

UNIVERSIDAD TÉCNICA DEL NORTE



Facultad de Ingeniería en Ciencias Aplicadas
Carrera de Ingeniería en Sistemas Computacionales

TEMA:

**ESTUDIO DE LA HERRAMIENTA DE DESARROLLO MÓVIL XAMARIN,
MEDIANTE LA CREACIÓN DE UN APLICATIVO MULTIPLATAFORMA PARA
ORIENTACIÓN E INFORMACIÓN TURÍSTICA DEL CANTÓN COTACACHI.**

Trabajo de grado previo a la obtención del título de Ingeniero en Sistemas
Computacionales.

AUTOR:

Jefferson Esteban Andrade Pijuango

DIRECTOR:

Msc. Diego Javier Trejo España

Ibarra, 2019



UNIVERSIDAD TÉCNICA DEL NORTE
BIBLIOTECA UNIVERSITARIA
AUTORIZACIÓN DE USO Y PUBLICACIÓN A FAVOR DE LA
UNIVERSIDAD TÉCNICA DEL NORTE

1. IDENTIFICACIÓN DE LA OBRA

En cumplimiento del Art. 144 de la Ley de Educación Superior, hago la entrega del presente trabajo a la Universidad Técnica del Norte para que sea publicado en el Repositorio Digital Institucional, para lo cual pongo a disposición la siguiente información:

| DATOS DEL CONTACTO | |
|-----------------------------|--|
| CÉDULA DE IDENTIDAD: | 1003653415 |
| APELLIDOS Y NOMBRES: | Andrade Pijuango Jefferson Esteban |
| DIRECCIÓN: | Cotacachi-Pilchibuela |
| EMAIL: | jefflionel.94@outlook.com jeandradep@utn.edu.ec |
| CELULAR: | 0985630179 |

| DATOS DE LA OBRA | |
|--------------------------------|--|
| TÍTULO: | “ESTUDIO DE LA HERRAMIENTA DE DESARROLLO MÓVIL XAMARIN, MEDIANTE LA CREACIÓN DE UN APLICATIVO MULTIPLATAFORMA PARA ORIENTACIÓN E INFORMACIÓN TURÍSTICA DEL CANTÓN COTACACHI” |
| AUTOR: | Andrade Pijuango Jefferson Esteban |
| FECHA: | 25-11-2019 |
| TÍTULO POR EL QUE OPTA: | Ingeniería en sistemas computacionales |
| DIRECTOR: | Msc. Diego Trejo España |

2. CONSTANCIAS

El autor manifiesta que la obra objeto de la presente autorización es original y la desarrolló sin violar derechos de autor de terceros, por lo tanto, la obra es original y que es el titular de los derechos patrimoniales, por lo que asume la responsabilidad sobre el contenido de la misma y saldrá en defensa de la Universidad en caso de reclamación por parte de terceros.

EL AUTOR:

Firma 
Nombre: Andrade Pijuango Jefferson Esteban
c.c.: 1003653415
Ibarra, 25 de noviembre del 2019

CERTIFICACIÓN DIRECTOR

Por medio del presente certifico que el Sr. Jefferson Esteban Andrade Pijuango, portador de la cédula de identidad Nro. 100365341-5 ha trabajado en el desarrollo del proyecto de tesis: **“ESTUDIO DE LA HERRAMIENTA DE DESARROLLO MÓVIL XAMARIN, MEDIANTE LA CREACIÓN DE UN APLICATIVO MULTIPLATAFORMA PARA ORIENTACIÓN E INFORMACIÓN TURÍSTICA DEL CANTÓN COTACACHI”**, previo a la obtención del título de Ingeniero en Sistemas Computacionales, lo cual ha realizado en su totalidad con responsabilidad.

Es todo cuanto puede certificar en honor a la verdad.



MSc. Diego Javier Trejo España

DIRECTOR DE TESIS



UNIVERSIDAD TÉCNICA DEL NORTE
FACULTAD DE INGENIERÍA EN CIENCIAS APLICADAS
CARRERA DE INGENIERÍA EN SISTEMAS COMPUTACIONALES

ACTA DE ENTREGA RECEPCIÓN

Tema: Aplicación móvil multiplataforma para orientación e información turística del cantón Cotacachi

En las instalaciones del GAD Santa Ana de Cotacachi, el día lunes 25 de noviembre de 2019, la Carrera de Ingeniería en Sistemas Computacionales de la Universidad Técnica del Norte, por medio del trabajo de titulación **"ESTUDIO DE LA HERRAMIENTA DE DESARROLLO MÓVIL XAMARIN, MEDIANTE LA CREACIÓN DE UN APLICATIVO MULTIPLATAFORMA PARA ORIENTACIÓN E INFORMACIÓN TURÍSTICA DEL CANTÓN COTACACHI"** hace la entrega de la aplicación a la Dirección de Desarrollo Económico y Producción en la que se encuentra la Jefatura de Turismo Sustentable del Municipio de Cotacachi.

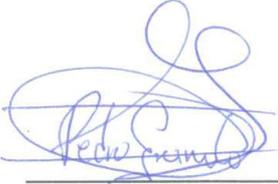
El desarrollo del sistema informático lo realizó el Sr. **Jefferson Esteban Andrade Pijuango** con **C.I 1003653415** bajo la tutoría del Msc. Diego Trejo, y apoyo del Ing. Heriberto Sanipatin director de Tecnología de la municipalidad.

A continuación, se detalla los productos entregados:

- Sistema Web (Código Fuente) para gestión de información turística como:
 - Atractivos naturales, patrimonio y de Intag con georreferenciados.
 - Manifestaciones culturales.
 - Ofertas turísticas.
 - Gastronomía.
 - Establecimientos (comercio, hospedaje, transporte y financieros)
 - Eventos con georreferenciación.
- Aplicación móvil (Código Fuente) enlazada al sistema web para información y orientación turística.
- Manual Técnico.
- Capacitación al personal encargado.

Atentamente, ENTREGA CONFORME


Sr. Jefferson Andrade
TESISTA
UNIVERSIDAD TÉCNICA DEL NORTE


Ing. Pedro Granda
Coord. Carrera Ingeniería en
Sistemas Computacionales


Msc. Diego Trejo
DOCENTE TUTOR
UNIVERSIDAD TÉCNICA DEL NORTE


Ing. Flor María Vaca
Directora de Comunicación



Dedicatoria

Este trabajo está dedicado a Dios, a mi padre: Luis Andrade, a mi madre: Rosa Pijuango, a mis hermanos Samir, Antonella y a mi familia Jessica, Camila y Elián quienes me apoyaron incondicionalmente para alcanzar el objetivo de culminar mi carrera universitaria.

Jefferson Andrade

Agradecimientos

A Dios, a mis padres por ser el pilar fundamental en mi vida, por enseñarme buenos valores que me sirvieron como las principales herramientas para cumplir mis metas.

Al Gobierno Autónomo Descentralizado de Santa Ana de Cotacachi por haberme brindado el apoyo y haber dado la apertura a la realización de este proyecto.

Ing. Diego Trejo, Director de Trabajo de Grado por ser la guía principal para el desarrollo del proyecto. A mis docentes, que a lo largo de estos años de estudios me supieron impartir sus conocimientos.

Tabla de Contenido

| | |
|---|-------|
| AUTORIZACIÓN DE USO Y PUBLICACIÓN A FAVOR DE LA UNIVERSIDAD TÉCNICA DEL NORTE | II |
| | IV |
| Dedicatoria | VI |
| Agradecimientos..... | VII |
| Índice de figuras | XI |
| Índice de tablas | XIV |
| Resumen | XV |
| Abstract | XVI |
| INTRODUCCIÓN..... | XVII |
| Antecedentes | XVII |
| Situación Actual | XVII |
| Prospectiva | XVII |
| Planteamiento del problema..... | XVIII |
| Objetivos..... | XIX |
| Objetivo General..... | XIX |
| Objetivos Específicos | XIX |
| Alcance..... | XIX |
| Justificación | XX |
| Justificación Tecnológica | XX |
| Impacto Económico..... | XX |
| Contexto | XXI |
| 1. MARCO TEÓRICO | 1 |
| 1.1. Turismo..... | 1 |
| 1.1.1. Gestión Turística..... | 1 |
| 1.1.2. Paquetes Turísticos | 2 |
| 1.1.3. Turismo y tecnologías de la información | 2 |
| 1.2. Dispositivos Móviles..... | 3 |
| 1.2.1. Sistemas operativos de dispositivos móviles | 3 |
| 1.2.2. Sistema Operativo Android | 4 |

| | | |
|--------|--|----|
| 1.2.3. | Sistema Operativo iOS | 5 |
| 1.2.4. | Sistema Operativo Windows Phone | 6 |
| 1.3. | Aplicaciones Móviles..... | 6 |
| 1.3.1. | Tipos de aplicaciones | 7 |
| 1.3.2. | Aplicaciones móviles turísticas | 9 |
| 1.4. | Desarrollo de aplicaciones móviles | 11 |
| 1.4.1. | Desarrollo multiplataforma | 11 |
| 1.5. | Herramientas de desarrollo móvil multiplataforma | 12 |
| 1.5.1. | Xamarin | 12 |
| 1.5.2. | Delphi (Embarcadero RAD Estudio)..... | 20 |
| 1.6. | Metodología ágil SCRUM..... | 24 |
| 1.6.1. | Reseña Histórica | 24 |
| 1.6.2. | ¿Qué es Scrum?..... | 24 |
| 1.6.3. | Roles de Scrum | 25 |
| 1.6.4. | Eventos de Scrum | 26 |
| 1.6.5. | Artefactos de Scrum | 27 |
| 2. | DESARROLLO METODOLÓGICO | 29 |
| 2.1. | Fase 1: Iniciación..... | 29 |
| 2.1.1. | Visión de proyecto | 29 |
| 2.1.2. | Comparativa de las herramientas | 29 |
| 2.1.3. | Definición de roles de Scrum | 35 |
| 2.1.4. | Pila de Producto | 36 |
| 2.2. | Fase 2: Planificación..... | 38 |
| 2.2.1. | Creación de Historias de Usuario..... | 38 |
| 2.2.2. | Aprobar, estimar y asignar las Historias de Usuario..... | 49 |
| 2.2.3. | Elaboración y estimación de tareas | 50 |
| 2.3. | Fase 3: Desarrollo | 55 |
| 2.3.1. | Arquitectura de la aplicación | 55 |
| 2.3.2. | Descripción de los Sprints | 56 |

| | |
|---|----|
| 3. RESULTADOS | 69 |
| 3.1. Resultados de la comparación del consumo de recurso al ejecutar los prototipos diseñados. 69 | |
| 3.1.1. Memoria..... | 69 |
| 3.1.2. Rendimiento | 71 |
| 3.1.3. Almacenamiento | 71 |
| 3.1.4. Evaluación final..... | 72 |
| CONCLUSIONES..... | 74 |
| RECOMENDACIONES..... | 75 |
| BIBLIOGRAFÍA..... | 76 |
| ANEXOS | 81 |
| Anexo 1: Pruebas de iteraciones con prototipos..... | 81 |
| Anexo 2: Casos de uso diseñados durante el desarrollo del sistema..... | 88 |
| Anexo 3: Representación del desarrollo con SCRUM..... | 90 |

Índice de figuras

| | |
|--|-----|
| Fig. 0.1. Patrón de diseño MVVM..... | XIX |
| Fig. 1.1. Arquitectura del Sistema Operativo Android | 4 |
| Fig. 1.2. Arquitectura del Sistema Operativo iOS | 6 |
| Fig. 1.3. Variedad de aplicaciones móviles..... | 7 |
| Fig. 1.4. Aplicaciones turísticas más populares | 10 |
| Fig. 1.5. Aplicaciones turísticas en iOS y Android | 11 |
| Fig. 1.6. Desarrollo multiplataforma con Xamarin | 12 |
| Fig. 1.7. Panel de herramientas de Xamarin.Forms..... | 14 |
| Fig. 1.8. Lenguaje de enmarcado XAML | 15 |
| Fig. 1.9. Arquitectura de una aplicación con Xamarin..... | 16 |
| Fig. 1.10. Tipos de proyectos en Xamarin | 17 |
| Fig. 1.11. Proyecto PCL | 17 |
| Fig. 1.12. Estructura de un proyecto PCL..... | 18 |
| Fig. 1.13. Instalación de Xamarin | 20 |
| Fig. 1.14. Detalles de instalación de Xamarin..... | 20 |
| Fig. 1.15. Instalación de Delphi y C++ con RAD Studio..... | 24 |
| Fig. 1.16. Los 3 componentes de SCRUM | 25 |
| Fig. 2.1. Primera iteración | 30 |
| Fig. 2.2. Primera iteración – monitoreo de las aplicaciones. | 30 |
| Fig. 2.3. Segunda iteración..... | 31 |
| Fig. 2.4. Segunda iteración – monitoreo de las aplicaciones | 32 |
| Fig. 2.5. Tercera iteración..... | 32 |
| Fig. 2.6. Tercera iteración – monitoreo de las aplicaciones | 33 |
| Fig. 2.7. Cuarta iteración | 33 |
| Fig. 2.8. Cuarta iteración – monitoreo de las aplicaciones..... | 34 |
| Fig. 2.9. Patrón MVVM | 55 |
| Fig. 2.10. Diagrama de base de datos..... | 57 |
| Fig. 2.11. Login del sistema web | 57 |
| Fig. 2.12. Vista para crear un nuevo usuario | 58 |
| Fig. 2.13. Validación de usuario | 58 |
| Fig. 2.14. Registro de contraseña del nuevo usuario..... | 58 |
| Fig. 2.15. Lista de usuarios | 59 |
| Fig. 2.16. Menú sistema web Visita Cotacachi | 59 |
| Fig. 2.17. Tipos de atractivos | 59 |
| Fig. 2.18. Lista de atractivos..... | 59 |

| | |
|--|----|
| Fig. 2.19. Formulario de registro de atractivos..... | 60 |
| Fig. 2.20. Lista de eventos | 60 |
| Fig. 2.21. Registro de eventos..... | 61 |
| Fig. 2.22. Lista de manifestaciones culturales | 61 |
| Fig. 2.23. Registro de manifestaciones culturales | 61 |
| Fig. 2.24. Lista de restaurantes | 62 |
| Fig. 2.25. Registro de restaurantes | 62 |
| Fig. 2.26. Lista de ofertas turísticas..... | 63 |
| Fig. 2.27. Lista de establecimientos | 63 |
| Fig. 2.28. Registro de establecimientos..... | 63 |
| Fig. 2.29. Lista de clientes..... | 64 |
| Fig. 2.30. Registro de clientes aplicación móvil | 64 |
| Fig. 2.31. Menú hamburgués..... | 65 |
| Fig. 2.32. Lista de eventos aplicación móvil | 65 |
| Fig. 2.33. Diseño del menú principal aplicación móvil..... | 65 |
| Fig. 2.34. Tipos de atractivos aplicación móvil..... | 66 |
| Fig. 2.35. Detalle de un atractivo aplicación móvil | 66 |
| Fig. 2.36. Lista de gastronomía aplicación móvil | 66 |
| Fig. 2.37. Lista de manifestaciones aplicación móvil | 67 |
| Fig. 2.38. Lista de restaurantes aplicación móvil | 67 |
| Fig. 2.39. Establecimientos aplicación móvil..... | 67 |
| Fig. 2.40. Georreferenciación aplicación móvil | 68 |
| Fig. 2.41. Consulta del clima aplicación móvil | 68 |
| Fig. 3.1. Iterador RAM | 70 |
| Fig. 3.2. Iterador CPU | 71 |
| Fig. 3.3. Iterador almacenamiento | 72 |
| Fig. 3.4. Evaluación de triangulación..... | 73 |
| Fig. 3.5. Evaluación de iteradores global..... | 73 |
| Fig. 4.1. Quinta iteración | 81 |
| Fig. 4.2. Quinta iteración – monitoreo de las aplicaciones. | 82 |
| Fig. 4.3. Sexta iteración..... | 82 |
| Fig. 4.4. Sexta iteración – monitoreo de las aplicaciones. | 83 |
| Fig. 4.5. Séptima iteración..... | 83 |
| Fig. 4.6. Séptima iteración – monitoreo de las aplicaciones. | 84 |
| Fig. 4.7. Octava iteración..... | 84 |
| Fig. 4.8. Séptima iteración – monitoreo de las aplicaciones. | 85 |
| Fig. 4.9. Novena iteración..... | 85 |

| | |
|--|----|
| Fig. 4.10. Novena iteración – monitoreo de las aplicaciones. | 86 |
| Fig. 4.11. Decima iteración..... | 87 |
| Fig. 4.12. Décima iteración – monitoreo de las aplicaciones. | 87 |

Índice de tablas

| | |
|---|----|
| TABLA 1.1 COMPARACIÓN DE LOS TIPOS DE APLICACIONES | 9 |
| TABLA 1.2 ELEMENTOS DE DISEÑO DE XAML..... | 18 |
| TABLA 1.3 REQUERIMIENTOS DE INSTALACIÓN DE XAMARIN | 19 |
| TABLA 1.4 HISTORIA DE RAD STUDIO | 21 |
| TABLA 1.5 REQUISITOS DE RAD STUDIO | 23 |
| TABLA 1.6 ROLES DE SCRUM..... | 25 |
| TABLA 2.1 ITERACIONES MEMORIA RAM(Mb)..... | 34 |
| TABLA 2.2 ITERACIONES TRABAJO EN CPU (%) | 35 |
| TABLA 2.3 ITERACIONES DE ALMACENAMIENTO (Mb) | 35 |
| TABLA 2.4 DESCRIPCIÓN DE ROLES DEL EQUIPO SCRUM..... | 36 |
| TABLA 2.5 PARÁMETROS DE ESTIMACIÓN..... | 36 |
| TABLA 2.6 PILA DE PRODUCTO..... | 36 |
| TABLA 2.7 HISTORIA DE USUARIO H-01 – Preparación del ambiente de trabajo | 38 |
| TABLA 2.8 HISTORIA DE USUARIO H-02 – Acceso al sistema web | 39 |
| TABLA 2.9 HISTORIA DE USUARIO H-03 – Atractivos turísticos..... | 40 |
| TABLA 2.10 HISTORIA DE USUARIO H-04 – Atractivos turísticos..... | 41 |
| TABLA 2.11 HISTORIA DE USUARIO H-05 – Información cultural | 42 |
| TABLA 2.12 HISTORIA DE USUARIO H-06 – Gastronomía y restaurantes..... | 43 |
| TABLA 2.13 HISTORIA DE USUARIO H-07 – Ofertas turísticas..... | 44 |
| TABLA 2.14 HISTORIA DE USUARIO H-08 – Establecimientos y comercio..... | 45 |
| TABLA 2.15 HISTORIA DE USUARIO H-09 – Registro de clientes | 46 |
| TABLA 2.16 HISTORIA DE USUARIO H-10 – Aplicación móvil turística..... | 47 |
| TABLA 2.17 HISTORIA DE USUARIO H-11 – Georreferenciación | 48 |
| TABLA 2.18 HISTORIA DE USUARIO H-12 – Clima | 49 |
| TABLA 2.19 APROBACIÓN, ESTIMACIÓN Y ASIGNACIÓN DE HISTORIAS..... | 50 |
| TABLA 2.20 SPRINT 1 | 50 |
| TABLA 2.21 SPRINT 2 | 51 |
| TABLA 2.22 SPRINT 3 | 52 |
| TABLA 2.23 SPRINT 4 | 54 |
| TABLA 3.1 VENATAJAS Y DESVENTAJAS ENTRE LAS PLATAFORMAS XAMARIN Y DELPHI..... | 69 |

Resumen

El presente documento se encuentra conformado por tres capítulos, donde se detalla todo el proceso del Trabajo de Grado: “ESTUDIO DE LA HERRAMIENTA DE DESARROLLO MÓVIL XAMARIN, MEDIANTE LA CREACIÓN DE UN APLICATIVO MULTIPLATAFORMA PARA ORIENTACIÓN E INFORMACIÓN TURÍSTICA DEL CANTÓN COTACACHI”, empleando la metodología de desarrollo ágil SCRUM.

En la parte Introdutoria se describen los antecedentes, situación actual, prospectiva, planteamiento del problema, objetivo general y objetivos específicos. Así como también se incluye el alcance del proyecto, junto con la justificación de la realización del mismo.

En el Capítulo 1, se presenta todo lo que corresponde a la investigación realizada, se describen temas como el desarrollo de aplicaciones móviles, el estudio de las herramientas de desarrollo multiplataforma Xamarin y Delphi, la metodología de desarrollo, así como también los conceptos básicos o definición de turismo, gestión turística y las principales aplicaciones en mercados turísticos.

En el Capítulo 2, se incluye el desarrollo de prototipos comparativos en Xamarin y Delphi, además de la ejecución de todas las fases de la metodología de desarrollo SCRUM, con sus respectivos informes que validan la calidad del software.

En el Capítulo 3, se definen los resultados obtenidos de la comparativa de las herramientas de estudio.

Abstract

This document is made up of three chapters where include the full process of the Degree Work: “STUDY OF THE MOBILE DEVELOPMENT TOOL XAMARIN, THROUGH THE CREATION OF A MULTIPLATFORM APPLICATION FOR ORIENTATION AND TOURISTIC INFORMATION ABOUT THE CANTON COTACACHI” using the Agile SCRUM development methodology.

In the Introduction part are the historic description, current situation, prospective, problem statement, general objective and specific objectives. Also, this topic includes the scope that the project will have along with the justification for the realization for the realization of it.

In Chapter 1, corresponds to the research carried for the work, topics such as the mobile applications development, the study of the multiplatform development tools as Xamarin and Delphi, the development methodology, as well as the basic concepts or definitions of tourism, tourism management and the main applications in tourist markets.

The Chapter 2 includes the development of comparative prototypes in Xamarin and Delphi, in addition to the execution of all phases of the SCRUM development methodology, with their respective reports that validate the quality of the software.

In the Chapter 3, are set the results obtained from the comparison of study about Xamarin and Delphi tools.

INTRODUCCIÓN

Antecedentes

Las aplicaciones móviles desde su aparición en los 90s han tomado una evolución rápida gracias a las recientes innovaciones tecnológicas, haciendo que los dispositivos móviles sean los más usados a nivel mundial, así como también el crecimiento de la industria, todo debido a la creación de nuevas aplicaciones que cumplen con funciones como: apoyo, portabilidad, diversión, accesibilidad y reducción de costos.

Gracias a estas nuevas tecnologías se ha podido establecer un contacto más cercano entre las empresas y sus clientes, esto ha llevado a las organizaciones a optar por adquirir sus propias aplicaciones con el fin de ofrecer mayor comodidad a sus clientes. El mercado además se adiere a la tecnología como una oportunidad de crecimiento, por esta razón han creado departamentos de software que permiten ofrecer mejor servicio al cliente y variedad de aplicaciones en las diferentes marcas de telefonía. (San Marin & Carpio, 2012)

Situación Actual

En la actualidad gran parte de las empresas para su crecimiento recurren al desarrollo de aplicaciones para teléfonos que promuevan su mercado, un número elevado de estas aplicaciones son diseñadas para sistemas operativos como: iOS, Android y Windows, lo que la implementación de una aplicación multiplataforma para un profesional de Informática implica crear distintas aplicaciones, una para cada sistema con el fin proveer un mismo servicio.

Entonces, desde el mundo comercial surge la necesidad de crear aplicaciones con distintas características a corto plazo y de bajo costo, dando así paso a la aparición de nuevas herramientas de desarrollo que facilitan la construcción de aplicaciones que se puedan implementar en los diferentes sistemas operativos. Pero, la empresa VisionMobile en su informe Developer Economics, afirma que la mayoría de desarrolladores en la actualidad trabajan solamente para una plataforma específica, de un total de 7.000 en 127 países distintos, el desarrollo de aplicaciones móviles lidera Android y iOS con un 71% y 55% respectivamente. (Pastor, 2014)

Prospectiva

El mercado de las aplicaciones móviles avanza a gran velocidad provocando que se convierta en un importante generador de empleo. Con el pasar de los días los dispositivos

móviles siguen cambiando la vida de los usuarios por medio de sistemas escalables a sus necesidades. (Carrasco & Devece, 2015) Por lo tanto, con el presente proyecto se busca proponer el uso de herramientas de desarrollo multiplataforma que garanticen dichas necesidades como productividad, accesibilidad y reducción de tiempo de desarrollo.

Planteamiento del problema

La evolución de las aplicaciones móviles desde su aparición es constante gracias a la variedad, apoyo y accesibilidad, de modo que, despertaron el interés de empresas quienes buscan integrarse a nuevas tecnologías y alzar su productividad.

Por su lado, a nivel de programación, existen varias formas de desarrollar una aplicación, cada una de ellas tiene diferentes características y limitaciones, especialmente desde el punto de vista técnico. Una de estas limitaciones es que los desarrolladores producen aplicaciones monoplataforma y de programación nativa, es decir que operan en un solo sistema operativo móvil, provocando que estas sean de baja escalabilidad y con altos costos en tiempos de desarrollo excesivo.

Por estudios realizados se dice que, de un total de 7.000 desarrolladores un 26% de estos trabaja de forma paralela en distintas plataformas móviles, pero gracias a empresas como Microsoft a futuro esta estadística cambiará, ya que adquieren y mejoran herramientas de desarrollo para hacer posible la programación multiplataforma, reduciendo costos, mejorando tiempos de trabajo y logrando una experiencia igual al desarrollo específico de cada plataforma.

Entonces, en la actualidad el desarrollo móvil multiplataforma es una nueva modalidad de programación, por la cantidad de beneficios que ofrece y la importancia que representa poder sumergirse en las tiendas de aplicaciones de las empresas más grandes como lo son Android, iOS y Windows. Para ello Microsoft trabaja con Xamarin que forman una plataforma extraordinariamente veloz, con capacidades de acceder a todas las características de los dispositivos, de escribir controles y compilar como si fueran 100% nativas, centrando así posible la reutilización de código.

Al no contar con el conocimiento necesario de herramientas multiplataforma se limitan las capacidades de los programadores.

Objetivos

Objetivo General

- Estudiar la herramienta de desarrollo móvil Xamarin mediante la construcción de una aplicación multiplataforma para orientación e información turística del cantón Cotacachi.

Objetivos Específicos

- Identificar las características de los sistemas operativos móviles.
- Analizar la estructura para la determinación de propiedades de la herramienta.
- Comparar las herramientas de desarrollo móvil multiplataforma Xamarin y Delphi.
- Planificar el desarrollo de la aplicación mediante la metodología SCRUM.
- Desarrollar una aplicación multiplataforma de información turística para el cantón Cotacachi.

Alcance

El siguiente proyecto está direccionado a la investigación y análisis de las características, alcances y limitaciones que nos brinda la herramienta de desarrollo móvil Xamarin en su entorno Open Source.

El desarrollo de la aplicación turística para el cantón Cotacachi será un aplicativo que constará con un entorno bilingüe español e inglés, adicional a esto constará con dos módulos principales:

- Consulta de Información relevante de los lugares turísticos del cantón.
- Consulta de guías y rutas por medio de google-maps.

La arquitectura que se utilizará en la aplicación será de tipo MVVM “Modelo, Vista y Vista-Modelo”, como se muestra en la siguiente figura.



Fig. 0.1. Patrón de diseño MVVM

Fuente: Propia

Se hará uso de la metodología SCRUM por su rápido desempeño al llevar a cabo proyectos complejos y escalarlos en módulos o cajas que completan el producto final.

La demostración de la aplicación se la realizará en un dispositivo Android, y para las plataformas iOS y Windows Phone se realizará con la utilización de emuladores.

Justificación

Con el fin de justificar el estudio de la herramienta Xamarin se hace referencia al “Análisis Experimental de Desarrollo de Aplicaciones Móviles Multiplataforma” desarrollado por la Universidad Nacional de Plata (Argentina) lo cual toma como casos de estudio a Xamarin, Delphi y Titanium 3, por ser entornos multiplataforma. (Delía, Galdamez, Thomas, Corbalan, & Pesado, 2015)

Dicho análisis optó como una validación el desarrollo de una misma aplicación móvil con las tres herramientas, donde en todos los casos se obtuvo aplicaciones multiplataforma totalmente funcionales, relevando así a Xamarin como una de las mejores alternativas que conjugan buen rendimiento y bajo costo. Se puede mencionar bajo costo de desarrollo por la razón que Xamarin en la actualidad es un entorno de desarrollo gratuito desde febrero del 2016, puesto que cuenta con su propio IDE llamado Xamarin Studio, facilitando también la generación de aplicaciones para Windows, y tablets (Nahuel , 2017).

Justificación Tecnológica

Los motivos o las razones por las cuales se lleva a cabo la realización de este trabajo final de titulación son por dar a conocer alternativas de aprendizaje que usen como recurso los dispositivos de la vida cotidiana, Esta herramienta fue elegida con el fin de contribuir a la creación de aplicaciones dirigidas hacia distintos sistemas operativos móviles, aprovechando al máximo el código que se emplea en estas, ya que actualmente los programadores no tienen el conocimiento suficiente para emprender aplicaciones escalables con herramientas nuevas, como lo es Xamarin.

Impacto Económico

En un futuro se busca que el desarrollo de aplicaciones móviles multiplataforma se convierta en una forma de programación rápida, accesible y a bajo costo, donde las empresas no tendrán que invertir grandes cantidades de recursos con el fin de reescribir sus aplicaciones para cada plataforma móvil.

Cabe mencionar que con la implementación de la aplicación multiplataforma para el cantón Cotacachi se pretende aumentar el nivel turístico y fomentar el desarrollo del cantón.

Contexto

Dentro del ámbito móvil no se ha podido apreciar un aplicativo accesible y de múltiple plataforma que dé a conocer la información turística del cantón Cotacachi, en el mercado existen diversas aplicaciones de este tipo pero que no abarcan al cantón, lo cual genera una oportunidad de desarrollo móvil que sea exclusiva para el cantón.

Mediante el repositorio digital de la UTN se pudo observar un estudio de aplicaciones multiplataforma diseñadas en la herramienta IBM WORKLIGHT STUDIO, a diferencia de este trabajo final de carrera se toma como objeto de estudio a Xamarin como una nueva herramienta de desarrollo.

Adicional a este estudio existe en trabajo de carrera similar con el nombre de “Guía Turística bajo un entorno WEB para la ciudad de Ibarra” correspondiente a Pusda Chulde Segundo Eliceo, el mismo que posee un entorno turístico pero no está desarrollada para ambientes móviles, como se pretende en este trabajo.

CAPÍTULO 1

1. MARCO TEÓRICO

1.1. Turismo

Turismo es el desplazamiento temporal de las personas, en periodos no mayores a un año a lugares que se ubican fuera del entorno habitual, las actividades realizadas durante el tiempo de visita y las facilidades ofrecidas son creadas para complacer las exigencias de los turistas. (MATHIESON, 1982)

La actividad turística comprende un sistema que interrelaciona diferentes factores como la demanda, la oferta, el espacio geográfico y las operaciones del mercado. Donde la demanda está formada por los turistas o consumidores, la oferta son los productos y servicios involucrados en la experiencia turística, el espacio geográfico es el lugar donde se da el encuentro de la oferta con la demanda, y los operadores del mercado son los organismos cuya función es brindar facilidades de interrelación entre la oferta y la demanda, en casos son las agencias de viajes o promociones turísticas.

1.1.1. Gestión Turística

La gestión turística son todas las actividades que comprende el turismo, en el mercado los operadores son quienes están encargados de esta gestión, innovando procesos que impulsen y mejoren la asistencia técnica, capacitación y la implementación de buenas experiencias turísticas en los establecimientos. Además, tratan de reconocer y diferenciar los trabajos y ofertas de los empresarios turísticos con la entrega de reconocimientos a la calidad. De modo que, al ofrecer reconocimiento, las empresas se comprometen en mejorar la calidad de sus servicios y productos, en efecto, fomenta el crecimiento de la demanda turística. (EMPQT, 2014)

En el marco turístico un proceso tiene inicio con el registro de todas las ofertas turísticas de la zona, donde constan los establecimientos de alojamiento, bebidas, alimentación y gastronomía, entretenimiento, transportes de turismo, agencias de viajes, deportes y lugares de ocio. Dando fin al proceso con la debida asistencia técnica que apoya con especialistas que son guía y controladores del cumplimiento de las normativas de calidad dentro y fuera de los establecimientos. Dichos controles se realizan periódicamente en coordinación con una agencia de regulación para fortalecer la calidad de la oferta turística.

1.1.2. Paquetes Turísticos

El conjunto de servicios que se mantienen en oferta forma un paquete turístico, en base a un itinerario organizado con anterioridad. Se lo adquiere en forma de bloque con un solo precio unificado, también es reconocido como un conjunto de servicios ya que en el momento de su ejecución involucra servicios combinados tales como:

- Alojamiento
- Transporte
- Gastronomía
- Recreación
- Visitas guiadas
- Transporte local
- Lugares de diversión

1.1.3. Turismo y tecnologías de la información

La relación entre el Turismo con las Tecnologías de la Información y las Comunicaciones (TIC) forma un potente impulsor en diversos sectores económicos. En los últimos años el turismo y la tecnología han sufrido un crecimiento considerable, ya que el turista con toda la información disponible mediante el uso de tecnología móvil puede organizar y gestionar el proceso de viajes antes, durante y después. (Brilhante & Correa, 2015)

El ámbito turístico es una fuente muy importante en la adquisición de tecnologías; por mantener toda la información aún más accesible para la demanda turística, permite que la oferta y la demanda se comuniquen y compartan información visual, de forma que mejoran las propuestas y paquetes turísticos de las empresas oferentes en el mercado.

La evolución tecnológica siempre ha sido un impulso de crecimiento; en el turismo este avance genera un proceso de transformación que cambiaron las etapas de ejecución de los viajes. Los principales factores son: la distribución de los productos a través de internet, la experiencia turística (ET), que es una visión previa que los turistas reciben, para evaluar la oferta si presenta un grado de satisfacción positiva, y a partir de la evaluación de la ET, adquirir el servicio o no, la razón es sencilla, el consumidor lo único que busca es conocer el producto antes de adquirirlo para no tener ET negativas durante el consumo. (Soares, Mendes, & Barbosa, 2017). Por lo tanto, el mejor lugar para exponer dicha información son los dispositivos móviles inteligentes, como lo recalca también el INEC en su último estudio de encuesta de TIC, afirmando un crecimiento aproximado del 7% anual desde el 2015, de personas que usan teléfonos inteligentes a nivel nacional. (INEC, 2018)

1.2. Dispositivos Móviles

Se define como dispositivo móvil a un pequeño aparato electrónico con capacidades de almacenar y procesar información internamente en su memoria limitada o a través de la conexión a una red, existe un sinnúmero de modelos que llevan a cabo funciones como reproductores de audio y video, geo localizadores GPS, editores de archivos, entre otros. Por ello la demanda de dispositivos móviles ha incrementado notablemente, el grupo etario que conforman las personas de entre 25 a 34 años es el grupo con mayor tenencia de dispositivos con un incremento de 56,9% desde el año 2012 al 2018. (INEC, 2018)

1.2.1. Sistemas operativos de dispositivos móviles

El Sistema Operativo (SO) el software que administra y controla los recursos y procesos de un equipo tecnológico, tanto del hardware que son el disco duro, pantalla y teclado, como del software que son los distintos programas que se instalan comúnmente en un computador, además de mantener activa la interfaz de presentación con la que interactúa el usuario, manipula los archivos, administra procesos. Dicha administración se enfoca en asignar a cada tarea o aplicación en ejecución, el tiempo y recursos de memoria necesarios para que todos terminen su ciclo eficientemente. (López, 2014)

Los SO móviles actúan de la misma manera que actúan los SO comunes, por consiguiente, administra y controla los recursos que tiene un dispositivo móvil como sensores, cámara, pantalla, multimedia, notificaciones, etc.

El primer SO móvil fue el Palm OS creado por Jeff Hawkins en 1992, Jeff con la finalidad de lanzar un producto táctil, económico y básico, logró dar un gran salto en la tecnología para lo que actualmente se conocen como teléfonos inteligentes, en 2001 salió a flote el nuevo sistema operativo Symbian que fue aún más potente que era capaz de enviar y recibir mensajes de fax. (Condori, 2015)

Actualmente los SO móviles han obtenido un extenso crecimiento en diferentes lugares del mundo, de esta manera se genera una enorme demanda de teléfonos móviles en diferentes marcas y modelos, con sistemas que trabajan de igual o más papel que un SO de una computadora, excelente cobertura en comunicaciones y funciones extraordinarias como la edición de documentos y presentaciones. En estos días las empresas más grandes que compiten el mercado son Apple, Google, Microsoft y Huawei.

1.2.2. Sistema Operativo Android

Android es la plataforma móvil más popular por su sistema escalable a múltiples dispositivos, además de ser la más reciente, Android fue desarrollado por un grupo de más de 30 empresas tecnológicas siendo Google el principal aportante. Este SO se lo puede encontrar en numerosos modelos de equipos como lo son los equipos de geolocalización, multimedia, teléfonos móviles y tabletas con características que soportan gráficos 2D, 3D, multimedia, audio, video e imágenes en distintos formatos, opciones de conexión desde Bluetooth, EDGE, 3G, 4G y Wifi. (Joseph, 2013).

Una plataforma en su mayoría de código abierto ya que está fundamentada en Linux, por lo cual muchos fabricantes de equipos móviles lo usan para desarrollar aplicaciones y mejorarlas, una de las principales características de Android es su buena administración de memoria, así como también de procesos. Debido a que la arquitectura está compuesta por secciones; en la primera sección se encuentran todas las aplicaciones que se pueden ejecutar (como Facebook, Gmail, Instagram, WhatsApp, Twitter, etc.), generalmente escritas en lenguaje JAVA, la segunda es el marco de trabajo de las aplicaciones, la cual facilita a los desarrolladores Interfaces de Programación de Aplicaciones (APIs) que permiten programar nuevos componentes capaces de manipular gráficos, archivos, notificaciones, entre otros. La tercera sección corresponde a las librerías o programas integrados escritos en C o C++ como librerías de gráficos, 3D y de bases de datos SQLite. La cuarta es el entorno de ejecución de las capas mencionadas bajo una máquina virtual llamada Dalvik. Por último, se encuentra la sección del núcleo de Linux que gestiona los servicios como gestión de recursos de memoria, administración de tareas, seguridades, controladores de GPS, cámara, audio, Bluetooth, Wifi, entre otros. (Borja, 2014) En la figura 1.1 se representa el contenido de las capas de un SO Android.

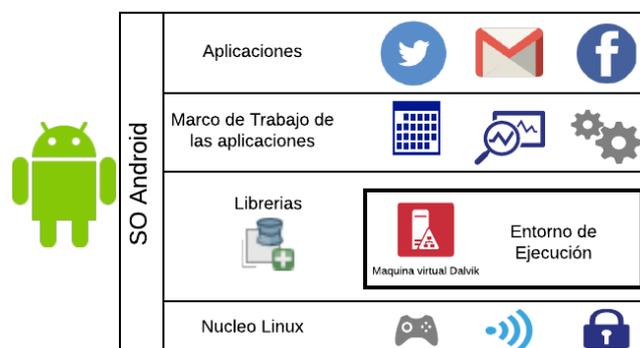


Fig. 1.1. Arquitectura del Sistema Operativo Android

Fuente: Propia

La desventaja de Android es que su mercado esta fragmentado por diversas actualizaciones, esto produce que muchas aplicaciones presentan problemas de compatibilidad o rendimiento.

1.2.3. Sistema Operativo iOS

iOS SO derivado de macOS, creado por la empresa privada de marca Apple Inc. en el 2007, originalmente diseñado para el iPhone, pero también es compatible con iPod Touch, iPad y Apple TV, su última versión es iOS 12 lanzada en 2018. Muchas empresas a nivel mundial lo usan por su alta disponibilidad y su poderosa seguridad, encriptando la información de sus usuarios en tres áreas separadas, puede además acceder de forma segura a redes empresariales privadas por medio de sus protocolos de Red Privada Virtual (VPN), su funcionalidad depende de Microsoft Exchange con fuentes de información basados en estándares para entregar correo electrónico. (Muñoz, 2018)

Posee potentes componentes capaces de tener en pie múltiples tareas sin la necesidad de cerrar alguna, su interfaz fluida transmite simplicidad, optimización y enorme capacidad multimedia, tales características funcionan como pilares fundamentales para que su mercado se mantenga entre los primeros puestos en telefonía móvil, a diferencia de Android este es un SO cerrado, es decir no se puede modificar. Su sistema de seguridad se integra desde el momento en que se enciende, con características diseñadas para protección de virus y malware, garantizando la seguridad de información de sus usuarios, haciéndolo el más seguro y confiable dentro del entorno móvil. (Area Tecnología, s.f.)

La arquitectura de iOS está formada por cuatro capas fundamentadas en el SO MAC OS X. La primera está constituida la capa Cocoa Touch que cumple la función de interactuar con los usuarios a través de las aplicaciones, la siguiente capa es la denominada Media que da acceso a los datos multimedia, a través de tecnologías de lenguajes C y Objective C. La tercera capa brinda los servicios disponibles del equipo como gestión de redes, base de datos SQLite, y sostén para archivos XML. Por último, está la capa Core OS basada en OS X de Apple que gestiona recursos para seguridades, memoria, tareas y manipulación de archivos. (Vida Navarro, 2017) En la figura 1.2 se muestra la arquitectura básica de un sistema operativo iOS.

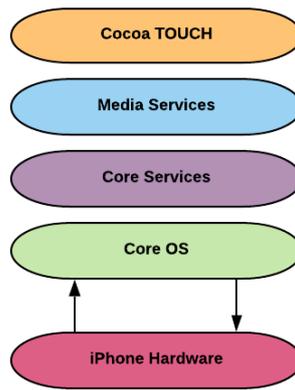


Fig. 1.2. Arquitectura del Sistema Operativo iOS

Fuente: Propia, adaptada de (Vida Navarro, 2017)

1.2.4. Sistema Operativo Windows Phone

Windows Phone es un SO para teléfonos inteligentes lanzado por Microsoft en octubre de 2010, ahora actualizado a Windows 10 Mobile, diseñado con una famosa interfaz de ventanas completamente nueva integrando todos los servicios y aplicaciones basados en el núcleo del SO Windows. La característica principal es la capacidad de optimizar su rendimiento de acuerdo al entorno en que se encuentre. (VENEMEDIA, 2018)

Su desventaja es la poca cantidad de aplicaciones disponibles, debido al desvío de desarrolladores hacia Android.

Según el estudio realizado por Gartner Inc. en 2015 se vendieron 353 millones de dispositivos móviles en el mundo, 15,5% más que en el periodo del año anterior donde el amplio dominador del mercado es Android, utilizado por el 84,7% de los dispositivos vendidos. En segundo lugar, bastante lejos, se ubica iOS, perteneciente a los teléfonos Apple, con 13,1%, luego aparece Windows, con 1,7% y otros más pequeños se reparten el 0,5% restante.

1.3. Aplicaciones Móviles

Una aplicación móvil es un programa de la rama de la informática, diseñado para ser ejecutado en SO de teléfonos móviles, tablets, Smartwatch, iPad, entre otros. Estas aplicaciones dan paso a los usuarios a realizar tareas de distinta índole, tanto como profesional, educativo, de servicio o de ocio.

Entre las más sofisticadas se encuentran: las de gestión de almacenamiento donde es posible acceder a cuentas como Dropbox, permitiendo subir, descargar y enviar archivos de imágenes, videos y documentos enlazados, por otro lado, existen aplicaciones de lectura de

archivos PDF que son de las más utilizadas. En los dispositivos con SO iOS es más común encontrar aplicaciones que rompen los esquemas como la llamada PowerTeacher Mobile, su principal función es llevar el registro de alumnos, sus notas, asistencia y avances, facilitando la importación de los datos a sus mismos alumnos, no obstante, también existe la Keynote permitiendo crear presentaciones de primera categoría con gráficos, animaciones y transiciones similares a las de PowerPoint. (Deusto, 2017)

En lo referente a la figura 1.3. se aprecia la variedad de enfoques que se han utilizado para desarrollar aplicaciones.



Fig. 1.3. Variedad de aplicaciones móviles

Fuente: Tomada de (Deusto, 2017)

Las aplicaciones tienen varias formas de desarrollarse, a causa de que cada una posee diferentes características y limitaciones diferentes, de esta forma se condiciona el diseño visual y la interacción con el usuario final.

1.3.1. Tipos de aplicaciones

Se identifican tres tipos de aplicaciones para equipos móviles de acuerdo a una especificación clara como es el tipo de desarrollo.

- **Aplicaciones Nativas**

Son aquellas que podemos ejecutar dentro del SO móvil, diseñadas para aprovechar todos los recursos de software que ofrecen los dispositivos, estos recursos son bibliotecas de clases o SDK que facilitan el desarrollo. Android, iOS y Windows Phone, cada una posee

diferentes SDK por tener su propia arquitectura de construcción, lo que conlleva a las aplicaciones crecer de forma independientemente. (Cuello & Vittole, 2013)

Por ese motivo las aplicaciones nativas en general son desarrolladas por las mismas compañías que desarrollan los SO móviles, reduciendo el porcentaje de error. Su mejor ventaja es el uso de las notificaciones del SO, lo cual permite mostrar avisos importantes aun cuando no se esté usando la aplicación. (Deusto, 2017)

Al hablar de desarrollo, la principal ventaja es el acceso a todas las características del hardware del dispositivo móvil como GPS, Cámara, Almacenamiento, entre otros, esto logra que los usuarios tengan una mejor experiencia. No obstante, algunos especialistas en la materia consideran que son sistemas cerrados y su desarrollo demanda de un elevado uso de recursos.

En los últimos años los desarrolladores sostienen la hipótesis de que el diseño e implementación de aplicaciones presenta niveles altos de complejidad, esto ha abierto las puertas a soluciones inmediatas orientadas al diseño en HTML, como son las aplicaciones híbridas.

- **Aplicaciones Híbridas**

Combinan el desarrollo nativo con tecnología Web. Los desarrolladores construyen la mayor parte de sus aplicaciones con tecnología Web para diversas plataformas móviles, las cuales están constituidas con un motor de búsqueda HTML incorporado, que funciona como un enlace entre el navegador y las APIs del sistema operativo, permitiendo así que las aplicaciones exploten todos los recursos que disponen los dispositivos. (IBM Corporation, 2012)

La porción Web de una aplicación híbrida puede ser una página Web escrita en HTML, JavaScript, CSS, incluido en el código fuente de la aplicación, haciendo posible introducir pequeñas actualizaciones sin la necesidad de cumplir requerimientos dentro de una tienda de aplicaciones, ventajosamente este modo de crear aplicaciones móviles disminuye su complejidad, pero su principal desventaja consta de la eliminación de cualquier trabajo fuera de una conexión a internet.

- **Aplicaciones Web**

Las aplicaciones web buscan soporte de dispositivos modernos con navegadores capaces de soportar funcionalidades complejas de HTML5, ya que las características de

HTML5 son el acceso a múltiples tipos de medios, servicios de geolocalización y disponibilidad fuera de internet lo que al emplear estas características se puede crear aplicaciones avanzadas únicamente basadas en la Web. Estos sitios especiales tienen la capacidad de reconocer cuando se accede desde una computadora o de un smartphone y modelar las páginas para brindar su información en pantallas escaladas al modelo del dispositivo. Los creadores de este tipo de aplicaciones últimamente han incorporado accesos directos idénticos al que se utiliza para iniciar aplicaciones nativas. (IBM Corporation, 2012)

Su bajo costo de desarrollo y soporte para múltiples plataformas son sus principales ventajas, beneficios bien aprovechados por proveedores móviles para ofrecer aplicaciones alojadas en la plataforma WebKit que es un motor de navegadores que ofrece la más completa implementación de HTML5. A diferencia de las aplicaciones nativas que se ejecutan sobre el SO y tienen acceso completo al dispositivo, las aplicaciones Web se ejecutan dentro del navegador pasando por alto la mayoría de funcionalidades y características que dispone un smartphone.

La TABLA 1.1 representa un análisis de los tipos de aplicaciones mencionados anteriormente.

TABLA 1.1 COMPARACIÓN DE LOS TIPOS DE APLICACIONES

| | NATIVO | HIBRIDO | WEB |
|----------------------|---------------|------------------|-----------------------------|
| Lenguajes | Java, C, .NET | HTML, JavaScript | CSS, HTML, JavaScript, CSS, |
| Costos de desarrollo | Alto | Medio | Económico |
| Interfaz de usuario | Buena | Buena | Regular |
| Rendimiento | Bueno | Medio | Malo |
| Multiplataforma | No | Si | Si |
| Tiempo de desarrollo | Alto | Medio | Bajo |
| Tiendas de APP | Si | Si | No |

Fuente: Propia, adaptado de (Carrasco A. , 2017)

1.3.2. Aplicaciones móviles turísticas

En turismo, las nuevas tecnologías son la herramienta que más apoya a la demanda turística para realizar planes y paquetes de viajes, en el mismo contexto los teléfonos móviles pueden proporcionar sinnúmero de servicios que brindan información que faciliten a los turistas a realizar planes, reservaciones, transporte y encontrar lugares de interés como estaciones de combustible, bares, realizar paseos, etc. (Wang & Park, Julio 2012)

Las aplicaciones en los smartphones durante un viaje ofrecen accesos a servicios que se basan en la geolocalización del usuario para guiar y sitios de esta forma los turistas tengan distintas actividades no planificadas a las cuales podría acudir.

En Ecuador la tendencia del teléfono inteligente en los últimos años ha crecido aceleradamente, en el 2016 el porcentaje de personas que poseen un teléfono inteligente es de 29,7%, pero al 2017 se incrementa al 37,2 % (INEC, 2018), según estudios del INEC el incremento se debe a la demanda que existe por las aplicaciones que poseen los dispositivos, esto quiere decir que cada día los dispositivos y las aplicaciones van tomando mayor fuerza en la vida cotidiana de los usuarios. En el ámbito turístico, también es un fuerte promotor del uso de las aplicaciones móviles por la accesibilidad que poseen, una de las más conocidas es minube, que se ejecuta en Android y iOS para mayor alcance, otra muy reconocida también es Airbnb que también es multiplataforma e interactiva. (Hosteltur, 2019))

De acuerdo al Informe Global con respecto a los contenidos para móviles realizados por Strategy Analytics, los ingresos globales logrados a través del móvil (smartphone & tablet) han alcanzado un 17% más en ventas para el año 2017. En el aspecto turístico estas cifras son favorables ya que también se han elevado el número de descargas de las aplicaciones que se dedican a dicha actividad. De acuerdo a las estadísticas Google Maps y Google Earth engloban más descargas que el resto de sus oponentes, tal como se muestra en la figura.

1.4.

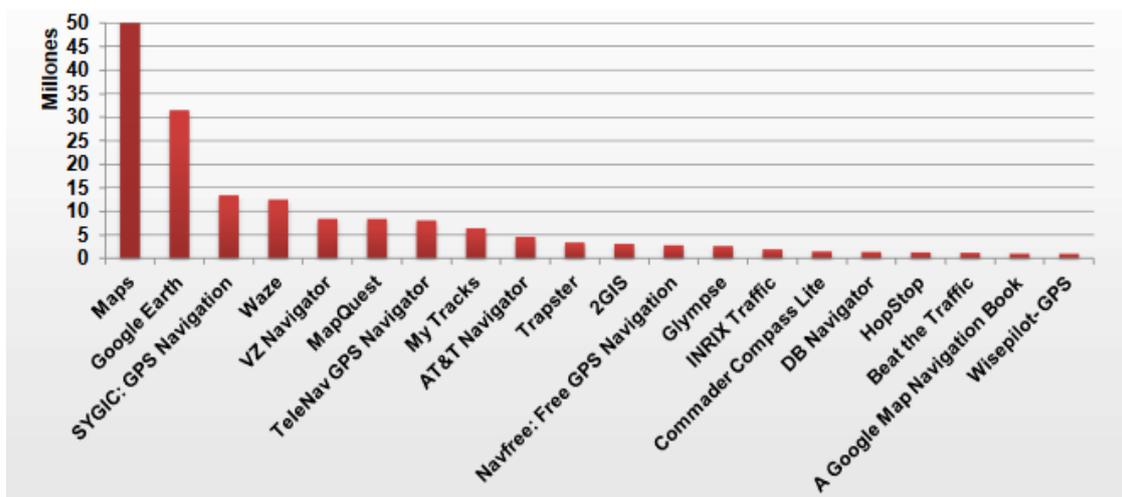


Fig. 1.4. Aplicaciones turísticas más populares

Fuente: (Segittur Turismo e Innovación, 2016)

Los tipos de sistemas más conocidos que contienen aplicaciones turísticas son Android y iOS, pero su principal diferencia se deriva del perfil de usuario. Las aplicaciones iOS tienen un mejor poder económico, multimedia y transacciones por la seguridad que brinda. En la

figura siguiente se plasman las cifras de descargas de las aplicaciones turísticas en los dos tipos de dispositivos.

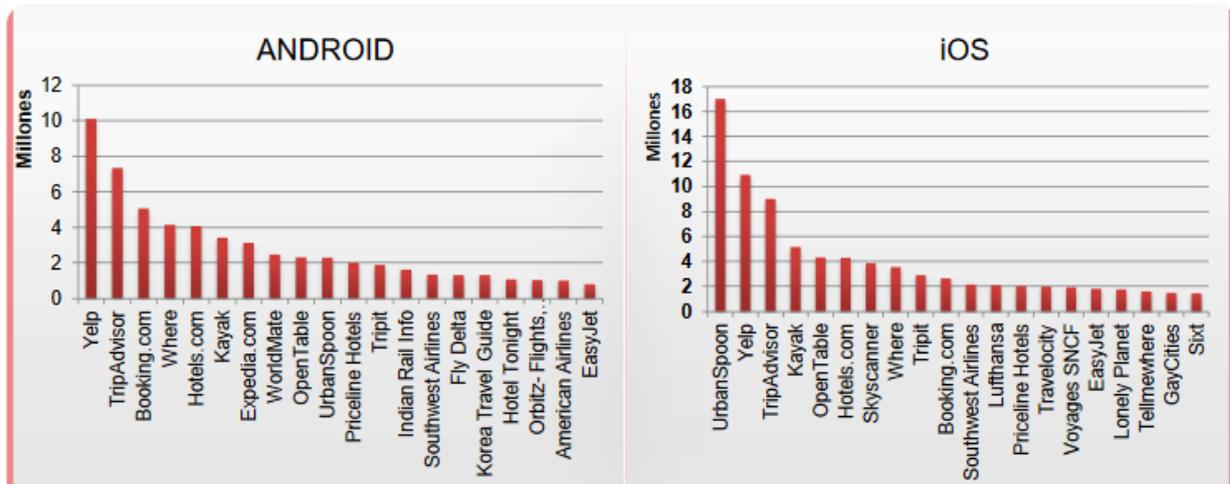


Fig. 1.5. Aplicaciones turísticas en iOS y Android

Fuente: (Segittur Turismo e Innovación, 2016)

1.4. Desarrollo de aplicaciones móviles

El incremento comercial por los dispositivos móviles y las aplicaciones ha obligado a la industria a presentar cambios masivos. Por este motivo existe una inmensa cantidad de oportunidades y disposiciones para crear este tipo de proyectos.

1.4.1. Desarrollo multiplataforma

El desarrollo multiplataforma es una opción para crear aplicaciones para los distintos SO, sin embargo, es considerado como un obstáculo por los diversos retos que se presentan, uno de ellos es la cantidad de paradigmas de interfaz de usuario, puesto que cada plataforma tiene diferentes convenciones para la presentación de datos, diferentes formas de mostrar menús e incluso diferentes reacciones táctiles.

Los diferentes entornos de desarrollo también son un retroceso ya que los desarrolladores usan sofisticados entornos de desarrollo integrado (IDE) para una plataforma específica como lo es Xcode en Mac, que construye aplicaciones para iOS, Android Studio para Android y Visual Studio para aplicaciones de Windows. Si bien, para crear aplicaciones con cada IDE mencionado nace la necesidad de dominar sus lenguajes de programación, en general cada plataforma está asociada con un lenguaje en particular:

- Java para Android
- Objective-C para iPhone
- C# para Windows

1.5. Herramientas de desarrollo móvil multiplataforma

Para terminar con las dificultades del desarrollo enfocado a un solo SO móvil han aparecido entornos de desarrollo multiplataforma como: Xamarin, Delphi y PhoneGap que aportan con grandes herramientas que facilitan la visión, programación y ejecución de las aplicaciones. (Ceballos, 2017)

Xamarin es una plataforma robusta perteneciente a Microsoft, destinada al desarrollo de aplicaciones nativas, basado en el lenguaje de programación C#, al poner en práctica el desarrollo con Xamarin es posible construir métodos potentes compartibles, para esta práctica cabe la necesidad de tener destrezas con C# y conocimiento acerca de .Net con el fin de explotar satisfactoriamente los recursos nativos de cada plataforma (Android, iOS, Windows Phone como se muestra en la figura 1.6).



Fig. 1.6. Desarrollo multiplataforma con Xamarin

Fuente: Propia, adaptada de (Lazaro, 2017)

Como objeto de estudio también se ha tomado al entorno de desarrollo Delphi para un análisis comparativo entre las aplicaciones que se desarrollarán en las dos plataformas.

Delphi es un entorno de desarrollo móvil rápido, ofrece a sus usuarios tres ediciones personalizables para cada tipo de desarrollador, si es el caso de desarrolladores novatos, profesionales y desarrollo empresarial. Hace uso del lenguaje de programación Object Pascal y C# para generar código nativo para las plataformas Windows, macOS, iOS, Android y Linux. (Leboso, 2016)

1.5.1. Xamarin

1.5.1.1. Historia

La necesidad de construir aplicaciones novedosas llevó a estudiosos como Miguel de Icaza y Nat Friendman a fundar la empresa Ximian , en 2001 ponen en marcha su proyecto

Mono para permitir a los desarrolladores de Linux el uso de un nuevo y potente lenguaje de programación con el nombre de C#, en el año de 2003 al ver la calidad del sistema la empresa Novell adquiere el proyecto Mono ya que no se limitaba a un solo sistema operativo Linux sino que su diseño tenía inmerso tecnologías como Android, Mac OS, iOS, y Wii. Novell incluyó a su nuevo proyecto sus proyectos Mono Touch y Mono Android haciendo posible la construcción y ejecución de aplicaciones escritas con un solo lenguaje en los sistemas móviles Android y iOS.

En 2011 la empresa Novell es comprada por una compañía mayor llamada Attachmate quien tuvo diferentes ideales que provocaron recortes de personal, a partir de este hecho las personas que fueron despedidas y tenían suficientes conocimientos se asociaron para darse a conocer como Xamarin, quien creció rápidamente por impulsos económicos que apoyaban su potencial. Seguidamente Microsoft adquiere la empresa Xamarin para ofrecer a sus desarrolladores la mejor herramienta multiplataforma combinando Xamarin con .Net, pero lo que más impresionó a sus usuarios es que se lanzó como una herramienta de código libre.

1.5.1.2. ¿Qué es Xamarin?

Es un entorno de desarrollo, destinado para crear aplicaciones multiplataforma que se ejecutan en dispositivos con SO Android, iOS y Windows, usa un mismo lenguaje de programación C# para todas las plataformas, beneficiando a los desarrolladores ya que para lograr una aplicación multiplataforma no hay necesidad de aprender el lenguaje de cada plataforma, es decir que puede trabajar conociendo simplemente las características de Xamarin y C#. Como se menciona anteriormente Xamarin pertenece a la empresa Microsoft por tanto se encuentra incluido dentro del paquete de Visual Studio. (Muñoz, 2018)

Las características que sobresalen son:

- Posee conexiones para la mayoría de paquetes SDK de las plataformas haciendo aún más compatible, esto reduce en gran medida posibles errores al momento de realizar la ejecución.
- Permite usar código específico para cada plataforma como Object-C para iOS, Java para Android y C++ para Windows.
- Dispone de una API unificada que hace posible manipular los recursos de las plataformas, centrando así en una sola área el control de las tres plataformas.

1.5.1.3. Elementos de Xamarin

Xamarin depende también de herramientas como

- **Xamarin.Forms**

Es un paquete de herramientas encargado de diseñar la interfaz de usuario de las plataformas, logrando crear aplicaciones con características semejantes a las aplicaciones nativas comunes. En un proyecto Xamarin, los elementos se comparten al momento de desplegar la aplicación, es decir mantiene separado el código compartido, pero al momento de ejecutarlo se adhiere a la plataforma elegida. (Xamarin, 2016)

El desarrollo con Xamarin se puede realizar en equipos de Windows y macOS, no obstante, para el desarrollo de aplicaciones Windows necesariamente se debe disponer de un equipo Windows con una instalación de Visual Studio, de igual manera la ejecución de una aplicación iOS requiere de un entorno de desarrollo propio, dando lugar a la necesidad de disponer de un equipo MAC conectado.

Xamarin.Forms no tiene limitaciones si de acceso a las APIs se trata, ya que puede usar cualquier API de las tres plataformas y acceder a los recursos como cámaras, sensores, red, etc. El diseño de una interfaz con Xamarin se puede desarrollar de dos maneras, la primera es mediante la programación en código C# y también mediante el código de enmascado MAML.

Xamarin.Forms implementa un cuadro de herramientas para facilitar la construcción de la interfaz, apoyado en un panel donde muestra automáticamente el diseño de la interfaz solo con arrastrar los componentes. En la figura 1.7 se aprecia una captura de las herramientas disponibles de diseño con Xamarin.

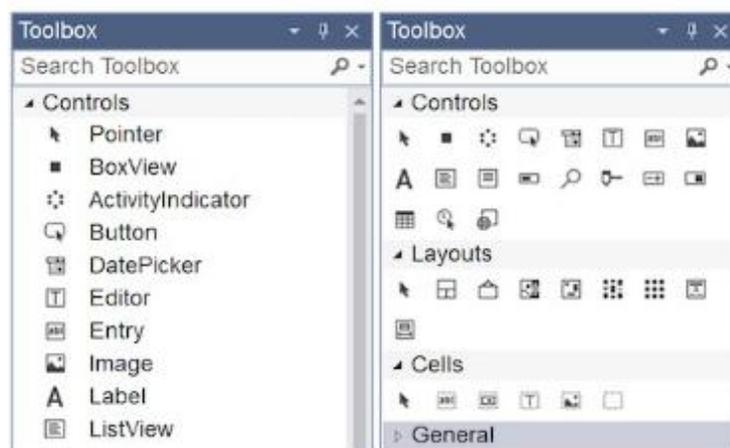


Fig. 1.7. Panel de herramientas de Xamarin.Forms

Fuente: Propia

- **.NET Framework**

Es una plataforma de perfil móvil de código abierto con múltiples herramientas y bibliotecas disponibles bajo estándares de calidad. Muchas implementaciones de .NET están

disponibles para las tres plataformas es por eso que puede crear una infraestructura común, es decir que, la plataforma Xamarin al combinarse con .NET Framework puede ofrecer a sus desarrolladores aplicaciones nativas multiplataforma con un rendimiento similar a las aplicaciones creadas Java para Android y Swift para iOS.

En los últimos años ha nacido .NET Core que es la última versión de .NET, actuando como una sola instalación en particular de .NET ya que anteriormente se incluía dentro de la instalación de Microsoft Windows.

- **Lenguaje de marcado XAML**

El lenguaje XAML es parte de Xamarin.Forms, fue creado para diseñar interfaces de usuario más organizado, además de ser ampliamente compatible con .NET framework y adaptable a la arquitectura MVVM (Modelo Vista Vista-Modelo).

Ventajas del código XML

- Posee una jerarquía de controles organizada
- Se los puede configurar a través de herramientas visuales y manuales.
- Posee un archivo enlazado de código que define eventos a través de los controladores.
- Cada control posee propiedades adjuntas administrables.

Después de crear la solución dentro del proyecto principal podemos visualizar la clase MainPage.Xaml y su archivo asociado MainPage.Xaml.cs en C#.

La primera parte del archivo consta de un ContentPage que contiene los enlaces de referencia a las clases de Xamarin.Forms, .NET Standard la subclase en C# y la parte interna StackLayout como se representado en la figura 1.8 consta del diseño y personalización de la vista.

```
<ContentPage xmlns="http://xamarin.com/schemas/2014/forms"
             xmlns:x="http://schemas.microsoft.com/winfx/2009/xaml"
             xmlns:local="clr-namespace:XamlSamples"
             x:Class="XamlSamples.MainPage">

    <StackLayout>
        <!-- Place new controls here -->
        <Label Text="Welcome to Xamarin Forms!"
              VerticalOptions="Center"
              HorizontalOptions="Center" />
    </StackLayout>

</ContentPage>
```

Fig. 1.8. Lenguaje de enmarcado XAML

Fuente: (Carrasco A. , 2017)

1.5.1.4. Arquitectura de una aplicación en Xamarin

El patrón MVVM (Modelo Vista Vista-Modelo), el Modelo representa la capa de datos y la lógica de negocio de forma que contiene la información, pero no los componentes que la manipulan, la Vista contiene la interfaz de usuario, incluidos los elementos visuales y de entrada que componen la aplicación, y el Vista-Modelo contiene la lógica de presentación de la información, se comunica directamente con la Vista a través de enlaces de datos que devuelven notificaciones que indican si la información fue recibida o no. (Petzold, 2016)

Xamarin se adapta el patrón MVVM para separar el código mediante la creación de cuatro proyectos para cada plataforma, el primer proyecto es la parte de la aplicación que es independiente de las plataformas, puede ser un proyecto de archivos compartidos (SAP) o una biblioteca de clases portables (PCL) donde se encuentra todo el código común que puede ser referenciado por las plataformas, a su vez contiene acceso a la biblioteca de clases de .NET Framework, por lo que puede acceder a servicios web y descomponer archivos JSON y XML. En la figura 1.9 se muestra como Visual Studio genera una solución de Xamarin que contiene los cuatro proyectos interrelacionados, donde las aplicaciones implementadas hacen llamadas al proyecto común.

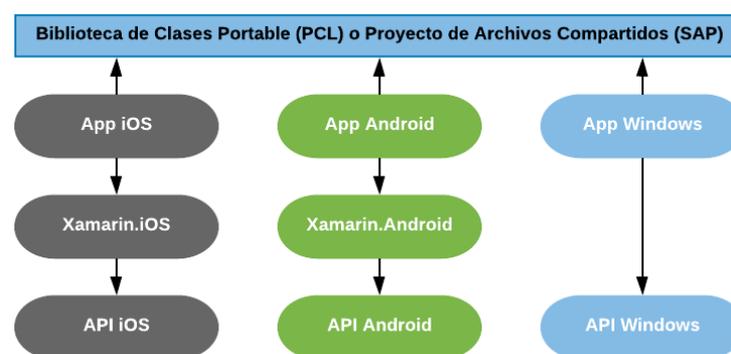


Fig. 1.9. Arquitectura de una aplicación con Xamarin.

Fuente: Propia, adaptada de (Petzold, 2016)

1.5.1.5. Desarrollo con Xamarin

- **Tipos de proyectos**

Xamarin permite crear dos tipos de soluciones, el primero con un proyecto de archivos compartidos (SAP) y el otro con una biblioteca de clases portables (PCL).

La solución generada con un proyecto PCL quiere decir que todo el código se agrupa en una biblioteca con vínculos donde cada aplicación se vincula en tiempos de ejecución, la ventaja es la centralización del código compartido pero la desventaja es que los proyectos tanto iOS como Android y Windows Phone poseen accesos a versiones diferentes de .NET

que pueden producir descoordinación. Para ello Xamarin posee una opción que se encarga de crear una solución con una versión .NET Standard 2.0 que proporciona el mejor nivel de compatibilidad para las aplicaciones.

Las aplicaciones basadas en SAP en cambio permiten escribir código común que se incorpora en todos los proyectos de destino, es decir que al momento de compilar se comporta como una copia para cada proyecto que hace referencia.

En la figura 1.10 se puede apreciar las diferencias de los tipos de proyectos en Xamarin.

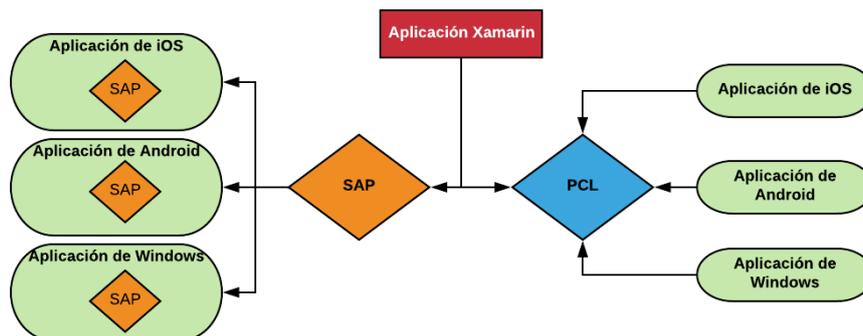


Fig. 1.10. Tipos de proyectos en Xamarin

Fuente: Propia

- **Estructura de una aplicación PCL**

Xamarin organiza el código en distintos proyectos, como se puede mostrar en la figura 1.11, la solución consta de cuatro proyectos.

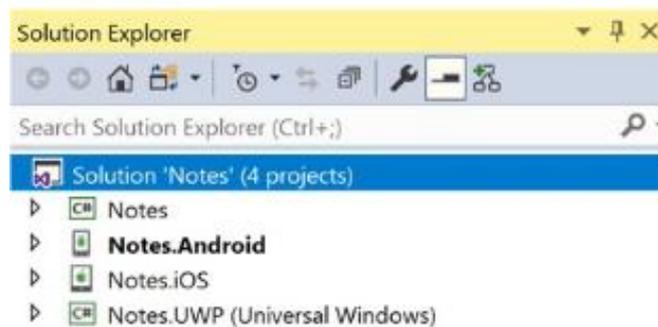


Fig. 1.11. Proyecto PCL

Fuente: Propia

El primer proyecto corresponde a la interfaz de usuario compartida y de la biblioteca .NET Estándar, los demás proyectos corresponden al código de cada plataforma.

La estructura de archivos de una aplicación .NET Standar se describe en la figura 1.12.

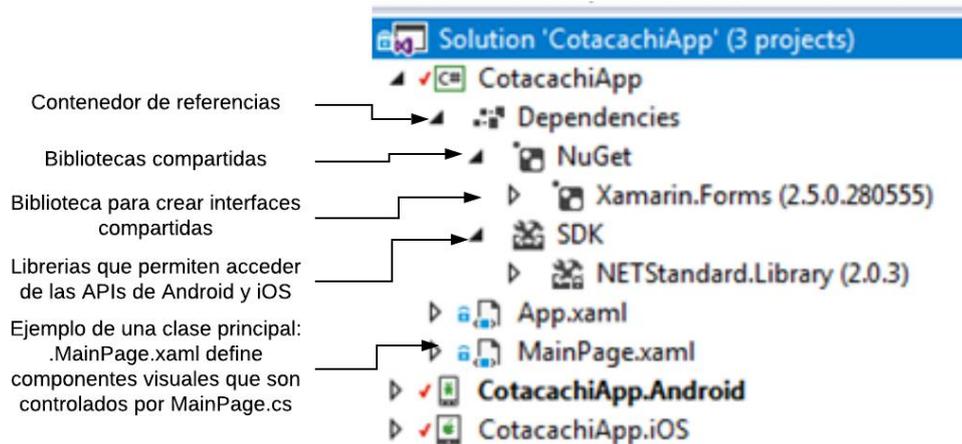


Fig. 1.12. Estructura de un proyecto PCL

Fuente: Propia

- **Diseño de una interfaz (Diseños de Xamarin.Forms)**

Una interfaz de usuario diseñada por Xamarin.Forms proviene también de propiedades de alineación o expansión expuestas dentro de las opciones del diseño de vista de un componente.

En la siguiente tabla 1.2 se detallan los campos usados para precisar alineaciones de vista dentro de un elemento principal o contenedor.

TABLA 1.2 ELEMENTOS DE DISEÑO DE XAML

| Campo | Alineación | |
|--------|---|---|
| | Horizontal | Vertical |
| Start | Lado izquierdo | Parte superior |
| Center | Alineación al centro | Alineación al centro |
| End | Lado derecho | Parte inferior |
| Fill | Rellena el diseño del elemento primario | Rellena el diseño del elemento primario |

Fuente: Propia, resumida de (Xamarin, 2016)

Los campos de expansión de un control mantienen la alineación, pero establecen el espacio que ocupa si existe espacio disponible dentro del elemento que lo contiene.

Dentro de las propiedades de los controles también están inmersos los márgenes y rellenos que son parte fundamental para un diseño personalizado de la interfaz de usuario.

- **Fuentes de Datos**

La fuente de datos de una aplicación Xamarin proviene de servicios web consumidos por APIs muy eficientes y rápidas que permiten mostrar una vista recompilada con todo tipo de información.

Las fuentes de datos usadas por Xamarin son a) Fuentes de archivos JSON, b) Fuentes de datos de Azure y c) fuentes de datos de AzureEazyTable.

Las bases de datos locales también son un fuerte para el almacenamiento, ya que permite la manipulación de un motor de base de datos SQLite para cargar y guardar información en código compartido. Para implementarlo existe la necesidad de importar un paquete NuGet `sqlite-net-pcl` que contiene las librerías que permiten su configuración.

1.5.1.6. Requerimientos e Instalación

Para lograr estabilidad y buen rendimiento de la plataforma en su versión 2017 el equipo debe cumplir con los requisitos mínimos descritos en la TABLA 1.3.

TABLA 1.3 REQUERIMIENTOS DE INSTALACIÓN DE XAMARIN

| Parámetros | Windows | Mac |
|-------------------|-------------------------------------|-------------------------------|
| Sistema Operativo | Windows 10 | macOS High Sierra |
| Xamarin.iOS | SDK de iOS 10 (instalado en un Mac) | SDK de iOS 11 |
| Xamarin.Android | Android 6.0 (nivel de API 23) | Android 6.0 (nivel de API 23) |
| Procesador | 1,8 GHz o superior | 1,8 Ghz o superior |
| Memoria RAM | 4 GB | 4 GB |
| Disco | 130 GB | 130 GB |

Fuente: Propia, adaptada de (Microsoft Visual Studio, 2018)

El proceso de instalación se la realiza como una instalación de Visual Studio, para lo cual se descargada el software desde la página oficial visualstudio.microsoft.com, en este caso se cita Visual Studio Community por ser una versión gratuita. Una vez descargado se ejecuta para iniciar con la instalación.

En la pantalla de instalación se selecciona el entorno de Desarrollo Móvil con .NET como se muestra en la figura 1.13.

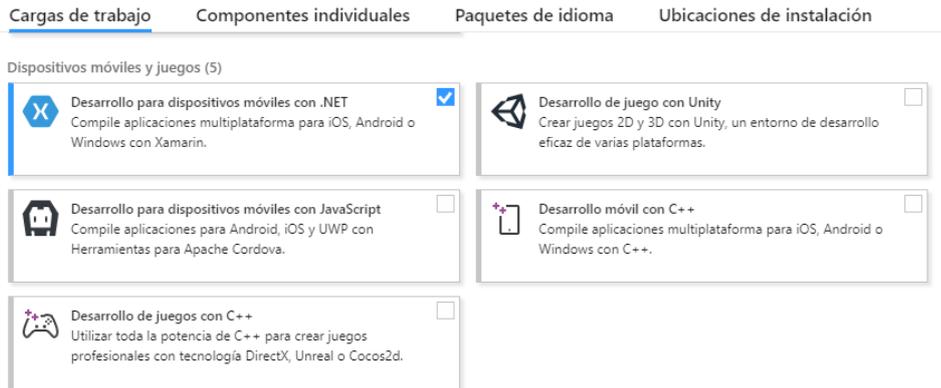


Fig. 1.13. Instalación de Xamarin

Fuente: Propia

En la figura 1.14 se observa los detalles de instalación donde se agrega paquetes adicionales.

Detalles de la instalación

- > Editor de núcleo de Visual Studio *
- ✓ Desarrollo para dispositivos móviles con .NET *
- Incluidos
 - ✓ Xamarin
 - ✓ Herramientas de desarrollo de .NET Framework 4.6.1
 - ✓ C# y Visual Basic
 - ✓ Paquete de compatibilidad de bibliotecas portátil...
- Opcional
 - ✓ Instalación de Android SDK (nivel de API 27)
 - ✓ Google Android Emulator (nivel de API 27)
 - Xamarin Workbooks
 - ✓ Hardware Accelerated Execution Manager (HAXM) ...
 - ✓ Herramientas de la plataforma universal de Windo...

Fig. 1.14. Detalles de instalación de Xamarin

Fuente: Propia

Se recomienda instalar los detalles por defecto, pero el paquete de idiomas debe ser en inglés para recibir soporte por cualquier problema que se pueda presentar.

1.5.2. Delphi (Embarcadero RAD Estudio)

1.5.2.1. Historia

Desde 1993 la empresa estadounidense Embarcadero Technologies desarrolla herramientas para reducir limitaciones, costos y facilitar la creación de aplicaciones sobre múltiples plataformas, así como su diseño y la gestión de bases de datos.

A partir de su fundación Embarcadero ha presentado las siguientes versiones de RAD Studio para el desarrollo multiplataforma:

TABLA 1.4 HISTORIA DE RAD STUDIO

| Versión | Lanzamiento | Características |
|------------------------|--------------------|--|
| RAD Studio XE5 | 2013 | Desarrollo de Android con Delphi y aplicaciones iOS con C++ Nuevos componentes para iOS y Android |
| RAD Studio XE6 | 2014 | Desarrollo de aplicaciones iOS 7 Componentes de barra de tareas Aplicaciones de compras y publicidad Servicios en la nube Desarrollo para Google Glass |
| RAD Studio XE7 | 2015 | Los controles se ajustan según la plataforma destino El componente MultiView, muestra directa como se ve la aplicación Acceso a bases de datos. |
| RAD Studio 10 Seattle | 2017 | Incluye SDK de Windows para Windows 7, 8 y 10 Plataforma FMX para crear aplicaciones nativas Android y iOS |
| RAD Studio 10.1 Berlin | 2018 | Sugerencias en interfaz y diseños Desarrollo en múltiples formatos de teléfonos, tabletas, equipos de escritorio. Vista en tiempo real de interfaces Inspector de objetos mejorado. |

Fuente: Tomado de (Leboso, 2016)

1.5.2.2. Qué es RAD Studio Delphi

RAD Studio es un entorno de desarrollo multiplataforma diseñado para construir aplicaciones para móviles y ordenadores mayormente catalogados como lo son: Android, iOS, Windows y Mac. Esta plataforma facilita la programación orientada a objetos con el uso de lenguajes muy potentes como C++ y Pascal o denominado también Delphi.

Comercializado por Embarcadero Technologies desde 2008 hasta la actualidad en su última versión RAD Studio 10.1 Berlin lanzada en agosto de 2018, Embarcadero realiza herramientas que mejoran la calidad de los trabajos en distintas plataformas que manipulan bases de datos, sistemas operativos y lenguajes de programación, pretende facilitar el diseño, la administración y programación de aplicaciones en todos los entornos o plataformas posibles. RAD Studio es una herramienta no pública, es decir no accesible a cualquier usuario.

Entre las características más relevantes se encuentran:

- Desarrolla aplicaciones combinadas con los dos lenguajes combinados C++ y Pascal.
- Las herramientas que ofrece dependen directamente de la versión que se esté usando.
- Acceso a bases de datos.

1.5.2.3. Elementos de RAD Studio Delphi

RAD Studio Delphi este compuesto por tres bibliotecas de clases como:

- **FMX**

FireMonkey es una biblioteca diseñada para construir interfaces de usuario especialmente para aplicativos multiplataforma, su principal característica es que puede desarrollar dos tipos de interfaces como las clásicas 2D y la visualización de paisajes en 3D, o también combinadas.

- **VCL**

Esta biblioteca (Visual Component Library) exclusivamente es usada para el desarrollo de interfaces con componentes visuales destinadas para Windows, aplicaciones de tipo web y de bases de datos.

- **RTL**

La librería Run Time Library o también llamada como Librería de tiempo de ejecución es un conjunto de tareas que pone en marcha el entorno de desarrollo de las plataformas haciendo peticiones a los compiladores correspondientes, se compone de unidades totalmente compatibles con las librerías de diseño de interfaces VCL y FMX.

1.5.2.4. Estructura de una aplicación en RAD Studio

Una arquitectura confiable para Rad Studio es la arquitectura en capas, ya que trata al máximo que la aplicación sea escalable, a su vez para seguir la buena práctica de programación se sigue una arquitectura de tres capas y un patrón de diseño donde se explote la reutilización de código y de componentes. (Leboso, 2016) En la arquitectura de tres capas actúa la capa de presentación que comunica al usuario la información que la aplicación maneja y recibe sus peticiones. Una explicación clara de la arquitectura se manifiesta en el literal 1.3.1 del presente informe.

1.5.2.5. Desarrollo con RAD Studio

Como se ha mencionado, las plataformas iOS, Windows, MacOS y Android son el objetivo principal de RAD Studio, pero el desarrollo en iOS y Mac se dificulta ya que es necesaria una conexión a un entorno de desarrollo de aplicaciones Mac, lo que significa que obligatoriamente debe conectarse a un equipo Mac y configurar el IDE con certificados de Apple obtenidos. Si la plataforma de destino es Windows no se necesitará de una conexión extra ya que está desarrollado bajo un SO Windows.

El desarrollo con RAD Studio pretende ser el más completo por incorporar funcionalidades y herramientas tales como:

- **Desarrollo para el “Internet de las cosas” (IoT)**

Funcionalidades de IoT que permiten desarrollar aplicaciones que se pueden usar en el vivir de cada día como monitores de ritmo cardiaco, linternas y controles de electrodomésticos. Estos componentes ayudan a aumentar las propiedades de una aplicación al tener varios dispositivos conectados, estos componentes se los puede obtener gratuitamente por medio de un enlace GetIt con diseños estándar fáciles de integrar.

- **Bases de datos de alto rendimiento**

La base de datos de alto rendimiento de RAD Studio ofrece sus desarrolladores es InterBase, una base de datos relacional escalable para quienes buscan integrar una en bajo costo y liviana, además garantiza la seguridad, recuperación de información. Al adquirir el producto también se adquiere la licencia de InterBase para el desarrollo más flexible, un número ilimitado de implementaciones.

- **Desarrollo para internet con IP Works**

IP Works reduce considerablemente limitaciones de desarrollo para internet por sus increíbles componentes programables para tareas tales como enviar correos, archivos y la publicación de servicios web. A su vez hace posible la rápida interacción de protocolos de comunicación como FTP, HTTP, SOAP, DNS, SSMPP.

- **Compilación de una aplicación**

La compilación de una aplicación varía según el entorno donde se encuentre desarrollando, en el caso de Android se requiere de la instalación de paquetes SDK para la construcción de la aplicación, conectar y configurar un dispositivo para realizar las pruebas pertinentes o en su opción instalar un simulador compatible con la plataforma.

1.5.2.6. Requerimientos e Instalación

Antes de instalar la herramienta es necesario cumplir con un mínimo de requerimientos de hardware descritos en la TABLA 1.5.

TABLA 1.5 REQUISITOS DE RAD STUDIO

| Especificaciones | RAD Studio |
|-------------------------|--|
| Sistema Operativo | Windows 7 SP1 o superior |
| Procesador | 1.6 GHz o superior |
| Memoria RAM | 2 GB de RAM / 4GB si se desea compatibilidad con FireMonkey 3D |
| Disco | Android 6.0 (nivel de API 23) |
| Procesador | 1,8 GHz o superior |
| Memoria RAM | 4 GB |
| Disco | 6GB y 60 GB |
| Tarjeta de video | Compatible con Direct X 11 con resolución de 1024x768 o superior |

Fuente: Propia, adaptada de (Embarcadero, 2018)

La instalación de RAD Studio inicia directamente desde su sitio oficial <http://docwiki.embarcadero.com> donde se puede obtener la clave de licencia y el archivo de instalación. Al ejecutar el archivo debemos escoger el tipo de activación recibido durante la compra del producto y lo más importante, la instalación de los dos lenguajes de programación.

En la figura 1.15 se detallan las opciones de instalación de RAD Studio.

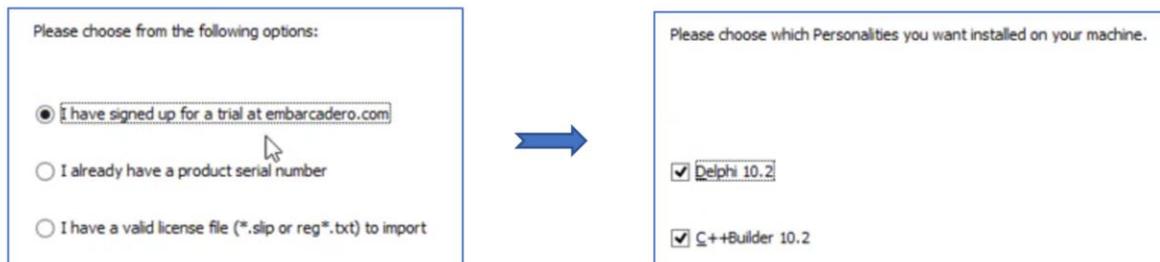


Fig. 1.15. Instalación de Delphi y C++ con RAD Studio

Fuente: Propia

Para las configuraciones siguientes se recomienda usar las selecciones por defecto si no se tiene el conocimiento suficiente.

1.6. Metodología ágil SCRUM

1.6.1. Reseña Histórica

Los métodos ágiles de trabajo nacieron bajo la creencia que un acercamiento profundo con la realidad humana y el desarrollo basados en aprender innovar y cambio son la fuente para obtener mejores resultados. De esta forma se moderniza el modelo de trabajar, de un modelo empírico a un modelo de desarrollo ágil con multifunciones capaces de tomar decisiones que pueden cambiar el rumbo total del equipo de trabajo. En 1986 el artículo Harvard Business Review hizo gran énfasis en trabajos realizadas por grupos exitosos de desarrollo, introduciendo el nombre SCRUM como término al concepto de que el trabajo organizado se mueve mejor en un campo de desarrollo. (Deemer, Benefield, Larman, & Vodde, 2009)

1.6.2. ¿Qué es Scrum?

Scrum es en términos al trabajo asociado, dinámico e incremental para la construcción de proyectos informáticos, dentro de sus funciones está, el definir el ciclo de vida del software, promover un enfoque organizado y de respuestas rápidas a futuros cambios. (Ugan, Çizmel, & Demirors, 2014). La estructura de trabajo de esta metodología está basada en ciclos llamados Sprint.

Capa Sprint o iteración se suele planificar por meses debido a que los requerimientos van cambiando, al terminar un sprint se revisa el trabajo realizado, se procede a realizar las validaciones del mes anterior y planificar las tareas del siguiente Sprint. En la siguiente figura 1.16 se manifiestan los componentes que actúan en la metodología Scrum.

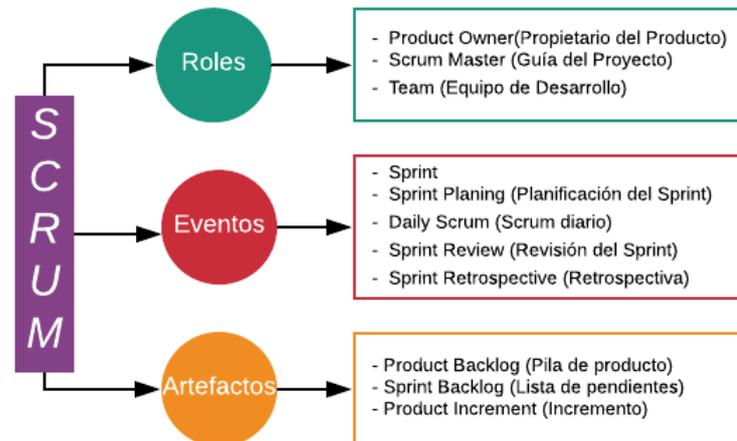


Fig. 1.16. Los 3 componentes de SCRUM
Fuente: Propia, adaptada de (Scrum, 2018)

1.6.3. Roles de Scrum

Scrum crea roles a los participantes de un proyecto, el primer rol lo representa la persona que plantea los requerimientos y prioridades de funcionamiento es decir el dueño del proyecto, el segundo rol está conformado por el equipo de trabajo que desarrolla el producto para el cliente, un equipo de desarrollo de Scrum debe ser capaz de cumplir sus funciones con alto índice de autonomía y responsabilidad. Dentro de las funciones esta la planificación, análisis de requerimientos, programación y pruebas de producto final.

El tercer y último rol es el individuo llamado ScrumMaster quien ayuda al equipo de desarrollo a comprender y aplicar la metodología, este rol no es similar al de un jefe sino el de un guía.

Las actividades de cada rol se detallan en la TABLA 1.6:

TABLA 1.6 ROLES DE SCRUM

| ROL | Descripción | Cantidad | Actividades |
|------------|---|-----------------|---|
| 1 | Dueño de Producto (Product Owner) | 1 | Presentar claramente los requisitos del sistema Ayudar a entender las necesidades Prioriza requisitos |
| 2 | Equipo de Desarrollo (Development Team) | Máximo 6 | Planifica Analiza requerimientos Programa |

| | | | |
|---|---------------------------------------|--|--|
| | | | Realiza Pruebas |
| 3 | Guía del Proyecto 1 (Scrum Master) | | Enseñar al equipo a ser autoorganizado y multifuncional Protege de interferencias externas Guiar al equipo a aplicar SCRUM |

Fuente: Propia, adaptada de (Coronado, 2019)

1.6.4. Eventos de Scrum

Los eventos de Scrum regulan y minimizan posibles reuniones que no se encuentran en la planificación. Cada evento es un tiempo definido donde una vez empleado no puede acortarse o aplazarse, cada evento debe ejecutarse uno después de otro, con un control de cantidad de tiempo especial para no desviarse o desperdiciar recurso. (Shwaber & Sutherland, 2017)

- **El Sprint**

Parte fundamental de Scrum, es un lapso de tiempo definido según se crea un avance del proyecto. Son más productivos si la duración de Sprints se mantiene durante el proceso de desarrollo. Como se mencionaba con anterioridad, cada Sprint se ejecuta después de otro.

- El Sprint Planning (Planificación de Sprint)

Se crea una planificación a partir del trabajo conjunto de todo el equipo Scrum, el tiempo máximo de una planificación es de 8 horas para un Sprint de 30 días. El Scrum Master es el encargado de coordinar y llevar a cabo la actividad dentro del tiempo previsto.

- El Daily Scrum (Scrum Diario)

El equipo de desarrollo toma un tiempo de 15 minutos cada día para planear las tareas de las próximas 24 horas, este Sprint se realiza a la misma hora y en el mismo lugar para de esta forma crear los avances esperados.

- Sprint Review (Revisión de Sprint)

Al finalizar un Sprint el equipo Scrum ejecuta una revisión de Sprint para evaluar el incremento obtenido y las demás actividades realizadas en el transcurso, si en caso existen cambios el equipo socializa y optimiza la problemática, esta reunión se lleva a cabo con una duración límite de 4 horas para un Sprint de 30 días.

- Sprint Retrospective (Retrospectiva de Scrum)

Dentro del Sprint de Retrospectiva cada miembro del equipo se inspecciona a sí mismo para buscar mejoras que sirvan de impulso en el siguiente Sprint. El tiempo límite de la retrospectiva es de 3 horas para un Sprint de 30 días. (Shwaber & Sutherland, 2017)

1.6.5. Artefactos de Scrum

Se denominan Artefactos a todos los elementos obtenidos como resultado al terminar un Sprint. Los Artefactos definidos por Scrum son:

- Product Backlog (Lista de Producto)

Es la lista ordenada jerárquicamente re requisitos propuestos por el Product Owner o dueño del proyecto. Esta lista puede aumentar según avanzan los Sprints. El equipo de desarrollo se encarga de comunicar todas las estimaciones, por su parte el dueño del proyecto socializa y explicar sus necesidades.

- Lista de Pendientes del Sprint

Son los elementos que contiene una Lista de Producto específica de un Sprint, el dueño del proyecto actúa haciendo seguimiento al terminarse cada ciclo.

- Incremento

El incremento es la cantidad de progreso obtenida durante cada Sprint, es importante que al culminar con cada ciclo exista un progreso significativo, de no ser el caso el equipo Scrum si está marchando con satisfacción. (Coronado, 2019)

CAPÍTULO 2

2. DESARROLLO METODOLÓGICO

2.1. Fase 1: Iniciación

2.1.1. Visión de proyecto

La Jefatura de Turismo del Cantón Cotacachi cumple actividades con el objetivo de fortalecer la actividad turística y ofertar servicios de calidad, el proceso turístico del cantón se realiza mediante la identificación de todos los lugares de comercio de la zona mediante documentos y guías turísticas físicas; presentaciones en los medios de comunicación, lo que significa que las hojas físicas no terminan siendo lo necesariamente convincentes para los turistas, por el idioma y la accesibilidad de la información.

La finalidad de la realización de una aplicación turística, es desarrollar un sistema que permita a la Jefatura de Turismo del cantón Cotacachi realizar la gestión de la información turística existente mediante el uso de un sistema web y a su vez mostrarla en la aplicación móvil mediante el uso del Internet, lo que va permitir tener una nueva forma de información digital actualizable de mucha accesibilidad para los usuarios. También el sistema va a permitir tener la información en dos idiomas, inglés y español lo que facilita aún más el proceso de gestión turística de llegar hacia los usuarios con ofertas turísticas comprensibles. A su vez mediante el aplicativo móvil se va a poder realizar consultas de las direcciones de los atractivos desde un mapa georreferenciado que localiza la ubicación del usuario y el lugar a donde desea dirigirse.

2.1.2. Comparativa de las herramientas

A continuación, se muestra el desarrollo de dos prototipos diseñados en las plataformas Xamarin y Delphi, en las cuales se realiza iteraciones concurrentes, para ello se generó un laboratorio de pruebas usando el evaluador de rendimiento de Android, donde se tomará en cuenta el consumo de recursos como: **Energía** (batería), **Rendimiento** (CPU), **Memoria** (RAM) y **Almacenamiento**.

Primera Iteración

La primera iteración se realiza al ejecutar dos aplicaciones recientemente creadas correspondientes a las plataformas ya mencionadas, en la figura 2.1 se muestra el resultado respectivo.

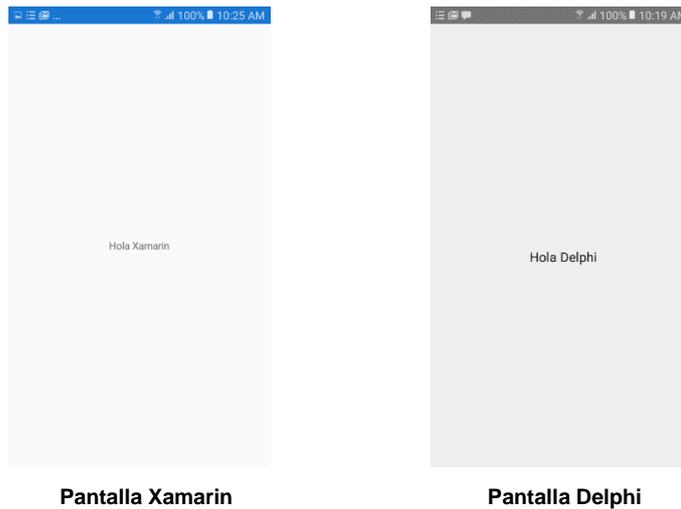


Fig. 2.1. Primera iteración

Fuente: Propia

Energía: Al hacer la primera iteración el consumo de batería es del 0% en ambas plataformas.

Memoria, Rendimiento:

Xamarin: Al realizar la primera iteración consume 36,9 Mb de memoria RAM y un porcentaje del 0% de trabajo en CPU.

Delphi: Al realizar la primera atención consume 50,6 Mb de memoria RAM y un porcentaje de 4,33% de trabajo en CPU.

Almacenamiento:

Xamarin: Al realizar la primera iteración ocupa 33,7 Mb de almacenamiento en el dispositivo

Delphi: Al realizar la primera iteración ocupa 40,6 Mb de almacenamiento-

En la siguiente figura 2.2 se muestran los resultados obtenidos de la primera iteración:

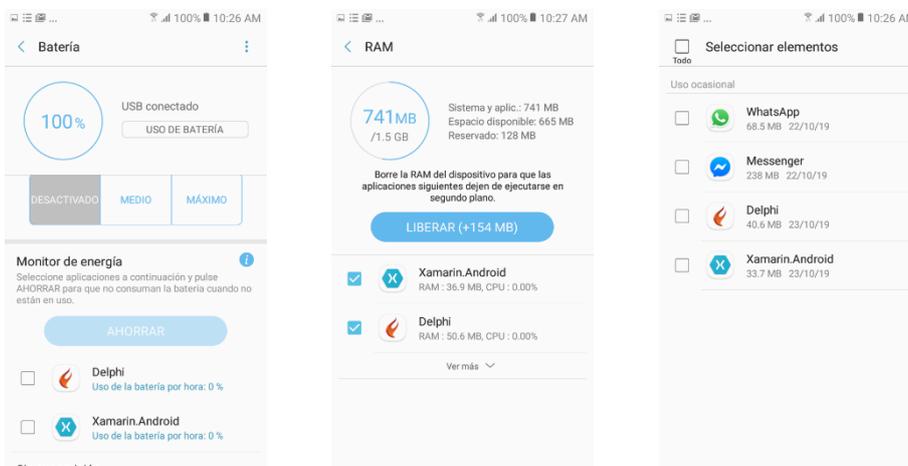


Fig. 2.2. Primera iteración – monitoreo de las aplicaciones.

Fuente: Propia

Segunda Iteración

La segunda iteración se toma al ejecutar el menú principal correspondiente a la aplicación turística que se pretende implementar.

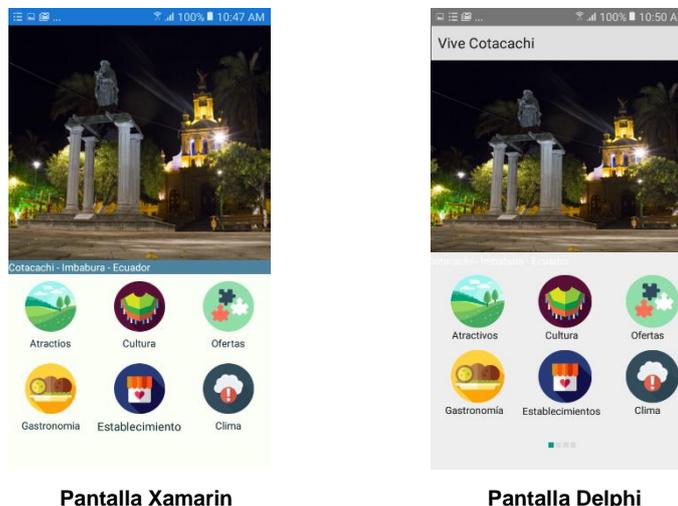


Fig. 2.3. Segunda iteración

Fuente: Propia

Energía: Al hacer la segunda iteración el consumo de batería es del 0% en ambas plataformas.

Memoria, Rendimiento:

Xamarin: Al hacer la segunda iteración consume 85,6 Mb de memoria RAM y un porcentaje del 0,33% de afectación a las tareas que se procesan en CPU.

Delphi: Al hacer la segunda atención consume 107 Mb de memoria RAM y un 5,94% de consumo de CPU.

Almacenamiento:

Xamarin: Al hacer la segunda iteración ocupa 34,5 Mb de almacenamiento en el dispositivo.

Delphi: Al hacer la segunda iteración ocupa 44,6 Mb de almacenamiento.

En la siguiente figura 2.3 se muestran los resultados obtenidos de la segunda iteración:

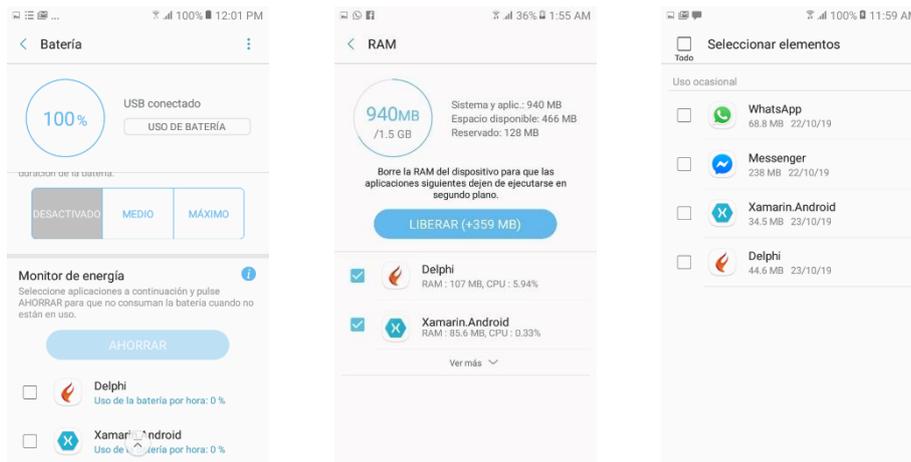


Fig. 2.4. Segunda iteración – monitoreo de las aplicaciones

Fuente: Propia

Tercera Iteración

La siguiente iteración se toma al realizar la consulta del submenú de los tipos de atractivos que se presentarán.

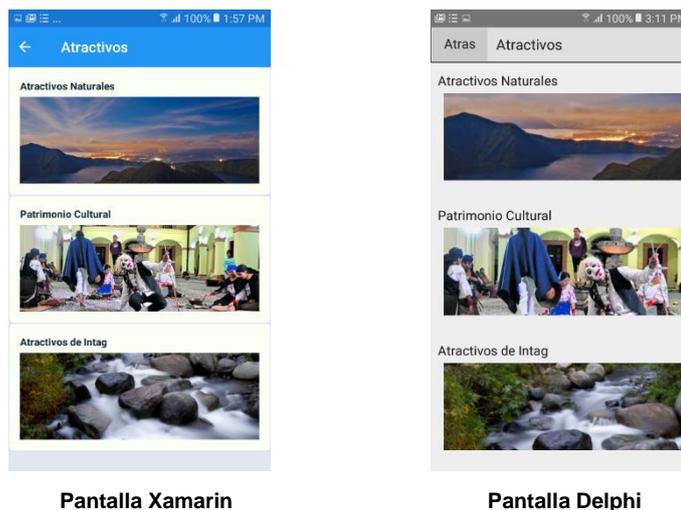


Fig. 2.5. Tercera iteración

Fuente: Propia

Energía: Se realiza con un consumo de batería del 0% en ambas plataformas.

Memoria, Rendimiento:

Xamarin: Al hacer la tercera repetición consume 80,8 Mb de memoria RAM y un 3,04% del total de trabajo en CPU.

Delphi: Al hacer la tercera repetición consume 118 Mb de memoria RAM y un 4,91% del total de trabajo en CPU.

Almacenamiento:

Xamarin: Al hacer la tercera repetición ocupa 35,7 Mb de almacenamiento en el dispositivo.

Delphi: Al hacer la tercera repetición ocupa 58,9 Mb de almacenamiento en el dispositivo.

En la siguiente figura 2.2 se muestran los resultados obtenidos de la tercera iteración:

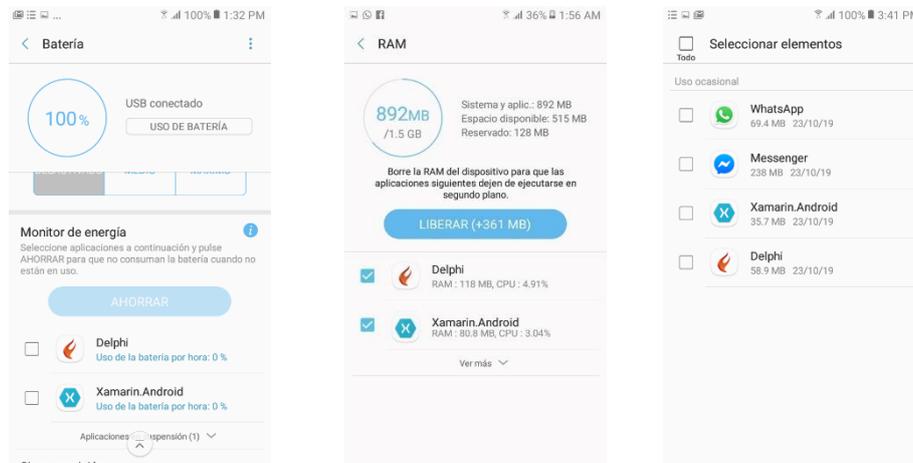
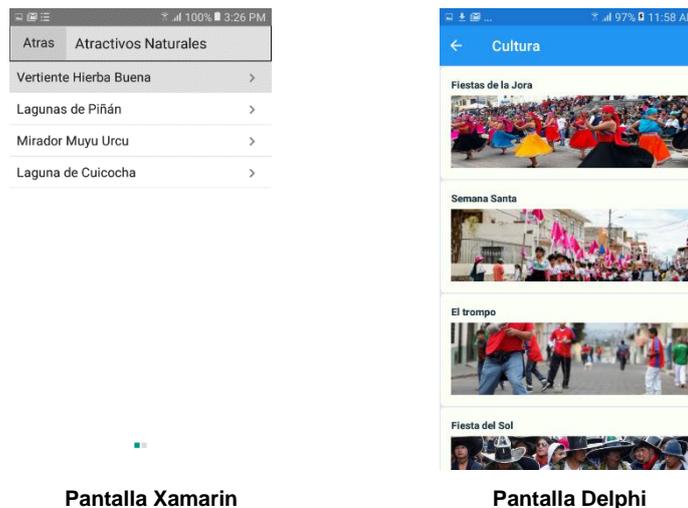


Fig. 2.6. Tercera iteración – monitoreo de las aplicaciones

Fuente: Propia

Cuarta Iteración

La cuarta iteración consta de los parámetros obtenidos al realizar el consumo de servicios web y desplegar el listado de las manifestaciones culturales.



Pantalla Xamarin

Pantalla Delphi

Fig. 2.7. Cuarta iteración

Fuente: Propia

Energía: Al hacer la siguiente iteración se realiza con un consumo manteniendo el 0% en ambas plataformas.

Memoria, Rendimiento:

Xamarin: Al realizar la siguiente iteración disminuye el consumo a un total de 89 Mb y un porcentaje de 4,66% del trabajo realizado en CPU.

Delphi: Al realizar la siguiente iteración consume 131 Mb de memoria RAM y un 5,53% del trabajo en CPU.

Almacenamiento:

Xamarin: Al realizar la cuarta iteración ocupa 37,1 Mb de almacenamiento en el dispositivo.

Delphi: Al ser la cuarta iteración ocupa 59,7 Mb de almacenamiento en el dispositivo.

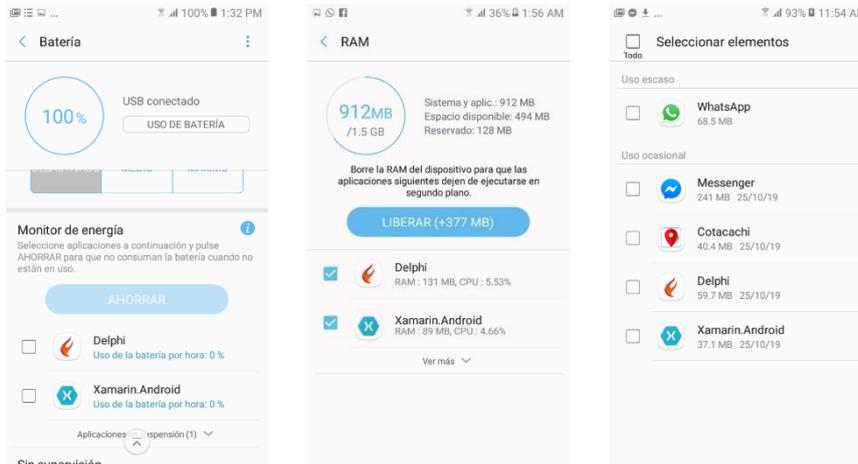


Fig. 2.8. Cuarta iteración – monitoreo de las aplicaciones

Fuente: Propia

Al finalizar el laboratorio del consumo de recursos de los prototipos de cada plataforma de desarrollo se obtuvo los siguientes resultados:

Energía (batería): De acuerdo a cada iteración se muestra un consumo de batería de 0%, por consiguiente, el desarrollo con cada plataforma no provoca un impacto considerable al consumo de energía de los dispositivos.

Memoria (RAM): De acuerdo al análisis de cada iteración se muestra el espacio que requiere en memoria cada prototipo al realizar ejecuciones de nuevos formularios como se observa en la Tabla 2.1.

TABLA 2.1 ITERACIONES MEMORIA RAM(Mb)

| Iteraciones | Xamarin | Delphi |
|-------------|---------|--------|
| 1 | 36,9 | 50,6 |
| 2 | 85,6 | 107 |
| 3 | 80,8 | 118 |
| 4 | 89 | 131 |
| 5 | 93,5 | 157 |
| 6 | 96,3 | 157 |
| 7 | 95,9 | 154 |
| 8 | 101 | 165 |
| 9 | 103 | 169 |
| 10 | 93,5 | 140 |

Fuente: Propia

RENDIMIENTO (CPU): De acuerdo al análisis de cada iteración se muestra el porcentaje del trabajo realizado en la CPU del dispositivo como se muestra en la Tabla 2.2.

TABLA 2.2 ITERACIONES TRABAJO EN CPU (%)

| Iteraciones | Xamarin | Delphi |
|-------------|---------|--------|
| 1 | 0 | 4,33 |
| 2 | 0,33 | 5,94 |
| 3 | 3,04 | 4,91 |
| 4 | 4,66 | 5,53 |
| 5 | 3,69 | 6,03 |
| 6 | 1,81 | 4,43 |
| 7 | 0,59 | 4,58 |
| 8 | 2,7 | 7,71 |
| 9 | 0,57 | 4,39 |
| 10 | 0,47 | 4,09 |

Fuente: Propia

ALMACENAMIENTO: Después de realizar las iteraciones necesarias se muestra el tamaño de almacenamiento que requiere el dispositivo después de realizar sobrecarga de formularios en los prototipos diseñados.

En la Tabla 2.3 se presenta con detalle los resultados obtenidos.

TABLA 2.3 ITERACIONES DE ALMACENAMIENTO (Mb)

| Iteraciones | Xamarin | Delphi |
|-------------|---------|--------|
| 1 | 33,7 | 40,6 |
| 2 | 34,5 | 44,6 |
| 3 | 35,7 | 58,9 |
| 4 | 27,1 | 59,7 |
| 5 | 37,5 | 59,8 |
| 6 | 37,5 | 59,8 |
| 7 | 37,5 | 59,8 |
| 8 | 37,7 | 61,9 |
| 9 | 38,2 | 61,9 |
| 10 | 38,2 | 61,9 |

Fuente: Propia

2.1.3. Definición de roles de Scrum

Según la metodología Scrum se asignan los roles a cada persona participante, con el nombre, la descripción de las tareas y el rol, como se detalla en la TABLA 2.4

TABLA 2.4 DESCRIPCIÓN DE ROLES DEL EQUIPO SCRUM

| Nombre | Descripción | Rol |
|-----------------------------------|-------------------------|--|
| Msc. Diego Trejo | Guía de la metodología | Guía del Proyecto (Scrum Master) |
| Lic. Verónica Cerpa | Pruebas | Propietario del Producto (Product Owner) |
| Jefferson Andrade | Desarrollo del sistema | Equipo de Desarrollo (Development Team) |
| Mauricio Rea Alexander Guevara | Test y validación QA | |

Fuente: Propia

2.1.4. Pila de Producto

Como se puede apreciar en la TABLA 2.5 la pila de producto es una lista donde constan todos los requerimientos que tiene el propietario del producto. Cada requerimiento contiene un id, en nombre de Historia, la prioridad para el desarrollo, la estimación del trabajo que requiere cumplir con el requerimiento y la descripción del mismo.

La estimación de las historias se clasifica en 4 tipos, según el valor de trabajo que representa cada historia al equipo de desarrollo.

TABLA 2.5 PARÁMETROS DE ESTIMACIÓN

| ESTIMACIÓN | VALOR |
|-------------------|--------------|
| Imperativas | 1 |
| Importantes | 2 |
| Necesarias | 3 |
| Prescindibles | 4 |

Fuente: Tomado de (Medina, 2018)

TABLA 2.6 PILA DE PRODUCTO

PROYECTO: LA CREACIÓN DE UN APLICATIVO MULTIPLATAFORMA PARA ORIENTACIÓN E INFORMACIÓN TURÍSTICA DEL CANTÓN COTACACHI

| ID | HISTORIA | PRIORIDAD | ESTIMACIÓN | DESCRIPCIÓN |
|-------------|-------------------------------------|------------------|-------------------|---|
| H-01 | Preparación del ambiente de trabajo | Media | 1 | Instalar todas las herramientas de desarrollo necesarias. |
| H-02 | Acceso al sistema web | Media | 4 | Se debe diseñar un sistema de Login que garantice la seguridad de la información. |
| H-03 | Atractivos turísticos | Alta | 4 | Debe poder crear, modificar y obtener un listado de todos los lugares turísticos como: atractivos |

| | | | | |
|-------------|--------------------------------|-------|---|---|
| | | | | naturales, atractivos patrimoniales y atractivos de Intag. |
| H-04 | Eventos turísticos | Alta | 4 | Debe poder crear, modificar y obtener un listado de todos los eventos turísticos |
| H-05 | Información cultural | Media | 3 | Debe poder crear, modificar y obtener un listado de todas las actividades culturales. |
| H-06 | Gastronomía y restaurantes | Media | 3 | Debe poder crear, modificar y obtener un listado de la gastronomía y restaurantes. |
| H-07 | Ofertas turísticas | Media | 3 | Debe poder crear, modificar y obtener un listado de la gastronomía y restaurantes. |
| H-08 | Establecimientos y comercio | Media | 3 | Debe poder crear, modificar y obtener un listado de los establecimientos de carácter financiero, transporte, tiendas comerciales y hospedaje. |
| H-09 | Registro de clientes | Alta | 4 | Se debe mostrar el listado de todos los clientes que se registren en la aplicación móvil. |
| H-10 | Presentación de la información | Alta | 4 | Se debe desarrollar un aplicativo móvil multiplataforma que permita visualizar la información de lugares turísticos, eventos turísticos, información cultural, gastronomía, restaurantes, oferta cultural y establecimientos. |
| H-11 | Georreferenciación | Alta | 4 | Se debe visualizar un mapa con la ubicación exacta de los lugares turísticos y eventos en relación |

| | | | | |
|-------------|---------------------|-------|---|--|
| | | | | a la ubicación del dispositivo. |
| H-12 | Clima | Media | 3 | Se debe mostrar un campo donde se visualice el estado climático del cantón Cotacachi |
| H-13 | Aplicación bilingüe | Alta | 4 | Toda la información debe mostrarse en dos idiomas. |

Fuente: Propia

2.2. Fase 2: Planificación

2.2.1. Creación de Historias de Usuario

Después de agrupar definir y organizar la pila de producto, se debe descomponer cada una de las historias, detallando qué se debe realizar en cada una y la persona responsable de su ejecución.

TABLA 2.7 HISTORIA DE USUARIO H-01 – Preparación del ambiente de trabajo

| Historia de Usuario | |
|---|--------------------------------------|
| ID: H-01 | Usuario: Analista de Sistemas |
| Nombre de la Historia: Preparación del ambiente de trabajo | |
| Prioridad: Media | Estimación: 1 |
| Riesgo: Bajo | |
| Responsable: Jefferson Andrade | |
| Descripción: | |
| Para el desarrollo del sistema se debe instalar las siguientes herramientas: | |
| <ul style="list-style-type: none"> • Apache/2.4.38 (Win64) OpenSSL/1.1.1b PHP/7.3.2 • MySQL 5.0.12-dev • Composer v1.6 • Laravel 5.5 • Microsoft Visual Studio Community 2017 • Microsoft .NET framework Version 4.7 • VisualStudio.Mac – 1.0 • Xamarin – 4.12.3.81 • Xamarin.Android SDK – 9.17 • Xamarin.iOS y Xamarin.Mac SDK – 12.2 | |

Fuente: Propia

TABLA 2.8 HISTORIA DE USUARIO H-02 – Acceso al sistema web

| Historia de Usuario | |
|---|--------------------------------------|
| ID: H-02 | Usuario: Analista de Sistemas |
| Nombre de la Historia: Acceso al sistema web | |
| Prioridad: Media | Estimación: 4 |
| Riesgo: Alto | |
| Responsable: Jefferson Andrade | |
| Descripción: | |
| <p>El sistema debe tener un registro de usuarios y un inicio de sesión confiable para precautelar la información turística.</p> <p>La gestión del sistema debe ser parametrizada para los tipos de usuario como:</p> <ul style="list-style-type: none">• SuperUsuario: puede ingresar, modificar y actualizar información en todo el sistema web.• Administrador: puede ingresar, modificar, y actualizar información del sistema, sin embargo, la sección de creación de usuario será restringida para este tipo de usuario.• Cliente: el usuario cliente puede solamente visualizar la información del sistema. | |
| Pruebas de aceptación: | |
| <ul style="list-style-type: none">• Al momento de registrar un nuevo usuario se debe enviar un correo de validación del usuario.• Validar el ingreso de acuerdo con las restricciones respectivas para cada tipo de usuario.• El formulario debe mostrar alertas de error si la información no resulta correcta. | |

Fuente: Propia

TABLA 2.9 HISTORIA DE USUARIO H-03 – Atractivos turísticos

| Historia de Usuario | |
|--|--------------------------------------|
| ID: H-03 | Usuario: Analista de Sistemas |
| Nombre de la Historia: Atractivos turísticos | |
| Prioridad: Alta | Estimación: 4 |
| Riesgo: Bajo | |
| Responsable: Jefferson Andrade | |
| Descripción: | |
| El sistema debe tener una sección para administrar la información de todos los atractivos turísticos donde: | |
| <ul style="list-style-type: none">• Se permita ingresar un nuevo atractivo turístico, con los datos siguientes:<ul style="list-style-type: none">Nombre del lugar.Dirección en inglés y español.Horarios de atención en inglés y español.Descripción del atractivo en inglés y español.Responsable del atractivo.La latitud y longitud del atractivo.Teléfono de contacto.Una imagen de representación del atractivo.• Se debe mostrar un listado de todos los atractivos registrados.• Se debe permitir eliminar y editar un atractivo turístico.• Se debe crear una ruta de servicio web donde se listen los atractivos turísticos solamente con la información en español.• Se debe crear una ruta de servicio web donde se listen los atractivos turísticos solamente con la información en inglés. | |
| Pruebas de aceptación: | |
| <ul style="list-style-type: none">• La información de los atractivos naturales, patrimoniales y de Intag debe ser organizada en un menú.• Validar los campos vacíos, numéricos.• Validar los formatos jpg y png para la fotografía.• Mostrar mensajes de alerta en caso de que existan errores en el ingreso.• Las rutas de información deben ser en formato JSON. | |

Fuente: Propia

TABLA 2.10 HISTORIA DE USUARIO H-04 – Atractivos turísticos

| Historia de Usuario | |
|---|--------------------------------------|
| ID: H-04 | Usuario: Analista de Sistemas |
| Nombre de la Historia: Eventos turísticos | |
| Prioridad: Alta | Estimación: 4 |
| Riesgo: Bajo | |
| Responsable: Jefferson Andrade | |
| Descripción: | |
| El sistema debe tener una sección para administrar la información de todos los eventos turísticos donde: | |
| <ul style="list-style-type: none">• Se permita ingresar un nuevo evento turístico, con los datos siguientes: Nombre del evento en español. Nombre descriptivo del evento en inglés. Dirección en inglés y español. Latitud y longitud del lugar donde se va a realizar el evento. Descripción del evento en inglés y español. Fecha del evento. Hora del evento. Una imagen de representación del evento. Estado del evento activo como predeterminado.• Se debe mostrar un listado de todos los eventos registrados.• Se debe permitir eliminar y editar un evento turístico.• Se debe crear una ruta de servicio web donde se listen los eventos activos turísticos solamente con la información en español.• Se debe crear una ruta de servicio web donde se listen los eventos activos turísticos solamente con la información en inglés. | |
| Pruebas de aceptación: | |
| <ul style="list-style-type: none">• La información de los lugares naturales, patrimoniales y de Intag debe ser organizada en un menú.• Validar los campos vacíos, numéricos. formatos jpg y png para la fotografía.• Mostrar mensajes de alerta en caso de que existan errores en el ingreso.• Las rutas de información deben ser en formato JSON.• Cuando se realiza una edición se un evento, se puede cambiar el estado de activo a inactivo. | |

Fuente: Propia

TABLA 2.11 HISTORIA DE USUARIO H-05 – Información cultural

| Historia de Usuario | |
|--|--------------------------------------|
| ID: H-05 | Usuario: Analista de Sistemas |
| Nombre de la Historia: Información cultural | |
| Prioridad: Media | Estimación: 3 |
| Riesgo: Bajo | |
| Responsable: Jefferson Andrade | |
| Descripción: | |
| El sistema debe tener una sección para administrar la información de las actividades culturales donde: | |
| <ul style="list-style-type: none">• Se permita ingresar una nueva actividad cultural, con los datos siguientes: Nombre de la actividad cultural. Dirección en inglés y español. Descripción de la actividad cultural en inglés y español. Fecha de celebración. Una imagen de representación de la manifestación cultural. Estado del evento activo como predeterminado.• Se debe mostrar un listado de todas las actividades registradas.• Se debe permitir eliminar y editar una actividad cultural.• Se debe crear una ruta de servicio web donde se listen las actividades turísticas solamente con la información en español• Se debe crear una ruta de servicio web donde se listen las actividades turísticas solamente con la información en inglés. | |
| Pruebas de aceptación: | |
| <ul style="list-style-type: none">• La información de las actividades turísticas debe listarse y mostrar un acceso para visualizar la información completa.• Validar los campos vacíos• Validar los campos numéricos• Validar los formatos jpg y png para la fotografía.• Mostrar mensajes de alerta en caso de que existan errores en el ingreso.• Las rutas de información deben ser en formato JSON. | |

Fuente: Propia

TABLA 2.12 HISTORIA DE USUARIO H-06 – Gastronomía y restaurantes

| Historia de Usuario | |
|---|--------------------------------------|
| ID: H-06 | Usuario: Analista de Sistemas |
| Nombre de la Historia: Gastronomía y restaurantes | |
| Prioridad: Media | Estimación: 3 |
| Riesgo: Bajo | |
| Responsable: Jefferson Andrade | |
| Descripción: | |
| <p>El sistema debe tener una sección para administrar la información de gastronomía del cantón y los restaurantes reconocidos en la Jefatura de Turismo de Cotacachi.</p> <ul style="list-style-type: none">• Se permita ingresar un nuevo restaurante, con los datos siguientes: Nombre del restaurante. Dirección en inglés y español. Horario de atención. Propietario. Correo electrónico. Una imagen de representación del restaurante.• Se debe mostrar un listado de todos los restaurantes registrados.• Se debe permitir eliminar y editar un restaurante.• Se debe crear una ruta de servicio web donde se listen los restaurantes solamente con la información en español• Se debe crear una ruta de servicio web donde se listen los restaurantes solamente con la información en inglés. | |
| Pruebas de aceptación: | |
| <ul style="list-style-type: none">• La información de los restaurantes debe listarse y mostrar un acceso para visualizar la información completa.• Validar los campos vacíos• Validar los campos numéricos• Validar los formatos jpg y png para la fotografía.• Mostrar mensajes de alerta en caso de que existan errores en el ingreso.• Las rutas de información deben ser en formato JSON.• En el horario se debe poder escoger la hora de inicio y fin de la jornada de servicio. | |

Fuente: Propia

TABLA 2.13 HISTORIA DE USUARIO H-07 – Ofertas turísticas

| Historia de Usuario | |
|---|--------------------------------------|
| ID: H-07 | Usuario: Analista de Sistemas |
| Nombre de la Historia: Ofertas turísticas | |
| Prioridad: Media | Estimación: 3 |
| Riesgo: Bajo | |
| Responsible: Jefferson Andrade | |
| Descripción: | |
| <p>El sistema debe tener una sección para administrar la información de las ofertas turísticas donde:</p> <ul style="list-style-type: none">• Se permita ingresar una nueva oferta turística, con los datos siguientes: Nombre de la oferta turística en español. Un nombre representativo en inglés. Dirección en ingles y español. Descripción de la oferta turística en inglés y español. La zona a la que pertenece el emprendimiento, puede ser de la zona andina, zona rural o de la zona sub tropical de Intag. Una imagen de representación del emprendimiento. Estado del evento activo como predeterminado.• Se debe mostrar un listado de todos los emprendimientos registrados.• Se debe permitir eliminar y editar un nuevo emprendimiento.• Se debe crear una ruta de servicio web donde se listen los emprendimientos turísticos solamente con la información en español.• Se debe crear una ruta de servicio web donde se listen los emprendimientos turísticos solamente con la información en inglés. | |
| Pruebas de aceptación: | |
| <ul style="list-style-type: none">• La información de los emprendimientos turísticos debe listarse y tener un acceso para visualizar la información completa.• Validar los campos vacíos.• Validar los campos numéricos.• Validar los formatos jpg y png para la fotografía.• Mostrar mensajes de alerta en caso de que existan errores en el ingreso.• Las rutas de información deben ser en formato JSON.• Al momento de ingresar la zona a la que pertenece se mostrará un campo donde el usuario pueda seleccionar una de las tres zonas. | |

Fuente: Propia

TABLA 2.14 HISTORIA DE USUARIO H-08 – Establecimientos y comercio

| Historia de Usuario | |
|---|--------------------------------------|
| ID: H-08 | Usuario: Analista de Sistemas |
| Nombre de la Historia: Establecimientos y comercio | |
| Prioridad: Media | Estimación: 3 |
| Riesgo: Bajo | |
| Responsable: Jefferson Andrade | |
| Descripción: | |
| <p>El sistema debe tener una sección para administrar la información de los principales establecimientos de comercio, transporte, comercio y hospedaje donde:</p> <ul style="list-style-type: none">• Se permita ingresar un nuevo establecimiento, con los datos siguientes: Nombre del establecimiento en español. Un nombre representativo en inglés. Propietario o titular del establecimiento. Teléfono de contacto. Dirección en ingres y español. El tipo de establecimiento al que se clasifica. Una imagen de representación del emprendimiento. Estado del evento activo como predeterminado.• Se debe mostrar un listado de todos los establecimientos registrados.• Se debe permitir eliminar y editar un nuevo establecimiento.• Se debe crear una ruta de servicio web donde se listen los establecimientos turísticos solamente con la información en español.• Se debe crear una ruta de servicio web donde se listen los establecimientos turísticos solamente con la información en inglés. | |
| Pruebas de aceptación: | |
| <ul style="list-style-type: none">• La información de los establecimientos turísticos debe listarse y tener un acceso para visualizar la información completa.• Validar los campos vacíos.• Validar los campos numéricos.• Validar los formatos jpg y png para la fotografía.• Mostrar mensajes de alerta en caso de que existan errores en el ingreso.• Las rutas de información deben ser en formato JSON.• Al momento de ingresar un nuevo establecimiento el usuario podrá seleccionar el tipo de establecimiento al que pertenece.• Al momento de modificar o eliminar un establecimiento se deben mostrar mensajes de confirmación o de alerta si la actividad fue realizada con éxito o fue rechazada. | |

Fuente: Propia

TABLA 2.15 HISTORIA DE USUARIO H-09 – Registro de clientes

| Historia de Usuario | |
|--|--------------------------------------|
| ID: H-09 | Usuario: Analista de Sistemas |
| Nombre de la Historia: Registro de clientes | |
| Prioridad: Alta | Estimación: 4 |
| Riesgo: Alto | |
| Responsable: Jefferson Andrade | |
| Descripción: | |
| El sistema debe tener una ventana para constatar la información de los clientes que se descargan la aplicación, donde: | |
| <ul style="list-style-type: none">• Se permita visualizar un listado de los usuarios que registran sus datos en la aplicación móvil.• Se debe crear una ruta de servicio web donde permita ingresar la información de usuario con los siguientes campos:<ul style="list-style-type: none">Nombre del cliente.Apellido.País de Origen.Ciudad de Origen.La fecha de registro. | |
| Se iniciará el desarrollo de la aplicación móvil para registrar clientes, donde: | |
| <ul style="list-style-type: none">• Al iniciar la aplicación se debe mostrar una pantalla de registro del cliente con los campos necesarios.• La información se guardará tanto en la aplicación como en el sistema web. | |
| Pruebas de aceptación: | |
| <ul style="list-style-type: none">• La aplicación móvil se debe iniciar en las tres plataformas:<ul style="list-style-type: none">AndroidiOSWindows UWP• Validar los campos vacíos.• Cuando se ingrese el país del cliente, la aplicación debe tener una lista donde el usuario pueda escoger su país de origen.• El registro se realizará únicamente una vez para un dispositivo, la información se mantendrá guardada como una sesión en la aplicación.• La fecha de registro se guardará automáticamente por interno.• La aplicación deberá ser capaz de adaptarse y traducirse a inglés o español según la configuración del dispositivo. | |

Fuente: Propia

TABLA 2.16 HISTORIA DE USUARIO H-10 – Aplicación móvil turística

| Historia de Usuar | |
|---|--------------------------------------|
| ID: H-10 | Usuario: Analista de Sistemas |
| Nombre de la Historia: Aplicación móvil turística | |
| Prioridad: Alta | Estimación: 4 |
| Riesgo: Alto | |
| Responsable: Jefferson Andrade | |
| Descripción: | |
| Se debe continuar el desarrollo de la aplicación con la presentación de la información turística, donde: | |
| <ul style="list-style-type: none">• Se pueda visualizar un menú de opciones para: Atractivos turísticos. Manifestaciones culturales. Oferta turística. Gastronomía y restaurantes. Establecimientos.• Debe tener un submenú para los atractivos naturales, patrimonios y de Intag.• Cada atractivo tendrá su espacio donde se visualice toda su información.• Se debe mostrar un listado de las manifestaciones culturales registradas.• Se debe mostrar un listado de las ofertas turísticas registradas, organizadas por un submenú por zona.• Se debe poder visualizar la gastronomía y el listado de restaurantes correctamente registrados.• Se debe tener una interfaz organizada de los establecimientos registrados por transportes, entidades financieras, tiendas artesanales y de hospedaje. | |
| La aplicación móvil debe tener un menú lateral donde: | |
| <ul style="list-style-type: none">• Se permita visualizar la información general del cantón Cotacachi.• Se permita acceder directamente al listado de eventos activos registrados en el sistema web.• Se debe mostrar una opción de “IR” para el desarrollo de georreferenciación posterior. | |
| Pruebas de aceptación: | |
| <ul style="list-style-type: none">• La aplicación móvil se debe iniciar en las tres plataformas: Android iOS Windows UWP• La aplicación debe ser capaz de presentar la información en inglés o español según la configuración de idioma del dispositivo. | |

Fuente: Propia

TABLA 2.17 HISTORIA DE USUARIO H-11 – Georreferenciación

| Historia de Usuario | |
|---|--------------------------------------|
| ID: H-11 | Usuario: Analista de Sistemas |
| Nombre de la Historia: Georreferenciación. | |
| Prioridad: Alta | Estimación: 4 |
| Riesgo: Alto | |
| Responsable: Jefferson Andrade | |
| Descripción: | |
| La aplicación móvil debe tener incorporado un sistema de georreferenciación donde: | |
| <ul style="list-style-type: none">• Se debe incluir una opción donde se muestre la ubicación de los atractivos en un mapa.• Se debe mostrar también la ubicación del dispositivo con relación a la ubicación del atractivo.• En la opción de “IR” se completará con la implementación de un mapa georreferenciado ubicando la posición del dispositivo y localización del cantón Cotacachi.• Se implementará una opción de “IR” a cada evento registrado, mostrando un mapa con la ubicación del dispositivo y el lugar del evento. | |
| Pruebas de aceptación: | |
| <ul style="list-style-type: none">• La aplicación móvil se debe iniciar en las tres plataformas Android iOS Windows UWP• La aplicación debe ser capaz de presentar la información en inglés o español según la configuración de idioma del dispositivo.• La georreferenciación debe ser validada realizando las pruebas necesarias en diversos lugares reflejando la dirección del dispositivo en el mapa.• Todas las direcciones de los eventos y atractivos deben consumirse de las coordenadas ingresadas (latitud y longitud) en el sistema web.• Validar la conexión a internet. | |

Fuente: Propia

TABLA 2.18 HISTORIA DE USUARIO H-12 – Clima

| Historia de Usuario | |
|--|--------------------------------------|
| ID: H-12 | Usuario: Analista de Sistemas |
| Nombre de la Historia: Clima | |
| Prioridad: Media | Estimación: 4 |
| Riesgo: Alto | |
| Responsable: Jefferson Andrade | |
| Descripción: | |
| La aplicación móvil debe tener una pantalla para mostrar la información siguiente: | |
| <ul style="list-style-type: none">• Debe presentar una descripción del estado climático en tiempo real.• Informar sobre el clima en grados centígrados.• Temperatura Máxima y temperatura mínima.• Inflación adicional como la velocidad del viento y la humedad. | |
| Pruebas de aceptación: | |
| <ul style="list-style-type: none">• La aplicación móvil se debe iniciar en las tres plataformas: Android iOS Windows UWP• La aplicación debe ser capaz de presentar la información en inglés o español según la configuración de idioma del dispositivo. | |

Fuente: Propia

2.2.2. Aprobar, estimar y asignar las Historias de Usuario

El desarrollo metodológico con SCRUM consta también de la definición de Sprints necesarios para cumplir el desarrollo del proyecto, dichos Sprints son lapsos de tiempo donde se debe presentar avances del sistema.

En la Tabla 2.19 se puede apreciar que el desarrollo del sistema se ha dividido en 4 Sprints de semanas correspondientes a 4 meses, se agrupan las historias de usuario a cada sprint en función a la prioridad requerida.

TABLA 2.19 APROBACIÓN, ESTIMACIÓN Y ASIGNACIÓN DE HISTORIAS

| ID | HISTORIA | SPRINT | ESTIMACIÓN | ENCARGADO |
|------|--------------------------------------|--------------|------------|-------------------|
| H-01 | Preparación del ambiente de trabajo. | 1 | Media | Jefferson Andrade |
| H-02 | Acceso al sistema web. | (08-04-2019) | Media | Jefferson Andrade |
| H-03 | turísticos. | (03-05-2019) | Alta | Jefferson Andrade |
| H-04 | Eventos turísticos. | | Alta | Jefferson Andrade |
| H-05 | Información cultural. | | Media | Jefferson Andrade |
| H-06 | Gastronomía y restaurantes. | 2 | Media | Jefferson Andrade |
| H-07 | Ofertas turísticas. | (06-05-2019) | Media | Jefferson Andrade |
| H-08 | Establecimientos y comercio. | (31-05-2019) | Media | Jefferson Andrade |
| H-09 | Registro de clientes. | | Alta | Jefferson Andrade |
| H-10 | Aplicación móvil turística. | 3 | Alta | Jefferson Andrade |
| | | (03-06-2019) | | |
| | | (29-06-2019) | | |
| H-11 | Georreferenciación. | 4 | Alta | Jefferson Andrade |
| H-12 | Clima | (01-07-2019) | Media | Jefferson Andrade |
| | | (26-07-2019) | | |

Fuente: Propia

2.2.3. Elaboración y estimación de tareas

Después de analizar, crear y asignar los Sprints con sus respectivas historias de usuario se procede a dividir cada historia en pequeñas tareas con un tiempo de ejecución.

Las Tablas siguientes presenta la elaboración de los Sprint necesarios.

TABLA 2.20 SPRINT 1

| SPRINT 1 | | | |
|---|-------------------------------------|---|--------------------|
| Fecha de inicio: | 08-04-2019 | Fecha fin: | 03-05-2019 |
| OBJETIVO: | | | |
| Habilitar un sistema web que permita gestionar la información de atractivos turísticos y la publicación de eventos. | | | |
| ID | HISTORIA | TAREA | ESTIMACIÓN (HORAS) |
| H-01 | Preparación del ambiente de trabajo | Instalación del servidor web Apache y manejador de dependencias Composer. | 1 |
| | | Instalación de la base de datos MySQL 5.0.12-dev y gestor Navicat. | 1 |
| | | Instalación del framework Laravel v5.5 y gestor de texto Sublime Text. | 1 |
| | | Instalación de Visual Studio y .NET framework. | 6 |

| | | | |
|------------------------|-----------------------|---|-----------|
| | | Descarga e instalación de Xamarin y derivados. | 2 |
| | | Diseñar y crear el diagrama y base de datos. | 5 |
| H-02 | Acceso al sistema web | Creación de un formulario para el login del sistema. | 5 |
| | | Creación de un formulario para registro de nuevos usuarios. | 4 |
| | | Diseñar una pantalla para listar los usuarios registrados. | 3 |
| | | Crear modales para editar y eliminar usuarios. | 3 |
| | | Validar a usuarios por medio del correo electrónico. | 6 |
| | | Validar el acceso por roles al sistema web. | 5 |
| H-03 | Atractivos turísticos | Diseño de un formulario para el registro de atractivos turísticos. | 3 |
| | | Creación de un menú para los tipos de atractivos. | 3 |
| | | Listado de los atractivos clasificados por su tipo. | 3 |
| | | Crear modales para editar y eliminar atractivos turísticos. | 3 |
| H-04 | Eventos turísticos | Diseño de un formulario para el registro de eventos turísticos. | 3 |
| | | Listado de los eventos registrados. | 2 |
| | | Crear modales para editar y eliminar eventos. | 3 |
| | | Crear rutas REST para consultar la información. | 3 |
| | | Validar todos los campos necesarios, mensajes de alerta y confirmación. | 8 |
| TOTAL DE HORAS: | | | 73 |

Fuente: Propia

TABLA 2.21 SPRINT 2

| SPRINT 2 | | | |
|---|----------------------|---|--------------------|
| Fecha de inicio: | 06-05-2019 | Fecha fin: | 31-05-2019 |
| OBJETIVO: | | | |
| Desarrollar métodos para que el usuario pueda gestionar la información de manifestaciones culturales, ofertas, restaurantes y establecimientos. | | | |
| ID | HISTORIA | TAREA | ESTIMACIÓN (HORAS) |
| H-05 | Información cultural | Diseño de un formulario para el registro de manifestaciones culturales. | 2 |
| | | Listado de todas las manifestaciones culturales. | 2 |
| | | Crear modales para editar y eliminar manifestaciones. | 3 |

| | | | | | | |
|-------------|---------------------------|---|--|-----|--|-----------|
| | | | Diseño de un formulario para el registro de restaurantes. | 3 | | |
| H-06 | Gastronomía restaurantes | y | Listado de los restaurantes registrados. | 3 | | |
| | | | Crear modales para editar y eliminar restaurantes. | 3 | | |
| | | | Diseño de un formulario para el registro de establecimientos turísticos. | 3 | | |
| H-07 | Ofertas turísticas | | Creación de un menú para los tipos de establecimientos. | 2 | | |
| | | | Listado de los establecimientos clasificados por su tipo. | 2 | | |
| | | | Crear modales para editar y eliminar establecimientos. | 3 | | |
| | | | Diseño de un formulario para el registro de ofertas turísticas. | 3 | | |
| H-08 | Establecimientos comercio | y | Creación de un menú para las ofertas turísticas por zona. | 2 | | |
| | | | Listado de las ofertas clasificados por su zona. | 2 | | |
| | | | Crear modales para editar y eliminar ofertas turísticas | 3 | | |
| | | | Validar todos los campos necesarios, mensajes de alerta y confirmación. | 5,5 | | |
| H-09 | Registro de clientes | | Diseño de un formulario para listar los clientes registrados. | 2 | | |
| | | | Creación de la ruta REST para el registro de clientes, listado de restaurantes y establecimientos. | 2 | | |
| | | | Creación de la aplicación móvil multiplataforma en Xamarin. | 0,5 | | |
| | | | Diseño de un formulario bilingüe para el registro de clientes. | 5 | | |
| | | | Validar todos los campos del formulario de registro de clientes | 3 | | |
| | | | TOTAL DE HORAS: | | | 54 |

Fuente: Propia

TABLA 2.22 SPRINT 3

| SPRINT 3 | | | |
|--|----------------------------|--|--------------------|
| Fecha de inicio: | 03-06-2019 | Fecha fin: | 28-06-2019 |
| OBJETIVO: | | | |
| Diseñar la aplicación móvil multiplataforma para que el cliente visualice toda la información que se registra en el sistema web. | | | |
| ID | HISTORIA | TAREA | ESTIMACIÓN (HORAS) |
| H-10 | Aplicación móvil turística | Diseño de un menú principal para acceder a la información turística. | 5 |
| | | En la primera opción colocar los atractivos turísticos naturales, patrimonios y de Intag | 4 |

H-10 Aplicación móvil turística

| | |
|--|-----------|
| Listar y visualizar el detalle de los atractivos turísticos. | 5 |
| En la segunda opción listar las manifestaciones culturales. | 4 |
| Diseñar una pantalla para visualizar el detalle de los atractivos turísticos. | 5 |
| En la tercera opción mostrar el listado de las ofertas culturales. | 4 |
| Diseñar un menú para organizar las ofertas por zonas y una pantalla para ver el detalle de cada uno | 4 |
| En la cuarta opción mostrar la gastronomía y los restaurantes registrados. | 4 |
| Diseñar una pantalla para visualizar el detalle de los restaurantes registrados. | 3 |
| En la quinta opción mostrar los establecimientos registrados. | 3 |
| Diseño de un menú para organizar los establecimientos registrados. | 3 |
| Diseñar una pantalla para visualizar el detalle de los establecimientos. | 3 |
| Diseño de un menú hamburguesa para presentar información general de la aplicación. | 5 |
| En el primer Item diseñar una vista para presentar información del cantón Cotacachi. | 3 |
| En el segundo Item diseñar una vista para listar los eventos programados activos. | 5 |
| Realizar validaciones. | 3 |
| Implementar un método para cambiar de direcciones de rutas REST para presentar la información en inglés o español. | 6 |
| Implementar archivos de recurso para parametrizar los textos y labels para mostrarlos en inglés o español. | 4 |
| Realización de pruebas y ajustes de pantallas para los dispositivos móviles. | 6 |
| TOTAL DE HORAS: | 79 |

Fuente: Propia

TABLA 2.23 SPRINT 4

| SPRINT 4 | | | |
|---|--------------------|---|--------------------|
| Fecha de inicio: | 01-07-2019 | Fecha fin: | 19-07-2019 |
| OBJETIVO: | | | |
| Georreferenciar los atractivos turísticos, eventos que se registren en la aplicación móvil, además el cliente podrá consultar el clima del cantón en tiempo real. | | | |
| ID | HISTORIA | TAREA | ESTIMACIÓN (HORAS) |
| H-11 | Georreferenciación | Incluir las librerías y configuraciones necesarias. | 3 |
| | | En el tercer ítem del menú diseñar una pantalla con un mapa georreferenciado. | 3 |
| | | Implementar un método para obtener la localización del dispositivo. | 5 |
| | | Implementar un método para obtener la localización del cantón. | 4 |
| | | Diseñar una opción de localización en la sección de eventos. | 3 |
| | | Obtener la localización de los eventos y graficarlos en el mapa. | 4 |
| | | En los atractivos se debe incluir una opción de ir en cada uno. | 3 |
| | | Localizar la ubicación del atractivo seleccionado y mostrarlo en el mapa. | 4 |
| | | Validar la conexión a internet y ubicación del dispositivo. | 3 |
| | | H-12 | Clima |
| Consumir un servicio web libre para tomar la descripción del estado climático. | 4 | | |
| Ajustar los resultados para reflejarlos en inglés y español. | 6 | | |
| Realizar la compilación del desarrollo en las tres plataformas, Android, iOS y Windows. | 3 | | |
| Configurar la escalabilidad de pantalla de la aplicación. | 2 | | |
| Realizar las pruebas respectivas | 4 | | |
| Capacitación del software | 4 | | |
| TOTAL DE HORAS: | | | 58 |

Fuente: Propia

2.3. Fase 3: Desarrollo

En esta fase se hace presente el diseño de la aplicación, cumpliendo con los requerimientos descritos en cada una de las historias presentadas con anterioridad.

2.3.1. Arquitectura de la aplicación

Para realizar un software de calidad se ha decidido implementar la arquitectura por capas y por medio del patrón de diseño MVVM (Modelo Vista Vista-Modelo).

El desarrollo de las interfaces de usuario para las aplicaciones no es sencillo, ya que en muchos de los casos son una mezcla de objetos visuales, conectividad multitarea, seguridades y validaciones. Para resolver esta problemática se usa el patrón de diseño MVVM que separa adecuadamente y enfrenta a los problemas que comúnmente se atraviesan al momento de programar.

El patrón MVVM se basa en un mecanismo de enlace de datos que simplifica el desarrollo de la capa de separación de la vista, el patrón puede también usar el lenguaje de enmarcado y crear enlaces de datos con el Vista-Modelo, que está escrito en lenguaje C#. Esto permite que se desarrolle las Vistas sin tener que preocuparse por un controlador que condicione su trabajo. En la figura 2.12 se presenta una representación gráfica de la comunicación de las capas.

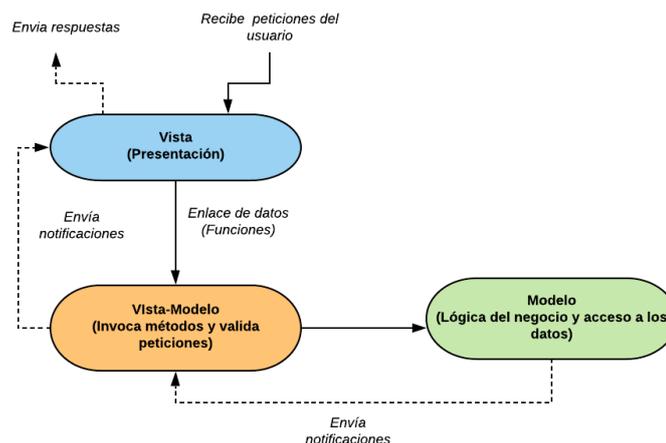


Fig. 2.9. Patrón MVVM

Fuente: Propia

2.3.2. Descripción de los Sprints

SPRINT 1

- **H-01 – Preparación del ambiente de trabajo**

Después de la instalación correcta de las herramientas de desarrollo, se diseñó los casos de uso necesarios para el sistema Web, como también para la aplicación móvil. Ver anexo 2.

Seguidamente se diseñó el diagrama de base de datos para el sistema de información turística.

| turismo role_user | |
|-------------------|--------------------|
| id | : int(10) unsigned |
| # user_id | : int(10) unsigned |
| # role_id | : int(10) unsigned |
| created_at | : timestamp |
| updated_at | : timestamp |

| turismo clientes | |
|------------------|--------------------|
| id_cliente | : int(10) unsigned |
| nombre | : varchar(30) |
| apellido | : varchar(30) |
| ciudad | : varchar(30) |
| pais | : varchar(30) |
| created_at | : timestamp |
| updated_at | : timestamp |

| turismo migrations | |
|--------------------|--------------------|
| id | : int(10) unsigned |
| migration | : varchar(191) |
| # batch | : int(11) |

| turismo user_evento | |
|---------------------|-----------|
| # user | : int(11) |
| # evento | : int(11) |

| turismo restaurantes | |
|----------------------|--------------------|
| id_restaurante | : int(10) unsigned |
| nombre | : varchar(30) |
| propietario | : varchar(30) |
| telefono | : varchar(10) |
| direccion | : varchar(50) |
| address | : varchar(50) |
| atencion | : varchar(30) |
| atention | : varchar(30) |
| inicio | : varchar(8) |
| fin | : varchar(8) |
| foto | : varchar(100) |
| created_at | : timestamp |
| updated_at | : timestamp |

| turismo users | |
|----------------|--------------------|
| id | : int(10) unsigned |
| name | : varchar(191) |
| email | : varchar(191) |
| password | : varchar(191) |
| avatar | : varchar(191) |
| code | : varchar(191) |
| # active | : int(11) |
| remember_token | : varchar(100) |
| created_at | : timestamp |
| updated_at | : timestamp |

| turismo eventos | |
|-----------------|--------------------|
| id_evento | : int(10) unsigned |
| nombre | : varchar(30) |
| name | : varchar(30) |
| lugar | : varchar(30) |
| place | : varchar(30) |
| fecha | : date |
| hora | : varchar(20) |
| descripcion | : varchar(500) |
| description | : varchar(500) |
| latitud | : varchar(30) |
| longitud | : varchar(30) |
| estado | : varchar(1) |
| foto | : varchar(100) |
| created_at | : timestamp |
| updated_at | : timestamp |

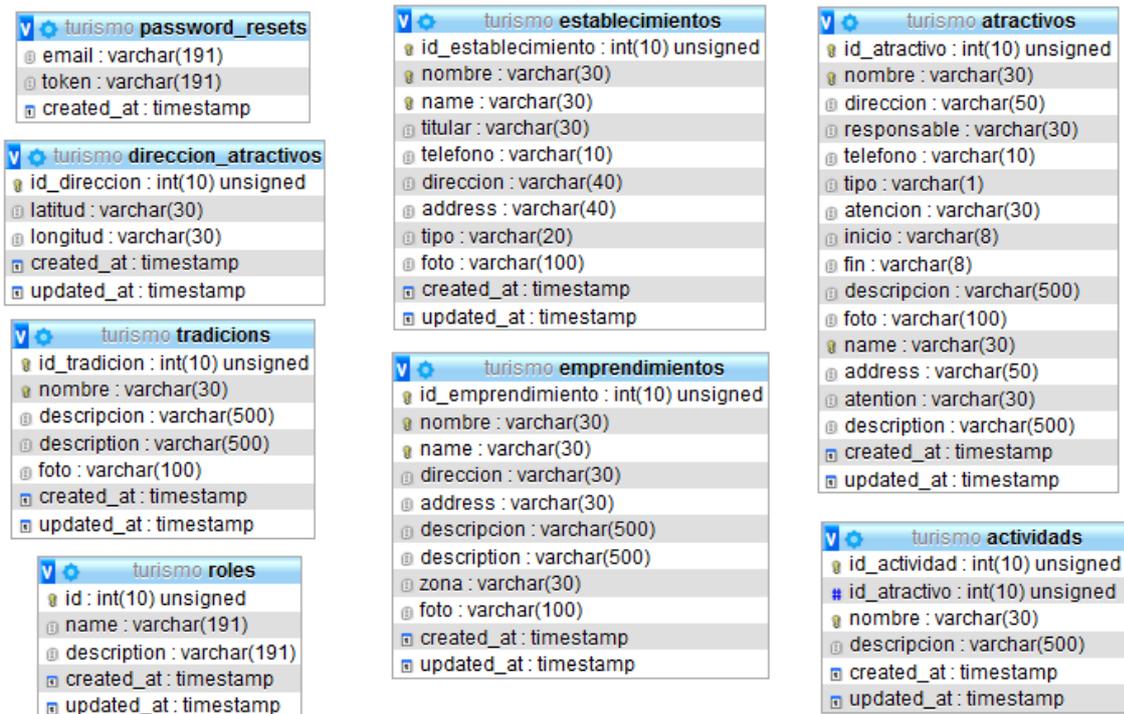


Fig. 2.10. Diagrama de base de datos

Fuente:

- **H-02 – Acceso al sistema web Vive Cotacachi**

En el desarrollo del proyecto se diseñó la interfaz de ingreso al sistema web. En la figura se muestra la captura del login del sistema. Ver anexo 2.a.

La interfaz de login muestra los siguientes elementos:

- Título: Login
- Campo E-Mail Address: admin@gmail.com
- Campo Password: oculto con puntos
- Checkbox: Remember Me (desactivado)
- Botón: Login
- Enlace: Forgot Your Password?

Fig. 2.11. Login del sistema web

Fuente: Propia

Para la administración del sistema se establecieron dos usuarios, el super usuario que es capaz de gestionar absolutamente todo el sistema, el administrador que gestiona la información turística y el usuario que simplemente visualiza la información registrada. La Fig. 2.12 muestra el registro de un nuevo usuario ingresando el nombre completo, su correo y rol.

Fig. 2.12. Vista para crear un nuevo usuario

Fuente: Propia

En la Fig. 2.13 se presenta como el sistema envía un correo para validar al usuario.



Fig. 2.13. Validación de usuario

Fuente: Propia

En la Fig. 2.14 se completa el registro del usuario ingresando y confirmando la contraseña.

Fig. 2.14. Registro de contraseña del nuevo usuario

Fuente: Propia

En la Fig. 2.15 se muestra la lista de usuarios registrados.

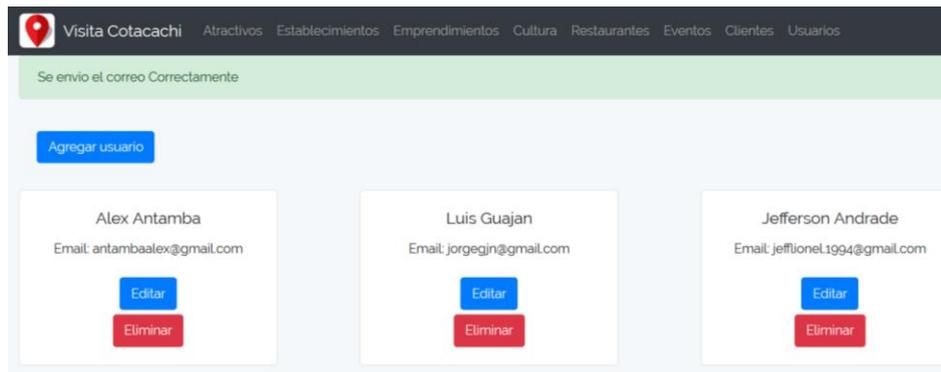


Fig. 2.15. Lista de usuarios
Fuente: Propia

- **H-03 – Atractivos turísticos**

Para administrar la plataforma es necesario acceder como un usuario de rol administrador. Caso contrario el sistema negará cualquier actividad. En Fig. 2.16 se muestra el menú principal del sistema de gestión turística. Ver anexo 2.c.

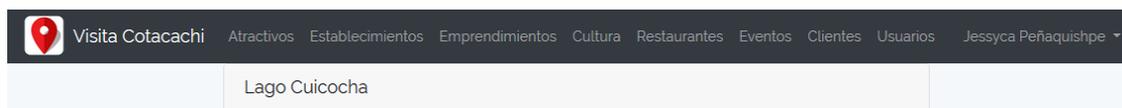


Fig. 2.16. Menú sistema web Visita Cotacachi
Fuente: Propia

Se diseñó un submenú para que los usuarios visualicen la lista de atractivos por tipo, realicen el registro de un nuevo atractivo y sus opciones de editar y eliminar.



Fig. 2.17. Tipos de atractivos
Fuente: Propia



Fig. 2.18. Lista de atractivos
Fuente: Propia

Formulario de registro de atractivos con los siguientes campos:

- Nombre (input)
- Dirección (input)
- Atención (input)
- Descripción (input)
- Responsable (input)
- Teléfono (09xxxxxxxx) (input)
- Desde (dropdown)
- Hasta (dropdown)
- Imagen (Examinar... No se ha seleccionado ningún archivo)
- Botones: Guardar (azul), Cancelar (rojo)

Fig. 2.19. Formulario de registro de atractivos
Fuente: Propia

- **H-04 – Eventos turísticos**

El usuario puede ver la lista de eventos, editarlos, eliminarlos y realizar el registro de un nuevo evento. Al editar un evento puede escoger la opción de desactivar eventos pasados.

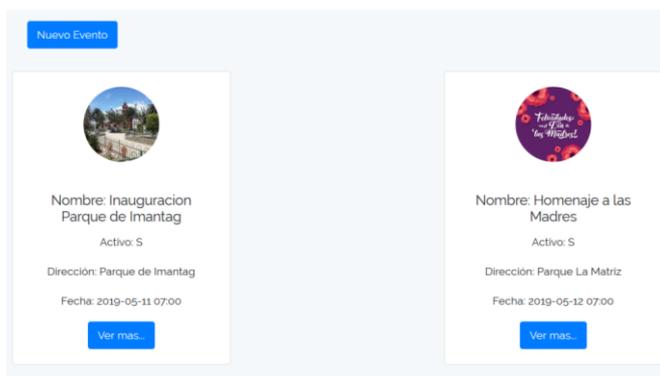


Fig. 2.20. Lista de eventos
Fuente: Propia

Form for event registration with the following fields:

- Nombre (Name)
- Lugar (Place)
- Latitud (Latitude)
- Longitud (Longitude)
- Descripción (Description)
- Fecha (Date) - format: dd / mm / aaaa
- Hora (Time)
- Imagen (Image) - Examinar... No se ha seleccionado ningún archivo.

Buttons: Guardar (Save), Cancelar (Cancel)

Fig. 2.21. Registro de eventos

Fuente: Propia

SPRINT 2

- H-05 – Información cultural**

Los usuarios como se puede observar en la figura 2.22 pueden ver el listado de manifestaciones culturales, además las funciones eliminación y edición.

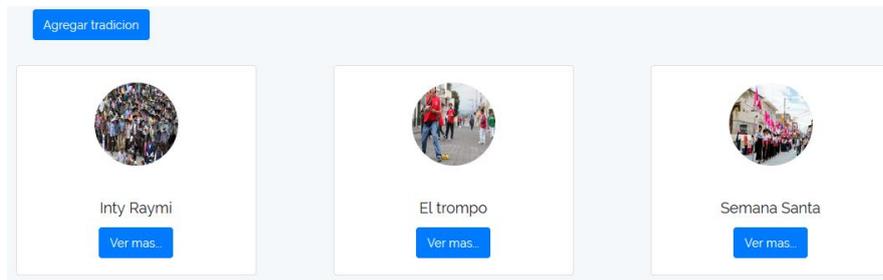


Fig. 2.22. Lista de manifestaciones culturales

Fuente: Propia

Form for cultural manifestation registration with the following fields:

- Nombre
- Descripción
- Descripción
- Imagen (Image) - Examinar... No se ha seleccionado ningún archivo.

Buttons: Guardar (Save), Cancelar (Cancel)

Fig. 2.23. Registro de manifestaciones culturales

Fuente: Propia

- **H-06 – Gastronomía y restaurantes**

Los restaurantes tienen su propia lista, su formulario de registro en inglés y español, así como también las actividades de eliminar y editar un registro.

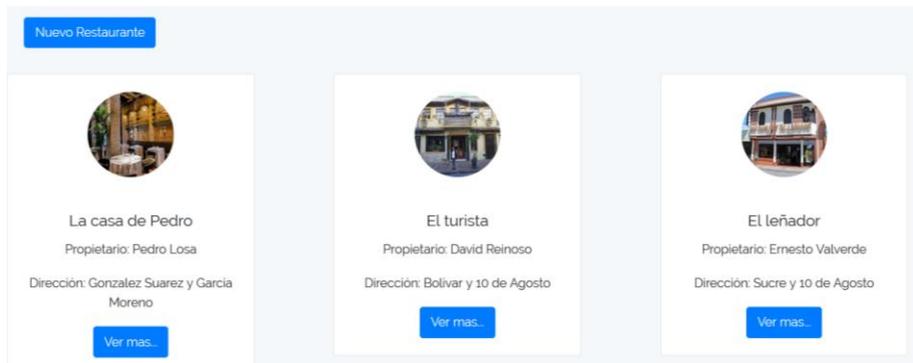


Fig. 2.24. Lista de restaurantes

Fuente: Propia

Fig. 2.25. Registro de restaurantes

Fuente: Propia

- **H-07 - Ofertas turísticas**

Se diseñó el ingreso de ofertas turísticas, el listado y los cuadros de diálogo de eliminación y edición del registro.

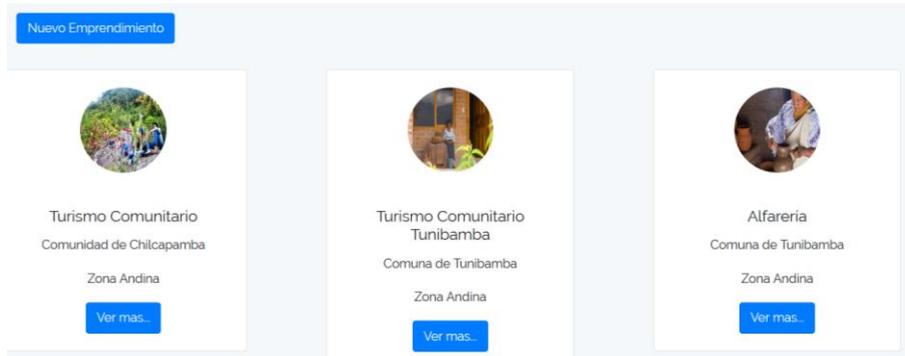


Fig. 2.26. Lista de ofertas turísticas

Fuente: Propia

- **H-08 – Establecimientos y comercio**

En Fig. 2.27. muestra la lista de establecimientos turísticos.

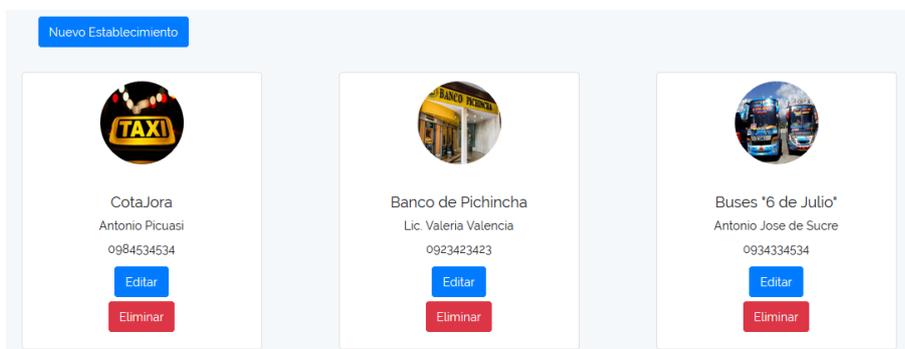


Fig. 2.27. Lista de establecimientos

Fuente: Propia

The screenshot shows a registration form for an establishment. It consists of two columns of input fields, a dropdown menu, and a file upload section.

| Field Label | Placeholder Text |
|-----------------------|---|
| Nombre | Ingresa el nombre del establecimiento |
| Name | Add the institution name |
| Titular | Ingresa el nombre de su titular |
| Telefono (09xxxxxxxx) | Ingresa el telefono |
| Direccion | Ingresa la direccion del establecimiento |
| Address | Add the address |
| Tipo | -- |
| Imagen | Examinar... No se ha seleccionado ningún archivo. |

Buttons: Guardar (blue), Cancelar (red)

Fig. 2.28. Registro de establecimientos

Fuente: Propia

- **H-09 - Registro de clientes**

El registro de los clientes en la aplicación móvil se guarda también en el sistema web de Vive Cotacachi, en la Fig. 2.29 se visualiza la lista de clientes que hicieron su registro en la aplicación móvil. Ver Anexo E.

| | | |
|--|--|---|
| Jefferson Andrade Mexico-Cotacachi 2019-06-04 20:29:03-Cotacachi | Camila Andrade Ecuador-Cotacachi 2019-06-05 15:45:47-Cotacachi | Jefferson Pijuango Belize-Cotacachi 2019-06-05 17:39:38-Cotacachi |
| Andres Tipa Mexico-Mexico 2019-06-06 13:45:49-Mexico | Alex Antamba Ecuador-Imbabura 2019-06-07 08:23:16-Imbabura | Jessyca Andrade Ecuador-Cotacachi 2019-06-10 19:34:18-Cotacachi |
| Danilo Aguirre China-HonKong 2019-06-11 22:29:19-HonKong | Anderson Aguirre Ecuador-Quito 2019-06-26 15:24:15-Quito | Mauricio Aguirre Ecuador-Cotacachi 2019-07-25 23:12:26-Cotacachi |

Fig. 2.29. Lista de clientes

Fuente: Propia

La Fig. muestra el formulario de registro en la aplicación móvil.

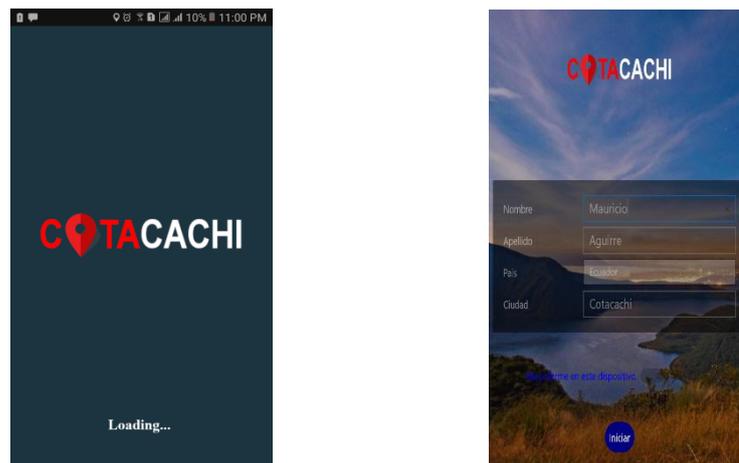


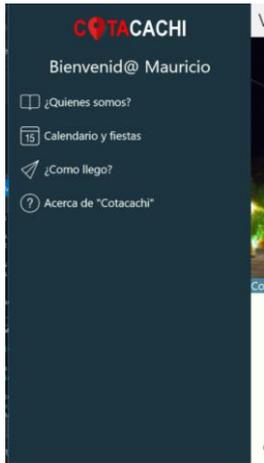
Fig. 2.30. Registro de clientes aplicación móvil

Fuente: Propia

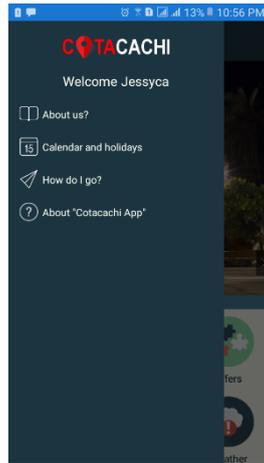
SPRINT 3

- **H-10 – Aplicación móvil turística**

Se diseñó la aplicación móvil para presentar al cliente final toda la información registrada en el sistema web, tomando en cuenta los requerimientos puntuados anteriormente. Ver anexo 2.f.



Pantalla Windows



Pantalla Android



Pantalla iOS

Fig. 2.31. Menú hamburgués

Fuente: Propia



Pantalla Windows



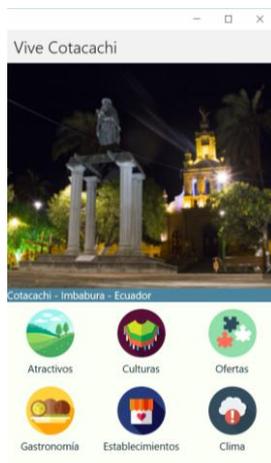
Pantalla Android



Pantalla iOS

Fig. 2.32. Lista de eventos aplicación móvil

Fuente: Propia



Pantalla Windows



Pantalla Android



Pantalla iOS

Fig. 2.33. Diseño del menú principal aplicación móvil

Fuente: Propia



Pantalla Windows



Pantalla Android



Pantalla iOS

Fig. 2.34. Tipos de atractivos aplicación móvil

Fuente: Propia



Pantalla Windows



Pantalla Android



Pantalla iOS

Fig. 2.35. Detalle de un atractivo aplicación móvil

Fuente: Propia



Pantalla Windows



Pantalla Android



Pantalla iOS

Fig. 2.36. Lista de gastronomía aplicación móvil

Fuente: Propia



Pantalla Windows



Pantalla Android



Pantalla iOS

Fig. 2.37. Lista de manifestaciones aplicación móvil

Fuente: Propia



Pantalla Windows



Pantalla Android



Pantalla iOS

Fig. 2.38. Lista de restaurantes aplicación móvil

Fuente: Propia



Pantalla Windows



Pantalla Android



Pantalla iOS

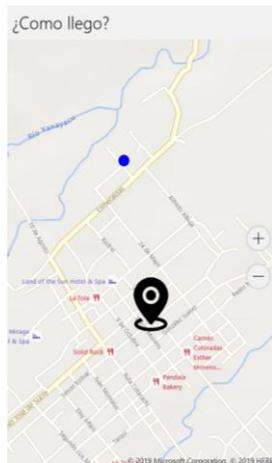
Fig. 2.39. Establecimientos aplicación móvil

Fuente: Propia

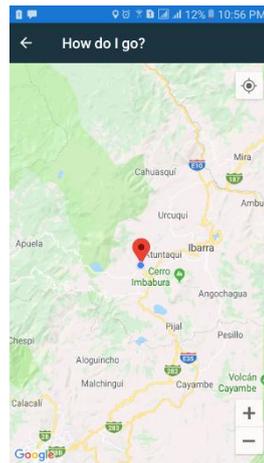
SPRINT 4

- **H-11 – Georreferenciación**

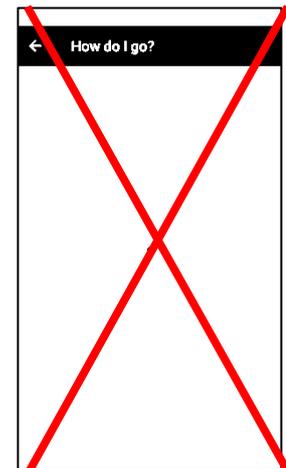
Se implemento la vista de un mapa georreferenciado para el usuario cliente se guie sobre la ubicación del cantón con respecto a su ubicación, la dirección de los eventos programados y la localización de cada uno de los atractivos turísticos. Cabe mencionar que al no tener una licencia de desarrollador de iOS el mapa georreferenciado no se visualiza, por ese motivo no se logró obtener el resultado deseado.



Pantalla Windows



Pantalla Android



Pantalla iOS

Fig. 2.40. Georreferenciación aplicación móvil

Fuente: Propia

- **H-12 – Clima**

Se diseño una vista que informa al cliente sobre el estado actual del clima.



Pantalla Windows



Pantalla Android



Pantalla iOS

Fig. 2.41. Consulta del clima aplicación móvil

Fuente: Propia

CAPÍTULO 3

3. RESULTADOS

Al haber finalizado el estudio comparativo entre los prototipos desarrollados en Xamarin y Delphi se obtuvo ventajas y desventajas dentro de desarrollo y ejecución de las apps en un entorno de dispositivo móvil inteligente, como se define en la Tabla 3.1.

TABLA 3.1 VENTAJAS Y DESVENTAJAS ENTRE LAS PLATAFORMAS XAMARIN Y DELPHI

| | VENTAJAS | DESVENTAJAS |
|----------------|---|--|
| Xamarin | <ul style="list-style-type: none">- Interfaz agradable de Visual Studio- Tiene una fácil y rápida implementación en Android y Windows.- Produce paquetes APK con tamaños aceptables.- Es una plataforma gratuita.- Proporciona facilidades para parametrizar y traducir texto a diferentes idiomas. | <ul style="list-style-type: none">- Comparte el código en un 90%- Su interfaz de diseño de aplicaciones no es lo suficientemente flexible. |
| Delphi | <ul style="list-style-type: none">- Brinda rapidez para crear prototipos y ejecutar en dispositivos Android y Windows.- Fácil de manipular la interfaz de diseño de aplicaciones.- Comparte el código en un 95% | <ul style="list-style-type: none">- Es una plataforma privada (Con costo de Licencia)- Produce paquetes APK con tamaños altos en consideración a Xamarin. |

Fuente: Propia

3.1. Resultados de la comparación del consumo de recurso al ejecutar los prototipos diseñados.

3.1.1. Memoria

En la figura 3.1 se muestra los megabytes que consume cada aplicación en cada una de las iteraciones.

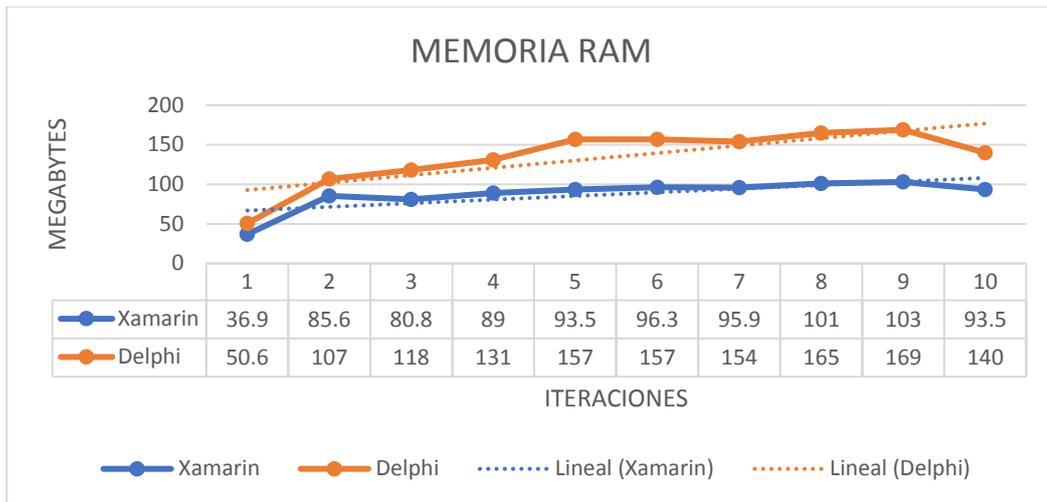


Fig. 3.1. Iterador RAM

Fuente: Propia

De acuerdo a la figura 3.1, se observa la franja azul correspondiente a la ejecución del prototipo desarrollado en Xamarin, donde las dos primeras iteraciones del uso de memoria RAM está en un rango de 36,9 Mb a 85,6 Mb, la tercera iteración disminuye 5Mb de consumo de memoria debido que únicamente muestra un submenú de atractivos. La cuarta iteración aumenta su consumo a 89Mb por la consulta realizada a servicios REST, es decir que las consultas a servicios remotos requieren mayor memoria, sin embarco a partir de la quinta iteración aumenta su consumo hasta 103Mb que posteriormente en la última iteración regresa su consumo a 93,5Mb demostrando un resultado favorable.

Por su parte el prototipo diseñado en la plataforma RAD Studio Delphi representado por la franja naranja, en su primera iteración consume ya 50,6 Mb de memoria, su consumo asciende al doble en la segunda iteración por la presentación de imágenes. A partir de la tercera iteración mantiene un rango de 107 Mb a 140 Mb que son poco favorables a diferencia de Xamarin.

De acuerdo al análisis realizado se concluye que, en la medición del consumo de memoria RAM, la más idónea es la aplicación desarrollada por Xamarin, por un 25% menos de consumo de recursos en memoria RAM.

3.1.2. Rendimiento

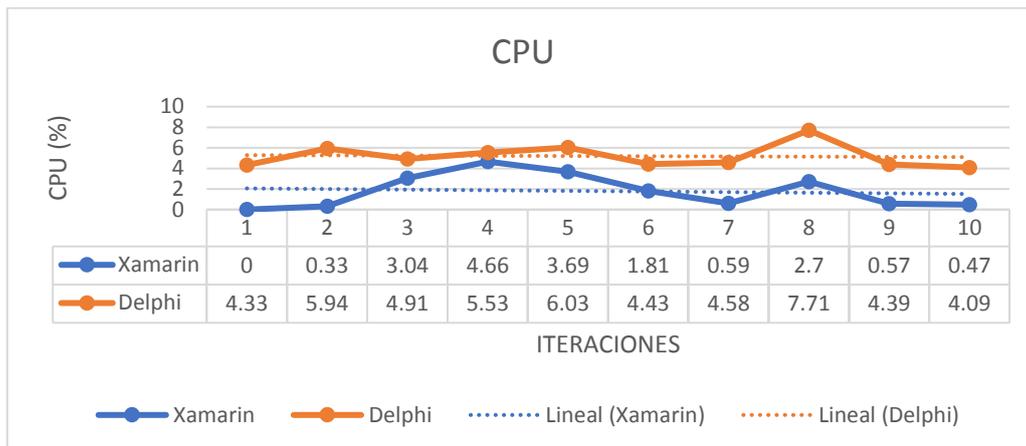


Fig. 3.2. Iterador CPU

Fuente: Propia

De acuerdo a la figura 3.2. se observa una franja azul que comprende el porcentaje de trabajo de la unidad central de procesamiento (CPU) del dispositivo, con respecto al prototipo diseñado en Xamarin, en las dos primeras iteraciones causa un impacto del 0,33%, pero en la tercera iteración aumenta al 3,04% del trabajo total en CPU, en las iteraciones siguientes va aumentando su trabajo por la sobrecarga de formularios y peticiones realizadas hacia las fuentes de datos remotos, durante toda la trayectoria no demuestra cambios severos en su consumo, tampoco que sobrepasen el consumo de Delphi.

Delphi por su parte está representado por la franja de color naranja, donde en las primeras tres iteraciones su porcentaje de trabajo se encuentra en un rango de 4,33% a 4,91%, en las iteraciones siguientes al igual que Xamarin aumenta su trabajo por la sobrecarga de peticiones realizadas y presentación de imágenes.

De acuerdo al análisis realizado se concluye que, el más idóneo en cuanto al menor trabajo producido en CPU en el prototipo realizado en Xamarin, ya que su porcentaje del trabajo en la unidad de procesamiento tiene un promedio del 1,7 % a diferencia de Delphi que consume un promedio de 5,19%.

3.1.3. Almacenamiento

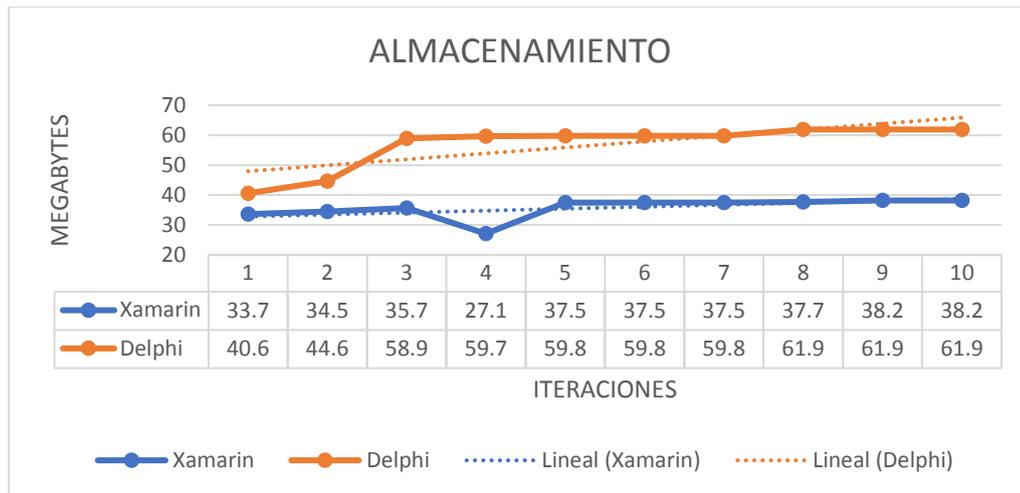


Fig. 3.3. Iterador almacenamiento

Fuente: Propia

De acuerdo a la figura 3.3, se puede observar la franja azul que representa a las iteraciones realizadas al prototipo de Xamarin, en las primeras iteraciones sus requerimientos de memoria son de 33,7 Mb a 35,7 que se obtiene en la tercera iteración, esto quiere decir que al sobrecargar dos formularios más a la aplicación su requerimiento de memoria es de más o menos 2 Mb y un Mb por formulario agregado. A partir de la cuarta se agregan los 6 formularios correspondientes y su tamaño en memoria se aumenta aproximadamente 3 Mb ya que todas las imágenes no se almacenan en la aplicación, sino que son consumidas desde nuestro servidor web y parte de ellas se almacenan en memoria cache.

Por su parte Delphi que está representado por la franja naranja en la primera iteración ya ocupa 7 Mb de espacio más que Xamarin. Hasta la tercera iteración su volumen aumenta a 59,7 Mb que es una diferencia de aproximadamente 20 Mb de memoria, que es ya el doble de memoria que ocupa el prototipo de Xamarin. A partir de la tercera hasta la décima su volumen se mantiene en un rango de 59,7 Mb a 61,9 Mb que no representa un cambio considerable.

Después de realizar el análisis respectivo del almacenamiento requerido por cada prototipo se puede obtener fácilmente que la herramienta más idónea para el desarrollo de la aplicación resulta ser Xamarin.

3.1.4. Evaluación final

A continuación, se muestra la proporción de recursos consumidos por cada prototipo diseñado.



Fig. 3.4. Evaluación de triangulación
Fuente: Propia

En la figura 3.4 se determina la relación entre memoria, rendimiento y almacenamiento al ejecutar aplicaciones en Xamarin y Delphi, donde Delphi que esta representada por la franja naranja engloba y consume más recursos que Xamarin en los tres iteradores evaluados.

Para concluir con la evaluación cabe mencionar que las aplicaciones con enfoque turístico poseen un mejor rendimiento si se desarrollan en el entorno de Xamarin ya que por medio de la figura 3.5 se demuestra que la aplicación desarrollada en Delphi engloba mayor cantidad de recursos en todos los aspectos. Mientras que Xamarin aproximadamente ocupa la mitad de los recursos totales.



Fig. 3.5. Evaluación de iteradores global
Fuente: Propia

CONCLUSIONES

Después de terminar con el estudio y desarrollo del software se puede concluir lo siguiente:

El marco conceptual ha permitido una mejor organización y orientación para el estudio de las herramientas y conocimiento profundo de las funciones que cumple cada plataforma.

Mediante el estudio de las dos herramientas para la construcción de aplicaciones multiplataforma se determinó que, Xamarin es la herramienta más favorable porque permite crear aplicaciones escalables a los dispositivos que disponen de bajos recursos como memoria y almacenamiento. Además de disponer de distintos recursos que permitieron crear una aplicación bilingüe y georreferenciada para cumplir en su totalidad con los requerimientos puntuados.

La aplicación de la metodología ágil SCRUM, permitió desarrollar el proyecto de una forma ordenada, donde todos los involucrados estaban informados de lo que sucedía con el proyecto, y como resultado se obtuvo un software que cumple con todos los requerimientos del usuario.

La utilización de herramientas de desarrollo de código abierto como Xamarin ha sido muy favorecedora en el desarrollo de la aplicación, hablando desde el factor económico ya que se creó una aplicación de calidad sin tener grandes gastos monetarios. Cumpliendo también con el Decreto Ejecutivo No. 1014 emitido el 10 de abril de 2008, donde se dispone el uso de software libre en los sistemas y equipamientos tecnológicos en las empresas Públicas del Ecuador.

El uso de buenas prácticas e instrumentos formales en el desarrollo de software fueron factores importantes de apoyo, para cumplir con los objetivos propuestos que se enfocaron a la calidad del software.

RECOMENDACIONES

Se recomienda el uso de modelos adaptativos como SCRUM para un cumplimiento organizado de los requerimientos y finalmente presentar un producto de software de calidad. Además de interpretar los requerimientos de manera minuciosa para evitar inconvenientes a futuro.

Se recomienda al GAD de Santa de Cotacachi que la aplicación móvil creada, sirva como un punto de partida para seguir desarrollando nuevos sistemas tecnológicos que faciliten procesos informativos.

Es recomendable el trabajo colaborativo entre docentes, estudiantes, industria y sociedad que permitan cumplir los objetivos propuestos.

Las aplicaciones tecnológicas en modelos económicos sobre turístico se recomienda el uso herramienta tecnológicas multiplataforma para lograr llegar el máximo número de usuarios posible.

BIBLIOGRAFÍA

- Area Tecnología. (s.f.). Obtenido de <http://www.areatecnologia.com/informatica/sistemas-operativos-moviles.html>
- Borja, S. (2 de 14 de 2014). Seguridad en Android. Obtenido de HighSec: <http://highsec.es/2014/02/seguridad-en-android-parte-i-android-y-su-arquitectura/>
- Brilhante, M., & Correa, C. (2015). análisis Comparativo de las guías de turismo en formato de apliCativo: lonely planet y mtrip. Revista Turismo Vol. 17 , 33.
- Cabraca, M., Bustamante, M., & Marco, F. (2015). Sistemas de Calidad. Obtenido de <https://portafolio-grupo3mcbf.weebly.com/>
- Carrasco, A. (19 de Abril de 2017). Xamarin Latino. Obtenido de <https://xamarinlatino.com/ventajas-de-utilizar-xamarin-f6bfad83ae9>
- Carrasco, S., & Devece, C. (2015). Análisis de la aplicación de la tecnología móvil en las empresas. Valencia, España.
- Carrion, G. (2018). COMPARATIVA DE TRES HERRAMIENTAS DE REALIDAD AUMENTADA UTILIZANDO UNA METODOLOGÍA DE MEDICIÓN DE SOFTWARE ISO 25010. Ibarra Ecuador.
- Ceballos, J. (14 de 3 de 2017). Redes@Zone. Obtenido de Redes@Zone: <https://www.redeszone.net/2017/03/24/conoce-estos-6-entornos-programacion-ide-programar-varios-lenguajes/>
- Condori, J. (27 de 8 de 2015). slideshare. Obtenido de slideshare: <https://es.slideshare.net/JohnCM2/sistemas-operativosmobilesterminado>
- Coronado, C. (2019). DESARROLLO DE UN SISTEMA WEB PARA EL FORTALECIMIENTO DE LOS PROCESOS DE GESTIÓN ADMINISTRATIVA Y FINANCIERA DEL CONDOMINIO SOLAR DEL RÍO DE LA CIUDAD DE IBARRA UTILIZANDO MICROSOFT AZURE . Ibarra Ecuador: Universidad Tecnica del Norte.
- Cuello, J., & Vittole, J. (2013). Diseñando apps para móviles. Barcelona: Maga&Seba.
- Deemer, P., Benefield, G., Larman, C., & Vodde, B. (2009). INFORMACIÓN BÁSICA DE SCRUM(THE SCRUM PRIMER). Agile-Spain: Scrum Training Worldwide.
- Delía, L., Galdamez, N., Thomas, P., Corbalan, L., & Pesado, P. (1 de 1 de 2015). Análisis Experimental de desarrollo de Aplicaciones Móviles Multiplataforma. Obtenido de http://sedici.unlp.edu.ar/bitstream/handle/10915/42355/Documento_completo.pdf?sequence=1
- Deusto. (17 de 7 de 2017). Aprender para enseñar. Obtenido de Innovación docente y TIC Universidad de Deusto: <https://blogs.deusto.es/aprender-ensenar/aplicaciones-moviles-para-el-aprendizaje-i/>
- Embarcadero. (0 de 5 de 2018). embarcadero. Obtenido de www.embarcadero.com

EMPQT. (2014). Empresa Pública Metropolitana de Quito Turismo. Obtenido de <https://www.quito-turismo.gob.ec/component/k2/itemlist/category/198-desarrollo-y-gestion-turistica>

Friedman, N. (11 de julio de 2012). Nat Friedman. Obtenido de https://es.wikipedia.org/wiki/Nat_Friedman

García, C. O., & Ibarra García, C. (30 de Abril de 2016). Ventajas de RAD. Obtenido de <https://iswugaps2rad.wordpress.com/ventajas-y-desventajas-de-rad/>

Gonzales, S., Gutierrez, E., & Vasquez, H. (2017). Metodología de proyectos informáticos. Santiago de Chile.

Hosteltur. (18 de 06 de 2019). hosteltur25. Obtenido de https://www.hosteltur.com/124100_10-mejores-aplicaciones-turismo.html

<https://sg.com.mx/revista/47/desarrollo-apps-cross-platform-xamarinforms>. (s.f.).

<https://visualstudio.microsoft.com/es/xamarin/>. (s.f.).

IBM Corporation. (2012). El desarrollo de aplicaciones móviles nativas, Web o híbridas. Estados Unidos de América: Thought Leadership White Paper.

INEC. (Abril de 2018). Tecnologías de la Infomormación y Comunicacion TIC. Obtenido de <http://www.ecuadorencifras.gob.ec/tecnologias-de-la-informacion-y-comunicacion-tic/>

José H. Canús, Patricio Letelier y Carmen Penadés. (2012). Método logías ágiles en el Desarrollo de Software. Valencia España: DSIC -Universidad Politécnica.

Joseph, J. (2013). Mobile OS. Journal of Engineering, Computers & Applied Sciences (JEC&AS), 10.

Lazaro, F. (15 de 2 de 2017). Observaciones y Tecnología | Obux. Obtenido de Observaciones y Tecnología | Obux: <https://obux.wordpress.com/2017/02/15/que-es-xamarin-y-como-funciona/>

Leboso, L. M. (2016). Estudio de la herramienta RAD Studio para el desarrollo de aplicaciones en android, iOS, Mac y Windows. Gandia España: UNIVERSIDAD POLITECNICA DE VALENCIA.

López, G. M. (2014). Informática básica: sistema operativo, Internet y correo electrónico. Manual teórico. España: Editorial CEP, S.L.

Marsala, G. (2015). Apperware. Obtenido de App Development y QR MArketing: <https://asesoresenturismoperu.files.wordpress.com/2016/02/27-aplicaciones-mc3b3viles-para-sector-turismo.pdf>

Martínez, M. (7 de 4 de 2016). CICE. Obtenido de La escuela profesional de nuevas tecnologías: <https://www.cice.es/noticia/top-5-plataformas-desarrollo-ios-android/>

MATHIESON, A. a. (1982). Tourism: Economic,Physical and Social Impacts. Longman, London.

- Medina, T. (2018). DESARROLLO DE UN SOFTWARE PARA LA GESTIÓN TÉCNICA DE LOS FACTORES DE RIESGO LABORALES DE LOS LINIEROS DE LA EMPRESA ELÉCTRICA REGIONAL NORTE EMELNORTE S.A. Ibarra Ecuador.
- Microsoft Visual Studio. (05 de 06 de 2018). Obtenido de <https://docs.microsoft.com/en-us/visualstudio/productinfo/vs2017-system-requirements-vs>
- Muñoz, M. M. (2018). Desarrollo de una aplicación móvil mediante Xamarin para el apoyo a terapeutas en la rehabilitación cognitiva de pacientes. Valladolid.
- Nahuel, L. (1 de 3 de 2017). Desarrollo de aplicaciones móviles multiplataforma. Obtenido de http://sedici.unlp.edu.ar/bitstream/handle/10915/60497/Documento_completo___.pdf-PDFA.pdf?sequence=3
- OMT. (24 de 11 de 2017). Organización Mundial de Turismo. Obtenido de Turismo no Brasil: <http://www.turismo.gov.br> Ac
- Pastor, J. (12 de 3 de 2014). Xataka Mobile. Obtenido de Desarrollo de aplicaciones móviles (I): así está el mercado: <https://www.xatakamovil.com/mercado/desarrollo-de-aplicaciones-moviles-i-asi-esta-el-mercado>
- Petzold, C. (2016). Creating Mobile Apps with Xamarin.Forms. Washington: John Pierce, Flying Squirrel Press.
- Ruano, R. (2018). ESTUDIO COMPARATIVO DE LOS FRAMEWORKS IONIC Y REACT NATIVE” APLICACIÓN MÓVIL DE PEDIDOS A DOMICILIO BASADA EN LA NORMA ISO 9126. Ibarra, Ecuador.
- San Marín, S., & Carpio, M. (11 de 5 de 2012). UNIVERSIA BUSINESS REVIEW. Obtenido de <http://www.redalyc.org/html/433/43323196007/>
- Santiago, R. (2015). Mobile learning nuevas realidades en el aula. Navarra: EDITORIAL OCEANO S.L.U.
- Scrum, E. 3.-5.-3. (1 de Agosto de 2018). scruminc. Obtenido de <https://www.scruminc.com/the-3-5-3-of-scrum/>
- Segittur Turismo e Innovación. (2016). Estudio de Mercado de Apps Turísticas. Obtenido de https://www.segittur.es/opencms/export/sites/segittur/.content/galerias/descargas/documentos/Segittur_APPS-Turismo.pdf
- Segovia, J. (17 de 3 de 2015). Desarrollador Android. Obtenido de Herramientas de Prueba: <https://desarrollador-android.com/desarrollo/herramientas/herramientas-de-prueba/>
- Shwaber, K., & Sutherland, J. (Noviembre de 2017). La Guía de Scrum. Obtenido de <https://www.scrumguides.org/docs/scrumguide/v2017/2017-Scrum-Guide-Spanish-SouthAmerican.pdf>
- Soares, A., Mendes, L., & Barbosa, A. (2017). EVALUACIÓN DE LA INFORMACIÓN DE UNA APLICACIÓN TURÍSTICA. Estudios y Perspectivas en Turismo, 15-16.

- Ugan, E., Çizmel, N., & Demirors, O. (2014). Comparison of Functional Size Based Estimation and Story Points, Based on Effort Estimation Effectiveness in SCRUM Projects. Presentado en 2014 40th EUROMICRO Conference on Software Engineering and Advanced Applications. Obtenido de IEEE: <https://doi.org/10.1109/SEAA.2014.83>
- VENEMEDIA. (Septiembre de 2018). VENEMEDIA COMUNICACIONES, C.A. Obtenido de <https://conceptodefinicion.de/windows-phone/>
- Vida Navarro, R. (20 de 7 de 2017). Tech lover. Senior IT Freelance & Developer. Obtenido de Programación IOS ¿por donde empiezo? (II): xCode, IOS SDK y Objective C. ¿Qué son?: <http://www.vidanavarro.es/programacion-ios-por-donde-empiezo-ii-xcode-ios-sdk-y-objective-c-que-son/>
- Wang, D., & Park, S. (Julio 2012). The Role of Smartphones in Mediating the Touristic Experience. *Journal of Travel Research*, 8.
- Xamarin. (26 de Junio de 2016). Introducción a Xamarin.Forms. . Obtenido de <https://docs.microsoft.com/es-es/xamarin/get-started/>

ANEXOS

Anexo 1: Pruebas de iteraciones con prototipos

a) Quinta Iteración

La quinta iteración consta de la visualización de un Tabbed Page que expone la información gastronómica del cantón Cotacachi como se ve en la segura 3.1 siguiente.

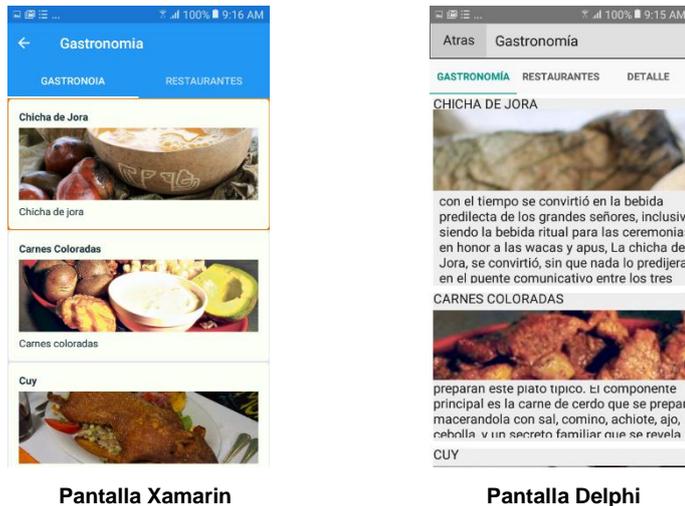


Fig. 4.1. Quinta iteración

Fuente: Propia

Energía:

Xamarin: Al ser la quinta iteración consume 0% de batería.

Delphi: Al ser la quinta atención consume 0,01% de batería al realizar esta petición.

Memoria, Rendimiento:

Xamarin: Al ser la quinta iteración consume 93,5 Mb de memoria RAM y un total del 3,69% de trabajo en CPU.

Delphi: Al ser la quinta atención consume 157 Mb de memoria RAM y un 6,03% de consumo de CPU.

Almacenamiento:

Xamarin: Al ser la quinta iteración ocupa 37,5 Mb de almacenamiento en el dispositivo.

Delphi: Al ser la quinta iteración ocupa 59,8 Mb de almacenamiento en el dispositivo

En la siguiente figura 3.2 se muestran los resultados obtenidos de la quinta iteración:

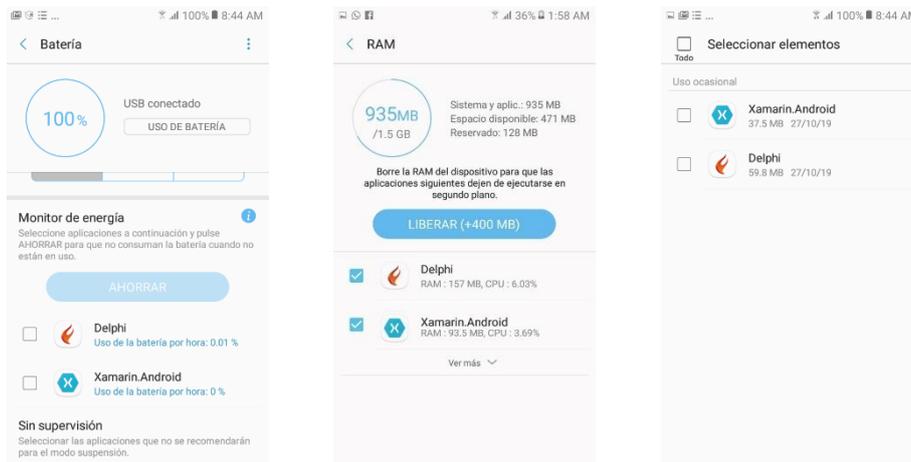
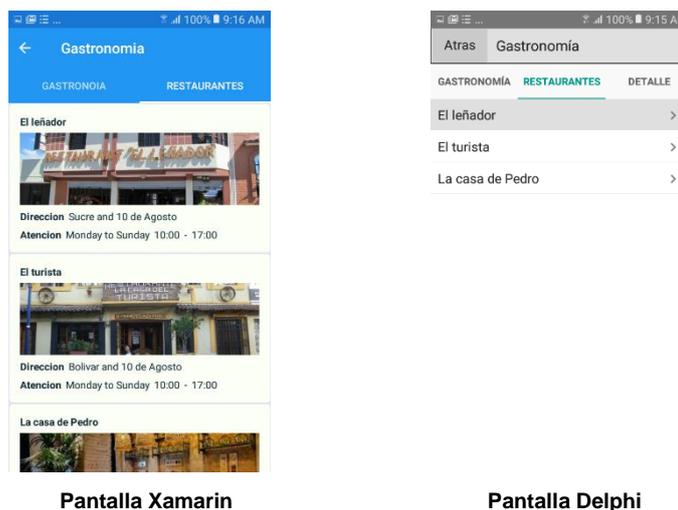


Fig. 4.2. Quinta iteración – monitoreo de las aplicaciones.

Fuente: Propia

b) Sexta Iteración

La sexta iteración consta de la visualización de un Tabbed Page que expone el listado de restaurantes del sistema como se puede observar en la figura 3.1.



Pantalla Xamarin

Pantalla Delphi

Fig. 4.3. Sexta iteración

Fuente: Propia

Energía:

Xamarin: Al ser la sexta iteración consume 0% de batería

Delphi: Al ser la sexta atención consume 0,01% de batería al realizar esta petición.

Memoria, Rendimiento:

Xamarin: Al ser la sexta iteración consume 96,3 Mb de memoria RAM y un total del 1,81% en CPU.

Delphi: Al ser la sexta atención consume 157 Mb de memoria RAM y un 4,43% de consumo de CPU.

Almacenamiento:

Xamarin: Al ser la sexta iteración ocupa 37,5 Mb de almacenamiento en el dispositivo.

Delphi: Al ser la quinta iteración ocupa 59,8 Mb de almacenamiento en el dispositivo.
 En la siguiente figura 3.4 se muestran los resultados obtenidos de la sexta iteración:

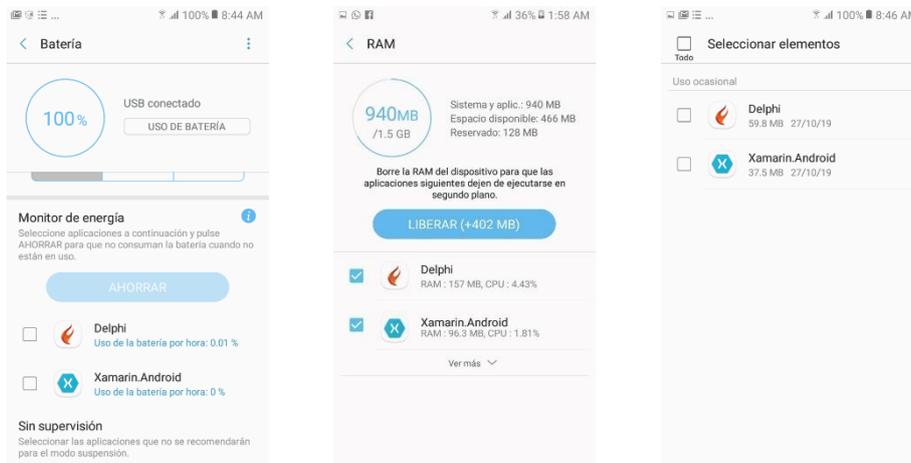
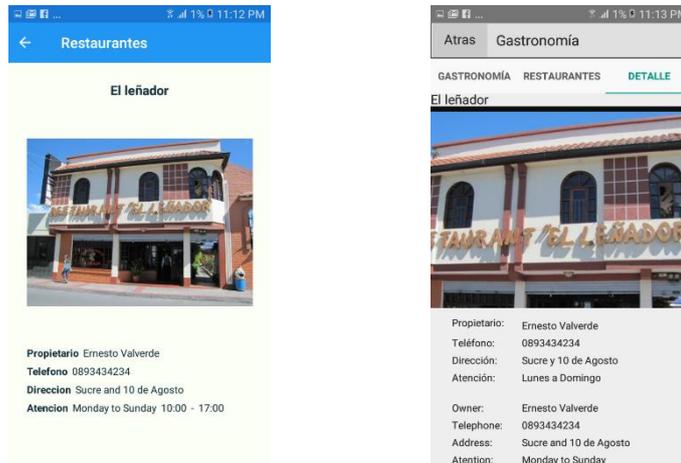


Fig. 4.4. Sexta iteración – monitoreo de las aplicaciones.

Fuente: Propia

c) Séptima Iteración

La séptima iteración consta del detalle de un restaurante seleccionado en la lista de restaurantes del Tabbed Page presentado en la sexta iteración.



Pantalla Xamarin

Pantalla Delphi

Fig. 4.5. Séptima iteración

Fuente: Propia

Energía:

Xamarin: Al ser la séptima iteración consume 0% de batería

Delphi: Al ser la séptima atención consume 0,01% de batería al realizar esta petición.

Memoria, Rendimiento:

Xamarin: Al ser la séptima iteración consume 95,9 Mb de memoria RAM y un total del 0,59% en CPU.

Delphi: Al ser la séptima atención consume 154 Mb de memoria RAM y un 4,58% de consumo de CPU.

Almacenamiento:

Xamarin: Al ser la séptima iteración ocupa 37,5 Mb de almacenamiento en el dispositivo.

Delphi: Al ser la séptima iteración ocupa 59,8 Mb de almacenamiento en el dispositivo.

En la siguiente figura 3.6 se muestran los resultados obtenidos de la séptima iteración:

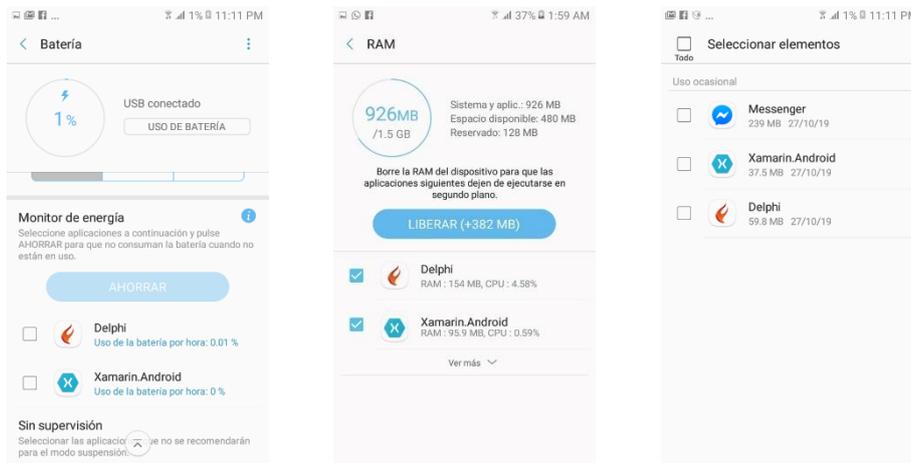


Fig. 4.6. Séptima iteración – monitoreo de las aplicaciones.

Fuente: Propia

d) Octava Iteración

La octava iteración consta del menú de establecimientos registrados en la aplicación.

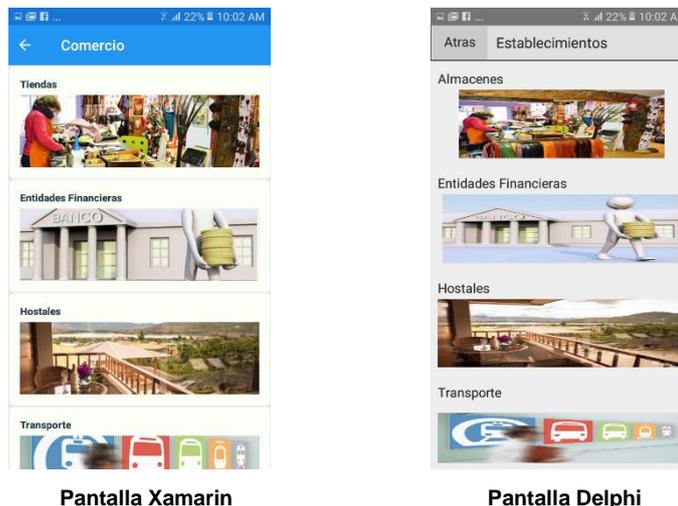


Fig. 4.7. Octava iteración

Fuente: Propia

Energía:

Xamarin: Al ser la octava iteración consume 0% de batería

Delphi: Al ser la octava atención consume 0,01% de batería al realizar esta petición.

Memoria, Rendimiento:

Xamarin: Al ser la octava iteración consume 101 Mb de memoria RAM y un total del 2,7% en CPU.

Delphi: Al ser la octava atención consume 165 Mb de memoria RAM y 7,71% de consumo de CPU.

Almacenamiento:

Xamarin: Al ser la octava iteración ocupa 37,7 Mb de almacenamiento en el dispositivo.

Delphi: Al ser la octava iteración ocupa 61,9 Mb de almacenamiento en el dispositivo.

En la siguiente figura 3.8 se muestran los resultados obtenidos de la octava iteración:

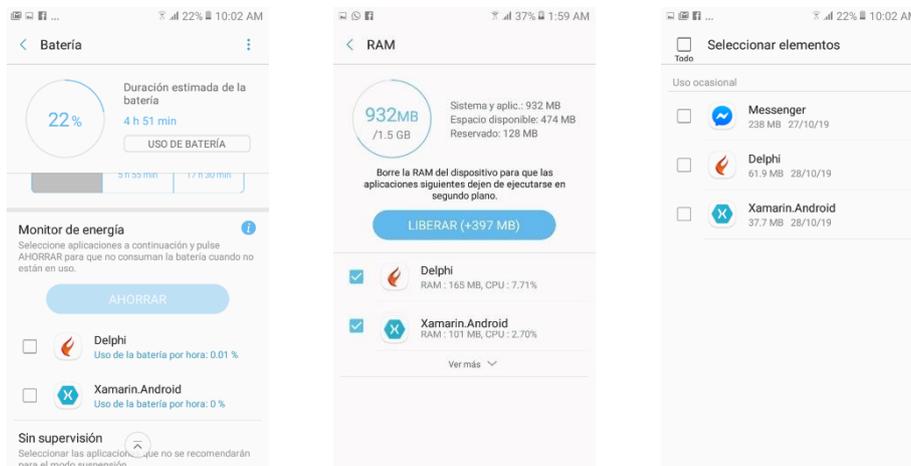


Fig. 4.8. Séptima iteración – monitoreo de las aplicaciones.

Fuente: Propia

e) Novena Iteración

La novena iteración consta del listado de establecimientos de transporte tomado desde un servicio web.

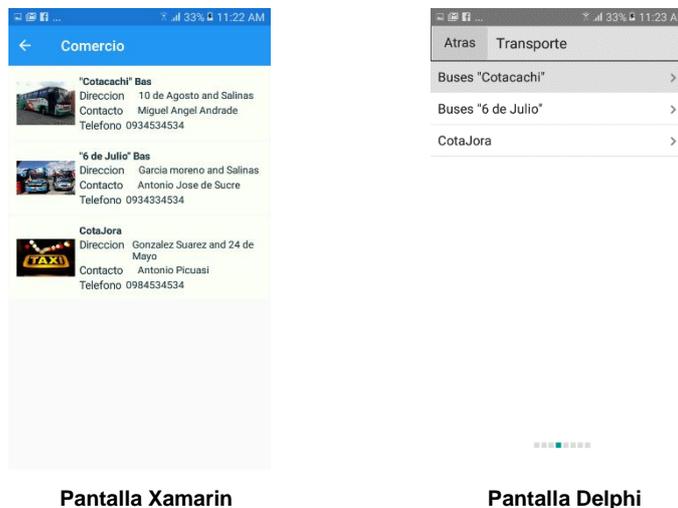


Fig. 4.9. Novena iteración

Fuente: Propia

Energía:

Xamarin: Al ser la novena iteración consume 0% de batería

Delphi: Al ser la novena atención consume 0,01% de batería al realizar esta petición.

Memoria, Rendimiento:

Xamarin: Al ser la novena iteración consume 103 Mb de memoria RAM y un total del 0,57% en CPU.

Delphi: Al ser la novena atención consume 169 Mb de memoria RAM y 4,39% de consumo de CPU.

Almacenamiento:

Xamarin: Al ser la novena iteración ocupa 38,2 Mb de almacenamiento en el dispositivo.

Delphi: Al ser la novena iteración ocupa 61,9 Mb de almacenamiento en el dispositivo.

En la siguiente figura 3.8 se muestran los resultados obtenidos de la novena iteración:

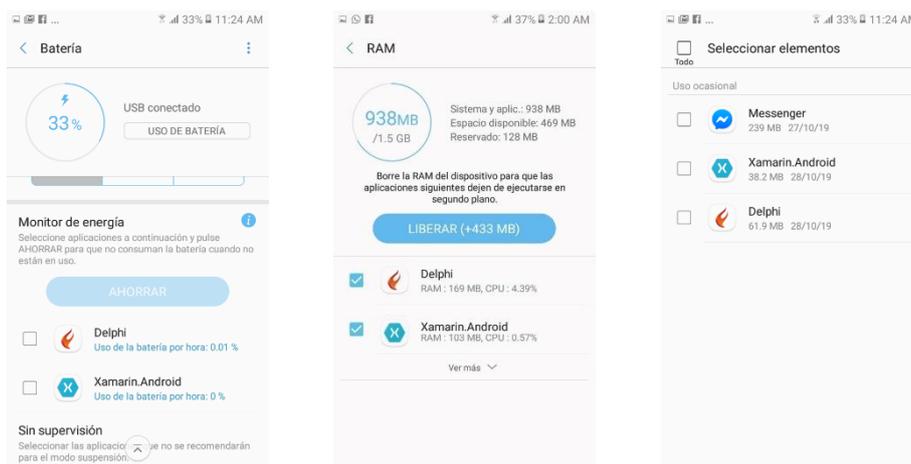
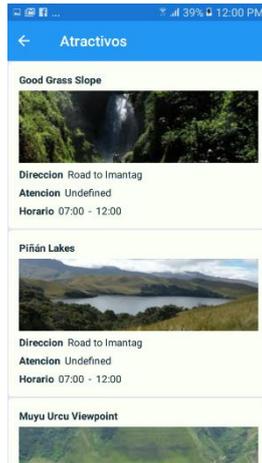


Fig. 4.10. Novena iteración – monitoreo de las aplicaciones.

Fuente: Propia

f) Décima Iteración

La décima iteración consta del listado de atractivos naturales pertenecientes al menú de atractivos del cantón Cotacachi.



Pantalla Xamarin



Pantalla Delphi

Fig. 4.11. Décima iteración

Fuente: Propia

Energía:

Xamarin: Al ser la décima iteración consume 0% de batería

Delphi: Al ser la décima atención consume 0,01% de batería al realizar esta petición.

Memoria, Rendimiento:

Xamarin: Al ser la décima iteración consume 93,5 Mb de memoria RAM y un total del 0,47% en CPU.

Delphi: Al ser la décima atención consume 140 Mb de memoria RAM y 4,09% de consumo de CPU.

Almacenamiento:

Xamarin: Al ser la décima iteración ocupa 38,2 Mb de almacenamiento en el dispositivo.

Delphi: Al ser la décima iteración ocupa 61,9 Mb de almacenamiento en el dispositivo.

En la siguiente figura 3.8 se muestran los resultados obtenidos de la décima iteración:

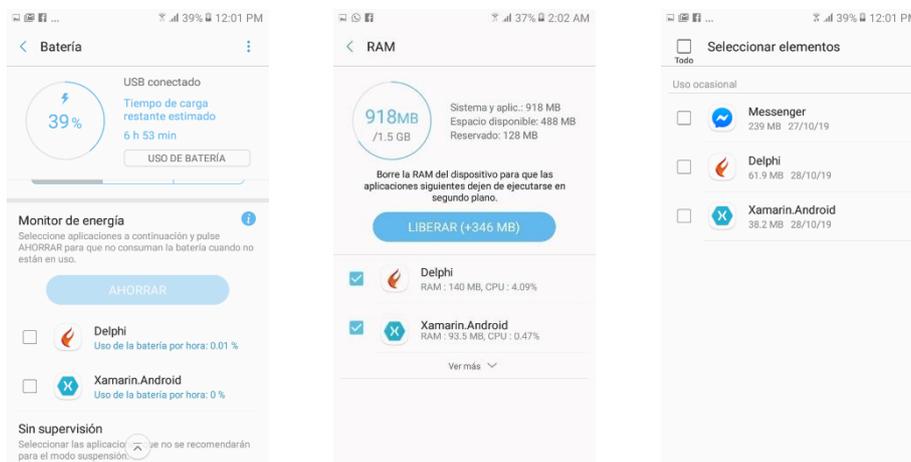
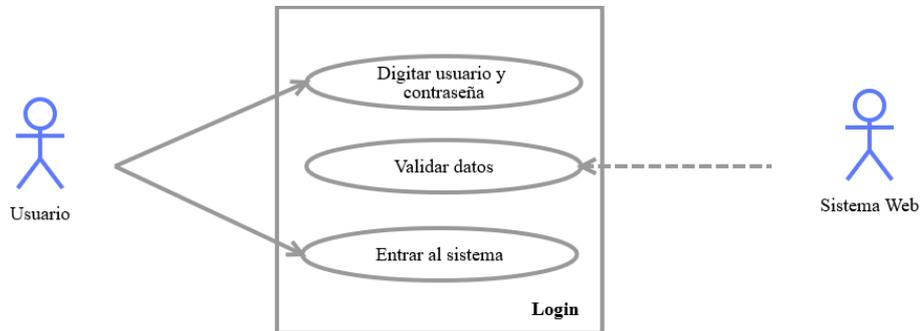


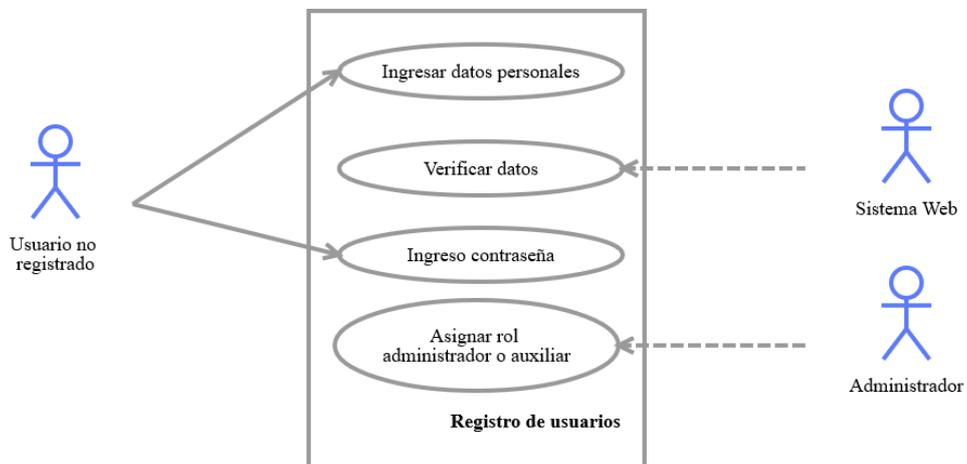
Fig. 4.12. Décima iteración – monitoreo de las aplicaciones.

Anexo 2: Casos de uso diseñados durante el desarrollo del sistema

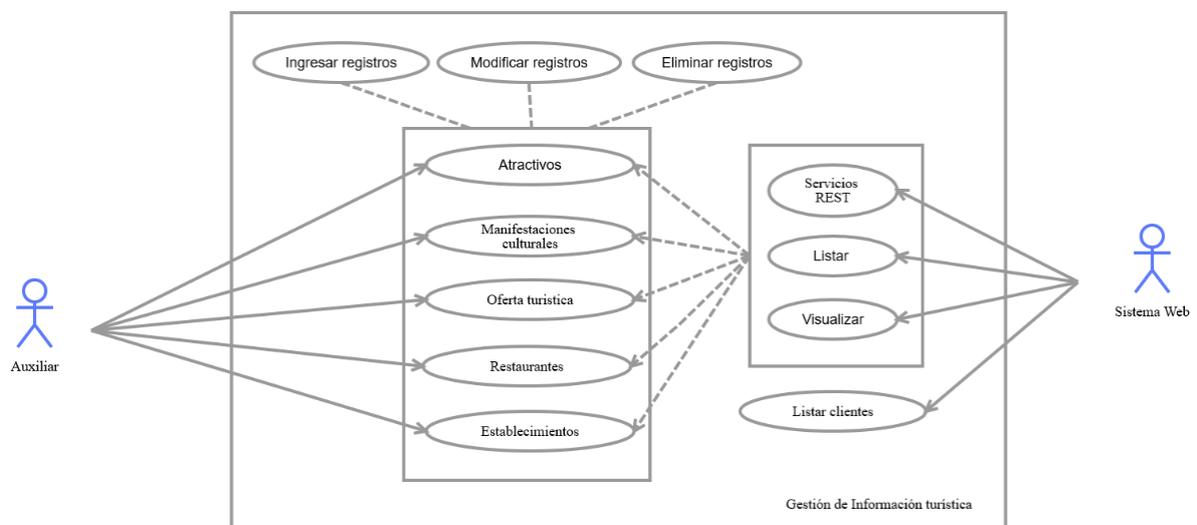
a) Login de acceso al sistema web



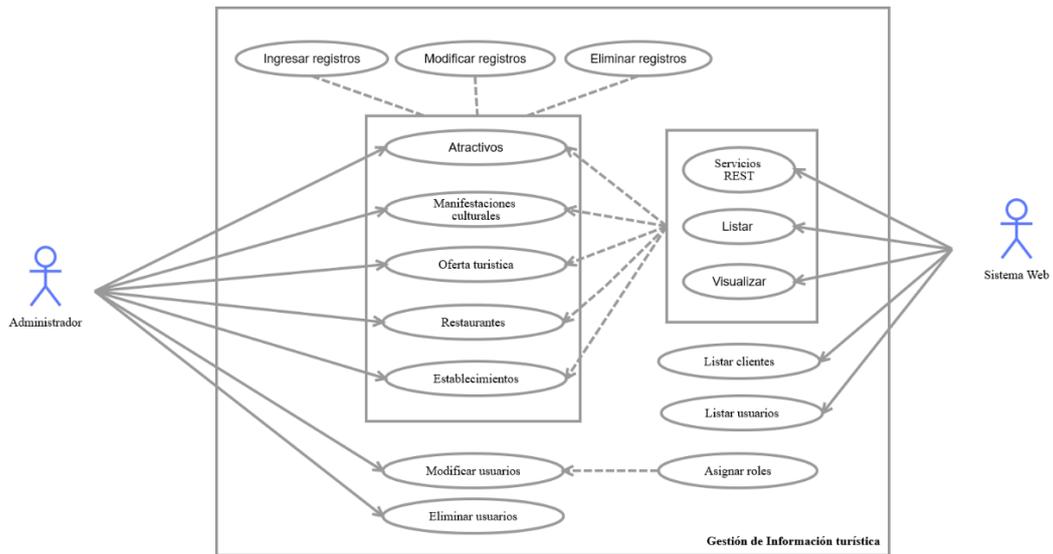
b) Registro de usuarios a la aplicación web



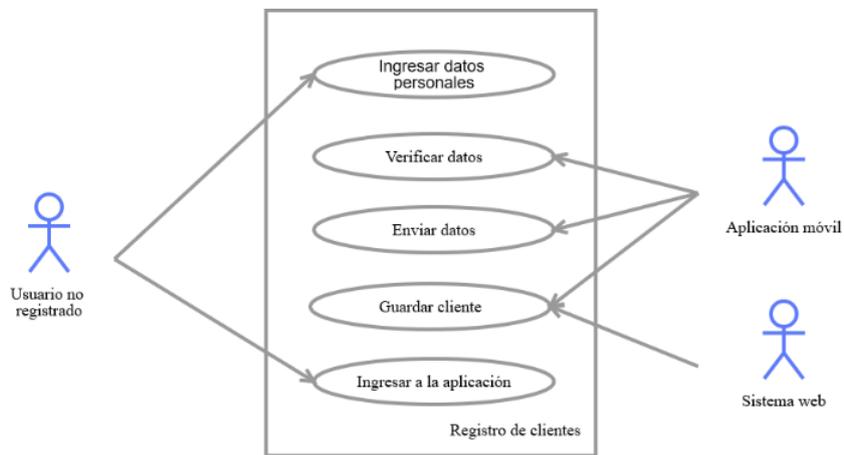
c) Rol del usuario auxiliar



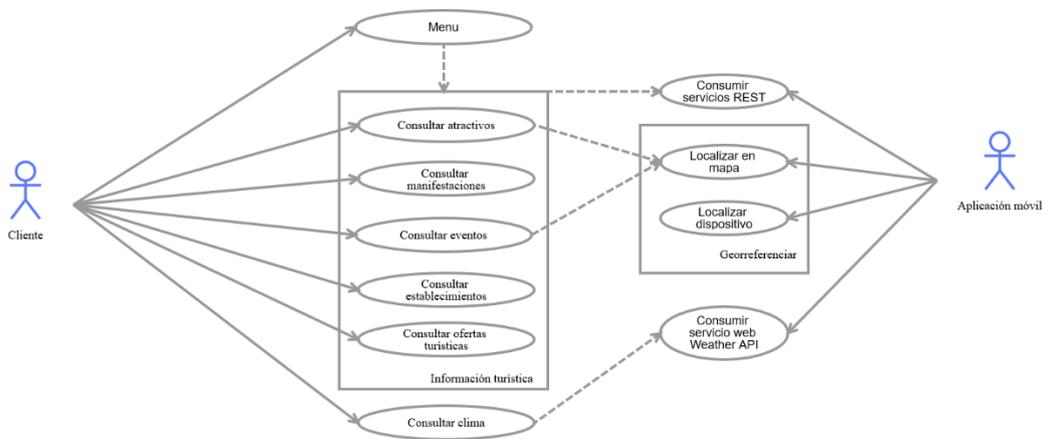
d) Rol del usuario administrador



e) Registro de clientes por medio de la aplicación móvil.



f) Rol del cliente en la aplicación turística.



| SPRINT | INICIO | DIAS |
|--------|-----------|------|
| 4 | 1-jul.-19 | 15 |

Georreferenciar los atractivos turísticos, eventos que se registren en la aplicación móvil, además el cliente podrá consultar el clima del cantón en tiempo real.

| | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| L | M | X | J | V | L | M | X | J | V | L | M | X | J | V |
| 1 | 1 | 1 | 1 | 1 | 2 | 2 | 2 | 2 | 2 | 3 | 3 | 3 | 3 | 3 |
| 3 | 3 | 5 | 4 | 3 | 4 | 3 | 4 | 3 | 4 | 4 | 5 | 4 | 4 | 4 |

Tareas pendientes
Horas de trabajo pendientes

| ID Historia | PILA DEL SPRINT | | ESTIMACIÓN EN HORAS | | | | | | | | | | | | | | Total | | | |
|-------------|---|---------------|---------------------|-------------------|---|---|---|---|---|---|---|---|---|---|---|---|-------|---|---|---|
| | Tarea | Tipo | Estado | Encargado | L | M | X | J | V | L | M | X | J | V | L | M | | X | J | V |
| H-11 | Incluir las librerías y configuraciones necesarias. En el tercer ítem del menú diseñar una pantalla con un mapa georreferenciado. | Configuración | Terminada | Jefferson Andrade | 3 | | | | | | | | | | | | | | | 3 |
| H-11 | Implementar un método para obtener la localización del dispositivo. | Codificación | Terminada | Jefferson Andrade | 3 | | | | | | | | | | | | | | | 3 |
| H-11 | Implementar un método para obtener la localización del cantón. | Codificación | Terminada | Jefferson Andrade | 5 | | | | | | | | | | | | | | | 5 |
| H-11 | Diseñar una opción de localización en la sección de eventos. | Codificación | Terminada | Jefferson Andrade | 4 | | | | | | | | | | | | | | | 4 |
| H-11 | Obtener la localización de los eventos y graficarlos en el mapa. | Codificación | Terminada | Jefferson Andrade | 3 | | | | | | | | | | | | | | | 3 |
| H-11 | En los tractivos se debe incluir una opción de ir en cada uno. | Codificación | Terminada | Jefferson Andrade | 4 | | | | | | | | | | | | | | | 4 |
| H-11 | Localizar la ubicación del atractivo seleccionado y mostrarlo en el mapa. | Codificación | Terminada | Jefferson Andrade | 3 | | | | | | | | | | | | | | | 3 |
| H-11 | Validar la conexión a internet y ubicación del dispositivo. | Codificación | Terminada | Jefferson Andrade | 4 | | | | | | | | | | | | | | | 4 |
| H-11 | En el menú principal incluir una pantalla para consultar el clima del cantón. | Codificación | Terminada | Jefferson Andrade | 3 | | | | | | | | | | | | | | | 3 |
| H-12 | Consumir un servicio web libre para tomar la descripción del estado climático. | Codificación | Terminada | Jefferson Andrade | 1 | | | | | | | | | | | | | | | 1 |
| H-12 | Ajustar los resultados para reflejarlos en inglés y español. | Codificación | Terminada | Jefferson Andrade | 2 | | | | | | | | | | | | | | | 2 |
| H-12 | Realizar la compilación del desarrollo en las tres plataformas, Android, iOS y Windows | Pruebas | Terminada | Jefferson Andrade | 1 | | | | | | | | | | | | | | | 1 |
| H-12 | Configurar la escalabilidad de pantalla de la aplicación. | Configuración | Terminada | Jefferson Andrade | 2 | | | | | | | | | | | | | | | 2 |
| H-12 | Realizar las pruebas respectivas | Pruebas | Terminada | Jefferson Andrade | 4 | | | | | | | | | | | | | | | 4 |
| H-12 | Capacitación del software | Codificación | Terminada | Jefferson Andrade | 4 | | | | | | | | | | | | | | | 4 |