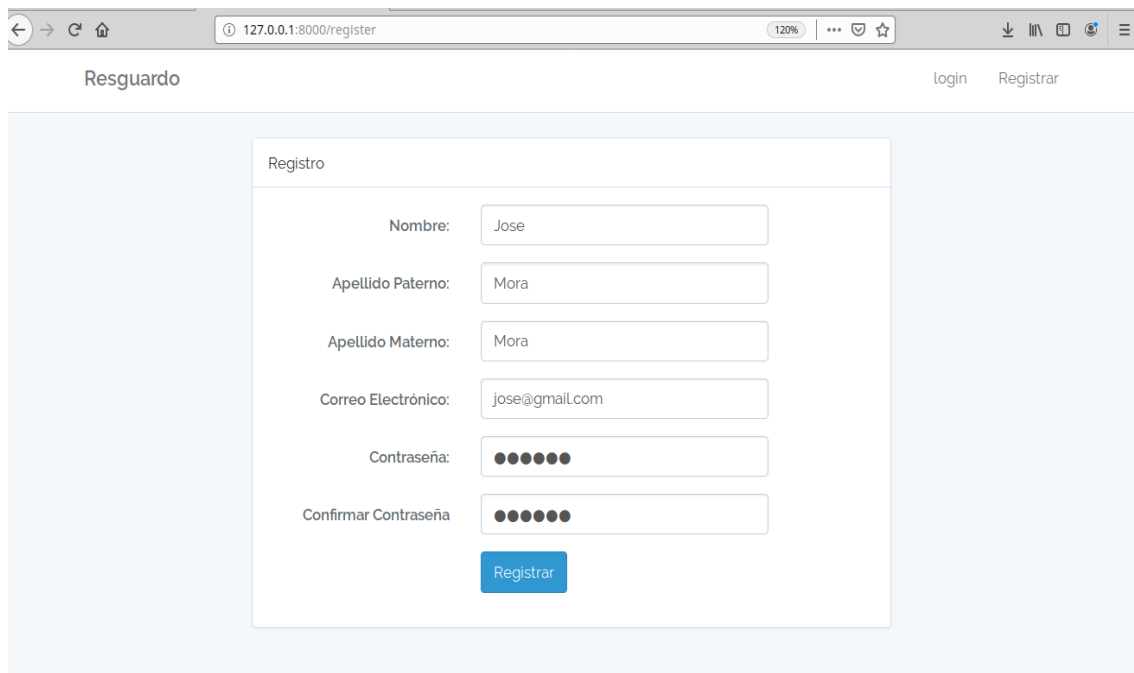


Figura 9. Registro.

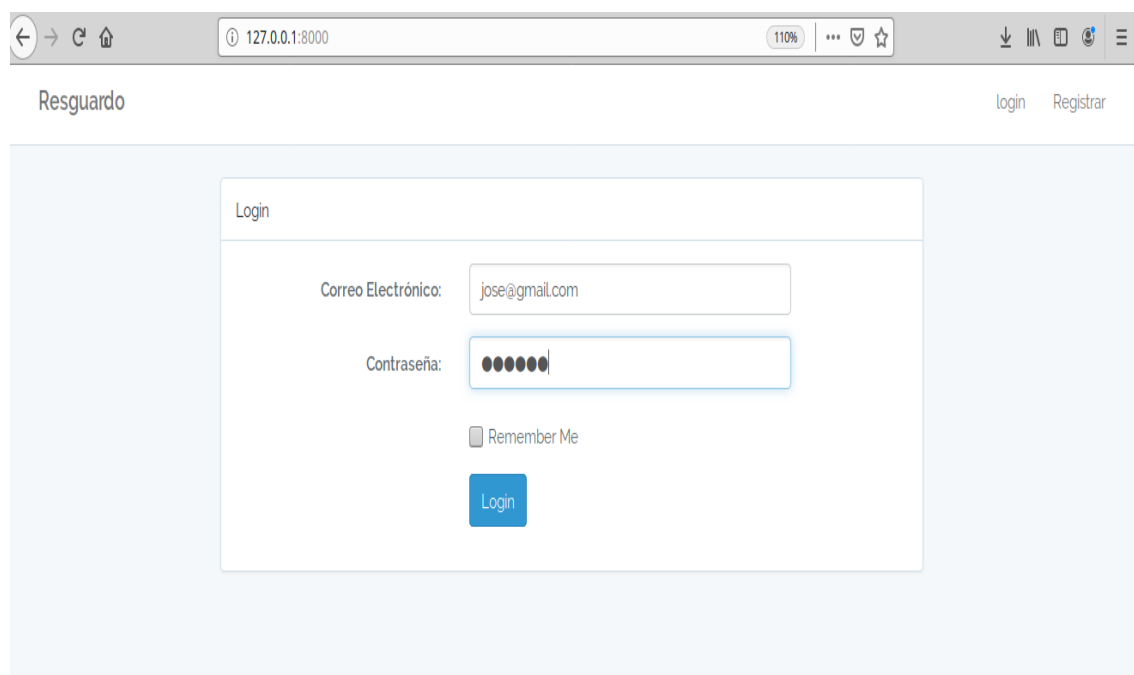


Figura 10. Login

Al ingresar el correo electrónico y contraseñas correctas se mostrará el home del usuario como se muestra en la Figura 11.

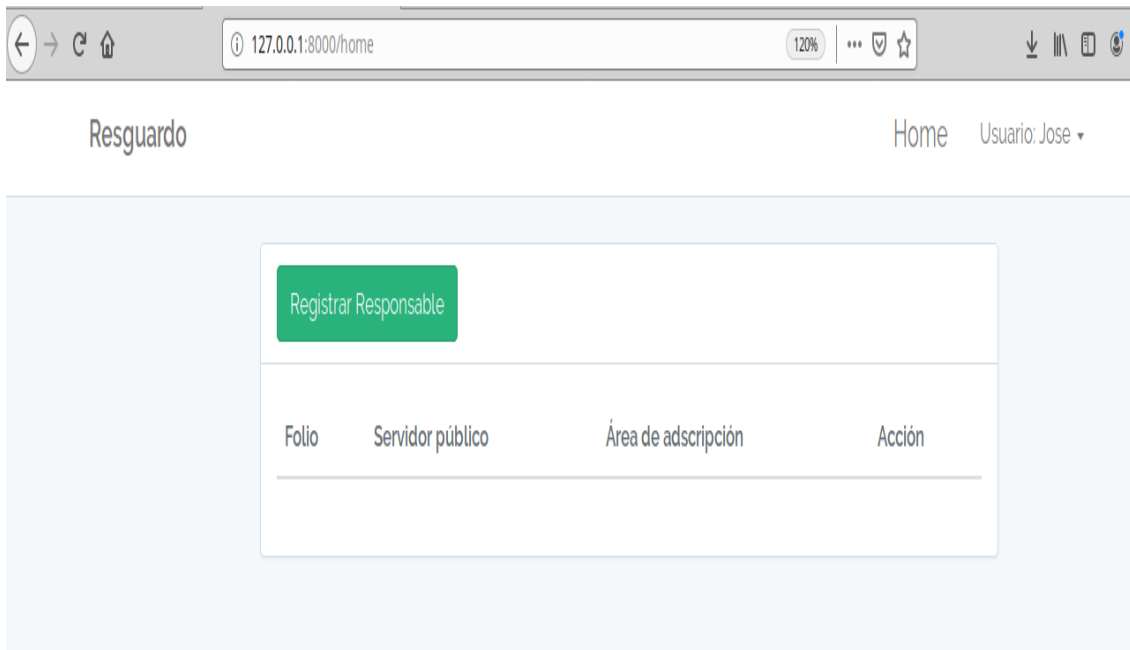


Figura 11. Home

Interfaz usuario puede insertar y consultar datos. En el botón Registro Responsable se muestra el interfaz servidor público que retira en la Figura 12.

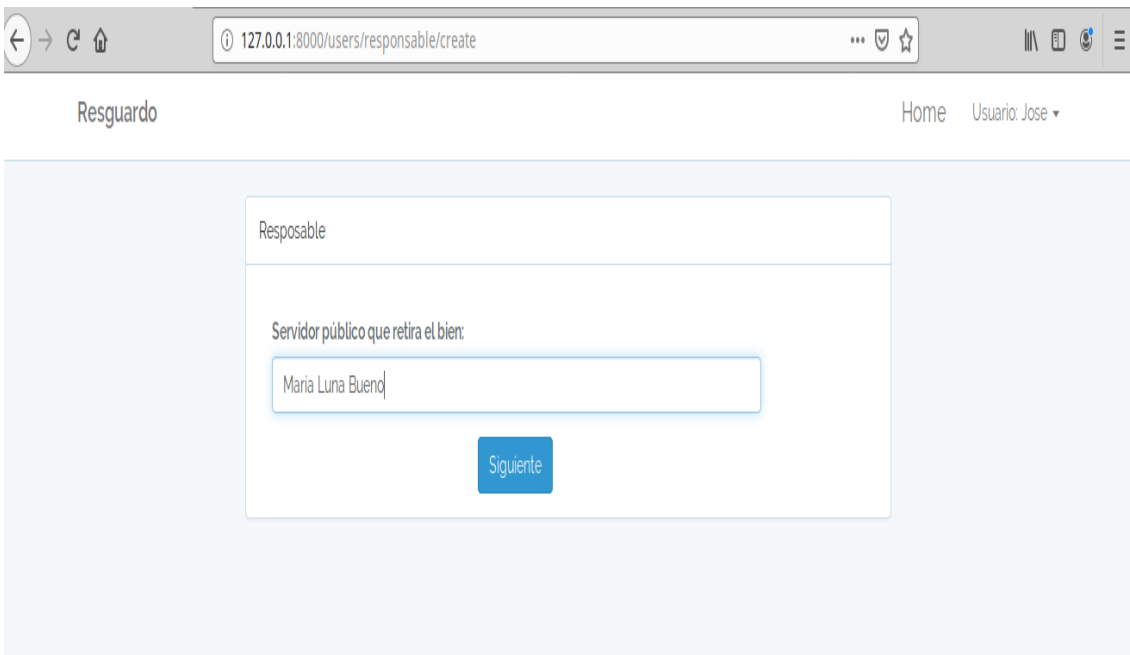


Figura 12. Interfaz servidor público que retira.

Luego de ingresar el nombre del servidor público que retira y botón siguiente mostrará el interfaz de formulario en la Figura 13 y Figura 14.

Resguardo Home Usuario: Jose ▾

Pase de salida y pase multiple

Folio: Tipo de Pase:

Servidor público que retira el bien:

Área de adscripción:

Motivo:

Fecha de salida: Fecha máxima de retorno:

Figura 14. Formulario.

Laboratorio de Visión por Computadora

Motivo:

Fecha de salida: Fecha máxima de retorno:

Características del equipo/material.

No	Descripción	No. De Serie o Modelo	Número de Inventario
1	<input type="text" value="Acer"/>	<input type="text" value="IUUIHI-6455-SDFSDF"/>	<input type="text" value="46545"/>
2	<input type="text" value="HP"/>	<input type="text" value="SDFSDF-SUIOHIU32121"/>	<input type="text" value="9498"/>

Nota adicional del equipo:

Figura 15. Formulario

En botón eliminar fila manda un alerta “¿Desea eliminar la fila de la tabla?” como se ve en la figura 16.

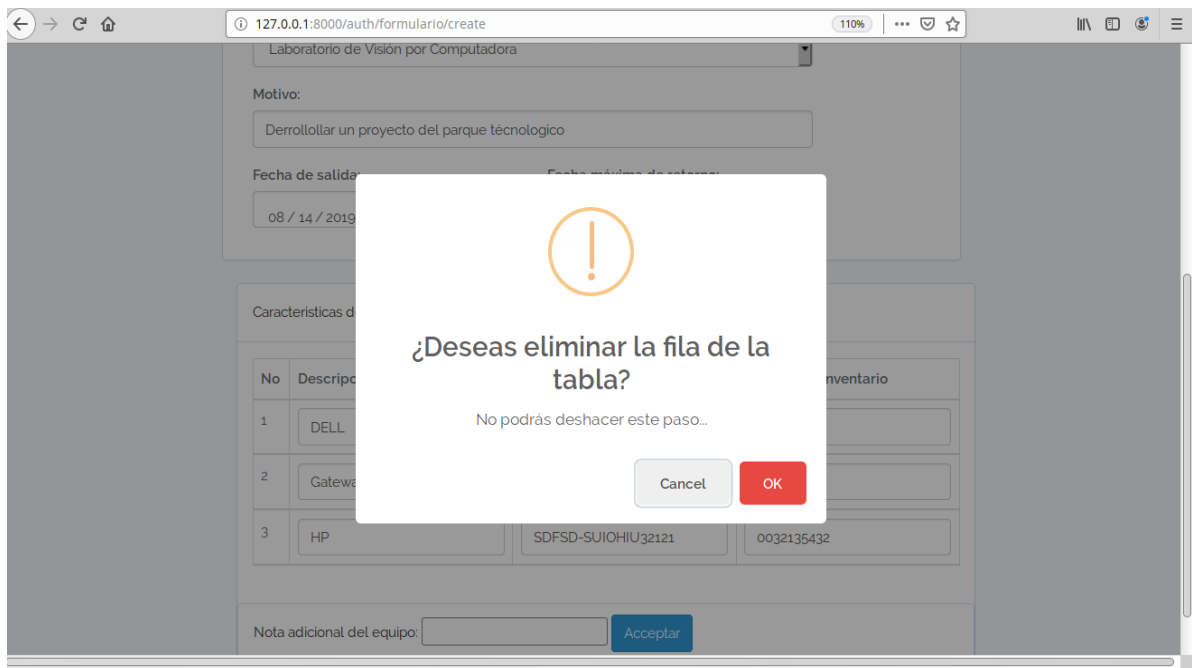


Figura 16. Mensaje eliminar fila.

Si se ingresan los datos correctamente en el home en la Figura 17 se muestra folio, servidor público y área de adscripción y el botón consultar mostrará los datos que fueron ingresados en la Figura 18.

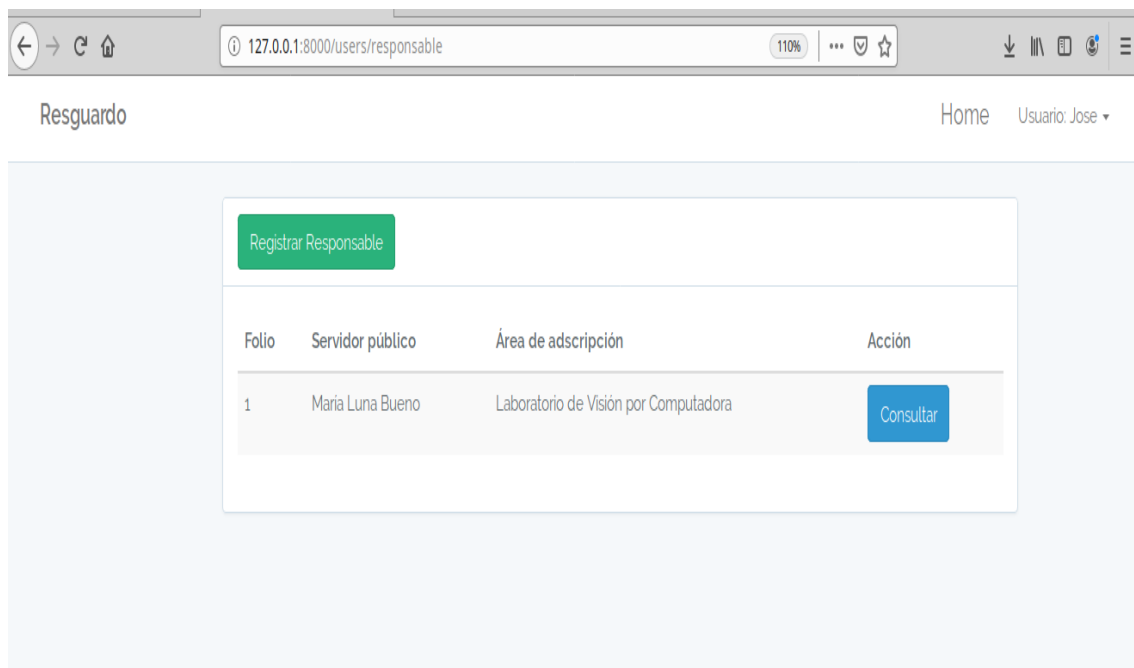


Figura 17. Home.

Consulta

Folio: 1 Pase de salida: X Pase de salida multiple:

Servidor público que retira el bien:
María Luna Bueno

Área de adscripción:
Parque Tecnológico

Motivo:
Derrollolar un proyecto del parque tecnológico

Fecha de salida: 2019-08-12 Fecha máxima de retorno: 2019-08-31

Figura 18. Consulta.

Interfaz administrador lo que hace es insertar, eliminar, actualizar y consultar los datos. Ingresar el correo y contraseña del administrador correctamente se mostrará el interfaz del administrador como se muestra en la Figura 19 y Figura 20.

Resguardo login Registrar

Login

Correo Electrónico: admin@gmail.com

Contraseña: ●●●●●●

Remember Me

Login

Figura 19. Login.

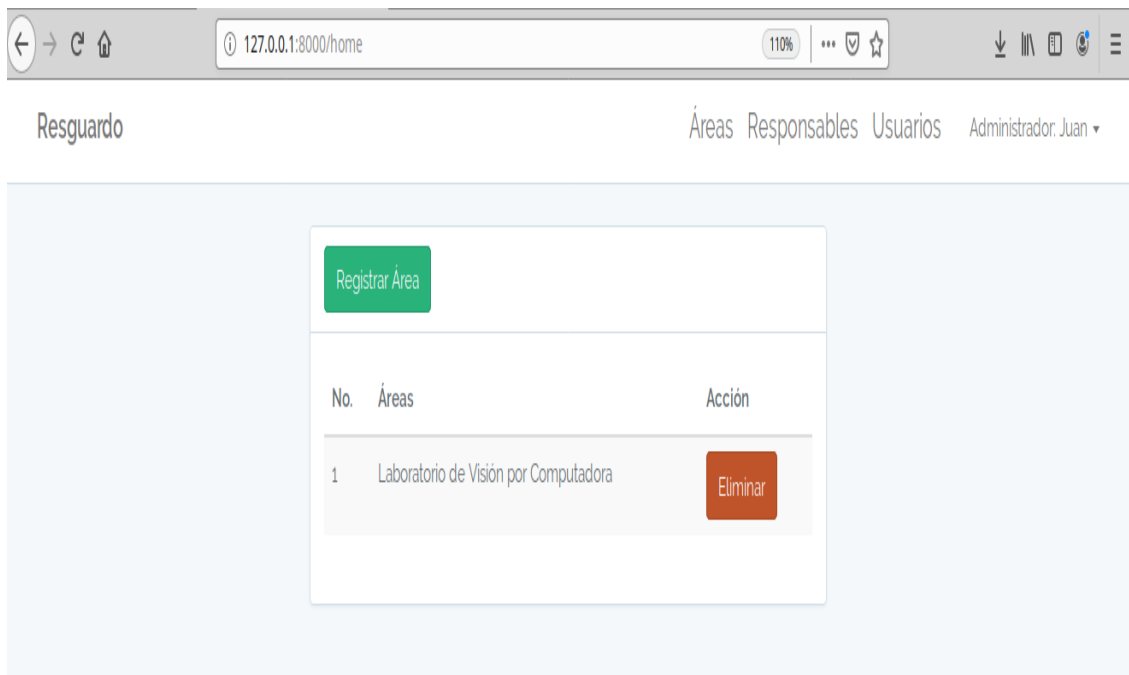


Figura 20. Áreas

Botón Registrar Área lanzará el interfaz para inserta el dato como se muestra en la Figura 21.

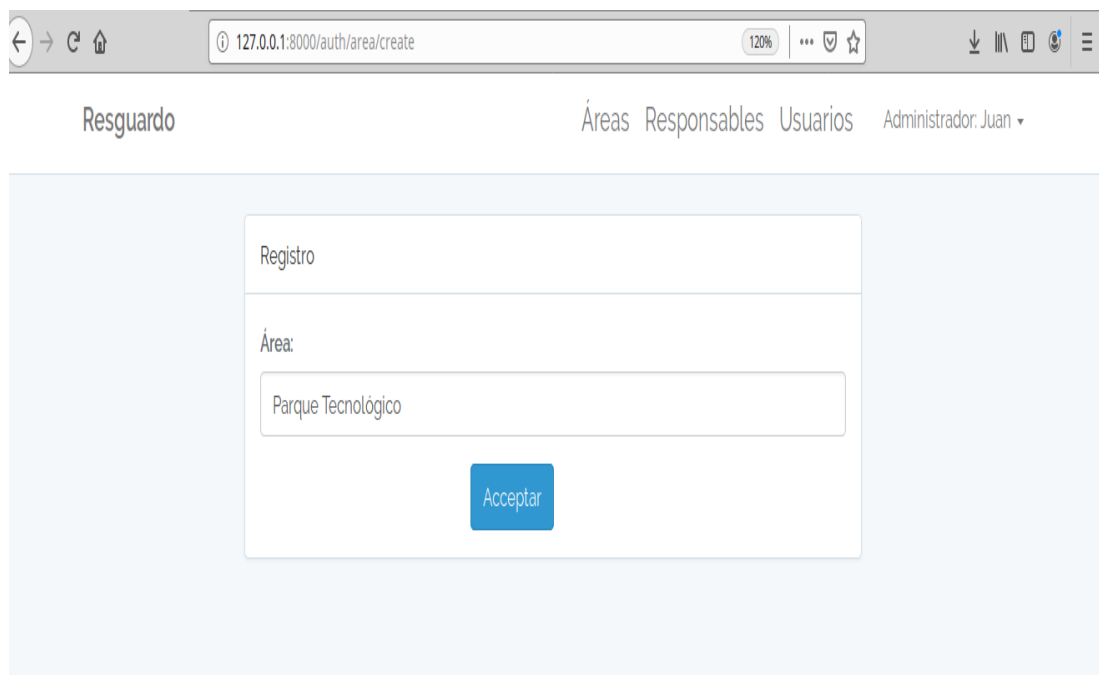


Figura 21. Registro de área.

En el interfaz áreas se encuentra el botón eliminar para eliminar el área lanza un alert “¿Deseas Eliminar?” como se muestra en la Figura 22 y Figura 23.



Figura 22. Interfaz áreas.

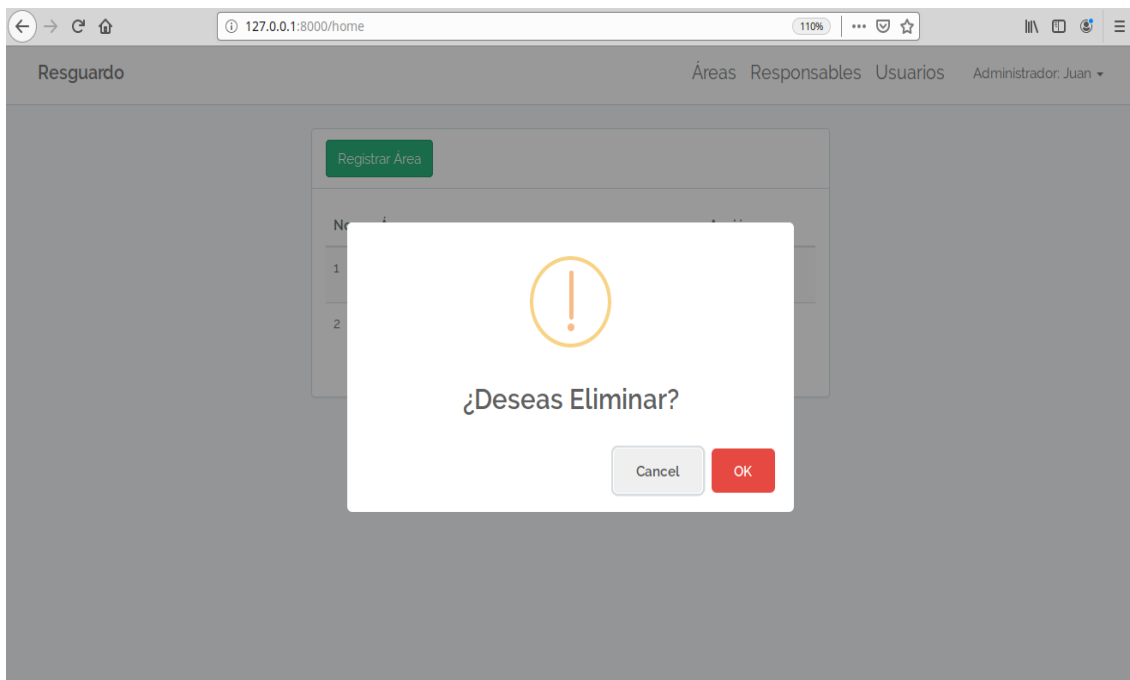


Figura 23. Mensaje eliminar área.

Interfaz “responsables” muestra todos los registros que se haya hecho como se ve en la Figura 24.

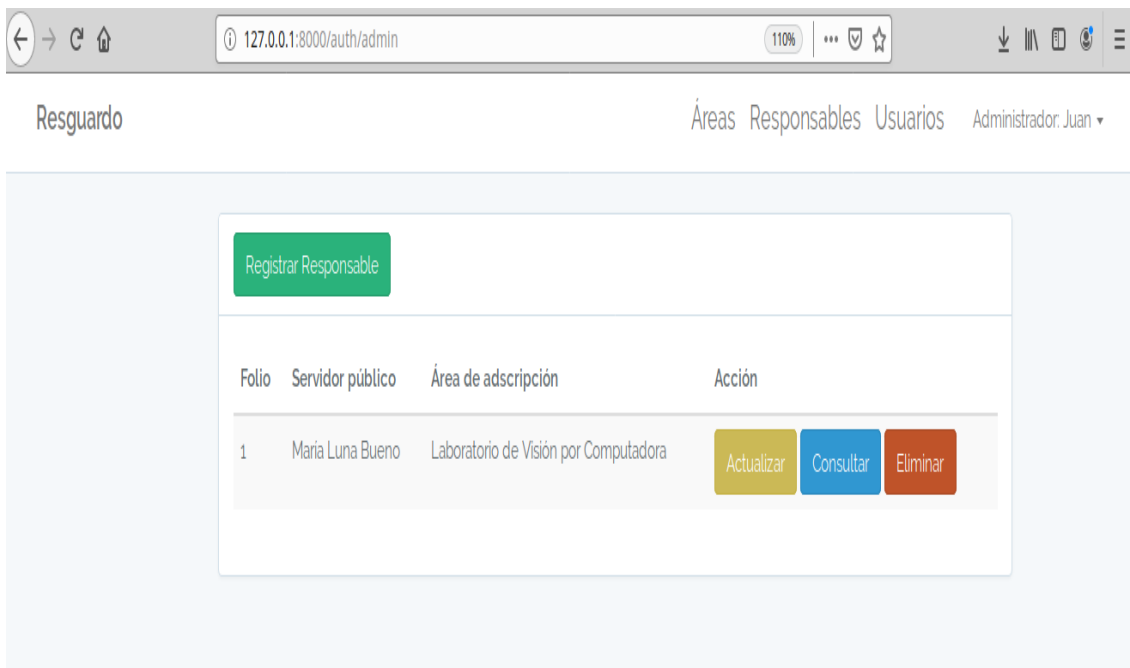


Figura 24. Responsables.

El administrador también puede registrar los servidor público que retira y el formulario como se muestra en la Figura 25, 26, 27.

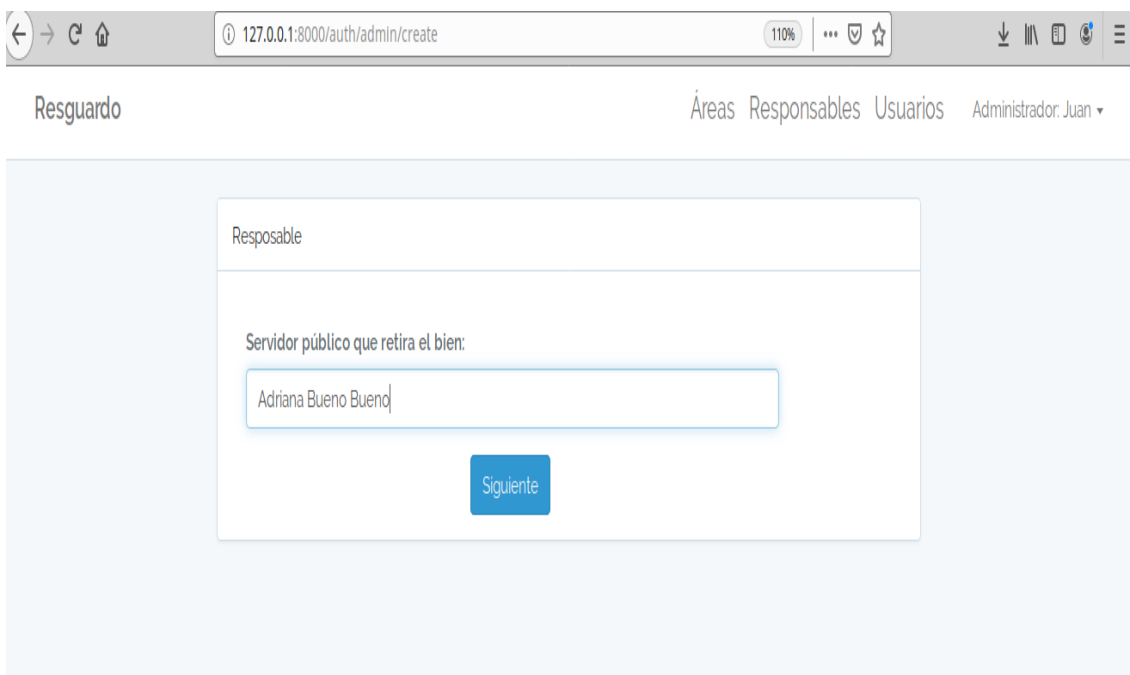


Figura 25. Servidor público que retira.

Resguardo

Áreas Responsables Usuarios Administrador: Juan

Pase de salida y pase multiple

Folio: Tipo de Pase:

Servidor público que retira el bien:

Área de adscripción:

Motivo:

Fecha de salida: Fecha máxima de retorno:

Figura 26. Formulario.

Área de adscripción:

Motivo:

Fecha de salida: Fecha máxima de retorno:

Características del equipo/material

No	Descripción	No. De Serie o Modelo	Número de Inventario
1	<input type="text" value="Gateway"/>	<input type="text" value="HJKH-231-00"/>	<input type="text" value="0032135432"/>

Nota adicional del equipo:

Figura 27. Formulario.

Al ser insertado correctamente los datos, se regresará en el interfaz “responsables” como se muestra en la Figura 28.

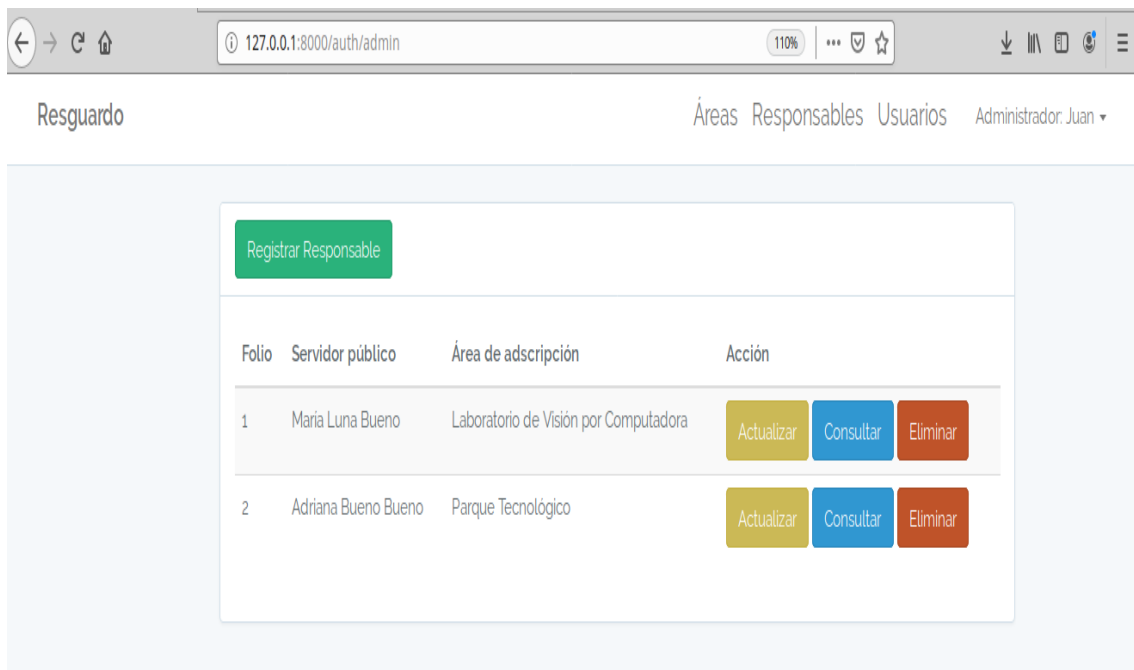


Figura 28. Interfaz responsables.

En la Figura 29, 30, 31 muestra en el botón consultar la direcciona en el interfaz consultar y el botón eliminar se mostrará un mensaje “¿Deseas Eliminar?”.

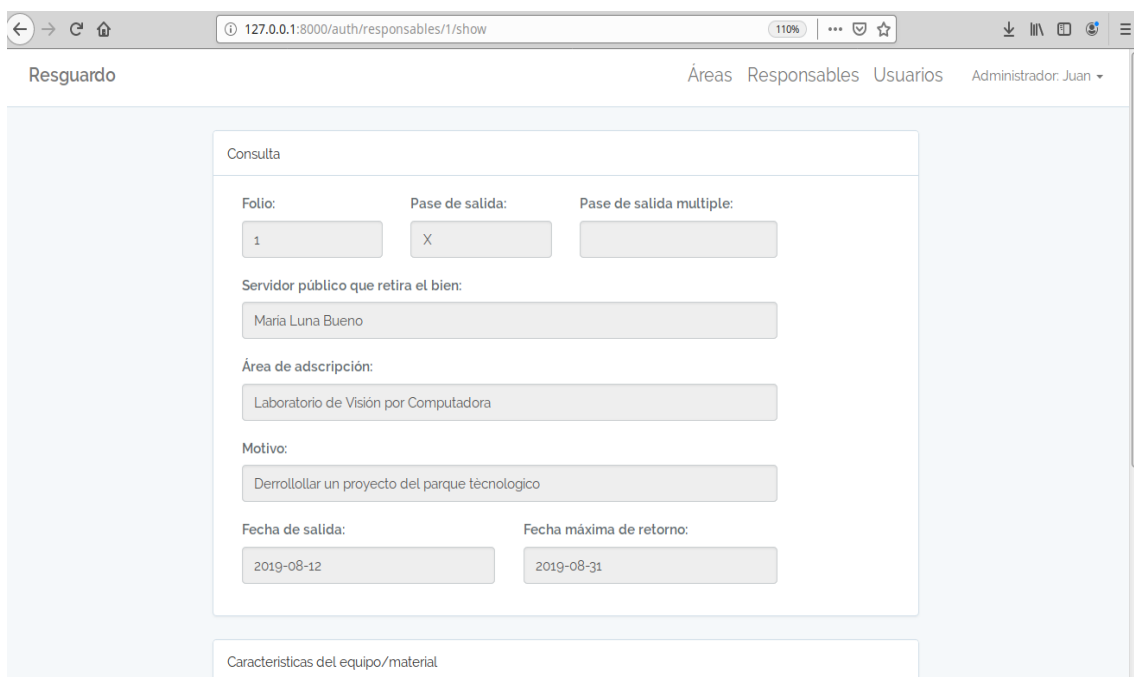


Figura 29. Interfaz consulta.

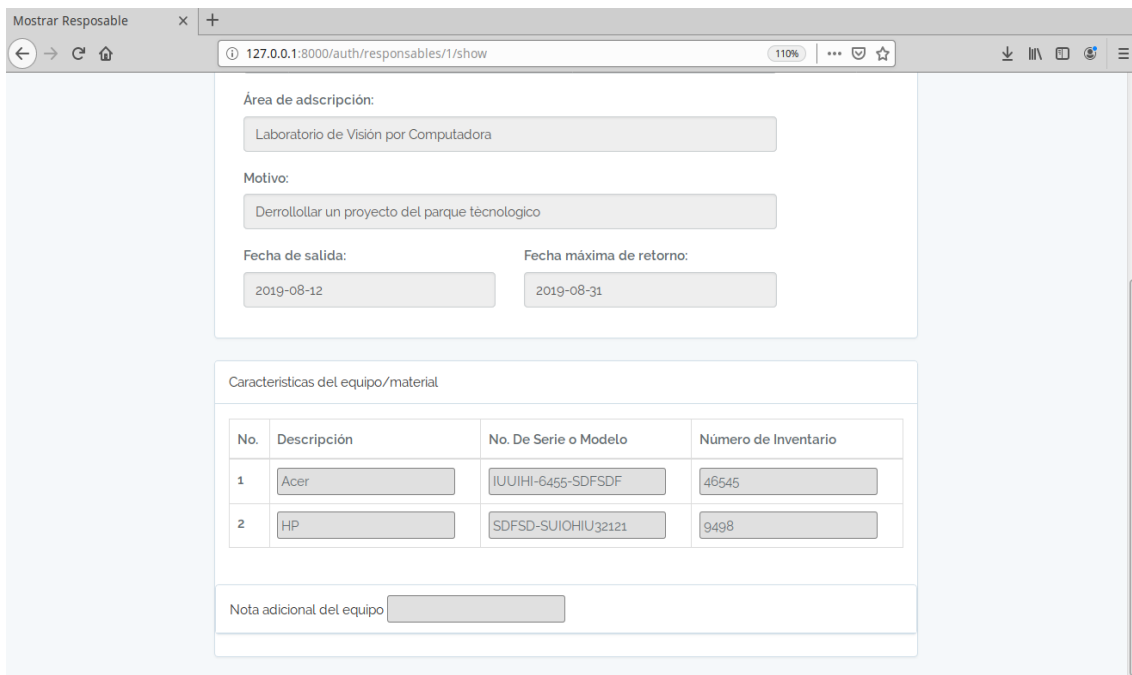


Figura 30. Interfaz consulta

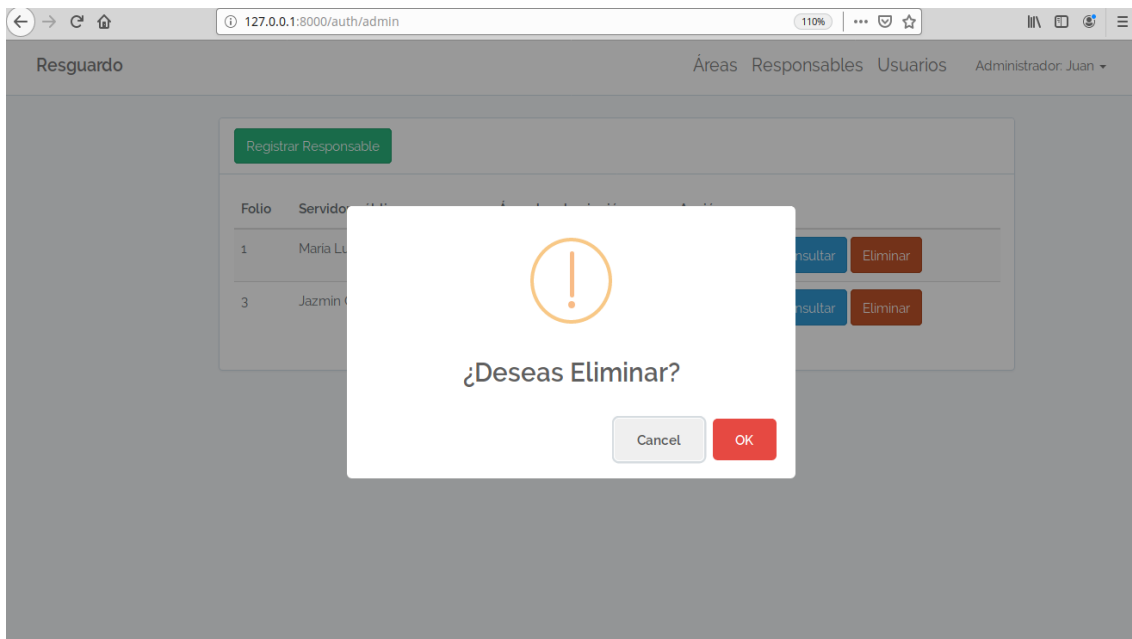


Figura 31. Mensaje eliminar responsable.

En el botón actualizar mostrará el interfaz actualizar formulario como se muestra en la Figura 32 y 33.

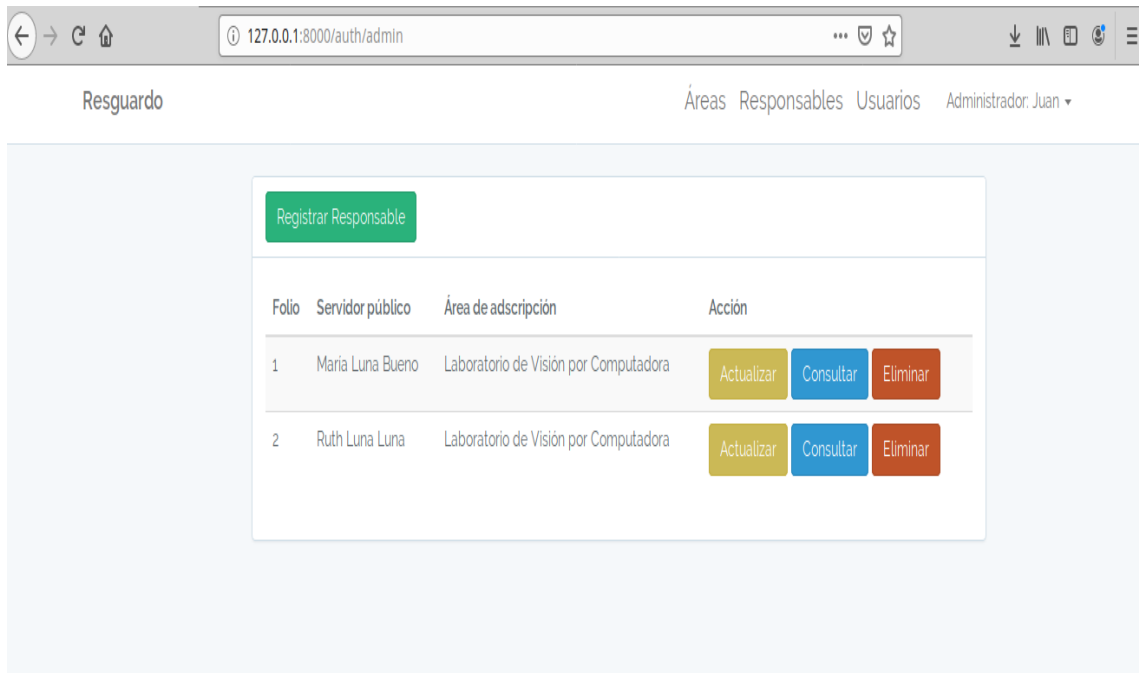


Figura 32. Interfaz responsables.

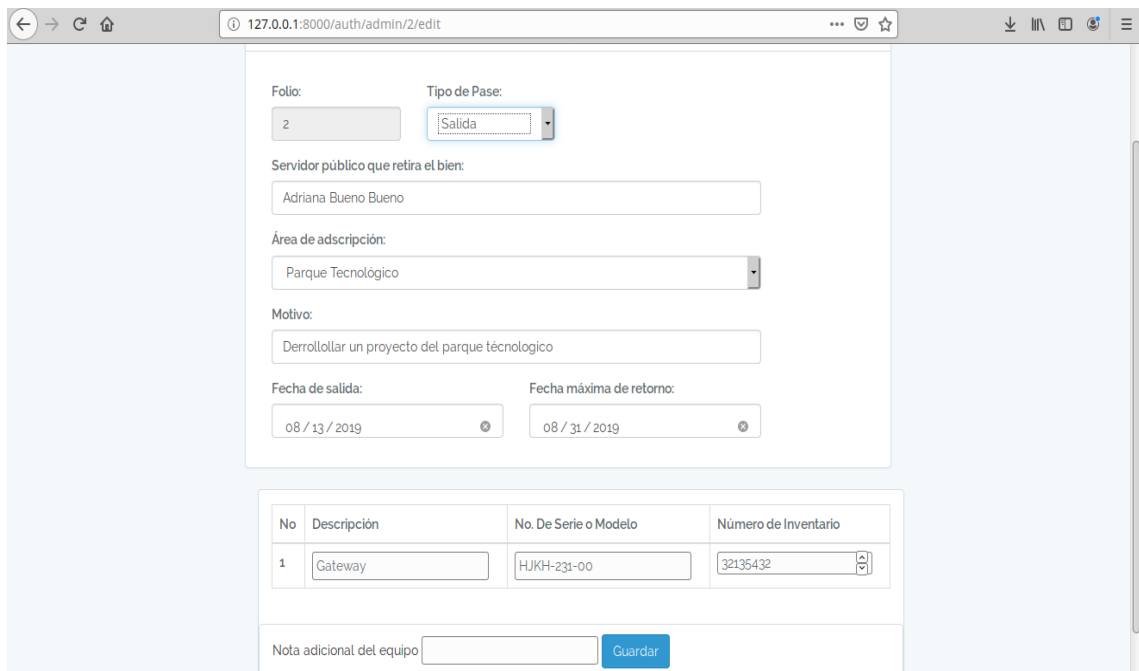


Figura 32. Interfaz actualizar.

Interfaz “usuarios” se muestra los usuarios que están registrados como se muestra en la Figura 33.

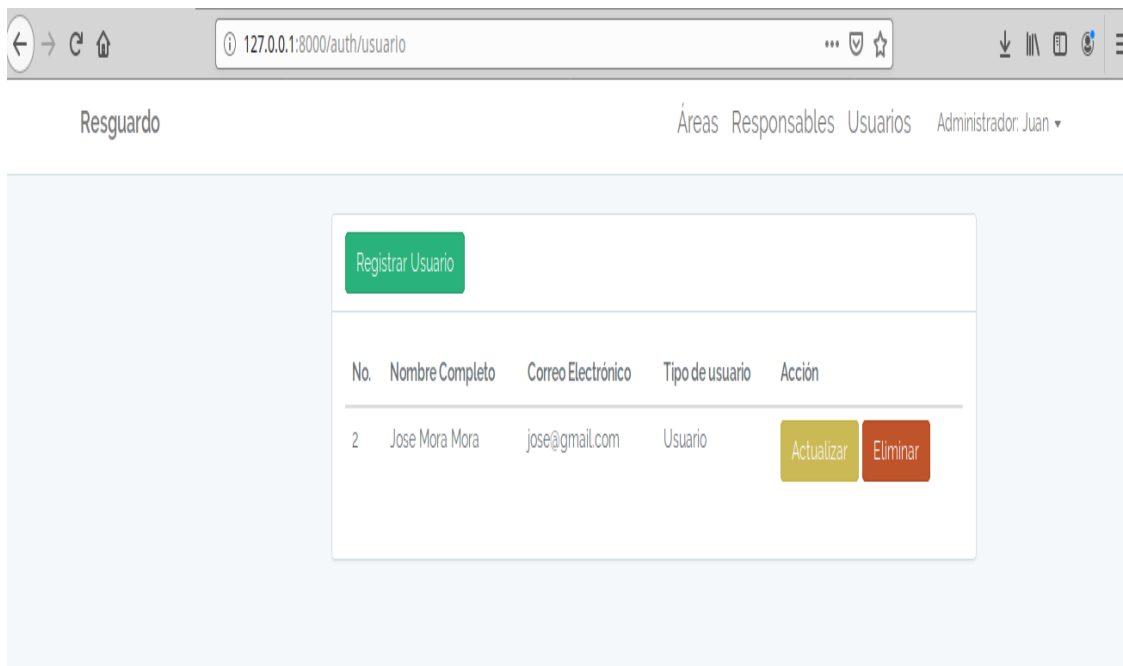


Figura 33. Interfaz usuarios.

En lo botón registrar usuario lanza el interfaz, se insertará los datos en campo tipo de usuario selecciona si será administrador o usuario, el botón actualizar se modifica los datos como seleccionar el tipo de usuario y también se visualiza el mensaje de eliminar como se muestra en las siguientes Figuras 34, 35, 36.

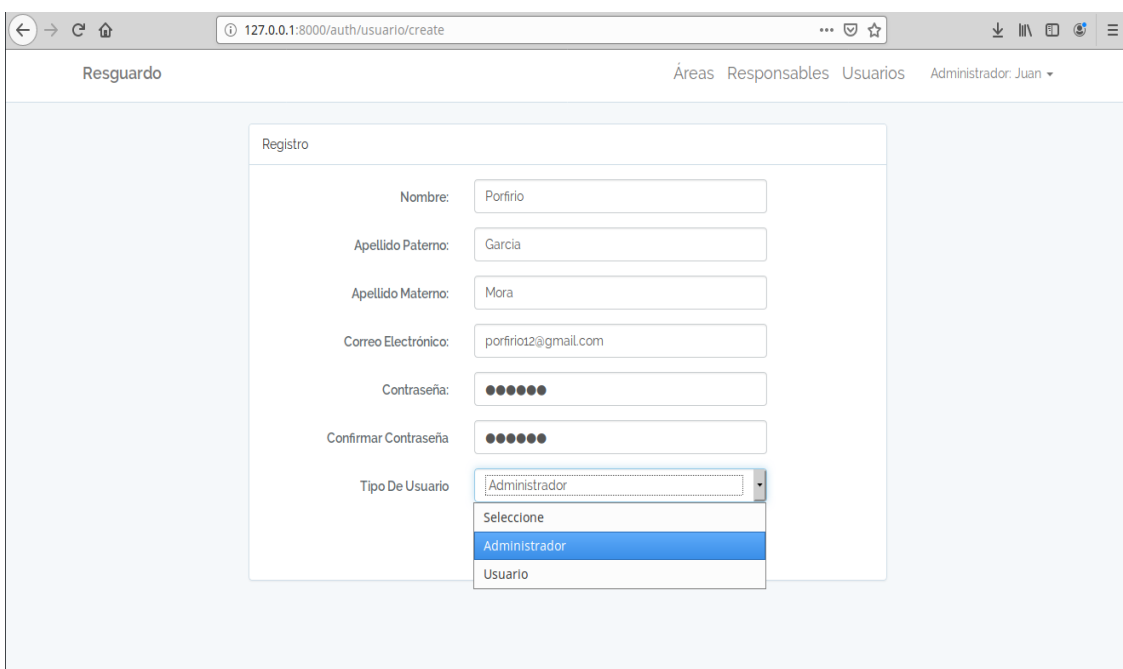


Figura 34. Interfaz registrar usuarios.

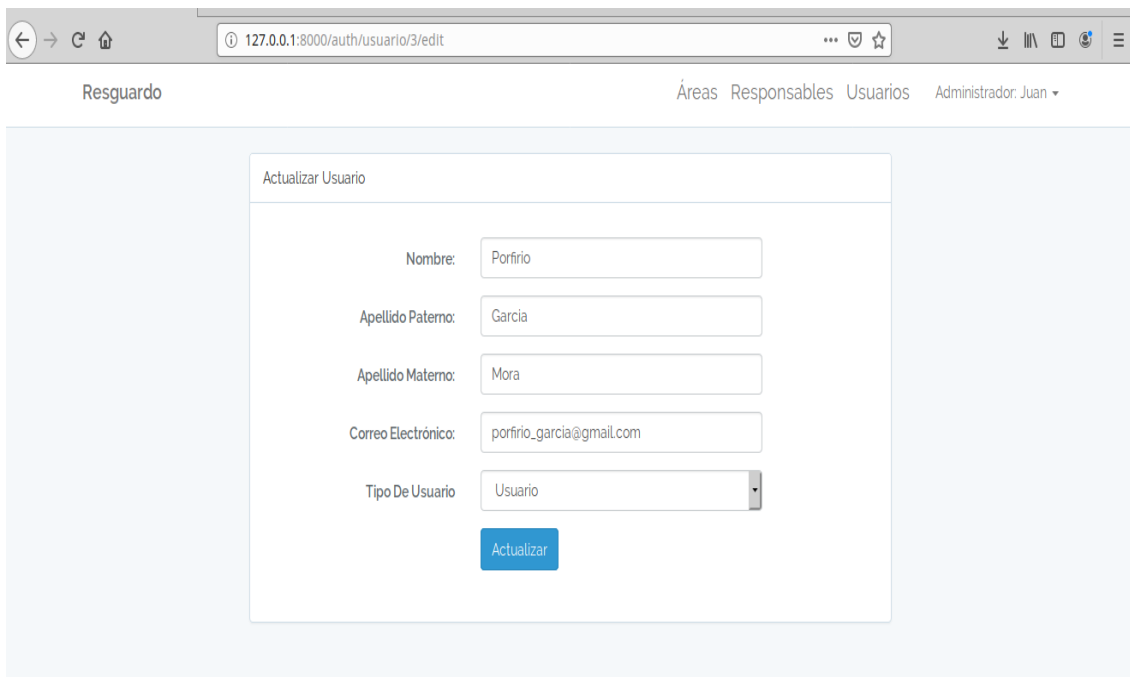


Figura 35. Interfaz actualizar usuarios.

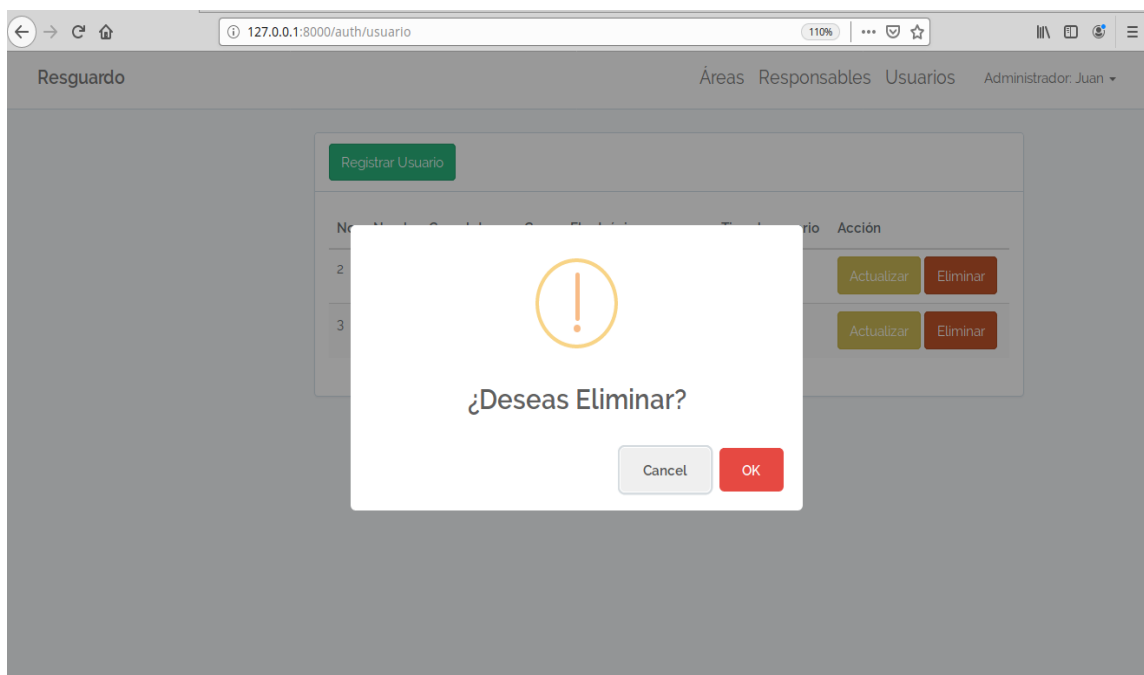


Figura 36. Eliminar usuarios.

3.4 Pruebas

Se realizaron las pruebas del sitio web con un usuario para verificar que no tenga fallas y que cumpla con la funcionalidad como se muestra en las Figuras 37 - 41.

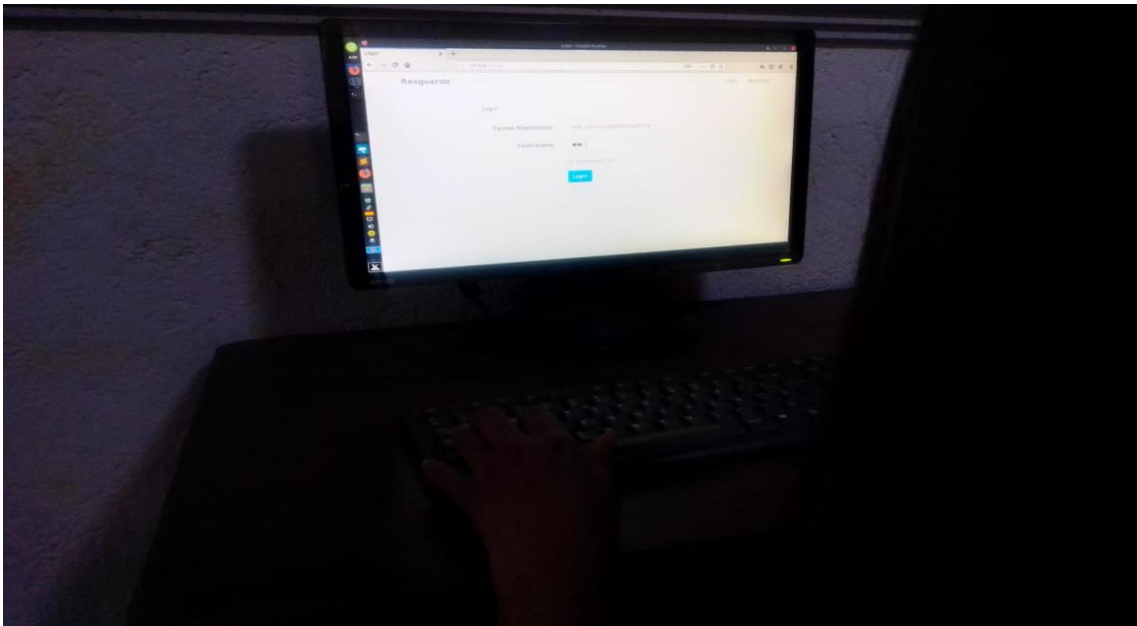


Figura 37. Pruebas del sitio web.

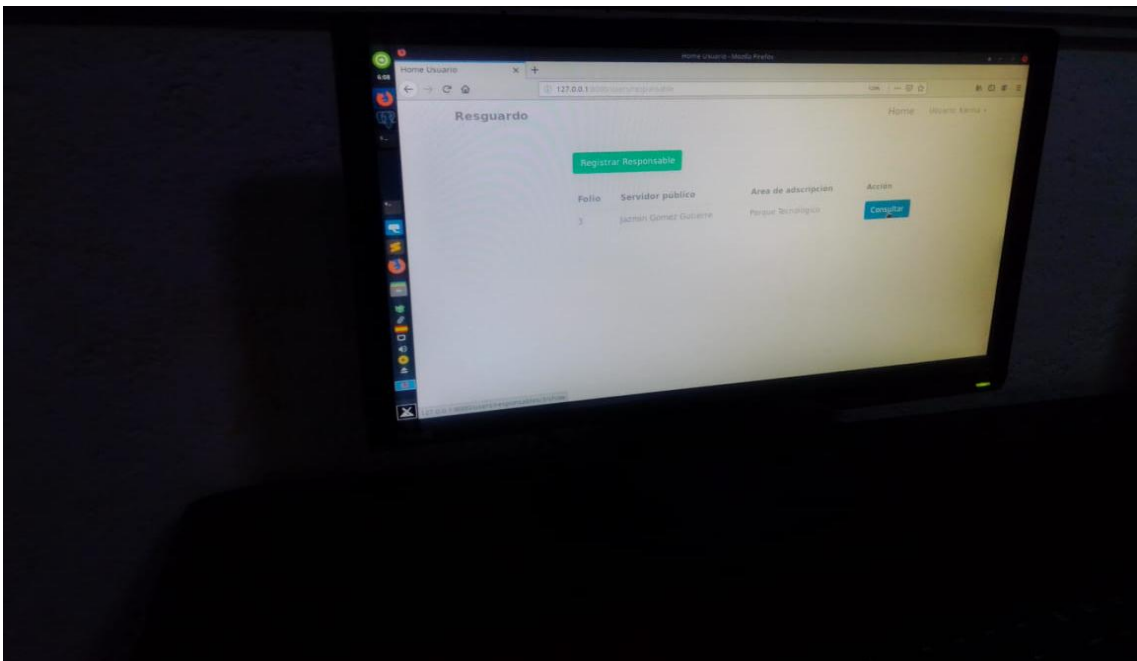


Figura 38. Pruebas del sitio web

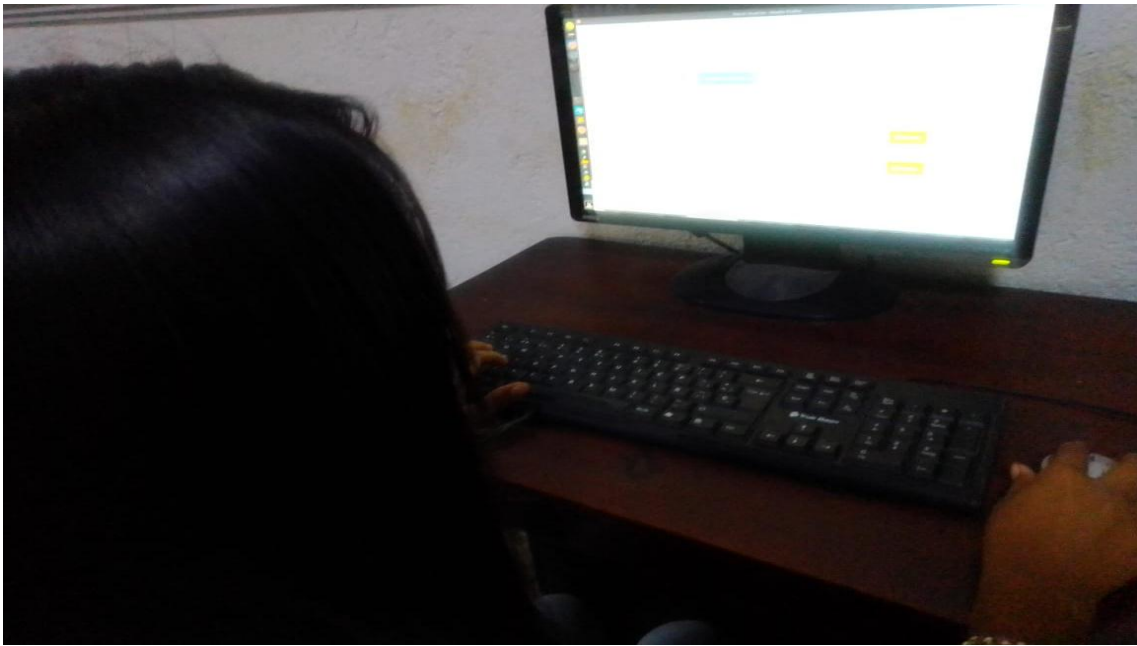


Figura 39. Pruebas del sitio web.

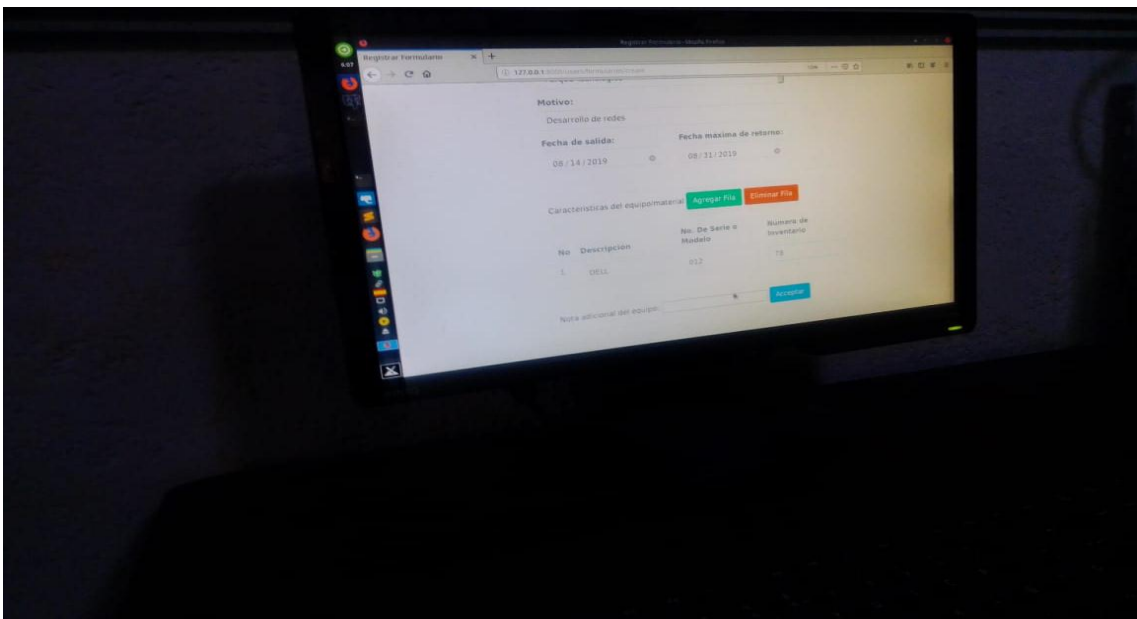


Figura 40. Pruebas del sitio web.

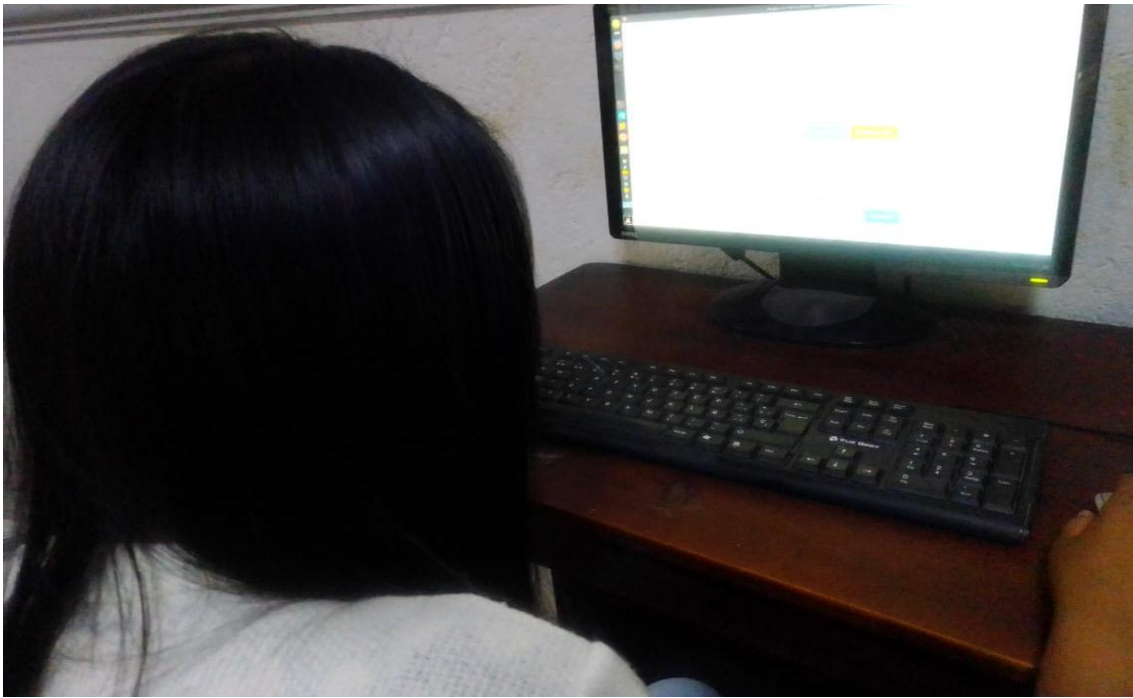


Figura 41. Pruebas del sitio web.

3.5 Mantenimiento

En etapa de mantenimiento se agregó en el modelo entidad-relación la entidad áreas como se muestra en la Figura 42.

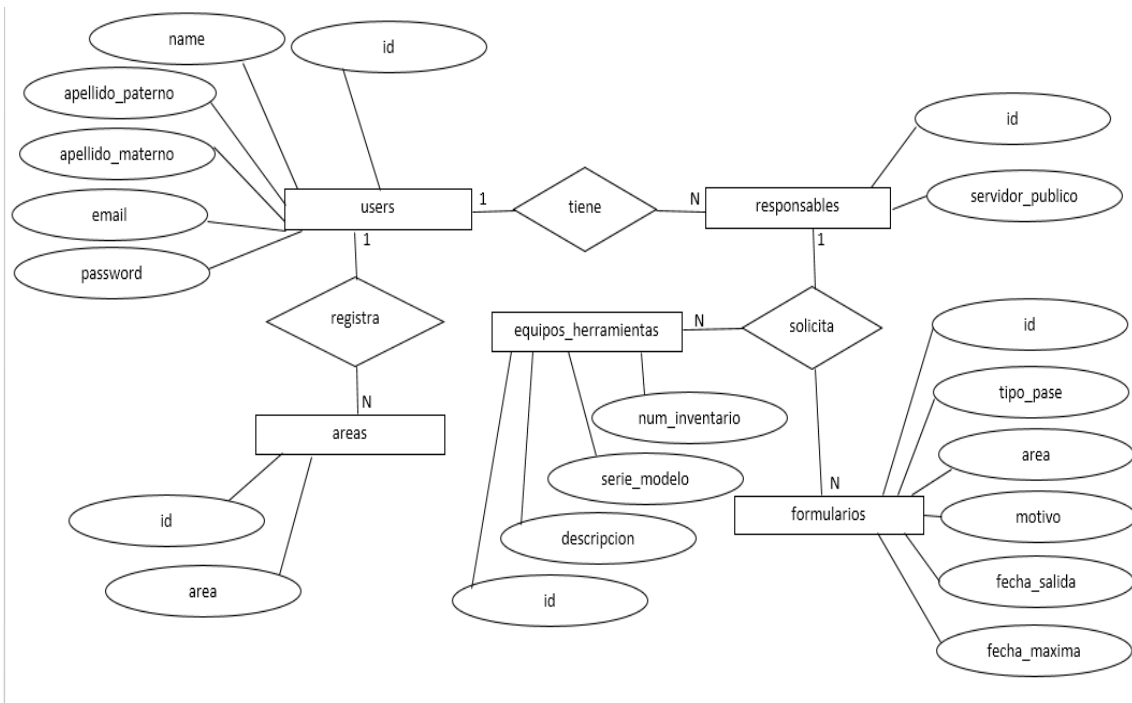


Figura 42. Modelo Entidad-Relación.

4. Conclusiones y recomendaciones

Al finalizar este proyecto hemos comprobado que se han cumplido satisfactoriamente todos y cada uno de los objetivos como por ejemplo: analizar las necesidades específicas del sitio web, diseñar estructura, arquitectura y composición de la base de datos e interfaz, implementar código fuente.

Durante su desarrollo se han ido adquiriendo los conocimientos y experiencias necesarios para crear cada parte del sitio web. Al principio existieron muchas dificultades al realizar el proyecto, empezando por el framework Laravel debido a que no tenía la experiencia necesaria para realizar migración de datos, modelos, controladores y rutas en el sitio web. En este proyecto no sólo se realizó la parte de creación y funcionalidad del sitio web, sino que también se tuvo que leer la documentación de Laravel e investigar cómo funciona cada parte del framework.

Para concluir este proyecto fue útil conocer un framework de código abierto para desarrollar aplicaciones y servicios web con PHP 5 y PHP 7. Las personas interesadas en continuar con este proyecto se recomiendan tener mucha tolerancia porque la curva inicial de aprendizaje puede ser compleja si no has utilizado un framework o no manejas bien el lenguaje, además de leer la documentación para conocer de forma teórica el funcionamiento del framework en el cual trabajarán.

4. Referencias bibliográficas

[1] Shari Lawrence Pfleeger, Joanne M. Atlee. Software Engineering; 3rd Edition. Pearson, (Enero, 2006). Fecha de consulta: 09/07/2019

[2] J. Cervantes Ojeda, María del Carmen Gómez Fuentes. Taxonomía de los modelos y metodologías de desarrollo de software más utilizado. Distrito Federal: Unión de Universidades de América Latina y el Caribe, (Enero-marzo, 2012). Fecha de consulta: 27/06/2019

[3] Carlos Blé Jurado. Diseño Ágil con TDD. España: Lulu.Com, (Enero 2010). Fecha de consulta: 27/06/2019

[4] URL: <http://metodologiaencascada.blogspot.com/> Misty Soloriio. (16 de abril de 2013). METODOLOGÍA EN CASCADA. 2010, de BRAUDE. Fecha de consulta: 27/06/2019.

[5] URL:<http://modelosprocesosdesoftware.blogspot.com/p/modelo-en-cascada-o-lineal-secuencial.html> Unknown. (Mayo 12, 2013). MODELO EN CASCADA O LINEAL SECUENCIAL. Fecha de consulta: 27/06/2019

[6] URL: https://es.ryte.com/wiki/Modelo_en_Cascada#C.C3.B3mo_funciona Modelo de Desarrollo en Cascada - Ryte Wiki, Es.ryte.com. Fecha de consulta: 28/06/2019.

[7] Richos Organization, Laravel, Creative commons (Febrero 22, 2018). Fecha de consulta: 28/06/2019

[8] URL: <https://www.slideshare.net/RafaelLobo4/framework-laravel>, Helen Briceño, Luis Miguel Balza, Betzabeth Linares, Rafael Lobo, Leandro Godoy. Framework Laravel. (Septiembre 19, 2017). Fecha de consulta: 28/06/2019

[9] Javier Novella Latorre, Sistema de Gestión de Base de Datos PostgreSQL (30 Septiembre 2012). Fecha de consulta: 28/06/2019

[10]URL:<http://empresayeconomia.republica.com/aplicaciones-para-empresas/ventajas-y-desventajas-de-postgresql.html> Lucia Velázquez. Ventajas y desventajas postgresQL. (Enero 6, 2011). Fecha de consulta: 28/0/2019

[11] Álvaro Martínez Echevarría, Manual práctico de HTML, Universidad Politécnica de Madrid, España (Mayo del 1995). Fecha de consulta: 28/07/2019

[12] Jorge Andrés Saitama Otálora, VARIAS ASI II, TAREASISTEMAS-Distr.2, ¿Cuáles son las ventajas y desventajas de HTML, URL y HTTP como

tecnologías de base para la consulta y visualización de información?, Universidad Central Colombia, (Enero, 2012). Fecha de consulta: 09/07/2019

[13] Antonio Javier Gallego, Curso Bootstrap 4, Creative commons, (Febrero 20, 2018). Fecha de consulta: 28/06/2019

[14]URL:<http://jorgelessin.com/ventajas-y-desventajas-de-usar-bootstrap/> Jorge Lessi, Ventajas y desventajas de usar Bootstrap, (Mayo 28, 2014). Fecha de consulta: 09/06/2019



Universidad Politécnica de Puebla
Ingeniería en Informática

José Javier Morales Romero
Diana Ivone Tapia López
Rebeca Rodríguez Huesca

Este documento se distribuye para los términos de la
Licencia 2.5 Creative Commons (CC-BC-NC-ND 2.5 MX)