

PONTIFICIA UNIVERSIDAD CATÓLICA DEL ECUADOR

FACULTAD DE INGENIERÍA

ESCUELA DE SISTEMAS

DISERTACIÓN PREVIA A LA OBTENCIÓN DEL TÍTULO DE INGENIERO EN SISTEMAS Y COMPUTACIÓN

"ANÁLISIS, DESARROLLO DE UN SISTEMA WEB PARA LA GESTIÓN DE LA INFORMACIÓN. CASO DE ESTUDIO: SANTUARIO DE NUESTRA SEÑORA DE LA PRESENTACIÓN DE EL QUINCHE"

MARIANO ALBERTO NAVARRETE TEJADA

DIRECTOR: MTR. FRANCISCO RODRÍGUEZ C.

QUITO, 2020

RESUMEN

El Santuario de Nuestra Señora de la Presentación de El Quinche acoge a devotos nacionales y extranjeros brindando servicios de carácter religioso. Actualmente, cuenta con una página web que muestra los servicios y actividades del Santuario, sin embargo, dicha información es escasa y no se muestra fácilmente a los usuarios, de igual manera, el Santuario no cuenta con un sistema que administre la información.

Es por ello, que en el presente trabajo de titulación se diseñó y desarrolló un sistema web para que administre la información del Santuario, mediante el empleo de la metodología ágil Scrum. Para el desarrollo del sistema, en primer lugar, se revisaron las tecnologías que se adapten al caso en cuestión, siendo React para el desarrollo de front-end y Laravel para el back-end. Seguido, se aplicó la metodología Scrum, en el cual el sistema se desarrolló en 4 Sprints y las historias de usuario se utilizaron para el levantamiento de requerimientos.

Posteriormente, se desarrolló un prototipo de página web que muestre la información en tiempo real y la geolocalización de las distintas instalaciones que cuenta el Santuario. Finalmente, se procedió a realizar las pruebas de cada funcionalidad creada basadas en los criterios de aceptación.

DEDICATORIA

Quiero dedicar este trabajo de titulación a mis padres y a mi hermana, por ser esas personas que me brindaron su apoyo incondicional, por ser el ejemplo a seguir y a superarme día a día.

AGRADECIMIENTOS

A Dios y a Nuestra Señora de la Presentación de El Quinche, que con su bendición llenan siempre mi vida y guían cada paso que doy.

A mis padres y a mi hermana, por darme su apoyo incondicional en cada paso de mi vida.

Al Reverendísimo Padre Javier Piarpuzán, Rector del Santuario de Nuestra Señora de la Presentación de El Quinche, por permitirme realizar mi disertación y su presta colaboración para su desarrollo.

A mi director de tesis Mtr. Francisco Rodríguez C. por su dirección, apoyo y confianza en mi trabajo, y sobretodo por su invaluable conocimiento para guiar mis ideas.

A la Pontificia Universidad Católica del Ecuador, mi Alma Máter, y a mis maestros, quienes con sus conocimientos orientaron con profesionalismo mi formación.

A mis amigos, los hermanos que la vida me regaló, por todos los momentos vividos que quedarán por siempre guardados en mi memoria.

CONTENIDO

RESUMEN ..... *I*
DEDICATORIA ..... *II*
AGRADECIMIENTOS ..... *III*
CONTENIDO ..... *VII*
TABLAS ..... *IX*
GRÁFICOS ..... *XI*
1. CAPÍTULO 1: INTRODUCCIÓN *1*
1.1 TEMA *1*
1.2 JUSTIFICACIÓN *1*
1.3 PLANTEAMIENTO DEL PROBLEMA *2*
2. CAPÍTULO 2: MARCO TEÓRICO *4*
2.1 APLICACIONES WEB *4*
2.1.1 Ventajas *4*
2.1.2 Desventajas *4*
2.2 MODELO VISTA CONTROLADOR (MODEL VIEW CONTROLLER) *5*
2.2.1 Definición de MVC *5*
2.2.2 Componentes del MVC *6*
a. Modelo *6*
b. Vista o vistas *6*
c. Controlador *7*
2.3 SCRUM *7*
2.3.1 Teoría de Scrum *8*
2.3.2 El Equipo de Scrum *9*
2.3.2.1 Dueño del producto *9*
2.3.2.2 El Equipo de Desarrollo *9*
2.3.2.3 Scrum Master *10*
2.3.3 Eventos de Scrum *10*

2.3.3.1 Sprint 11
2.3.3.2 Reunión de planificación del Sprint 12
2.3.4 Artefactos de Scrum 13
2.3.4.1 Lista de Producto 13
2.3.4.2 Historias de usuario 13
2.3.4.3 Lista de Pendientes del Sprint 13
2.3.4.4 Incremento 14
2. CAPÍTULO 3: ANÁLISIS DE LA SITUACIÓN ENTRE EL SANTUARIO Y SUS MEDIOS DIGITALES 15
2.1 HISTORIA DEL SANTUARIO 15
2.2 ESTADO ACTUAL DEL SANTUARIO 17
3. CAPÍTULO 4: HERRAMIENTAS ADECUADAS QUE SE ADAPTEN AL CASO DE ESTUDIO EN CUESTIÓN 18
3.1 FRONT-END 18
3.1.1 Frameworks y librerías para el desarrollo de front-end 20
3.1.1.1 Angular 20
3.1.1.2 React 21
3.1.1.3 Vue.JS 22
3.1.2 Selección del framework Front-end 24
3.2 BACK-END 25
3.2.1 Laravel 26
3.2.2 Ruby on Rails 28
3.2.3 Node.JS 30
3.2.4 Selección del framework back-end 31
4. CAPÍTULO 5: APLICACIÓN DE UNA METODOLOGÍA QUE FACILITE Y ASEGURE EL CORRECTO DESARROLLO DEL SISTEMA WEB 33
4.1 ANÁLISIS DE REQUERIMIENTOS 33
4.2 HISTORIAS DE USUARIO 33
4.3 PRODUCT BACKLOG 41
4.4 DESCRIPCIÓN DEL SISTEMA 47
4.4.1 Administrador 47
4.4.2 Usuario 47

4.4.3 Funcionalidades generales del sistema 47
4.4.4.1 Sacerdotes 48
4.4.4.2 Instalaciones 49
4.4.4.3 Servicios 49
4.4.4.4 Cursos 49
4.4.4.5 Sacramentos 50
4.4.5 Funcionalidades aplicadas al usuario 50
4.4.5.1 Cursos 50
4.4.5.2 Servicios 50
4.4.5.3 Sacramentos 51
5. CAPÍTULO 6: DISEÑO DE LA BASE DE DATOS, DESARROLLO DEL SISTEMA WEB FRONT-END Y BACK-END E IMPLEMENTACIÓN DE LA GEOLOCALIZACIÓN DE LOS SERVICIOS DEL SANTUARIO DE NUESTRA SEÑORA DE LA PRESENTACIÓN DE EL QUINCHE 52
5.1 DISEÑO DE LA BASE DE DATOS 52
5.1.1 Entidades 52
5.1.1.1 Instalaciones 52
5.1.1.2 Cursos 53
5.1.1.3 Horarios 53
5.1.1.4 Sacerdotes 54
5.1.1.5 Sacramentos 54
5.1.1.6 Hora de Atención 54
5.1.1.7 Servicios 55
5.1.1.8 Fotos 55
5.1.1.9 Administrador 55
5.1.2 Diagrama entidad relación 55
5.2 SCRUM 56
5.2.1 Especificación de roles 56
5.2.2 Sprint 1 57
5.2.2.1 Planificación del sprint 57
5.2.2.2 Sprint Backlog 57
5.2.2.3 Duración del sprint 60

5.2.2.4 Revisión del sprint 60
5.2.3 Sprint 2 60
5.2.3.1 Planificación del sprint 60
5.2.3.3 Duración del sprint 63
5.2.3.4 Revisión del sprint 63
5.2.4 Sprint 3 63
5.2.4.1 Planificación del sprint 63
5.2.4.2 Sprint Backlog 64
5.2.4.3 Duración del sprint 67
5.2.4.4 Revisión del sprint 67
5.2.5 Sprint 4 67
5.2.5.1 Planificación del sprint 67
5.2.5.2 Sprint Backlog 68
5.2.5.3 Duración del sprint 70
5.2.5.4 Revisión del sprint 70
6. CAPÍTULO 7: EVALUACIÓN DEL CORRECTO FUNCIONAMIENTO DEL SISTEMA WEB Y GUÍA DE INSTALACIÓN 71
6.1 SPRINT 1 71
6.2 SPRINT 2 80
6.3 SPRINT 3 85
6.4 SPRINT 4 95
6.5 GUÍA PARA LA EJECUCIÓN DEL SISTEMA WEB 103
6.5.1 Base de Datos 103
6.5.1.1 Pasos necesarios para la base de datos 103
6.5.2 Sistema Web 103
6.5.2.1 Requisitos 103
6.5.2.2 Instalación del Sistema Web 104
6.5.3 Prototipo de Página Web 104
6.5.3.1 Requisitos 104
6.5.3.2 Instalación del prototipo de Página Web 105
7. CONCLUSIONES Y RECOMENDACIONES 106

7.1 CONCLUSIONES.....106
7.2 RECOMENDACIONES.....107
8. BIBLIOGRAFIA 108

TABLAS

Tabla N 1 Arquitectura de Angular.....20
Tabla N 2 Caracterización y selección del framework front-end para el caso de estudio 24
Tabla N 3 Lenguajes de programación y frameworks más utilizados en back-end.....25
Tabla N 4 Servicios y herramientas de Laravel.....27
Tabla N 5 Caracterización y selección del framework back-end para el caso de estudio 31
Tabla N 6 Historias de Usuario 38
Tabla N 7 Product Backlog.....42
Tabla N 8 Roles del equipo Scrum.....57
Tabla N 9 Sprint Backlog 1.....58
Tabla N 10 Sprint Backlog 2.....62
Tabla N 11 Sprint Backlog 3.....65
Tabla N 12 Sprint Backlog 4.....69

GRÁFICOS

Gráfico N 1 Arquitectura de una aplicación web 5
Gráfico N 2 Interacción de los componentes del MVC 7
Gráfico N 3 Contenidos del Sprint 11
Gráfico N 4 Diagrama de un Sprint 12
Gráfico N 5 Incremento en Scrum 14
Gráfico N 6 Diagrama entidad relación 56
Gráfico N 7 Pantalla Login del administrador 71
Gráfico N 8 Pantalla de inicio del sistema 72
Gráfico N 9 Pantalla de inicio del sistema 72
Gráfico N 10 Pantalla de ingreso de una instalación 73
Gráfico N 11 Pantalla para finalizar el ingreso o ingreso de nuevas fotos 73
Gráfico N 12 Pantalla de ingreso de una foto para una instalación 74
Gráfico N 13 Tabla de información de las instalaciones 74
Gráfico N 14 Modal para buscar una instalación por su nombre 75
Gráfico N 15 Pantalla e información de una instalación 75
Gráfico N 16 Pantalla de edición de la información de una instalación 76
Gráfico N 17 Pantalla para la confirmación de la eliminación de la instalación 76
Gráfico N 18 Pantalla para la confirmación de la eliminación de la instalación 77
Gráfico N 19 Pantalla de ingreso de un nuevo sacerdote 78
Gráfico N 20 Tabla de información de los sacerdotes 78
Gráfico N 21 Pantalla de edición de un sacerdote 79
Gráfico N 22 Pantalla de confirmación de eliminación de un sacerdote 80
Gráfico N 23 Tabla para listar los cursos y/o servicios a cargo del sacerdote 80
Gráfico N 24 Pantalla de ingreso de información de un nuevo servicio 81
Gráfico N 25 Pantalla de ingreso de los horarios del servicio 81
Gráfico N 26 Modal para ingresar un horario del servicio 82
Gráfico N 27 Tabla para listar los servicios 82
Gráfico N 28 Pantalla para editar la información de un servicio 83
Gráfico N 29 Pantalla para listar los horarios que contiene un servicio 83
Gráfico N 30 Modal para buscar un servicio por su nombre 84
Gráfico N 31 Pantalla de información del servicio buscado 84
Gráfico N 32 Pantalla para confirmar la eliminación de un servicio 85
Gráfico N 33 Pantalla para ingresar un curso 86
Gráfico N 34 Pantalla para ingresar los horarios a un curso 87
Gráfico N 35 Pantalla para ingresar los horarios a un curso 87
Gráfico N 36 Pantalla para listar los cursos 88
Gráfico N 37 Modal para buscar un curso por su nombre 88
Gráfico N 38 Pantalla de información de un curso 89
Gráfico N 39 Pantalla de confirmación de eliminación de un curso 90
Gráfico N 40 Pantalla de ingreso de un sacramento 90
Gráfico N 41 Pantalla para agregar horarios a un sacramento 91
Gráfico N 42 Modal de ingreso de horarios para un sacramento 91
Gráfico N 43 Tabla para listar los sacramentos 92

Gráfico N 44 Pantalla de edición de un sacramento.....92
Gráfico N 45 Pantalla para listar las horas de atención de un sacramento 93
Gráfico N 46 Pantalla de confirmación de eliminación de un sacramento 93
Gráfico N 47 Modal para buscar un sacramento por su tipo.....94
Gráfico N 48 Pantalla de información de un sacramento.....94
Gráfico N 49 Pantalla de encabezado de la página web.....95
Gráfico N 50 Pantalla de la barra de navegación, pie de la página y mapa de ubicación del Santuario de Nuestra Señora de la Presentación de El Quinche 96
Gráfico N 51 Pantalla de información de los servicios ofertados.....96
Gráfico N 52 Pantalla de información de los sacramentos 97
Gráfico N 53 Pantalla de información de los cursos.....97
Gráfico N 54 Pantalla de información de un servicio en específico.....98
Gráfico N 55 Pantalla de información de un servicio en específico 98
Gráfico N 56 Pantalla de información de un sacramento en específico 99
Gráfico N 57 Pantalla de información de un sacramento en específico 99
Gráfico N 58 Pantalla de información de un curso en específico.....100
Gráfico N 59 Pantalla de información de un curso en específico.....100
Gráfico N 60 Pantalla de información de un curso en específico.....101
Gráfico N 61 Componente de ubicación de la instalación 101
Gráfico N 62 Carta de aceptación de las funcionalidades 102

1. CAPÍTULO 1: INTRODUCCIÓN

1.1 TEMA

"Análisis, Desarrollo de un Sistema Web para la Gestión de la Información. Caso de Estudio: Santuario de Nuestra Señora de la Presentación de El Quinche"

1.2 JUSTIFICACIÓN

El Santuario de Nuestra Señora de la Presentación de El Quinche es un atractivo turístico religioso muy visitado de la ciudad de Quito. Este Santuario cuenta con una página web que no brinda información actualizada y detallada, puesto que no cuenta con un sistema que administre dicha información, por ende, los turistas nacionales y extranjeros no tienen la información actualizada de los servicios, cursos y sacramentos.

Teniendo en consideración que un aplicativo web es un programa que interpreta los navegadores sin requisitos para su ejecución surge la idea de desarrollar un aplicativo web para este Santuario.

Este aplicativo web tendrá las características de ser multiplataforma y multidispositivo, que podrá administrar, centralizar y facilitar el acceso a la información del Santuario.

Gracias a la información centralizada de este sistema, se podrá mantener al tanto a los turistas nacionales y extranjeros de la información detallada y en tiempo real.

1.3 PLANTEAMIENTO DEL PROBLEMA

El Santuario de Nuestra Señora de la Presentación de El Quinche es reconocido a nivel nacional e internacional, pues en su interior alberga a la imagen de la Virgen de El Quinche, conocida por ser muy milagrosa y a quien acuden miles de fieles devotos mensualmente y acceden a los servicios que brinda el Santuario.

El Santuario ofrece servicios de carácter religioso, es decir, sobre la celebración de sacramentos (bautizos, matrimonios, fiestas rosadas y celestes, celebraciones de bodas de oro y plata), confesiones y sobre todo, la bendición de carros. Esta información se encuentra disponible en trípticos que entrega el Santuario y en la página web del mismo.

No obstante, la página web no puede ser actualizada en tiempo real y la información de los distintos sacramentos, cursos y servicios no es lo suficientemente detallada, ya que la información no se encuentra administrada.

Por este motivo, es importante diseñar y desarrollar un aplicativo web para la administración de la información del Santuario, ya que este aplicativo web podrá ser de fácil acceso desde cualquier dispositivo conectado a Internet.

Igualmente, las sacramentos, cursos y servicios podrán ser visibilizados de manera fácil y amigable, con el fin de brindar una experiencia grata a los usuarios que naveguen en la página web.

1.4 OBJETIVOS

1.4.1 Objetivo General

Desarrollar un sistema web que permita la gestión de la información en el Santuario de Nuestra Señora de la Presentación de El Quinche.

1.4.2 Objetivos Específicos

- Analizar la situación entre el Santuario de El Quinche y sus medios digitales.
- Examinar las herramientas de desarrollo adecuadas que de adaptan al caso de estudio en cuestión.
- Aplicar una metodología que facilite y asegure el correcto desarrollo del sistema web.
- Diseñar la base de datos para almacenar la información de la manera más óptima.
- Desarrollar el sistema web su frontend y backend.
- Implementar la ubicación en tiempo real de los servicios que dispone el Santuario.
- Evaluar el correcto funcionamiento del sistema web.

2. CAPÍTULO 2: MARCO TEÓRICO

2.1 APLICACIONES WEB

Se define como aplicación web a las aplicaciones que los usuarios pueden utilizar mediante internet o una red local (EcuRed, 2020).

La popularidad de las aplicaciones web se debe a la facilidad de su uso y a la eliminación de los instalables para su ejecución, puesto que se necesita solo conexión a internet (EcuRed, 2020).

2.1.1 Ventajas

El uso de aplicaciones web permite ahorrar tiempo en el desarrollo de tareas, no necesitan un instalable y por ende no ocupan espacio de almacenamiento (EcuRed, 2020).

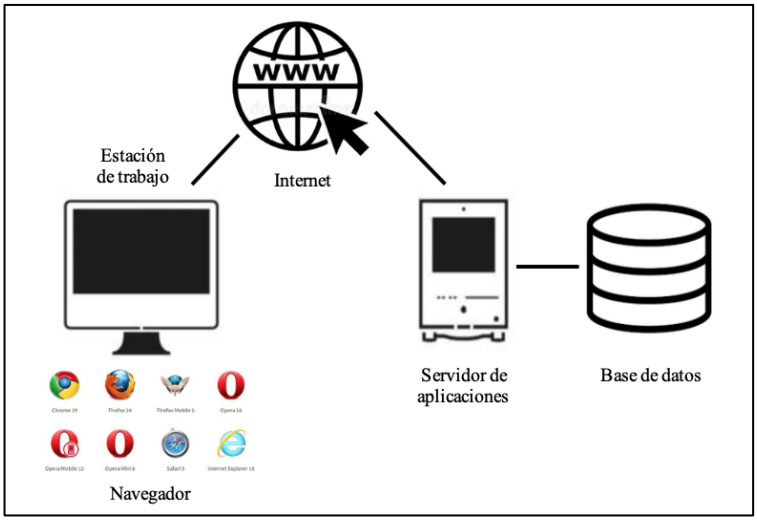
Tiene un consumo de recursos bajo, puede ejecutarse en cualquier equipo sin muchos requerimientos bases y se ejecuta en cualquier sistema (EcuRed, 2020).

Son de alta disponibilidad porque se pueden ejecutar en el momento que el usuario lo requiera, debido a que se alojan en servidores locales o en la nube, por ende la integridad de los datos no se ve comprometida por daños locales del usuario (EcuRed, 2020).

2.1.2 Desventajas

El navegador es limitado para ejecutar ciertas tareas y algunas de estas no aprovechan el nivel de cómputo (Ictea, 2020). Además, depende enteramente de proveedores de internet para su uso (Ictea, 2020).

Gráfico N 1 Arquitectura de una aplicación web



Fuente: Neosoft, 2018
Elaboración: Mariano Navarrete

2.2 MODELO VISTA CONTROLADOR (MODEL VIEW CONTROLLER)

La ingeniería de software se dedica al estudio y creación de procesos que aseguren la calidad de los programas que se crean, el concepto de calidad recae en diversos parámetros como la estructuración de los programas o reutilización del código, y crear patrones que permitan el desarrollo de aplicaciones de mayor calidad, ahí se inserta el MVC ya que cumple con dichos requerimientos (Álvarez, 2020; Espitia, Armao y Carbajo, 2016:11).

2.2.1 Definición de MVC

El MVC cumple con un diseño arquitectónico de software, que separa las responsabilidades de sus componentes base en modelo, vista y controlador (García, 2017; Espitia, Armao y Carbajo, 2016:5).

En su arquitectura existe un sistema central o también llamado controlador, cuyo objetivo es gestionar las interfaces y el repositorio de datos (García, 2017). El MVC es muy utilizado en el desarrollo web, ya que permite modificar cada componente en solitario (García, 2017).

A pesar de que en un principio el MVC fue creado para aplicaciones de escritorio, con el transcurrir de los años ha ido cambiando a multipropósitos, y permite que el modelo orqueste una o más vistas (Espitia, Armao y Carbajo, 2016:9).

2.2.2 Componentes del MVC

El MVC se conforma de tres partes: el modelo, vista o vistas y el controlador:

a. Modelo

En este componente se trabaja con los datos, de manera que contendrá mecanismo para tener acceso a la información y para actualizar su estado (Álvarez, 2020).

El modelo no depende de la vista ni del controlador, puesto que se comunica directamente con el repositorio de datos (Fernández Romero y Díaz González, 2012:49).

b. Vista o vistas[1]

Es la interfaz con la que el usuario va a interactuar con el sistema, todo lo referente a las pantallas, ventanas, páginas y formularios, es decir, que es el resultado de la solicitud (García, 2017).

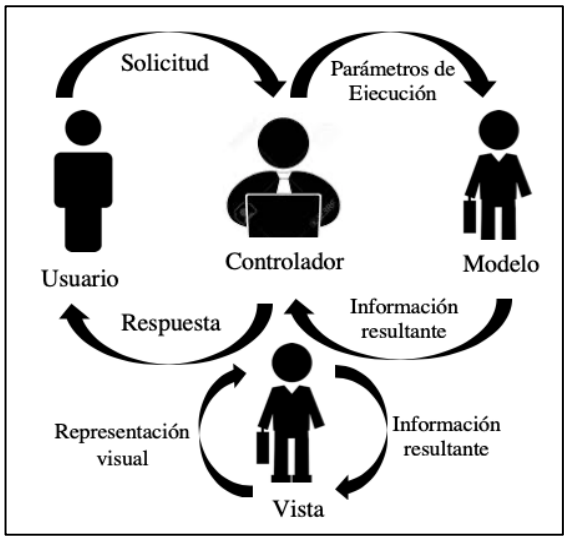
La vista trabaja con datos pero esta nunca se comunicará con el modelo, sino que estos datos serán proporcionados por el controlador (Álvarez, 2020).

[1] Antes de definir este componente, es preciso señalar que algunos autores la denominan como vista y otros como vistas, por cuanto, para la redacción de este componente se denominará como vista.

c. Controlador

Este componente actúa como intermediario entre los componentes: modelo, vista, y el controlador realiza una petición de datos al modelo para manipularlos y que sean servidos hacia las vistas (Universidad de Alicante, 2020; García, 2017).

Gráfico N 2 Interacción de los componentes del MVC



Fuente: Universidad de Alicante, 2020
Elaboración: Mariano Navarrete

La arquitectura del MVC es un modelo utilizado en el desarrollo de muchos software, dado que mediante este patrón se logra una división de las partes de los componentes, esto permite tanto su mantenimiento como la actualización del software de forma fácil y sencilla en poco tiempo, todo llego permite una elevada organización del trabajo (Fernández Romero y Díaz González, 2012:56).

2.3 SCRUM

Scrum es una metodología ágil orientada a objetos, cuyo objetivo es controlar y planificar proyectos de gran volumen, a través de una adaptación iterativa, rápida, flexible, eficaz y fácil

de entender, dándole un valor significativo a través de las mejoras de los Sprints o pequeños bloques (SCRUMstudy, 2013:2; Schwaber y Sutherland, 2013:4; Canive y Balet, 2020).

El marco de trabajo de esta metodología se basa en los equipos Scrum, roles, eventos y artefactos, los cuales tienen objetivos propios que facilitan su uso (Schwaber y Sutherland, 2013:4).

2.3.1 Teoría de Scrum

Esta metodología utiliza un control de procesos empíricos, los cuales se basan en tomar decisiones fundamentadas en la observación (SCRUMStudy, 2013:4).

Scrum se tiene sus cimientos es:

- Transparencia: este pilar es fundamental para que todos los miembros del equipo, así como los interesados de la organización, tengan un conocimiento y entendimiento compartido (Francia, 2019; Schwaber y Sutherland, 2013:5).
- Inspección: es evaluar la información del avance del producto, de su adopción y los outcomes o impacto en los clientes al lanzarse al mercado (Francia, 2019). Los usuarios de Scrum deben realizar inspecciones frecuentes a los artefactos de Scrum para detectar variaciones, no obstante, las inspecciones deben tener intervalos de frecuencia no tan seguidos para que no interrumpa con el trabajo (Schwaber y Sutherland, 2013:5).
- Adaptación: cuando se determinan desvíos en el trabajo el proceso o material que se desarrolla deberá ser ajustados, puesto que el producto resultante no será aceptable (Schwaber y Sutherland, 2013:5).

  El ajuste se deberá realizar lo antes posible para evitar desviaciones mayores (Schwaber y Sutherland, 2013:5).

2.3.2 El Equipo de Scrum

Se conforma por el equipo de desarrollo, el Scrum máster y el dueño del producto,

cada uno de ellos tiene diferentes responsabilidades y debe informar sobre su progreso de diferente manera, tanto entre ellos como para el resto de la organización (Schwaber y Sutherland, 2013:5-6; Roche, 2020).

Quienes conforman este equipo son responsables de entregar productos de forma incremental e iterativa para su retroalimentación (Schwaber y Sutherland, 2013:6).

2.3.2.1 Dueño del producto

Se encarga de tener los mejore resultados en el producto y el equipo de desarrollo, la manera de realizar el trabajo depende de la organización, equipos de Scrum y las personas involucradas, además se encarga de la gestión de la Lista del Producto (Schwaber y Sutherland, 2013:6).

Para que el dueño del producto pueda realizar su trabajo de manera apropiada, quienes forman parte de la organización deben respetar sus decisiones (Schwaber y Sutherland, 2013:6).

Con cada Sprint, el dueño del producto debe realizar una inversión en desarrollo, la cual deberá producir valor, por lo tanto, es indispensable que el poder recaiga en el dueño del producto para que toda decisión que se ejecute sea en bien del producto, ya que él es el portavoz del cliente, el transmisor y retroalimentador de las demandas (Roche, 2020).

2.3.2.2 El Equipo de Desarrollo

Son las personas que entregan incrementalmente el producto terminado que puede ser puesto en producción (Schwaber y Sutherland, 2013:7).

Cabe recalcar que éstas son la únicas responsables de los incrementos, dicho equipo debe estar cohesionado para la efectividad del trabajo (Schwaber y Sutherland, 2013:7).

El equipo debe contar con un número de personas que le permita ser ágil y productivo para realizar el trabajo, es decir, no debe tener menos de tres integrantes porque reduciría la interacción entre pares, y podrían tener limitaciones al momento de ejecutar un Sprint, ya que no pueden entregar a tiempo su trabajo y por ende, se tendrían ganancias pequeñas por productividad (Schwaber y Sutherland, 2013:7).

Por otro lado, no debe tener más de nueve miembros debido a que es necesario la coordinación entre compañeros, además que si el equipo supera este número se vuelve complejo al gestionarlo (Schwaber y Sutherland, 2013:7).

2.3.2.3 Scrum Master

Se encarga de que el equipo trabaje respetando esta metodología, y es el líder del equipo puesto que se encarga de las interacciones con este equipo (Schwaber y Sutherland, 2013:8).

2.3.3 Eventos de Scrum

Son cinco principales para cumplir con el control del proceso, siendo imprescindibles para esta metodología (Roche, 2020). Los eventos en Scrum son predefinidos, con lo cual se crea regularidad y minimiza las reuniones que no hayan sido definidas, con lo cual se facilita también que el control empírico de procesos funcione (Schwaber y Sutherland, 2013:9; Roche, 2020).

Asimismo, son bloques de tiempo y tienen una duración máxima (Schwaber y Sutherland, 2013:9). Cuando comienza el Sprint su lapso de tiempo no es modificable, además, estos terminan cuando se cumpla su objetivo, con ello se asegura que no exista ´desperdicios de tiempo´ durante el proceso (Schwaber y Sutherland, 2013:9).

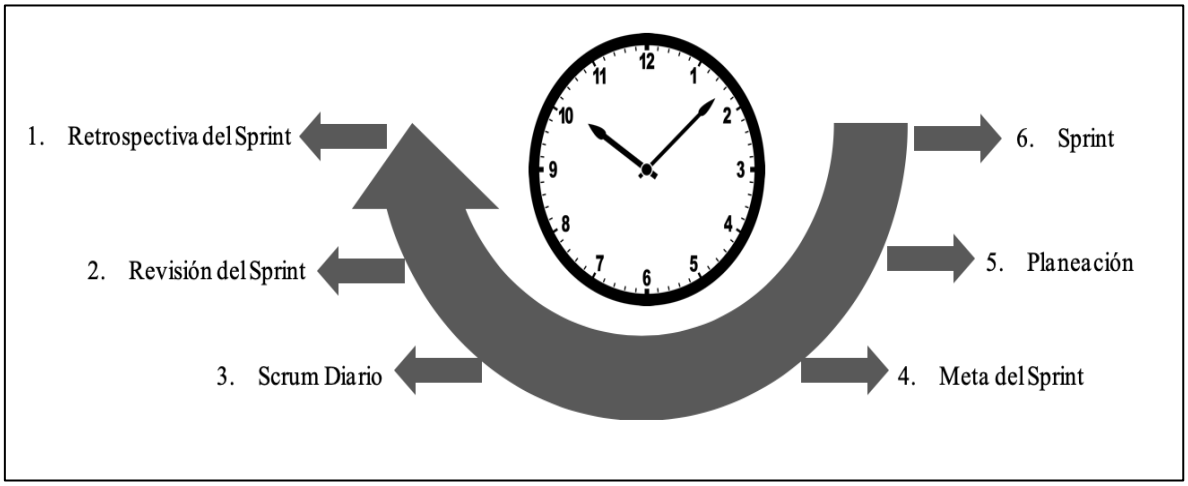
2.3.3.1 Sprint

Es el eje central de esta metodología, se realiza en un tiempo de un mes o menos, y que al final se obtiene un incremento del producto, totalmente utilizable y con la capacidad de estar en producción, cada que finalice el Sprint inmediatamente empecerá uno nuevo(Schwaber y Sutherland, 2013:9).

Cada Sprint debe ser menor a un mes, para evitar riesgo y desperdicio, y, al igual que un proyecto, el Sprint se utiliza para lograr un objetivo (Clip, 2017). El incremento realizado durante cada sprint debe estar terminado, lo que quiere decir: completamente operativo y útil para el cliente (Menzinsky, López y Palacio, 2016: 26).

Cada Sprint contiene una Reunión de Planificación, Scrums diarios, trabajo de desarrollo, revisión del Sprint y la retrospectiva del Sprint.

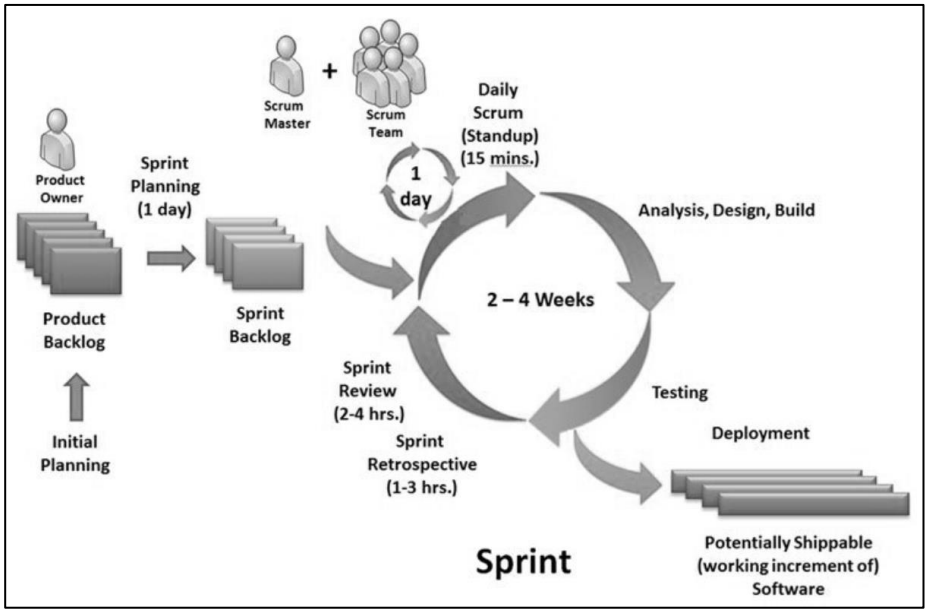
Gráfico N 3 Contenidos del Sprint



Fuente: Clip, 2017
Elaboración: Mariano Navarrete

Los proyectos en Scrum se financian por cada Sprint y es el dueño del producto quien decide a qué o cómo dedicar los recursos, si se logra entender esto, se asegura el éxito en el empleo de Scrum en la organización (Roche, 2020).

Gráfico N 4 Diagrama de un Sprint



Fuente: Lara, 2015
Elaboración: Heys, 2011

2.3.3.2 Reunión de planificación del Sprint

Se realiza para obtener las necesidades del dueño del producto que serán las funcionalidades nuevas del producto que serán implementadas en el siguiente Sprint (Menzinsky, López y Palacio, 2016: 27).

Los participantes serán el Scrum Master, el dueño del producto y el equipo de desarrollo, su duración será de una jornada de trabajo completa (ocho horas), dependiendo del volumen o complejidad de las historias de usuario que se desee incluir en el próximo incremento (Menzinsky, López y Palacio, 2016: 27; Schwaber y Sutherland, 2013:10).

2.3.4 Artefactos de Scrum

En el marco de trabajo de Scrum, se denominan Artefactos a todos entregables tras aplicar Scrum (Schwaber y Sutherland, 2013:15; Roche, 2020). Los principales artefactos de Scrum son tres: Lista de Producto, Lista de Pendientes del Sprint y el incremento.

2.3.4.1 Lista de Producto

Es la lista de las necesidades en el producto y es la única fuente para realizar cambios en el producto (Schwaber y Sutherland, 2013:15). El responsable es el Dueño del Producto, su función es priorizar aquellos elementos que tienen más valor en cada etapa, y realizar un detalle para que el equipo de desarrollo sea capaz de valorarlos y ejecutarlos (Roche, 2020).

Los elementos que la conforman se caracterizan por describir, ordenar, estimar y dar valor (Schwaber y Sutherland, 2013:16).

Esta lista se caracteriza por estar incompleta, puesto que a media que pasa el tiempo ésta irá evolucionando según las necesidades del producto (Schwaber y Sutherland, 2013:15).

2.3.4.2 Historias de usuario

Se denominan como historias de usuario porque en ellas se brinda la información sobre cómo deben ser los requerimientos del proyecto que se va a trabajar, de igual manera, su función es facilitar la interacción directa del cliente (Lara, 2015).

2.3.4.3 Lista de Pendientes del Sprint

Son los elementos que se van a realizar en el Sprint (Schwaber y Sutherland, 2013:16). En esta se visibiliza lo necesario que debe cumplir el equipo de desarrollo para cumplir con el objetivo del Sprint (Schwaber y Sutherland, 2013:16).

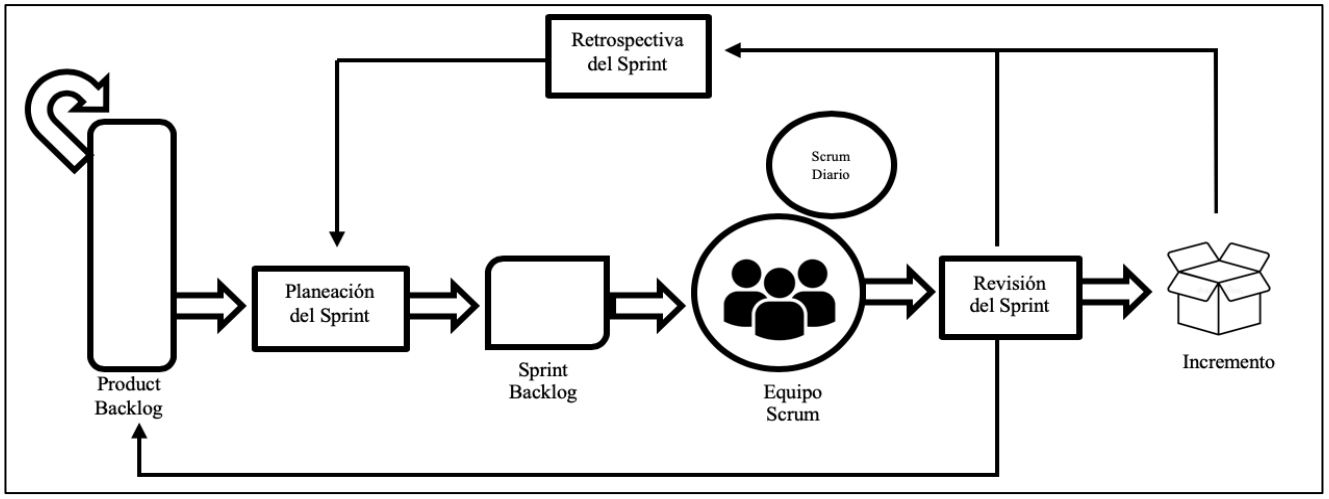
Las personas encargadas de modificarlas son el equipo de desarrollo y ocurre durante el Sprint (Schwaber y Sutherland, 2013:17).

2.3.4.4 Incremento

Es el cumplimiento de todos los elementos de la Lista del Producto en el Sprint (Schwaber y Sutherland, 2013:17; Roche, 2020).

Al finalizar un Sprint el incremento va a ser completado y podrá ser utilizado , y dependerá del dueño su puesta en producción (Schwaber y Sutherland, 2013:17).

Gráfico N 5 Incremento en Scrum



Fuente: Francia, 2017
Elaboración: Mariano Navarrete

2. CAPÍTULO 3: ANÁLISIS DE LA SITUACIÓN ENTRE EL SANTUARIO Y SUS MEDIOS DIGITALES

2.1 HISTORIA DEL SANTUARIO

El Santuario se ubica en la parroquia San Pedro de El Quinche, perteneciente al cantón Quito provincia de Pichincha, y es uno de los sitios religiosos más conocidos a nivel nacional e internacional (Guachamín Lara, Verónica y Tello Gorozabel, María, 2017: XVI).

A este Santuario, acuden diariamente muchos fieles devotos y turistas interesados en visitar y conocer este atractivo religioso, pues en su interior alberga una hermosa imagen católica y mariana, de 62 cm de alto, tallada en cedro por el escultor Don Diego de Robles en el siglo XVI, conocida como la Virgen de El Quinche o "La Pequeñita" como muchos la llaman. Esta venerada imagen fue coronada en 1943 y cuya fiesta se celebra el 21 de noviembre de cada año, éste es el único día en el cual la Virgen deja su altar (Misioneros Oblatos, 2019).

La construcción del Santuario actual data de 1905 y se consagra en su totalidad en el año 1928, ya que el santuario sufrió varias modificaciones y tuvo que ser reconstruido debido al terremoto de 1869 (Archivos de exvotos, 2019). El 02 de mayo de 1959 fue nombrado con la categoría de Basílica Menor por el Papa Juan XXIII, y promulgado como Santuario Nacional en 1985 por la Conferencia Episcopal Ecuatoriana y por Roma, tras una celebración precedida por Monseñor Antonio González, Monseñor de la ciudad de Quito (Iglesias Católicas, 2017; Archivos de exvotos, 2019).

El Santuario está bajo la responsabilidad de la Congregación de Padres Misioneros Oblatos de los Corazones Santísimos de Jesús y de María (Misioneros Oblatos, 2019).

La Congregación de Misioneros Oblatos nombra al Padre Párroco del Santuario conforme a las normas que manejan, dignidad que desde el 02 de marzo de 2014 recae en el Reverendísimo Padre Javier Ramiro Piarpuzán Castro, quien promueve la fe y devoción a María Santísima desde su nombramiento.

El Santuario posee una librería de carácter religioso, un convento ubicado en su interior destinado a los Misioneros Oblatos y personal que labora en él, un Campo Mariano ubicado en la parte posterior y un museo al costado derecho, en el cual se exhiben cuadros de los milagros de la Virgen, sus vestidos bordados en piedras preciosas, sus joyas de oro y plata y los obsequios que recibe de autoridades nacionales y devotos (Misioneros Oblatos, 2019). En el interior del santuario se encuentra una habitación especial denominada "recamarín", la cual alberga a la preciosa imagen y permite verla de cerca, a este lugar sólo se puede acceder con autorización previa del Padre Párroco y en una fecha y horario establecido (Misioneros Oblatos, 2019).

El Santuario ha sido visitado por varias autoridades nacionales y extranjeras, presidentes constitucionales, seleccionados de fútbol, artistas nacionales, entre otros, de estas visitas resaltan las de los Papas Juan Pablo II en 1985 y Francisco en 2015 (Ecuavisa, 2014; Iglesias Católicas, 2017).

La popularidad de la Parroquia El Quinche se debe a su Santuario, el cual es una importante actividad no solo turística sino también económica para la parroquia, pues la principal fuente de ingresos es mediante el turismo reliogoso que practica, y cuyo pico más alto es en el mes de noviembre.

Es por ello, que a partir de esta información es importante crear una página web para el santuario, para mantener al día a los turistas sobre las actividades que se realizan y de esta

manera fomentar y fortalecer el crecimiento del mismo ante los miles de personas que acuden a él.

2.2 ESTADO ACTUAL DEL SANTUARIO

El Santuario cuenta con un sitio web el cual muestra información acerca de los servicios disponibles. Dicho sitio web es estático, es decir, que los cambios que se realizan en el Santuario no se reflejan en la misma. La información no se encuentra centralizada y es reducida, y el Santuario no cuenta con un sistema para organizar y administrar la información acerca de sus sacramentos, cursos y servicios.

Considerando estos datos, se llega a la decisión de crear un sistema web para administrar la información que posee el Santuario, con el fin de mantener organizada la información que se brinda a los devotos acerca de los diferentes sacramentos, cursos y servicios.

Aprovechando los beneficios que nos brinda el uso de sistemas web, la información que sea cambiada o agregada se reflejará a los devotos en tiempo real, y el diseño web de la página del Santuario presentará la información ampliada de cada tema antes mencionado.

3. CAPÍTULO 4: HERRAMIENTAS ADECUADAS QUE SE ADAPTEN AL CASO DE ESTUDIO EN CUESTIÓN

El desarrollo de aplicaciones web modernas se ha convertido en una tarea compleja, puesto que estas aplicaciones deben cumplir con objetivos que respondan a un modelo conceptual, navegacional y de interfaz de usuario, además, están ligadas a estándares de calidad muy altos. Es por esto que se ha dividido la manera de desarrollar las aplicaciones web en el Frontend y Backend, lo cual permite disminuir la dificultad de su elaboración y facilita la división del equipo de desarrollo, así como también de sus tareas.

La teoría muestra que existen diversas tecnologías para desarrollar el Frontend y Backend que se adaptan a las funcionalidades que pretenden cumplir. No obstante, la elección específica de una tecnología demanda conocer el objetivo del proyecto, puesto que cada una de ellas cumple con un propósito y herramientas determinadas, lo que permite que el proyecto sea realizado con éxito.

Por consiguiente, el análisis antes mencionado del Santuario, permitirá la correcta selección de la mejor tecnología que se adapte a las necesidades y características del caso de estudio de la disertación, y constituye el desarrollo del presente capítulo que se detalla a continuación.

3.1 FRONT-END

La palabra front-end puede escribirse como front-end o front end, asimismo puede utilizarse en idioma español como ´frontal´ esto hace referencia a la parte de la aplicación desarrollada del lado front-end (Escuela Tecnologías de la Información S.L, 2020).

Es la capa de presentación de un dispositivo o programa que se encarga de la interactividad con los usuarios, es por ello que se dice que está del lado del cliente (Chapaval, 2018; Escuela Tecnologías de la Información S.L, 2020).

Según Kavourgias (2015), en el desarrollo front-end se puede elaborar sin contar con una aplicación back-end, esto dependerá exclusivamente de los objetivos de la aplicación y el desarrollador web deberá tener en cuenta la usabilidad y legibilidad de la misma (Valdivia Caballero, 2016:191). Por cuanto es indispensable conocer los lenguajes de programación, así como también los frameworks y librerías que se utilizan dentro de front-end.

Los lenguajes front-end son aquellos con los que se realiza toda la parte de la presentación de la página web, siendo los principales HTM, CSS y JavaScript o JS; de los cuales se desprende una gran cantidad de frameworks y librerías que permiten expandir sus capacidades creando cualquier tipo de interfaz de usuarios (Chapaval, 2018; Escuela Tecnologías de la Información S.L, 2020).

Los frameworks y librerías más utilizadas al momento de desarrollar aplicaciones web son Angular, Bootstrap, Saas, React, Redux, Vue y Svelte, igualmente pueden utilizarse herramientas de diseño gráfico como Photoshop o Fireworks (Chapaval, 2018; Valdivia Caballero, 2016:191).

La elección de un tipo de lenguaje front-end dependerá del foco de ejecución de las aplicaciones web, ya que el objetivo del diseño web es brindar una interfaz amigable al usuario al momento de usar la aplicación web.

3.1.1 Frameworks y librerías para el desarrollo de front-end

3.1.1.1 Angular

Fue creado por Google para desarrollar páginas web de tipo Single Page Application [SPA], utiliza como principal lenguaje de programación TypeScript y cumple con la especificación ES6 (JavaScript/ECMAScript) que facilita su desarrollo (Puciarelli, 2020:8; Campus MVP, 2017; Acosta, 2020).

Angular permite el uso de MVC, está orientado a teléfonos inteligentes o smarthphones y usa directamente el DOM. El DOM es un conjunto de objetos que permite representar documentos tipo HTML, XHTML y XML (Acosta, 2020; Ojeda Valiente, 2018).

La arquitectura de Angular está conformada por módulos, componentes, servicios y directivas, los cuales se describen a continuación.

Tabla N 1 Arquitectura de Angular

Módulos	Son el elemento principal de Angular, proporcionan un contexto o dominio de aplicación para los componentes y servicios que contiene
	Permiten organizar el código de manera que ayudan a gestionar el desarrollo de aplicaciones complejas
Componentes	Es una proporción de la aplicación o página y está contenido dentro de un módulo Está orientado a la experiencia del usuario
Servicios	Es un componente lógico de código reutilizable que cumple con una función específica
	Objetivo principal: proveer de funcionalidad extra a un componente
Directivas	Permite añadir, manipular o eliminar elementos del DOM del HTML y es representada como un atributo más una etiqueta HTML

Fuente: Puciarelli, 2020
Elaboración: Mariano Navarrete

La elección de este tipo de framework dependerá de las necesidades del desarrollador web.

3.1.1.2 React

Es creada por Facebook y nació con el objetivo de ofrecer un mayor rendimiento comparado con otras alternativas existentes en el mercado, y basándose en las necesidades concretas derivadas del desarrollo de la web de esta red social (Álvarez, 2019; García, 2016).

React es muy utilizado para el desarrollo de interfaces de usuario y aplicaciones front-end; su desarrollo se basa en componentes, este tipo de arquitectura permite una mayor reutilización del código y realizar aplicaciones complejas pero sencillas de programar y mantener (ITDO, 2019; Álvarez, 2019). React es una de las primeras librerías cuyo desarrollo es basado en componentes, lo que cual nos permite realizar la arquitectura de la aplicación frontend con muchas ventajas (Álvarez, 2019).

De igual manera, React permite crear aplicaciones con poco código desarrollar aplicaciones web de una manera más ordenada y utilizando menos código, además al tener las vistas asociadas a los datos, no es necesario escribir un código página cuando los datos cambian, cosa que en librerías sencillas es difícil de conseguir y React lo hace automáticamente (Álvarez, 2019).

Uno de los problemas de las aplicaciones tipo JavaScript, es que en muchas ocasiones reciben los datos en crudo del servidor o de un API o servicio web en formato JSON; las librerías de tipo JavaScript y frameworks toman esos datos para producir el HTML que debe representar al navegador, esto complica a buscadores como por ejemplo Google, debido a que el cuerpo de la página no tiene contenido (Álvarez, 2019).

Para evitar esto React permite isomorfismo, un concepto relativamente nuevo pero interesante en el desarrollo de páginas web; esto quiere decir que con el mismo código se puede renderizar tanto en el cliente como en el servidor (ITDO, 2019; Álvarez, 2019).

Por lo tanto, cuando un buscador como Google llega, se le puede entregar con la misma base de código el HTML con el contenido ya renderizado (ITDO, 2019; Álvarez, 2019).

Además, el isomorfismo que permite React lo convierte en una librería completa y adecuada a muchos proyectos distintos, puesto que permite un desarrollo ágil, ordenado y presenta una arquitectura mantenible enfocada en componentes, ofreciendo así un gran ´performance´; por eso se dice que está del lado del cliente (Álvarez, 2019).

El gran performance de React se debe al DOM virtual que posee, el desempeño de React se debe a su funcionamiento, es decir a la hora de renderizar que lo consigue mediante el DOM virtual, lo cual lo hace más rápido. El DOM virtual se encarga de la memoria y gracias a la herramienta de diferenciación que posee entre él y el real, el DOM del navegador se actualiza, estas operaciones permiten actualizaciones de hasta 60 frames por segundo, produciendo aplicaciones mucho más fluidas y con movimientos suavizados (García, 2016; Álvarez, 2019).

Estas características hacen de React una librería interesante que se ocupa de las interfaces de usuario, y la serie de componentes y herramientas que brinda su ecosistema, permite desarrollar cualquier cosa que podríamos hacer con un completo framework, de ahí la gran cantidad de usuarios que lo eligen (García, 2016).

3.1.1.3 Vue.JS

Es utilizado para construir interfaces de usuarios de manera sencilla, fue creado en 2014 por Evan You -quien trabajaba como desarrollador front-end en Google y participó en el desarrollo de Meteor- (Vuejs, 2020; García Pérez, 2019).

Su creador define a Vue.JS como un framework progresivo, esto quiere decir que Vue.JS puede implementarse para proyectos desde básicos hasta complejos como un SPA (Single Page Aplication), asimismo es progresivo porque el framework está dividido en diferentes librerías

con responsabilidades específicas, así el desarrollador va incluyendo los diferentes módulos dependiendo sus necesidades, además puede implementarse fácilmente al crecimiento del proyecto sin necesidad de realizar cambios en el workflow (Caffaratto, 2017:1; Platzi, 2020; Dongil Sánchez, s.f.:8).

Igualmente, Vue.JS se caracteriza por ser escalable, puesto que su librería es pequeña y permite adaptarse a proyectos grandes mediante plugins[2]; es reactivo, esto quiere decir que si cambia una variable en una parte de la vista, Vue.JS actualizará automáticamente su nuevo valor sin necesidad de hacerlo manualmente; es modular, por cuanto el desarrollador tiene más libertad para usar características específicas que podrán ser instaladas a posteriori de acuerdo a sus necesidades (Caffaratto, 2017:6; Platzi, 2020; *"Introducción a VueJS",* 2019).

Vue.JS utiliza el DOM virtual y se basa en componentes, permite crear componentes y utilizarlos en diversas maneras, aunque los componentes pueden ser creados directamente con JavaScript, Vue.JS facilita su creación y ahorra tiempo, de igual manera, el ecosistema de Vue.JS permite a los programadores no depender de una sola herramienta (Caffaratto, 2017:6; Platzi, 2020; *"Introducción a VueJS",* 2019).

A pesar de que en un inicio Vue.JS fue pensado para ser una biblioteca personal, la comunidad de usuarios hizo que crezca exponencialmente, pues ahorra tiempo al programador al no tener que escribir tanto código JavaScript y tiene una buena curva de aprendizaje, convirtiéndolo en uno de los frameworks más populares junto a React y Angular (García Pérez, 2019; *"Introducción a VueJS",* 2019; Dongil Sánchez, 2018).

[2] Plugin: es un fragmento o componente de código que en un programa informático permite añadir una nueva característica al software (Pérez Porto y Merino, 2015).

3.1.2 Selección del framework Front-end

Tras la revisión teórica mostrada se procede a realizar la selección del framework que se adapte a las necesidades y características del caso de estudio, de la siguiente manera:

Tabla N 2 Caracterización y selección del framework front-end para el caso de estudio

	React	Angular	Vue
Características	Respaldado por Facebook	Respaldado por Google	Open Source, respaldado por la comunidad
	Su lenguaje utilizado para el desarrollo es JSX	Su lenguaje utilizado para el desarrollo es TypeScript	Su lenguaje utilizado para el desarrollo es JavaScript
	Utiliza el DOM virtual	Utiliza el DOM real	Utiliza el DOM virtual basado en reat
	Alto rendimiento en aplicaciones a gran escala.	Excelente rendimiento en aplicaciones a gran escala	Alto rendimiento en aplicaciones a gran escala.
	Alta flexibilidad en migraciones a diferentes versiones de la librería	Casi nula flexibilidad de migraciones a diferentes versiones del framework.	Alta flexibilidad en migraciones a diferentes versiones del framework
	Baja curva de aprendizaje	Alta curva de aprendizaje	Baja curva de aprendizaje
	Peso ligero aproximadamente 100kb	Peso ligero aproximadamente 80kb	Peso alto aproximadamente 500kb
	Comunidad en continuo crecimiento	Comunidad ha dejado de crecer, aunque sigue teniendo una comunidad grande.	Comunidad en continuo crecimiento
	Amplia documentación oficial	Amplia documentación oficial	Amplia documentación oficial

Fuente: Andrade, 2018; Abraham, 2019
Elaboración: Mariano Navarrete

Bajo las características anteriormente presentadas tenemos que, Angular es utilizado para crear aplicaciones web que utilicen grandes cantidades de datos, por ende, este framework está optimizado y es utilizado para aplicaciones web de gran escala. Por otro lado, Vue.JS tiene una gran similitud a React, pero la diferencia está en la fácil migración de un aplicativo web a un aplicativo móvil nativo. React está siendo adoptado por grandes empresas como librería de front-end principal, gracias a la labor de Facebook por mejorar su rendimiento en cada versión que libera, igualmente la comunidad se siente muy cómoda desarrollando sus proyectos. Debido a la naturaleza de la aplicación web que se pretende desarrollar, el Front-end escogido es React,

puesto que la aplicación web es de baja o mediana escala y React ofrece una mejor experiencia de desarrollo.

3.2 BACK-END

El back-end se encarga de que todo lo que está detrás de una página web funcione debidamente, se encuentra en pos del escenario para conectarse con el repositorio de datos y el servidor que ocupa la página web, por eso se dice que el back-end está del lado del servidor (Chapaval, 2018; Arjonilla, 2020).

El back-end no tiene contacto con el usuario, puesto que se trata solo de la lógica del negocio y no de gráficas como en el front-end, sino que se trata de programación pura, por tal motivo, es de vital importancia tener un buen back-end, caso contrario la página desarrollada podría dejar de funcionar a pesar de que su front-end se encuentre bien diseñado (Mobile Market Association, 2016; Falcón, 2014; Arimetrics, 2020; Campus MVP, 2015).

Al igual que el front-end, el back-end trabaja con varios lenguajes de programación, entre los cuales destacan; PHP, Phyton, Ruby, C#, Node.js y Java, y de los cuales se desprenden sus respectivos frameworks (Chapaval, 2018; Campus MVP, 2015; Falcón, 2014).

Los lenguajes y frameworks que se muestran a continuación que se seleccionaron, se debe a la gran comunidad de usuarios que los respaldan.

Tabla N 3 Lenguajes de programación y frameworks más utilizados en back-end

Lenguajes de programación	Frameworks
PHP	Laravel, Symfony, Codelgniter, CakePHP, Phalcon
Ruby	Ruby on Rails, Sinatra, Padrino
Python	Django, Pyramid, Web2py
Node.js	Express.js, Koa2, Sails,js
Java	Spring, Dropwizard, Blade

Fuente: Rojas Parra, 2019 y Campus MVP, 2015
Elaboración: Mariano Navarrete

El desarrollo del back-end es esencial para conseguir una buena experiencia para el usuario, pues es en el back-end donde las imágenes, formas, letras y demás hechas código pasan al front-end y son visualizadas por el usuario, así, aunque tengamos las mejores ideas y diseños novedosos, si no existen buenos cimientos programados detrás, la página o aplicación no será viable y terminará siendo un fracaso (Mobile Market Association, 2016).

Por consiguiente, es preciso analizar los principales frameworks para el desarrollo de páginas web, que servirá para la selección del mejor framework que se adapte a las características de la página web a desarrollar en la disertación, los mismos que escogieron por el respaldo que tienen en su comunidad de usuarios.

3.2.1 Laravel

Creado para trabajar con PHP por Taylor Otwell en 2011, es considerado como un ´framework joven´, con una curva de aprendizaje rápida y sintaxis expresiva y elegante que evita el llamado ´código espagueti´, permitiendo un sin fin de funcionalidades (Hostalia, 2016; Baquero García, 2015; Antón Dorantes, 2015).

Laravel se enfoca en la calidad de código, facilidad de mantenimiento, escalabilidad, y robustez en la base de datos, esto le permite desarrollar proyectos pequeños y grandes o muy grandes. A pesar de que utiliza PHP como lenguaje de programación, es fácil de usar y desplegar, captando de esta manera una gran cantidad de usuarios (Laravel, s.f.:3; Desarrollo Web, 2020).

Laravel tiene la mayor parte de su estructura formada por dependencias, de manera especial por Symfony, esto implica que su desarrollo dependa también de sus dependencias, además su arquitectura es de tipo MVC (Baquero García, 2015).

Entre las ventajas de utilizar Laravel tenemos costos y tiempos en el desarrollo y su mantenimiento bajos, cuenta con una gran cantidad de paquetes para extender sus

funcionalidades de manera sencilla, robusta y segura; es sencillo al momento de utilizar datos mediante Eloquent, utiliza el Routes with Clousures para reducir las líneas de código; propone un fácil manejo de las rutas de la página o aplicación, y de la generación de URLs amigables que permiten el posicionamiento web; utiliza el sistema de plantillas Blade y dispone de un potente sistema de comandos de consola que permite resumir tareas que toman mucho tiempo mediante Artisan (Hostalia, 2016; Desarrollo Web, 2020; Antón Dorantes, 2015).

Asimismo, Laravel cuenta con varios paquetes de emails con diversas plataformas, permite accesar al repositorio de datos en tiempo real, además Laravel desde su creación a mostrado solidez en cada una de sus versiones y se ha adaptado fácilmente a todas herramientas tecnológicas de la actualidad, de ahí la gran cantidad de usuarios que lo respalda (Desarrollo Web, 2020).

El ecosistema de Laravel está dotado de herramientas y servicios que son útiles para sus usuarios los cuales se describen a continuación:

Tabla N 4 Servicios y herramientas de Laravel

Tipo	Descripción	Característica
Forge	Sistema de administración de servidores tipo panel con el cual se puede crear dominios, instalar Laravel, desplegar aplicaciones, brinda una serie de herramientas para instalar aplicaciones Laravel.	Tiene costo
Vapor	Despliega aplicaciones Laravel sin necesidad de tener un servidor, es decir, es un back-end serverless.	Tiene costo
Nova	Sistema de gestión de contenido para crear los paneles de administración de manera fácil y ahorra tiempo en la programación.	Tiene costo
Envoyer	Permite la integración continua de las aplicaciones Laravel con solo hacer un ´push´ al repositorio, la aplicación se despliega en su nueva versión sin interrumpir el servicio.	Tiene costo
Spark	Permite la creación de aplicaciones con una base de código encima de Laravel, con ello se evita que se repitan determinadas partes del proyecto.	Tiene costo
Homestead	Permite máquinas virtuales que facilitan el desarrollo de Laravel profesionalmente.	Gratuito
Valet	Permite que se creen entornos de ejecución de Laravel, ligeros y fáciles de configurar en ordenadores tipo Mac.	Gratuito
Cashier	Permite pagos electrónicos y suscripciones.	Gratuito

Mix	Es una envoltura de Webpack para compilar assets y realizar tareas del front-end sin las necesidades de administrar el entorno directo desde Webpack.	Gratuito
Echo	Permite trabajar con datos en tiempo real.	Gratuito
Horizon	Sirve para la monitorización de colas de procesos.	Gratuito

Fuente: Rojas Parra, 2019 y Campus MVP, 2015
Elaboración: Mariano Navarrete

Si bien es cierto, los primeros servicios y herramientas tienen un costo, pero el mismo sirve para financiar la maquinaria de Laravel, no obstante, los demás servicios y herramientas son gratuitos o están incluidos entre las funcionalidades del framework (Desarrollo Web, 2020).

Laravel cuenta con innumerables ventajas para que los desarrolladores se decanten por este framework, pues al tener una sintaxis elegante, innovadora y creativa, le permite desatacar entre otros frameworks disponibles en leguaje tipo PHP, ya que este lenguaje de programación suele ser dificultoso, sin embargo, con Laravel es sencillo de entender, de ahí la importancia de su estudio y uso (Laravel, s.f.:4).

3.2.2 Ruby on Rails

Fue creado por David Heinemeier Hansson en 2003 (Rubyonrails, 2020; Uniwebsidad, 2020).

Ruby on Rails o llamado también Rails o RoR, es un conjunto de librerías, convenciones y automatismos enfocados en resolver los principales problemas al momento de implementar una página web, pues permite escribir el código de una manera clara evitando así el llamado código espagueti (Rubyonrails, 2020).

Ruby fue desarrollado con Rails y sigue el modelo de Programación Orientado a Objetos [POO], para desarrollar aplicaciones en el lado del servido, Rails combina Ruby con los lenguajes HTML, CSS y JavaSript, entre sus principales usuarios destacan Twitter, Scrib, Github, Walmart, CISCO, IBM, JP Morgan, NASA y Yahoo (CLE, 2017; Rubyonrails, 2020: Garza, 2017; Arias, 2019).

La arquitectura de Rails sigue el MVC, de modo que, la vista incluye plantillas responsables para insertar datos entre etiquetas tipo HTML; el modelo contiene objetos inteligentes de dominio que engloban la lógica del negocio y, el controlador gestiona las peticiones que ingresan manipulando el modelo y redirigiendo a la vista, garantizando de esta manera una independencia de los diferentes dominios (De Ipiña, 2006). Rails a pesar de que por defecto soporta la biblioteca SQLite, también permite trabajar con bibliotecas como MySQL, PostgreSQL y Oracle (Ortega Delgado, 2017; Rubyonrails, 2020).

Rails se rige mediante tres principios: dogmatismo, Convention Over Configuration y el conocido Don´t Repeat Yourself [DRY], los cuales se detallan a continuación:

- Dogmatismo: Rails asume que siempre existe una ´mejor forma´ para hacer las cosas y, en algunos casos para desanimar alternativas con ello el desarrollador aprovecha de mejor manera su tiempo (Uniwebsidad, 2020).
- Convention Over Configuration: Rails evita una multiple cantidad de configuraciones en archivos pues asume cosas, es decir, que realiza suposiciones sobre lo que el desarrollador quiere hacer y cómo lo hará, evitando así una pérdida de tiempo en otras tareas (Rubyonrails, 2020; Arias, 2019; CLE, 2019).
- DRY: Es un principio de desarrollo de software y evita los duplicados, los cuales hacen que el código sea complejo, vulnerable a errores y difícil de mantener, además DRY elimina códigos donde sea posible para evitar el código espagueti, logrando de esta manera tener una aplicación en el aldo del servidor sin muchas lineas de código(Arias, 2019; Ortega Delgado, 2017).

A pesar de que Rails ha aprovechado la maleabilidad del lenguaje de programación Ruby, es heredero de la bondad y debilidad de este lenguaje, esto es, que es mucho más lento que otros

frameworks. Si bien es cierto, la velocidad no sería un problema de peso, pero se convierte en uno cuando se almacena información y se requiere alt disponibilidad pues en ese momento se pierde velocidad, por lo cual no es adecuado para trabajar con procesos largos (Guerrero Benalcázar, 2016:20).

El uso de este framework dependerá de las necesidades del desarrollador basadas en las características de la página o aplicación web que desee realizar.

3.2.3 Node.JS

Node.JS usa el modelo de entrada y salida sin bloqueo con el objetivo de ser liviano y eficiente (Lucas, 2019).

Node es muy bueno para desarrollar aplicaciones en tiempo real, las mismas que necesitan mantener una conexión durable entre el browser y el servidor (como por ejemplo juegos en línea, chats, herramientas de colaboración, entre otros (Guerrero, 2018; Lucas, 2019).

La escalabilidad y alta velocidad de ejecución mediante JavaScript no son las únicas ventajas de Node, pues este framework presenta una programación asíncrona como otra de sus ventajas, dado que Node dispone del Bucle de Eventos que le permite manejar grandes cantidades de clientes de forma asíncrona (Cavana, 2017).

No obstante, Node no se implementa fácilmente en alojamientos existentes, esto quiere decir que se debe tener muy en claro la infraestructura en la que se va a montar, pues tiene nuevo versionamiento, por ende el desarrollador deberá aplicar cambios frecuentes en el código para que se ajuste a la versión actual; no cuenta con una librería estándar, esto es herencia de JavaScript quien tiene poca librería estándar; Node no tiene ventaja alguna si se desea realizar un trabajo enfocado en CPU, como codificación de video o manipulación de imagen; además Node tiene un reconocimiento (footprint) de memoria menor, los procesos de Node ocupan

niveles de memoria menores que los otros lenguajes (Cangás, 2015; Guerrero, 2018; Morales, 2012).

3.2.4 Selección del framework back-end

Tras la revisión teórica mostrada se procede a realizar la selección del framework back-end que se adapte a las necesidades y características del caso de estudio, de la manera siguiente:

Tabla N 5 Caracterización y selección del framework back-end para el caso de estudio

	Laravel	Ruby on Rails	NodeJs
Características	Open Source	Open Source	Open Source
	Framework basado en PHP	Framework basado en Ruby	Framework basado en JavaScript
	Excelente rendimiento en proyectos pequeños o grandes.	Excelente rendimiento en proyectos pequeños o grandes.	Excelente rendimiento en proyectos pequeños. No recomendable para proyectos grandes.
	Curva de aprendizaje baja	Curva de aprendizaje media	Curva de aprendizaje baja
	Altamente estructurado	Altamente estructurado	No estructurado
	Permite trabajar con múltiples conexiones	No permite trabajar con múltiples conexiones	Permite manejar múltiples conexiones
	Altamente documentado	Medianamente documentado	Altamente documentado

Fuente: Stackshare, 2020
Elaboración: Mariano Navarrete

La elección de un framework para desarrollar un determinado proyecto depende del objetivo que éste pretende cumplir, y como será utilizado por sus usuarios. Laravel y Node.Js son frameworks que han estado creciendo en su comunidad, puesto que Laravel utiliza PHP y Node.JS utiliza JavaScript, los dos son lenguajes altamente populares en el desarrollo web.

El framework escogido es Laravel, por su gran robustez a la hora de desarrollar proyectos medianos, la escalabilidad que brinda este framework y la posibilidad de manejar cientos de conexiones a la vez. Igualmente, gracias a que Laravel utiliza PHP puede manejar fotos y videos de una manera muy eficiente. Estas características hacen que Laravel sea el mejor framework que se acople al aplicativo web.

Por consiguiente, el framework para desarrollar el sistema web es Laravel, mientras que, para el desarrollo del prototipo de página web se empleará React. Luego de ello se aplicará la metodología Scrum.

4. CAPÍTULO 5: APLICACIÓN DE UNA METODOLOGÍA QUE FACILITE Y ASEGURE EL CORRECTO DESARROLLO DEL SISTEMA WEB

4.1 ANÁLISIS DE REQUERIMIENTOS

En esta sección se obtendrá las funcionalidades con las que estará compuesto el sistema. Para ello, es necesario realizar una reunión con el padre Párroco Javier Piarpuzán, con el objetivo de que exponga sus necesidades y posteriormente poder planificar el desarrollo del sistema mediante la metodología Scrum.

En la planificación se establecerá el tiempo estimado para desarrollar cada funcionalidad con su respectiva importancia.

4.2 HISTORIAS DE USUARIO

Son pequeñas frases que muestran un requerimiento específico del sistema. Culminada la reunión con el padre Párroco Javier Piarpuzán, se llegó al acuerdo de tener dos perfiles que interactúen con el sistema: usuario y administrador. Asimismo, se identificó 24 historias de usuario las cuáles son detalladas a continuación.

4.2.1 Como administrador quiero ingresar al sistema para mantener una seguridad y seguimiento de las actividades realizadas dentro del mismo (Piarpuzán, entrevista personal, 22 de febrero de 2020).

4.2.2 Como administrador quiero ingresar las instalaciones del Santuario para que sean asignadas a los cursos y servicios (Piarpuzán, entrevista personal, 22 de febrero de 2020).

4.2.3 Como administrador quiero modificar las instalaciones del Santuario para corregir la información ingresada (Piarpuzán, entrevista personal, 22 de febrero de 2020).

4.2.4 Como administrador quiero visualizar todas las instalaciones del Santuario para tener una idea de las instalaciones presentes en el sistema (Piarpuzán, entrevista personal, 22 de febrero de 2020).

4.2.5 Como administrador quiero buscar una instalación del Santuario en específico para modificarla o eliminarla (Piarpuzán, entrevista personal, 22 de febrero de 2020).

4.2.6 Como administrador quiero ingresar los sacerdotes que están designados al Santuario de El Quinche para llevar un control de los mismos (Piarpuzán, entrevista personal, 22 de febrero de 2020).

4.2.7 Como administrador quiero borrar los sacerdotes que terminan su periodo en el Santuario de El Quinche, para ingresar a los nuevos sacerdotes y modificar la información en que los antiguos sacerdotes participaban (Piarpuzán, entrevista personal, 22 de febrero de 2020).

4.2.8 Como administrador quiero ingresar los servicios que brinda el Santuario para darlos a conocer la ciudadanía (Piarpuzán, entrevista personal, 22 de febrero de 2020).

4.2.9 Como administrador quiero visualizar los servicios que están ingresado en el sistema para poder eliminarlos o modificarlos (Piarpuzán, entrevista personal, 22 de febrero de 2020).

4.2.10 Como administrador quiero asignar la instalación en donde se brinda el servicio (Piarpuzán, entrevista personal, 22 de febrero de 2020).

4.2.11 Como administrador quiero asignar los diferentes horarios que tiene cada servicio para llevar un control de los servicios ofertados por días (Piarpuzán, entrevista personal, 22 de febrero de 2020).

4.2.12 Como administrador quiero visualizar los diferentes horarios que tiene el servicio para poder agregar, modificar o eliminar dichos horarios según el caso lo amerite (Piarpuzán, entrevista personal, 22 de febrero de 2020).

4.2.13 Como administrador quiero ingresar los cursos que son ofertados por el Santuario para dar a conocer su información (Piarpuzán, entrevista personal, 22 de febrero de 2020).

4.2.14 Como administrador quiero asignar la instalación en la que se va a dictar el curso (Piarpuzán, entrevista personal, 22 de febrero de 2020).

4.2.15 Como administrador quiero agregar los diferentes horarios en los que se dictan los cursos (Piarpuzán, entrevista personal, 22 de febrero de 2020).

4.2.16 Como administrador quiero visualizar los cursos para poder modificarlos o eliminarlos (Piarpuzán, entrevista personal, 22 de febrero de 2020).

4.2.17 Como administrador quiero visualizar los distintos horarios de cada curso para poder agregarlos, modificarlos o eliminarlos (Piarpuzán, entrevista personal, 22 de febrero de 2020).

4.2.18 Como administrador quiero ingresar los sacramentos que se realizan en el Santuario para brindar la información de los mismos (Piarpuzán, entrevista personal, 22 de febrero de 2020).

4.2.19 Como administrador quiero asignar el curso necesario para poder realizar el sacramento (Piarpuzán, entrevista personal, 22 de febrero de 2020).

4.2.20 Como administrador quiero visualizar los sacramentos, para poder eliminarlos o modificarlos (Piarpuzán, entrevista personal, 22 de febrero de 2020).

4.2.21 Como usuario quiero una barra de navegación para desplazarme fácilmente por el sitio web (Piarpuzán, entrevista personal, 22 de febrero de 2020).

4.2.22 Como usuario quiero visualizar todos los cursos, servicios o sacramentos para verificar lo que necesito (Piarpuzán, entrevista personal, 22 de febrero de 2020).

4.2.23 Como usuario quiero ver un curso, servicio o sacramento en particular para obtener la información necesaria (Piarpuzán, entrevista personal, 22 de febrero de 2020).

4.2.24 Como usuario quiero ver un mapa donde se visualice un punto que haga referencia a la instalación o lugar donde se ofertan los cursos, servicios o sacramentos para poder tener una ubicación exacta (Piarpuzán, entrevista personal, 22 de febrero de 2020).

Luego de haber identificado las historias de usuario se documenta la información mediante la siguiente tabla.

Los campos que se van a utilizar para la tabla serán los siguientes:

- Número: identificación única de cada historia de usuario, se aconseja que se utilice la numeración incremental de 1.
- Identificación: nombre que se le asigna a la historia de cliente.
- Descripción: referencia a la historia dictada por el usuario.

- Story point: referencia al esfuerzo y tiempo necesario para cumplir con cada historia de usuario. Dicha referencia está dada por números entre 1 y 5, siendo 1 lo más bajo y 5 lo más alto.
- Rol: persona previamente identificada que va a utilizar el aplicativo web.
- Prioridad: nivel de importancia definida por el propietario del producto, este nivel esta dado por números entre 1 y 5, siendo 1 baja prioridad y 5 alta prioridad.

Tabla N 6 Historias de Usuario

Número	Identificación	Descripción	Story Point	Rol	Prioridad
1	Autenticación al sistema	Como administrador quiero ingresar al sistema para mantener una seguridad y seguimiento de las actividades realizadas dentro del mismo.	4	Administrador	5
2	Ingreso de instalaciones	Como administrador quiero ingresar las instalaciones del Santuario para que sean asignadas a los cursos y servicios.	2	Administrador	4
3	Modificación de instalaciones	Como administrador quiero modificar las instalaciones del Santuario para corregir la información ingresada.	3	Administrador	3
4	Visualización de instalaciones	Como administrador quiero visualizar todas las instalaciones del Santuario para tener una idea de las instalaciones presentes en el sistema.	2	Administrador	3
5	Búsqueda por parámetro de las instalaciones	Como administrador quiero buscar una instalación del Santuario en específico para modificarla o eliminarla.	2	Administrador	2
6	Ingreso de los sacerdotes	Como administrador quiero ingresar los sacerdotes que están designados al Santuario de El Quinche para llevar un control de los mismos.	2	Administrador	5
7	Eliminación de Sacerdotes	Como administrador quiero borrar los sacerdotes que terminan su periodo en el Santuario de El Quinche, para ingresar a los nuevos sacerdotes y modificar la información en que los antiguos sacerdotes participaban.	3	Administrador	1
8	Ingreso de servicios	Como administrador quiero ingresar los servicios que brinda el Santuario para darlos a conocer la ciudadanía.	2	Administrador	3
9	Visualizar servicios	Como administrador quiero visualizar los servicios que están ingresado en el sistema para poder eliminarlos o modificarlos.	2	Administrador	2

Número	Identificación	Descripción	Story Point	Rol	Prioridad
10	Asignar instalación	Como administrador quiero asignar la instalación en donde se brinda el servicio.	1	Administrador	3
11	Asignar horarios al servicio	Como administrador quiero asignar los diferentes horarios que tiene cada servicio para llevar un control de los servicios ofertados por días.	3	Administrador	3
12	Visualizar los horarios del servicio	Como administrador quiero visualizar los diferentes horarios que tiene el servicio para poder agregar, modificar o eliminar dichos horarios según el caso lo amerite.	2	Administrador	3
13	Ingresar cursos	Como administrador quiero ingresar los cursos que son ofertados por el Santuario para dar a conocer su información.	2	Administrador	4
14	Asignar instalación al curso	Como administrador quiero asignar la instalación en la que se va a dictar el curso.	1	Administrador	2
15	Agregar horarios al curso	Como administrador quiero agregar los diferentes horarios en los que se dictan los cursos.	3	Administrador	3
16	Visualizar los cursos	Como administrador quiero visualizar los cursos para poder modificarlos o eliminarlos.	2	Administrador	2
17	Visualizar los horarios del curso	Como administrador quiero visualizar los distintos horarios de cada curso para poder agregarlos, modificarlos o eliminarlos.	2	Administrador	2
18	Ingresar sacramentos	Como administrador quiero ingresar los sacramentos que se realizan en el Santuario para brindar la información de los mismos.	2	Administrador	5
19	Asignar un curso al sacramento	Como administrador quiero asignar el curso necesario para poder realizar el sacramento.	1	Administrador	3
20	Visualizar sacramentos	Como administrador quiero visualizar los sacramentos, para poder eliminarlos o modificarlos.	2	Administrador	3

Número	Identificación	Descripción	Story Point	Rol	Prioridad
21	Navbar en la página web	Como usuario quiero una barra de navegación para desplazarme más fácilmente por el sitio web.	3	Usuario	4
22	Página informativa en general	Como usuario quiero visualizar todos los cursos, servicios o sacramentos para verificar lo que necesito.	3	Usuario	4
23	Página informativa en específico	Como usuario quiero ver un curso, servicio o sacramento en particular para obtener la información necesaria.	3	Usuario	4
24	Mapa de ubicación	Como usuario quiero ver un mapa donde se visualice un punto que haga referencia a la instalación o lugar donde se ofertan los cursos, servicios o sacramentos para poder tener una ubicación exacta.	5	Usuario	5

Fuente: Entrevista personal Padre Javier Piarpuzán, 2020
Elaboración: Mariano Navarrete

4.3 PRODUCT BACKLOG

Prioriza las tareas a realizar, estas tareas son los requisitos y expectativas que tiene el cliente con el sistema. Dicha lista es construida a partir de las historias de usuario con la principal característica de que deben ser comprensibles entre las partes involucradas.

El producto backlog será construido mediante una tabla que contendrá los siguientes campos:

- Identificación: número que identifica cada tarea a realizar, este número es único para cada tarea.
- Descripción: interpretación de las funcionalidades a partir de las historias de usuario.
- Story point: referencia al esfuerzo y tiempo necesario para cumplir con cada funcionalidad de usuario. Dicha referencia está dada por números entre 1 y 5, siendo 1 lo más bajo y 5 lo más alto.
- Rol: persona que va a interactuar con el sistema
- Prioridad: nivel de importancia definida por el propietario del producto, este nivel esta dado por números entre 1 y 5, siendo 1 baja prioridad y 5 alta prioridad.
- Sprint: número del sprint en que se desarrollará la funcionalidad.
- Tiempo: estimación del tiempo en horas para desarrollar la funcionalidad.
- Criterio de aceptación: reglas por las cuáles la funcionalidad se deberá basar para su desarrollo y para aceptar que se cumplió con la funcionalidad acordada.

Tabla N 7 Product Backlog

Número	Descripción	Story Point	Rol	Prioridad	Sprint	Tiempo en horas	Criterio de Aceptación
1	Elaborar una pantalla para la autenticación del sistema	4	Administrador	5	1	6	El sistema debe guardar debe validar las credenciales y guardar los logs que realice el administrador.
2	Elaborar una pantalla que muestre los distintos módulos del sistema	2	Administrador	5	1	2	La pantalla debe indicar todos los módulos del sistema y un submenú para visualizar los módulos o ingresar una nueva entrada
3	Elaborar una pantalla para ingresar una instalación	3	Administrador	5	1	4	En la pantalla se debe ingresar toda la información necesaria y debe permitir ingresar una o más fotos asociadas a la instalación
4	Listarlas instalaciones con el nombre de las mismas en una tabla	2	Administrador	5	1	3	Se debe visualizar en la pantalla los nombres de las instalaciones.
5	Agregar un botón en la lista de las instalaciones para poder buscar una instalación en particular.	2	Administrador	5	1	4	En la pantalla donde se visualice las instalaciones debe haber un botón para buscar la instalación y debe mostrar la instalación obtenida de la búsqueda en una nueva pantalla.
6	Elaborar una pantalla para la modificación de la información de las instalaciones	3	Administrador	5	1	5	La pantalla de edición de datos de las instalaciones debe mostrar los campos cargados con la información ya ingresada.
7	Elaborar una pantalla para indicar si esta seguro de eliminar una instalación	4	Administrador	5	1	1	La pantalla debe indicar el mensaje de eliminar la instalación, si se elimina una instalación debe indicarse que se deje de utilizarse en los módulos que están obtenido su información y eliminar todas las fotos asociadas al mismo.

Número	Descripción	Story Point	Rol	Prioridad	Sprint	Tiempo en horas	Criterio de Aceptación
8	Listar los cursos y/o servicios que están ocupando una instalación que se necesite dar de alta	3	Administrador	5	1	3	Se debe listar en una pantalla los servicios o cursos con un hipervínculo que los lleve directamente a editar dicho curso o servicio.
9	Elaborar una pantalla para ingresar los datos de un sacerdote que ingrese al Santuario	2	Administrador	5	1	4	La pantalla debe mostrar todos los campos necesarios para ingresar un nuevo sacerdote, al finalizar el ingreso debe redirigirse a la misma pantalla para permitir un nuevo ingreso. Debe tener un botón de finalización para dirigirse al menú principal.
10	Elaborar una tabla para listar a los sacerdotes.	2	Administrador	3	1	3	La tabla debe mostrar a todos los sacerdotes. Debe tener dos botones, uno de eliminar y el otro de editar la información del sacerdote seleccionado.
11	Elaborar una pantalla modificar la información del sacerdote.	2	Administrador	3	2	5	La pantalla debe ser cargada con la información obtenida del sacerdote.
12	Elaborar una pantalla para la confirmación de la eliminación de un sacerdote.	2	Administrador	3	2	2	La pantalla debe indicar si está seguro de eliminar la información, en caso de que el sacerdote este a cargo de cursos y/o servicios se debe redirigir a una nueva pantalla.
13	Listar los cursos y/o servicios que están a cargo del sacerdote que se desee eliminar.	3	Administrador	3	2	4	Debe mostrar todos los cursos y/o servicios que el sacerdote este a cargo con un hipervínculo para su edición.

Número	Descripción	Story Point	Rol	Prioridad	Sprint	Tiempo en horas	Criterio de Aceptación
14	Elaborar una pantalla para el ingreso de un servicio.	3	Administrador	5	2	4	La pantalla se debe visualizar todos los campos necesarios y debe existir la forma de seleccionar una instalación.
15	Elaborar una pantalla de ingreso de información extra.	3	Administrador	5	2	3	Debe tener dos botones uno de ingreso de un horario y el otro para finalizar el ingreso del servicio.
16	Elaborar un modal para el ingreso de horarios al servicio.	2	Administrador	2	2	3	Debe permitir el ingreso de los horarios necesarios para el servicio.
17	Listar los servicios en una tabla.	2	Administrador	3	2	3	Debe listar los servicios con la posibilidad de editarlos o eliminarlos.
18	Elaborar una pantalla para la edición de un servicio.	3	Administrador	3	2	5	La pantalla debe tener la información del servicio, un botón para editar los horarios y un botón para finalizar la edición.
19	Listar los horarios de un servicio en una tabla.	3	Administrador	3	2	3	La tabla debe indicar todos los horarios asociados al servicio y debe tener los botones necesarios para la eliminación o edición del horario, de igual manera debe contar con un botón para la finalización de la edición del servicio.
20	Crear un componente para la búsqueda de un servicio en particular.	2	Administrador	2	2	4	El componente debe permitir buscar un servicio por su nombre.
21	Elaborar una pantalla para la eliminación del servicio.	2	Administrador	3	2	4	Debe indicar si está seguro de eliminar el servicio y deben eliminarse todos los horarios asociados al servicio

Número	Descripción	Story Point	Rol	Prioridad	Sprint	Tiempo en horas	Criterio de Aceptación
22	Elaborar una pantalla para el ingreso de un curso.	2	Administrador	5	2	4	La pantalla debe permitir el ingreso de todos los datos necesario y debe incluir una manera de ingresar el sacerdote a cargo como la instalación.
23	Elaborar una pantalla para ingresar los horarios del curso.	2	Administrador	3	3	3	La pantalla debe permitir ingresar todos los horarios que sean necesarios para el curo y debe tener un botón para finalizar el ingreso.
24	Elaborar un modal para el ingreso de los horarios.	2	Administrador	3	3	3	El modal debe tener la información necesaria para la creación del horario.
25	Listar todos los cursos en una tabla.	2	Administrador	3	3	3	La tabla debe contener botones para eliminación, o la edición de un curso.
26	Crear un componente para la búsqueda en particular de un curso.	3	Administrador	2	3	5	El componente debe permitir buscar a un curso por su nombre.
27	Elaborar una pantalla para la eliminación del curso.	3	Administrador	3	3	5	La pantalla debe indicar si está seguro de eliminar el curso y debe eliminar a todos los horarios que están asociados al curso.
28	Elaborar una pantalla para el ingreso de un sacramento.	2	Administrador	5	3	5	La pantalla debe permitir el ingreso de toda la información del sacramento y debe permitir seleccionar el curso necesario para el sacramento.
29	Elaborar una pantalla para el ingreso de horarios.	2	Administrador	4	3	3	La pantalla debe permitir el ingreso de todos los horarios necesarios y debe tener un botón para la finalización.
30	Elaborar un modal para el ingreso de los horarios.	2	Administrador	4	3	3	El modal debe permitir toda la información del horario.

Número	Descripción	Story Point	Rol	Prioridad	Sprint	Tiempo en horas	Criterio de Aceptación
31	Listar los sacramentos ingresados en una tabla	2	Administrador	3	3	3	Debe permitir mostrar todos los sacramentos con botones para la edición o eliminación.
32	Elaborar una pantalla para la edición de un sacramento.	3	Administrador	3	3	5	La pantalla debe tener la información cargada del sacramento y debe tener un botón para la edición de los horarios
33	Listar los horarios del sacramento.	2	Administrador	3	3	4	Se debe listar todos los horarios asociados al sacramento para poder editarlos o eliminarlos.
34	Elaborar una pantalla para la eliminación de un sacramento.	3	Administrador	3	3	4	Se debe indicar si está seguro de eliminar el sacramento y se debe eliminar todos los horarios asociados al sacramento.
35	Elaborar un componente para buscar el sacramento	3	Administrador	2	3	4	El componente debe permitir buscar un sacramento por su tipo.
36	Elaborar una pantalla que muestre la información general de las actividades del Santuario.	3	Usuario	2	4	6	La pantalla debe tener una barra de navegación que contenga todas las categorías que ofrece el santuario.
37	Elaborar una pantalla que muestre una categoría en específico.	4	Usuario	2	4	6	La pantalla debe contener todos los cursos, servicios o sacramentos ofertados.
38	Elaborar una pantalla que muestre un curso, servicio o sacramento.	4	Usuario	2	4	6	La pantalla debe mostrar toda la información de un curso, servicio o sacramento.
39	Crear un componente para indicar la geolocalización de las instalaciones.	5	Usuario	2	4	9	En el componente se debe visualizar un mapa con el punto de la ubicación real de la instalación a la que se haga referencia.

Elaboración: Mariano Navarrete

Al haber terminado de documentar cada uno de los requerimientos obtenidos de las historias de usuario, se obtuvo un tiempo estimado de 156 horas. El tiempo estimado se dividirá en 4 sprints, los cuales se realizarán dependiendo del nivel de prioridad de cada requerimiento, siendo 5 el que tiene mayor prioridad y 2 el que tiene menor prioridad.

4.4 DESCRIPCIÓN DEL SISTEMA

Tras haber obtenido toda la información brindada por el Padre Párroco Javier Piarpuzán acerca de las funcionalidades que debe cumplir el sistema, se han definido 2 roles que se utilizarán en el sistema web, los cuáles son detallados a continuación.

4.4.1 Administrador

El rol de administrador es la figura que administra todo el sistema, es decir, es la persona que puede ingresar nueva información, modificarla o eliminarla según sea necesario. Esta persona es la única que puede autenticarse y mantener un control de sus logs. Además, esta persona puede administrar la información que se quiera indicar al público.

4.4.2 Usuario

El rol de usuario es la figura que puede consultar toda la información brindada por la aplicación web. No tiene necesidad de identificarse para tener acceso a la información, esto por las políticas con las que se maneja el Santuario. Esta persona solo es capaz de consumir la información, por lo tanto, no puede alterarla.

4.4.3 Funcionalidades generales del sistema

Una vez finalizado el producto, el Santuario contará con un sistema con el que pueda gestionar toda su información de las características más importante que se ofertan en el mismo. De igual

manera, se desarrollará un prototipo de página web parala población devota a la Virgen de El Quinche, que le permita informarse de los distintos cursos, servicios o sacramentos que se realizan en este Santuario.

Para el uso del sistema web se deberá contar con un dispositivo electrónico con conexión a internet, preferentemente un ordenador de escritorio o un portátil. Por otra parte, los devotos de la Virgen de El Quinche podrán acceder al prototipo de página web desde cualquier dispositivo electrónico con acceso a internet, esto gracias a su diseño responsivo. Todas las características estarán disponibles, así como una de las más importantes como la geolocalización.

La información que maneja el Santuario estará centralizada y controlada. Cualquier información con respecto a las instalaciones, sacerdotes, cursos, servicios y sacramentos estarán contralas y actualizadas para brindar una información veraz a las personas que las necesiten.

Los devotos podrán ubicarse de una manera exacta cuando visiten el Santuario o quieran dirigirse a una instalación que el pertenezca al Santuario o podrán consultar algún tipo de servicio en particular en el que deseen participar.

4.4.4 Funcionalidades aplicadas al administrador

El sistema brindará una página principal donde se encuentren todas las actividades que pueda realizar.

4.4.4.1 Sacerdotes

En este módulo el administrador podrá agregar a un nuevo sacerdote o visualizar todos los sacerdotes ingresados. La lista de los sacerdotes estará disponible en una tabla que brindará el nombre completo y la cédula. Podrá eliminar o modificar la información del sacerdote que

seleccione. Además, podrá buscar un Sacerdote por su nombre y presentarle una información completa de lo que se desea buscar, permitiéndole eliminar o modificar la información.

4.4.4.2 Instalaciones

En esta funcionalidad el administrador podrá agregar una nueva instalación o visualizar todas las instalaciones que estén ingresadas. Se podrá ingresar diferentes fotografías sin un número límite, con el objeto de tener un completo catálogo de ellas para que puedan ser mostradas. Se podrá eliminar o modificar las instalaciones desde la lista donde se visualice estas instalaciones.

4.4.4.3 Servicios

En este módulo el administrador podrá ingresar los servicios que ofrece el Santuario, igualmente, podrá visualizar todos los servicios ingresados mediante una lista ubicada en una tabla. Podrá seleccionar la instalación en la que se brinda el servicio, así como ingresar los horarios y días en los que está disponible este servicio.

En la lista de visualización podrá eliminar todos los servicios, eliminando consigo todos sus horarios. En el apartado de modificación podrá modificar toda la información que requiera y podrá visualizar todos los horarios que están enlazados para poder modificarlos o eliminarlos.

4.4.4.4 Cursos

En este módulo el administrador podrá ingresar todos los cursos que se oferten, así mismo, podrá ingresar los cursos que son requisito para poder realizar un sacramento. De igual forma, podrá ingresar todos los horarios y días disponibles del curso, así como su fecha de inicio y finalización. Podrá visualizar todos los cursos mediante una tabla que le permitirá modificarlos o eliminarlos. Cuando se requiera modificar un curso podrá modificarse toda su información,

así como sus horarios que serán presentados en una lista. Si se desea eliminar el curso, éste presentará una lista de los sacramentos que están utilizando dicho curso.

4.4.4.5 Sacramentos

En este módulo el administrador podrá ingresar un sacramento seleccionando el tipo de sacramento que desea y agregar su curso necesario. Mostrará una lista de los sacramentos ingresados para poder editarlos y eliminarlos. Podrá ingresar todos los horarios en que el sacramento está disponible en forma permanente, puesto que existe la posibilidad de realizar un sacramento en una fecha escogida por el devoto, pero necesita una previa cita con el Padre Párroco. De igual manera, podrá eliminar o editar los horarios ingresados indicados en una lista.

4.4.5 Funcionalidades aplicadas al usuario

El usuario tendrá una barra de navegación que estará presente y fija en la parte superior en cualquier módulo que requiera. Estas funcionalidades estarán ubicadas en la página web dedica a los usuarios.

4.4.5.1 Cursos

En esta funcionalidad se podrá visualizar todos los cursos que están presentes en el sistema web. De igual manera, podrá consultar un curso en particular o buscar en la lista que se presentará. Cuando se seleccione o busque ´curso´ se desplegará una nueva página con la información del curso, así como un mapa con la dirección exacta de la instalación.

4.4.5.2 Servicios

En esta funcionalidad el usuario podrá observar todos los servicios que están ingresados en la base de datos. Podrá buscar por el nombre del servicio o podrá identificar el servicio que

requiere en la lista que se presenta. Cuando tenga un servicio al que se quiera consultar, se le desplegará una nueva página con la información del servicio, la cual contendrá un mapa en la parte inferior con la ubicación exacta en donde se brinda.

4.4.5.3 Sacramentos

En esta funcionalidad el usuario podrá verificar todos los sacramentos que están siendo celebrados en el Santuario. La forma en que podrá buscar el sacramento deseado será mediante la lista que se muestre en la sección de Sacramentos.

Se desplegará la información necesaria para poder acceder al sacramento y el curso necesario, éste podrá ser visto a profundidad en su página principal gracias al hipervínculo que se presentará, igualmente, al final de la interfaz se mostrará el mapa de la ubicación exacta en donde se encuentra la instalación en la que se celebra el Sacramento.

Finalizada la aplicación de la metodología Scrum para facilitar y asegurar el correcto desarrollo del sistema web, se procede a diseñar la base de datos, que constituye el elemento fundamental del siguiente capítulo.

5. CAPÍTULO 6: DISEÑO DE LA BASE DE DATOS, DESARROLLO DEL SISTEMA WEB FRONT-END Y BACK-END E IMPLEMENTACIÓN DE LA GEOLOCALIZACIÓN DE LOS SERVICIOS DEL SANTUARIO DE NUESTRA SEÑORA DE LA PRESENTACIÓN DE EL QUINCHE

El desarrollo del presente capítulo abarca el diseño de la base de datos que será empleada en el sistema, especificando las entidades y sus atributos para generar al final un diagrama entidad relación para una mayor comprensión, asimismo, se desarrolla el sistema web para la gestión de la información del Santuario. El desarrollo utilizará la metodología Scrum, la cual está compuesta por 4 sprints con una duración aproximada de 156 horas. La forma de desarrollo será según la prioridad que se definió y de manera incremental, siendo los módulos independientes los primeros en desarrollarse y luego se desarrollarán los módulos independientes.

5.1 DISEÑO DE LA BASE DE DATOS

5.1.1 Entidades

Para representar los objetos que se pretenden modelar son necesarias las entidades, las cuáles deben tener su propia distinción. Los atributos de cada entidad fueron seleccionados por las necesidades del usuario.

A continuación, se presentan las entidades identificadas.

5.1.1.1 Instalaciones

La entidad instalaciones está definida para almacenar la información que posee una instalación propia del Santuario. Los atributos identificados son los siguientes:

- Nombre
- Dirección
- Latitud
- Longitud

5.1.1.2 Cursos

La entidad cursos está definida para almacenar la información que posee un curso que es ofertado por el Santuario. Los atributos identificados son los siguientes:

- Nombre
- Descripción
- Imagen
- Fecha de inicio
- Fecha de finalización
- Cupos disponibles

5.1.1.3 Horarios

La entidad horarios está definida para almacenar la información que posee el horario de un curso o sacramento en particular. Los atributos identificados son los siguientes:

- Hora de inicio
- Hora de finalización
- Días hábiles

5.1.1.4 Sacerdotes

La entidad sacerdotes está definida para almacenar la información que posee un sacerdote que está cumpliendo su labor en el Santuario. Los atributos identificados son los siguientes:

- Cédula
- Nacionalidad
- Cargo
- Nombre
- Apellido
- Fecha de nacimiento
- Activo
- Imagen

5.1.1.5 Sacramentos

La entidad sacramentos está definida para almacenar la información que posee un sacramento que es ofertado en el Santuario. Los atributos identificados son los siguientes:

- Tipo
- Precio
- Foto
- Requisitos

5.1.1.6 Hora de Atención

La entidad horaatencion está definida para almacenar la información de cada una de las horas de atención que dispone el Santuario para un determinado sacramento. Los atributos identificados son los siguientes:

- Hora
- Días hábiles

5.1.1.7 Servicios

La entidad servicios está definida para almacenar la información que posee un servicio que brinda el Santuario. Los atributos identificados son los siguientes:

- Nombre
- Descripción
- Imagen

5.1.1.8 Fotos

La entidad fotos está definida para almacenar las distintas fotografías que posee una instalación del Santuario. Los atributos identificados son los siguientes:

- Imagen

5.1.1.9 Administrador

La entidad administrador está definida para almacenar la información de inicio de sesión del administrador.

- Nombre
- Email
- Contraseña

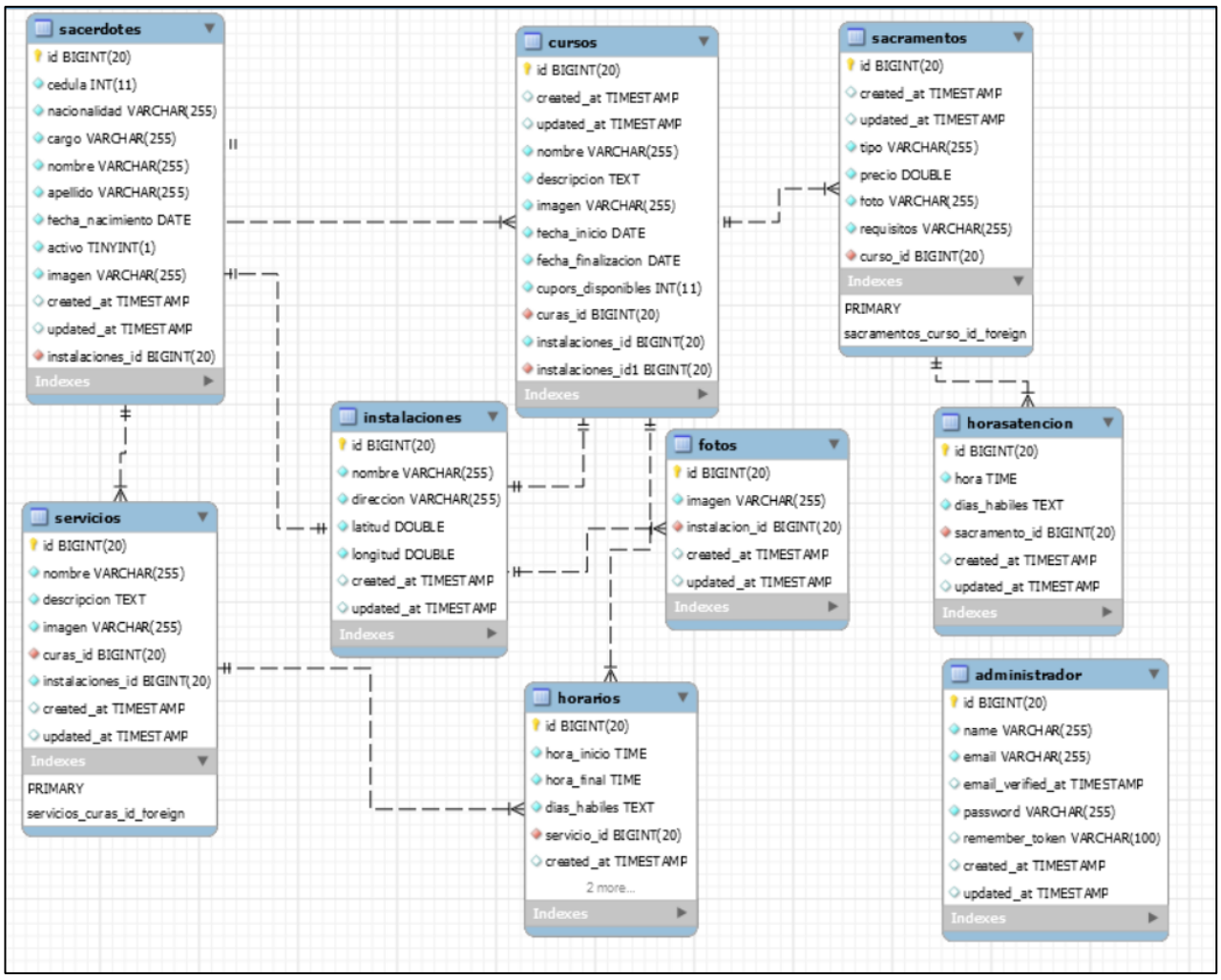
5.1.2 Diagrama entidad relación

A continuación, se muestra el diagrama entidad relación a construirse. Gracias a este diagrama se puede tener una mejor visualización de las entidades y las relaciones con las demás entidades.

Así mismo se puede observar el tipo de dato que posee cada atributo con su respectiva longitud.

Para desarrollar este diagrama se utilizó la herramienta MySQL Workbench.

Gráfico N 6 Diagrama entidad relación



Elaboración: Mariano Navarrete

5.2 SCRUM

5.2.1 Especificación de roles

Se presenta una tabla especificando los roles y quien será el responsable de cumplir cada rol. En scrum se debe definir los roles para construir un software de calidad y facilitar su desarrollo.

Tabla N 8 Roles del equipo Scrum

Rol	Responsable
Product Owner	Mariano Navarrete
Scrum Master	Mariano Navarrete
Development Team	Mariano Navarrete
Cliente	P. Javier Piarpuzan

Elaboración: Mariano Navarrete

5.2.2 Sprint 1

5.2.2.1 Planificación del sprint

Aquí se desarrollará la funcionalidad de autenticación del administrador, desarrollando la pantalla de inicio con su respectivo formulario de ingreso permitiéndole la opción de mantenerse conectado, recordando el navegador su correcto ingreso. De igual manera se desarrollarán los módulos de sacerdotes e instalaciones. Cada módulo contara con sus respectivas pantallas para poder ser ingresados, listados, modificados o eliminados, también contarán con la capacidad de buscar un sacerdote o una instalación según su nombre. Al final del sprint se tendrá los dos módulos base, los cuales podrán ser utilizados para los siguientes módulos, respetando así el desarrollo incremental.

5.2.2.2 Sprint Backlog

En el sprint backlog se detallan la lista de tareas asignadas para el primer sprint. Las tareas o funcionalidades a completarse durante el sprint han sido escogidas mediante su orden de prioridad por lo cual se ha obtenido un número de 12 funcionalidades las cuáles son listadas a continuación.

Tabla N 9 Sprint Backlog 1

Número	Descripción	Story Point	Rol	Prioridad	Sprint	Tiempo en horas	Criterio de Aceptación
1	Elaborar una pantalla para la autenticación del sistema	4	Administrador	5	1	6	El sistema debe guardar debe validar las credenciales y guardar los logs que realice el administrador.
2	Elaborar una pantalla que muestre los distintos módulos del sistema	2	Administrador	5	1	2	La pantalla debe indicar todos los módulos del sistema y un submenú para visualizar los módulos o ingresar una nueva entrada
3	Elaborar una pantalla para ingresar una instalación	3	Administrador	5	1	4	En la pantalla se debe ingresar toda la información necesaria y debe permitir ingresar una o más fotos asociadas a la instalación
4	Listarlas instalaciones con el nombre de las mismas en una tabla	2	Administrador	5	1	3	Se debe visualizar en la pantalla los nombres de las instalaciones.
5	Agregar un botón en la lista de las instalaciones para poder buscar una instalación en particular.	2	Administrador	5	1	4	En la pantalla donde se visualice las instalaciones debe haber un botón para buscar la instalación y debe mostrar la instalación obtenida de la búsqueda en una nueva pantalla.
6	Elaborar una pantalla para la modificación de la información de las instalaciones	3	Administrador	5	1	5	La pantalla de edición de datos de las instalaciones debe mostrar los campos cargados con la información ya ingresada.
7	Elaborar una pantalla para indicar si está seguro de eliminar una instalación	4	Administrador	5	1	1	La pantalla debe indicar el mensaje de eliminar la instalación, si se elimina una instalación debe indicarse que se deje de utilizarse en los módulos que están obtenido su información y eliminar todas las fotos asociadas al mismo.
8	Listar los cursos y/o servicios que están ocupando una instalación que se necesite dar de alta	3	Administrador	5	1	3	Se debe listar en una pantalla los servicios o cursos con un hipervínculo que los lleve directamente a editar dicho curso o servicio.

Número	Descripción	Story Point	Rol	Prioridad	Sprint	Tiempo en horas	Criterio de Aceptación
9	Elaborar una pantalla para ingresar los datos de un sacerdote que ingrese al Santuario	2	Administrador	5	1	4	La pantalla debe mostrar todos los campos necesarios para ingresar un nuevo sacerdote, al finalizar el ingreso debe redirigirse a la misma pantalla para permitir un nuevo ingreso. Debe tener un botón de finalización para dirigirse al menú principal.
10	Elaborar una tabla para listar a los sacerdotes.	2	Administrador	3	1	3	La tabla debe mostrar a todos los sacerdotes. Debe tener dos botones, uno de eliminar y el otro de editar la información del sacerdote seleccionado.
11	Elaborar una pantalla modificar la información del sacerdote.	2	Administrador	3	1	5	La pantalla debe ser cargada con la información obtenida del sacerdote.
12	Elaborar una pantalla para la confirmación de la eliminación de un sacerdote.	2	Administrador	3	1	2	La pantalla debe indicar si está seguro de eliminar la información, en caso de que el sacerdote este a cargo de cursos y/o servicios se debe redirigir a una nueva pantalla.

Elaboración: Mariano Navarrete

5.2.2.3 Duración del sprint

La estimación de tiempo para el desarrollo de las funcionalidades que han sido tomas para este sprint es de 42 horas. Dichas horas equivalen a 5 días de trabajo de 8 horas y un día de trabajo de 2 horas.

5.2.2.4 Revisión del sprint

Una vez culminado el primer sprint, se presentó al cliente mediante una reunión las funcionalidades realizadas. Se presentó los formularios de ingreso, así como la capacidad de ingresar varias imágenes a una instalación. Igualmente se pudo evidenciar la lista de los sacerdotes y la posibilidad de dar de alta a un sacerdote, así como modificar su información. Finalmente, el cliente expresó la satisfacción con las funcionalidades desarrolladas y la conformidad con los datos requeridos en los formularios y la forma en que son ingresados.

5.2.3 Sprint 2

5.2.3.1 Planificación del sprint

En este segundo sprint se desarrollará la funcionalidad de los servicios que ofrece el Santuario. Se desarrollarán las pantallas para el ingreso, modificación, visualización y eliminación de los servicios. En este módulo se permitirá seleccionar una instalación en donde se realiza el servicio y el sacerdote que está a cargo de dicho servicio. De igual manera debe permitir el ingreso de los distintos horarios y días que el servicio esté disponible.

5.2.3.2 Sprint Backlog

En el sprint backlog se detallan la lista de tareas pendientes para el segundo sprint. Las tareas o funcionalidades a completarse durante el sprint han sido escogidas mediante su orden de

prioridad por lo cual se ha obtenido un número de 9 funcionalidades las cuáles son listadas a continuación.

Tabla N 10 Sprint Backlog 2

Número	Descripción	Story Point	Rol	Prioridad	Sprint	Tiempo	Criterio de Aceptación
1	Listar los cursos y/o servicios que están a cargo del sacerdote que se desee eliminar.	3	Administrador	3	2	4	Debe mostrar todos los cursos y/o servicios que el sacerdote esté a cargo con un hipervínculo para su edición.
2	Elaborar una pantalla para el ingreso de un servicio.	3	Administrador	5	2	4	La pantalla debe visualizar todos los campos necesarios y debe existir la forma de seleccionar una instalación.
3	Elaborar una pantalla de ingreso de información extra.	3	Administrador	5	2	3	Debe tener dos botones uno de ingreso de un horario y el otro para finalizar el ingreso del servicio.
4	Elaborar un modal para el ingreso de horarios al servicio.	2	Administrador	2	2	3	Debe permitir el ingreso de los horarios necesarios para el servicio.
5	Listar los servicios en una tabla.	2	Administrador	3	2	3	Debe listar los servicios con la posibilidad de editarlos o eliminarlos.
6	Elaborar una pantalla para la edición de un servicio.	3	Administrador	3	2	5	La pantalla debe tener la información del servicio, un botón para editar los horarios y un botón para finalizar la edición.
7	Listar los horarios de un servicio en una tabla.	3	Administrador	3	2	3	La tabla debe indicar todos los horarios asociados al servicio y debe tener los botones necesarios para la eliminación o edición del horario, de igual manera debe contar con un botón para la finalización de la edición del servicio.
8	Crear un componente para la búsqueda de un servicio en particular.	2	Administrador	2	2	4	El componente debe permitir buscar un servicio por su nombre.
9	Elaborar una pantalla para la eliminación del servicio.	2	Administrador	3	2	4	Debe indicar si está seguro de eliminar el servicio y deben eliminarse todos los horarios asociados al servicio

Elaboración: Mariano Navarrete

5.2.3.3 Duración del sprint

La estimación de tiempo para el desarrollo de las funcionalidades que han sido tomadas para este sprint es de 33 horas. Dichas horas equivalen a 4 días de trabajo de 8 horas y un día de trabajo de 1 hora.

5.2.3.4 Revisión del sprint

Una vez culminado el segundo sprint, se presentó al cliente mediante una reunión las funcionalidades realizadas. Se presentó el formulario de ingreso y como se puede seleccionar una instalación y el sacerdote al servicio que se está creando, así como sus pantallas para eliminarlos y modificarlos. Finalmente, el cliente expresó la satisfacción con las funcionalidades desarrolladas y la conformidad con los datos requeridos en los formularios y la forma en que son ingresados.

5.2.4 Sprint 3

5.2.4.1 Planificación del sprint

En este tercer sprint se desarrollarán las funcionalidades las cuales son cursos y sacramentos que ofrece el Santuario. Se desarrollarán las pantallas para el ingreso, modificación, visualización y eliminación de los cursos y sacramentos. En este sprint se sigue manteniendo el orden de prioridad y el desarrollo incremental, por lo cual la primera funcionalidad desarrollada será los cursos, luego los sacramentos, ya que un sacramento puede consumir un curso.

5.2.4.2 Sprint Backlog

Las tareas o funcionalidades a completarse durante el sprint han sido escogidas mediante su orden de prioridad por lo cual se ha obtenido un número de 14 funcionalidades las cuáles son listadas a continuación.

Tabla N 11 Sprint Backlog 3

Número	Descripción	Story Point	Rol	Prioridad	Sprint	Tiempo	Criterio de Aceptación
1	Elaborar una pantalla para el ingreso de un curso.	2	Administrador	5	3	4	La pantalla debe permitir el ingreso de todos los datos necesario y debe incluir una manera de ingresar el sacerdote a cargo como la instalación.
2	Elaborar una pantalla para ingresar los horarios del curso.	2	Administrador	3	3	3	La pantalla debe permitir ingresar todos los horarios que sean necesarios para el curo y debe tener un botón para finalizar el ingreso.
3	Elaborar un modal para el ingreso de los horarios.	2	Administrador	3	3	3	El modal debe tener la información necesaria para la creación del horario.
4	Listar todos los cursos en una tabla.	2	Administrador	3	3	3	La tabla debe contener botones para eliminar o la edición de un curso.
5	Crear un componente para la búsqueda en particular de un curso.	3	Administrador	2	3	5	El componente debe permitir buscar a un curso por su nombre.
6	Elaborar una pantalla para la eliminación del curso.	3	Administrador	3	3	5	La pantalla debe indicar si esta seguro de eliminar el curso y debe eliminar a todos los horarios que están asociados al curso.
7	Elaborar una pantalla para el ingreso de un sacramento.	2	Administrador	5	3	5	La pantalla debe permitir el ingreso de toda la información del sacramento y debe permitir seleccionar el curso necesario para el sacramento.
8	Elaborar una pantalla para el ingreso de horarios.	2	Administrador	4	3	3	La pantalla debe permitir el ingreso de todos los horarios necesarios y debe tener un botón para la finalización.
9	Elaborar un modal para el ingreso de los horarios.	2	Administrador	4	3	3	El modal debe permitir toda la información del horario.
10	Listar los sacramentos ingresados en una tabla	2	Administrador	3	3	3	Debe permitir mostrar todos los sacramentos con botones para la edición o eliminación.
11	Elaborar una pantalla para la edición de un sacramento.	3	Administrador	3	3	5	La pantalla debe tener la información cargada del sacramento y debe tener un botón para la edición de los horarios

Número	Descripción	Story Point	Rol	Prioridad	Sprint	Tiempo	Criterio de Aceptación
12	Listar los horarios del sacramento.	2	Administrador	3	3	4	Se debe listar todos los horarios asociados al sacramento para poder editarlos o eliminarlos.
13	Elaborar una pantalla para la eliminación de un sacramento.	3	Administrador	3	3	4	Se debe indicar si está seguro de eliminar el sacramento y se debe eliminar todos los horarios asociados al sacramento.
14	Elaborar un componente para buscar el sacramento	3	Administrador	2	3	4	El componente debe permitir buscar un sacramento por su tipo.

Elaboración: Mariano Navarrete

5.2.4.3 Duración del sprint

La estimación de tiempo para el desarrollo de las funcionalidades que han sido tomadas para este sprint es de 54 horas. Dichas horas equivalen a 6 días de trabajo de 8 horas y un día de trabajo de 6 horas.

5.2.4.4 Revisión del sprint

Una vez culminado el tercer sprint, se presentó al cliente mediante una reunión las funcionalidades realizadas. Se presentó el formulario de ingreso tanto para los cursos como para los sacramentos, igualmente, se presentó las distintas pantallas para la visualización, modificación y eliminación de los mismos. Las dos funcionalidades tenían la característica de ingresar varios horarios los cuales fueron verificados. Finalmente, el cliente expresó la satisfacción con las funcionalidades desarrolladas y la conformidad con los datos requeridos en los formularios y la forma en que son ingresados.

5.2.5 Sprint 4

5.2.5.1 Planificación del sprint

Aquí se desplegará la información de los diferentes cursos, sacramentos y servicios que ofrece el Santuario. Se implementará la geolocalización de las instalaciones la cuál será implementada en las páginas informativas de los módulos antes mencionados.

5.2.5.2 Sprint Backlog

Las tareas o funcionalidades a completarse durante el sprint han sido escogidas mediante su orden de prioridad por lo cual se ha obtenido un número de 4 funcionalidades las cuáles son listadas a continuación.

Tabla N 12 Sprint Backlog 4

Número	Descripción	Story Point	Rol	Prioridad	Sprint	Tiempo	Criterio de Aceptación
1	Elaborar una pantalla que muestre la información general de las actividades del Santuario.	3	Usuario	2	4	6	La pantalla debe tener una barra de navegación que contenga todas las categorías que ofrece el Santuario.
2	Elaborar una pantalla que muestre una categoría en específico.	4	Usuario	2	4	6	La pantalla debe contener todos los cursos, servicios o sacramentos ofertados.
3	Elaborar una pantalla que muestre un curso, servicio o sacramento.	4	Usuario	2	4	6	La pantalla debe mostrar toda la información de un curso, servicio o sacramento.
4	Crear un componente para indicar la geolocalización de las instalaciones.	5	Usuario	2	4	9	En el componente se debe visualizar un mapa con el punto de la ubicación real de la instalación a la que se haga referencia.

Elaboración: Mariano Navarrete

5.2.5.3 Duración del sprint

La estimación de tiempo para el desarrollo de las funcionalidades que han sido tomas para este sprint es de 27 horas. Dichas horas equivalen a 3 días de trabajo de 8 horas y un día de trabajo de 3 horas.

5.2.5.4 Revisión del sprint

Una vez culminado el cuarto sprint, se presentó al cliente mediante una reunión las funcionalidades realizadas. Se presentó la página principal y su barra de navegación hacia los distintos módulos, así como la información de cada módulo con su respectivo mapa de geolocalización de la instalación. Finalmente, el cliente expresó la satisfacción con las funcionalidades desarrolladas y la conformidad con los datos requeridos en los formularios y la forma en que son ingresados.

6. CAPÍTULO 7: EVALUACIÓN DEL CORRECTO FUNCIONAMIENTO DEL SISTEMA WEB Y GUÍA DE INSTALACIÓN

En este capítulo se evaluarán las funcionalidades construidas. Para que una funcionalidad sea aceptada tendrá que cumplir con los criterios de aceptación, las cuales están ubicadas en el Product Backolog. Las pruebas están dividas en las funcionalidades construidas en cada sprint.

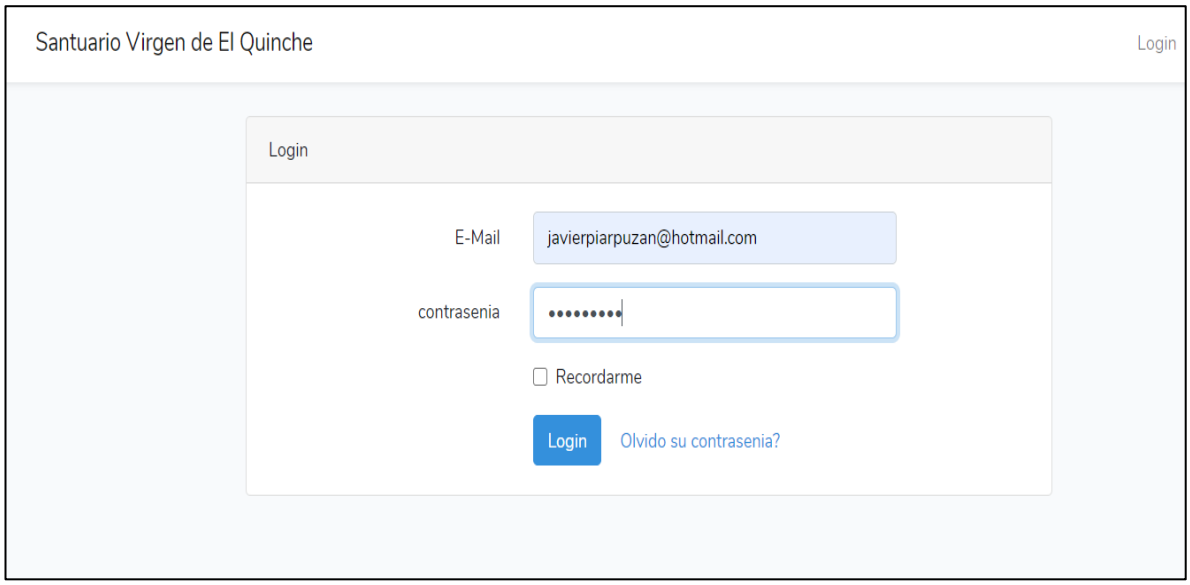
6.1 SPRINT 1

Descripción: Elaborar una pantalla para la autenticación del sistema.

Rol: Administrador

Criterio de aceptación: El sistema debe guardar, validar las credenciales y guardar los logs que realice el administrador.

Gráfico N 7 Pantalla Login del administrador



Elaboración: Mariano Navarrete

Gráfico N 8 Pantalla de inicio del sistema



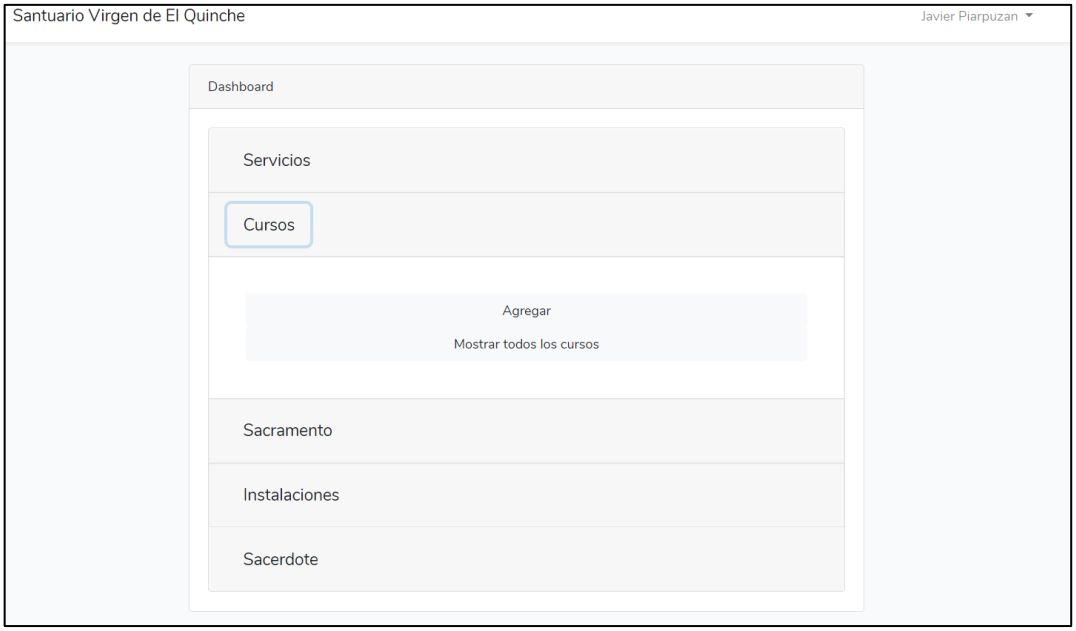
Elaboración: Mariano Navarrete

Descripción: Elaborar una pantalla que muestre los distintos módulos del sistema.

Rol: Administrador

Criterio de aceptación: La pantalla debe indicar todos los módulos del sistema y un submenú para visualizar los módulos o ingresar una nueva entrada.

Gráfico N 9 Pantalla de inicio del sistema



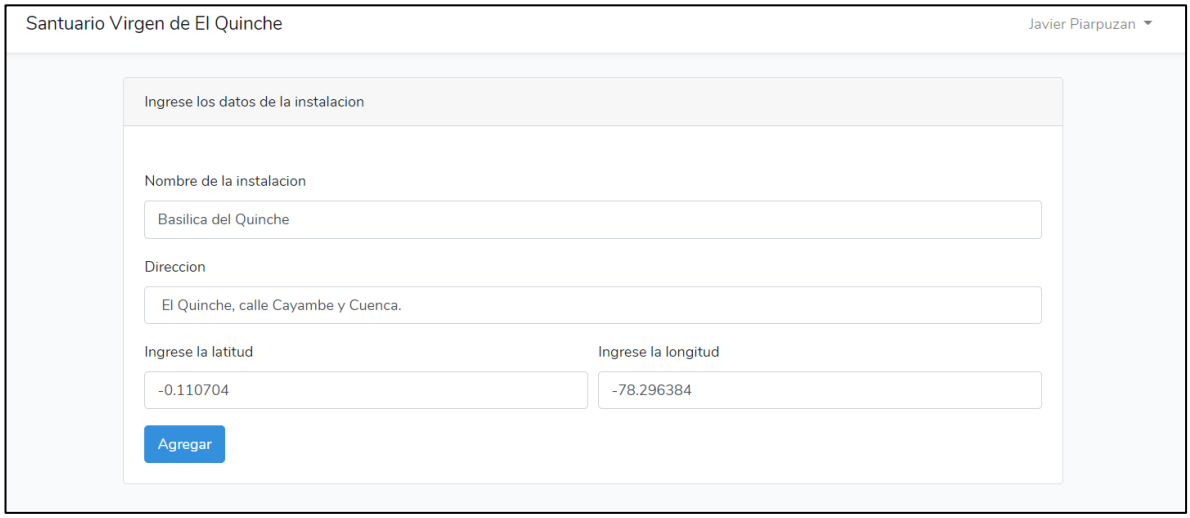
Elaboración: Mariano Navarrete

Descripción: Elaborar una pantalla para ingresar una instalación.

Rol: Administrador

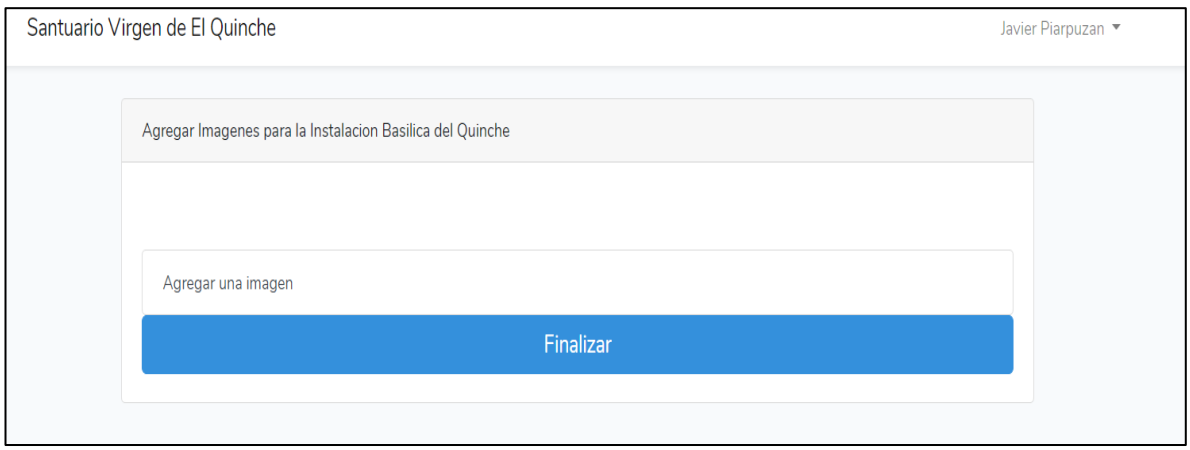
Criterio de aceptación: En la pantalla se debe ingresar toda la información necesaria y debe permitir ingresar una o más fotos asociadas a la instalación.

Gráfico N 10 Pantalla de ingreso de una instalación



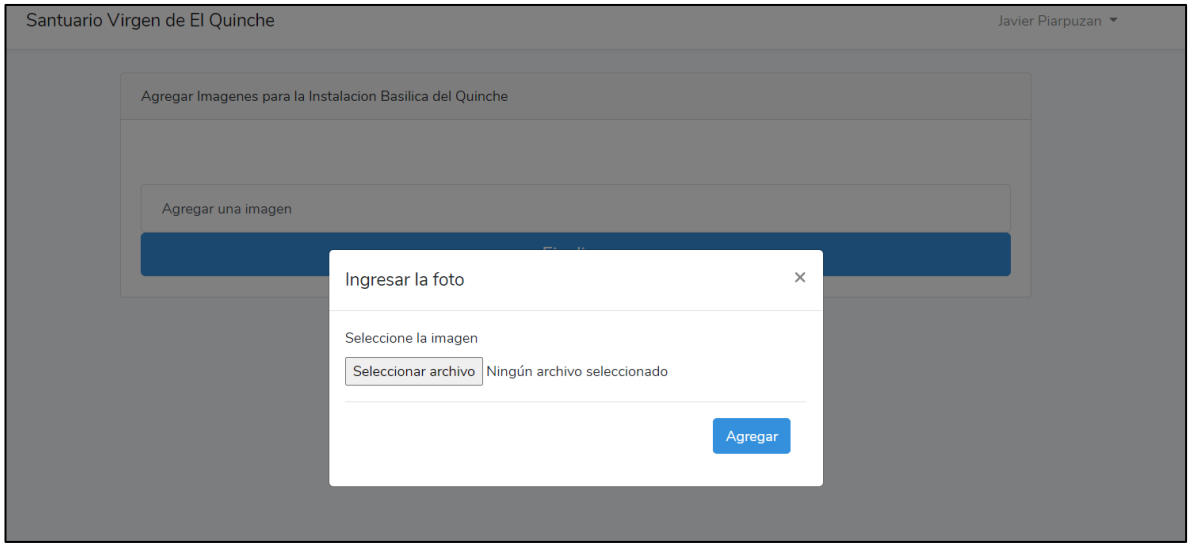
Elaboración: Mariano Navarrete

Gráfico N 11 Pantalla para finalizar el ingreso o ingreso de nuevas fotos



Elaboración: Mariano Navarrete

Gráfico N 12 Pantalla de ingreso de una foto para una instalación



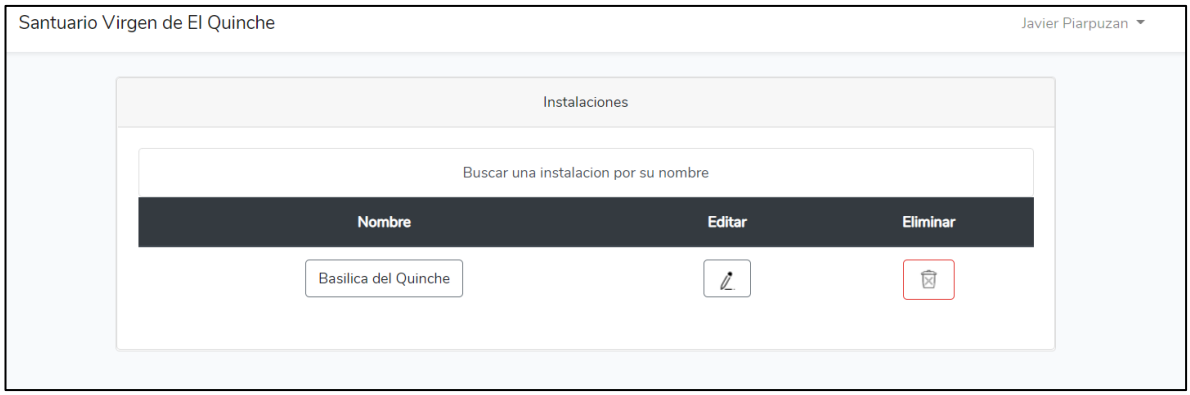
Elaboración: Mariano Navarrete

Descripción: Listarlas instalaciones con el nombre de las mismas en una tabla.

Rol: Administrador

Criterio de aceptación: Se debe visualizar en la pantalla los nombres de las instalaciones.

Gráfico N 13 Tabla de información de las instalaciones



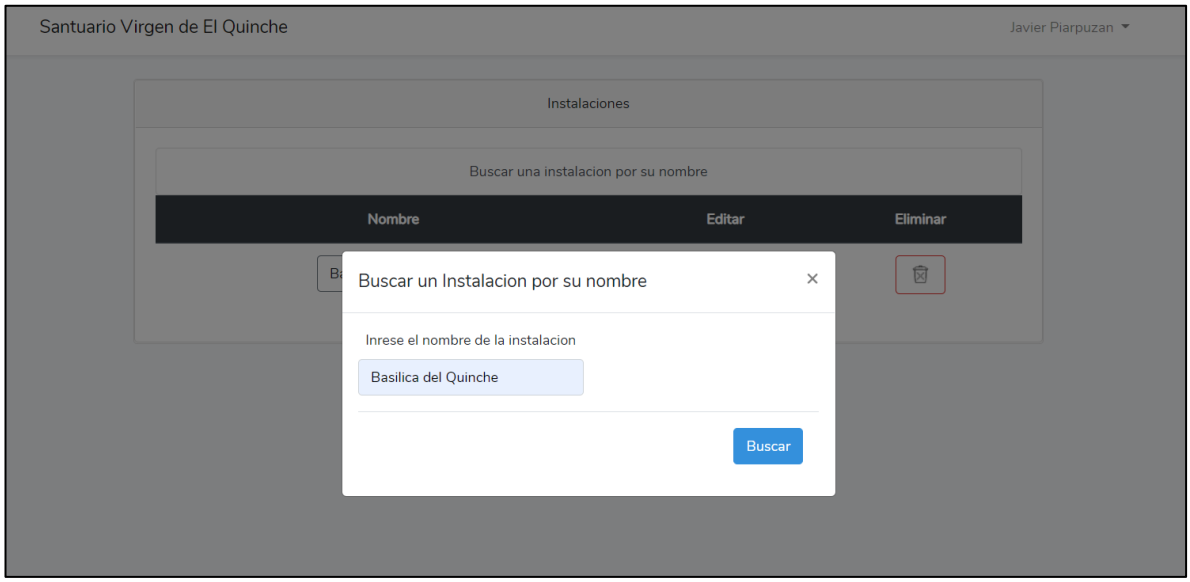
Elaboración: Mariano Navarrete

Descripción: Agregar un botón en la lista de las instalaciones para poder buscar una instalación en particular.

Rol: Administrador

Criterio de aceptación: En la pantalla donde se visualice las instalaciones debe haber un botón para buscar la instalación y debe mostrar la instalación obtenida de la búsqueda en una nueva pantalla.

Gráfico N 14 Modal para buscar una instalación por su nombre



Elaboración: Mariano Navarrete

Gráfico N 15 Pantalla e información de una instalación



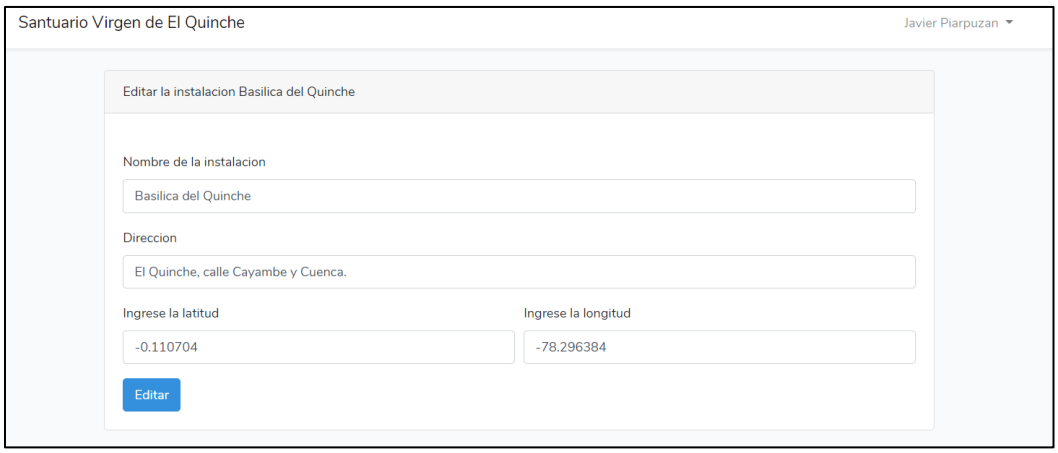
Elaboración: Mariano Navarrete

Descripción: Elaborar una pantalla para la modificación de la información de las instalaciones.

Rol: Administrador

Criterio de aceptación: La pantalla de edición de datos de las instalaciones debe mostrar los campos cargados con la información ya ingresada.

Gráfico N 16 Pantalla de edición de la información de una instalación



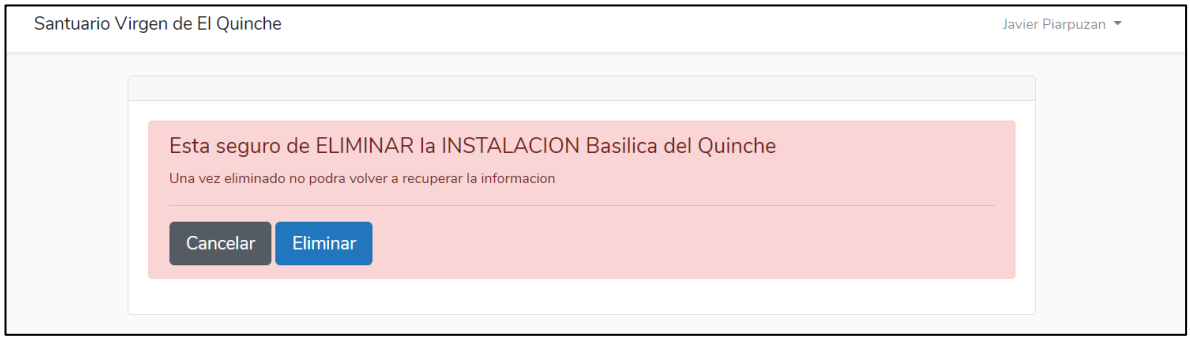
Elaboración: Mariano Navarrete

Descripción: Elaborar una pantalla para indicar si está seguro de eliminar una instalación.

Rol: Administrador

Criterio de aceptación: La pantalla debe indicar el mensaje de eliminar la instalación, si se elimina una instalación debe indicarse que se deje de utilizarse en los módulos que están obtenido su información y eliminar todas las fotos asociadas al mismo.

Gráfico N 17 Pantalla para la confirmación de la eliminación de la instalación



Elaboración: Mariano Navarrete

Descripción: Listar los cursos y/o servicios que están ocupando una instalación que se necesite dar de alta.

Rol: Administrador

Criterio de aceptación: Se debe listar en una pantalla los servicios o cursos con un hipervínculo que los lleve directamente a editar dicho curso o servicio.

Gráfico N 18 Pantalla para la confirmación de la eliminación de la instalación



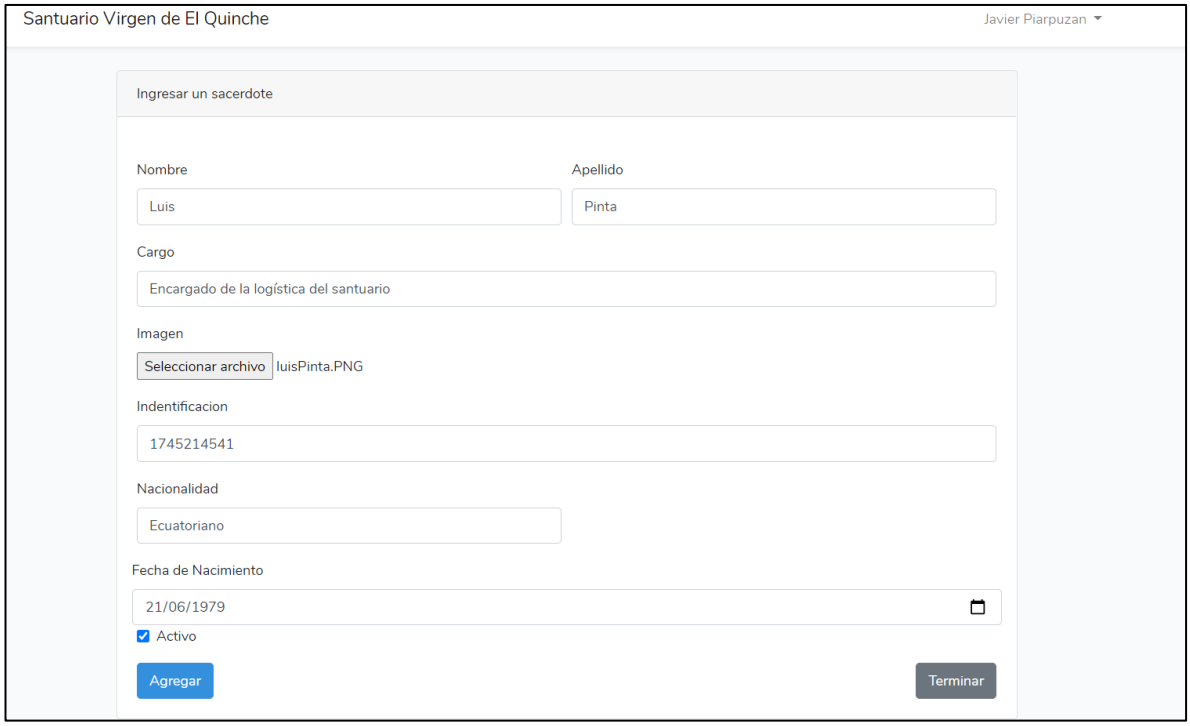
Elaboración: Mariano Navarrete

Descripción: Elaborar una pantalla para ingresar los datos de un sacerdote que ingrese al Santuario.

Rol: Administrador

Criterio de aceptación: La pantalla debe mostrar todos los campos necesarios para ingresar un nuevo sacerdote, al finalizar el ingreso debe redirigirse a la misma pantalla para permitir un nuevo ingreso. Debe tener un botón de finalización para dirigirse al menú principal.

Gráfico N 19 Pantalla de ingreso de un nuevo sacerdote



Elaboración: Mariano Navarrete

Descripción: Elaborar una tabla para listar a los sacerdotes.

Rol: Administrador

Criterio de aceptación: La tabla debe mostrar a todos los sacerdotes. Debe tener dos botones, uno de eliminar y el otro de editar la información del sacerdote seleccionado.

Gráfico N 20 Tabla de información de los sacerdotes



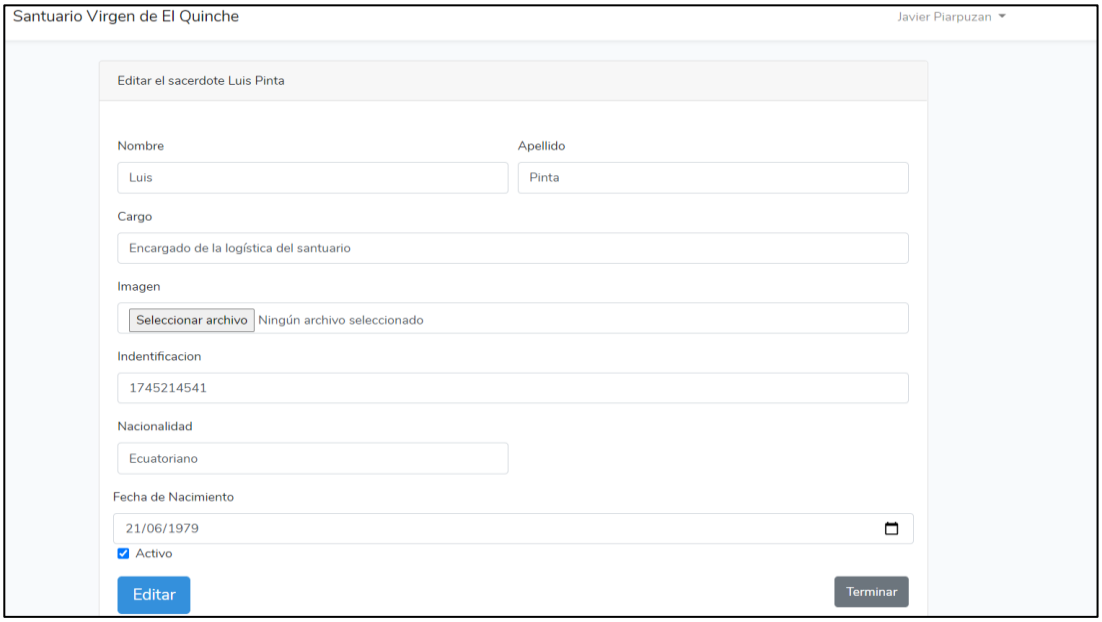
Elaboración: Mariano Navarrete

Descripción: Elaborar una pantalla modificar la información del sacerdote.

Rol: Administrador

Criterio de aceptación: La pantalla debe ser cargada con la información obtenida del sacerdote.

Gráfico N 21 Pantalla de edición de un sacerdote



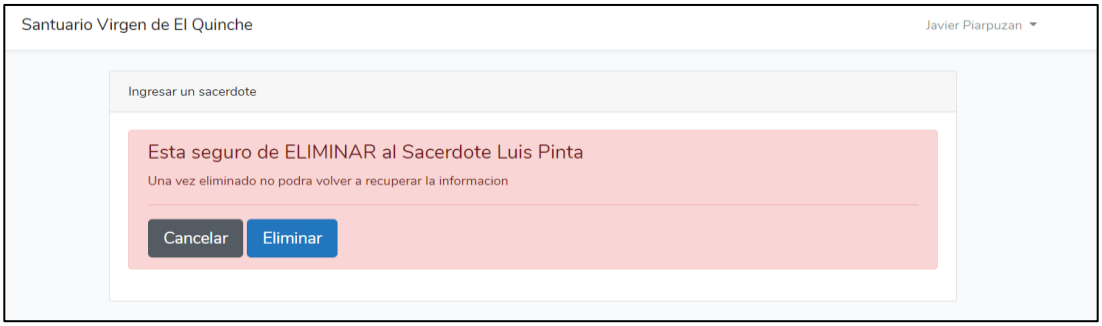
Elaboración: Mariano Navarrete

Descripción: Elaborar una pantalla para la confirmación de la eliminación de un sacerdote.

Rol: Administrador

Criterio de aceptación: La pantalla debe indicar que, si está seguro de eliminar la información, en caso de que el sacerdote este a cargo de cursos y/o servicios se debe redirigir a una nueva pantalla.

Gráfico N 22 Pantalla de confirmación de eliminación de un sacerdote



Elaboración: Mariano Navarrete

6.2 SPRINT 2

Descripción: Listar los cursos y/o servicios que están a cargo del sacerdote que se desee eliminar.

Rol: Administrador

Criterio de aceptación: Debe mostrar todos los cursos y/o servicios que el sacerdote este a cargo con un hipervínculo para su edición.

Gráfico N 23 Tabla para listar los cursos y/o servicios a cargo del sacerdote



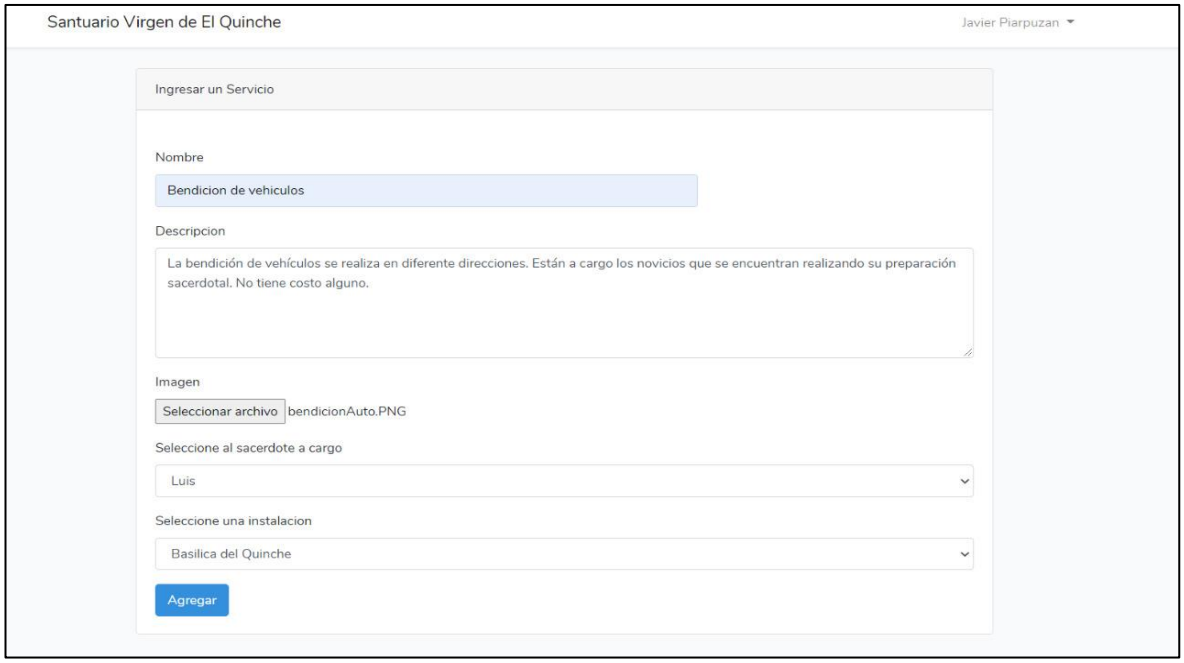
Elaboración: Mariano Navarrete

Descripción: Elaborar una pantalla para el ingreso de un servicio.

Rol: Administrador

Criterio de aceptación: La pantalla se debe visualizar todos los campos necesarios y debe existir la forma de seleccionar una instalación.

Gráfico N 24 Pantalla de ingreso de información de un nuevo servicio



Elaboración: Mariano Navarrete

Descripción: Elaborar una pantalla de ingreso de información extra.

Rol: Administrador

Criterio de aceptación: Debe tener dos botones uno de ingreso de un horario y el otro para finalizar el ingreso del servicio.

Gráfico N 25 Pantalla de ingreso de los horarios del servicio



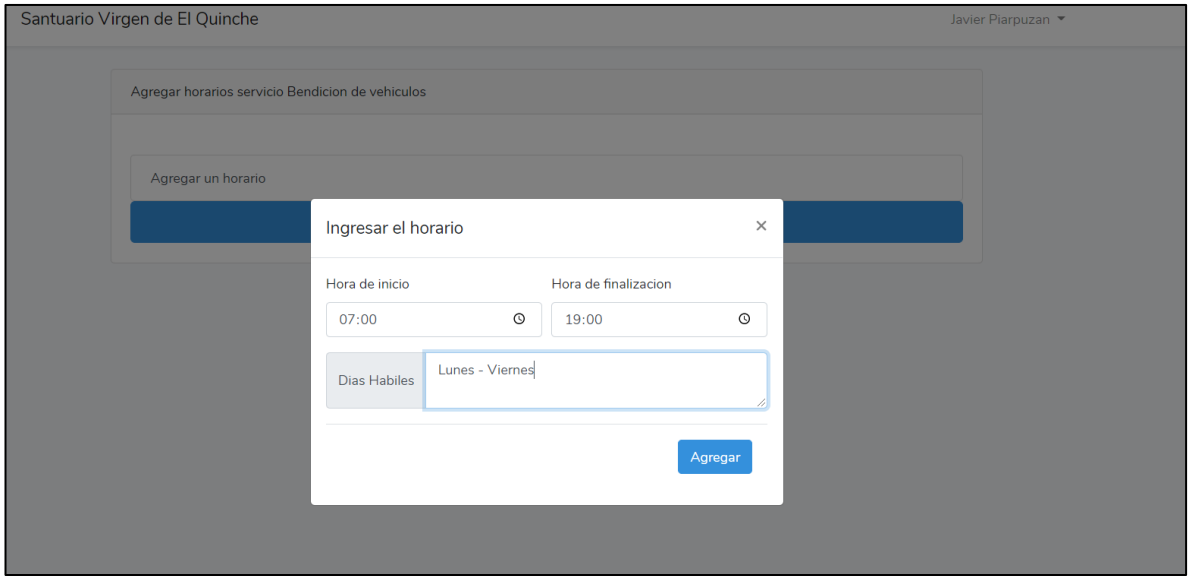
Elaboración: Mariano Navarrete

Descripción: Elaborar un modal para el ingreso de horarios al servicio.

Rol: Administrador

Criterio de aceptación: Debe permitir el ingreso de los horarios necesarios para el servicio.

Gráfico N 26 Modal para ingresar un horario del servicio



Elaboración: Mariano Navarrete

Descripción: Listar los servicios en una tabla.

Rol: Administrador

Criterio de aceptación: Debe listar los servicios con la posibilidad de editarlos o eliminarlos.

Gráfico N 27 Tabla para listar los servicios



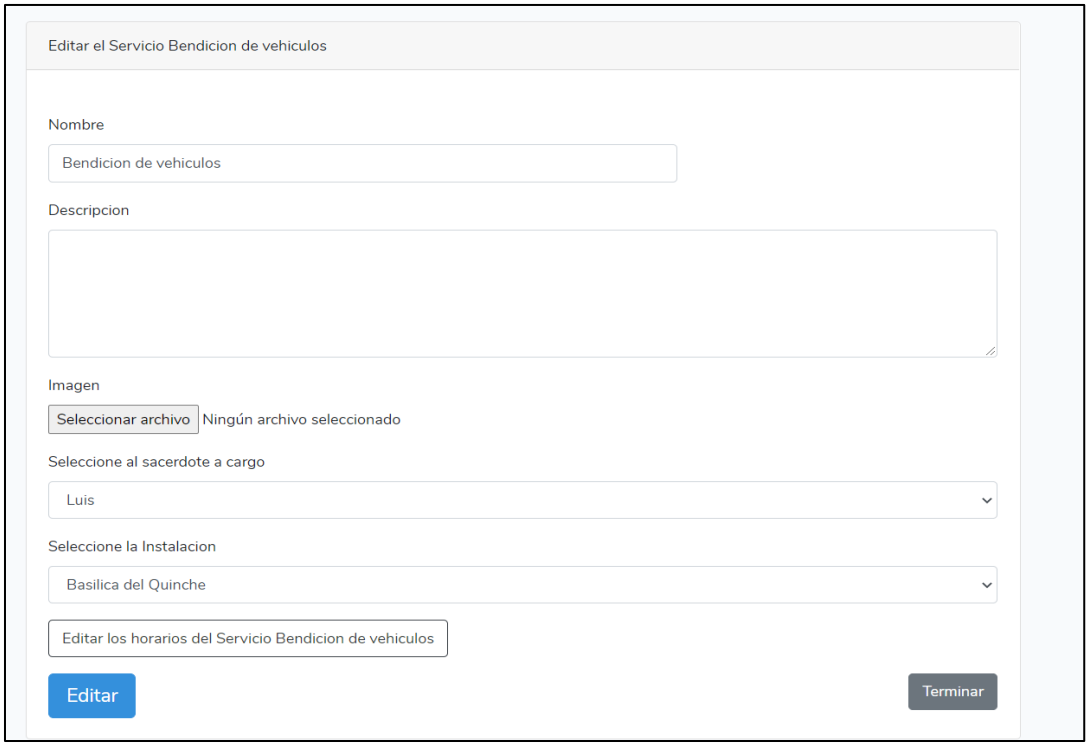
Elaboración: Mariano Navarrete

Descripción: Elaborar una pantalla para la edición de un servicio.

Rol: Administrador

Criterio de aceptación: La pantalla debe tener la información del servicio, un botón para editar los horarios y un botón para finalizar la edición.

Gráfico N 28 Pantalla para editar la información de un servicio



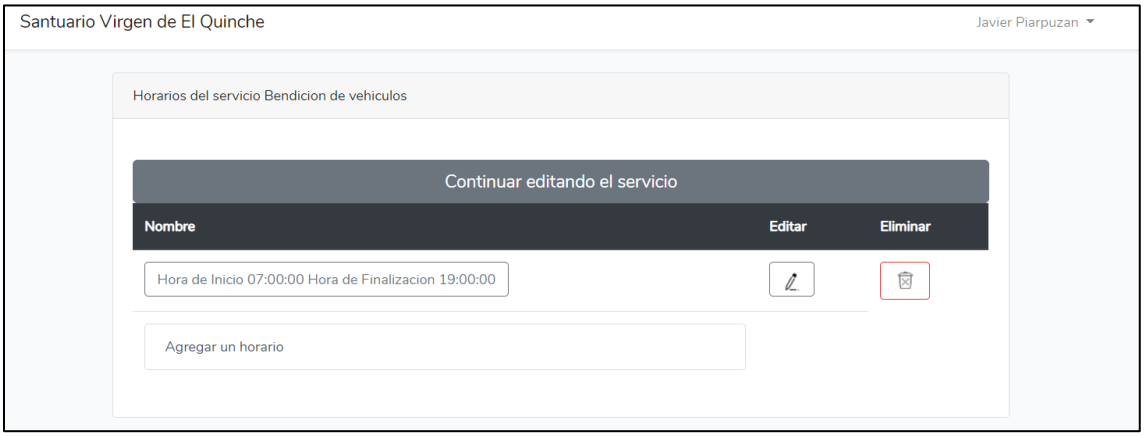
Elaboración: Mariano Navarrete

Descripción: Listar los horarios de un servicio en una tabla.

Rol: Administrador

Criterio de aceptación: La tabla debe indicar todos los horarios asociados al servicio y debe tener los botones necesarios para la eliminación o edición del horario, de igual manera debe contar con un botón para la finalización de la edición del servicio.

Gráfico N 29 Pantalla para listar los horarios que contiene un servicio



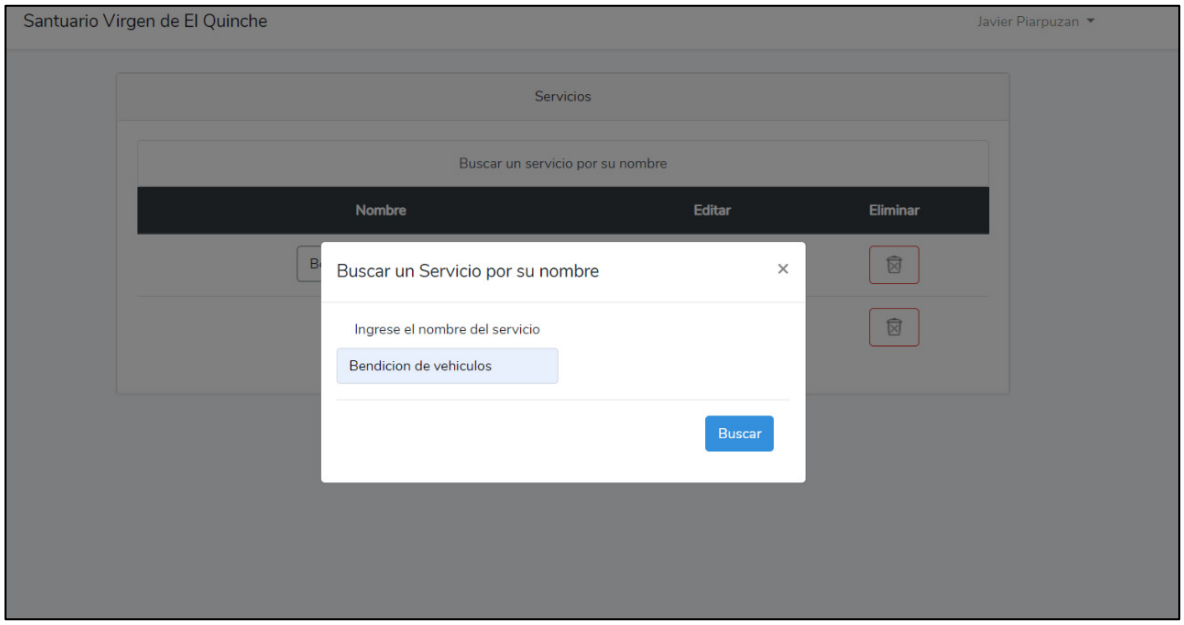
Elaboración: Mariano Navarrete

Descripción: Crear un componente para la búsqueda de un servicio en particular.

Rol: Administrador

Criterio de aceptación: El componente debe permitir buscar un servicio por su nombre.

Gráfico N 30 Modal para buscar un servicio por su nombre



Elaboración: Mariano Navarrete

Gráfico N 31 Pantalla de información del servicio buscado



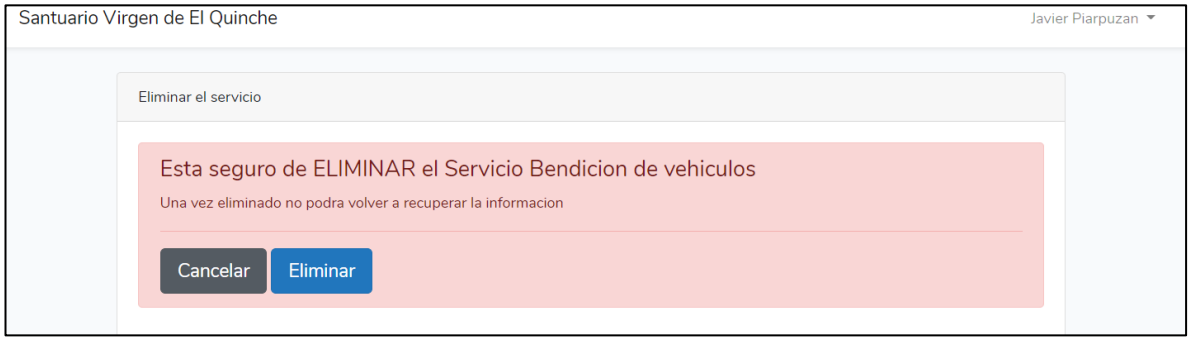
Elaboración: Mariano Navarrete

Descripción: Elaborar una pantalla para la eliminación del servicio.

Rol: Administrador

Criterio de aceptación: Debe indicar si está seguro de eliminar el servicio y deben eliminarse todos los horarios asociados al servicio.

Gráfico N 32 Pantalla para confirmar la eliminación de un servicio



Elaboración: Mariano Navarrete

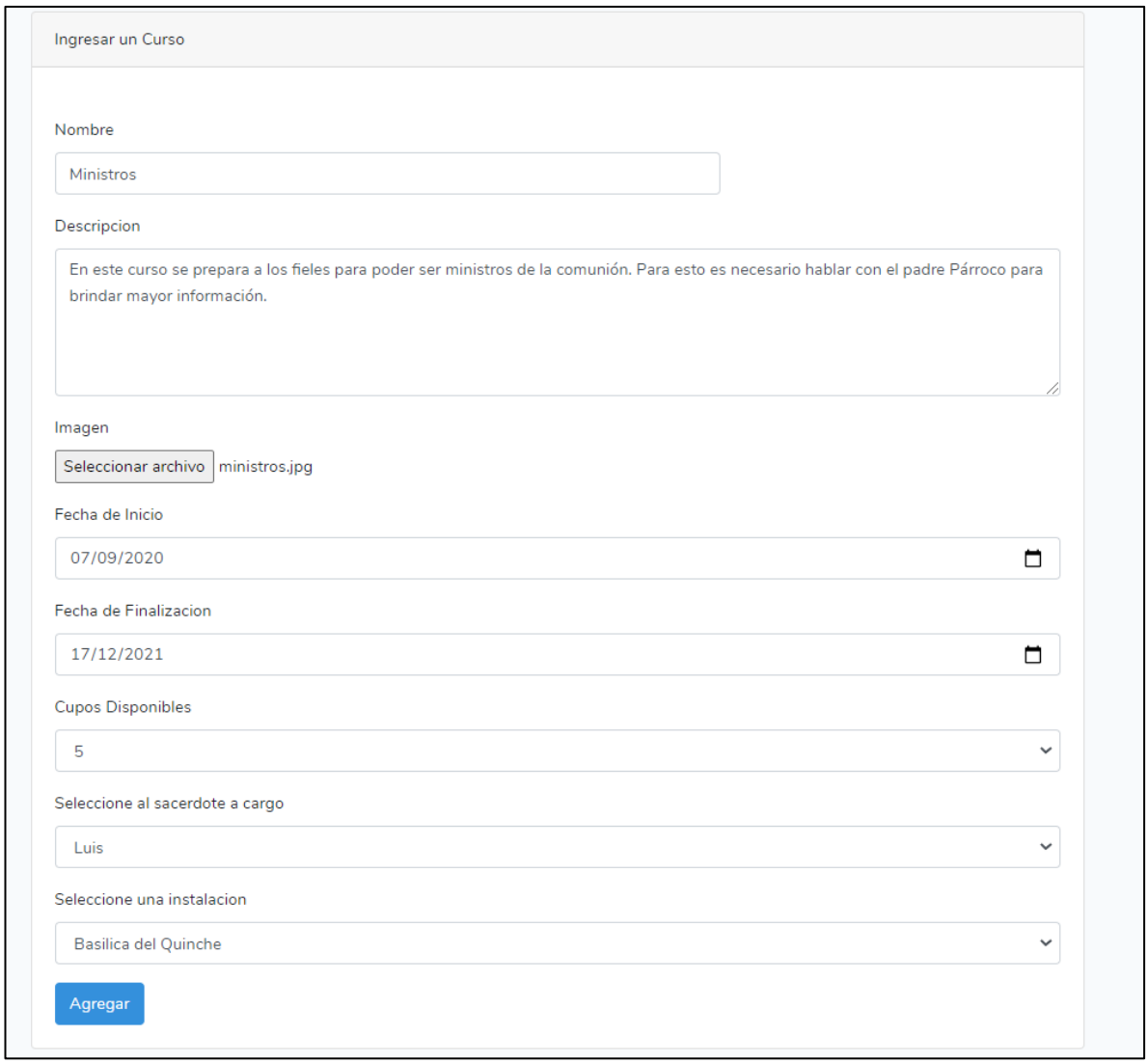
6.3 SPRINT 3

Descripción: Elaborar una pantalla para el ingreso de un curso.

Rol: Administrador

Criterio de aceptación: La pantalla debe permitir el ingreso de todos los datos necesario y debe incluir una manera de ingresar el sacerdote a cargo como la instalación.

Gráfico N 33 Pantalla para ingresar un curso



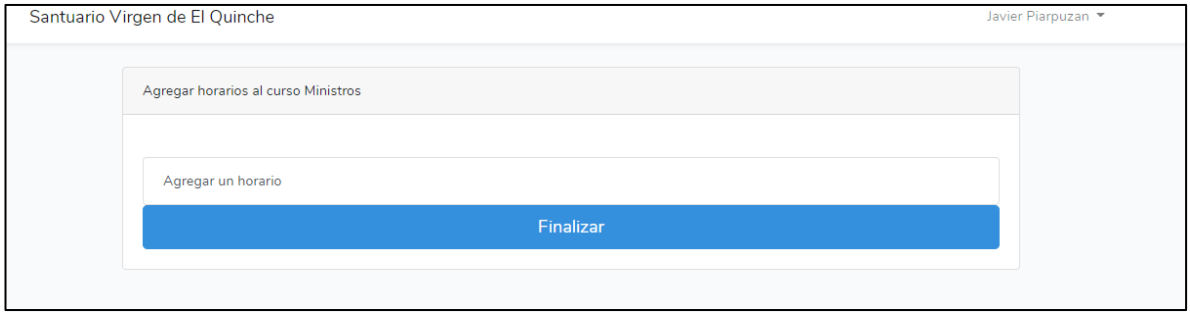
Elaboración: Mariano Navarrete

Descripción: Elaborar una pantalla para ingresar los horarios del curso.

Rol: Administrador

Criterio de aceptación: La pantalla debe permitir ingresar todos los horarios que sean necesarios para el curso y debe tener un botón para finalizar el ingreso.

Gráfico N 34 Pantalla para ingresar los horarios a un curso



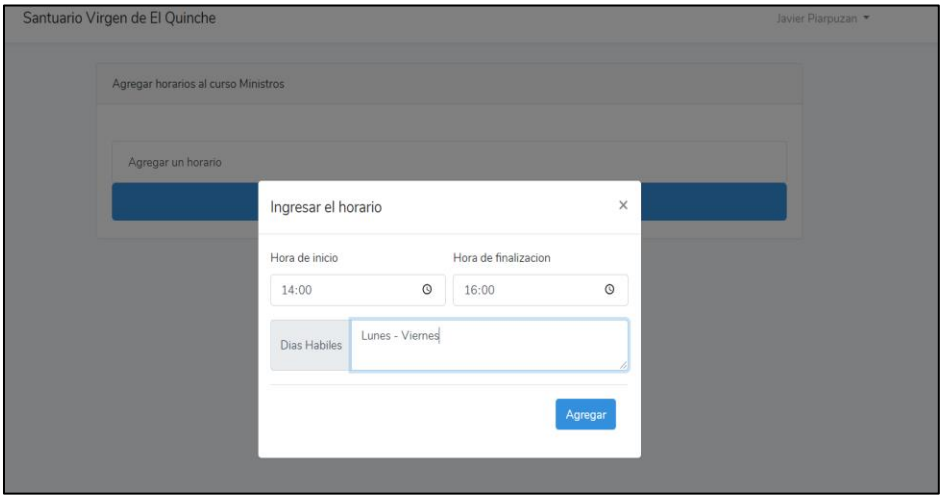
Elaboración: Mariano Navarrete

Descripción: Elaborar un modal para el ingreso de los horarios.

Rol: Administrador

Criterio de aceptación: El modal debe tener la información necesaria para la creación del horario.

Gráfico N 35 Pantalla para ingresar los horarios a un curso



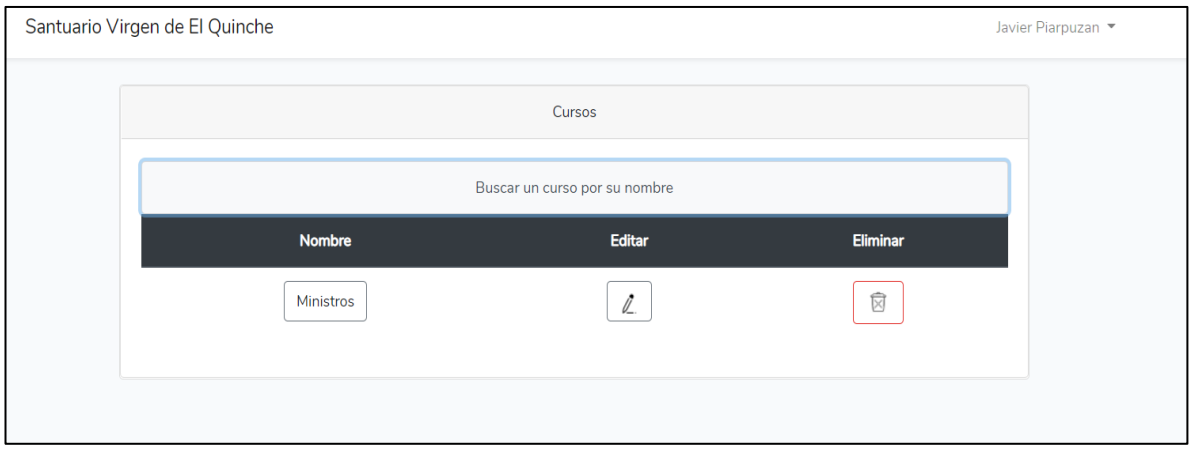
Elaboración: Mariano Navarrete

Descripción: Listar todos los cursos en una tabla.

Rol: Administrador

Criterio de aceptación: La tabla debe contener botones para eliminación, o la edición de un curso.

Gráfico N 36 Pantalla para listar los cursos



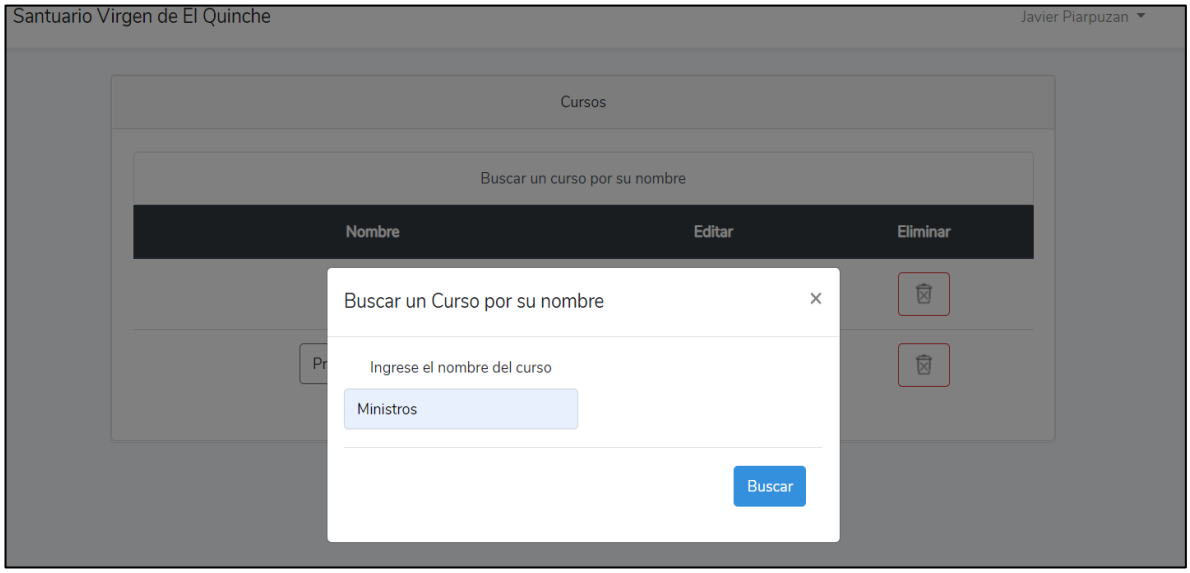
Elaboración: Mariano Navarrete

Descripción: Crear un componente para la búsqueda en particular de un curso.

Rol: Administrador

Criterio de aceptación: El componente debe permitir buscar a un curso por su nombre.

Gráfico N 37 Modal para buscar un curso por su nombre



Elaboración: Mariano Navarrete

Gráfico N 38 Pantalla de información de un curso



Elaboración: Mariano Navarrete

Descripción: Elaborar una pantalla para la eliminación del curso.

Rol: Administrador

Criterio de aceptación: La pantalla debe indicar si está seguro de eliminar el curso y debe eliminar a todos los horarios que están asociados al curso.

Gráfico N 39 Pantalla de confirmación de eliminación de un curso



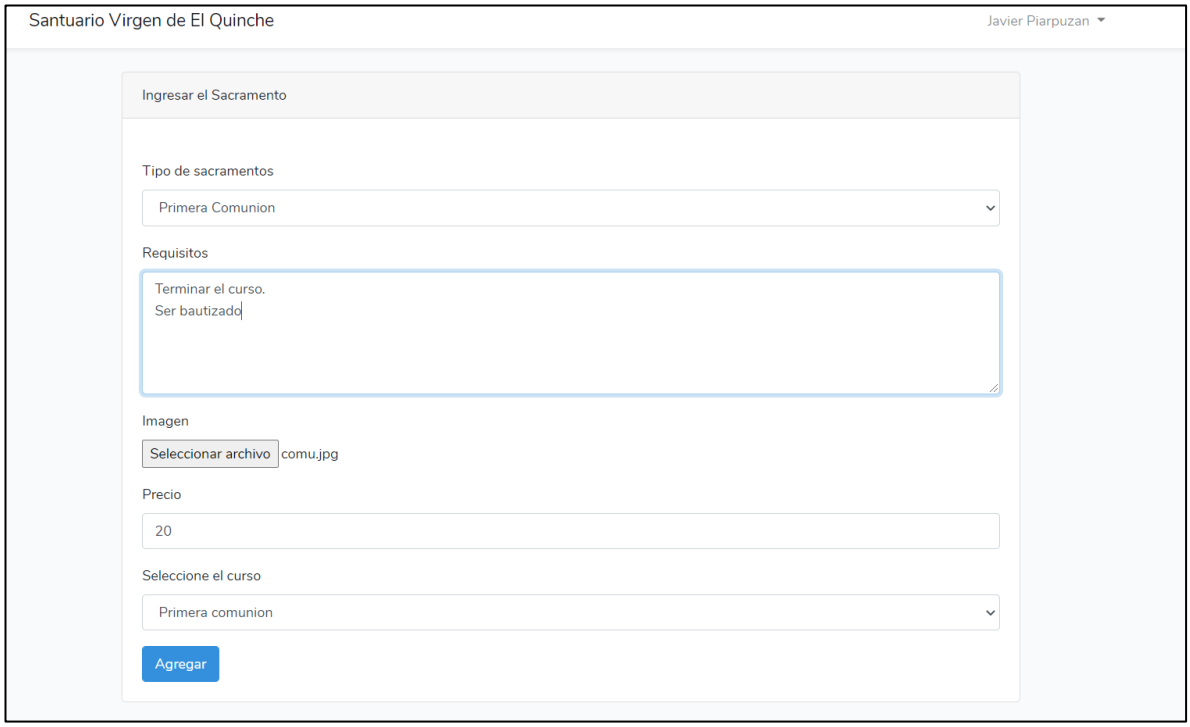
Elaboración: Mariano Navarrete

Descripción: Elaborar una pantalla para el ingreso de un sacramento.

Rol: Administrador

Criterio de aceptación: La pantalla debe permitir el ingreso de toda la información del sacramento y debe permitir seleccionar el curso necesario para el sacramento.

Gráfico N 40 Pantalla de ingreso de un sacramento



Elaboración: Mariano Navarrete

Descripción: Elaborar una pantalla para el ingreso de horarios.

Rol: Administrador

Criterio de aceptación: La pantalla debe permitir el ingreso de todos los horarios necesarios y debe tener un botón para la finalización.

Gráfico N 41 Pantalla para agregar horarios a un sacramento



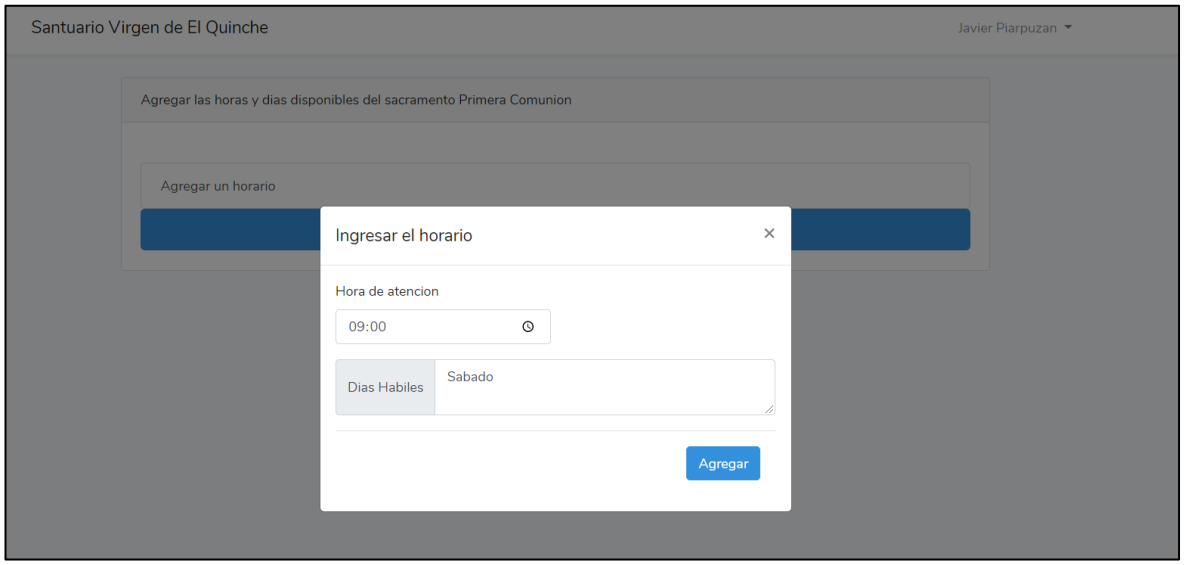
Elaboración: Mariano Navarrete

Descripción: Elaborar un modal para el ingreso de los horarios.

Rol: Administrador

Criterio de aceptación: El modal debe permitir toda la información del horario.

Gráfico N 42 Modal de ingreso de horarios para un sacramento



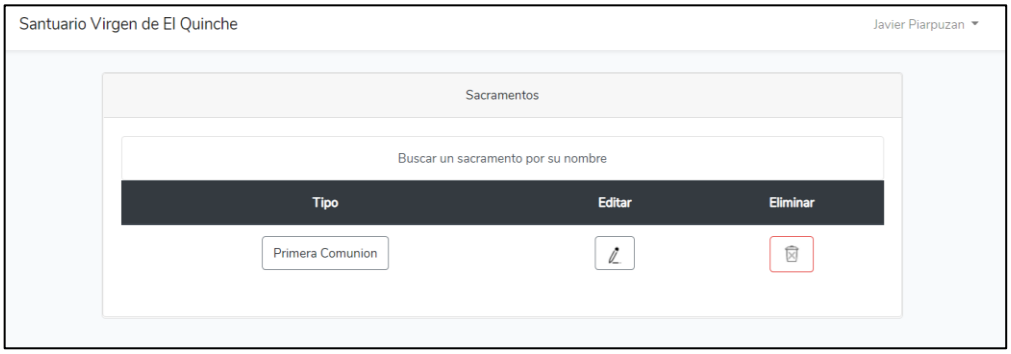
Elaboración: Mariano Navarrete

Descripción: Listar los sacramentos ingresados en una tabla.

Rol: Administrador

Criterio de aceptación: Debe permitir mostrar todos los sacramentos con botones para la edición o eliminación.

Gráfico N 43 Tabla para listar los sacramentos



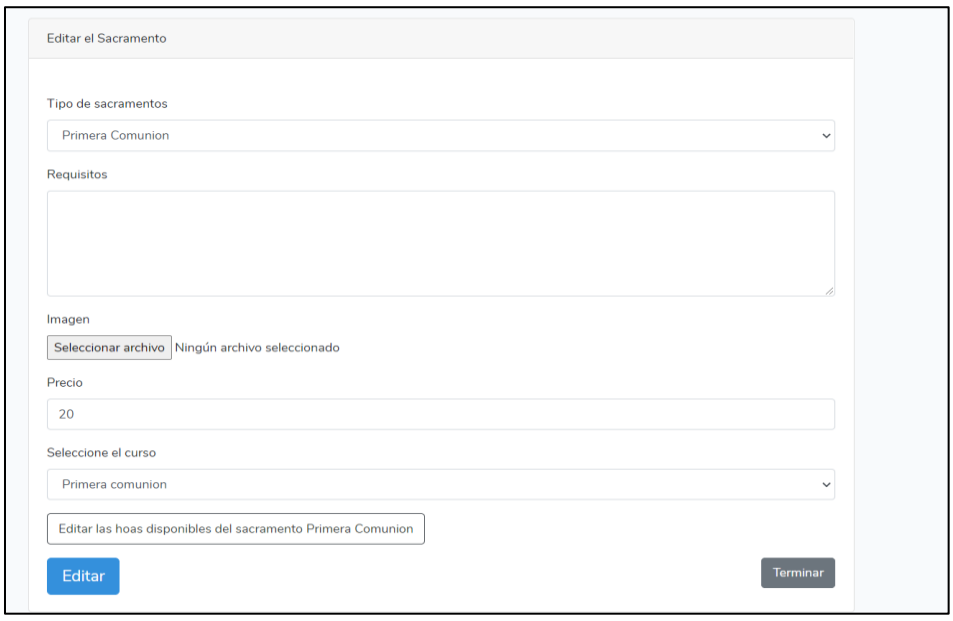
Elaboración: Mariano Navarrete

Descripción: Elaborar una pantalla para la edición de un sacramento.

Rol: Administrador

Criterio de aceptación: La pantalla debe tener la información cargada del sacramento y debe tener un botón para la edición de los horarios.

Gráfico N 44 Pantalla de edición de un sacramento



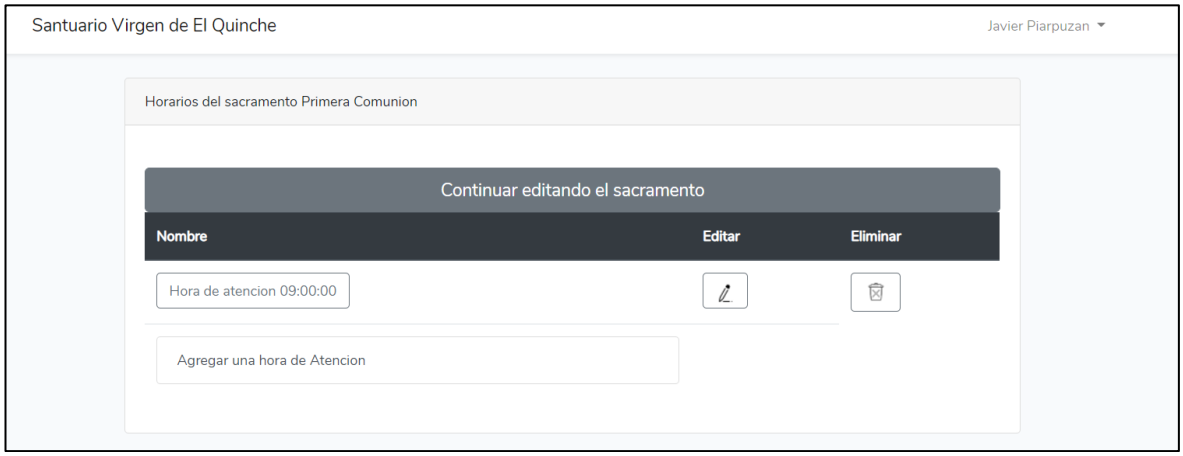
Elaboración: Mariano Navarrete

Descripción: Listar los horarios del sacramento.

Rol: Administrador

Criterio de aceptación: Se debe listar todos los horarios asociados al sacramento para poder editarlos o eliminarlos.

Gráfico N 45 Pantalla para listar las horas de atención de un sacramento



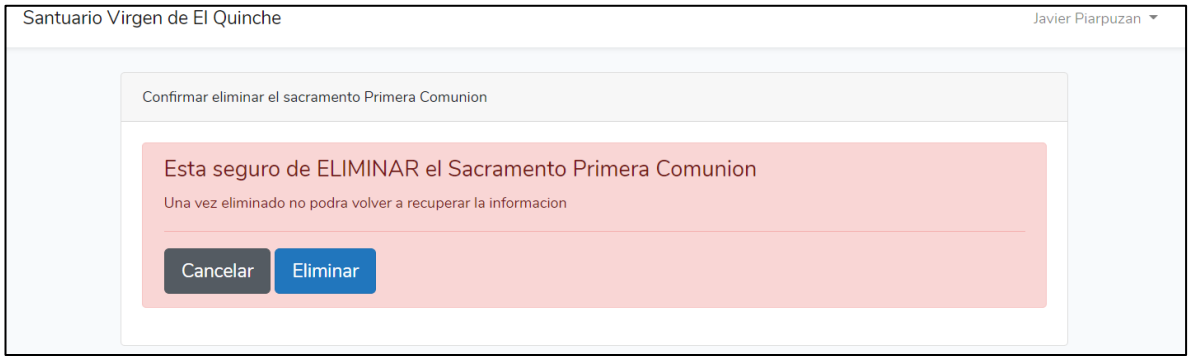
Elaboración: Mariano Navarrete

Descripción: Elaborar una pantalla para la eliminación de un sacramento.

Rol: Administrador

Criterio de aceptación: Se debe indicar si está seguro de eliminar el sacramento y se debe eliminar todos los horarios asociados al sacramento.

Gráfico N 46 Pantalla de confirmación de eliminación de un sacramento



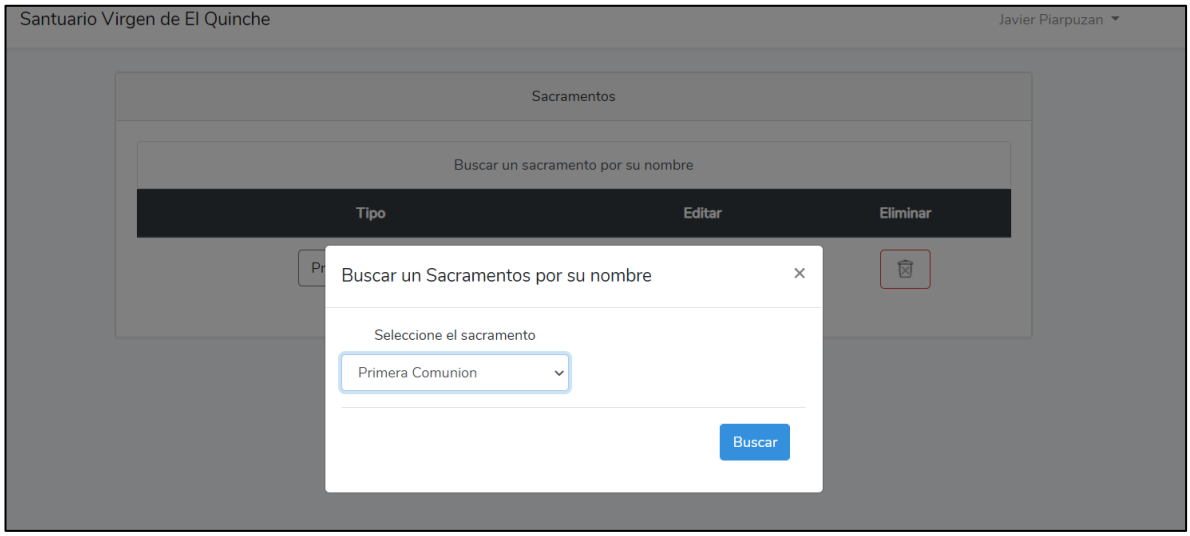
Elaboración: Mariano Navarrete

Descripción: Elaborar un componente para buscar el sacramento.

Rol: Administrador

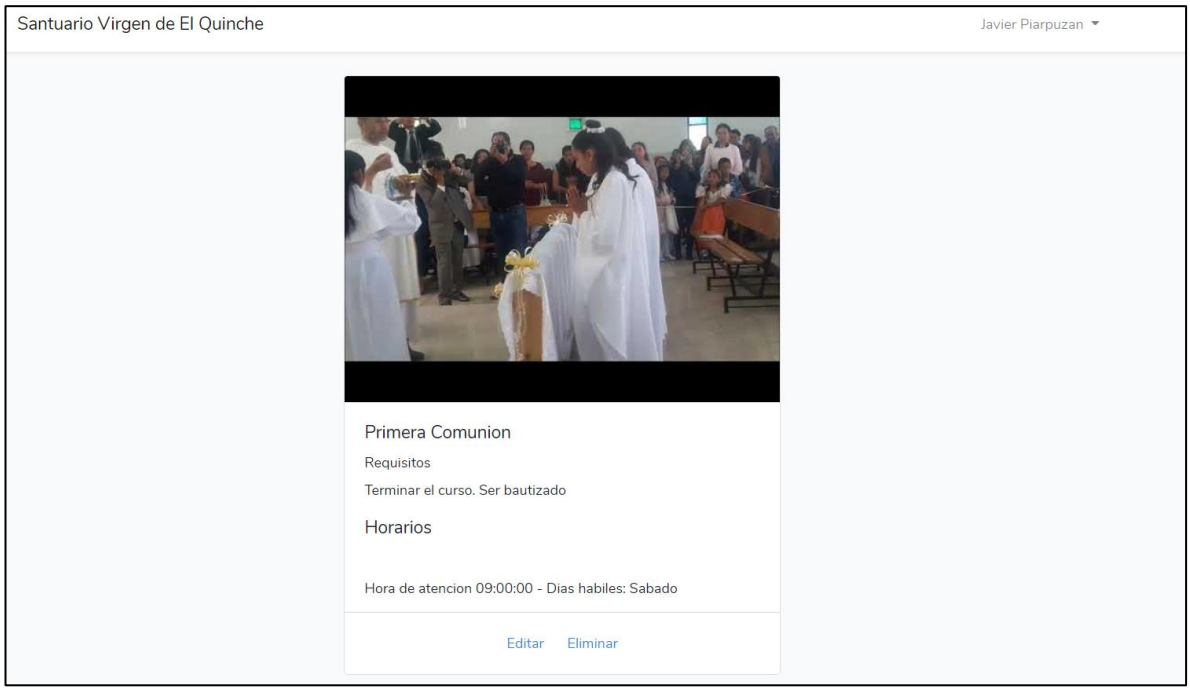
Criterio de aceptación: El componente debe permitir buscar un sacramento por su tipo.

Gráfico N 47 Modal para buscar un sacramento por su tipo



Elaboración: Mariano Navarrete

Gráfico N 48 Pantalla de información de un sacramento



Elaboración: Mariano Navarrete

6.4 SPRINT 4

Descripción: Elaborar una pantalla que muestre la información general de las actividades del Santuario.

Rol: Usuario

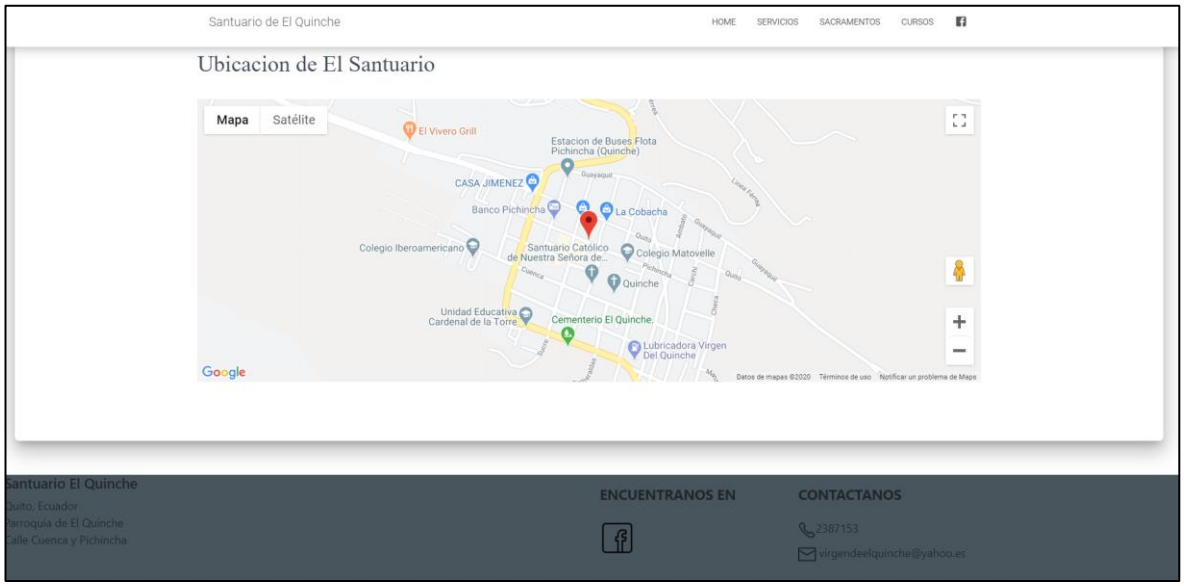
Criterio de aceptación: La pantalla debe tener una barra de navegación que contenga todas las categorías que ofrece el santuario.

Gráfico N 49 Pantalla de encabezado de la página web



Elaboración: Mariano Navarrete

Gráfico N 50 Pantalla de la barra de navegación, pie de la página y mapa de ubicación del Santuario



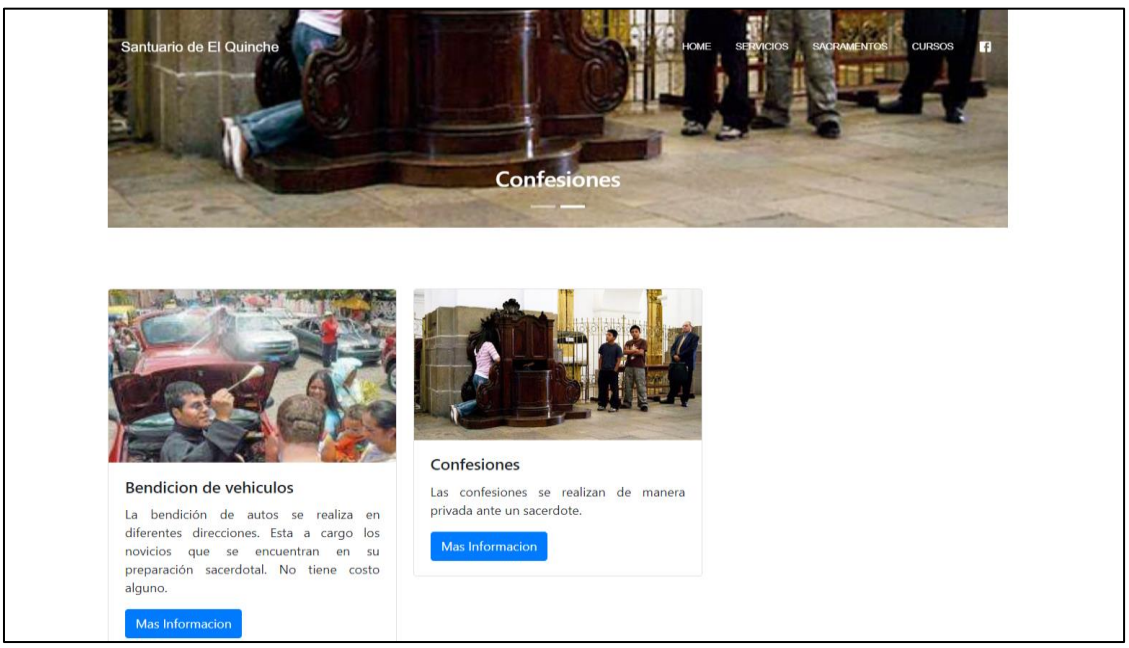
Elaboración: Mariano Navarrete

Descripción: Elaborar una pantalla que muestre una categoría en específico.

Rol: Usuario

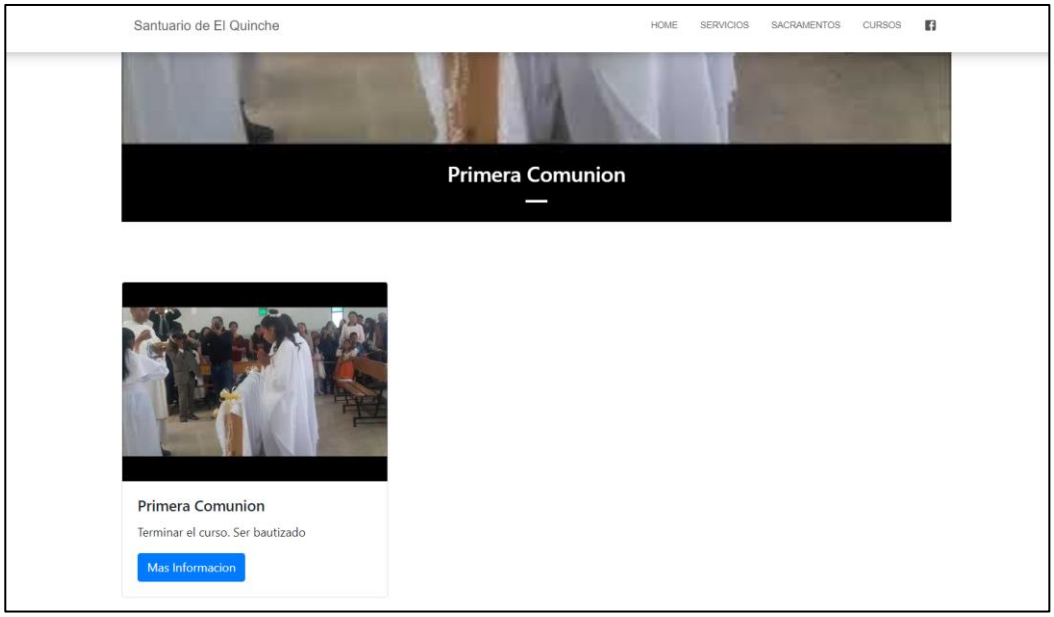
Criterio de aceptación: La pantalla debe contener todos los cursos, servicios o sacramentos ofertados.

Gráfico N 51 Pantalla de información de los servicios ofertados



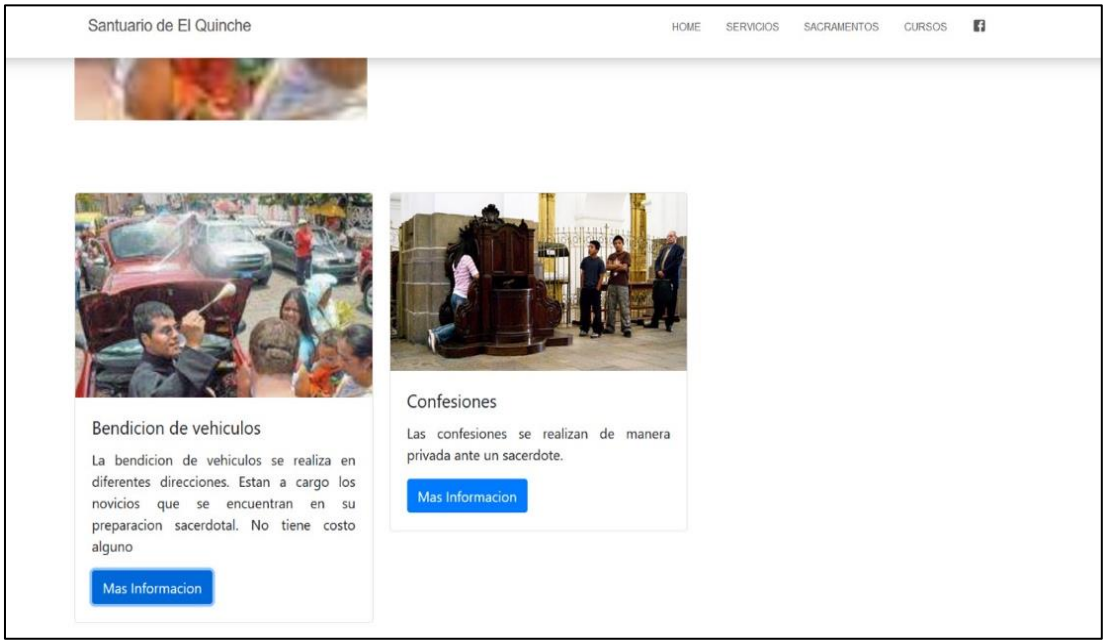
Elaboración: Mariano Navarrete

Gráfico N 52 Pantalla de información de los sacramentos



Elaboración: Mariano Navarrete

Gráfico N 53 Pantalla de información de los cursos



Elaboración: Mariano Navarrete

Descripción: Elaborar una pantalla que muestre un curso, servicio o sacramento.

Rol: Usuario

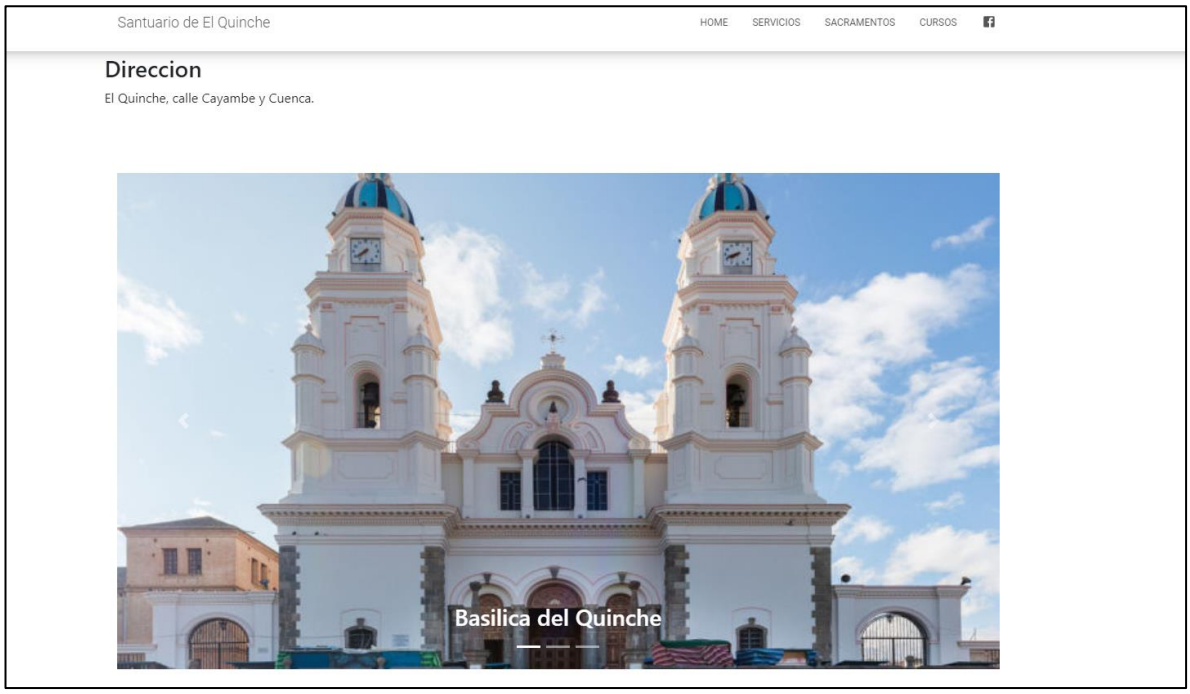
Criterio de aceptación: La pantalla debe mostrar toda la información de un curso, servicio o sacramento.

Gráfico N 54 Pantalla de información de un servicio en específico



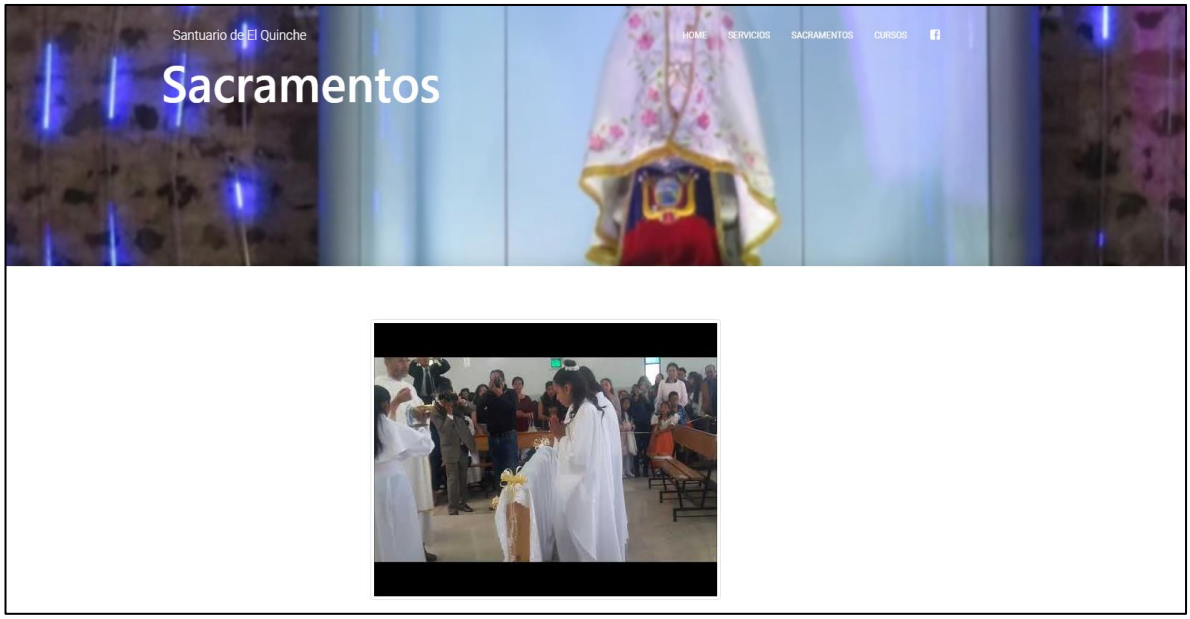
Elaboración: Mariano Navarrete

Gráfico N 55 Pantalla de información de un servicio en específico



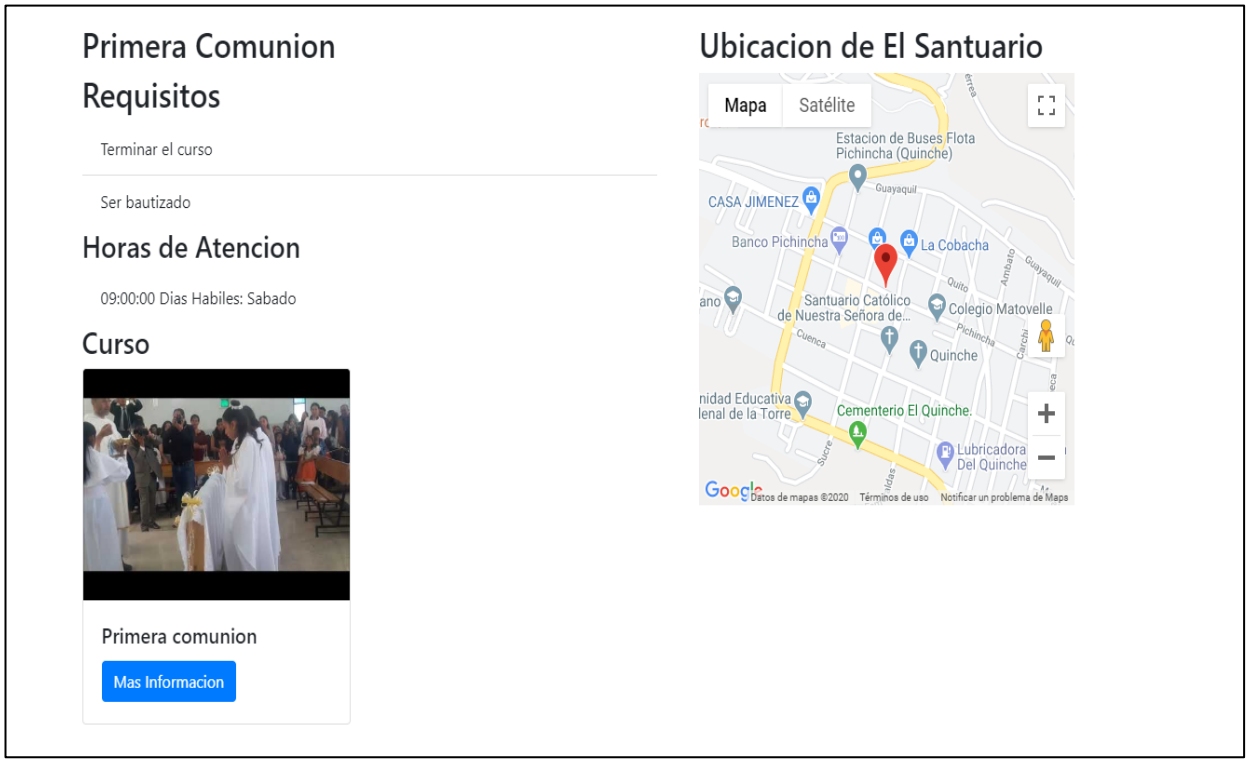
Elaboración: Mariano Navarrete

Gráfico N 56 Pantalla de información de un sacramento en específico



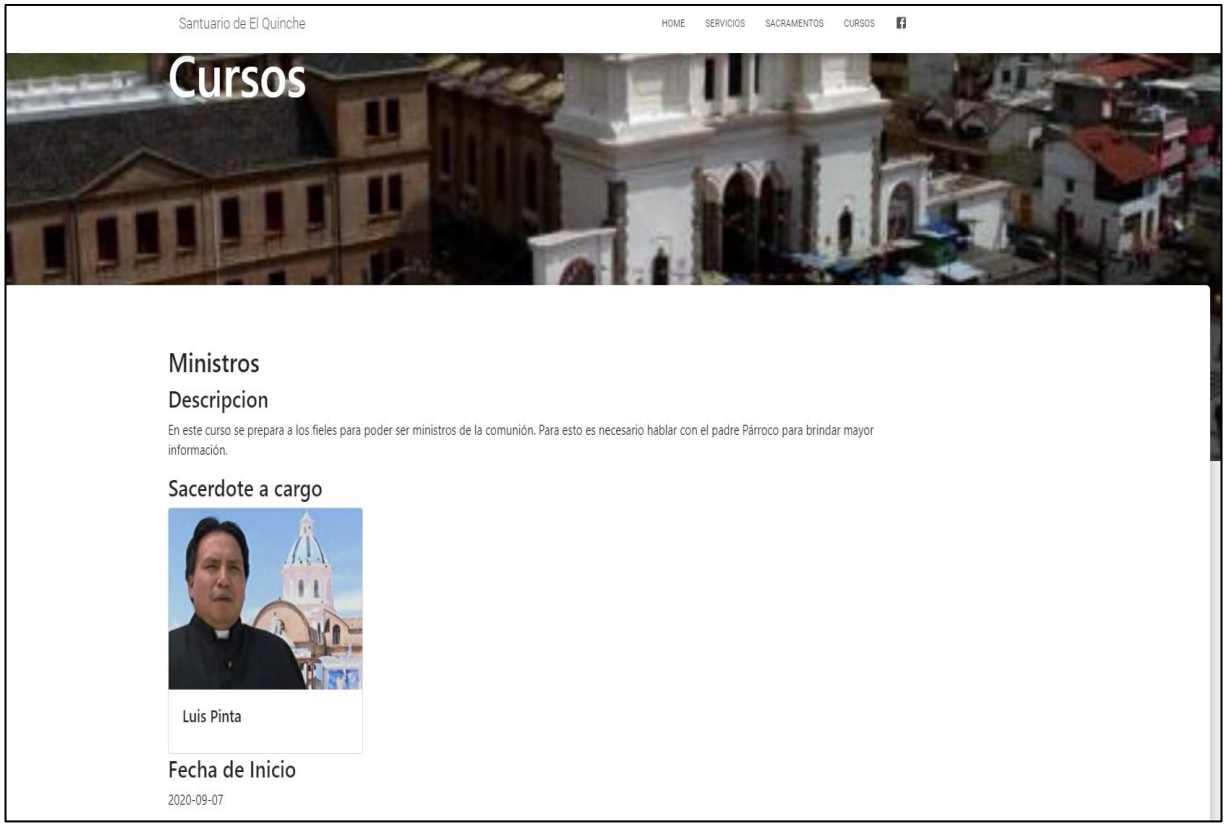
Elaboración: Mariano Navarrete

Gráfico N 57 Pantalla de información de un sacramento en específico



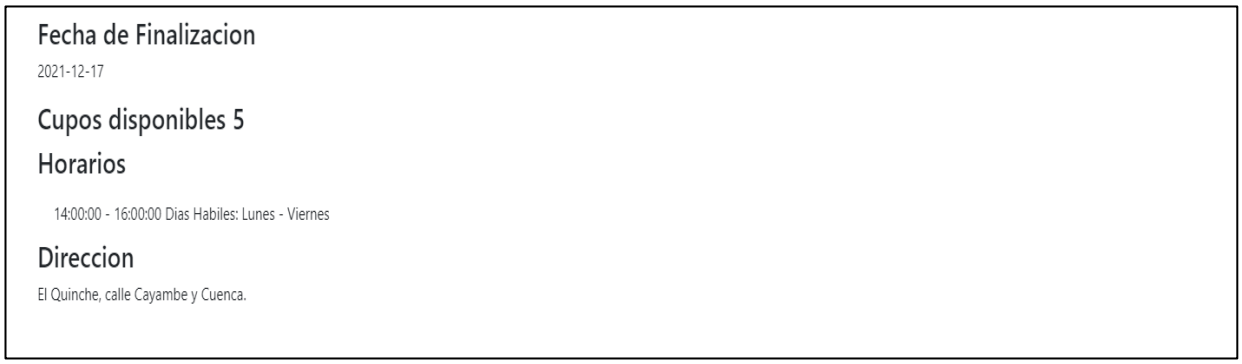
Elaboración: Mariano Navarrete

Gráfico N 58 Pantalla de información de un curso en específico



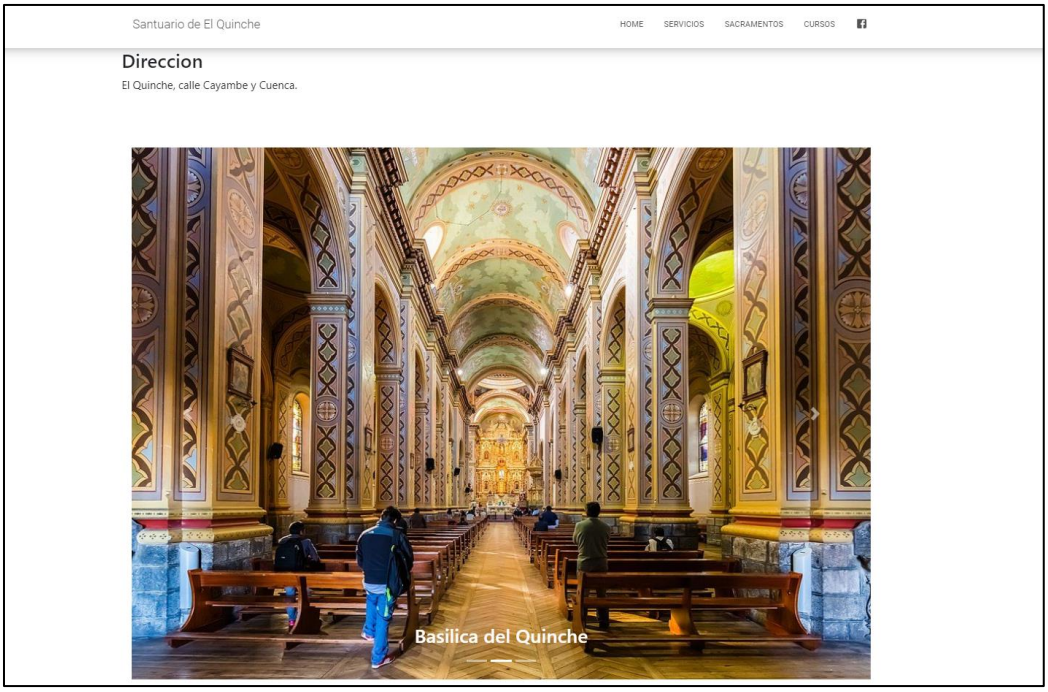
Elaboración: Mariano Navarrete

Gráfico N 59 Pantalla de información de un curso en específico



Elaboración: Mariano Navarrete

Gráfico N 60 Pantalla de información de un curso en específico



Elaboración: Mariano Navarrete

Descripción: Crear un componente para indicar la geolocalización de las instalaciones.

Rol: Usuario

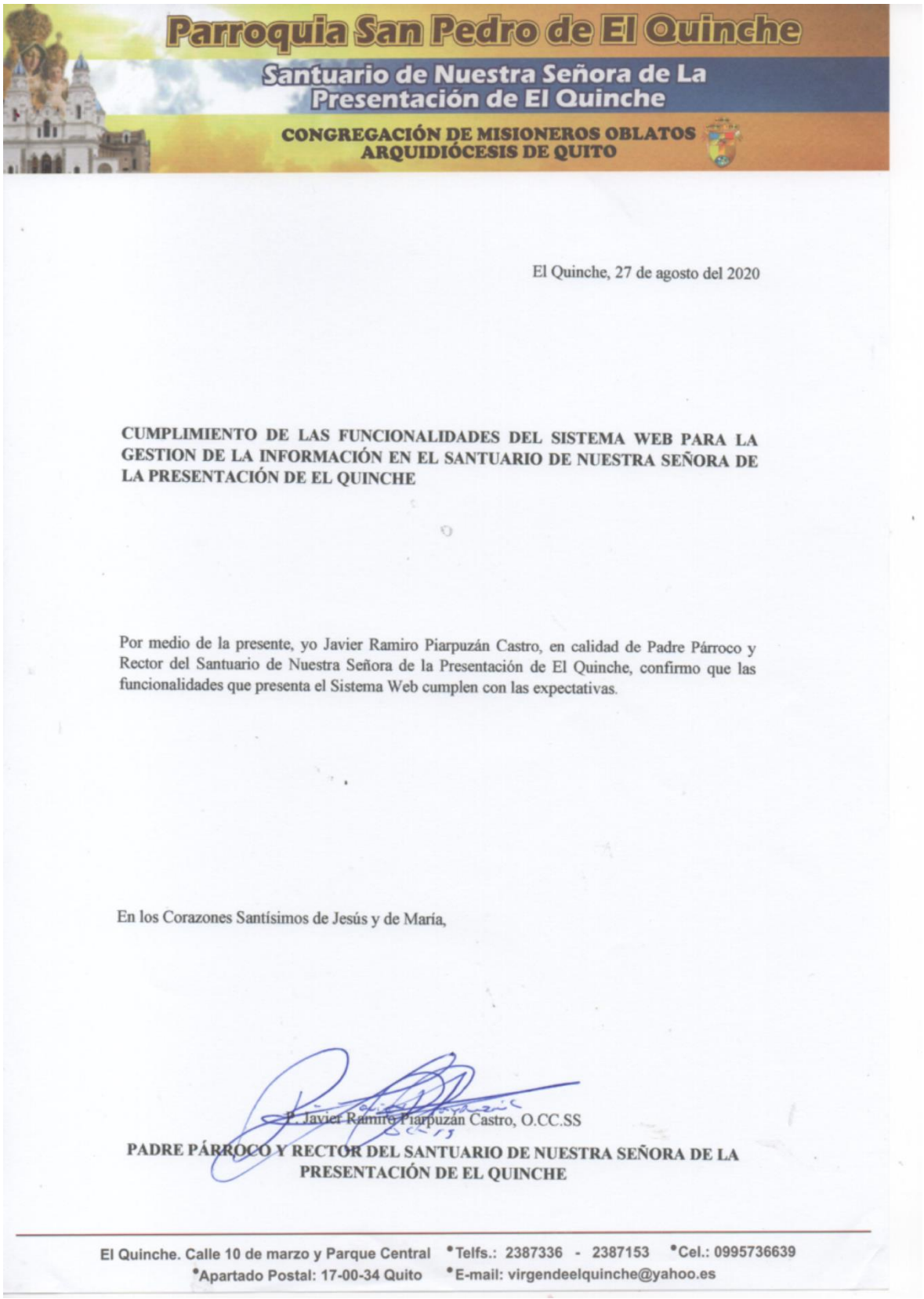
Criterio de aceptación: En el componente se debe visualizar un mapa con el punto de la ubicación real de la instalación a la que se haga referencia.

Gráfico N 61 Componente de ubicación de la instalación



Elaboración: Mariano Navarrrete

Gráfico N 62 Carta de aceptación de las funcionalidades



Parroquia San Pedro de El Quinche

Santuario de Nuestra Señora de La Presentación de El Quinche

CONGREGACIÓN DE MISIONEROS OBLATOS
ARQUIDIÓCESIS DE QUITO

El Quinche, 27 de agosto del 2020

CUMPLIMIENTO DE LAS FUNCIONALIDADES DEL SISTEMA WEB PARA LA GESTION DE LA INFORMACIÓN EN EL SANTUARIO DE NUESTRA SEÑORA DE LA PRESENTACIÓN DE EL QUINCHE

Por medio de la presente, yo Javier Ramiro Piarpuzán Castro, en calidad de Padre Párroco y Rector del Santuario de Nuestra Señora de la Presentación de El Quinche, confirmo que las funcionalidades que presenta el Sistema Web cumplen con las expectativas.

En los Corazones Santísimos de Jesús y de María,

P. Javier Ramiro Piarpuzán Castro, O.CC.SS

PADRE PÁRROCO Y RECTOR DEL SANTUARIO DE NUESTRA SEÑORA DE LA PRESENTACIÓN DE EL QUINCHE

El Quinche. Calle 10 de marzo y Parque Central • Telfs.: 2387336 - 2387153 • Cel.: 0995736639
• Apartado Postal: 17-00-34 Quito • E-mail: virgendeelquinche@yahoo.es

6.5GUÍA PARA LA EJECUCIÓN DEL SISTEMA WEB

En esta guía se indicará la forma de ejecutar el Sistema Web y el prototipo de Página Web. Por consiguiente, se presentan los requisitos que son necesarios para ejecutar el Sistema Web, cabe reclacar que las versiones que se indican son bases y se pueden trabajar con versiones superiores a las mismas.

Por otro lado, los pasos que se mencionan a continuación son los más sencillos para que el Sistema Web funcione de la manera esperada, y para la instalación es necesario tener git como manejador de versiones.

6.5.1 Base de Datos

MySQL versión 8.0.14

6.5.1.1 Pasos necesarios para la base de datos

1. Se debe tener en cuenta las credenciales de la base de datos, puesto que se va a ocupar en pasos posteriores.
2. Crear una base de datos
3. Una vez teniendo en cuenta los pasos 1 y 2 está lista para utilizarse por medio de Laravel, las tablas se crearán con las migraciones de Laravel.

6.5.2 Sistema Web

6.5.2.1 Requisitos

- PHP versión 7.2 o superior
- Composer versión v1.10.10
- Laravel versión 7.4.0
- Git versión 2.28.0

6.5.2.2 Instalación del Sistema Web

1. Clonar el proyecto a la carpeta destino. El proyecto está subido en la git. Para clonar el repositorio se ejecuta el siguiente comando:

   `git clone https://github.com/MarianoNavarrete/Gestion-de-la-infromacion-del-santuraio-de-El-Quinche.git`

   Una vez clonado el repositorio se procede a instalar los paquetes y dependencias con el siguiente comando:

   composer install
2. Luego es necesario clonar el contenido del archivo .env.example

   Para realizar esto realizamos el siguiente comando:

   cp.env.example.env
3. Es necesario generar el APP_KEY, la cual es utilizada por todas las cookies cifradas.

   Para realizar lo antes mencionado se ejecuta el siguiente comando:

   Php artisan key:generate
4. Una vez terminado estos pasos ya está listo y funciona Laravel con la aplicación web.
5. El siguiente paso es colocar las credenciales de las bases de datos en el archivo *.env*
6. Finalmente se ejecuta las migraciones para crear las tablas con los campos necesarios para que se ejcute el Sistema Web.

6.5.3 Prototipo de Página Web

6.5.3.1 Requisitos

- NodeJS versión 12.18.3
- Git versión 2.28.0

6.5.3.2 Instalación del prototipo de Página Web

1. Clonar el proyecto en la carpeta destino. Para clonar el repositorio se ejcua el siguiente comando:

   ```
   git clone https://github.com/MarianoNavarrete/Gestion-de-la-infromacion-del-santuraio-de-El-Quinche.git
   ```

2. Una vez clonado el repositorio se procede a instalar los paquetes y dependencias con el siguiente comando:

   npm install

3. Una vez instalada las dependencias, el prototipo de la Página Web está lista para ser utilizada

7. CONCLUSIONES Y RECOMENDACIONES

7.1 CONCLUSIONES

- El análisis de los medios digitales se realizó con la ayuda del Padre Párroco a cargo del Santuario. Con lo cual se evidenció que la información brindada era escasa e inclusive en algunos casos estaba desactualizada.
- Las herramientas que se pueden utilizar para desarrollar un Sistema Web son muy variadas y cada una de dichas herramientas ofrecen características que se acoplan a cada proyecto. Razón por la cual, es necesario realizar un análisis de las herramientas a utilizar para sacar el mayor beneficio posible y evitar complicaciones el momento de desarrollar el Sistema Web.
- El desarrollo del Sistema Web utilizó la metodología de desarrollo ágil Scrum, ya que gracias a esta metodología el Padre Párroco pudo estar involucrado en el sistema y verificaba si las funcionalidades desarrolladas eran las correctas.
- El diseño de la base de datos se basó en una base de datos relacional, dado que, la información que tiene el Santuario no crece en grandes cantidades y la información debe ser consistente.
- Para desarrollar el Sistema Web se utilizaron las tecnologías previamente analizadas de React para el front-end y Laravel para el back-end, así mismo el desarrollo estuvo divido en 4 sprints con un tiempo estimado de 154 horas.
- La ubicación de los distintos lugares que posee el Santuario fue construida utilizando la tecnología de Google Maps, con lo cual se puede visualizar un mapa que está continuamente siendo mejorado y se colocaron marcadores como puntos referenciales de la ubicación.

- Las pruebas fueron realizadas bajo el criterio de aceptación ubicadas en el Product Backlog, gracias a estos criterios se pudo decir que el Sistema Web cumple con todas las características previstas.

7.2 RECOMENDACIONES

- Si de decide subir a producción el Sistema Web, se recomienda utilizar proveedores que soporten subidas de demanda de la aplicación web en determinadas fechas, por la alta demanda que se provee en las fiestas religiosas del Santuario.
- Para la creación de nuevas funcionalidades o el mejoramiento de las funcionalidades desarrolladas, se recomienda seguir con los estándares con los cuáles el sistema fue desarrollado.
- Para el mantenimiento del Sistema Web, se recomienda utilizar la misma versión del framework con la que fue construido y actualizar las librerías utilizadas.
- Para el correcto funcionamiento del Sistema Web, se recomienda que el Santuario cuente con un proveedor de internet confiable, puesto que dicho sistema se encontrará en la red.

8. BIBLIOGRAFIA

"Introducción a VueJS" (2019) https://codingpotions.com/que-es-vue [Consulta: 15/04/2020]

Acosta, Víctor Manuel (2020) ***Frontend con Angular: Todo lo que debes saber sobre esta herramienta.*** https://revistadigital.inesem.es/informatica-y-tics/frontend-con-angular-todo-lo-que-debes-saber-sobre-esta-herramienta/ [Consulta: 06/04/2020]

Álvarez, Raúl (2020) ***El "Código Espagueti" y los patrones avanzados de programación.*** https://www.tecbolivia.com/index.php/articulos-y-tutoriales-microcontroladores/12-el-qcodigo-espaguetiq-y-los-patrones-avanzados-de-programacion [Consulta: 20/04/2020]

Álvarez, Miguel Ángel (2019) ***Qué es React. Por qué usar React.*** https://desarrolloweb.com/articulos/que-es-react-motivos-uso.html [Consulta: 08/04/2020]

Álvarez, Miguel Ángel (2020) ***Qué es MVC.*** https://desarrolloweb.com/articulos/que-es-mvc.html [Consulta: 21/07/2020]

Antón Dorantes, César (2015) ***Laravel, el mejor framework en PHP.*** https://platzi.com/blog/laravel-framework-php/ [Consulta: 19/04/2020]

Arias, Diego (2019) ***10 cosas que debes saber cobre Ruby on Rails.*** http://blog.desafiolatam.com/10-respuestas-a-ruby-on-rails/ [Consulta: 22/04/2020]

Arimetrics (2020) ***Backend.*** https://www.arimetrics.com/glosario-digital/backend [Consulta: 15/04/2020]

Arjonilla, Rafa (2020) ***BackEnd.*** https://rafarjonilla.com/que-es/backend/ [Consulta: 15/04/2020]

Baquero García, José María (2015) ***¿Qué es Laravel***? https://www.arsys.es/blog/programacion/que-es-laravel/ [Consulta: 19/04/2020]

Campus MVP (2015) ***Desarrollador web: Front-end, back-end y full stack. ¿Quién es quién?*** https://www.campusmvp.es/recursos/post/Desarrollador-web-Front-end-back-end-y-full-stack-Quien-es-quien.aspx [Consulta: 15/04/2020]

Campus MVP (2017) ***Las 5 principales ventajas de usar Angular para crear aplicaciones web.*** https://www.campusmvp.es/recursos/post/las-5-principales-ventajas-de-usar-angular-para-crear-aplicaciones-web.aspx [Consulta: 04/04/2020]

Cangás Cángas, Fernando Xavier (2015) ***Sistema Web transur con Node.Js para la gestión de transporte de la Cooperativa de transporte de pasajeros inter cantonal Urcuquí.*** (Disertación de grado), Universidad Técnica del Norte, Ecuador. http://repositorio.utn.edu.ec/handle/123456789/4630 [Consulta: 23/04/2020]

Canive, Teresa y Balet, Richard (2020) ***SCRUM.*** https://www.sinnaps.com/blog-gestion-proyectos/metodologia-scrum [Consulta: 31/07/2020]

Cavana, Judith (2017) ***¿Cómo funciona Node.js?.*** https://www.drauta.com/que-es-nodejs-y-para-que-sirve [Consulta: 23/04/2020]

Chapaval, Nicol (2018) ***Qué es Frontend y Backend.*** https://platzi.com/blog/que-es-frontend-y-backend/ [Consulta: 15/04/2020]

CLE (2017) ***Ruby on Rails (RoR), Desarrollo web rápido, fácil y potente.*** http://www.cleformacion.com/tic-tek/-/blogs/248776 [Consulta: 22/04/2020]

Clip, Sara (2017) ***Eventos en Scrum I.*** https://www.saraclip.com/eventos-en-scrum/ [Consulta: 31/07/2020]

De Ipiña, Diego (2006) ***Introducción a Ruby on Rails.*** http://paginaspersonales.deusto.es/dipina/ruby/RubyOnRailsDeusto.pdf [Consulta: 09/04/2020]

Desarrollo Web (2020) ***Laravel.*** https://desarrolloweb.com/home/laravel [Consulta: 19/04/2020]

Dongil Sánchez, José Antonio (s.f.) ***Por qué elegir VueJS: 5 razones para considerarlo nuestro framework de referencia.*** https://www.genbeta.com/desarrollo/por-que-elegir-vuejs-5-

razones-para-considerarlo-nuestro-proximo-framework-de-referencia [Consulta: 15/04/2020]

Ecuavisa (2014) ***Hace 29 años el Papa Juan Pablo II visitó la capital ecuatoriana.*** https://www.ecuavisa.com/articulo/canonizacion/noticias/59317-hace-29-anos-papa-juan-pablo-ii-visito-capital-ecuatoriana [Consulta: 07/11/2019]

EcuRed (2020) ***Aplicación web.*** https://www.ecured.cu/Aplicación_web [Consulta: 10/07/2020]

Escuela Tecnologías de la Información S.L (2020) ***Frontend. El ámbito que tiene que ver con la presentación.*** https://escuela.it/materias/frontend [Consulta: 27/03/2020]

Espitia, Nilsem, Armao, Oscar y Carbajo, Jhonnathan (2016) ***MODELO VISTA-CONTROLADOR (MVC).*** (Disertación de grado), Universidad Alejandro de Humbolt, Venezuela https://espejodeantagno.files.wordpress.com/2016/04/modelo-vista-controlador-mvc.pdf [Consulta: 21/07/2020]

Falcón, Carlos Arturo (2014) ***Explicando qué es Front-End, qué es Back-End y sus características.*** http://www.falconmasters.com/web-design/que-es-front-end-y-que-es-back-end/ [Consulta: 15/04/2020]

Fernández Romero, Yenisleidy y Díaz González, Yanette (2012) ***Patrón Modelo-Vista-Controlador.*** *Telem@tica, 11* (1), (47-57). http://revistatelematica.cujae.edu.cu/index.php/tele/article/view/15/10 [Consulta: 21/07/2020]

Francia, Joel (2019) ***Empirismo.*** https://www.scrum.org/resources/blog/empirismo#:~:text=Los%20tres%20pilares%20de%20Scrum,la%20inspección%2C%20adaptación%20y%20transparencia.&text=La%20inspección%20se%20puede%20dar,interesados%20y%20la%20organización%20misma. [Consulta: 31/07/2020]

García Pérez, Eduardo Ismael (2019) ***¿Qué es Vue.JS?*** https://codigofacilito.com/articulos/que-es-vue [Consulta: 12/04/2020]

García, David (2016) ***Características de React.*** https://desarrolloweb.com/articulos/caracteristicas-react.html [Consulta: 08/04/2020]

García, Miriam (2017) ***MVC (Modelo-Vista-Controlador): ¿qués es y para qué sirve?*** https://codingornot.com/mvc-modelo-vista-controlador-que-es-y-para-que-sirve [Consulta: 21/07/2020]

Garza, Juan Carlos (2017) ***Desarrolla tu primera página web con Ruby on Rails, parte 0.5.*** https://medium.com/devsteps-mx/desarrolla-tu-primera-p%C3%A1gina-web-con-ruby-on-rails-parte-0-5-8a20ded222aa [Consulta: 22/04/2020]

Guachamín Lara, Verónica Estefanía y Tello Gorozabel, María Dolores (2017) ***Análisis de la influencia del Santuario de la Virgen de El Quinche como eje de desarrollo turístico local.*** (Tesis de grado) Escuela Politécnica del Ejército, Ecuador. https://repositorio.espe.edu.ec/handle/21000/13772 [Consulta: 07/11/2019]

Guerrero Benalcázar, Rubén Ignacio (2016) ***Estudio comparativo de los frameworks Ruby on Rails y Django para la implementación de un sistema informático de control y administración de Network Marketing.*** (Disertación de grado), Universidad Técnica del Norte, Ecuador.

Guerrero, N (2018) ***Ventajas que se tiene al desarrollar en Node.js*** https://programaenlinea.net/ventajas-que-se-tiene-al-desarrollar-en-node-js/ [Consulta: 23/04/2020]

Hostalia (2016) ***Laravel, un framework de PHP.*** https://pressroom.hostalia.com/white-papers/laravel-framework-php/ [Consulta: 20/04/2020]

Ictea (2020) ***¿Qué es una aplicación web?*** https://www.ictea.com/cs/index.php?rp=%2Fknowledgebase%2F4205%2FWhat-is-a-web-application.html [Consulta: 12/07/2020]

Iglesias Católicas (2017) ***Santuario de El Quinche***. https://iglesias-catolicas.com/es/iglesia/el-quinche-santuario-de-el-quinche [Consulta: 07/11/2019]

ITDO (2019) ***React. El framework que revolucionó la web.*** https://www.itdo.com/blog/react-el-framework-que-revoluciono-la-web/ [Consulta: 08/04/2020]

Lara, Walter (2015) ***¿Cómo funciona la metodología Scrum?*** https://platzi.com/blog/metodologia-scrum-fases/ [Consulta: 31/07/2020]

Laravel (s.f.) ***Libro Laravel 5. Conceptos básicos y ejemplos.*** https://www.pdf-manual.es/programacion-web/php/175-libro-laravel-5-conceptos-basicos-y-ejemplos.html [Consulta: 19/04/2020]

Lucas, Jesús (2019) ***Qué es NodeJS y para qué sirve.*** https://openwebinars.net/blog/que-es-nodejs/ [Consulta: 19/04/2020]

Menzinsky, Alexander, López, Gertrudis y Palacio, Juan (2016) ***Scrum Manager.*** Info 4 Media SL. https://www.scrummanager.net/files/sm_proyecto.pdf [Consulta: 31/07/2020]

Misioneros Oblatos (2019) ***Nuestra Señora del Quinche 21 de noviembre.*** https://www.oblatos.com/nuestra-senora-del-quinche/ [Consulta: 07/11/2019]

Mobile Market Association (2016) ***All you need to know regarding Backend.*** https://www.mmaglobal.com/news/todo-lo-que-necesitas-saber-sobre-backend-all-you-need-know-regarding-backend [Consulta: 15/04/2020]

Morales, Alejandro (2012) ***Manual de NodeJS.*** https://desarrolloweb.com/manuales/manual-nodejs.html [Consulta: 24/04/2020]

Neosoft (2018) ***¿Qué es un apalicación Web?*** https://www.neosoft.es/blog/que-es-una-aplicacion-web/ [Consulta: 12/07/2020]

Ojeda Valiente, Walter (2018) ***Diferencias entre Angular y AngularJS.*** https://vfpavanzado.wordpress.com/tag/inteligente/ [Consulta: 06/04/2020]

Ortega Delgado, Daniel (2017) ***Introducción a Ruby on Rails.*** https://openwebinars.net/blog/introduccion-ruby-rails/ [Consulta: 22/04/2020]

Pérez Porto, Julián y Merino, María (2015) ***Definición de Plugin.*** https://definicion.de/plugin/ [Consulta: 15/04/2020]

Platzi (2020) ***Desarrollo Web Frontened con Vue.js.*** https://platzi.com/vue/ [Consulta: 12/04/2020]

Puciarelli, Luciano (2020) ***Angular. TypeScript-Arquitectura-Instalación-Directivas y Bindings-Forms-Ruteo y más.*** Six Ediciones: Buenos Aires. https://books.google.com.ec/books?id=6aHTDwAAQBAJ&pg=PA3&lpg=PA3&dq=que+es+typescript.+puciarelli&source=bl&ots=MGdPmWGRMA&sig=ACfU3U3wJLhpG_ZzkbA7TMdpxvKPBpVo0w&hl=es&sa=X&ved=2ahUKEwjv67iRuqXpAhUiVt8KHeAhDeYQ6AEwAHoECAoQAQ#v=onepage&q=que%20es%20typescript.%20puciarelli&f=false [Consulta: 04/04/2020]

Roche, Julio (2020) ***Artefactos Scrum: las 3 herramientas clave de gestión.*** https://www2.deloitte.com/es/es/pages/technology/articles/artefactos-scrum.html#:~:text=En%20el%20marco%20de%20trabajo,Sprint%20Backlog%20y%20el%20Incremento. [Consulta: 31/07/2020]

Roche, Julio (2020) ***Las 5 ceremonias Scrum: claves para la gestión de procesos.*** https://www2.deloitte.com/es/es/pages/technology/articles/ceremonias-scrum.html [Consulta: 31/07/2020]

Rojas Parra, Juan Pablo (2019) ***Lenguajes, frameworks y librerías Backend para 2019.*** https://platzi.com/blog/lenguajes-frameworks-librerias-backend-2019/ [Consulta: 15/04/2020]

Rubyonrails (2020) ***Ruby on Rails: El Desarrollo web que no molesta***. http://rubyonrails.org.es/ [Consulta: 22/04/2020]

Rubyonrails (2020) ***Ruby on Rails.*** https://tutorialesenpdf.com/ruby-on-rails/ [Consulta: 22/04/2020]

Schwaber, Ken y Sutherland, Jeff (2013) ***La Guía Definitiva de Scrum: Las Reglas del Juego.*** https://www.scrumguides.org/docs/scrumguide/v1/scrum-guide-es.pdf [Consulta: 31/07/2020]

SCRUMStudy (2013) ***Una guía para el CONOCIMIENTO DE SCRUM.*** Phoenix: VMEdu, Inc. https://www.tenstep.ec/portal/images/pdfs/Suscripciones_TenStep/Silver/SCRUMstudy_GUIA_SBOK_espanol.pdf [Consulta: 31/07/2020]

Universidad de Alicante (2020) ***Modelo Vista Controlador (MVC).*** https://si.ua.es/es/documentacion/asp-net-mvc-3/1-dia/modelo-vista-controlador-mvc.html [Consulta: 21/07/2020]

Uniwebsidad (2020) ***Capítulo 2. ¿Qué es Rails?*** https://uniwebsidad.com/libros/introduccion-rails/capitulo-2 [Consulta: 22/04/2020]

Valdivia Caballero, José Jesús (2016) ***Modelo de procesos para el desarrollo del Front-end de aplicaciones Web.*** Lima file:///Users/cathynavarrete/Downloads/Dialnet-ModeloDeProcesosParaElDesarrolloDelFrontendDeAplic-6043088%20(1).pdf [Consulta: 27/03/2020]

Vuejs (2020) ***Introducción. Qué es Vue.js?.*** https://es.vuejs.org/v2/guide/ [Consulta: 12/04/2020]