

ESCUELA POLITÉCNICA NACIONAL

ESCUELA DE FORMACIÓN DE TECNÓLOGOS

DESARROLLO DE UNA APLICACIÓN MÓVIL HÍBRIDA MEDIANTE REALIDAD AUMENTADA PARA POTENCIAR EL TURISMO EN LA CIUDAD DE QUITO

PROYECTO PREVIO A LA OBTENCIÓN DEL TÍTULO DE TECNÓLOGO EN ANÁLISIS DE SISTEMAS INFORMÁTICOS

JOSELYN SILVANA RODRÍGUEZ MEJÍA

josselyn.rodriguez@epn.edu.ec

HENRY DANIEL ROJAS CISNEROS

henry.rojas@epn.edu.ec

DIRECTOR: Ing. Juan Pablo Zaldumbide, MSc.

juan.zaldumbide@epn.edu.ec

Quito, noviembre 2017

DECLARACIÓN

Nosotros Rodríguez Mejía Josselyn Silvana y Rojas Cisneros Henry Daniel declaramos bajo juramento que el trabajo aquí descrito es de nuestra autoría; que no ha sido previamente presentada para ningún grado o calificación profesional; y, que hemos consultado las referencias bibliográficas que se incluyen en este documento.

A través de la presente declaración cedemos nuestros derechos de propiedad intelectual correspondientes a este trabajo, a la Escuela Politécnica Nacional, según lo establecido por la ley de propiedad intelectual, por su reglamento y por la normatividad institucional vigente.

Rodríguez Mejía Josselyn Silvana

Rojas Cisneros Henry Daniel

CERTIFICACIÓN

Certifico que el presente trabajo fue desarrollado por la señorita Rodríguez Mejía Josselyn Silvana y el señor Rojas Cisneros Henry Daniel, bajo nuestra supervisión.

Ing. Juan Pablo Zaldumbide, MSc

DIRECTOR DE PROYECTO

PhD. María Pérez

CODIRECTORA DE PROYECTO

AGRADECIMIENTO

A mi madre, a mi abuelo materno que lo considero mi padre, y a mi familia por el apoyo incondicional, pero sobre todo agradezco a Dios por acompañarme en este duro camino de formación profesional.

A mis amigos con quienes compartí experiencias, alegrías, frustraciones, celebraciones y muchas cosas más, pero sobre todo agradezco a mi gran amigo Henry con quien aprendí a estar en las buenas y en las malas, como dice él nos complementamos para investigar y sacar adelante un proyecto.

También agradezco a todos los profesores y en especial al director de tesis, quienes nos brindaron los conocimientos necesarios para la obtención de este título.

RODRÍGUEZ MEJÍA JOSSELYN SILVANA

AGRADECIMIENTO

Primero y como más importante, quiero agradecer a mis padres que estuvieron apoyándome durante todo este tiempo en mi carrera universitaria, con lo poco o mucho que tenían para ayudarme me hicieron la persona que soy, además a mis abuelitos que siempre estuvieron preocupados por mi familia y por mí en cada etapa de mi vida. Y principalmente quiero decirle a mi papá que ¡Lo logre!, a pesar de que un día tu dijiste que tal vez no podría estar al nivel, pero hoy te demostré todo lo contrario.

Doy gracias a mis amigos con los cuales viví grandes experiencias y un montón de ocurrencias en la universidad aprendiendo y divirtiéndome, a mi compañera y amiga Josselyn en serio tú eres mi complemento para todo y siempre recordaré todas y cada una de nuestras experiencias, vividas en cada proyecto que hicimos juntos y como crecíamos al realizarlos a pesar de tus enojos conmigo, siempre lo dimos todo en cada proyecto.

Finalmente agradezco a todos y cada uno de mis profesores que me brindaron sus conocimientos, y a los cuales considero mis amigos porque supieron ayudarme cuando más necesitaba y en especial a mi director que fue la guía para hacer posible este proyecto.

ROJAS CISNEROS HENRY DANIEL

DEDICATORIA

A mi madre por apoyarme e impulsarme para salir adelante, a pesar del difícil carácter que tengo nunca se dio por vencida.

Finalmente, recordar es vivir y vivir es reír, por tal motivo dedico este trabajo a los que me consideran su verdadera amiga ya que siempre me recordaran con una sonrisa en sus rostros.

RODRÍGUEZ MEJÍA JOSSELYN SILVANA

DEDICATORIA

A mis padres y abuelitos que son las personas más importantes de mi vida los cuales me ayudaron, me dieron todo su cariño y sus enseñanzas para llegar a ser la persona que soy.

Y finalmente, dedico este trabajo a todas y cada una de las personas que creyeron en mí y que, a pensar de mi forma de ser sabían que este día llegaría y que sabían que podía y puedo llegar lejos con mis conocimientos y mis destrezas adquiridas gracias a ellos y a toda una vida de sacrificios.

ROJAS CISNEROS HENRY DANIEL

TABLA DE CONTENIDO

DECLARACIÓN	I
CERTIFICACIÓN	II
AGRADECIMIENTO.....	III
AGRADECIMIENTO.....	IV
DEDICATORIA.....	V
DEDICATORIA.....	VI
TABLA DE CONTENIDO.....	VII
ÍNDICE DE FIGURAS	XI
ÍNDICE DE TABLAS	XIII
RESUMEN	XIV
ABSTRACT	XV
CAPÍTULO I	1
1. INTRODUCCIÓN	1
1.1 Planteamiento del problema.....	1
1.2 Objetivo General	2
1.3 Objetivos Específicos.....	2
1.4 Alcance	2
CAPÍTULO II	4
2. METODOLOGÍA	4
2.1 Metodología de desarrollo ágil Scrum	4
2.1.1 Roles.....	4
2.1.2 Artefactos.....	5
2.1.3 Actividades	6
2.2 Herramientas para el acceso de la base de datos.....	9
2.2.1 Sails.....	9
2.2.2 Base de Datos en Sails.....	9
2.2.3 CodeAnyWhere.....	10

2.3	Herramientas de desarrollo web.....	10
2.3.1	Html	10
2.3.2	Css	10
2.3.3	JavaScript	10
2.3.4	JQuery	10
2.3.5	Bootstrap	10
2.3.6	AngularJS	11
2.4	Herramientas de desarrollo móvil.....	12
2.4.1	Unity	12
2.4.2	SDK Vuforia	12
2.4.3	SDK Android	12
2.4.4	Xcode.....	13
2.5	Justificación de tecnologías	13
2.5.1	Framework de desarrollo.....	13
2.5.2	SDK para realidad aumentada	14
2.5.3	Base de datos para image target	15
2.6	Requerimientos hardware	16
CAPÍTULO III		17
3.	RESULTADOS Y DISCUSIÓN.....	17
3.1	Requerimientos generales de la página web.....	17
3.1.1	Perspectiva de la página web	17
3.1.2	Funciones de la página web.....	17
3.1.3	Características del usuario de la página web	18
3.1.4	Restricciones de la página web.....	18
3.2	Requerimientos específicos de la página web.....	18
3.2.1	Funcionalidades de la página web	18
3.2.2	Especificaciones de caso de uso de la página web.....	19
3.2.3	Clasificación de requerimientos funcionales de la página web	20
3.3	Requerimientos de ambiente de desarrollo de la página web	20

3.4	Requerimientos generales de la aplicación móvil.....	20
3.4.1	Perspectiva de la aplicación móvil.....	20
3.4.2	Funciones de la aplicación móvil.....	21
3.4.3	Características del usuario de la aplicación móvil	21
3.4.4	Restricciones de aplicación móvil.....	21
3.5	Requerimientos específicos de la aplicación móvil.....	22
3.5.1	Funcionalidades de la aplicación móvil	22
3.5.2	Especificaciones de caso de uso de aplicación móvil.....	22
3.5.3	Clasificación de requerimientos funcionales de la aplicación móvil	23
3.6	Requerimientos de ambiente de desarrollo para la aplicación móvil	23
3.7	Conformación del equipo de trabajo.....	24
3.8	Actividades	24
3.8.1	Historias de usuario	24
3.8.2	Iteraciones	29
3.9	Creación del servidor Sails.....	34
3.9.1	Creación del modelo en Sails.....	35
3.10	Desarrollo de la página de administración de sitios turísticos.....	36
3.10.1	Creación de la página iniciar sesión.....	37
3.10.2	Creación de la página de administración.....	37
3.10.3	Subir página de administración al servidor cloud 9.....	39
3.11	Desarrollo de la aplicación móvil	43
3.11.1	Creación de códigos QR	43
3.11.2	Creación de base de datos para image target.....	47
3.11.3	Obtención de SDKs para la aplicación	51
3.11.4	Creación de proyecto en Unity	52
3.12	Pruebas de la página de administración.....	63
3.12.1	Pruebas de software	63
3.12.2	Pruebas de aceptación	63
3.13	Pruebas de aplicación móvil.....	66

3.13.1 Pruebas de software	66
3.13.2 Pruebas de aceptación	66
CAPÍTULO IV.....	69
4. CONCLUSIONES Y RECOMENDACIONES.....	69
4.1 Conclusiones	69
4.2 Recomendaciones	70
REFERENCIAS BIBLIOGRÁFICAS	71
ANEXO 1: Prototipo página de administración	73
ANEXO 2: Manual de Usuario de la página de administración	75
ANEXO 3: Prototipo de la aplicación móvil.....	80
ANEXO 4: Códigos QR de los 50 sitios turísticos.....	82
ANEXO 5: Ejemplo de utilización de la aplicación móvil.....	91

ÍNDICE DE FIGURAS

Figura 1 Ejemplo de gráfica “Release burn-down”	6
Figura 2 Ejemplo de gráfica “Sprint burn-down”	6
Figura 3 Release en scrum	7
Figura 4 Componentes que se incluyen en un modelo en Sails	9
Figura 5 Modelo Vista Controlador en AngularJS	11
Figura 6 Diagrama de caso de uso de la página web	19
Figura 7 Diagrama de caso de uso de la aplicación móvil	22
Figura 8 Creación del servidor	35
Figura 9 Creación API en Sails	35
Figura 10 Creación del modelo de sitios turísticos en Sails	36
Figura 11 Página de inicio de sesión	37
Figura 12 Página de administración de sitios turísticos sección registrar sitio	38
Figura 13 Lista de sitios turísticos registrados	38
Figura 14 Modal para modificar datos	39
Figura 15 Modal de verificación para eliminar un sitio turístico	39
Figura 16 Crear un workspace para la página de administración	40
Figura 17 Colocar datos para crear el workspace	41
Figura 18 Abrir el workspace creado	41
Figura 19 Subir archivos y carpetas de la página de administración	42
Figura 20 Ejecutar el archivo login.html	42
Figura 21 Página ejecutada correctamente.	43
Figura 22 Estructura básica de un código QR	44
Figura 23 Esquina para el tipo de sitio iglesias	45
Figura 24 Código QR Basílica del Voto Nacional	45
Figura 25 Esquina para el tipo de sitio museos	45
Figura 26 Código QR Museo Aeronáutico y del espacio FAE	46
Figura 27 Esquina para el tipo de sitio parques	46
Figura 28 Código QR Parque Cumandá	46
Figura 29 Esquina para el tipo de sitios otros	47
Figura 30 Código QR Casa Cultural Trude Sojka	47
Figura 31 Creación de un proyecto en la cuenta de desarrollador de Vuforia	48
Figura 32 Sección Target Manager	48
Figura 33 Creación cloud data base	49
Figura 34 Ingresar image target a la cloud data base. Paso 1	50
Figura 35 Ingresar image target a la cloud database. Paso 2	50

Figura 36 Descargar SDK para Unity	51
Figura 37 Descargar SDK para traducción de texto.....	51
Figura 38 Creación de proyecto en Unity	52
Figura 39 Importar SDK. Paso 1.....	52
Figura 40 Importar SDK. Paso 2.....	53
Figura 41 Importar SDK. Paso 3.....	53
Figura 42 Añadir ARCamara a la escena.	54
Figura 43 ARCamara añadida a la escena.....	54
Figura 44 Componente Vuforia Behaviour presente en ARCamara.....	55
Figura 45 App Licence Key añadido.	55
Figura 46 Prueba de ARCamara en Unity.	55
Figura 47 Añadir image target a la escena.	56
Figura 48 Image target añadido a la escena.....	56
Figura 49 Creando componente dentro del image target.....	57
Figura 50 Visualización del componente 3D Text creado.	57
Figura 51 Diseño final de image target con sus componentes.....	58
Figura 52 Añadir un componente Canvas a la escena.	58
Figura 53 Configuración de Render Mode a Screen Space – Camera	59
Figura 54 Visualización interfaz Menú Inicial.....	59
Figura 55 Visualización interfaz vista de cámara.....	59
Figura 56 Visualización interfaz galería de imágenes en vista de cámara	60
Figura 57 Colocar CloudRecognition prefab a la escena.....	60
Figura 58 Colocar Access Key y Secret Key al CloudRecognition.....	61
Figura 59 Crear Script y añadir a CloudRecognition.....	62
Figura 60 Script añadido al proyecto y al prefab CloudRecognition.....	62

ÍNDICE DE TABLAS

Tabla 1 Comparación del framework de desarrollo.....	13
Tabla 2 Comparación de SDK para realidad aumentada.....	14
Tabla 3 Comparación de base de datos para image target	15
Tabla 4 Clasificación de requerimientos funcionales de la página web.....	20
Tabla 5 Clasificación de requerimientos funcionales de la aplicación móvil.....	23
Tabla 6 Equipo de trabajo y roles	24
Tabla 7 Historia de usuario 1: Identificar usuario.....	24
Tabla 8 Historia de usuario 2: Ingresar nuevo sitio turístico.....	25
Tabla 9 Historia de usuario 3: Modificar sitio turístico.....	25
Tabla 10 Historia de usuario 4: Eliminar sitio turístico	26
Tabla 11 Historia de usuario 5: Seleccionar idioma.....	27
Tabla 12 Historia de usuario 6: Mostrar información de sitio turístico	27
Tabla 13 Mostrar galería de imágenes del sitio turístico	28
Tabla 14 Ocultar información del sitio turístico	28
Tabla 15 Primera iteración: Identificar usuario.....	29
Tabla 16 Segunda iteración: Ingresar nuevo sitio turístico.....	30
Tabla 17 Tercera iteración: Modificar sitio turístico.....	31
Tabla 18 Cuarta iteración: Eliminar sitio turístico.....	31
Tabla 19 Quinta iteración: Seleccionar idioma.....	32
Tabla 20 Sexta iteración: Mostrar información de sitio turístico	32
Tabla 21 Séptima iteración: Mostrar galería de imágenes del sitio turístico.....	33
Tabla 22 Octava iteración: Ocultar información del sitio turístico.....	34
Tabla 23 Características de los navegadores.....	63
Tabla 24 Prueba de aceptación: Identificar usuario	63
Tabla 25 Prueba de aceptación: Ingresar nuevo sitio turístico.....	64
Tabla 26 Prueba de aceptación: Modificar sitio turístico.....	64
Tabla 27 Prueba de aceptación: Eliminar sitio turístico	65
Tabla 28 Características de los dispositivos móviles para plataforma Android	66
Tabla 29 Características de los dispositivos móviles para plataforma iOS	66
Tabla 30 Prueba de aceptación: Seleccionar idioma.....	66
Tabla 31 Prueba de aceptación: Mostrar información de sitio turístico	67
Tabla 32 Prueba de aceptación: Mostrar galería de imágenes del sitio turístico..	67
Tabla 33 Prueba de aceptación: Ocultar información del sitio turístico.....	68

RESUMEN

El presente proyecto está orientado al desarrollo de una aplicación móvil híbrida implementando realidad aumentada para fomentar el turismo en la ciudad de Quito, la cual será implementada en las plataformas Android y iOS.

Para el desarrollo de esta aplicación se utilizó la metodología scrum, ya que esta es una metodología ágil que permitió llevar el control de cada avance que se realizó en este proyecto. El presente documento se encuentra dividido en 4 capítulos, que se especifican a continuación.

En el capítulo I se realiza el planteamiento del problema y se detalla para qué se realizó la aplicación, también se especifica el objetivo general, los objetivos específicos y el alcance de este proyecto.

En el capítulo II se detalla la metodología de desarrollo, las herramientas que se utilizaron para realizar tanto la página web como la aplicación móvil y por último se justifica por qué se utilizaron dichas herramientas.

En el capítulo III se detalla los requerimientos de la aplicación, cómo se llevó a cabo el desarrollo de la página web y la aplicación móvil, así como también, se presenta los resultados obtenidos una vez realizadas las pruebas en las plataformas Android y iOS.

Por último, en el capítulo IV se presentan las conclusiones y recomendaciones obtenidas a lo largo de todo el proyecto.

Palabras clave: Realidad aumentada, Metodología SCRUM

ABSTRACT

This project is aimed at the development of a hybrid mobile application implementing augmented reality to foment tourism in the city of Quito, which will be tested on Android and iOS platforms.

Scrum methodology was used for the development of this application, as this is an agile methodology that allowed to take control of each progress that was made in this project. This document is divided into 4 chapters, which are specified below.

Chapter I outlines the problem to solve and explains why and for what the application was made. this sector also specifies the main objective, specific objectives and the scope of this project.

Chapter II shows the development methodology, the tools used to develop both the web page and the mobile application. Finally, why these tools where used.

Chapter III details the device technical requirements, how the development of the website and the mobile device were carried out, as well as the results obtained after testing on Android and iOS platforms.

Finally, chapter IV presents the conclusions and recommendations obtained throughout the project.

Keywords: *Augmented Reality, SCRUM Methodology*

CAPÍTULO I

1. INTRODUCCIÓN

1.1 Planteamiento del problema

En la década de los 80, empezó a desarrollarse una conciencia turística que trasciende hasta hoy en día y que sigue acogiendo y deslumbrando a miles de turistas. (Quito Turismo, 2014).

Así, en el año 2016 recibió un total de 627.626 turistas y se estima que vendrán para el 2017 650.000 turistas entre nacionales y extranjeros. Quito tiene muchos atractivos, tales como su cultura, su gastronomía y su arte; pero el más importante y que más llama la atención entre los turistas es su atractivo arquitectónico, siendo los sitios más visitados: Mitad del Mundo, Centro Histórico, Teleférico, La Mariscal, El Panecillo, La Ronda, Iglesias, entre otros. (Quito, & Cifras, 2017)

Actualmente, la urbe quiteña tiene para ofrecer infinidad de alternativas para conocer su historia, entre estas se encuentra El Museo de la Ciudad el cual es un espacio en el que converge la historia cotidiana de Quito, desde sus primeros asentamientos hasta los últimos años del siglo XIX. (Quito Adventure, s.f)

Además, se llevan a cabo diversas exposiciones en la urbe quiteña o en museos sobre la historia de la ciudad, una que se destacó por mostrar el atractivo arquitectónico antiguo de la ciudad y su gente, en fotografías fue “Miradas sobre Quito” presentado al público en el bulevar de las naciones unidas y con gran éxito por la visita de quiteños y extranjeros. (Ecuadorinmediato.com, 2012)

Uno de los ejes estratégicos y los pilares del desarrollo sostenible del DMQ, es la industria del turismo, debido al potencial crecimiento económico la Ciudad que el mismo generaría (Plan Estratégico de Desarrollo, Parte I, 2015–2025). Actualmente, la ciudad de Quito es un destino turístico de gran importancia e interés para los turistas internacionales y también para los turistas de otros puntos de Ecuador. Desde esta ciudad se pretende seguir mejorando en materia de turismo y lograr subir el PIB al 8%, que en el momento es de 4,5%. Sin embargo, no cuenta con infraestructura tecnológica apropiada para promoverlo de manera más eficiente.

Para ayudar a mejorar y potenciar el turismo, la promoción, es la estrategia clave. En la actualidad, la ciudad no cuenta con una herramienta que permita a los turistas tener información divulgativa y oportuna para conocer la historia de los sitios visitados, sin que

ellos tengan que depender en gran parte de una agencia de viajes para visitarlos, lo que puede ocasionar una fuga de potenciales turistas.

Por ello, se propuso en este trabajo, desarrollar una aplicación móvil para fomentar el turismo y que les permita a los turistas conocer información relevante del sitio que se esté visitando.

1.2 Objetivo General

Desarrollar una aplicación móvil híbrida mediante realidad aumentada para potenciar el turismo en la ciudad de Quito.

1.3 Objetivos Específicos

- Analizar la información de los sitios turísticos más relevantes en Quito.
- Diseñar la base de datos con la información de los sitios turísticos de Quito.
- Determinar la metodología y la plataforma de programación para el desarrollo de la aplicación móvil y la página web.
- Definir el kit de desarrollo de software que se utilizará para implementar realidad aumentada.
- Realizar las pruebas del funcionamiento de la aplicación móvil en dispositivos Android, iOS.

1.4 Alcance

Potenciar el turismo, favorecerá el crecimiento económico de Quito, uno de los ejes centrales de desarrollo local. Así, el presente proyecto contribuirá en el fortalecimiento del turismo en el DMQ a través del desarrollo de una aplicación móvil, la cual permitirá que el usuario de la misma tenga información oportuna y actualizada sobre la visita o ruta turística de interés. A través de dicha herramienta podrá visualizar una galería de imágenes, y la información sobre el lugar visitado.

Para mejorar la visualización de la información y galería de imágenes que se pretende ofrecer al turista, se incorporó realidad aumentada en dicha aplicación, permitiendo de esta manera localizar la fachada del lugar visitado mediante marcadores, enfocar con la cámara del dispositivo y finalmente sobreponer objetos que mostrarán lo anteriormente mencionado.

La aplicación móvil podrá ser usada como una herramienta para los guías turísticos y así ofrecer una mejor experiencia a los turistas de los lugares visitados en la ciudad de Quito.

Además, se proporcionó una página web que será de uso único para el administrador de la aplicación ya que permitirá el registro, actualización y eliminación de los datos de los lugares turísticos, esta página se encontrará disponible en un servidor de prueba.

En cuanto a la aplicación móvil, está disponible en el idioma español e inglés y fue probada para las plataformas Android y iOS.

CAPÍTULO II

2. METODOLOGÍA

2.1 Metodología de desarrollo ágil Scrum

Scrum es una metodología de desarrollo ágil para software propuesta por Jeff Sutherland, Schwaber y Beedle, la cual es utilizada para guiar las siguientes actividades: requerimientos, análisis, diseño evolución y entrega.

Para llevar a cabo las actividades mencionadas anteriormente Scrum propone roles, artefactos y actividades que se presentan a continuación.

2.1.1 Roles

Define las responsabilidades que cada integrante debe cumplir para garantizar que la información dada sea correcta y que exista comunicación con el cliente para la mejora continua del proyecto. Scrum propone los siguientes roles:

Product owner

El *product owner* traducido al español como dueño del producto es el vínculo entre el cliente y el equipo de desarrollo ya que debe cumplir con las siguientes funciones, (Monte, 2016):

- Definir la estrategia y los objetivos.
- Mantener el *product backlog*.
- Negociar el alcance y presupuesto con el cliente.
- Definir, junto con el *scrum master*, los criterios de aceptación del proyecto y de cada *sprint*.
- Participar en los *sprint reviews*.
- Ayuda al *Scrum master* y al *development team* a resolver cualquier cuestión en lo referente a proyecto, la funcionalidad y los productos.

Scrum master

Es importante mencionar que el *Scrum master* no es el *Project manager*, ya que desempeña otras funciones, las mismas se detallan a continuación:

- Es un *coach* para los componentes del *development team*.
- Proporciona soporte al *development team* y ayuda a solucionar los problemas.
- Realiza los reportes, archivo y lleva registros.

- Se encarga de promover e impulsar mejoras sobre el proceso y sobre el *scrum team*.

Development team

Lleva a cabo el desarrollo de producto y se caracteriza por:

- **Ser flexible:** Cada integrante puede desempeñar diferentes roles en el equipo.
- **Estar auto-organizado:** El equipo define sus roles y su método de trabajo.
- **Ser multidisciplinario:** Para llevar a cabo la ejecución del proyecto, el equipo debe poseer habilidades individuales y colectivas.

Stakeholder

Son aquellas personas que realizan la aceptación del producto acabado, por lo que están obligados a asistir a los *sprint reviews*.

2.1.2 Artefactos

Son las herramientas que plantea Scrum para que los roles mencionados en la sección anterior puedan llevar a cabo sus respectivas actividades. Las herramientas que se detallan a continuación no son las únicas, pero si las mínimas que requiere la metodología.

Product backlog

El *product backlog* es la lista de funcionalidades, que conforma el producto que se va a construir, este se escribe tal y como lo detalla el cliente y se compone de *user stories* (historias de usuario). Cada historia de usuario se va perfeccionando y detallando a medida que se necesita información. El *product owner* es el responsable de mantener la lista, y de encargarse que los usuarios proporcionen suficiente información útil para priorizarla.

Sprint backlog

El *sprint backlog* tiene las mismas características que el *product backlog*, pero aquí las historias de usuario definidas deben tener un nivel de detalle completo para el *sprint* en curso y así poder dividir las en subtarefas para facilitar la construcción de la funcionalidad.

Graphs

Scrum plantea un gráfico como herramienta principal para visualizar la evolución del proyecto. Es el gráfico *burn-down*, el cual posee dos vistas:

- **Release burn-down:** El gráfico presenta dos métricas: la evolución del proyecto y la velocidad del equipo. La línea correspondiente a la velocidad de equipo muestra el número de *story points* que el equipo ha resuelto en cada *sprint*. (Monte, 2016)

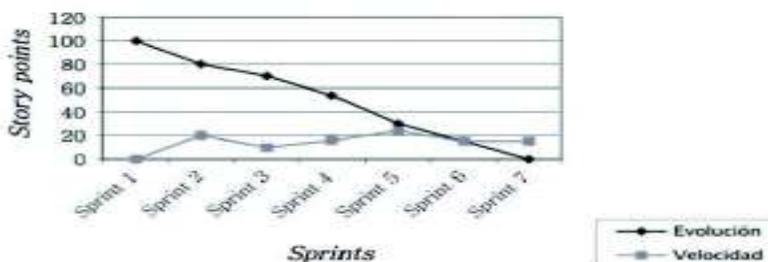


Figura 1 Ejemplo de gráfica “Release burn-down”

Fuente:(Monte, 2016)

- **Sprint burn-down:** El gráfico presenta dos métricas, la previsión que corresponde a la valoración de horas que se hizo para cada tarea, y la evolución real, que corresponde al consumo de horas real de cada tarea. (Monte, 2016)

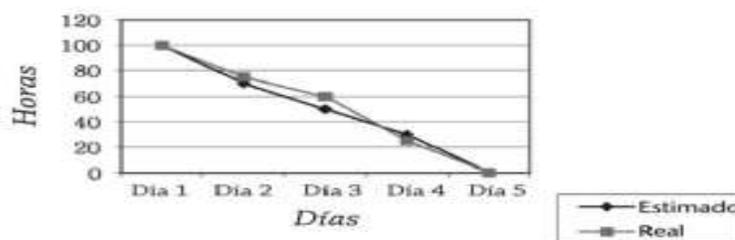


Figura 2 Ejemplo de gráfica “Sprint burn-down”

Fuente: (Monte, 2016)

2.1.3 Actividades

Las actividades son hitos¹ en el tiempo, en los cuales el equipo de proyecto o el cliente tienen que ejecutar alguna acción o llegar a algún acuerdo para el proyecto. Las actividades que propone Scrum son, (Monte, 2016):

¹ **Hito:** Un hito es una tarea de duración cero que simboliza el haber conseguido un logro importante en el proyecto. Obtenido de: <https://iaap.wordpress.com/2007/01/24/hitos-del-proyecto-milestones/>

Sprint 0 o first sprint

El *sprint 0* no es obligatorio ya que corresponde a la fase inicial del proyecto, que el *Scrum team* planifica, fija las reglas y crea la planificación inicial.

Sprint

Un *sprint* es la unidad de tiempo que determina un ciclo de desarrollo con Scrum. En este tiempo debe llevarse a cabo obligatoriamente las siguientes actividades (Monte, 2016):

- El *sprint planning*
- Una reunión diaria (*daily meeting*) con el *development team*.
- Tantas reuniones de refinamiento (*grooming*) como sean necesarias para resolver cuestiones del *sprint* y para preparar el *sprint* siguiente.

La finalidad de un *sprint* es la obtención de un incremento del producto estable, integrado y útil para los usuarios.

Sprint planning

El *sprint planning* se utiliza para planificar en detalle el *sprint*. Recoger la funcionalidad que se ha de desarrollar, resolver dudas, crear las *user stories*, determinar los criterios de aceptación del *sprint* y de cada *user story*. El *sprint planning* se divide en dos acciones.

- **What (qué se hará):** Detalle de la funcionalidad y participa el *scrum team* y, opcionalmente el cliente.
- **How (cómo se hará):** Se hace el desglose en tareas de las *user stories* incluidas.

Release

El *release* sirve para marcar puntos de entrega de producto, por lo tanto, se puede definir como un conjunto de *sprints*. Esto se lleva a cabo cuando el usuario no acepta entregas de funcionalidad al final de cada *sprint*.

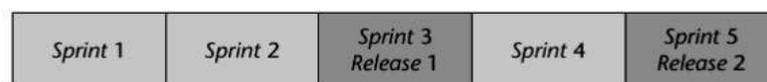


Figura 3 Release en scrum

Fuente:(Monte, 2016)

El *release* se utiliza principalmente cuando el cliente espera ver una funcionalidad acabada, y dicha funcionalidad se construye a lo largo de diversos *sprint*.

Daily scrum

Conocida también como reunión diaria, se debe llevar a cabo en el mismo lugar y la misma hora y debe durar como máximo 15 minutos. Por lo que se refiere a los asistentes, es necesario que esté presente todo el *development team*, el *scrum master* (opcional) y el *product owner* si es invitado. En esta reunión, cada integrante del *development team* debe responder rápidamente a las siguientes preguntas, según (Pressman, 2010):

- ¿Qué hiciste desde la última reunión?
- ¿Tienes algún obstáculo?
- ¿Qué harás antes de la próxima reunión?

Sprint review

Para poder comenzar con el siguiente *sprint*, en el *sprint review* el *development team* debe mostrar a los usuarios/clientes un avance del producto desarrollado, para que aprueben según los criterios establecidos.

Sprint retrospective

En esta reunión, el *development team* y el *scrum master* discuten sobre los sucesos registrados, se busca soluciones para los problemas dados durante el *sprint* que puedan haber sido causa para detener el desarrollo del producto.

Refinement /Grooming

El *grooming* es la principal herramienta para facilitar cambios que se presenten a lo largo del desarrollo del producto. Las reuniones de refinamiento (*grooming*) ayudan a adquirir conocimiento o tratar dichos cambios.

Time-box

En Scrum el principal factor es la duración de las actividades y de las reuniones que se llevan a cabo, por tal motivo no deben hacer más reuniones de las necesarias y deben tener un tiempo límite.

El *time-box* es el tiempo que durará las reuniones, por tal motivo ninguna reunión podrá durar más de lo establecido. En el caso que una reunión necesite ser alargada se debe realizar un receso y volver a convocarla. El *time-box* debe establecerse en el *first sprint* y tiene que ser público para el *scrum team*.

2.2 Herramientas para el acceso de la base de datos

Para elaborar el esquema de la base de datos no relacional que almacenará la información de los sitios turísticos, se utilizará *Sails* y un IDE multiplataforma para alojar la base de datos.

2.2.1 *Sails*

Sails es un *framework* que facilita la creación de aplicaciones, ya que su principal característica es que está diseñado para emular el patrón Modelo Vista Controlador (MVC), y sobre todo que cumple con requerimientos de aplicaciones modernas como es API orientadas a datos con arquitectura escalable y orientada al servicio.

Sails cuenta con un conjunto de módulos que trabajan en conjunto para proporcionar simplicidad, facilidad de mantenimiento y estructuras convenciones a las aplicaciones de Node.js (McNeil y Nathan, 2017)

2.2.2 Base de Datos en *Sails*

Waterline es un módulo de *Sails*, este le da una capa de abstracción a sus aplicaciones por encima de la base de datos como MySQL, MongoDB o PostgreSQL, facilitando métodos que permitan consultar y manipular fácilmente datos sin escribir código de integración específico de PostgreSQL. *Sails* organiza estos métodos en un diccionario llamado modelo.

Modelo en *Sails*

Un modelo en *Sails* representa una tabla como en cualquier otra base de datos. Cada modelo contiene atributos, métodos y otras configuraciones.

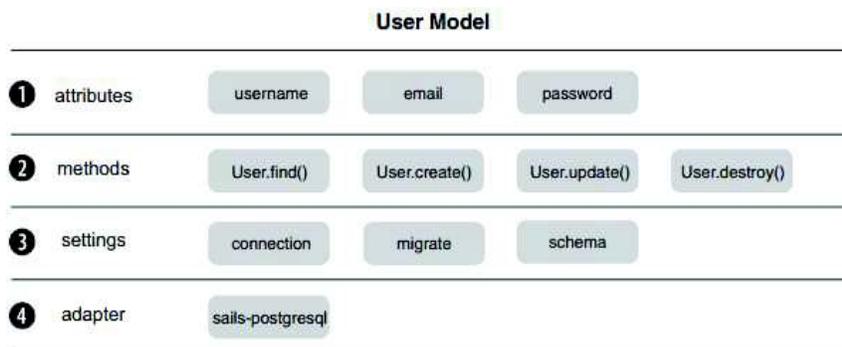


Figura 4 Componentes que se incluyen en un modelo en *Sails*

Fuente:(McNeil y Nathan, 2017)

2.2.3 CodeAnyWhere

CodeAnyWhere es un IDE multiplataforma en la nube con todas las características de un IDE de escritorio que permite crear un servidor de prueba, que soporta más de 120 sintaxis de lenguajes de programación. (Codeanywhere, 2017), y estará disponible mientras se encuentre activa la sesión del usuario que utiliza el IDE.

2.3 Herramientas de desarrollo web

2.3.1 Html

Denominado lenguaje de marcas de hipertexto por sus siglas en inglés (*HyperText Markup Language*), es un lenguaje utilizado para la elaboración de páginas web ya que define una estructura básica de contenido como texto, imágenes, videos, enlaces entre otros.

2.3.2 Css

Por sus siglas en inglés (*Cascading Style Sheets*) traducido como hojas de estilo en cascada, es el lenguaje para dar diseño visual a las páginas web escritas en HTML. CSS fue diseñado para separar el contenido del documento y la forma de presentación de este.

2.3.3 JavaScript

Desarrollado por Brendan Eich, conocido como un lenguaje interpretado para páginas web, con este lenguaje se puede crear aplicaciones web dinámicas. Principalmente es utilizado en lado cliente ya que es implementado como parte del navegador web.

2.3.4 JQuery

Librería de código abierto basada en JavaScript que permite manejar eventos y animaciones dentro de los documentos HTML. Una de ventajas de utilizar esta librería es que se puede actualizar el contenido de una página sin recargarla.

2.3.5 Bootstrap

Es un framework creado por Mark Otto y Jacob Thornton desarrolladores de Twitter, este sirve para crear interfaces de usuario visualmente agradables de páginas y sitios web. Bootstrap incluye un “*Grid system*” que se utiliza para crear diseños de páginas adaptables a cualquier tamaño de dispositivo a través de una serie de filas y columnas que alojan el contenido de la página.

2.3.6 AngularJS

Es un *framework* de JavaScript mantenido por Google, su objetivo es desarrollar aplicaciones basadas en el modelo vista controlador (MVC). La arquitectura MVC se implementa en JavaScript y HTML; la vista se define en HTML, mientras que el modelo y el controlador son implementados en JavaScript.

Modelo Vista Controlador (MVC)

La arquitectura MVC separa la lógica de negocio (el modelo) y la presentación (la vista), mientras que el controlador recibe todas las peticiones de la aplicación dadas por el usuario que son recibidas por el modelo para luego ser presentadas.

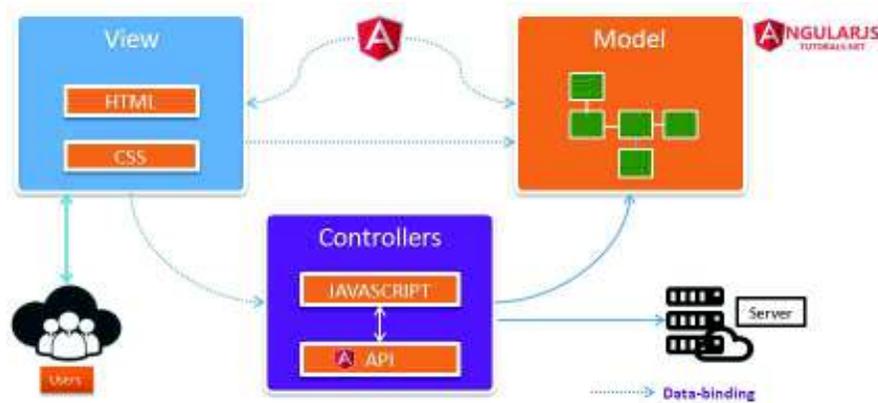


Figura 5 Modelo Vista Controlador en AngularJS

Fuente: <http://angularjstutorials.net/angularjs-interview-questions-with-sample-answers-part5.html>

Elementos importantes en AngularJS

- **Directivas:** Las directivas son marcadores en un elemento DOM² que le indican al compilador de Angular que debe asignar cierto comportamiento a dichos elementos o transformarlos según corresponda. AngularJs viene con un conjunto de directivas incorporadas como, por ejemplo: *ngApp*, *ngModel*, *ngController*, *ngRepeat*, *ngInit*. (AngularJS, 2017)
- **Módulo:** En AngularJS el módulo es un contenedor para las diferentes partes de una aplicación, así también como para los controladores.

² **DOM:** Por sus siglas en inglés (*Document Object Model*), es un conjunto de utilidades específicamente diseñadas para manipular el contenido de documentos XML, pero también se utiliza para manipular contenido de documentos XHTML y HTML. Obtenido de: http://librosweb.es/libro/ajax/capitulo_4.html

- **Controladores:** En AngularJS, un controlador está definido por una función en JavaScript. Cuando un controlador está unido al DOM a través de la directiva *ng-controller*, AngularJS instancia un nuevo objeto, usando el controlador especificado en la función definida por el controlador.
- **Expresiones:** Las expresiones en AngularJS son fragmentos de código JavaScript, que se utilizan en los enlaces de interpolación como por ejemplo `<div id="{{attribute}}">{{data}}</div>`, o directamente una directiva como por ejemplo: `ng-click="function({{argument}})".` (AngularJS, 2017)

2.4 Herramientas de desarrollo móvil

2.4.1 Unity

Unity es el motor ideal para la creación de juegos y aplicaciones que podrán ser compilados en computadoras, consolas de juegos y dispositivos móviles, este permite implementar fácilmente realidad aumentada (RA).

2.4.2 SDK Vuforia

SDK Vuforia utiliza la tecnología de visión por marcadores para reconocer y realizar un seguimiento de imágenes planas (Image Target) y objetos 3D simples. Esta capacidad de registro de imágenes permite a los desarrolladores posicionar y orientar los objetos virtuales, tales como modelos 3D y otros medios de comunicación, en relación con las imágenes del mundo real cuando éstos se ven a través de la cámara de un dispositivo móvil. (Vuforia Developer Portal, 2017)

Image target

Image target representan las imágenes que el Vuforia SDK puede detectar y rastrear. A diferencia de los marcadores de referencia tradicionales, códigos de matriz de datos y códigos QR, las imágenes que serán detectadas no necesitan regiones especiales en blanco y negro o códigos para ser reconocido.

El SDK detecta y rastrea las características que se encuentran de forma natural en la propia imagen mediante la comparación de estas características naturales contra una base de datos de recursos de destino conocido. (Library.vuforia.com, 2016)

2.4.3 SDK Android

Mediante el SDK de Android se puede generar un archivo con extensión .apk. Este archivo incluye todo el código de la aplicación, así como cualquier nuevo recurso multimedia o activo y el archivo AndroidManifest.xml (Jackson, 2012).

2.4.4 Xcode

Xcode es un entorno de desarrollo integrado para macOS, mediante este se puede generar un archivo con extensión .ipa, que significa aplicación de iPhone. Apple también se refiere a estos archivos como *Apple Devices Application Files* que contiene el código que será compilado por el dispositivo. (Landau, 2009).

2.5 Justificación de tecnologías

Para definir las herramientas para el desarrollo del proyecto se utilizó una variable cuantitativa que permitió elegir la tecnología adecuada. El mínimo valor será 1 cuando no cumpla con los requerimientos para el desarrollo y 5 cuando cumpla con los requerimientos para el desarrollo.

2.5.1 Framework de desarrollo

Tabla 1

Comparación de framework de desarrollo

características	Unity	puntuación (1-5)	Ionic	puntuación (1-5)
Uso	Desarrollo de aplicaciones móviles, escritorio, consolas de video juego y <i>Smart TV</i>	5	Desarrollo de aplicaciones móviles	2
Precio licencia	\$35.00 <i>Free</i> para desarrollador	4	\$29 <i>Free</i> para desarrollador	5
Componentes de realidad aumentada	3D <i>Game Object</i>	5	No cuenta con componentes propios en 3D	1
Total		14		8

Elaborado por: Josselyn Rodríguez, Henry Rojas

Para el desarrollo se necesita componentes en 3D específicos para realidad aumentada, por tal motivo la aplicación móvil fue desarrollada con el *framework* de Unity.

2.5.2 SDK para realidad aumentada

Tabla 2

Comparación de SDK para realidad aumentada

características	Vuforia	puntuación (1-5)	Wikitude	puntuación (1-5)
Plataformas	Android, iOS, UWP ³	5	Android, iOS	3
Precio licencia	Para <i>cloud recognition</i> : \$99 mensual por aplicación <i>Free</i> para desarrollador	4	Para <i>cloud recognition</i> : \$441,52 mensual por 2 aplicación <i>No Free</i> para desarrollador	1
Reconocimiento de objetos/imágenes	Image target, multiple target, cloud recognition, object target, text recognition	5	Object Recognition, Multiple image target, image recognition, cloud recognition	5
Total		14		9

Elaborado por: Josselyn Rodríguez y Henry Rojas

Para el desarrollo de la aplicación móvil se utilizó Vuforia SDK, que facilita cumplir con los requisitos de la aplicación, además de que al ser llevado a producción el precio de licencia sería más conveniente.

³ **UWP:** *Universal Windows Platform.*

2.5.3 Base de datos para *image target*

Tabla 3

Comparación de base de datos para *image target*

características	<i>Device database</i>	puntuación (1-5)	<i>cloud database</i>	puntuación (1-5)
Número de <i>image target</i> reconocidos	1000 o menos	1	Mayor a 1 000 000	5
Información descriptiva de la imagen (metadatata)	No soporta metadatata	1	Soporta metadatata de 1MB	5
Uso de red	No requiere conexión a internet para la detección de <i>images target</i>	5	Requiere de conexión a internet para la detección de <i>images target</i>	1
Tiempo de detección	Responde en 2 o 3 frames ⁴	5	Depende de las condiciones de la red	3
Total		12		14

Elaborado por: Josselyn Rodríguez y Henry Rojas

La aplicación móvil está orientada al turismo en la capital del país, por tal motivo los sitios turísticos serán actualizados constantemente. Para que la aplicación sea escalable a futuro se debe utilizar *cloud database* ya que el número de reconocimiento de imágenes es mayor y por tal motivo puede existir más sitios turísticos registrados para la aplicación.

⁴ **frames:** Velocidad a la cual un dispositivo muestra imágenes llamados cuadros o fotogramas.

2.6 Requerimientos hardware

Los requerimientos de hardware para el desarrollo de la aplicación móvil y de la página web son:

- Poseer una computadora Mac para generar la aplicación móvil para plataforma iOS.
- Para los demás requerimientos se puede utilizar computadora con sistema operativo Windows.

GPU

- Tarjeta gráfica con DX9 (modelo de shader 3.0) o DX11 con capacidades de funciones de nivel 9.3

Memoria

- 4GB RAM mínimo
- 10 GB en disco como mínimo

CAPÍTULO III

3. RESULTADOS Y DISCUSIÓN

3.1 Requerimientos generales de la página web

3.1.1 Perspectiva de la página web

Interfaces de usuario para la página web

El usuario administrador interactuará con la página web mediante el navegador instalado en el sistema operativo de la máquina que se esté usando. Los navegadores en los cuales se puede ejecutar, sin ninguna interrupción, la página web son:

- Google Chrome versión 54.0 y posteriores.
- FireFox versión 54.0 y posteriores
- Safari versión 10 y posteriores.

Interfaces de hardware para la página web

Las interfaces de hardware se manejarán a nivel de dispositivos. Para registrar, modificar o eliminar algún sitio turístico se utilizará el mouse y el teclado, mientras que para presentar la información se utilizará la pantalla de la computadora, siendo esta el medio principal donde se vinculará la interfaz que permitirá la comunicación entre software y hardware.

Interfaces de software para la página web

La interfaz de software entregará información para el usuario administrador de cada uno de los sitios turísticos, además de permitir identificarse con el administrador de la página.

Interfaces de comunicación para la página web

La página web mantendrá comunicación con el servidor de prueba levantado en el IDE CodeAnyWhere, ya que en este se guardará la data de los sitios turísticos.

Restricciones de memoria para la página web

La página web necesitará 2MB de espacio para ser almacenada en un servidor de prueba, además necesitará al menos 512MB de memoria principal para ser ejecutada sin ninguna interrupción.

3.1.2 Funciones de la página web

En la página web el usuario administrador podrá realizar las siguientes funciones:

- Iniciar sesión para identificarse como administrador.
- Registrar, modificar y eliminar sitios turísticos.
- Visualizar todos los sitios registrados.

3.1.3 Características del usuario de la página web

EL usuario deberá tener conocimientos de cómo usar una computadora, además de conocer sobre los sitios turísticos del distrito metropolitano de Quito.

3.1.4 Restricciones de la página web

- La página web es exclusivamente para el administrador de la aplicación móvil.
- El usuario administrador obligatoriamente debe identificarse con sus respectivas credenciales para el uso de la página.
- Se debe tener conexión a internet ya que la información de cada uno de los sitios registrados será guardada en un servidor en la nube.
- La página estará disponible en un servidor de pruebas y estará disponible únicamente cuando el usuario tenga activa la sesión del IDE *cloud 9*.

3.2 Requerimientos específicos de la página web

3.2.1 Funcionalidades de la página web

Identificación del usuario

El módulo de registro, modificación y eliminación de los sitios turísticos es de uso exclusivo para el usuario administrador, por tal motivo debe identificarse mediante sus credenciales que serán usuario y contraseña.

Ingreso de sitios turísticos

El usuario administrador podrá ingresar nuevos sitios turísticos con sus respectivos campos presentados en la página web.

Modificación de los sitios turísticos

El usuario administrador podrá modificar algún dato de un sitio que se encuentre registrado en la página web.

Eliminación de los sitios turísticos

El usuario administrador podrá eliminar un sitio que se encuentre registrado en la página web.

Visualización de los sitios registrados

El usuario administrador podrá visualizar los sitios turísticos registrados en la página web.

3.2.2 Especificaciones de caso de uso de la página web

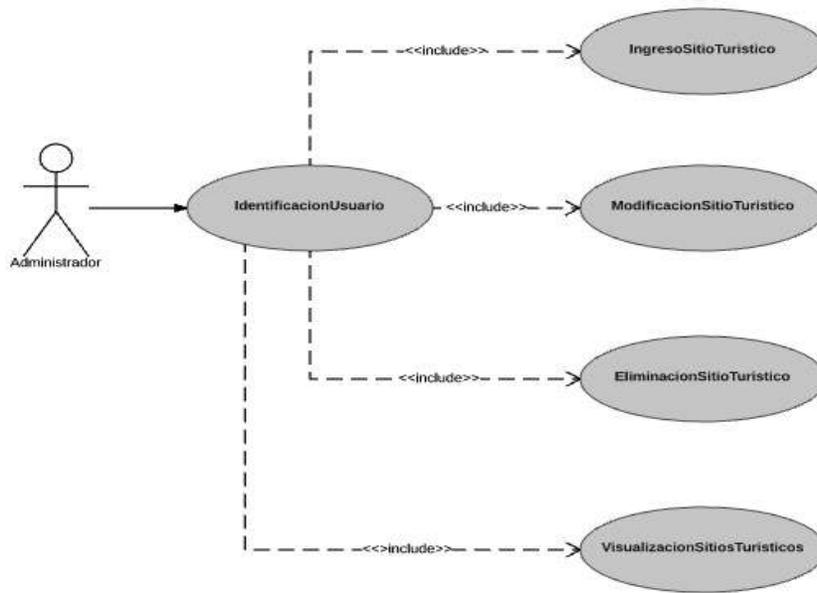


Figura 6 Diagrama de caso de uso de la página web

Elaborado por: Josselyn Rodríguez, Henry Rojas

En la Figura 6 se muestra los pasos o actividades que deberán realizarse para llevar a cabo algún proceso en la página web.

Los personajes o entidades que participan en los casos de uso se denominan actores, en este caso tenemos un único actor denominado administrador.

Se puede observar, Figura 6 algunas relaciones de caso de uso como el `<<include>>`, el cual indica la dependencia entre dos casos de usos que denota que un caso de uso es una especialización del otro, es decir, para que exista las funcionalidades de ingreso, modificación y eliminación de sitios turísticos es necesario que el módulo identificación de usuario exista.

3.2.3 Clasificación de requerimientos funcionales de la página web

Tabla 4

Clasificación de requerimientos funcionales de la página web

Funcionalidad	Tipo
Identificación del usuario	Esencial
Ingreso de sitios turísticos	Esencial
Modificación de sitios turísticos	Esencial
Eliminación de sitios turísticos	Esencial
Visualización de sitios registrados	Esencial

Elaborado por: Josselyn Rodríguez y Henry Rojas

3.3 Requerimientos de ambiente de desarrollo de la página web

La página web será desarrollada empleando herramientas de base de datos no relacional *Sails* y las herramientas de desarrollo son *HTML*, *Bootstrap* y *AngularJs*, estas herramientas se ejecutarán sobre el sistema operativo *Windows 10* o *Mac OS X*.

3.4 Requerimientos generales de la aplicación móvil

3.4.1 Perspectiva de la aplicación móvil

Interfaces de usuario para la aplicación móvil

El usuario interactuará con el sistema mediante la pantalla táctil del dispositivo en el cual se instale la aplicación móvil.

Interfaces de hardware para la aplicación móvil

Las interfaces de hardware se manejarán a nivel de dispositivos, estas serán usadas para interactuar y visualizar la información presentada en la pantalla del móvil, siendo así el medio principal donde se vinculará una interfaz que permitirá la comunicación entre software y hardware.

Interfaces de software para la aplicación móvil

La interfaz de software entregará información de cada lugar que sea consultado por el usuario mediante la aplicación móvil.

Interfaces de comunicación para la aplicación móvil

La aplicación móvil mantendrá comunicación con el servidor de prueba levantado en el IDE CodeAnyWhere, ya que este contiene la data de los sitios turísticos.

Restricciones de Memoria para la aplicación móvil

La limitación de memoria para el almacenamiento de aplicación en el dispositivo móvil será de 100MB.

Adicional a lo anterior el dispositivo debe disponer una memoria principal de al menos 1GB para que la aplicación se ejecute sin interrupciones.

3.4.2 Funciones de la aplicación móvil

La aplicación móvil mostrará información y una galería de imágenes acerca del lugar turístico visitado por el usuario, además de eso el usuario podrá seleccionar el idioma en el que quiere visualizar la información de dicho lugar.

3.4.3 Características del usuario de la aplicación móvil

El usuario deberá saber utilizar un celular inteligente.

3.4.4 Restricciones de aplicación móvil

- La aplicación móvil deberá contar con acceso a internet ya que la información que se mostrará será consumida de un servidor en la nube.
- La aplicación móvil estará únicamente disponible para los dispositivos que utilicen plataformas Android y iOS.
- El sistema operativo de Android debe ser 4.1⁵ o posterior y el sistema operativo de iOS debe ser 7.0⁶ o posterior.
- El usuario tendrá únicamente la opción de seleccionar el idioma inglés o español para visualizar la información.
- La aplicación móvil tendrá únicamente 50 lugares turísticos del distrito metropolitano de Quito registrados en la base de datos, esto es por motivos que el presente proyecto es únicamente un prototipo para la previa obtención del título profesional.
- La aplicación móvil no será publicada en las tiendas virtuales de las plataformas Android y iOS.

⁵ Obtenido de: <https://unity3d.com/es/unity/system-requirements>

⁶ Obtenido de: <https://unity3d.com/es/unity/system-requirements>

3.5 Requerimientos específicos de la aplicación móvil

3.5.1 Funcionalidades de la aplicación móvil

Selección del idioma

La aplicación móvil está orientada a potenciar el turismo en la ciudad de Quito, los sitios turísticos serán visitados por turistas nacionales y extranjeros, por tal motivo el usuario podrá escoger el idioma ya sea inglés o español.

Visualización de la galería de imágenes del sitio turístico

El usuario podrá visualizar en su dispositivo una galería del sitio visitado.

Visualización de la información del sitio turístico

El usuario podrá visualizar en su dispositivo información relevante del sitio turístico en el idioma seleccionado.

Ocultar información del sitio turístico

El usuario podrá ocultar la información del sitio turístico visitado, para visualizar de mejor manera la galería.

3.5.2 Especificaciones de caso de uso de aplicación móvil

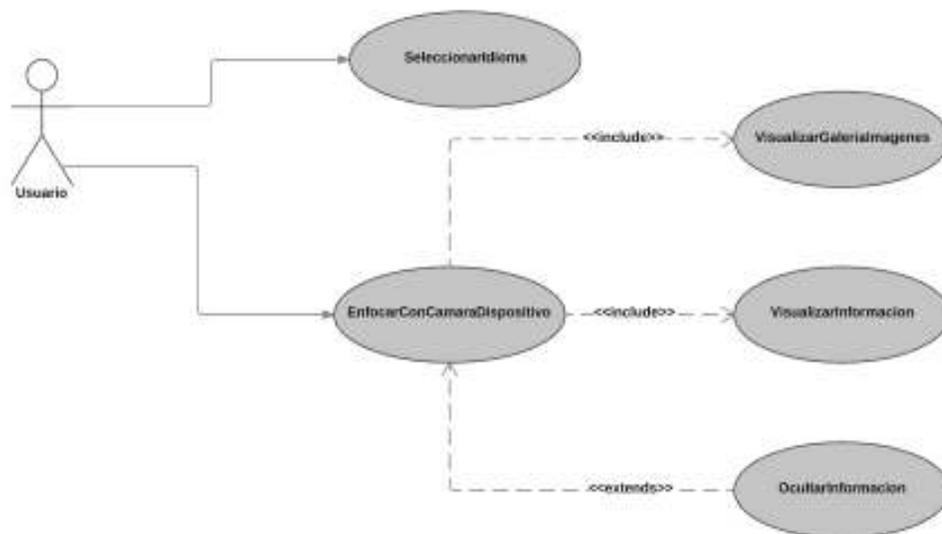


Figura 7 Diagrama de caso de uso de la aplicación móvil

Elaborado por: Josselyn Rodríguez, Henry Rojas

En la Figura 7 se muestra los pasos o actividades que deberán realizarse para llevar a cabo algún proceso en la aplicación móvil.

Los personajes o entidades que participan en los casos de uso se denominan actores, en este caso tenemos un único actor denominado usuario.

Se puede observar, Figura 7, algunas relaciones de caso de uso como el <<include>>, el cual indica la dependencia entre dos casos de usos que denota que un caso de uso es una especialización del otro, es decir, para que exista las funcionalidades de visualizar galerías de imágenes, visualizar información es necesario que el módulo enfocar con la cámara del dispositivo exista.

3.5.3 Clasificación de requerimientos funcionales de la aplicación móvil

Tabla 5

Clasificación de requerimientos funcionales de la aplicación móvil

Funcionalidad	Tipo
Selección del idioma	Esencial
Visualización de galería de imágenes del sitio turístico	Esencial
Visualización de información del sitio turístico	Esencial
Ocultar información del sitio turístico	Opcional

Elaborado por: Josselyn Rodríguez y Henry Rojas

3.6 Requerimientos de ambiente de desarrollo para la aplicación móvil

Para el desarrollo de la aplicación móvil se empleará la herramienta de desarrollo Unity utilizando C# como lenguaje de programación, SDK Android y Xcode, estas herramientas se ejecutarán sin ningún inconveniente con los requisitos mínimos que se detallan a continuación⁷:

- Windows 7 SP1+, 8, 10; Mac OS X 10.9+
- iOS: Computadora Mac con sistema operativo versión OS X 10.9.4 como mínimo y Xcode 7.0+.
- Android: Android SDK y Java Development Kit (JDK).
- Visual Studio 2015 como mínimo (opcional).

⁷ Obtenido de: <https://unity3d.com/es/unity/system-requirements>

3.7 Conformación del equipo de trabajo

Tabla 6

Equipo de trabajo y roles

Personas / Entidad	Rol
Escuela Politécnica Nacional	<i>Product Owner</i>
Ing. Juan Pablo Zaldumbide	<i>Scrum Master</i>
Josselyn Rodríguez	<i>Development Team</i>
Henry Rojas	<i>Development Team</i>

Elaborado por: Josselyn Rodríguez y Henry Rojas

3.8 Actividades

3.8.1 Historias de usuario

El *product backlog* se encuentra conformado por ocho historias de usuario, las cuales han permitido recopilar los requerimientos de la aplicación.

Tabla 7

Historia de usuario 1: Identificar usuario

Historia de Usuario	
Identificador (ID): 1	Usuario: Administrador
Nombre Historia: Identificación del usuario	
Prioridad en negocio: Alta	Riesgo en desarrollo: Baja
Iteración asignada: 1	
Programador(es) responsable(s): Josselyn Rodríguez- Henry Rojas	
Descripción: La página web será diseñada exclusivamente para el administrador de la aplicación móvil, por tal motivo es indispensable que el usuario se identifique con sus respectivas credenciales que serán usuario y contraseña. El usuario debe tener la opción de finalizar su sesión cuando no se encuentre usando la página web.	
Observación: Como solo habrá un único usuario, las credenciales serán asignadas por los autores del proyecto.	

Elaborado por: Josselyn Rodríguez, Henry Rojas

Tabla 8*Historia de usuario 2: Ingresar nuevo sitio turístico*

Historia de Usuario	
Identificador (ID): 2	Usuario: Administrador
Nombre Historia: Ingresar nuevo sitio turístico	
Prioridad en negocio: Alta	Riesgo en desarrollo: Baja
Iteración asignada: 1	
Programador(es) responsable(s): Josselyn Rodríguez- Henry Rojas	
<p>Descripción:</p> <p>Se deberá buscar la información y las imágenes de cada sitio turístico que se desea ingresar en la página web. Se ingresarán los datos del sitio turístico en la página web. Para cada sitio turístico se deberá ingresar los siguientes datos:</p> <ul style="list-style-type: none"> • Nombre del sitio • Tipo de sitio (iglesias, museos, parques, otros) • Descripción del sitio • Historia del sitio • Período de construcción/fundación • Estilo arquitectónico • Galería de imágenes 	
<p>Observación:</p> <p>La descripción y la historia del sitio no debe superar las 70 palabras. Cada sitio turístico debe tener al menos 5 imágenes.</p>	

Elaborado por: Josselyn Rodríguez, Henry Rojas**Tabla 9***Historia de usuario 3: Modificar sitio turístico*

Historia de Usuario	
Identificador (ID): 3	Usuario: Administrador
Nombre Historia: Modificar sitio turístico	
Prioridad en negocio: Media	Riesgo en desarrollo: Baja
Iteración asignada: 1	

Programador(es) responsable(s): Josselyn Rodríguez- Henry Rojas
<p>Descripción:</p> <p>Se modificarán los datos del sitio turístico que fueron ingresados en la página web. Para cada sitio turístico se podrá modificar los siguientes datos:</p> <ul style="list-style-type: none"> • Nombre del sitio • Tipo de sitio (iglesias, museos, parques, otros) • Descripción del sitio • Historia del sitio • Período de construcción/fundación • Estilo arquitectónico • Galería de imágenes
<p>Observación:</p> <p>La descripción y la historia del sitio no debe superar las 70 palabras. Cada sitio turístico debe tener al menos 5 imágenes.</p>

Elaborado por: Josselyn Rodríguez, Henry Rojas

Tabla 10

Historia de usuario 4: Eliminar sitio turístico

Historia de usuario	
Identificador (ID): 4	Usuario: Administrador
Nombre Historia: Eliminar sitio turístico	
Prioridad en negocio: Baja	Riesgo en desarrollo: Baja
Iteración asignada: 1	
Programador(es) responsable(s): Josselyn Rodríguez- Henry Rojas	
<p>Descripción:</p> <p>Se eliminará los datos del sitio turístico seleccionado por el usuario.</p>	
<p>Observación:</p> <p>Los datos del sitio serán eliminados en su totalidad de la base de datos.</p>	

Elaborado por: Josselyn Rodríguez, Henry Rojas

Tabla 11*Historia de usuario 5: Seleccionar idioma*

Historia de Usuario	
Identificador (ID): 5	Usuario: Usuario
Nombre Historia: Seleccionar idioma	
Prioridad en negocio: Alta	Riesgo en desarrollo: Baja
Iteración asignada: 1	
Programador(es) responsable(s): Josselyn Rodríguez- Henry Rojas	
Descripción: La aplicación móvil está orientada a turistas nacionales y extranjeros, por tal motivo el usuario podrá elegir el idioma de preferencia para mostrar la información del sitio turístico.	
Observación: Los idiomas que el usuario podrá elegir será español e inglés.	

Elaborado por: Josselyn Rodríguez y Henry Rojas**Tabla 12***Historia de usuario 6: Mostrar información del sitio turístico*

Historia de usuario	
Identificador (ID): 6	Usuario: Administrador
Nombre Historia: Mostrar información del sitio turístico	
Prioridad en negocio: Alta	Riesgo en desarrollo: Baja
Iteración asignada: 1	
Programador(es) responsable(s): Josselyn Rodríguez- Henry Rojas	
Descripción: El usuario deberá localizar la fachada del sitio y enfocar con la cámara del dispositivo el código QR diseñado para cada sitio turístico, de tal manera que al ser reconocido dicho código se muestre la información	
Observación: El código QR será diseñado por los autores del proyecto.	

Elaborado por: Josselyn Rodríguez y Henry Rojas

Tabla 13*Historia de usuario 7: Mostrar galería de imágenes del sitio turístico*

Historia de Usuario	
Identificador (ID): 7	Usuario: Usuario
Nombre Historia: Mostrar galería de imágenes del sitio turístico	
Prioridad en negocio: Alta	Riesgo en desarrollo: Baja
Iteración asignada: 1	
Programador(es) responsable(s): Josselyn Rodríguez- Henry Rojas	
Descripción: El usuario deberá localizar la fachada del sitio y enfocar con la cámara del dispositivo el código QR diseñado para cada sitio turístico, de tal manera que al ser reconocido dicho código se muestre la galería de imágenes.	
Observación: El código QR será diseñado por los autores del proyecto.	

Elaborado por: Josselyn Rodríguez y Henry Rojas**Tabla 14***Historia de usuario 8: Ocultar información del sitio turístico*

Historia de Usuario	
Identificador (ID): 8	Usuario: Usuario
Nombre Historia: Ocultar información del sitio turístico	
Prioridad en negocio: Baja	Riesgo en desarrollo: Baja
Iteración asignada: 1	
Programador(es) responsable(s): Josselyn Rodríguez- Henry Rojas	
Descripción: El usuario podrá ocultar la información del sitio para visualizar de mejor manera la galería de imágenes presentada para el sitio.	
Observación:	

Elaborado por: Josselyn Rodríguez y Henry Rojas

3.8.2 Iteraciones

Se ha distribuido los requerimientos en ocho iteraciones que se detallan a continuación. Cada iteración muestra una lista de actividades y el tiempo estimado que se utilizó para llevar a cabo las tareas requeridas por el cliente. El identificador y el enunciado del product backlog pertenecen a las historias de usuario.

Primera iteración

En la primera iteración se establece las actividades para realizar la página de identificación del usuario administrador.

Tabla 15

Primera iteración: Identificar usuario

Identificador de ítem de <i>product backlog</i>	Enunciado del ítem de <i>product backlog</i>	Tarea	Tiempo estimado (Horas)
1	Identificar usuario	Diseño de la interfaz de usuario para iniciar sesión	30
		Vincular campos de la interfaz con los datos del usuario administrador	
		Definición de las funciones para que el usuario inicie y cierre sesión	
		Comprobación que los campos no se encuentren vacíos o contenga información incorrecta	
		Pruebas con el usuario	

Elaborado por: Josselyn Rodríguez y Henry Rojas

Segunda iteración

En la segunda iteración se detalla las actividades para el ingreso de un nuevo sitio turístico en la página de administración, además del ingreso de un sitio en esta iteración se recopila la información que tendrá la aplicación móvil y la conexión correspondiente al servidor que contendrá la data de los sitios.

Tabla 16

Segunda iteración: Ingresar nuevo sitio turístico

Identificador de ítem de <i>product backlog</i>	Enunciado del ítem de <i>product backlog</i>	Tareas	Tiempo Estimado (Horas)
2	Ingresar nuevo sitio turístico	Definición de los atributos que tendrá cada sitio turístico	90
		Búsqueda de la información de los 50 lugares turísticos que conformaran la base de datos	
		Búsqueda de las imágenes de los 50 lugares turísticos que conformaran la base de datos	
		Diseño de la interfaz de usuario para ingresar los sitios turísticos	
		Creación del modelo de la base de datos en el servidor de <i>Sails</i>	
		Conexión de la página con el servidor <i>Sails</i> que contiene la base de datos	
		Definición de función para el ingreso del sitio a la base datos	
		Comprobación que los campos no se encuentren vacíos o contenga información incorrecta	
		Pruebas con el usuario	

Elaborado por: Josselyn Rodríguez y Henry Rojas

Tercera iteración

En la tercera iteración se detalla las actividades para modificar los datos de un sitio turístico registrado.

Tabla 17*Tercera iteración: Modificar sitio turístico*

Identificador de ítem de <i>product backlog</i>	Enunciado del ítem de <i>product backlog</i>	Tareas	Tiempo Estimado (Horas)
3	Modificar sitio turístico	Diseño de la interfaz de usuario para modificar un sitio turístico	30
		Definición de función para modificar el sitio en la base datos	
		Comprobación que los campos no se encuentren vacíos o contenga información incorrecta	
		Pruebas con el usuario	

Elaborado por: Josselyn Rodríguez y Henry Rojas**Cuarta iteración**

En la cuarta iteración se detalla las actividades que se llevan a cabo para eliminar un sitio turístico registrado. Los datos y las imágenes de cada sitio turístico serán eliminado totalmente del servidor y no se podrá recuperar dichos datos.

Al ser la última iteración del desarrollo de la página web, se incluirá la actividad para subir la página al servidor en la nube *cloud 9*.

Tabla 18*Cuarta iteración: Eliminar sitio turístico*

Identificador de ítem de <i>product backlog</i>	Enunciado del ítem de <i>product backlog</i>	Tareas	Tiempo Estimado (Horas)
4	Eliminar sitio turístico	Diseño de la interfaz de usuario para eliminar un sitio turístico	40
		Definición de función para eliminar el sitio en la base datos	
		Subir la página completa al servidor en la nube.	
		Pruebas con el usuario	

Elaborado por: Josselyn Rodríguez y Henry Rojas

Quinta iteración

En la quinta iteración se detalla las actividades necesarias para cambiar el idioma en el que se muestra la información del sitio turístico consultado.

Tabla 19

Quinta iteración: Seleccionar idioma

Identificador de ítem de <i>product backlog</i>	Enunciado del ítem de <i>product backlog</i>	Tareas	Tiempo Estimado (Horas)
5	Seleccionar idioma	Obtener cuenta en bluemix.net para obtener credenciales del servicio de <i>Language Translator</i>	60
		Descargar e instalar las librerías de watson sdk en el proyecto.	
		Configurar credenciales para el uso de la librería	
		Definir los métodos para traducir el idioma utilizando las librerías de Watson	
		Pruebas en el dispositivo móvil	

Elaborado por: Josselyn Rodríguez y Henry Rojas

Sexta iteración

En la sexta iteración se detalla las actividades necesarias para mostrar la información de un sitio turístico utilizando realidad aumentada, en esta iteración también se diseña los códigos QR (*image target*) que identificarán cada sitio turístico.

Tabla 20

Sexta iteración: Mostrar información del sitio turístico

Identificador de ítem de <i>product backlog</i>	Enunciado del ítem de <i>product backlog</i>	Tareas	Tiempo Estimado (Horas)
6	Mostrar información del sitio turístico	Creación de los códigos QR que representan los <i>images target</i> de los sitios turísticos	90

		Obtener cuenta en vuforia.com para crear y administrar bases de datos de <i>image target</i>
		Creación de clave de licencia para la aplicación
		Descargar e instalar los paquetes para Unity y colocar la clave de licencia
		Añadir los componentes (" <i>prefabs</i> ") necesarios para el funcionamiento de realidad aumentada (RA)
		Definición de método para obtener información de los sitios turísticos registrados en el servidor
		Implementar la interfaz para el uso de la base de datos de Vuforia y creación de métodos para el uso de la interfaz
		Pruebas en el dispositivo móvil

Elaborado por: Josselyn Rodríguez y Henry Rojas

Séptima iteración

En la séptima iteración se detalla las actividades necesarias para mostrar la galería de imágenes de los sitios turísticos registrados.

Tabla 21

Séptima iteración: Mostrar galería de imágenes del sitio turístico

Identificador de ítem de <i>product backlog</i>	Enunciado del ítem de <i>product backlog</i>	Tareas	Tiempo Estimado (Horas)
7	Mostrar galería de imágenes del sitio turístico	Añadir los <i>game Object</i> 3D a la escena	70
		Definición de método para obtener las imágenes del servidor y colocar dichas imágenes en el <i>game Object</i>	

		Definición de método para el cambio interactivo de imágenes en el <i>game Object</i>	
		Pruebas en el dispositivo móvil	

Elaborado por: Josselyn Rodríguez y Henry Rojas

Octava iteración

En la octava iteración se detalla las actividades para que el usuario oculte la información del sitio, es decir pueda visualizar de mejor manera las imágenes presentadas del sitio turístico consultado.

Tabla 22

Octava iteración: Ocultar información del sitio turístico

Identificador de ítem de <i>product backlog</i>	Enunciado del ítem de <i>product backlog</i>	Tareas	Tiempo Estimado (Horas)
8	Ocultar información del sitio turístico	Añadir los <i>game Object</i> UI a la escena	70
		Definición de método para colocar las imágenes de cada sitio en el <i>game Object</i> correspondiente	
		Pruebas en el dispositivo móvil	

Elaborado por: Josselyn Rodríguez y Henry Rojas

3.9 Creación del servidor *Sails*

Para crear el servidor *Sails* se obtuvo una cuenta en la página www.codeanywhere.com, una vez obtenido la cuenta se debe estableció una conexión con las especificaciones que se detallan a continuación:

- Insertar un nombre para el servidor
- Seleccionar el ambiente en el cual se va a desarrollar. Para este proyecto se debe elegir *Sails* en Ubuntu.

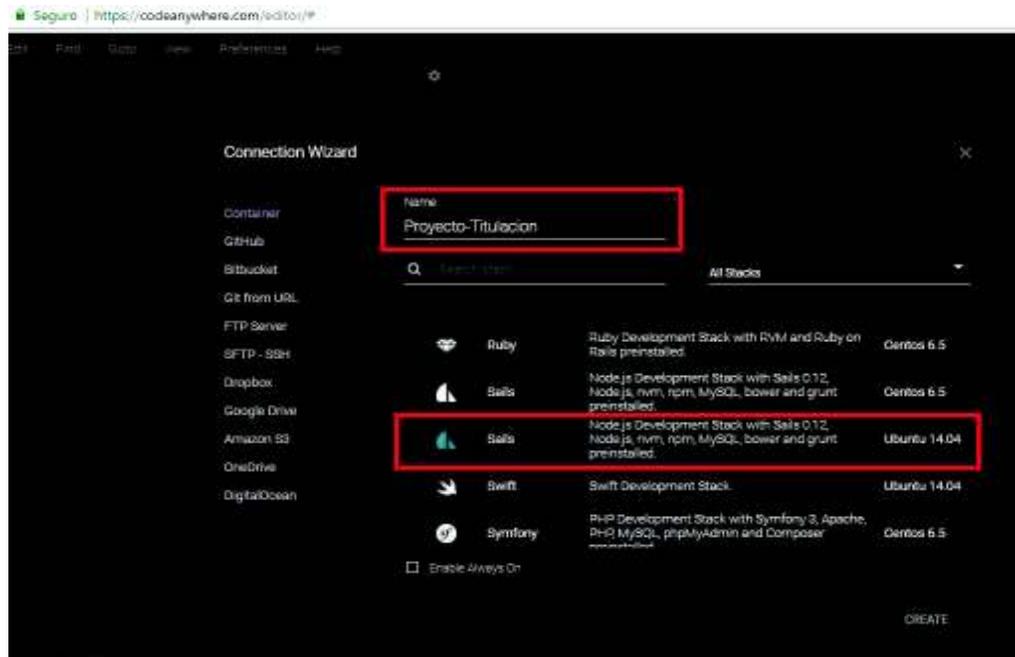


Figura 8 Creación del servidor

Elaborado por: Josselyn Rodríguez, Henry Rojas

3.9.1 Creación del modelo en Sails

Para la creación del modelo se creó un API en el servidor con el siguiente comando `sails generate api [nombreApi]`.

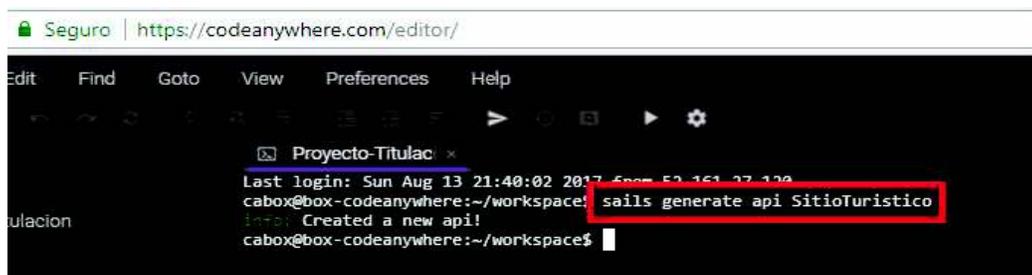
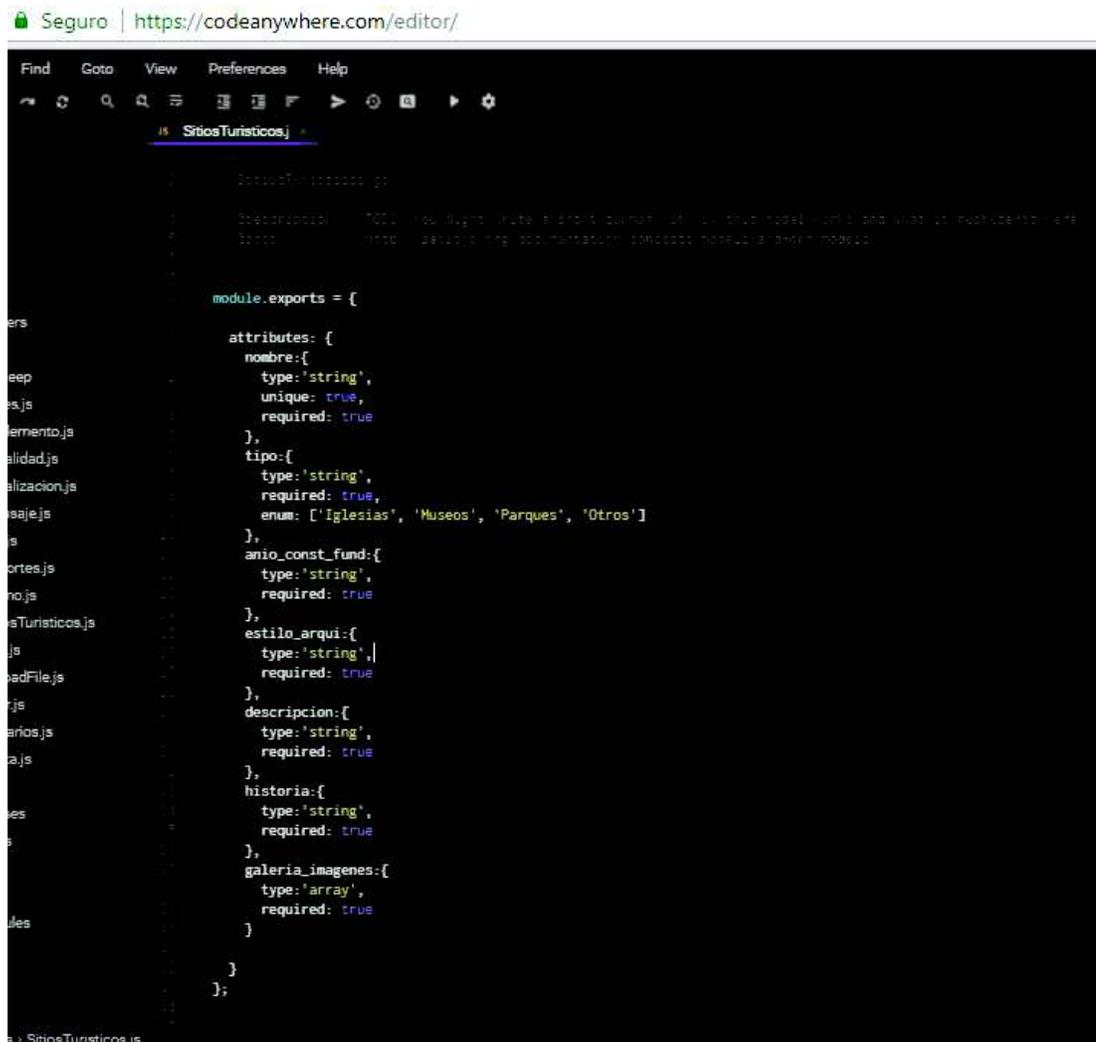


Figura 9 Creación API en Sails

Elaborado por: Josselyn Rodríguez, Henry Rojas

Una vez creado el API se colocó en el archivo del modelo los atributos correspondientes a los sitios turísticos, los atributos fueron definidos con el cliente. (Véase Tabla 8). Cada atributo tiene su respectiva validación del lado del servidor.

En el servidor todos los atributos fueron definidos como tipo *string* excepto la galería que fue definido como tipo *array* ya que almacena un conjunto de urls, el atributo período de creación/fundación fue tomado como *string* porque muchos de los sitios turísticos en la ciudad de Quito fueron creados hace décadas y por tal motivo no se logró encontrar fechas exactas.



```
module.exports = {
  attributes: {
    nombre: {
      type: 'string',
      unique: true,
      required: true
    },
    tipo: {
      type: 'string',
      required: true,
      enum: ['Iglesias', 'Museos', 'Parques', 'Otros']
    },
    anio_const_fund: {
      type: 'string',
      required: true
    },
    estilo_arqui: {
      type: 'string',
      required: true
    },
    descripcion: {
      type: 'string',
      required: true
    },
    historia: {
      type: 'string',
      required: true
    },
    galeria_imagenes: {
      type: 'array',
      required: true
    }
  }
};
```

Figura 10 Creación del modelo de sitios turísticos en Sails

Elaborado por: Josselyn Rodríguez, Henry Rojas

3.10 Desarrollo de la página de administración de sitios turísticos

Para la creación de las interfaces de usuario de la página de administración se realizó un prototipo que fue mostrado al *Product Owner* antes de llevarlo a cabo a producción. Véase Anexo 1.

La página de administración está conformada por dos páginas, la primera es para iniciar sesión y la segunda contiene el ingreso, modificación y eliminación de sitios turísticos, la misma tiene un diseño *one page* ya que se encuentra conformada por pocas secciones y únicamente se muestra información relevante de cada sitio turístico.

Esta página cuenta con las validaciones necesarias para que la información de cada uno de los sitios turísticos sea consistente. En el manual de usuario se muestra los avisos correspondientes si el usuario no ingresa la información necesaria para el sitio. Véase Anexo 2.

3.10.1 Creación de la página iniciar sesión

Como se ha mencionado en los requerimientos del proyecto la página será de uso exclusivo para el administrador de la aplicación, por tal motivo existe un único usuario definido por los autores del proyecto.



Figura 11 Página de inicio de sesión

Elaborado por: Josselyn Rodríguez, Henry Rojas

3.10.2 Creación de la página de administración

Sección registrar sitio turístico

Esta sección está conformada por un formulario con los campos que representan a cada uno de los atributos de los sitios turísticos.

The screenshot shows a web form titled "REGISTRAR SITIO" under the header "SITIOS TURÍSTICOS". The form is for registering a site and includes the following fields:

- Nombre:** A text input field.
- Periodo Construcción/Funcionamiento:** A text input field.
- Tipo:** A dropdown menu with "Seleccionar" as the current selection.
- Descripción:** A text area for entering details.
- Código Arquitectónico:** A dropdown menu with "Seleccionar" as the current selection.
- Paisaje:** A text input field.
- Imagenes:** A text input field with a "Cargar" button next to it.

 At the bottom of the form is a prominent green button labeled "REGISTRAR SITIO".

Figura 12 Página de administración de sitios turísticos sección registrar sitio

Elaborado por: Josselyn Rodríguez, Henry Rojas

La información y las imágenes de los 50 sitios turísticos fueron consultadas por los autores del proyecto y registradas en la página proporcionada para administrar los mismos.

Sección de portafolio

Esta sección está conformada por la lista de sitios turísticos registrados, así también se puede elegir un sitio para visualizar, modificar o eliminar la información del mismo. Al ser seleccionado un sitio para modificar se desplaza un modal mostrando los datos del sitio. Si el administrador elimina un sitio se muestra un modal solicitando la verificación del usuario.



Figura 13 Lista de sitios turísticos registrados

Elaborado por: Josselyn Rodríguez, Henry Rojas



Figura 14 Modal para modificar datos

Elaborado por: Josselyn Rodríguez, Henry Rojas



Figura 15 Modal de verificación para eliminar un sitio turístico

Elaborado por: Josselyn Rodríguez, Henry Rojas

El usuario administrador tiene la opción para descargar la metadata del sitio turístico, la cual es utilizada para subir el *image target* correspondiente a la *cloud database*.

3.10.3 Subir página de administración al servidor *cloud 9*

La página de administración de sitios turísticos fue subida a un servidor en la nube llamado *cloud 9*, en el alcance del proyecto se especificó que la página se encontrará disponible mientras se encuentre activa la sesión del usuario que utilice el IDE. Para subir dicha página se siguió los pasos que se detallan a continuación:

- Poseer una cuenta activa en c9.io
- Crear un *workspace*. Véase Figura 16.
- Colocar un nombre y una descripción para el *workspace*, además se debe elegir un *template*. Para este proyecto se seleccionó HTML 5, ya que la página no necesita de ningún servidor y/o *framework* para su correcto funcionamiento. Véase Figura 17.
- Presionar el botón *create workspace*.
- Una vez creado el *workspace*, abrir el editor presionando el botón *open*. Véase Figura 18.
- Subir todas las carpetas y archivo de la página de administración. Véase Figura 19.
- Finalmente, dar clic derecho sobre el documento login.html y presionar el ítem *run*. Véase Figura 20.
- Una vez que la página se ejecute aparecerá la dirección URL a la cual se podrá acceder para el uso de la misma. Véase Figura 21.

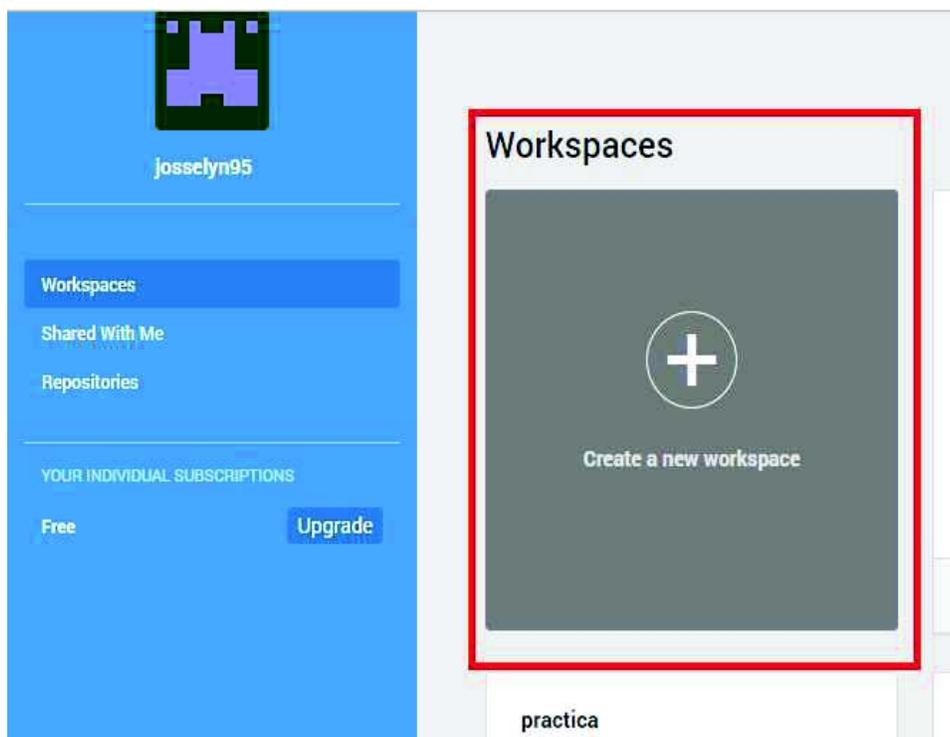


Figura 16 Crear un *workspace* para la página de administración

Elaborado por: Josselyn Rodríguez, Henry Rojas

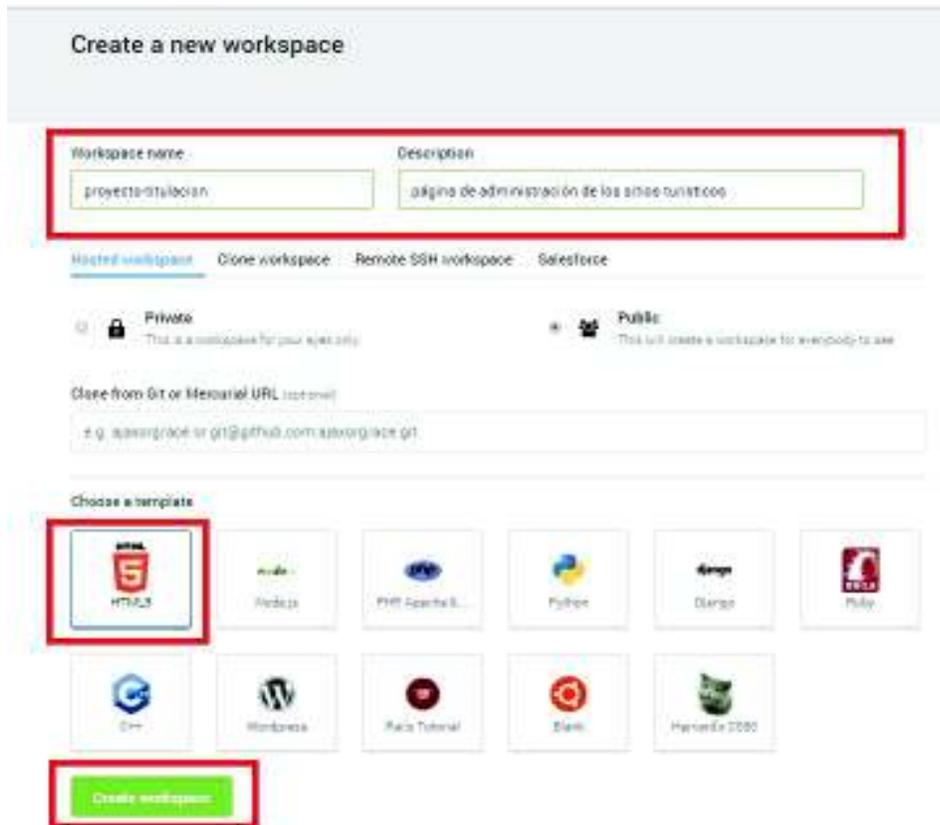


Figura 17 Colocar datos para crear el *workspace*

Elaborado por: Josselyn Rodríguez, Henry Rojas



Figura 18 Abrir el *workspace* creado

Elaborado por: Josselyn Rodríguez, Henry Rojas

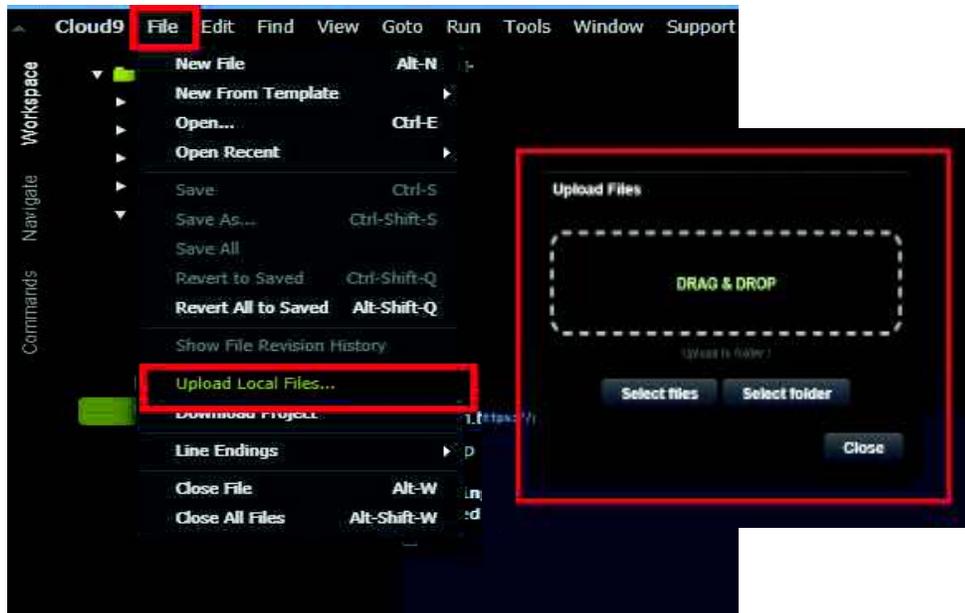


Figura 19 Subir archivos y carpetas de la página de administración

Elaborado por: Josselyn Rodríguez, Henry Rojas

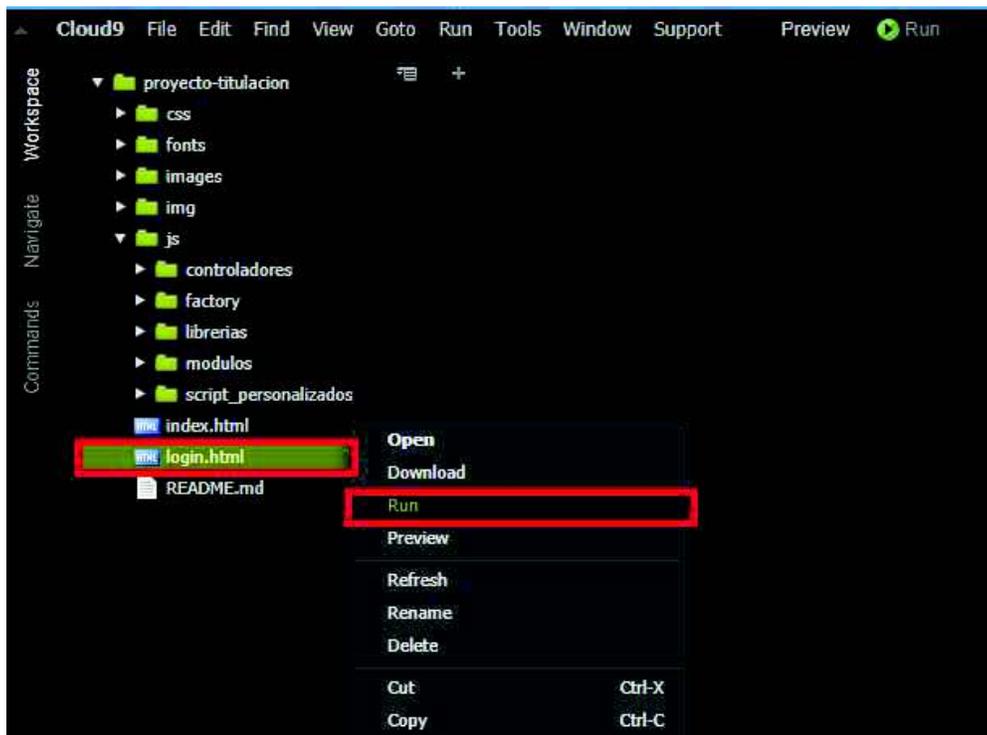


Figura 20 Ejecutar el archivo login.html

Elaborado por: Josselyn Rodríguez, Henry Rojas



Figura 21 Página ejecutada correctamente.

Elaborado por: Josselyn Rodríguez, Henry Rojas

Nota: El link de acceso es: <https://proyecto-titulacion-josselyn95.c9users.io/login.html>

3.11 Desarrollo de la aplicación móvil

Para la creación de las interfaces de usuario de la aplicación móvil se realizó un prototipo que fue mostrado al *Product Owner* antes de llevarlo a cabo a producción. Véase Anexo 3.

La aplicación móvil cuenta con 3 interfaces de usuario que son creadas en el *framework* de Unity, las cuales despliegan elementos del menú de inicio para seleccionar el idioma e iniciar la aplicación, cambiar el idioma, mostrar información y la galería de cada sitio turístico y la visualización y navegación entre las imágenes correspondientes a cada sitio respectivamente.

3.11.1 Creación de códigos QR

Historia

Los códigos QR fueron creados por la compañía japonesa Denso Wave, subsidiaria de Toyota, en 1994 por Euge Damm y Joaco Rete. El término QR significa *Quick Response* debido a la respuesta inmediata que ofrece a través de su lectura. (SoMaMFyC, 2017).

Definición

Un código QR es un código de barras bidimensional cuadrada, que por lo general se encuentra formado por cuadrados negros en fondo blanco, el cual permite interactuar con aplicaciones a través del móvil.

Estructura

Para que los códigos QR sean reconocidos por una aplicación deben cumplir con la estructura básica. (SoMaMFyC, 2017):

- Tres cuadrados en las esquinas que permiten detectar al lector la posición del código QR.
- Una serie de cuadrados dispersos que codifican el alineamiento y la sincronización.

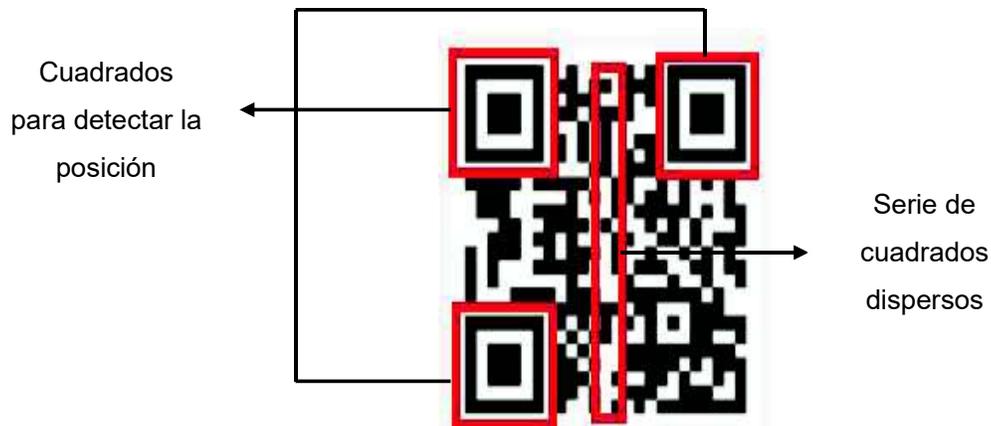


Figura 22 Estructura básica de un código QR

Fuente: (SoMaMFyC, 2017)

Editado por: Josselyn Rodríguez, Henry Rojas

Actualmente se puede realizar códigos QR personalizados, siempre y cuando cumpla con la estructura mencionada anteriormente. Al decir personalizados hace referencia que los cuadros de los bordes pueden cambiar, así como también se puede incorporar imágenes en el código QR.

Para este proyecto se realizó cuatro tipos de códigos que representan el tipo de lugar turístico en la ciudad de Quito, dichos códigos son identificadores únicos para cada uno de los sitios turísticos, los cuales permiten acceder a información desde el dispositivo móvil. Los códigos QR fueron elaborados en la página web unitag.io, esta página permite la creación de QR personalizados por el usuario.

Como se especificó en los requerimientos de la aplicación móvil únicamente se registró 50 lugares turísticos, el Anexo 4 muestra los 50 códigos QR creados para este proyecto.

El primer tipo representa a las iglesias, este tipo de código QR se caracteriza por tener sus esquinas como la Figura 23.

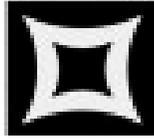


Figura 23 Esquina para el tipo de sitio iglesias

Elaborado por: Josselyn Rodríguez, Henry Rojas

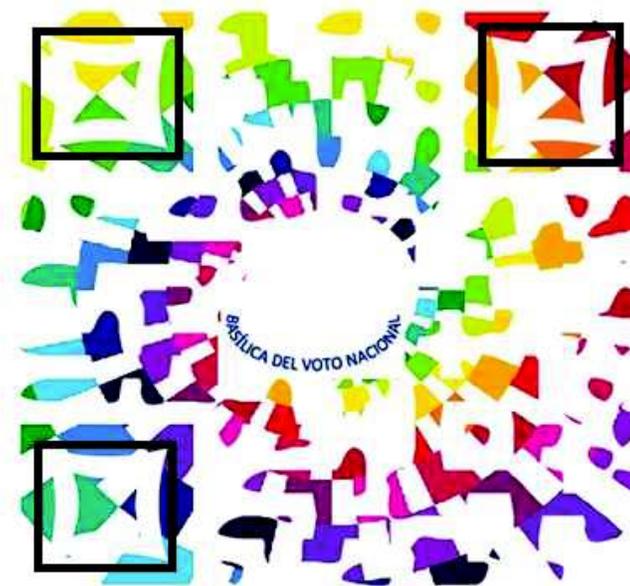


Figura 24 Código QR Basílica del Voto Nacional

Elaborado por: Josselyn Rodríguez, Henry Rojas

El segundo tipo representa los museos, este tipo de código QR se caracteriza por tener sus esquinas como la Figura 25.



Figura 25 Esquina para el tipo de sitio museos

Elaborado por: Josselyn Rodríguez, Henry Rojas

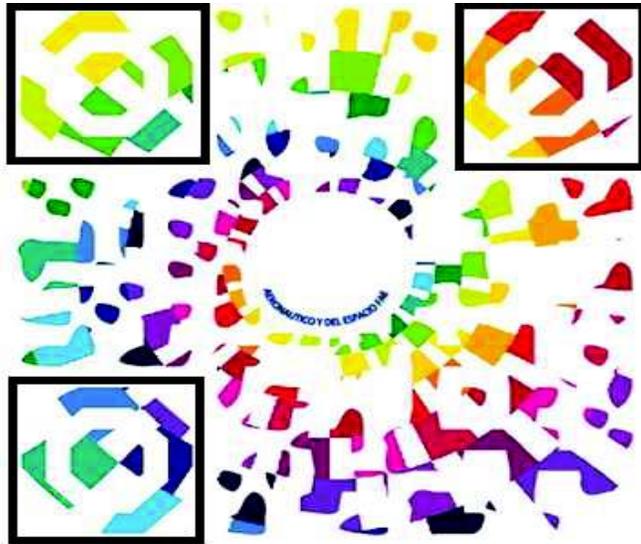


Figura 26 Código QR Museo Aeronáutico y del espacio FAE

Elaborado por: Josselyn Rodríguez, Henry Rojas

El tercer tipo representa los parques, este tipo de código QR se caracteriza por tener sus esquinas como la Figura 27.

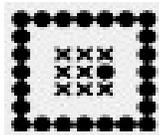


Figura 27 Esquina para el tipo de sitio parques

Elaborado por: Josselyn Rodríguez, Henry Rojas

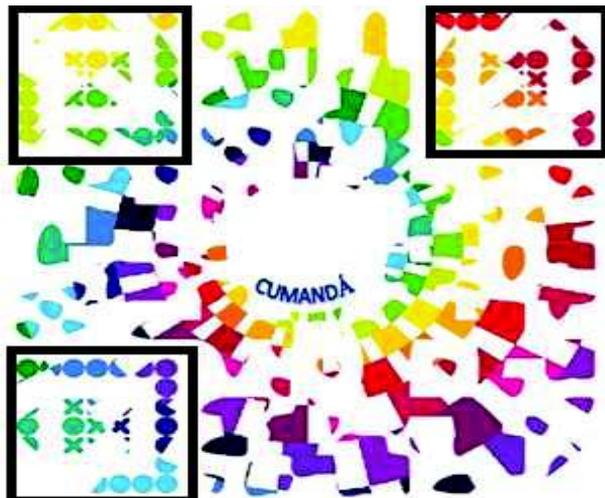


Figura 28 Código QR Parque Cumandá

Elaborado por: Josselyn Rodríguez, Henry Rojas

El cuarto tipo representa otros, este tipo de código QR se caracteriza por tener sus esquinas como la Figura 29.

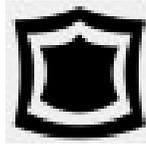


Figura 29 Esquina para el tipo de sitios otros

Elaborado por: Josselyn Rodríguez, Henry Rojas



Figura 30 Código QR Casa Cultural Trude Sojka

Elaborado por: Josselyn Rodríguez, Henry Rojas

3.11.2 Creación de base de datos para *image target*

Para acceder a una base de datos de Vuforia se obtuvo una cuenta en la página oficial de Vuforia *Developer Portal*: developer.vuforia.com/vui/user/register, esta cuenta permite descargar herramientas para desarrolladores, además de obtener las claves de licencia necesarias para utilizar realidad aumentada en la aplicación.

Una vez que obtenida dicha cuenta se creó un proyecto para obtener la clave de licencia, para la creación del proyecto se siguió los pasos que se especifican a continuación:

- Colocar un nombre para el proyecto.
- Aceptar los términos de licencia de Vuforia.

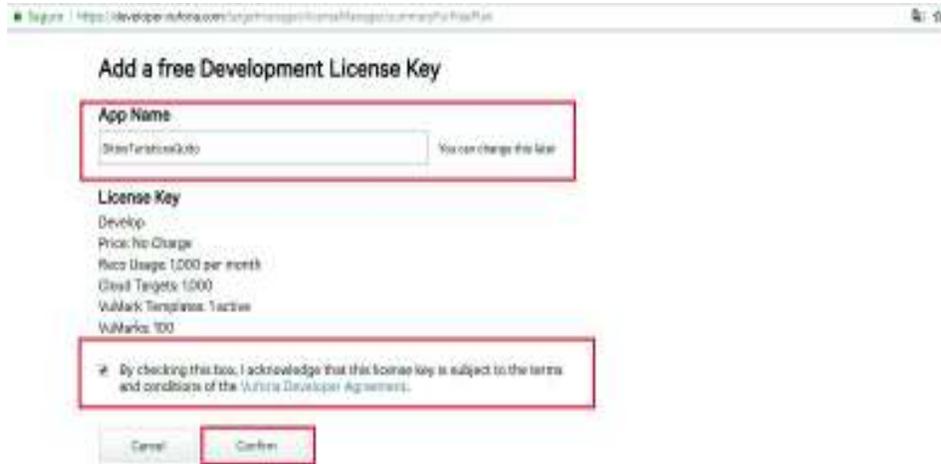


Figura 31 Creación de un proyecto en la cuenta de desarrollador de Vuforia

Elaborado por: Josselyn Rodríguez, Henry Rojas

Creación de *cloud database*

En la sección 2.5.3 de este documento se estableció por qué en este proyecto se debe utilizar una base de datos en la nube, por tal motivo para la creación de dicha base se siguió los pasos que se especifican a continuación:

- Dirigirse a la sección de *Target Manager*
- Añadir nueva base de datos (*Add Database*)
- Colocar un nombre para la base de datos.
- Seleccionar tipo de base de datos, en este caso se seleccionó *cloud*.
- Seleccionar la clave de licencia que fue creada anteriormente.



Figura 32 Sección *Target Manager*

Elaborado por: Josselyn Rodríguez, Henry Rojas

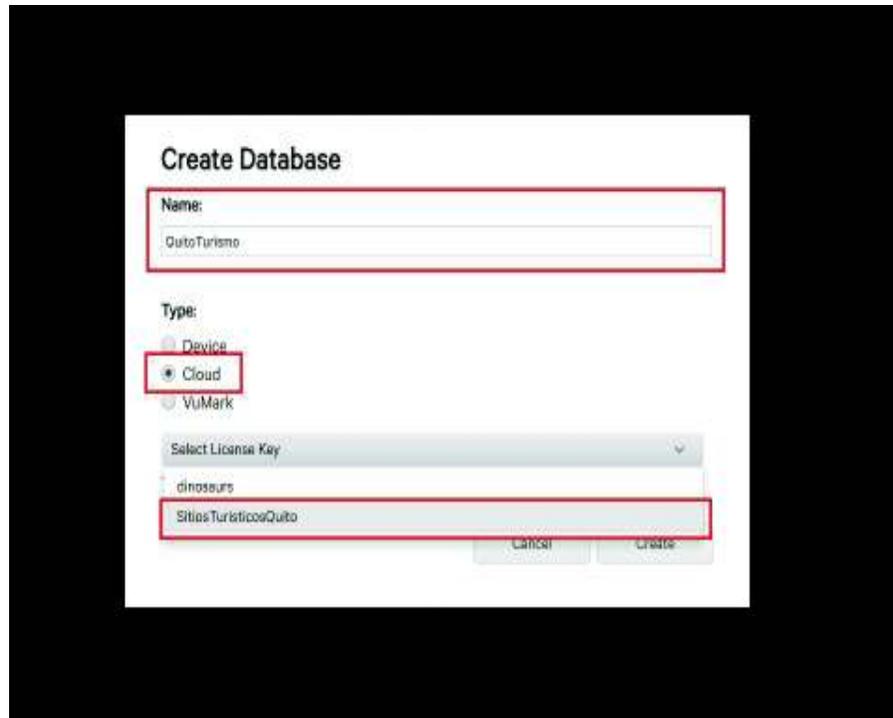


Figura 33 Creación *cloud data base*

Elaborado por: Josselyn Rodríguez, Henry Rojas

Ingresar *image target* en la *cloud database*

En la sección 3.11.1 de este documento se muestra como fueron creados los códigos QR que son las *images target* para cada uno de los sitios turísticos, dichos *images target* creados fueron subidos a la *cloud database*, para lo cual se siguió los pasos que se especifican a continuación:

- Dirigirse a la sección de *Target Manager*
- Seleccionar la base de datos creada previamente (QuitoTurismo) y presionar el botón *Add Target*. Véase Figura 27.
- Seleccionar la *image target* que se va a ingresar. Se debe ingresar el tamaño para obtener una escala adecuada del contenido que se va a mostrar. Véase Figura 28.
- Seleccionar un archivo .txt el cual contendrá la metadatata del *image target*. Para este proyecto se ha decidido utilizar como metadata el id de sitio turístico registrado en la página de administración ya que con dicho dato se facilitará la lectura de la información del sitio registrado en el servidor.



Figura 34 Ingresar *image target* a la *cloud data base*. Paso 1

Elaborado por: Josselyn Rodríguez, Henry Rojas

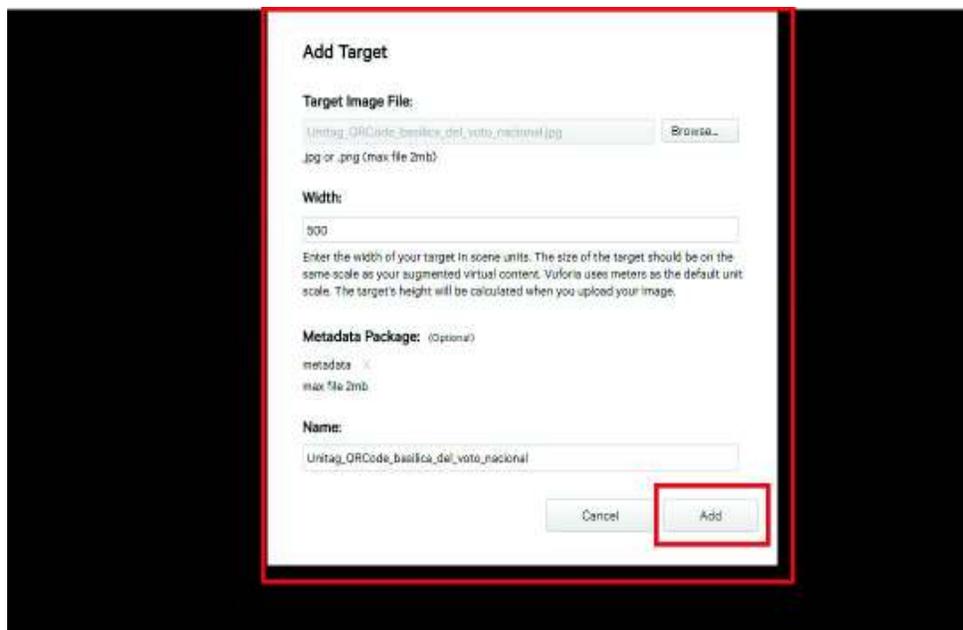


Figura 35 Ingresar *image target* a la *cloud database*. Paso 2

Elaborado por: Josselyn Rodríguez, Henry Rojas

Se debe aclarar que cada que se registre un sitio turístico en la página de administración se debe crear un nuevo *image target* y subir a la *cloud database*. En dicha página se puede obtener la metadata del sitio registrado.

3.11.3 Obtención de SDKs para la aplicación

Vuforia

El SDK de Vuforia para Unity se encuentra disponible en la página oficial de *Vuforia Developer Portal* developer.vuforia.com/downloads/sdk. El archivo descargado contiene todos los componentes necesarios propios de Vuforia para la implementación de realidad aumentada en el *framework* Unity.

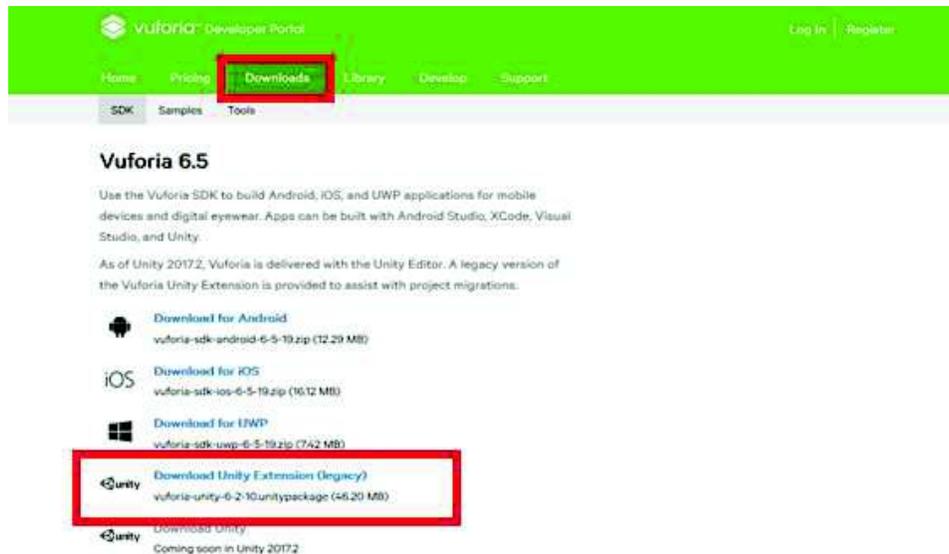


Figura 36 Descargar SDK para Unity

Elaborado por: Josselyn Rodríguez, Henry Rojas

Watson

El SDK de Watson se encuentra disponible en la página oficial github.com/watson-developer-cloud/unity-sdk. El archivo descargado contiene librerías necesarias para la traducción de texto.



Figura 37 Descargar SDK para traducción de texto

Elaborado por: Josselyn Rodríguez, Henry Rojas

3.11.4 Creación de proyecto en Unity

Para la creación de un nuevo proyecto en el *framework* Unity, se siguió los siguientes pasos:

- Ejecutar la aplicación de Unity
- Seleccionar nuevo proyecto *New*.
- Ingresar el nombre del Proyecto [QuitoSitosTuristicos]

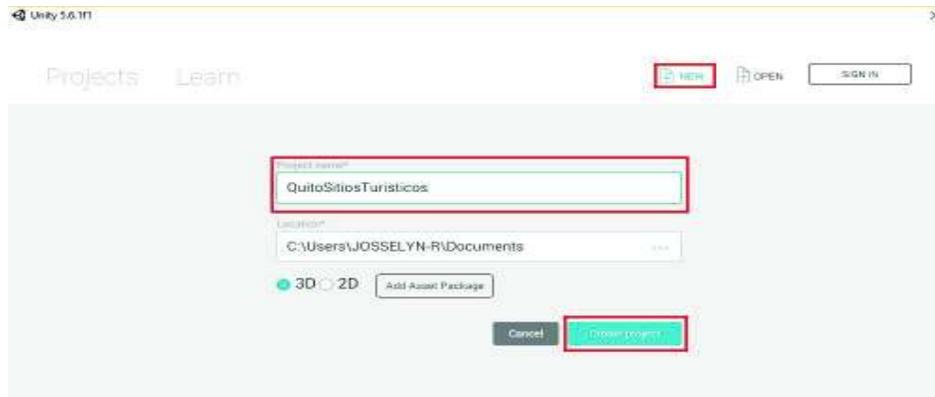


Figura 38 Creación de proyecto en Unity

Elaborado por: Josselyn Rodríguez, Henry Rojas

Importar SDKs

Se debe importar los SDKs previamente descargados y añadirlos al proyecto. Para importar se debe siguió los pasos que se detallan a continuación:

- En la barra de herramientas del proyecto seleccionar el ítem *Assets/Importar Package/Custom Package*. Véase Figura 32
- Seleccionar los SDKs descargados. Véase Figura 33.
- Seleccionar los paquetes que se van a utilizar en proyecto. Véase Figura 33.

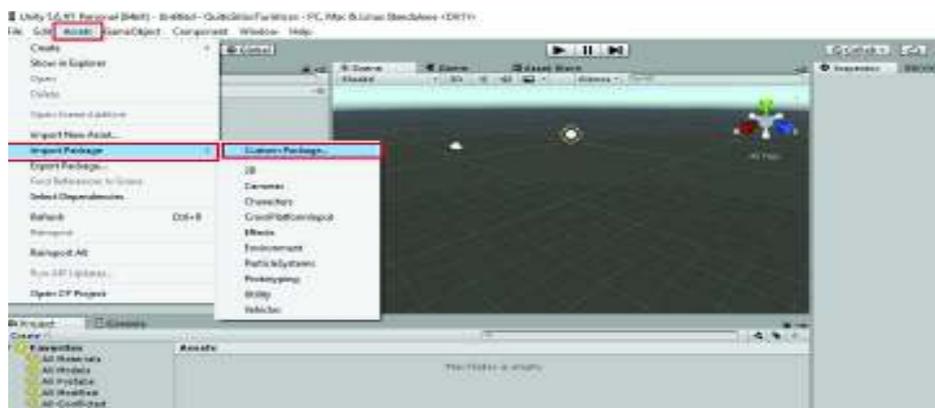


Figura 39 Importar SDK. Paso 1

Elaborado por: Josselyn Rodríguez, Henry Rojas

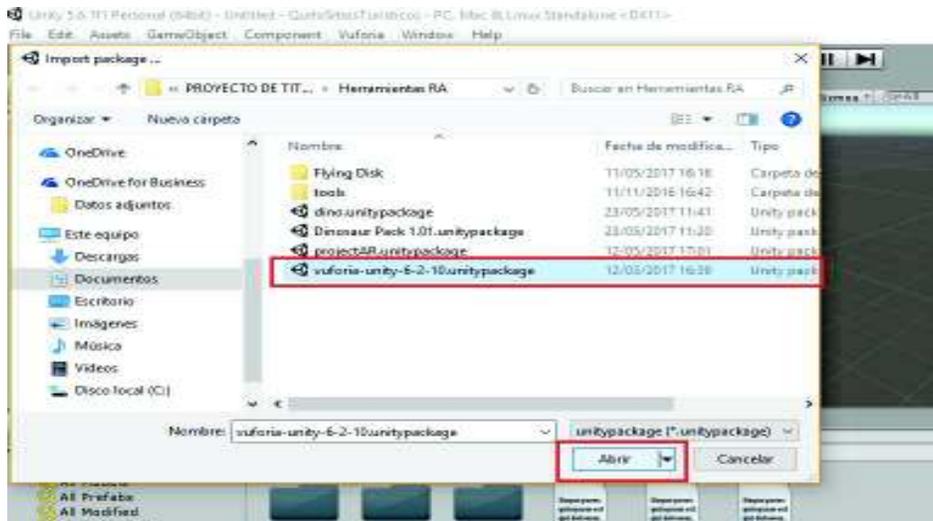


Figura 40 Importar SDK. Paso 2

Elaborado por: Josselyn Rodríguez, Henry Rojas

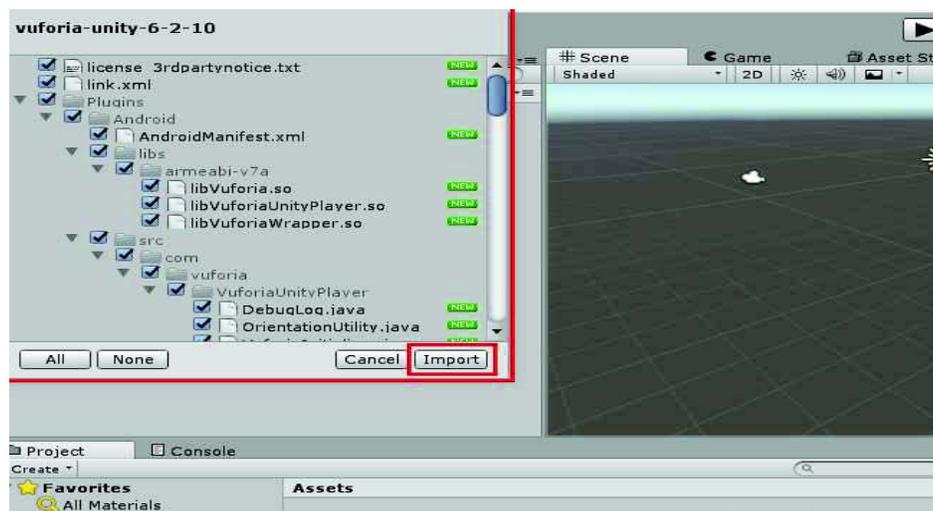


Figura 41 Importar SDK. Paso 3

Elaborado por: Josselyn Rodríguez, Henry Rojas

Implementación de *image target* con sus componentes

Para la implementación de realidad aumentada se utilizó *image target* que son *prefabs* propios de Vuforia SDK.

El principal *prefab* agregado al proyecto fue ARCamara que permitió el reconocimiento del *image target* y visualización de los componentes que se agregaron. Para agregar al proyecto se arrastró el *prefab* directamente a la escena desde la carpeta Assets/ Vuforia/ Prefabs.

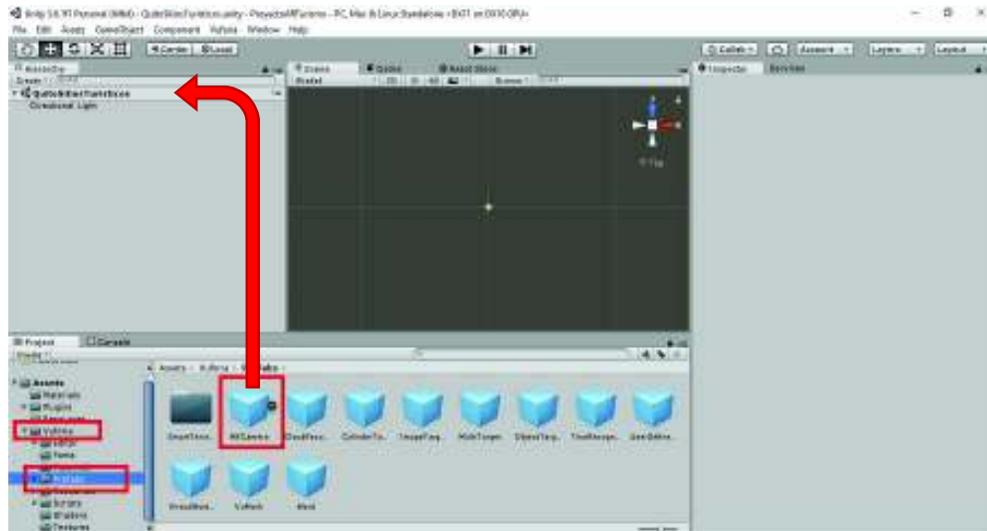


Figura 42 Añadir ARCamara a la escena.

Elaborado por: Josselyn Rodríguez, Henry Rojas

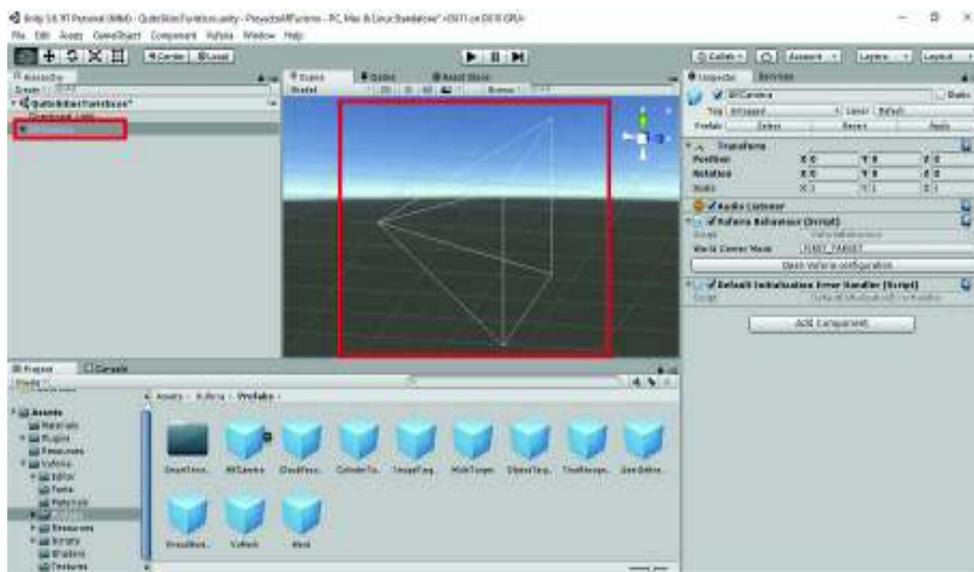


Figura 43 ARCamara añadida a la escena.

Elaborado por: Josselyn Rodríguez, Henry Rojas

ARCamara requiere añadir el *App License Key* presente en el proyecto Vuforia que fue creado anteriormente, en el componente *Vuforia Behaviour* contiene el archivo de configuración de Vuforia.

Para añadir se presionó el botón *open Vuforia configuration* y en el apartado de *App License Key*, se añadió el *App License Key* obtenido del proyecto Vuforia.

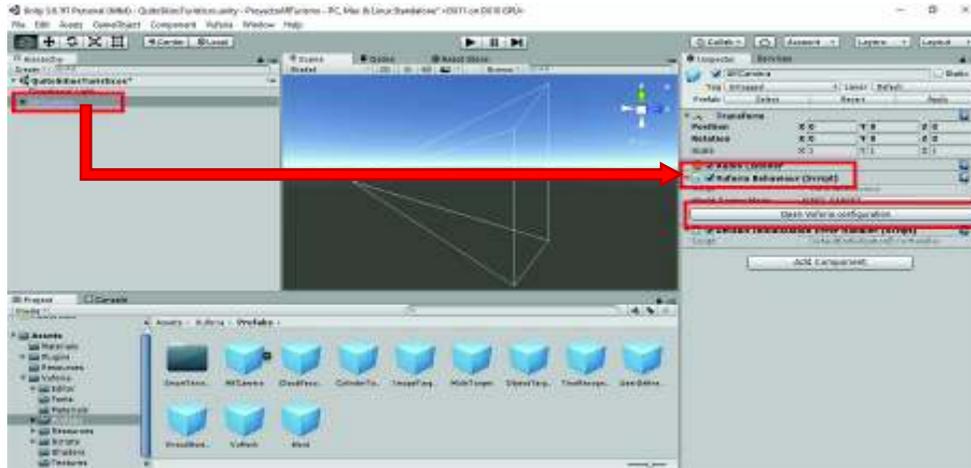


Figura 44 Componente Vuforia Behaviour presente en ARCamara

Elaborado por: Josselyn Rodríguez, Henry Rojas

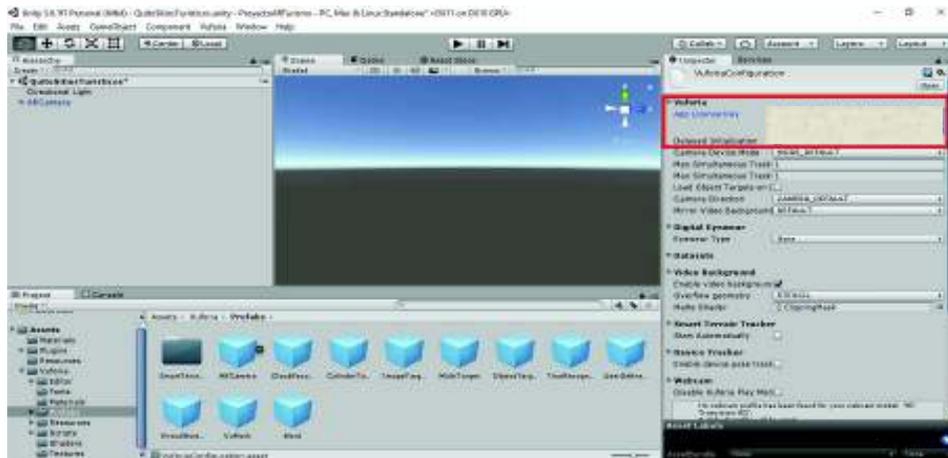


Figura 45 App Licence Key añadido.

Elaborado por: Josselyn Rodríguez, Henry Rojas



Figura 46 Prueba de ARCamara en Unity.

Elaborado por: Josselyn Rodríguez, Henry Rojas

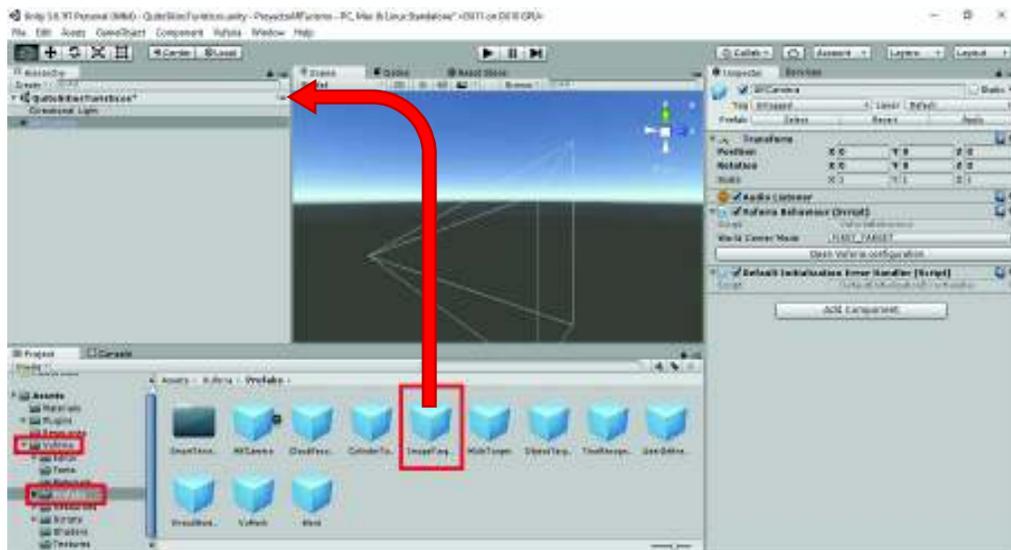


Figura 47 Añadir *image target* a la escena.

Elaborado por: Josselyn Rodríguez, Henry Rojas

Una vez añadido el *prefab* ARCamara, configurado y probado correctamente; se añadió el *prefab image target* en la escena, este *prefab* se encarga del reconocimiento y despliegue de los diferentes componentes añadidos a ARCamara, dependiendo de la imagen que se visualice desde la cámara.

De la misma forma como se agregó el *prefab* anterior se debe agregar el *image target* a la escena.

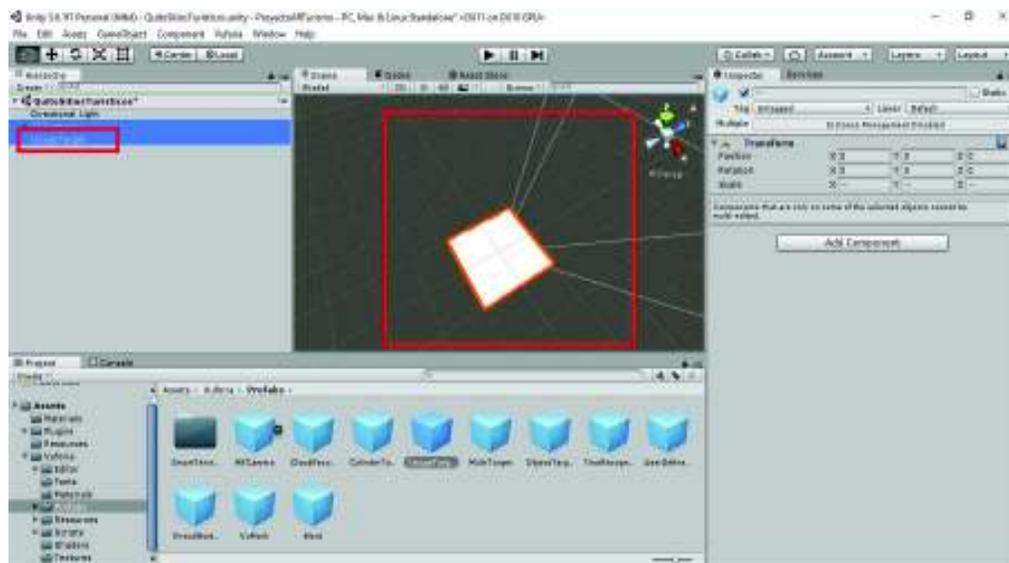


Figura 48 *Image target* añadido a la escena.

Elaborado por: Josselyn Rodríguez, Henry Rojas

Para añadir un componente, a un *image target* se siguió los pasos que se detallan a continuación:

- Seleccionar *image target* de la escena, clic derecho y en cuadro de opciones escoger el componente que se desea añadir, en este caso se añadió un *3D Text*.
- Se modifica su posición para tener una mejor visualización.

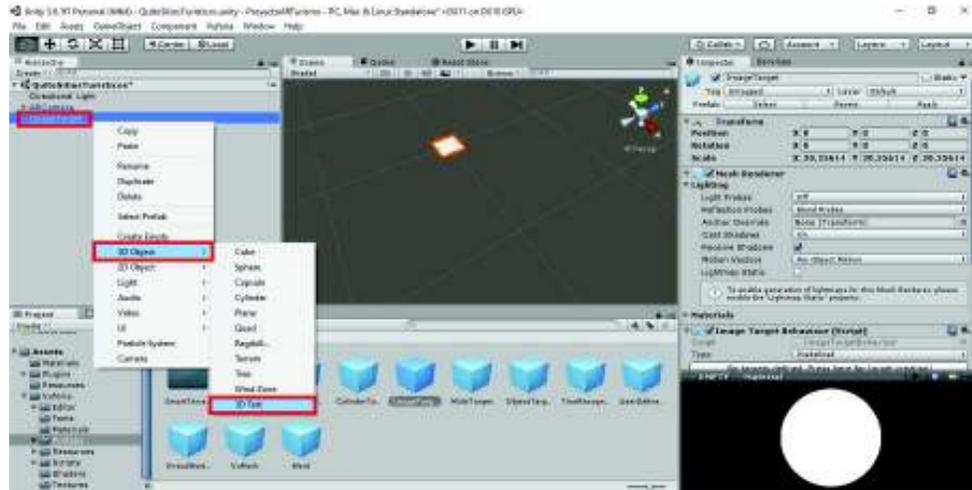


Figura 49 Creando componente dentro del *image target*.

Elaborado por: Josselyn Rodríguez, Henry Rojas

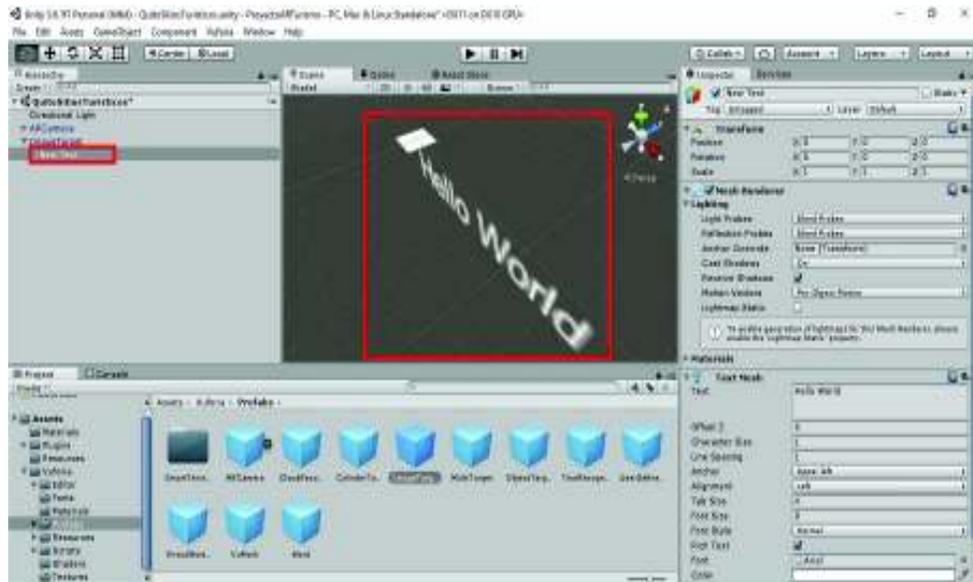


Figura 50 Visualización del componente 3D Text creado.

Elaborado por: Josselyn Rodríguez, Henry Rojas

Nota: Cada uno de los componentes que van a ser visualizados en el *image target*, deben ser añadidos dentro del *prefab*, de esta forma el funcionamiento de la aplicación no presentará problemas durante su ejecución.

Para la realización de este proyecto fue necesario el uso de *3D Object* como *3D Text* y *Plane*, los cuales permitieron la visualización de información del sitio turístico y de sus imágenes respectivamente.

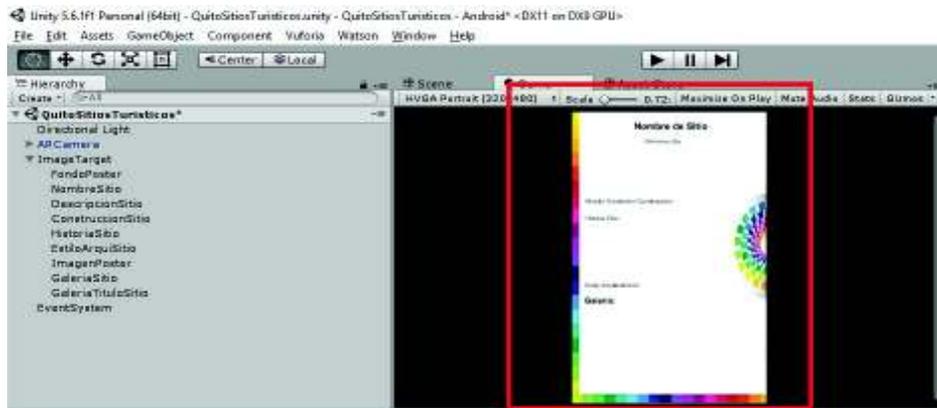


Figura 51 Diseño final de *image target* con sus componentes.

Elaborado por: Josselyn Rodríguez, Henry Rojas

Creación de Interfaces en Unity

Para la creación de interfaces se utilizó los componentes Canvas de Unity, que permitió añadir elementos como botones, texto, panel, dropdown, entre otros. Para añadir en la escena dichos elementos se siguió los pasos que detallan a continuación:

- Añadir un componente tipo Canvas desde la escena dando clic derecho y en cuadro de opciones la opción UI. Escoger el componente que se desea añadir y se agregará al Canvas creado automáticamente. Véase Figura 52.
- A este Canvas se lo configuró en uno de sus componentes para ser renderizado en la vista de la cámara y cada uno de sus elementos permanezca en su lugar. Véase Figura 53.



Figura 52 Añadir un componente Canvas a la escena.

Elaborado por: Josselyn Rodríguez, Henry Rojas



Figura 53 Configuración de *Render Mode* a *Screen Space – Camera*

Elaborado por: Josselyn Rodríguez, Henry Rojas

A partir de este Canvas se deben crear nuevos Canvas, para visualizarlos cuando sean necesarios.

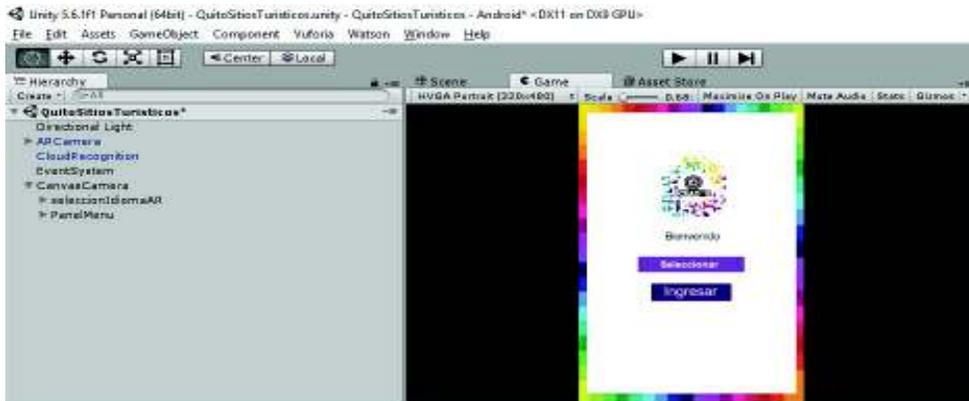


Figura 54 Visualización interfaz Menú Inicial

Elaborado por: Josselyn Rodríguez, Henry Rojas

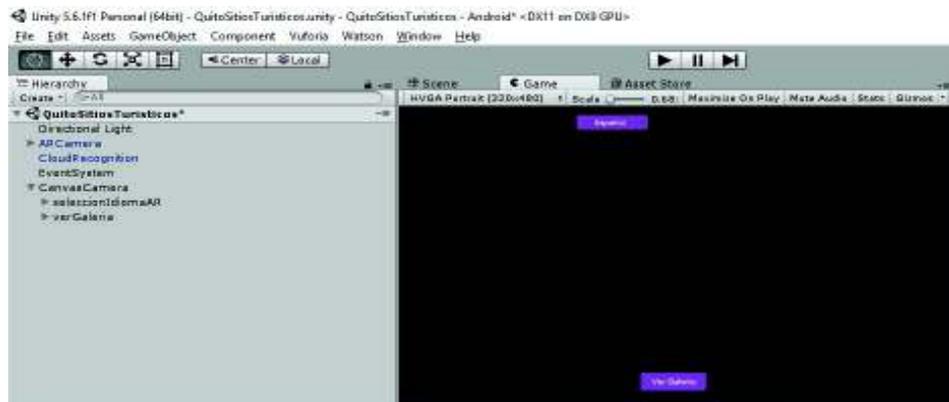


Figura 55 Visualización interfaz vista de cámara

Elaborado por: Josselyn Rodríguez, Henry Rojas

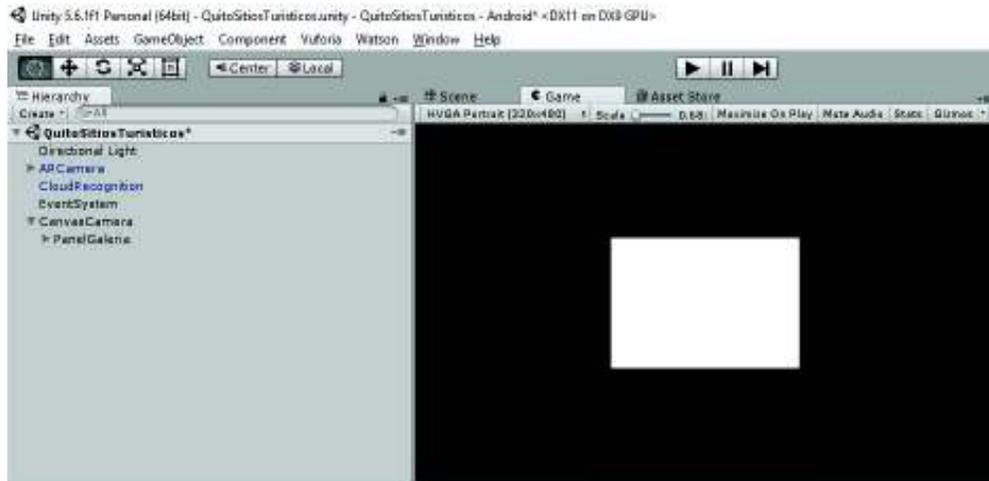


Figura 56 Visualización interfaz galería de imágenes en vista de cámara

Elaborado por: Josselyn Rodríguez, Henry Rojas

Implementación de CloudRecognition

Para la codificación del proyecto, se requirió del *prefab* *CloudRecognition*, que permite realizar consultas a la base de datos de los *images target* creados para el proyecto Vuforia, previamente creado detallado en la sección 3.11.2. Se debe arrastrar el *prefab* directamente a la escena desde la carpeta Assets/ Vuforia/ Prefabs.

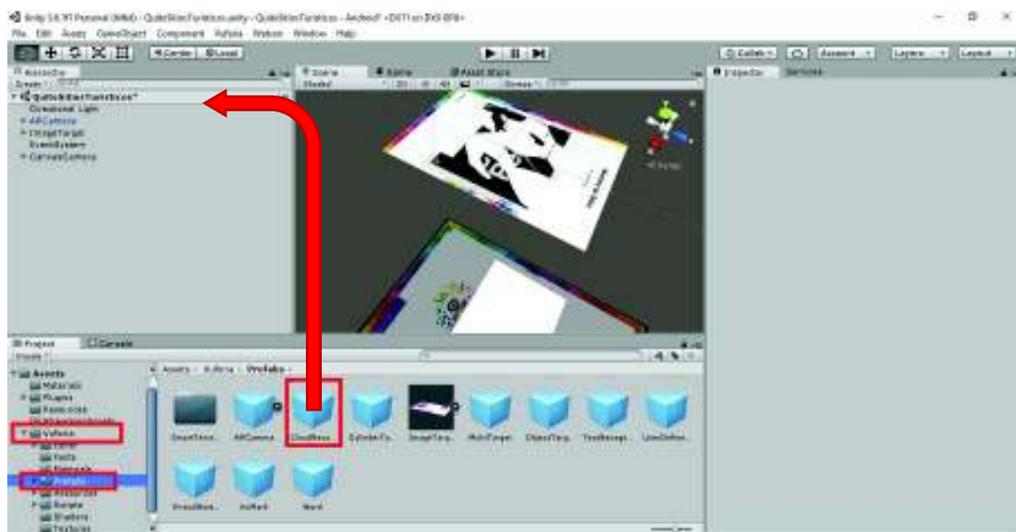


Figura 57 Colocar *CloudRecognition* prefab a la escena.

Elaborado por: Josselyn Rodríguez, Henry Rojas

A dicho componente se añadió el *Access* y *Secret Key*, son los permisos que tendrá la aplicación para autenticarse con la *Cloud Reco Service* y acceder a la base de datos, de forma segura. Para obtener estas se siguió los pasos que se especifican a continuación:

- Dirigirse a la sección de *Target Manager*
- Seleccionar la base de datos que fue creada en la sección 3.11.2.
- Dirigirse a la sección *Database Access Keys*.
- Copiar Access Key y Secret Key al proyecto en Unity.

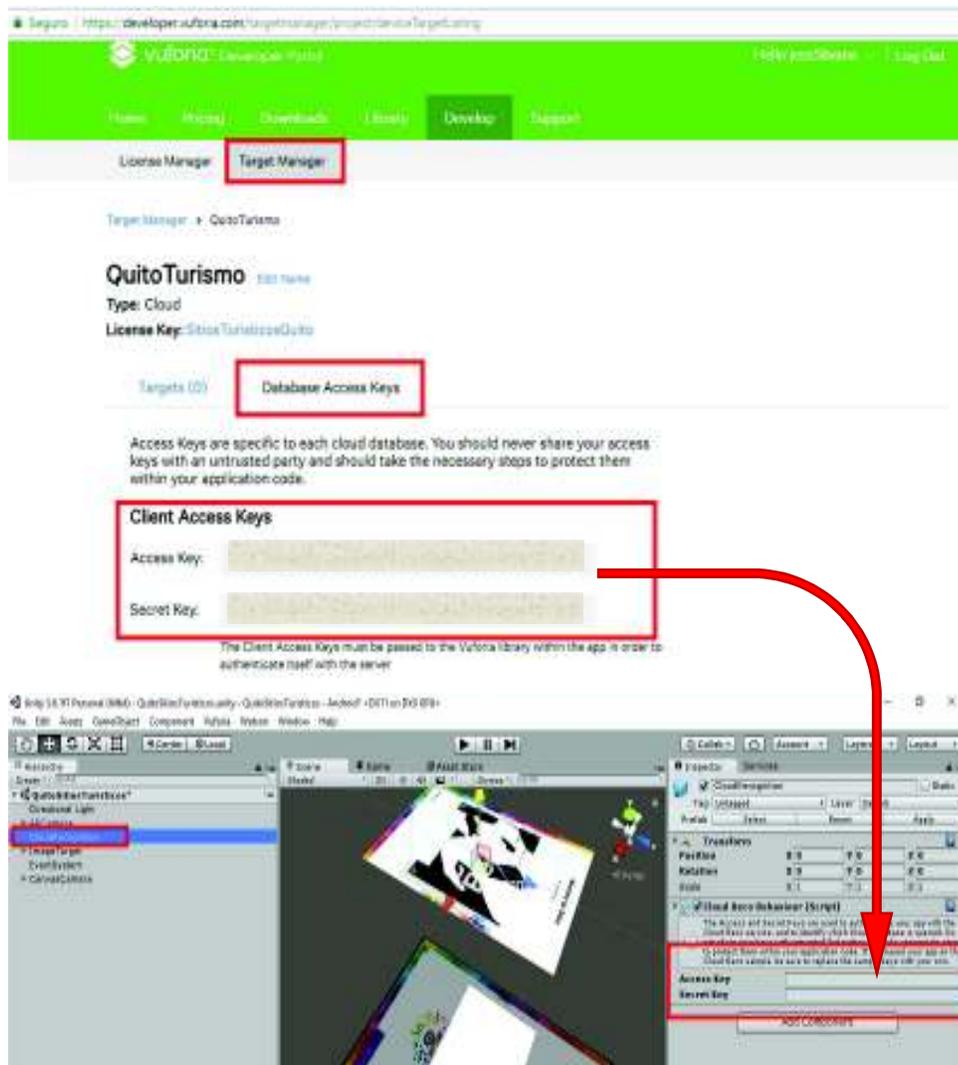


Figura 58 Colocar Access Key y Secret Key al CloudRecognition.

Elaborado por: Josselyn Rodríguez, Henry Rojas

Finalmente, se añadió un *Script dentro del prefab CloudRecognition*, el cual contiene la funcionalidad de la aplicación móvil. Se siguió los siguientes pasos:

- Presionar el botón *Add Component* en el *prefab* creado anteriormente.
- En el cuadro de opciones seleccionamos la opción *New Script*.
- Se ingresa un nombre al nuevo *Script* y se selecciona el lenguaje.
- Finalmente presionar el botón *Create and Add*.

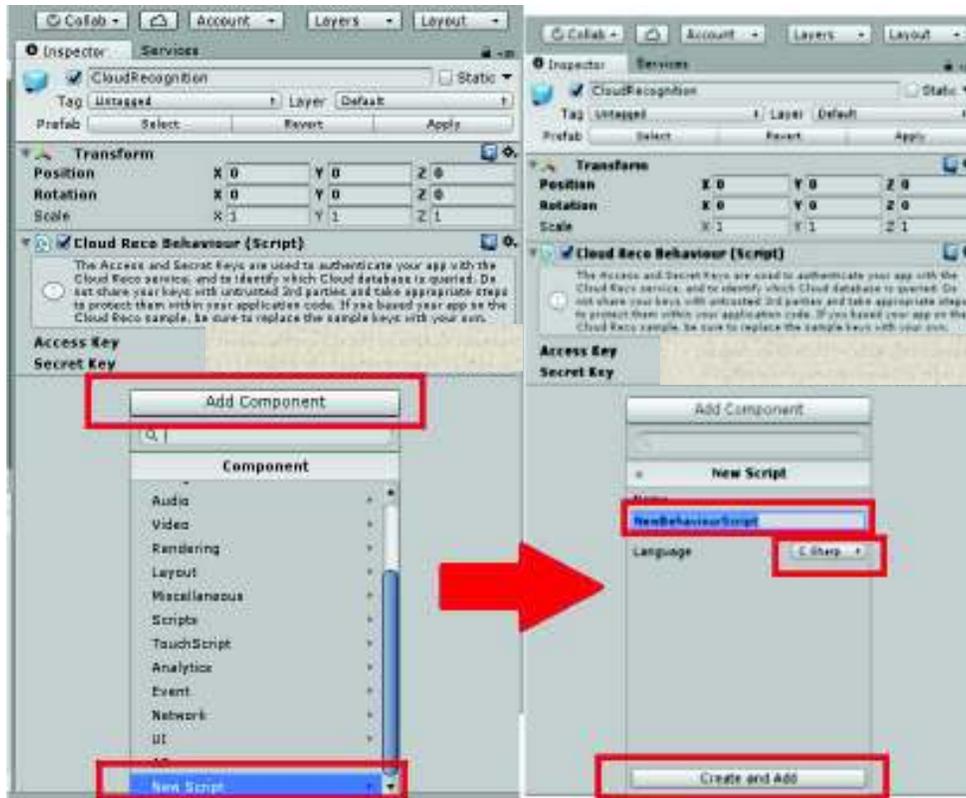


Figura 59 Crear Script y añadir a CloudRecognition

Elaborado por: Josselyn Rodríguez, Henry Rojas

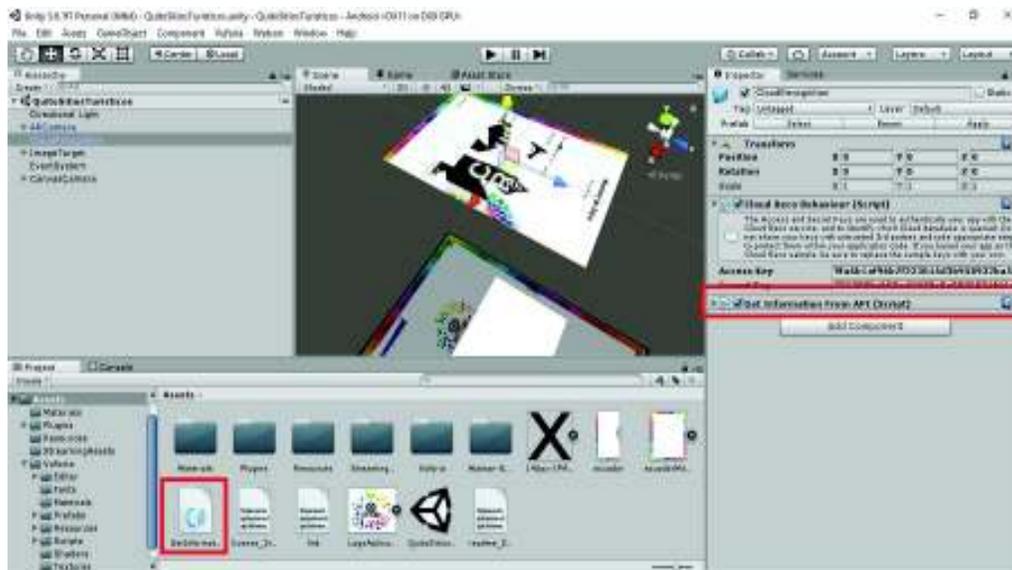


Figura 60 Script añadido al proyecto y al prefab CloudRecognition.

Elaborado por: Josselyn Rodríguez, Henry Rojas

Una vez finalizada la codificación del *Script*, se mostró un ejemplo de la utilización de la aplicación móvil. Véase Anexo 5.

3.12 Pruebas de la página de administración

3.12.1 Pruebas de software

Las pruebas de la página de administración de los sitios turísticos fueron realizadas en varios navegadores para comprobar su correcto funcionamiento, los cuales se muestran en la Tabla 23.

Tabla 23

Características de los navegadores

Navegador	Versión del navegador
Chrome	61.0.3163.100
FireFox	56.0
Safari	11.0

Elaborado por: Josselyn Rodríguez y Henry Rojas

3.12.2 Pruebas de aceptación

Las pruebas de aceptación fueron realizadas para comprobar la funcionalidad de la página de administración de los sitios turísticos descritas en las historias de usuario. Estas pruebas se ejecutaron para mostrar todos los casos posibles permitidos en la página de administración.

Tabla 24

Prueba de aceptación: Identificar usuario

Prueba de aceptación	
Identificador (ID): 1	Identificador historia de usuario: 1
Nombre prueba de aceptación: Identificar usuario	
Descripción: El usuario administrador debe ingresar usuario y contraseña.	
Pasos de ejecución: Ingresar usuario y contraseña. Ingresar contraseña. Presionar el botón ingresar.	
Resultado esperado: La página de administración debe reconocer al usuario y redirigir a la página principal de los sitios turísticos.	

<p>Evaluación de la prueba:</p> <p>Se mostró el resultado esperado.</p>
--

Elaborado por: Josselyn Rodríguez y Henry Rojas

Tabla 25

Prueba de aceptación: Registrar nuevo sitio turístico

Prueba de aceptación	
Identificador (ID): 2	Identificador historia de usuario: 2
Nombre prueba de aceptación: Registrar nuevo sitio turístico	
Descripción: El usuario administrador debe registrar un nuevo sitio turístico.	
Pasos de ejecución: Ingresar en los campos de texto: nombre, descripción e historia del sitio turístico. Seleccionar el tipo, período de construcción, estilo arquitectónico del sitio turístico. Seleccionar las imágenes correspondientes al sitio turístico. Presionar el botón registrar.	
Resultado esperado: Esperar que el sitio sea registrado en el servidor y se muestre en la sección de portafolio de la página.	
Evaluación de la prueba: Se mostró el resultado esperado.	

Elaborado por: Josselyn Rodríguez y Henry Rojas

Tabla 26

Prueba de aceptación: Modificar sitio turístico

Prueba de aceptación	
Identificador (ID): 3	Identificador historia de usuario: 3
Nombre prueba de aceptación: Modificar sitio turístico	
Descripción: El usuario administrador puede modificar información de un sitio turístico registrado.	

<p>Pasos de ejecución:</p> <p>Buscar en la sección portafolio el sitio que se va a modificar.</p> <p>Presionar el botón editar para mostrar un modal con la información del sitio seleccionado.</p> <p>Modificar los datos del sitio.</p> <p>Presionar el botón modificar.</p>
<p>Resultado esperado:</p> <p>Esperar que el sitio registrado actualice la información en el servidor.</p>
<p>Evaluación de la prueba:</p> <p>Se mostró el resultado esperado.</p>

Elaborado por: Josselyn Rodríguez y Henry Rojas

Tabla 27

Prueba de aceptación: Eliminar sitio turístico

Prueba de aceptación	
Identificador (ID): 4	Identificador historia de usuario: 4
Nombre prueba de aceptación: Eliminar sitio turístico	
Descripción:	
El usuario administrador puede eliminar un sitio turístico registrado.	
Pasos de ejecución:	
Buscar en la sección portafolio el sitio que se va a eliminar.	
Presionar el botón eliminar para mostrar un modal de confirmación.	
Confirmar si desea eliminar ese sitio turístico presionando el botón eliminar.	
Resultado esperado:	
Esperar que el sitio registrado sea borrado del servidor.	
Evaluación de la prueba:	
Se mostró el resultado esperado.	

Elaborado por: Josselyn Rodríguez y Henry Rojas

3.13 Pruebas de aplicación móvil

3.13.1 Pruebas de software

Las pruebas de la aplicación fueron realizadas en dispositivos móviles físicos para comprobar su correcto funcionamiento. En la Tabla 28 se muestra los dispositivos que se utilizó para la plataforma Android y en la Tabla 29 se muestra los dispositivos que se utilizó para la plataforma iOS.

Tabla 28

Características de los dispositivos móviles para plataforma Android

Modelo del dispositivo	Versión del sistema operativo
Huawei Y5 II	5.1
Sony Z5	7.0
Samsung J5	5.1
Samsung G130H	4.4.2

Elaborado por: Josselyn Rodríguez y Henry Rojas

Tabla 29

Características de los dispositivos móviles para plataforma iOS

Modelo del dispositivo	Versión del sistema operativo
Iphone 6	11.0.1
Iphone 4S	9.3.5
Ipad	10.3.3

Elaborado por: Josselyn Rodríguez y Henry Rojas

3.13.2 Pruebas de aceptación

Las pruebas de aceptación fueron realizadas para comprobar la funcionalidad de la aplicación móvil descritas en las historias de usuario. Estas pruebas se ejecutaron para mostrar todos los casos posibles permitidos en la aplicación.

Tabla 30

Prueba de aceptación: Seleccionar idioma

Prueba de aceptación	
Identificador (ID): 5	Identificador historia de usuario: 5
Nombre prueba de aceptación: Seleccionar idioma	
Descripción: El usuario debe seleccionar el idioma de preferencia	

<p>Pasos de ejecución:</p> <p>Seleccionar el idioma (inglés-español).</p> <p>Presionar el botón Ingresar.</p>
<p>Resultado esperado:</p> <p>Mostrar la cámara del dispositivo con el idioma seleccionado.</p>
<p>Evaluación de la prueba:</p> <p>Se mostró el resultado esperado.</p>

Elaborado por: Josselyn Rodríguez y Henry Rojas

Tabla 31

Prueba de aceptación: Mostrar información del sitio turístico

Prueba de aceptación	
Identificador (ID): 6	Identificador historia de usuario: 6
Nombre prueba de aceptación: Mostrar información del sitio turístico	
Descripción: El usuario debe enfocar con la cámara del dispositivo al código QR.	
Pasos de ejecución: Localizar el código QR en la fachada del sitio turístico. Enfocar con la cámara al código QR. Esperar que la información se cargue.	
Resultado esperado: Mostrar la información en el idioma seleccionado del sitio turístico visitado.	
Evaluación de la prueba: Se mostró el resultado esperado.	

Elaborado por: Josselyn Rodríguez y Henry Rojas

Tabla 32

Prueba de aceptación: Mostrar galería de imágenes del sitio turístico

Prueba de aceptación	
Identificador (ID): 7	Identificador historia de usuario: 7
Nombre prueba de aceptación: Mostrar galería de imágenes del sitio turístico	

<p>Descripción:</p> <p>El usuario debe enfocar con la cámara del dispositivo al código QR.</p>
<p>Pasos de ejecución:</p> <p>Localizar el código QR en la fachada del sitio turístico.</p> <p>Enfocar con la cámara al código QR.</p> <p>Esperar que las imágenes se carguen.</p>
<p>Resultado esperado:</p> <p>Mostrar la galería de imágenes del sitio turístico visitado.</p>
<p>Evaluación de la prueba:</p> <p>Se mostró el resultado esperado.</p>

Elaborado por: Josselyn Rodríguez y Henry Rojas

Tabla 33

Prueba de aceptación: Ocultar información del sitio turístico

Prueba de aceptación	
Identificador (ID): 8	Identificador historia de usuario: 8
Nombre prueba de aceptación: Ocultar información del sitio turístico	
<p>Descripción:</p> <p>El usuario podrá visualizar las imágenes en la pantalla completa del dispositivo.</p>	
<p>Pasos de ejecución:</p> <p>Localizar el código QR en la fachada del sitio turístico.</p> <p>Enfocar con la cámara al código QR.</p> <p>Esperar que las imágenes se carguen.</p> <p>Presionar el botón ver galería</p>	
<p>Resultado esperado:</p> <p>Mostrar las imágenes del sitio turístico visitado en la pantalla completa del dispositivo.</p>	
<p>Evaluación de la prueba:</p> <p>Se mostró el resultado esperado.</p>	

Elaborado por: Josselyn Rodríguez y Henry Rojas

CAPÍTULO IV

4. CONCLUSIONES Y RECOMENDACIONES

4.1 Conclusiones

En la recolección de información de sitios turísticos de la capital se pudo descubrir que existe una gran variedad de sitios poco conocidos los cuales cuentan con su propia historia y atractivos, mismos que contribuirán para impulsar el turismo a través de la aplicación móvil en el distrito metropolitano de Quito entre turistas nacionales y extranjeros.

La creación de códigos QR facilitó el reconocimiento de *images target* en los dispositivos móviles, desplegando de manera rápida la visualización de la realidad aumentada de los diferentes elementos creados en la aplicación móvil.

La utilización de *Sails* como servidor para la aplicación móvil y página administrativa permitió la creación de una API simple y fácil de mantener, al proveer de una capa de abstracción que facilita los procesos CRUD y a su vez el consumo de información. Al manejar la base de datos mediante una capa de abstracción facilita el diseño de tablas denominadas modelos.

El proyecto se llevó a cabo con la metodología ágil scrum, la cual permitió llevar a cabo entregas del proyecto en los tiempos establecidos. La aplicación móvil fue implementada con el *framework* Unity que ofrece una única compilación, lo cual significa reducir costos y tiempo de desarrollo. La utilización de este *framework* facilitó el alcance de los objetivos propuestos en el proyecto.

Vuforia SDK para Unity facilitó la implementación de realidad aumentada, permitiendo reconocimiento de *images target* en la nube, la creación y despliegue de objetos 3D en la aplicación móvil.

Se utilizó *Cloud Recognition* de Vuforia, debido a la escalabilidad y la capacidad de almacenamiento que podría tener el proyecto ya que en el distrito metropolitano de Quito no existen únicamente 50 sitios turísticos. Al utilizar este tipo de base de datos para *images target* la aplicación móvil proporcionará información actualizada a los usuarios, así como también nuevos sitios turísticos registrados en la misma.

Para comprobar la funcionalidad de la aplicación móvil se llevaron a cabo pruebas en dispositivos móviles físicos los cuales permitieron verificar el correcto funcionamiento de la misma, también se realizaron pruebas de aceptación de software para comprobar que se cumplió con todos los requisitos propuestos en las historias de usuario.

4.2 Recomendaciones

El usuario que administre la aplicación móvil debe consultar la información y las imágenes de sitios oficiales o de libros relacionados con la historia de Quito, ya que de esta manera se puede asegurar que la información de dicho sitio sea confiable y verídica.

Para la creación de un proyecto con realidad aumentada utilizando Vuforia SDK para Unity se debe considerar la escalabilidad de la aplicación, ya que de esto dependerá el tipo de licencia y a su vez el tipo de tecnología que se debe utilizar.

Si se desea realizar mejoras en la aplicación móvil, se debe considerar las posibles herramientas que provee Vuforia al momento, así como posibles actualizaciones, debido a que la realidad aumentada con Vuforia podría implementar más funciones que no fueron consideradas mientras se realizaba este proyecto.

Se debe considerar mejorar la interfaz de usuario con la que cuenta la aplicación, ya que se enfatizó en la funcionalidad de mostrar información en realidad aumentada, y se dejó de lado la experiencia del usuario final.

El consumo de datos móviles en la aplicación es un factor que debe ser tomado en cuenta y encontrar algún método efectivo de obtener un consumo mínimo y así evitar que el usuario prescinda de su uso.

Durante la carrera nunca se contempló una asignatura o un tema sobre realidad aumentada misma que hubiese permitido conocer herramientas exclusivas y que facilitarían aspectos como diseño y funcionalidad durante la creación de aplicaciones móviles.

REFERENCIAS BIBLIOGRÁFICAS

- Jackson, W. (2011). *Android Apps for Absolute Beginners*. New York: Apress.
- Landau, T. (2009). *Take Control of iPhone OS 3*. New York: TidBITS Publishing Inc.
- McNeil M & Nathan I. (2017) *Sails.js in Action*. Manning Publications
- Monte, J. (2016). *Implantar scrum con éxito*. Barcelona: Editorial UOC.
- Pressman, R. (2010). *Ingeniería del software*. 7ma ed. México: McGraw-Hill.
- AngularJS (2017). *Docs.angularjs.org*. Recuperado el 13 de Junio 2017 Obtenido de: <https://docs.angularjs.org/guide/>
- Codeanywhere. (2017). *Docs.codeanywhere.com*. Recuperado el 18 Mayo 2017, Obtenido de <http://docs.codeanywhere.com/whatiscodeanywhere.html>
- Company, T. (2017). *Sails.js | Realtime MVC Framework for Node.js*. *Sailsjs.com*. Recuperado el 18 Mayo 2017, Obtenido de <http://sailsjs.com/>
- Ecuadorinmediato.com (2012) *Exposición fotográfica "Miradas sobre Quito" de los años XIX y XX se inaugurará en bulevar Naciones Unidas*. Recuperado el 18 Mayo 2017, Obtenido de http://ecuadorinmediato.com/index.php?module=Noticias&func=news_user_view&id=170455&umt=exposicion_fotografica_miradas_sobre_quito_anos_xix_y_xx_se_inaugurara_en_bulevar_naciones_unidas
- Library.vuforia.com (2016) *Image Targets | Vuforia Library*. Recuperado el 18 Mayo 2017, Obtenido de <https://library.vuforia.com/articles/Training/Image-Target-Guide>
- SoMaMFyC (2017). *¿Qué es un código QR?. Definición y estructura*. Recuperado el 4 de Octubre 2017, Obtenido de: <https://nuevatecsomamfyc.wordpress.com/2012/05/08/que-es-un-codigo-qr-definicion-y-estructura/>
- Quito Adventure (s.f) *Museo de la Ciudad - Lugares Turísticos de Quito - Quito Adventure: Encuentra tu aventura en Ecuador y Galápagos*. Recuperado el 18 Mayo 2017, Obtenido de <http://www.quitoadventure.com/espanol/relax-ecuador/lugares-turisticos-quito/museos/museo-ciudad-quito.html>
- Quito Turismo (2014) *Quito Contemporáneo*. Recuperado el 18 Mayo 2017, Obtenido de <http://www.quito.com.ec/la-ciudad/sobre-quito/historia-de-quito/quito-contemporaneo>

Quito, P., & Cifras, Q. (2017). *Principales Indicadores de Quito*. *Quito-turismo.gob.ec*. Recuperado el 18 Mayo 2017, Obtenido de <http://www.quito-turismo.gob.ec/estadisticas/datos-turisticos-principales/category/82-quito-en-cifras>

Unity T. (2017). *Unity*. Recuperado el 18 Mayo 2017, Obtenido de <https://unity3d.com/es/unity>

Vuforia Developer Portal (2017). *Developer.vuforia.com*. Recuperado el 18 Mayo 2017, Obtenido de <https://developer.vuforia.com/>

ANEXO 1: Prototipo página de administración

Página: iniciar sesión

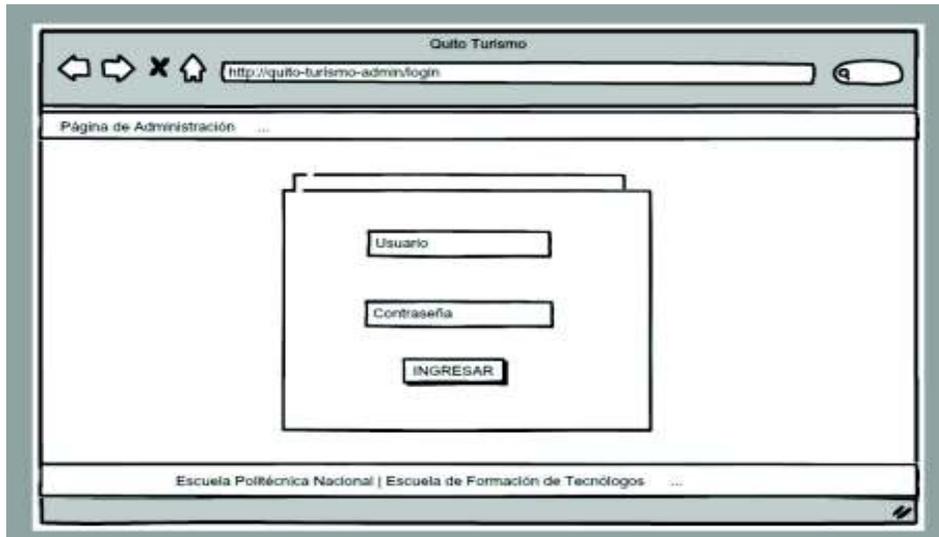


Figura 1 Página: iniciar sesión

Elaborado por: Josselyn Rodríguez, Henry Rojas

Página: administración de sitios

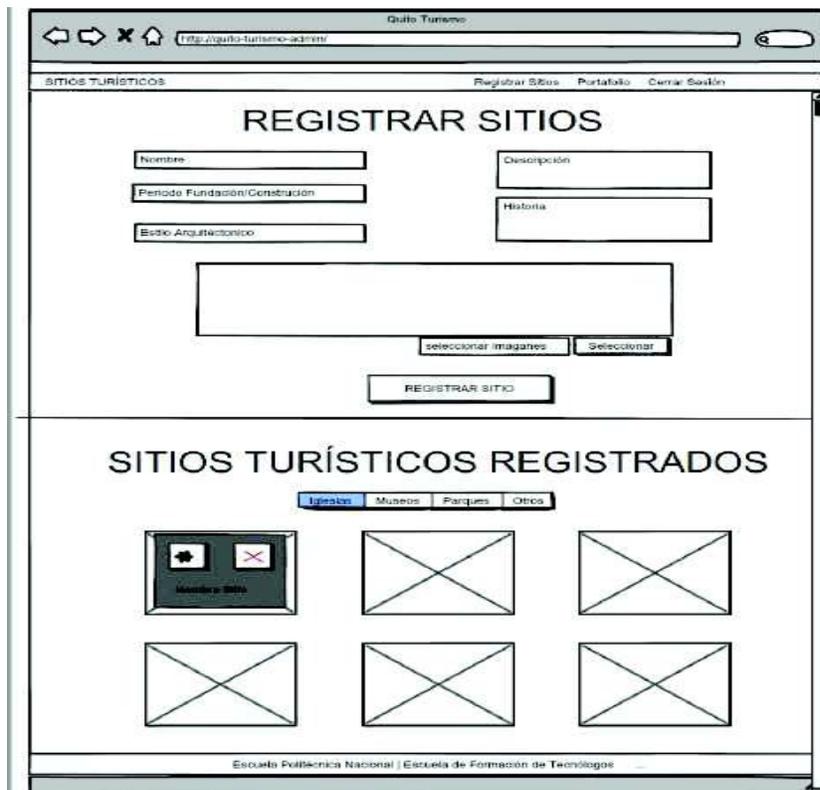


Figura 2 Página: administración de sitios

Elaborado por: Josselyn Rodríguez, Henry Rojas

Modal: modificar sitio turístico

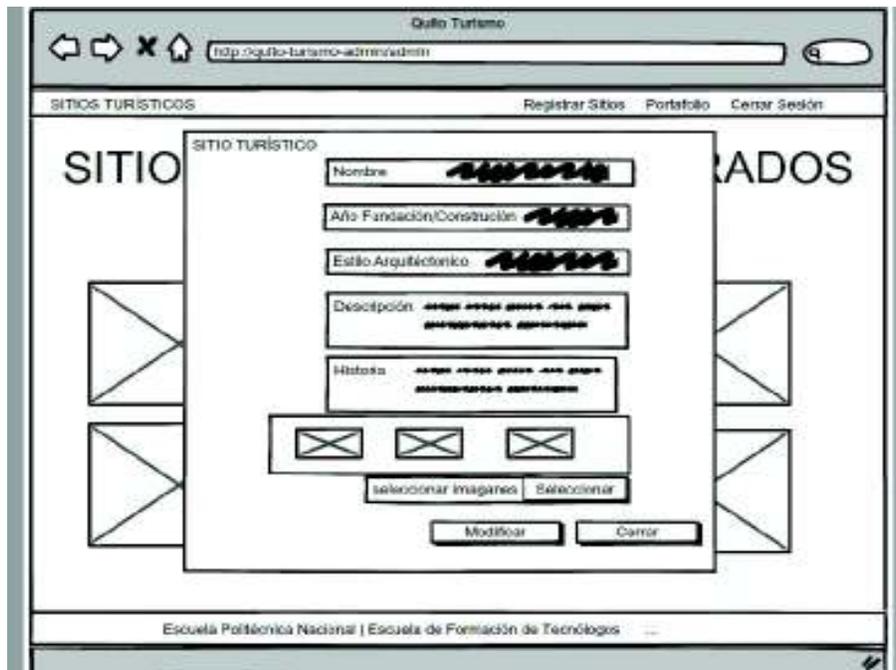


Figura 3 Modal: modificar sitio turístico

Elaborado por: Josselyn Rodríguez, Henry Rojas

Modal: verificar para eliminar sitio turístico

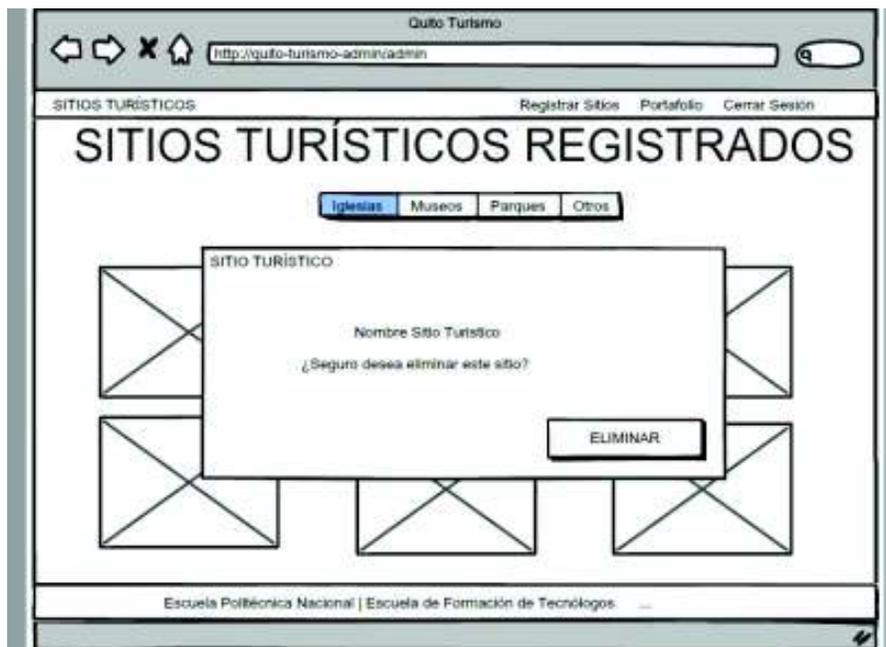


Figura 4 Modal: verificar para eliminar sitio turístico

Elaborado por: Josselyn Rodríguez, Henry Rojas

ANEXO 2: Manual de Usuario de la página de administración

Página: iniciar sesión

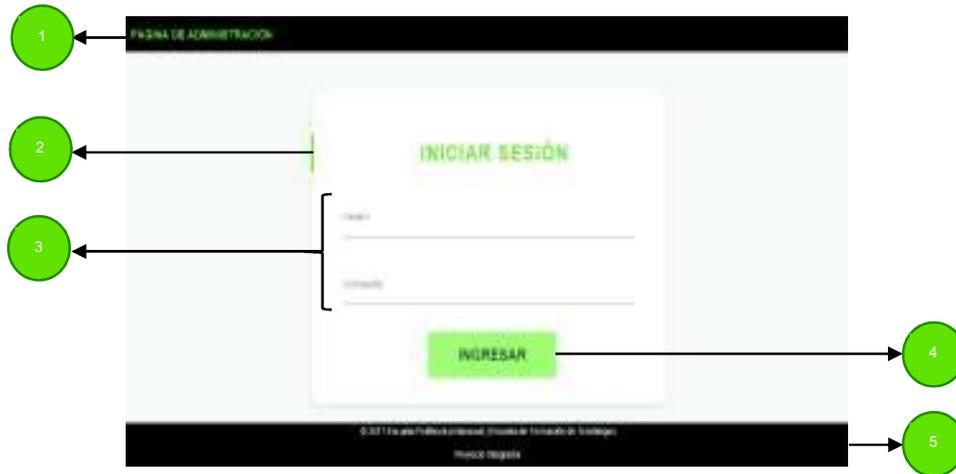


Figura 1 Página: iniciar sesión

Elaborado por: Josselyn Rodríguez, Henry Rojas

Identificación de las partes de la página de iniciar sesión:

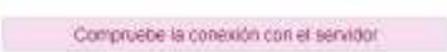
1. Encabezado de la página
2. Título de la página
3. Campos de texto para ingresar usuario y contraseña
4. Botón ingresar
5. Pie de página

El usuario debe ingresar las credenciales necesarias, las cuales serán validadas con el servidor para identificar al usuario administrador.

La Tabla 1 muestra los mensajes de validaciones que pudiera presentarse al usuario.

Tabla 1

Mensajes de validaciones de la página de iniciar sesión

Mensajes de validaciones	Observaciones
	La página de administración no se encuentra conectada con el servidor de base de datos.
	El usuario administrador ingresa credenciales incorrectas.

Elaborado por: Josselyn Rodríguez, Henry Rojas

Página: administración de sitios. Sección: Registro de sitios

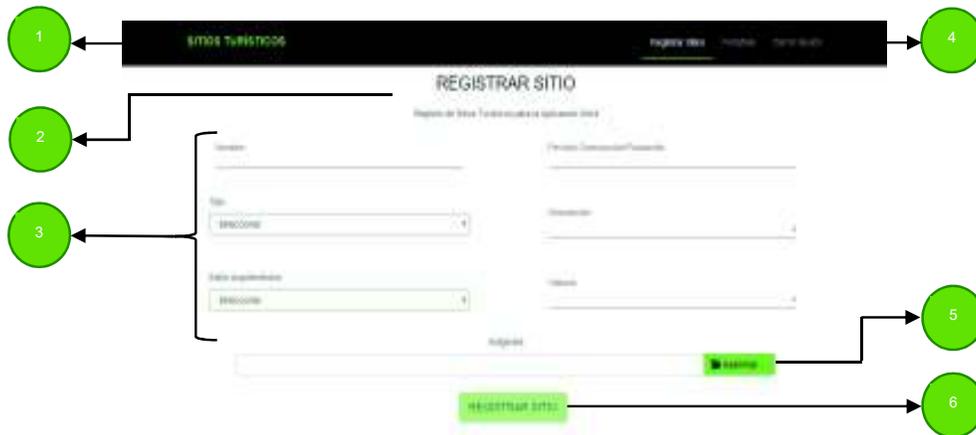


Figura 2 Página: administración de sitios. Sección: registro de sitios

Elaborado por: Josselyn Rodríguez, Henry Rojas

Identificación de las partes de la página administración de sitios, sección registro de sitios:

1. Encabezado de la página
2. Título de la sección
3. Campos de texto para registrar nuevo sitio turístico
4. Menú de la página administración de sitios
5. Botón para seleccionar las imágenes del sitio que se esté registrando
6. Botón para registrar sitio

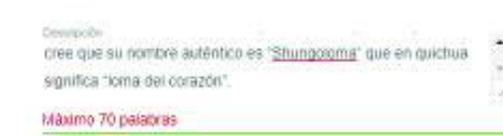
El usuario debe ingresar todos los campos solicitados para un registro exitoso.

La Tabla 2 muestra los mensajes de validaciones que pudiera presentarse al usuario.

Tabla 2

Mensaje de validación de la página de administración de sitios, sección registro de sitios

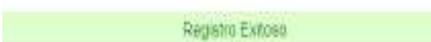
Mensajes de validaciones	Observaciones
	El usuario administrador ingresa un nombre existente para el nuevo sitio turístico.
	El usuario administrador deja vacío los campos requeridos para el registro.



El usuario administrador ingresa un texto de más de 70 palabras para la descripción o historia del sitio turístico que se esté registrando.



El usuario administrador selecciona menos de 5 imágenes para el sitio turístico que se esté registrando.



El usuario administrador ha cumplido con todos los requerimientos para el registro exitoso de sitio turístico.

Elaborado por: Josselyn Rodríguez, Henry Rojas

Página: administración de sitios. Sección: Portafolio



Figura 3 Página: administración de sitios. Sección: portafolio

Elaborado por: Josselyn Rodríguez, Henry Rojas

Identificación de las partes de la página administración de sitios, sección registro de sitios:

1. Encabezado de la página
2. Título de la sección
3. Menú de la página administración de sitios
4. Menú para filtrar los sitios por su tipo
5. Lista de sitios registrados

En esta sección se podrá seleccionar un sitio turístico ya sea para modificar su información, para eliminarlo o para descargar la metadata. Para acceder a estas funciones se debe pasar el mouse por encima del sitio.

Modal: Modificar sitio turístico

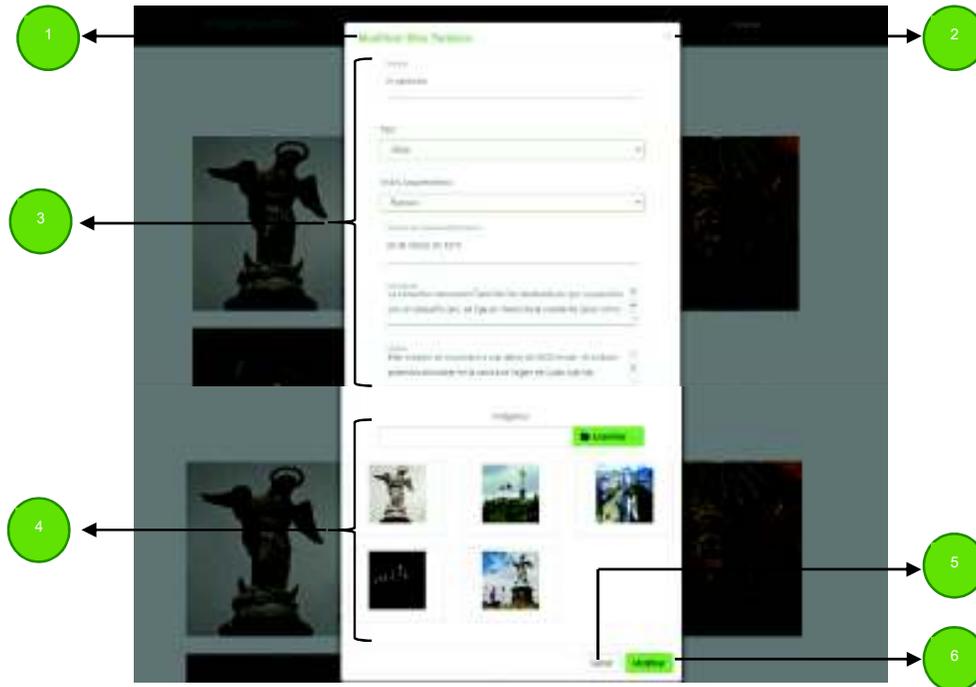


Figura 4 Modal: modificar sitio turístico

Elaborado por: Josselyn Rodríguez, Henry Rojas

Identificación de las partes de modal para modificar sitio turístico:

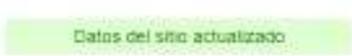
1. Título del modal
2. Botón cerrar modal
3. Información del sitio seleccionado
4. Imágenes del sitio seleccionado
5. Botón cerrar modal
6. Botón modificar

Los mensajes de validaciones para la información del sitio seleccionado son las mismas que se muestran cuando se registra un nuevo sitio. Véase Tabla 2.

En la Tabla 3 se muestran los mensajes de validaciones del modal.

Tabla 3

Mensajes de validaciones del modal modificar sitio turístico

Mensajes de validaciones	Observaciones
	Se muestra un botón para eliminar si existe más de 5 imágenes registradas en el sitio turístico seleccionado.
	Se muestra un mensaje de existo si el usuario modifica los datos del sitio turístico seleccionado.

Elaborado por: Josselyn Rodríguez, Henry Rojas

Modal: Verificar para eliminar un sitio turístico



Figura 4 Modal: verificación para eliminar un sitio turístico

Elaborado por: Josselyn Rodríguez, Henry Rojas

Identificación de las partes de modal para modificar sitio turístico:

1. Título del modal
2. Botón cerrar modal
3. Mensaje de verificación
4. Botón cerrar modal
5. Botón eliminar

El botón  descargará la metadata del sitio turístico, la cual es utilizada para subir el *image target* en la base de datos de Vuforia.

ANEXO 3: Prototipo de la aplicación móvil

Pantalla: Seleccionar idioma



Figura 1 Pantalla: seleccionar idioma

Elaborado por: Josselyn Rodríguez, Henry Rojas

Pantalla: Sitio turístico visitado con idioma en español

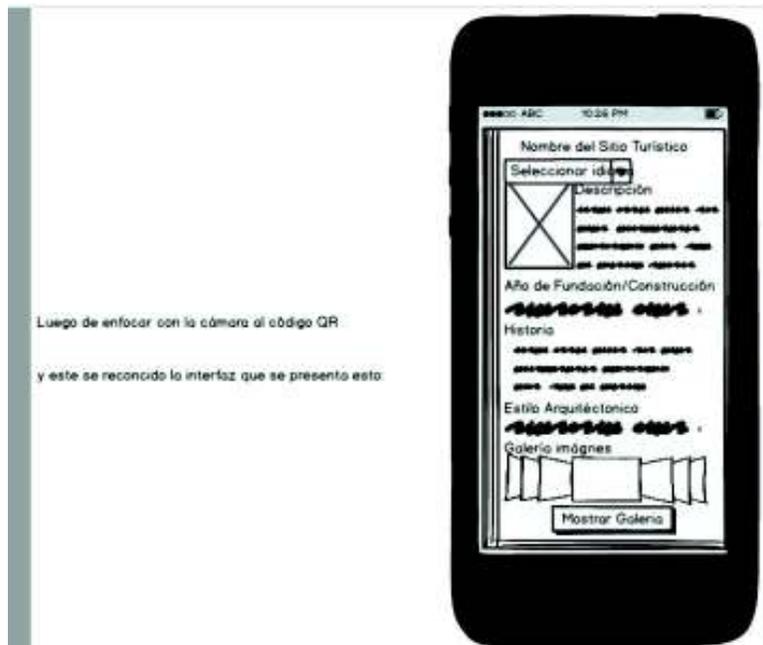


Figura 2 Pantalla: sitio turístico visitado con idioma en español

Elaborado por: Josselyn Rodríguez, Henry Rojas

Pantalla: Sitio turístico visitado con idioma en inglés

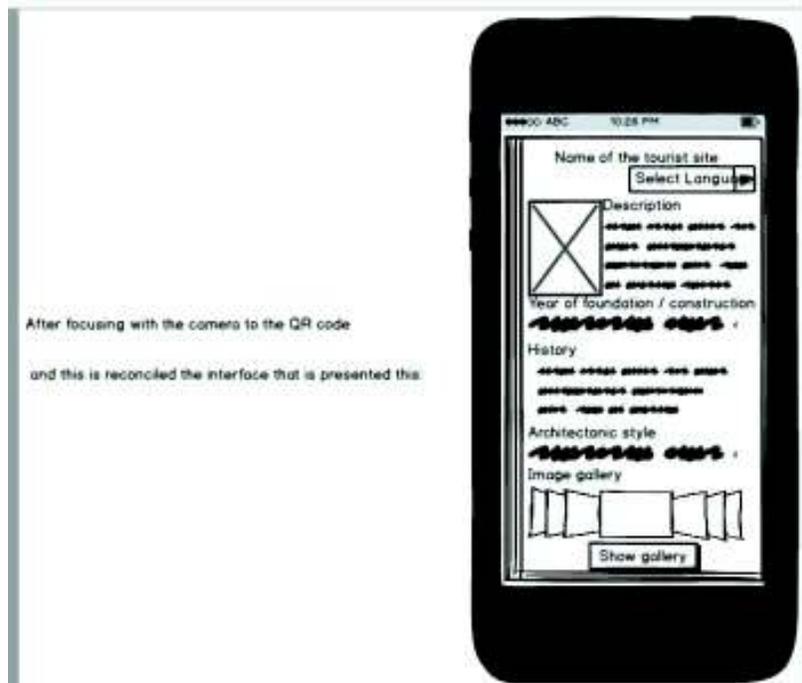


Figura 3 Pantalla: sitio turístico visitado con idioma en inglés

Elaborado por: Josselyn Rodríguez, Henry Rojas

ANEXO 4: Códigos QR de los 50 sitios turísticos

Códigos QR para tipo de sitios: Iglesias



Figura 1 Código QR: Basílica del Voto Nacional

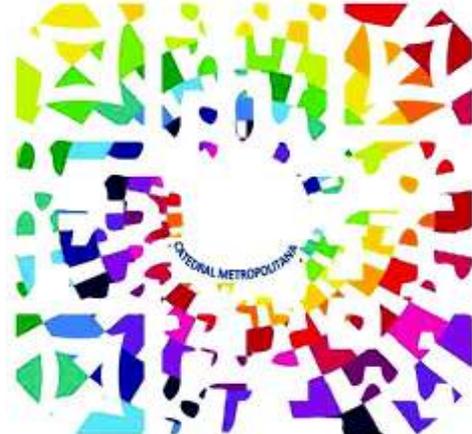


Figura 2 Código QR: Catedral Metropolitana

Elaborado por: Josselyn Rodríguez, Henry Rojas

Elaborado por: Josselyn Rodríguez, Henry Rojas



Figura 3 Código QR: La Compañía de Jesús



Figura 4 Código QR: San Blas

Elaborado por: Josselyn Rodríguez, Henry Rojas

Elaborado por: Josselyn Rodríguez, Henry Rojas



Figura 5 Código QR: El Sagrario



Figura 6: Código QR: Guápulo

Elaborado por: Josselyn Rodríguez, Henry Rojas

Elaborado por: Josselyn Rodríguez, Henry Rojas



Figura 7: Código QR: La Merced



Figura 8: Código QR: San Agustín

Elaborado por: Josselyn Rodríguez, Henry Rojas **Elaborado por:** Josselyn Rodríguez, Henry Rojas



Figura 9: Código QR: San Francisco



Figura 10: Código QR: Santo Domingo

Elaborado por: Josselyn Rodríguez, Henry Rojas **Elaborado por:** Josselyn Rodríguez, Henry Rojas

Códigos QR para tipo de sitios: Museos



Figura 11: Código QR: Aeronáutico y del Esp. FAE



Figura 12: Código QR: Arte Moderno

Elaborado por: Josselyn Rodríguez, Henry Rojas **Elaborado por:** Josselyn Rodríguez, Henry Rojas



Figura 13: Código QR: Ciencias Naturales

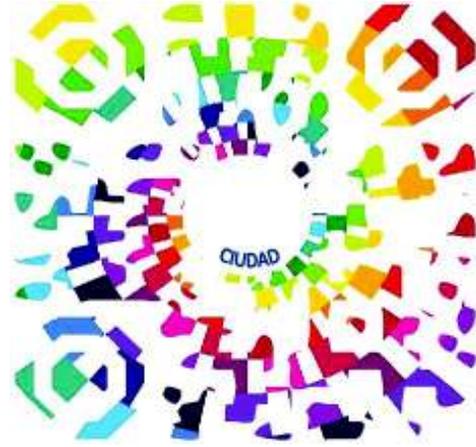


Figura 14: Código QR: Ciudad

Elaborado por: Josselyn Rodríguez, Henry Rojas **Elaborado por:** Josselyn Rodríguez, Henry Rojas

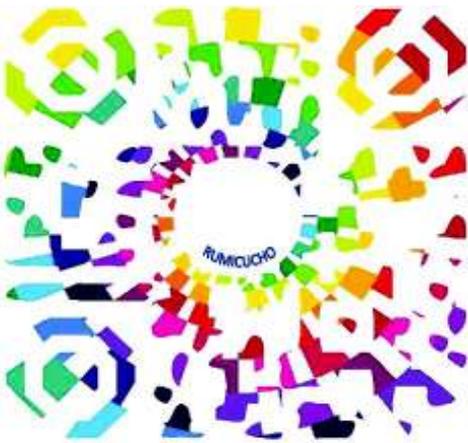


Figura 15: Código QR: Rumicucho

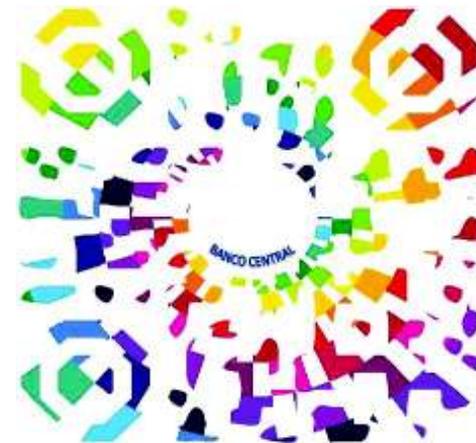


Figura 16: Código QR: Banco Central

Elaborado por: Josselyn Rodríguez, Henry Rojas **Elaborado por:** Josselyn Rodríguez, Henry Rojas

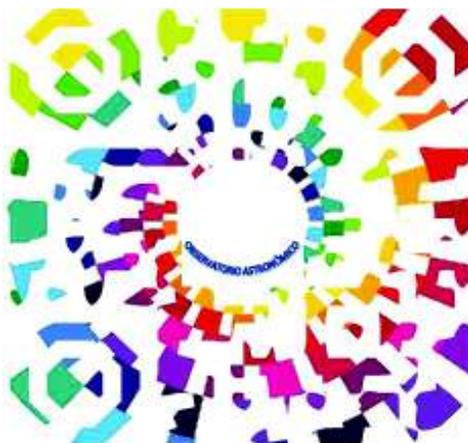


Figura 17: Código QR: Observatorio Astronómico



Figura 18: Código QR: Chimbacalle

Elaborado por: Josselyn Rodríguez, Henry Rojas **Elaborado por:** Josselyn Rodríguez, Henry Rojas



Figura 19: Código QR: Etnográfico Mejía

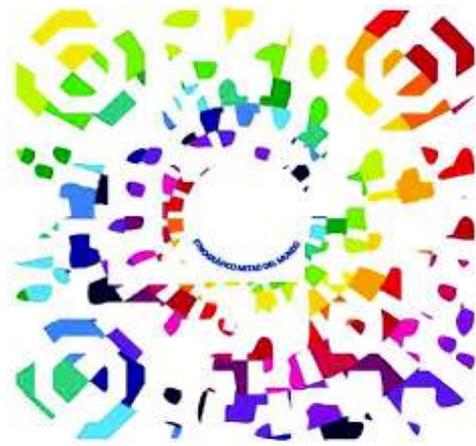


Figura 20: Código QR: Etnográfico Mitad Mundo

Elaborado por: Josselyn Rodríguez, Henry Rojas **Elaborado por:** Josselyn Rodríguez, Henry Rojas



Figura 21: Código QR: Fundación Hallo



Figura 22: Código QR: Fundación Guayasamín

Elaborado por: Josselyn Rodríguez, Henry Rojas **Elaborado por:** Josselyn Rodríguez, Henry Rojas



Figura 23: Código QR: Intiñan



Figura 24: Código QR: Jijón y Caamaño

Elaborado por: Josselyn Rodríguez, Henry Rojas **Elaborado por:** Josselyn Rodríguez, Henry Rojas



Figura 25: Código QR: Manuela Saenz



Figura 26: Código QR: Mindalae

Elaborado por: Josselyn Rodríguez, Henry Rojas **Elaborado por:** Josselyn Rodríguez, Henry Rojas



Figura 27: Código QR: Historia de la Medicina



Figura 28: Código QR: San Diego

Elaborado por: Josselyn Rodríguez, Henry Rojas **Elaborado por:** Josselyn Rodríguez, Henry Rojas



Figura 29: Código QR: Templete Héroes

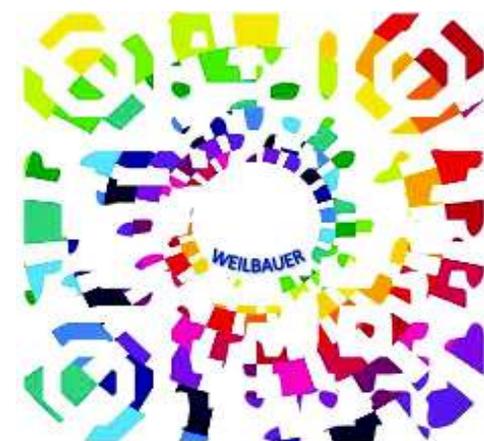


Figura 30: Código QR: Weilbauer

Elaborado por: Josselyn Rodríguez, Henry Rojas **Elaborado por:** Josselyn Rodríguez, Henry Rojas



Figura 31: Código QR: Yaku

Elaborado por: Josselyn Rodríguez, Henry Rojas

Códigos QR para tipo de sitio: Parques



Figura 32: Código QR: Cumandá

Elaborado por: Josselyn Rodríguez, Henry Rojas

Códigos QR para tipo de sitio: Otros



Figura 33: Código QR: Trude Sjka

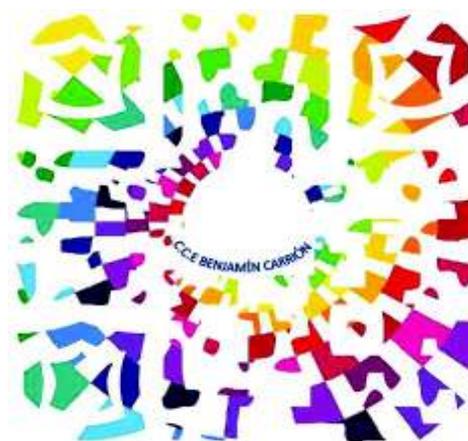


Figura 34: Código QR: C.C.E Benjamín Carrión

Elaborado por: Josselyn Rodríguez, Henry Rojas **Elaborado por:** Josselyn Rodríguez, Henry Rojas



Figura 35: Código QR: C. Música



Figura 36: Código QR: Casa Sucre

Elaborado por: Josselyn Rodríguez, Henry Rojas **Elaborado por:** Josselyn Rodríguez, Henry Rojas



Figura 37: Código QR: Itchimbia



Figura 38: Código QR: San Diego

Elaborado por: Josselyn Rodríguez, Henry Rojas **Elaborado por:** Josselyn Rodríguez, Henry Rojas



Figura 39: Código QR: El Panecillo



Figura 40: Código QR: Botánico de Quito

Elaborado por: Josselyn Rodríguez, Henry Rojas **Elaborado por:** Josselyn Rodríguez, Henry Rojas



Figura 41: Código QR: La capilla del hombre



Figura 42: Código QR: Mitad del Mundo

Elaborado por: Josselyn Rodríguez, Henry Rojas **Elaborado por:** Josselyn Rodríguez, Henry Rojas



Figura 43: Código QR: Arzobispal



Figura 44: Código QR: Carondelet

Elaborado por: Josselyn Rodríguez, Henry Rojas **Elaborado por:** Josselyn Rodríguez, Henry Rojas



Figura 45: Código QR: Municipal



Figura 46: Código QR: Foch

Elaborado por: Josselyn Rodríguez, Henry Rojas **Elaborado por:** Josselyn Rodríguez, Henry Rojas

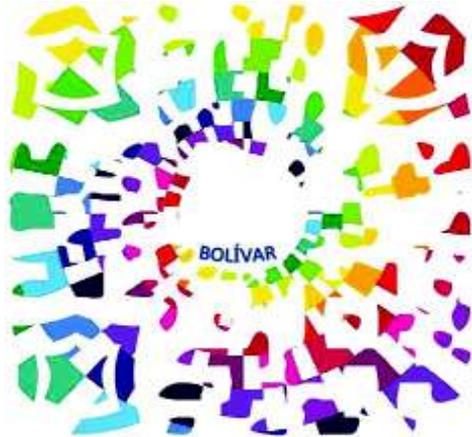


Figura 47: Código QR: Bolívar

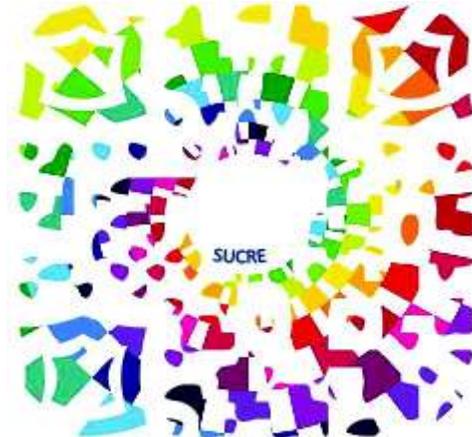


Figura 48: Código QR: Sucre

Elaborado por: Josselyn Rodríguez, Henry Rojas **Elaborado por:** Josselyn Rodríguez, Henry Rojas

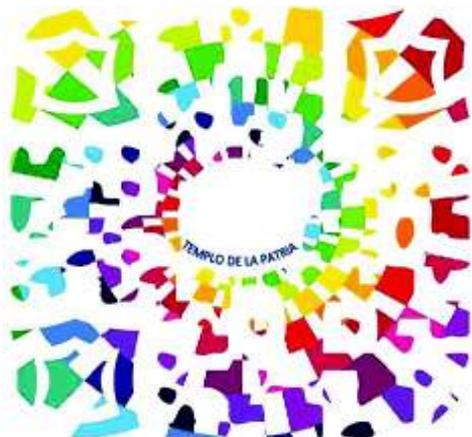


Figura 49: Código QR: Templo de la Patria



Figura 50: Código QR: Vivarium Quito

Elaborado por: Josselyn Rodríguez, Henry Rojas **Elaborado por:** Josselyn Rodríguez, Henry Rojas

ANEXO 5: Ejemplo de utilización de la aplicación móvil

Una vez instalada la aplicación en el móvil se debe proceder a iniciarla y esperar que se cargue la misma. Véase Figura 1 y 2.



Figura 1 Aplicación en el dispositivo Android



Figura 2 Aplicación en el dispositivo iOS

Elaborado por: Josselyn Rodríguez, Henry Rojas **Elaborado por:** Josselyn Rodríguez, Henry Rojas

Una vez cargada la aplicación aparecerá la pantalla de inicio en la cual se debe seleccionar el idioma de preferencia, ya sea inglés o español. Véase Figura 3 y 5. En el caso que el usuario no seleccione ningún idioma le aparece un aviso. Véase Figura 4 y 6.

La pantalla de inicio aparece solo la primera vez que se ejecuta la aplicación, por tal motivo el idioma seleccionado se quedara guardado para la próxima vez que el usuario utilice la aplicación.



Figura 3 Pantalla de inicio en dispositivo Android



Figura 4 Mensaje: seleccionar idioma dispositivo Android

Elaborado por: Josselyn Rodríguez, Henry Rojas

Elaborado por: Josselyn Rodríguez, Henry Rojas



Figura 5 Pantalla de inicio en dispositivo iOS

Elaborado por: Josselyn Rodríguez, Henry Rojas



Figura 6 Mensaje: seleccionar idioma dispositivo iOS

Elaborado por: Josselyn Rodríguez, Henry Rojas

Para este ejemplo se ha seleccionado el idioma español, una vez presionado el botón ingresar aparecerá la cámara del dispositivo listo para enfocar el código correspondiente al sitio turístico visitado. Véase Figura 7, 8, 9 y 10.



Figura 7 Cámara del dispositivo en Android



Figura 8 Enfocar código QR en dispositivo Android

Elaborado por: Josselyn Rodríguez, Henry Rojas **Elaborado por:** Josselyn Rodríguez, Henry Rojas



Figura 9 Cámara del dispositivo en iOS

Elaborado por: Josselyn Rodríguez,
Henry Rojas



Figura 10 Enfocar código QR en dispositivo iOS

Elaborado por: Josselyn Rodríguez,
Henry Rojas

La aplicación móvil necesita tener conexión a internet, si el usuario no se encuentra conectado le aparecerá un aviso y se cerrará la aplicación. Véase Figura 11 y 12.

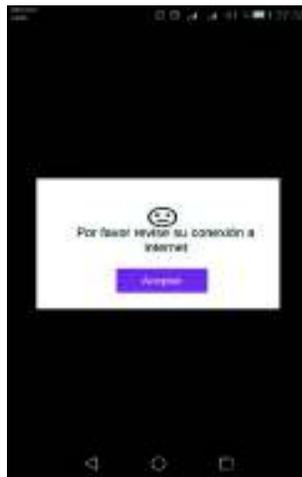


Figura 11 Mensaje: revisar conexión a internet en dispositivo Android

Elaborado por: Josselyn Rodríguez,
Henry Rojas



Figura 12 Mensaje: revisar conexión a internet en dispositivo iOS

Elaborado por: Josselyn Rodríguez,
Henry Rojas

Los datos que son presentados en la aplicación móvil son obtenidos desde un servidor, por tal motivo si el mismo no se encuentra activo se le mostrará un mensaje al usuario. Véase Figura 13 y 14.



Figura 13 Mensaje: problema con el servidor mostrado en Android

Elaborado por: Josselyn Rodríguez,
Henry Rojas



Figura 14 Mensaje: problema con el servidor mostrado en iOS

Elaborado por: Josselyn Rodríguez,
Henry Rojas

Finalmente, si el dispositivo cumple con todo lo requerido y el servidor se encuentra activo el usuario podrá visualizar la información y la galería de imágenes del sitio turístico visitado. Véase Figura 15 y 16.



Figura 15 Información y galería de imágenes del sitio turístico visitado mostrado en Android

Elaborado por: Josselyn Rodríguez, Henry
Rojas



Figura 16 Información y galería de imágenes del sitio turístico visitado mostrado en iOS

Elaborado por: Josselyn Rodríguez, Henry
Rojas

El usuario podrá interactuar con la aplicación cambiando de idioma y/o mostrando la galería de imágenes en la pantalla completa del dispositivo. Véase Figura 17, 18, 19 y 20.

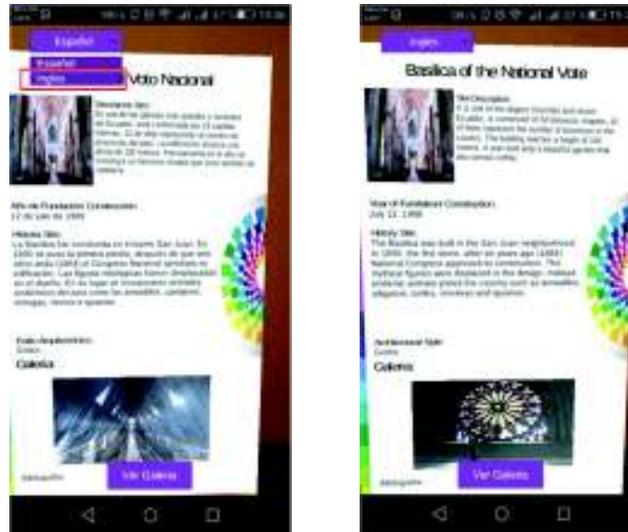


Figura 17 Cambio de idioma de español a inglés en dispositivo Android

Elaborado por: Josselyn Rodríguez, Henry Rojas

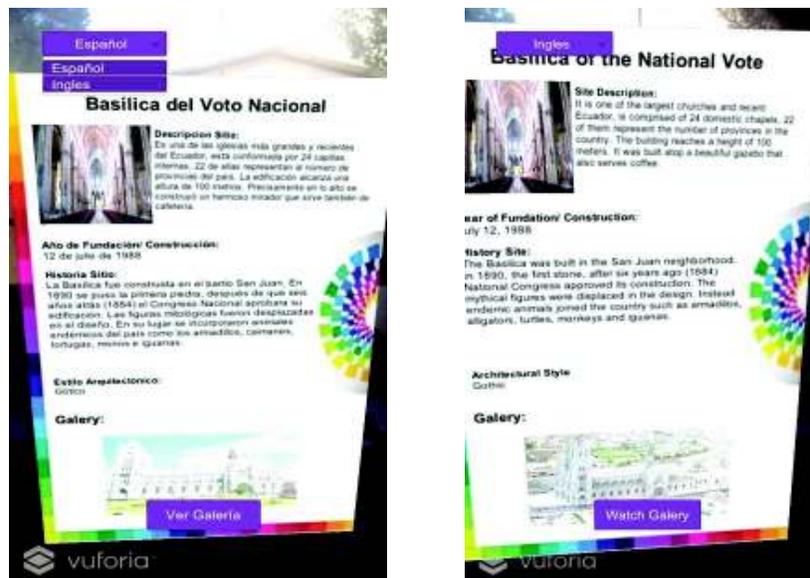
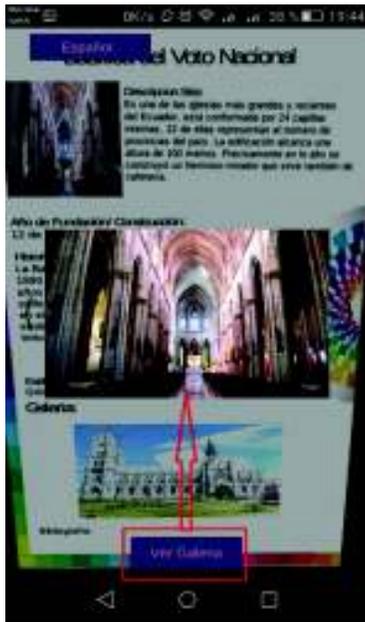


Figura 18 Cambio de idioma de español a inglés en dispositivo iOS

Elaborado por: Josselyn Rodríguez, Henry Rojas

Dispositivo móvil en forma vertical



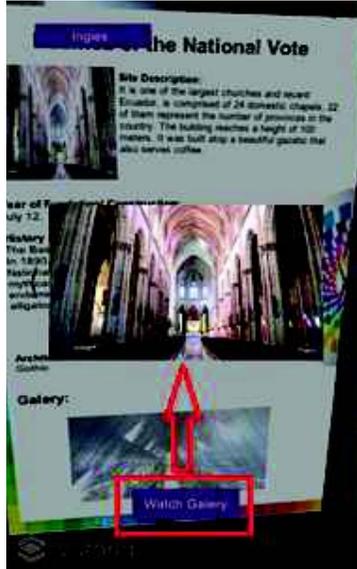
Dispositivo móvil en forma horizontal



Figura 19 Galería de imágenes en la pantalla completa en un dispositivo Android

Elaborado por: Josselyn Rodríguez, Henry Rojas

Dispositivo móvil en forma vertical



Dispositivo móvil en forma horizontal



Figura 20 Galería de imágenes en la pantalla completa en un dispositivo iOS

Elaborado por: Josselyn Rodríguez, Henry Rojas