



UNIVERSIDAD SIMÓN BOLÍVAR
DECANATO DE ESTUDIOS PROFESIONALES
COORDINACIÓN DE INGENIERÍA DE LA COMPUTACIÓN

**Desarrollo de una aplicación móvil basada en el sistema operativo AndroidOS
para el portal Tuguia.de**

Por:
Juan Vicente Rosas Ramirez

Realizado con la asesoría de:
Tutor Académico: Rosseline Rodríguez
Tutor Industrial: Gianpaolo Valero

PROYECTO DE PASANTÍA LARGA
Presentado ante la Ilustre Universidad Simón Bolívar
como requisito parcial para optar al título de
Ingeniero en Computación

Sartenejas, Septiembre de 2013



UNIVERSIDAD SIMÓN BOLÍVAR
DECANATO DE ESTUDIOS PROFESIONALES
COORDINACIÓN DE INGENIERÍA DE LA COMPUTACIÓN

ACTA FINAL PASANTÍA LARGA

**Desarrollo de una aplicación móvil basada en el sistema operativo AndroidOS
para el portal Tuguia.de**

Presentado por:
Juan Vicente Rosas Ramirez

Esta Pasantía Larga ha sido aprobado por el siguiente jurado examinador:

Prof. Rosseline Rodríguez

PROFESOR 2

Ing. Gianpaolo Valero

Sartenejas, 23/09/2013

Resumen

Empresas TD es una empresa dedicada a ofrecer soluciones tecnológicas e innovadoras orientadas a mejorar la calidad del e-commerce en Venezuela y a brindar la oportunidad de darse a conocer a los diferentes comercios y locales de las principales ciudades del país en este medio.

Bajo este precepto Empresas TD crea el portal Tugua.De con el objetivo de convertirse en la principal guía de locales y comercios del país. Dado el crecimiento de Tugua.De como proyecto, la empresa decide llevar esta idea a las plataformas móviles comenzando con el desarrollo de una aplicación nativa para el sistema operativo AndroidOs.

Este informe describe las actividades realizadas durante el desarrollo del proyecto de pasantía larga, que consistió en la elaboración de una aplicación móvil para el portal Tugua.de. Dicha aplicación permite llevar la iniciativa de este portal a un mayor número de usuarios de una forma sencilla, permitiendo así que éstos puedan acceder a los diferentes servicios prestados por esta guía desde cualquier lugar usando su teléfono celular Android.

El desarrollo de esta aplicación móvil, se realizó siguiendo el proceso iterativo e incremental de la metodología ágil Scrum, y los principios del desarrollo guiado por pruebas y comportamientos (*test-driven development* y *behavior-driven development*). Fue realizada con el lenguaje de programación *Java*, haciendo uso del entorno de trabajo dispuesto para Android (*Android SDK*) y SQLite como manejador de base de datos de la aplicación en el dispositivo.

Como resultado de este proyecto de pasantía se realizó el análisis, diseño e implementación de la aplicación móvil que cubre los requerimientos planteados en el alcance inicial. Adicionalmente se realizó la configuración de un servidor de ApacheSolr para mejorar el resultado de las búsquedas y diversas mejoras funcionales al API (*Application Programming Interface*) que permite acceder a las funcionalidades del portal Tugua.De siendo éste la materia prima de la aplicación móvil.

A Dios, mi familia, profesores y amigos.

Índice general

Índice general	vi
Índice de figuras	ix
Listas de Símbolos y Abreviaturas	x
1. Introducción	1
2. Entorno Empresarial	6
2.1. Antecedentes de la Empresa	6
2.2. Estructura Organizacional	7
3. Marco teórico	10
3.1. Aspectos del Negocio	10
3.1.1. Local	10
3.1.2. Taxonomías	11
3.1.3. Reseña	11
3.2. Aspectos Tecnológicos	11
3.2.1. Tecnologías Asociadas con la Aplicación Móvil	12
3.2.1.1. Teléfono Inteligente	12
3.2.1.2. Android	12
3.2.1.2.1. Arquitectura del Sistema Android	13
3.2.1.2.2. Estructura básica de una aplicación Android	15
3.2.1.2.3. <i>SQLite</i>	16
3.2.1.3. Eclipse	16
3.2.2. Tecnologías Asociadas con Tuguia.de	17
3.2.2.1. Drupal	17

3.2.2.2. Apache Solr	17
4. Marco Metodológico	19
4.1. Scrum	19
4.1.1. Roles en Scrum	20
4.1.2. El proceso de Scrum	20
4.2. Desarrollo Basado en Pruebas y Comportamientos	22
5. Desarrollo del Proyecto	24
5.1. Descripción del Sistema	24
5.1.1. Contenido del Sitio	24
5.2. Tareas de Realizadas en cada Iteración	26
5.2.1. Primera Iteración	26
5.2.1.1. Objetivos Planteados	27
5.2.1.2. Resultados Alcanzados	27
5.2.2. Segunda Iteración	29
5.2.2.1. Objetivos Planteados	30
5.2.2.2. Resultados Alcanzados	30
5.2.3. Tercera Iteración	34
5.2.3.1. Objetivos Planteados	34
5.2.3.2. Resultados Alcanzados	34
5.2.4. Cuarta Iteración	36
5.2.4.1. Objetivos Planteados	36
5.2.4.2. Resultados Alcanzados	36
5.2.5. Quinta Iteración	38
5.2.5.1. Objetivos Planteados	38
5.2.5.2. Resultados Alcanzados	38
5.2.6. Sexta Iteración	40
5.2.6.1. Objetivos Planteados	40
5.2.6.2. Resultados Alcanzados	40
5.2.7. Séptima Iteración	42
5.2.7.1. Objetivos Planteados	43
5.2.7.2. Resultados Alcanzados	43
5.2.8. Octava Iteración	44

5.2.8.1. Objetivos Planteados	44
5.2.8.2. Resultados Alcanzados	44
5.3. Resultados Generales y Estado Actual	46
6. Conclusiones y Recomendaciones	48
Bibliografía	51

Índice de figuras

2.1. Organigrama de Empresas TD.	7
2.2. Organigrama de Tuguia.de	9
3.1. Arquitectura del Sistema Operativo Android [AAT]	13
3.2. Interacción y flujo de Actividades en Android [TBS]	15
4.1. Proceso Scrum [DPB09]	21
4.2. Ciclo de <i>Test Driven Development</i>	22
5.1. Local visto en Tuguia.de [CTG]	25
5.2. Ejemplo de guión de una actividad a desarrollar	26
5.3. Esquema de trabajo para el desarrollo de funcionalidades en la aplicación móvil	31
5.4. <i>Login</i> en la aplicación móvil	32
5.5. Registro de usuarios en la aplicación móvil	33
5.6. Colaje que muestra la funcionalidad de búsqueda dada una palabra clave . .	35
5.7. Interfaz administrativa del módulo de integración de Drupal con ApacheSolr	37
5.8. Colaje que muestra la funcionalidad de búsqueda y de paginado.	39
5.9. Vista detallada de un local en la aplicación móvil.	41
5.10. Vista de un comentario completo en la aplicación móvil.	42
5.11. Interfaz para escribir comentarios en Tuguia.de.	45

Listado de Símbolos y Abreviaturas

API *Aplication Programming Interface*, Interfaz de programación de aplicaciones

BDD *Behavior Driven Development*, Desarrollo guiado por Comportamientos

CMS *Content Managememt System*, Sistema de Gestión de Contenidos

CPU *Central Processing Unit*, Unidad de Procesamiento Central

HTTP *Hypertext Transfer Protocol*, Protocolo de Transferencia de Hipertexto

IDE *Integrated Development Environment*, Entorno de Desarrollo Integrado

JSON *JavaScript Object Notation*, Notación de Objetos de *JavaScript*

NDK *Native Development Kit*, Kit de desarrollo de Nativo

SDK *Software Development Kit*, Kit de desarrollo de Software

SQL *Structured Query Language*, Lenguaje Estructurado de Consultas

TDD *Test Driven Development*, Desarrollo guiado por Pruebas

URL *Uniform Resource Locator*, Localizador de recursos uniforme

XML *Extensible Markup Language*, Lenguaje de Marcado Extensible

Capítulo 1

Introducción

En la sociedad actual se hace imprescindible que las empresas posean una página web; el mundo que hoy conocemos se basa en Internet y es en éste donde la gente busca las respuestas a sus inquietudes cotidianas. Actualmente en el mundo existen casi el doble de *smartphones* que computadores personales [DGT]. Según los resultados obtenidos en una encuesta llevada a cabo por Google en conjunto con IPSOS OTX mediaCT al final del 2010, aparte de hacer y recibir llamadas, las actividades realizadas con mayor frecuencia en los teléfonos inteligentes son: navegar en Internet, el uso de motores de búsquedas y en tercer lugar el uso de aplicaciones [TMM]. Para las empresas, ya no sólo es necesario estar en la web, deben estar al alcance de los teléfonos inteligentes.

Los *smartphones* han llegado a ser parte de la vida de las personas y son dispositivos mucho mas personales que una computadora portátil o de escritorio, estos teléfonos han penetrado hasta el punto que muchos usuarios raramente se separan de sus equipos móviles durante el día.

Tugua.de es una empresa joven que pretende ser la principal guía de comercios y locales de Venezuela, su objetivo no sólo es brindar información actualizada de los locales, sino que también exponer la experiencia que tuvieron los usuarios al visitar estos sitios. Tomando

en cuenta esto se hace importante para Tuguia.de convertirse en un servicio disponible a los usuarios 24 horas durante los 7 días de la semana, accesible desde cualquier momento y lugar. Por estas razones se hace necesario el desarrollo de un mecanismo para que los usuarios puedan acceder a la información del portal desde sus celulares.

Conscientes de la influencia que pueda tener el éxito de la versión móvil de Tuguia.de, se decide emprender un desarrollo que permita llevar la aplicación a los celulares inteligentes. Sin embargo para lograr esto existen varias alternativas a considerar; aplicaciones nativas, aplicaciones híbridas o multiplataforma y páginas web diseñadas específicamente para celulares.

Las aplicaciones nativas, llevan este nombre porque son desarrollos implementados en el lenguaje nativo del dispositivo. Gracias a esto, sacan el mejor rendimiento posible del *hardware* del móvil, así que tienen un alto desempeño y permiten acceder a las diferentes características presentes en los teléfonos inteligentes con mayor facilidad. Sin embargo, el hecho de necesitar un desarrollo en un lenguaje específico, distinto para cada plataforma eleva los costos y el tiempo de desarrollo.

Las aplicaciones híbridas permiten escribir código en un solo lenguaje para después exportarlo a código nativo y de esta manera obtener una aplicación que funcione en las múltiples plataformas con un solo desarrollo. Es de notar que estas aplicaciones tienen un bajo desempeño si las comparamos con las nativas, ademas en estos entornos multiplataforma se dificulta la tarea de acceder a los diferentes componentes del equipo móvil con efectividad.

Las páginas web de diseño específico para equipos móviles son sitios web que pueden cambiar la forma en la que muestran su contenido de acuerdo al equipo en el que se está visualizando y de esta manera aprovechar al máximo las dimensiones de las pantallas de los dispositivos. Así se consigue que con un solo desarrollo se obtiene un producto que puede operar tanto en las computadoras como en equipos móviles. Al igual que las otras páginas

web éstas se ejecutan en un navegador, por lo que se torna bastante complejo acceder a las características propias de los teléfonos inteligentes de hoy en día.

Luego de evaluar estas alternativas, la junta directiva de Tuguia.de decide que desarrollar una aplicación nativa para su proyecto. A pesar de los costos elevados y los altos tiempos de desarrollo desean que la aplicación sea lo mas extensible posible, no obstante, es necesario determinar cuál sistema operativo móvil es el más adecuado para iniciar las versiones móviles de Tuguia.de

Según un estudio realizado para finales del 2012 en los Estados Unidos, [NTD] para el tercer trimestre de ese año la participación de Android en el mercado de teléfonos inteligentes era del 52 %. En este aspecto, se hace notable la superioridad de Android sobre los demás sistemas operativos, por esta razón se escogió esta plataforma para iniciar el proceso de migración de Tuguia.de a las plataformas móviles.

Actualmente Tuguia.de cuenta con funcionalidades tales como búsqueda de locales dadas sus ubicación y/o una serie de taxonomías asociadas a éstos, además siempre que se muestra información de un local se debe mostrar su ubicación en el mapa, dirección, fotografías asociadas e información de contacto. Un aspecto importante para la visión de negocio de Tuguia.de es la capacidad de que los usuarios puedan realizar comentarios y establecer una puntuación de uno a cinco sobre los locales, esto con el objetivo de brindar no solo la información básica de un establecimiento, sino también priorizar los locales de acuerdo a la opinión de los usuarios. Adicionalmente, Tuguia.de prevé ofrecer la posibilidad a los dueños de cada local de gestionar la información de su comercio y campañas de publicidad en el sitio.

Por estas razones se plantea como objetivo general del proyecto de pasantía desarrollar una aplicación móvil basada en el sistema operativo AndroidOS para el portal Tuguia.de, que permita realizar búsquedas avanzadas sobre los locales, mostrar la información detallada de un local, realizar comentarios y que posea un manejo básico de usuarios.

Los objetivos específicos que se desean alcanzar con la ejecución de este proyecto, son los siguientes:

- Estudiar la arquitectura de la plataforma de desarrollo de Android, entornos de pruebas y demás tecnologías asociadas.
- Diseñar y desarrollar un mecanismo de comunicación con el API de Tuguia.de.
- Realizar los ajustes y extensiones necesarias sobre el API de Tuguia.de para garantizar el funcionamiento deseado.
- Diseñar e implementar un modelo de datos que permita gestionar los diferentes nodos de información presentes en Tuguia.de dentro la aplicación móvil.
- Desarrollar un módulo para la aplicación móvil que permita realizar búsquedas avanzadas, usando palabras claves y una ubicación, que puede ser las coordenadas geográficas o el nombre de un determinado lugar.
- Diseñar y desarrollar un componente que permita visualizar los locales de Tuguia.de en la aplicación móvil.
- Diseñar e implementar la funcionalidad que permita subir comentarios asociados a un local de Tuguia.de.
- Implementar diferentes mecanismos que permitan que la aplicación realice el menor uso posible de la red de datos.

El presente proyecto pretende generar una versión beta de la aplicación móvil de Tuguia.de en la plataforma Android; de manera que esta sea estable y se encuentre en un estado de completitud suficiente para ser aprobada por la directiva de la empresa y pasar a una fase de prueba mas extensa pre-despliegue.

En el presente informe se describirán aspectos del diseño e implementación de la aplicación móvil de Tuguaia.de. Está organizado en capítulos que representan los diferentes aspectos trabajados.

En el segundo capítulo, se describen las características que de la empresa con el fin de suministrar una idea global del ambiente de trabajo donde se realizó el proyecto. En el tercer capítulo, se presentan los fundamentos teóricos que fueron relevantes en la consecución de los objetivos de la pasantía. En el cuarto capítulo, se describe la metodología utilizada en el desarrollo del proyecto. En el quinto capítulo, se describen las tareas e iteraciones que se realizaron en el marco de la metodología Scrum y las soluciones desarrolladas. Finalmente el sexto capítulo, se detallan las conclusiones y recomendaciones futuras para el proyecto.

Capítulo 2

Entorno Empresarial

En este capítulo se realizará una descripción de Empresas TD y sus antecedentes, ademas se explicará en que consiste el portal web Tuguia.de que se tiene planteado logre convertirse en el núcleo de negocio para Empresas TD, así como el resto de su portafolio de productos y de esta forma dar a conocer el entorno laboral en el que se realizó el proyecto de pasantía.

2.1. Antecedentes de la Empresa

En el año 2010 la compañía Servicios 1488 C.A. realizó el lanzamientos del portal Tudescuenton.com, con el objetivo de brindar a todos los venezolanos los mejores descuentos en los diferentes comercios asociados, convirtiéndose así en la primera empresa de venta de cupones de descuento del país.

El crecimiento de Tudescuenton.com y las nuevas tendencias de comercio electrónico llevó a la compañía Servicios 1488 C.A. a consolidarse en lo que hoy se conoce como Empresas TD añadiendo a su portafolio de productos diferentes iniciativas que le permiten ofrecer soluciones tecnológicas orientadas a mejorar la calidad del comercio electrónico en Venezuela

y a brindar la oportunidad de darse a conocer a los diferentes comercios y locales de las principales ciudades del país en este medio.

De esta forma surge el portal Tugua.de con el objetivo de convertirse en la principal guía de locales y comercios del país, y en un espacio donde los usuarios puedan compartir sus experiencias al visitar estos comercios. Se tiene previsto que Tugua.de se convierta en el núcleo del negocio de Empresas TD, ya que este portal permitirá centralizar los diferentes servicios prestados por la compañía.

2.2. Estructura Organizacional

Empresas TD es una compañía con una gran proyección a futuro compuesta mayormente por gente joven. El perfil buscado en los miembros del equipo de trabajo es el de un profesional dinámico, creativo e innovador, comprometido por brindar a los asociados y usuarios experiencias de alta calidad.

Actualmente se gestionan cuatro proyectos en Empresas TD, como se puede ver en la figura 2.1 y se describen a continuación:

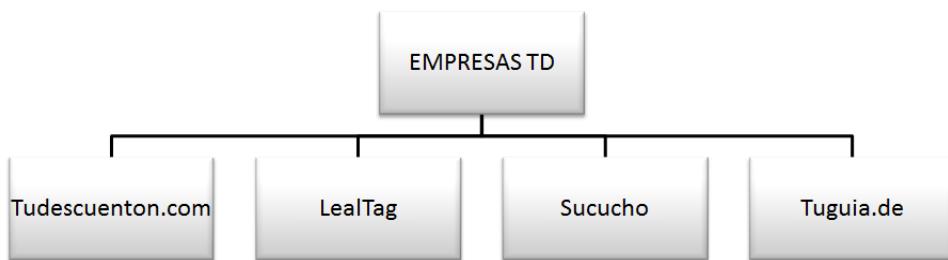


Figura 2.1: Organigrama de Empresas TD.

- **Tudescuenton:** es “una que plataforma le brinda a los venezolanos, la oportunidad de conocer las mejores cosas que hacer, en las principales ciudades del país, a través de descuentos insuperables.” [TDC]

- **Lealtag:** “es la plataforma que te permite recibir premios, descuentos y regalos en tus sitios favoritos por ser un cliente frecuente.” [LTG]
- **Sucucho:** “ofrece a los creativos venezolanos un espacio en donde pueden exponer su talento ante el país, en el cual brindamos la oportunidad de que cada uno tenga su tiendita propia 365 días al año.” [SCC].
- **Tuguiade:** “ Tuguía.de es una página que sirve para conectar los locales con sus clientes. Ofreciéndoles a los usuarios información verificada de los mismos y la posibilidad de escribir sobre sus experiencias, subir fotos e intercambiar historias con los otros clientes.” [TGD].

Cada proyecto tiene asignado un director general y un esquema organizacional específico que se ajusta a sus necesidades. En el caso de Tuguia.de, proyecto en el que desarrolló la pasantía durante periodo Abril-Septiembre de 2013, el organigrama se muestra en la figura 2.2. Como se puede observar el pasante fue parte de la coordinación de programación, esta unidad opera bajo la supervisión del Ing. Gianpaolo Valero Gerente del Departamento de Tecnología de Empresas TD y esta a cargo del Lic. Alexander Mayora Líder de Desarrollo del proyecto Tuguia.de.

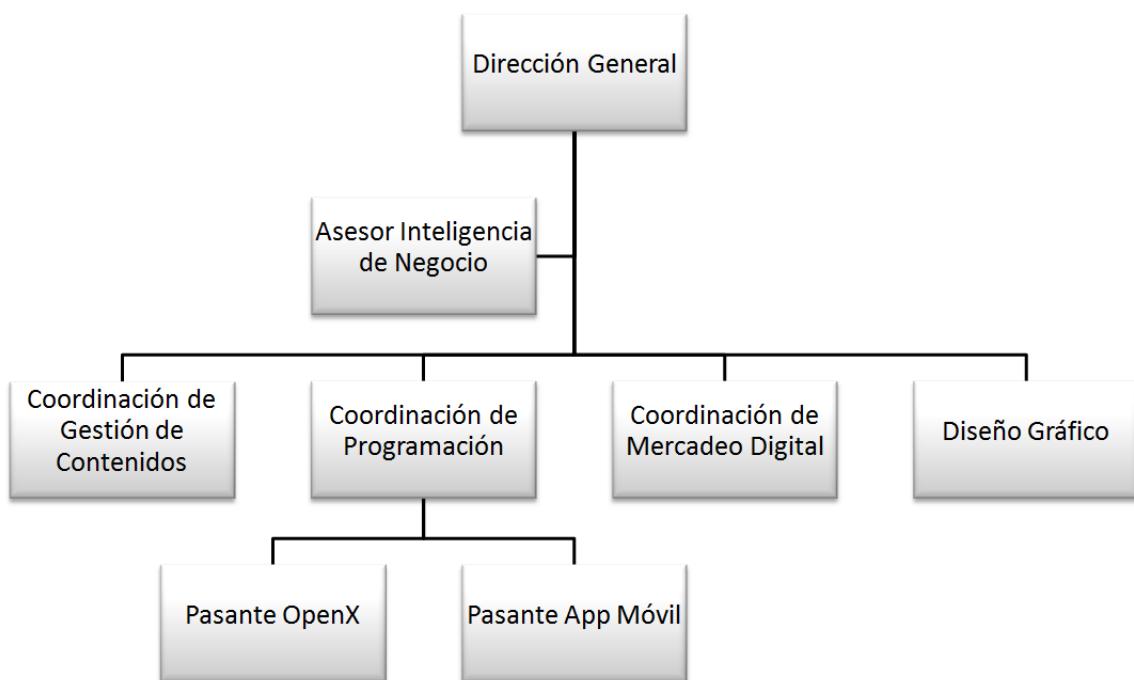


Figura 2.2: Organigrama de Tugua.de

Capítulo 3

Marco teórico

En este capítulo se presentan los fundamentos teóricos más relevantes involucrados con el proyecto de pasantía, así como también las herramientas y tecnologías aplicadas durante el desarrollo de la aplicación móvil.

3.1. Aspectos del Negocio

Al consistir el proyecto de pasantía en el desarrollo de la versión móvil para el portal Tugua.de, se hace necesario comprender algunos de los conceptos y términos asociados con el negocio y de esta manera facilitar el entendimiento de los posteriores capítulos.

3.1.1. Local

Un local comercial es un establecimiento que tiene como objetivo principal el desarrollo de alguna actividad comercial o económica. Los locales pueden ofrecer productos o servicios. Algunos ejemplos para los primeros son supermercados, farmacias, tiendas, etc. En el caso de los locales que ofrecen servicios podemos mencionar restaurantes, lugares de entretenimiento

como cines o discotecas, entre otros. [DLC]

3.1.2. Taxonomías

Una taxonomía es un tipo de vocabulario o conjunto de palabras en que todos los términos están conectados, su objetivo es organizar la información y mejorar la búsqueda de contenidos en los sitios web [Cen05]. En el caso de Tuguia.de las taxonomías usadas son: ciudad, urbanización, categoría, subcategoría y atributos.

3.1.3. Reseña

“Para Tuguía.de, una reseña es un pequeño texto en donde el usuario cuenta su experiencia. Puede ser también llamada comentario. Éstas serán escritas en los cuadros de texto dentro de cada local, y servirán para que cada usuario cuente su historia” [TGD]. Para que una reseña sea válida el usuario debe asignarle una calificación de uno a cinco al local que esta comentando.

3.2. Aspectos Tecnológicos

En esta sección se presentan las tecnologías y herramientas utilizadas en el desarrollo de la pasantía, así como también algunos términos asociados que son de obligatoria referencia para facilitar al lector la comprensión del contenido subsiguiente. Es necesario hacer una separación entre las tecnologías y herramientas utilizadas para el desarrollo de la aplicación móvil y las asociadas a Tuguia.de con las que se interactuó en el transcurso de la pasantía.

3.2.1. Tecnologías Asociadas con la Aplicación Móvil

A continuación se presentan las diferentes tecnologías y herramientas involucradas en la construcción de la aplicación Android para Tuguia.de

3.2.1.1. Teléfono Inteligente

Los teléfonos inteligentes o *smartphones*, son teléfonos celulares que ademas de poseer las características habituales de un celular común, permiten acceder a una series de aplicaciones integradas, navegar en Internet, tomar fotografías, grabar videos, entre otras funcionalidades que lo asemejan a una computadora portátil móvil [PCM]. El objetivo del proyecto de pasantía es llevar a Tuguia.de a estos dispositivos.

3.2.1.2. Android

Android es un sistema operativo basado en *Linux*, diseñado en principio para dispositivos móviles con pantalla táctil, es de código abierto y es liberado bajo la licencia Apache; una licencia bastante flexible, que permite que cualquier desarrollador realice aplicaciones y las integre al sistema sin dificultad.

Las aplicaciones nativas se desarrollan, usando el SDK (*Software Development Kit*) y escribiendo la aplicación en lenguaje *Java*. El SDK es un paquete proporcionado directamente por Google que encapsula todas las bibliotecas y herramientas de desarrollo necesarias para crear, probar, y depurar las aplicaciones [ASD]. Por otro lado existe la alternativa de usar el NDK (*Native Development Kit*) también provisto por Google que permite mediante el uso de lenguajes como *C* y *C++* interactuar directamente con el sistema operativo y crear bibliotecas. Esta última alternativa sólo es recomendada para situaciones específicas donde la aplicación en cuestión realiza cálculos intensivos sobre el *CPU* y debe emplearse sólo cuando el SDK no provee de la funcionalidad deseada [AND].

3.2.1.2.1. Arquitectura del Sistema Android

Gran parte del éxito de este sistema operativo se debe a su arquitectura. En un inicio el sistema fue pensado solo para ser usado en dispositivos móviles, sin embargo, dada su gran capacidad de adaptación, actualmente ademas de ser usado en celulares, también en televisores, reproductores de automóvil, computadores portátiles y diversos equipos electrónicos de todo tipo.

Como se observa en la figura 3.1 el sistema operativo se desglosa de la siguiente manera:

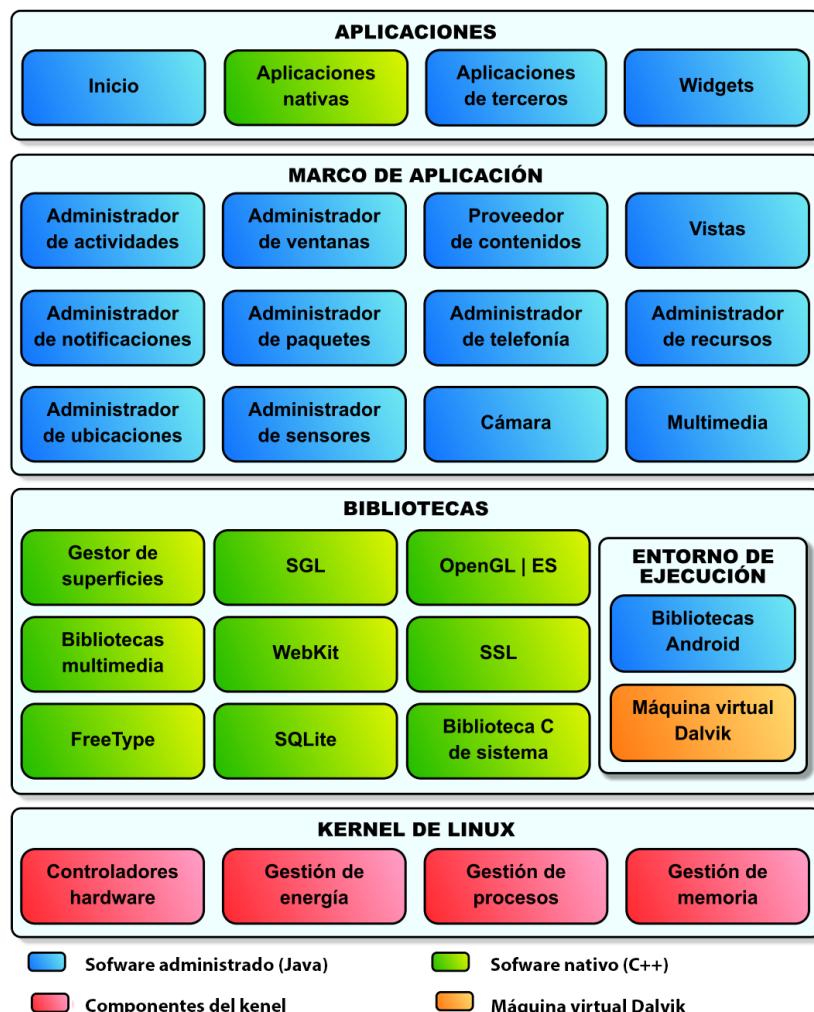


Figura 3.1: Arquitectura del Sistema Operativo Android [AAT]

- Aplicaciones: En este nivel se encuentran todas las aplicaciones presentes en el equipo, tanto las instaladas por el usuario como las provistas por el fabricante. Es importante destacar que todas cuentan con la misma capacidad para acceder los servicios que proporciona el sistema operativo. Gracias a esto, se pueden hacerse desarrollos que usen los mismos recursos que usan las aplicaciones nativas. [AAT].
- Marco de aplicación: Esta capa esta formada por las diferentes clases que componen el *API* que permiten a las aplicaciones realizar sus funciones, constituyen en el entorno de trabajo que posibilita acceder a los diferentes recursos del dispositivo [AAT].
- Bibliotecas: Esta capa se ubica justo sobre el Kernel del sistema operativo y está constituida por un conjunto de bibliotecas escritas en *C* o *C++* que son compiladas para la arquitectura específica del equipo, su función es interactuar con el sistema operativo para acceder a los diferentes recursos gestionados por éste y poner al alcance de las aplicaciones la funcionalidad respectiva. [AAT].
- Entorno de ejecución: Este no se considera una capa como tal puesto que se compone principalmente de un conjunto de bibliotecas que proporcionan la funcionalidad base. El elemento principal de esta sección es la maquina virtual Dalvik que ejecuta cada de las aplicaciones instaladas en el dispositivo [AAT].
- Kernel de Linux: El núcleo del sistema operativo Android es un kernel Linux bastante similar al que puede incluir cualquier distribución de GNU/Linux conocida. En el caso de Android se realizan adaptaciones que permita ajustarse a las características del hardware en el que se ejecutará. Su función principal es proveer una capa de abstracción para el hardware y gestionar los diferentes recursos del teléfono y del sistema operativo [AAT].

3.2.1.2.2. Estructura básica de una aplicación Android

Una actividad es una clase Java que contiene todo el código necesario para realizar una determinada tarea. Las actividades son el componente principal de las aplicaciones Android. Una aplicación puede estar compuestas por una o varias actividades que en conjunto desarrollan la funcionalidad de la aplicación.

Por lo general, las actividades requieren la interacción del usuario. Para lograr esto, la actividad genera en la pantalla una serie de elementos, como campos de textos, botones, formularios, imágenes, entre otros. Estos elementos reciben el nombre de vistas, y son organizados y parametrizados mediante archivos *XML* (*Extensible Markup Language*) llamados *layouts*. A pesar de que la disposición de las vistas es constante en los archivos XML, una actividad puede modificar totalmente las vistas y sus atributos a tiempo de ejecución.

Cada vez que una aplicación ejecuta una actividad específica, se genera una pila con las actividades relacionadas. Es decir, solo una actividad puede estar en primer plano a la vez, las actividades predecesoras son empiladas en la memoria del teléfono. Cuando una actividad termina o el usuario presiona el botón atrás se pasa la siguiente en la pila. La figura 3.2 ilustra el comportamiento.

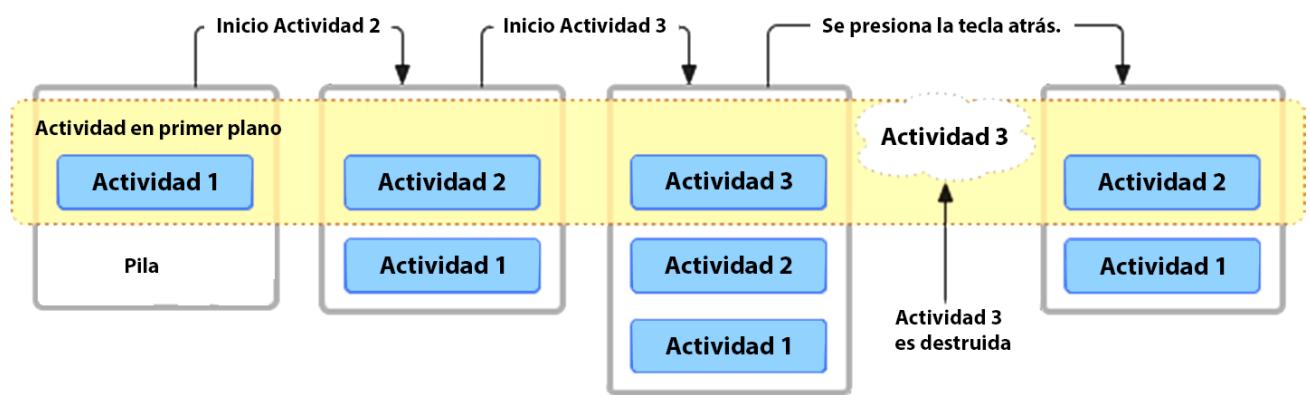


Figura 3.2: Interacción y flujo de Actividades en Android [TBS]

3.2.1.2.3. *SQLite*

SQLite es un manejador de base de datos relacional de código abierto, usa sintaxis *SQL* (*Structured Query Language*), implementa transacciones y soporta *triggers*, En *SQLite* el conjunto de la base de datos (definiciones, tablas, índices, y los propios datos), son guardados como un sólo archivo y el requerimiento de memoria durante la ejecución es bastante bajo aprox 250KByte, lo que hace que este manejador sea fácilmente embebido en otros sistemas. Gracias a estas características el sistema Android cuenta con una base de datos *SQLite* integrada y de fácil acceso para las aplicaciones.

3.2.1.3. *Eclipse*

Eclipse es un entorno integrado de desarrollo (*Integrated development environment*, IDE por sus siglas en inglés) gratuito y de código abierto. Está escrito en *Java* con el objetivo de realizar aplicaciones en este lenguaje. Sin embargo permite la integración con otros como *C*, *C++*, *Ada*, *Haskell*, *PHP*, *Python*, *Ruby*.

Este ambiente de trabajo es recomendado por Google para la construcción de aplicaciones Android, ya que a través de la instalación de un complemento distribuido en el sitio oficial, permite integrar en este entorno el soporte necesario para emplear diversas herramientas específicas para Android que facilitan el desarrollo. Cuenta con plantillas para las diversas clases básicas, instrumentos para el análisis del desempeño de la aplicación y facilidades para la gestión de emuladores que permiten simular *smartphones* con diversas características y funciones.

Por ser el entorno oficial de desarrollo posee un amplio soporte y el respaldo de la comunidad, por lo que es el ambiente escogido para implementar Tuguia.de móvil.

3.2.2. Tecnologías Asociadas con Tugua.de

Durante el desarrollo del proyecto, fue necesario incidir directamente sobre el API de Tugua.de para agregar funcionalidad, hacer mejoras y ajustes necesarios para garantizar un adecuado desarrollo de la aplicación móvil, las tecnologías implicadas en este proceso serán descritas a continuación.

3.2.2.1. Drupal

Drupal es un marco de gestión de contenidos de código abierto escrito en lenguaje *PHP*. En otra palabras, es una interfaz de programación de aplicaciones para crear sistema de gestión de contenidos (*Content Management System*, CMS por su siglas en inglés). Una instalación base del sistema o *el núcleo de Drupal* provee los elementos necesarios para el funcionamiento de un CMS básico, sin embargo, la funcionalidad se puede extender mediante la integración y desarrollo de módulos o paquetes [TV10].

El sitio Tugua.de fue desarrollado con esta herramienta y mediante la instalación, configuración y elaboración de varios módulos fue implementado un API que permite mediante peticiones HTTP (*Hypertext Transfer Protocol*) gestionar el contenido del portal.

3.2.2.2. Apache Solr

Apache Solr es una plataforma de búsqueda de alto desempeño para sitios web, es de código abierto, esta escrita en lenguaje *Java* y se basa en la biblioteca *Apache Lucene* [APS]. Apache Solr opera como un servicio separado del servidor web y la base de datos, puede incluso llegar a requerir un servidor dedicado para su operación.

De forma sencilla el funcionamiento de Apache Solr puede ser descrito de la siguiente manera: Mediante archivos *XML* llamados Documentos, se especifican los contenidos que serán indexados para su posterior búsqueda, estos índices pueden ser vistos como una lista

de palabras en la que cada palabra en la lista posee una referencia hacia los elementos que la contienen, cuando se realiza una consulta a Apache Solr se ejecuta una búsqueda sobre las listas y se retornan los elementos que cumplan con el criterio establecido en la consulta.

Apache Solr permite realizar búsquedas de texto (*Full-Text Search*) por palabra clave de forma similar al buscador de Google y tiene la capacidad para manejar búsquedas geoespaciales, por lo que en Tugua.de, es ésta la herramienta elegida para realizar las búsquedas y es integrada a Drupal mediante un módulo de código abierto ampliamente configurable.

Capítulo 4

Marco Metodológico

En Empresas TD es costumbre usar la metodología ágil Scrum, y los principios del desarrollo guiado por pruebas y comportamientos (*Test-driven development* y *Behavior-driven development*), para sus proyectos de desarrollo. Por lo tanto, el proceso de creación de la aplicación móvil para el portal Tuguia.de estuvo enmarcado de ésta forma. Este capítulo presenta y describe este marco de trabajo y las actividades que lo componen.

4.1. Scrum

Scrum es un marco de trabajo iterativo e incremental para el desarrollo de proyectos y aplicaciones. El desarrollo está organizado en ciclos de trabajo llamados *Sprints* o iteraciones que son de duración fija y van sucediendo uno detrás de otro. Es importante destacar que sin importar que no se haya terminado el trabajo un *Sprint* nunca se alarga [DPB09]. En el caso de Empresas TD las iteraciones tienen una duración de dos semanas.

4.1.1. Roles en Scrum

Dentro de la metodología Scrum existen tres roles principales que debe ser cumplidos para la aplicación efectiva del proceso, estos son el el Dueño de Producto (DP), el Equipo y el ScrumMaster (SM). A continuación se describe cada rol, estas definiciones fueron extraídas de [DPB09].

- El **Dueño del Producto** es el encargado de identificar las funcionalidades del producto y asignarles prioridad de acuerdo a las necesidades del negocio y debe revisar el resultado de cada iteración. Esta persona debe interactuar continuamente con el Equipo y tiene la autoridad final en el proyecto. En el caso de esta pasantía el dueño del producto fue el director general de Tuguia.de.
- El **Equipo** es el encargado de construir el producto, en este caso el pasante. Debe poder autogestionarse y contar con un alto grado de autonomía y responsabilidad ya que al final de cada *Sprint* debe poder tener un producto *entregable*.
- El **ScrumMaster** es el encargado de guiar al equipo y al DP en la aplicación y uso fructífero de Scrum, lo que hace es facilitar el proceso y ser un árbitro garante de la metodología y de que los demás actores ejecuten efectivamente su rol. Esta responsabilidad era de la gerencia e tecnología de Empresas TD.

4.1.2. El proceso de Scrum

Como ya fue mencionado la metodología Scrum está organizada en ciclos de trabajo que van ocurriendo uno tras otro. Estos ciclos están compuestos de actividades tal como se describe a continuación y como se muestra en la figura 4.1. Las definiciones siguientes fueron obtenidas de [DPB09].

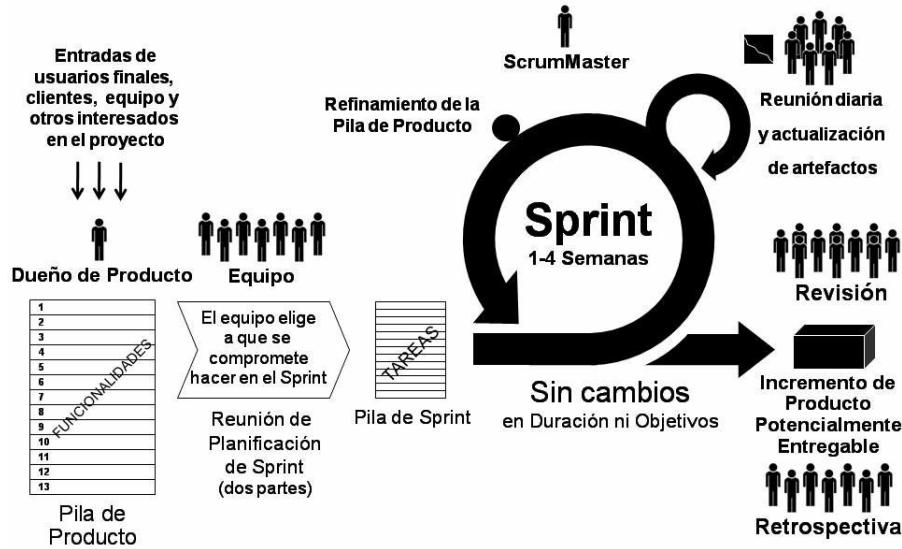


Figura 4.1: Proceso Scrum [DPB09]

- **Planificación de la iteración:** La primera fase consiste en una reunión realizada al comienzo de la iteración donde al comienzo el DP presenta al equipo la lista de requisitos priorizada, se resuelven las dudas y se seleccionan los requisitos que se compromete a completar en la iteración, cada requisito es separado en tareas y se estima la cantidad de esfuerzo en tiempo necesario para completar cada una función del grado de dificultad de la misma.
- **Ejecución de la iteración:** Esta es la fase más larga, en cuando se realiza el desarrollo del entregable. Durante esta fase, el equipo se reúne brevemente cada día para informar del progreso y dificultades. Cada miembro del equipo debe contestar únicamente las siguientes preguntas.

¿Qué ha hecho desde la última reunión?

¿Qué tiene planificado hacer antes de la siguiente reunión?

¿Qué impedimentos tiene o considera que va a tener?

- **Inspección y adaptación:** Al finalizar el *Sprint* se realiza una reunión, donde el equipo presenta el Dueño de la Aplicación los resultados obtenidos. En función de los resultados mostrados y de los cambios que haya habido en el contexto del proyecto, el DP realiza las adaptaciones necesarias de manera objetiva sobre la lista de requisitos. Por último el equipo debe analizar cómo ha sido su desempeño y cuáles son los problemas que podrían impedirle progresar adecuadamente, tratando de mejorar de manera continua su productividad.

4.2. Desarrollo Basado en Pruebas y Comportamientos

El proceso descrito por la metodología *Scrum* no establece prácticas específicas para la construcción del producto. Es por esto que en Empresas TD se integran los principios del desarrollo Basado en Pruebas o *Test Driven Development* (TDD) y el Desarrollo Basado en Comportamientos, en inglés *Behavior Driven Development* (BDD), como prácticas a seguir durante el fase de programación. El procedimiento de desarrollo usando TDD se muestra en la figura 4.2.

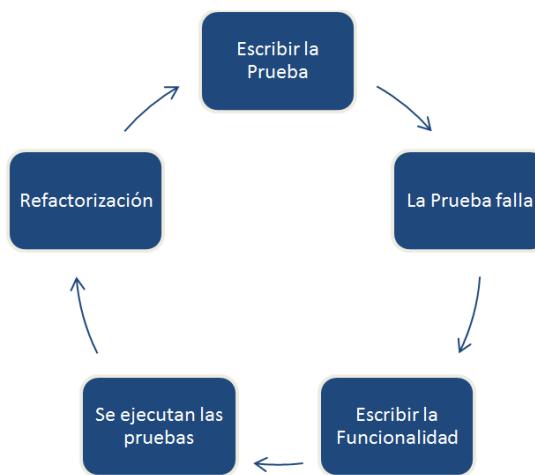


Figura 4.2: Ciclo de *Test Driven Development*

En primer lugar, se escribe una prueba y se verifica que ésta falla. Luego se implementa el código que hace que la prueba pase satisfactoriamente y seguidamente se refactoriza el código escrito. Por último, se agrega una nueva prueba y se inicia el proceso nuevamente. Esta técnica es utilizada hasta que la funcionalidad deseada esté completa y trabaja adecuadamente.

La forma más efectiva de aplicar esta técnica es segmentar la funcionalidad en pequeñas unidades que puedan ser probadas de manera automatizada e inequívoca, así se genera código simple y fácil de mantener gracias al proceso de refactorización asociado. Sin embargo aplicar esta técnica por sí sola es complejo, puesto que en ocasiones es difícil determinar qué probar y en qué nivel de detalle.

BDD surge como una manera de subsanar esta dificultad. En lugar de basar las pruebas en pequeñas unidades de funcionalidad, BDD establece que las pruebas deben reflejar los comportamientos deseados por los diferentes actores involucrados en el proyecto.

Para que se pueda aplicar BDD de manera adecuada es de vital importancia que todos los involucrados manejen un mismo léxico con respecto al proyecto, que les permita describir los comportamientos deseados en términos comunes sin caer en detalles de implementación. BDD sugiere la utilización de un esquema similar al siguiente para especificar los comportamientos [IBD].

- Dado: Un contexto inicial.
- Cuando: El evento a probar ocurre.
- Entonces: Se aseguran que se alcancen los resultados deseados.

Este enfoque permite que las especificaciones no deban ser reescritas ya que el comportamiento pocas veces cambia durante el desarrollo, sólo cambia la implementación. Además facilita el proceso de determinar los requerimientos mientras que acelera la documentación.

Capítulo 5

Desarrollo del Proyecto

Este capítulo describe, el proceso de investigación, diseño, desarrollo e implementación del proyecto de pasantías. Se realiza una descripción del sistema actual, se exponen las distintas tareas que conformaron las iteraciones de trabajo y se presentan los resultados obtenidos en cada una de ellas.

5.1. Descripción del Sistema

Antes de iniciar con el desarrollo de la aplicación móvil se realizó un estudio de Tuguia.de con el objetivo familiarizarse con el contenido del sitio y el funcionamiento del API que alimenta la aplicación móvil.

5.1.1. Contenido del Sitio

El contenido de Tuguia.de es bastante variado, si bien el sitio esta planteado como una guía de locales, el portal no solo ofrece información sobre estos, también cuenta con contenido que corresponde a otro tipo de establecimientos y servicios, que van desde entidades financieras,

instituciones gubernamentales, rutas de transporte, hasta instituciones educativas.

Los atributos relacionados con cada uno de estos *nodos* de información son los siguientes: nombre, ciudad, urbanización, dirección, coordenadas geográficas, teléfono de contacto, página web, correo electrónico, cuenta en *Facebook*, cuenta en *Twitter* una o varias imágenes relacionadas y una puntuación que va del uno al cinco. Esta puntuación es el resultado de promediar las calificaciones otorgadas por los usuarios en sus reseñas. Por último, los nodos poseen una serie de taxonomías que permiten clasificarlos en diferentes categorías.

Mediante la realización de peticiones de tipo HTTP en los servicios provistos por el API es posible gestionar los diferentes contenidos del sitio y sus atributos. En la figura 5.1 se presenta una imagen de un local en Tuguia.de.



Figura 5.1: Local visto en Tuguia.de [CTG]

5.2. Tareas de Realizadas en cada Iteración

Antes de iniciar cada iteración, se sostenía una reunión entre el Dueño del Producto, el *ScrumMaster* y el Pasante, donde se seleccionaban las actividades a realizar durante el *sprint*. Cada actividad se descomponía en tareas y se realizaba un guión sencillo. En la figura 5.2 se presenta un guión basado en una actividad realizada en el proceso de desarrollo.

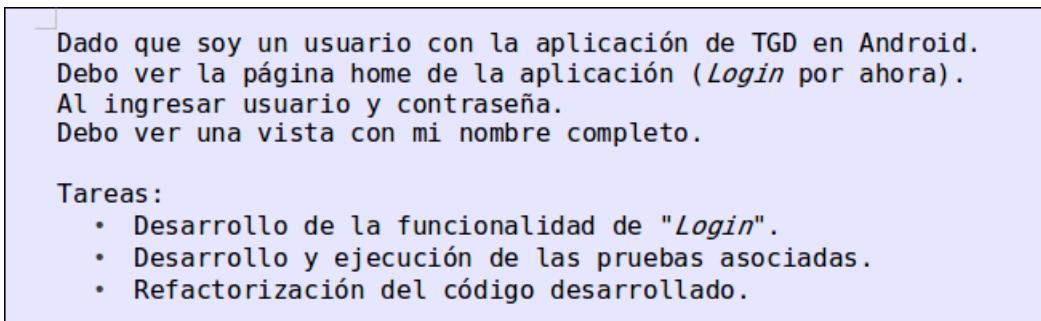


Figura 5.2: Ejemplo de guión de una actividad a desarrollar

En las secciones siguientes se realizará una descripción de las tareas realizas, los objetivos planteados en función de las actividades y los resultados obtenidos en cada una de las ocho iteraciones en las que se desarrolló la aplicación móvil. Aunque no esta planteado de forma implícita en los objetivos de cada iteración, en todos los desarrollos se incluye la construcción y ejecución de los casos de prueba.

5.2.1. Primera Iteración

El *sprint* inicial del proyecto consistió en una investigación con el fin de adquirir conocimientos acerca del funcionamiento de las aplicaciones móviles Android, así como de las metodologías y herramientas necesarias para el desarrollo del proyecto. Además fue necesario profundizar en el funcionamiento del API y sus características.

5.2.1.1. Objetivos Planteados

A continuación se enumeran los objetivos planteados:

- Adquirir conocimientos del funcionamiento y arquitectura del Sistema Operativo Android y de la estructura de las aplicaciones desarrolladas para esta plataforma.
- Evaluar las herramientas necesarias para el desarrollo del proyecto.
- Investigar los mecanismos para la aplicación de las metodologías TDD y BDD, en el entorno Android.
- Analizar las características de cada uno de los servicios prestados por el API de Tuguia.de.

5.2.1.2. Resultados Alcanzados

En la sección 3.2.1 referente a las tecnologías asociadas con la aplicación móvil se expone la información recaudada en esta iteración. Luego de analizarla, se preparó el entorno de trabajo instalando el IDE en la computadora asignada y se construyeron varias aplicaciones simples que reafirmaran los conceptos adquiridos.

Para la aplicación de las metodologías se evaluaron varias herramientas entre las que destacan el entorno de pruebas provisto en el SDK y *Robolectric* que es un proyecto de código abierto realizado por terceros.

En el SDK están incluidos todos los elementos necesarios para probar cada aspecto que compone a una aplicación Android, gracias a un entorno de pruebas basado en *JUnit*¹. La ventaja principal de esta herramienta es que al estar embebida no requiere instalar ni aprender ningún complemento externo. Sin embargo, esto trae como desventaja la dependencia de un emulador para su ejecución, lo que retrasa las pruebas en gran medida. *Robolectric* es un marco de trabajo que subsana esta situación, ya que permite simular el entorno Android y

¹*JUnit*: Es un marco de trabajo para el desarrollo de pruebas sobre aplicaciones Java [JUN]

ejecutar los casos de prueba directamente en la máquina virtual de Java, no obstante al ser una biblioteca externa implica un aprendizaje extra.

Dadas estas características, se decidió implementar la metodología usando *Robolectric*, ya que se consideró que el tiempo a invertir en aprender la herramienta es inferior al tiempo que se emplearía ejecutando los casos de pruebas en un entorno de trabajo emulado a lo largo de todo el proyecto.

La última fase de esta iteración consistió en el análisis del API de Tuguia.de. Este servicio se encuentra bajo dos capas de seguridad, la primera es una credencial compuesta por un nombre de usuario y una contraseña que deben ser suministrada en la cabecera de cada petición , esto con el objetivo de que el API no pueda ser accedido por entes externos a la organización. El segundo nivel es la jerarquía de permisos de usuario del sistema de Tuguia.de, de tal forma que un usuario del sistema puede realizar las mismas operaciones tanto en el sitio web como a través del API.

Al momento de iniciar la pasantía los recursos disponibles eran los siguientes:

- buscar local: Este servicio se emplea para hacer consultas sobre locales utilizando varios criterios de búsqueda, de no ser utilizado ningún parámetro de búsqueda, se devolverán todos los locales activos de Tuguia.de.
- buscar comentario: mediante este recurso se pueden obtener los comentarios para uno o mas locales, de no utilizar ningún parámetro de búsqueda, se devolverán todos los comentarios aprobados realizados sobre locales activos de Tuguia.de.
- buscar taxonomía: A través de esta funcionalidad se obtienen todos los términos de un vocabulario dado dentro de Tuguia.de, por ejemplo, ciudades, categorías, subcategorías, atributos, etc.
- *user*: Mediante este elemento se accede a las funcionalidades que permiten el manejo

de usuarios, específicamente a *login*, *logout* y registro.

- comentario: Este recurso permite registrar un comentario en Tuguia.de, para su uso es necesario estar conectado como un usuario registrado.
- local: Este servicio permite agregar locales al sitio. Al igual que en el recurso anterior para su uso es necesario estar conectado como un usuario registrado.

Cada uno de estos servicios fue probado y a pesar de funcionar adecuadamente en el caso de éxito, no disponían de un manejo de errores para el resto de los casos. Esta situación debía ser solventada puesto que representaba un gran riesgo para la aplicación móvil, por lo que se decidió investigar un poco más a fondo el funcionamiento del API. Afortunadamente, esta interfaz contaba con una documentación lo suficientemente extensa que permitió familiarizarse con los módulos que lo componían sin dificultad. Gracias a esto se agregó un manejo de errores básico sin alterar la planificación.

Las respuesta del API eran en formato *JSON*² lo que representó una ventaja gracias a la facilidad existente para manejar este lenguaje. Sin embargo, estas respuestas debían ser ajustadas a la funcionalidad deseada en la aplicación móvil. Este proceso de ajuste se pospuso y se realizó a medida que fue necesario y de esta forma invertir el menor tiempo posible fuera de la aplicación móvil.

5.2.2. Segunda Iteración

Luego del trabajo realizado en la primera iteración, se poseían los elementos necesarios para iniciar el desarrollo de la aplicación móvil, así que se inició con *login*, *logout*, registro y una vista sencilla que permita observar una lista de locales.

²JSON: *JavaScript Object Notation*, es un estándar para el intercambio de datos ligero y legible [JSO]

5.2.2.1. Objetivos Planteados

En la segunda iteración se plantearon los siguientes objetivos:

- Desarrollar la funcionalidad de *login* y *logout* para los usuarios registrados.
- Implementar un mecanismo de registro para los usuarios nuevos.
- Construir la funcionalidad que permita visualizar una lista de locales. Este mecanismo será usado posteriormente en las búsquedas.

5.2.2.2. Resultados Alcanzados

Por ser esta la primera iteración de desarrollo fue necesario construir un conjunto de funcionalidades secundarias antes de trabajar directamente en los objetivos planteados.

En primer lugar se elaboró un mecanismo *cliente* para realizar peticiones al API que se ajustasen a las políticas de seguridad de este y que pueda recibir las respuestas en el formato JSON. Este mecanismo debe poder ser configurado para generar cualquier tipo de petición.

En segundo lugar se diseñó y construyó un procedimiento para transformar las respuestas provenientes del API a objetos que pudieran ser manejados con mayor facilidad dentro del entorno de la aplicación. Esto fue posible gracias a la biblioteca GSON, que permite transformar un objeto Java en su representación en JSON y viceversa.

Con estas dos herramientas en funcionamiento se desarrollaron los elementos necesarios para alcanzar los objetivos planteados en la iteración siguiendo el esquema presentado en la figura 5.3

Diagrama de desarrollo

En primer lugar se realizó el *layout*, una interfaz sencilla para la funcionalidad en cuestión, no se hace énfasis en la apariencia de éste ya que al finalizar la aplicación, las vistas serán reactualizadas por el diseñador de Tuguia.de. Luego se revisó a fondo el funcionamiento del servicio

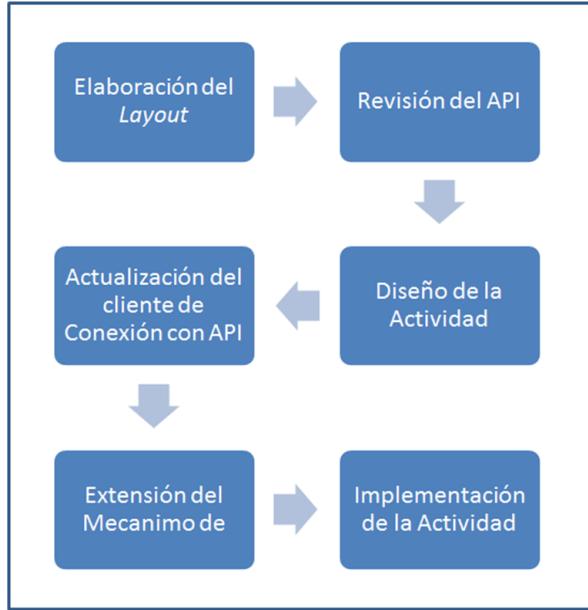


Figura 5.3: Esquema de trabajo para el desarrollo de funcionalidades en la aplicación móvil

del API con el que se iba a interactuar, con el fin de garantizar su correcto funcionamiento. En tercer lugar se realizó el diseño de la actividad que representaría el comportamiento a implementar. Seguidamente se actualizó el mecanismo cliente para poder realizar las peticiones necesarias para este comportamiento y se extendió el procedimiento para traducir las respuestas en formato JSON a objetos Java de acuerdo a las necesidades. Por último se realizó la construcción de la actividad diseñada.

Dadas las características incrementales de la metodología empleada, este flujo de trabajo fue usado repetidamente cada vez que se agregaba una nueva funcionalidad, adaptándolo a las necesidades según el caso y realizando los respectivos casos de prueba.

En el caso del *login* y el registro la funcionalidad se desarrolló sin realizar validaciones de contenido y con un manejo de errores incompleto, es decir, no se realizó una verificación de los casos de fallo en los que se requiere la participación del usuario. Por otro lado para esta iteración el mecanismo de visualizar la lista de locales construido operaba con una serie

de locales simulados sin conectarse a Tugua.de. Para cada elemento de la lista de locales se muestra la siguiente información: nombre, imagen principal, puntuación, número de comentarios, ciudad, urbanización, subcategoría, si este último atributo no estaba disponible, en su lugar se muestra la categoría. En las figuras 5.4 y 5.5 se puede observar la funcionalidad desarrollada para el *login* y el registro respectivamente.

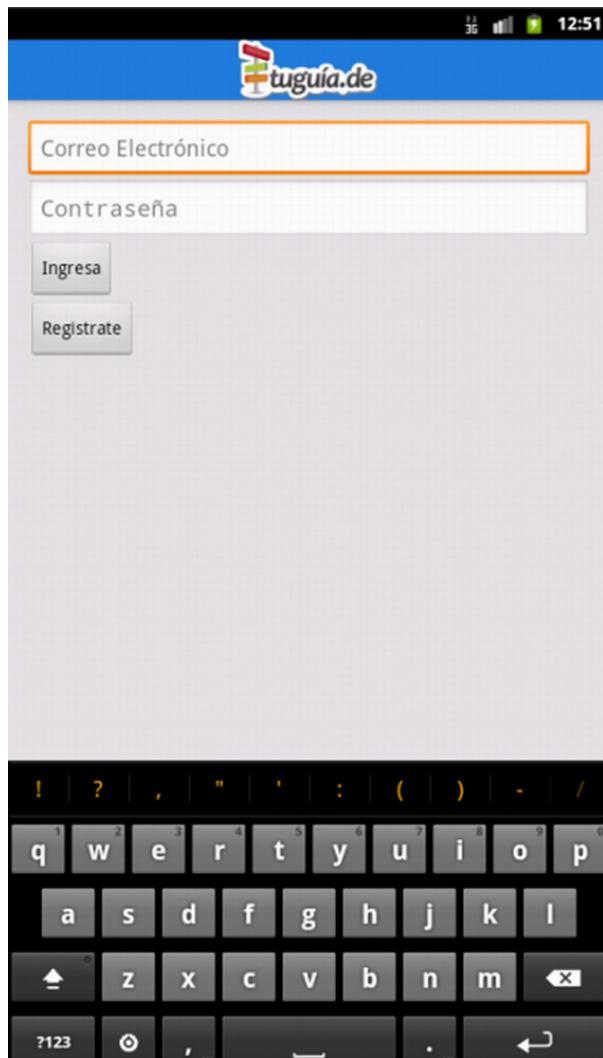


Figura 5.4: *Login* en la aplicación móvil



Figura 5.5: Registro de usuarios en la aplicación móvil

Un aspecto importante a resaltar es que la información ofrecida de la imagen por el API es el url (*uniform resource locator*); que no es más que la dirección donde se ubica este recurso en Internet. Para acceder a la imagen es necesario realizar una petición a esta dirección para obtenerla. Gracias a este situación y al gran tamaño de las imágenes manejadas en Tuguia.de el consumo de memoria y de la red de datos es bastante alto. Quedó pendiente para futuras iteraciones mejorar esta condición.

5.2.3. Tercera Iteración

En esta iteración se planteó implementar búsquedas sobre los locales de Tuguia.de dadas una palabra clave y una ubicación. Además se tenía pendiente elaborar el manejo de errores de la iteración anterior.

En este punto del proyecto el módulo del API donde se implementó el manejo de errores, recibió una actualización, por lo que había que instalarla y migrar el manejo de errores como una tarea de mantenimiento.

5.2.3.1. Objetivos Planteados

En la tercera iteración se plantearon los siguientes objetivos.

- Actualizar el módulo del API.
- Migrar el manejo de errores.
- Desarrollar la funcionalidad que permita realizar búsquedas dada una palabra clave y una ubicación.
- Completar los casos de fallo pendientes de la iteración dos.

5.2.3.2. Resultados Alcanzados

Luego de investigar los mecanismos para la actualización de módulos en Drupal, se hizo un respaldo del sistema y se procedió a actualizar el módulo en cuestión, éste proceso resultó ser bastante automatizado y no presentó mayores inconvenientes. No obstante al inspeccionar el módulo, este había sufrido una refactorización completa, por lo que el proceso de migrar la funcionalidad correspondiente al manejo de errores fue extenso pero se pudo lograr efectivamente.

Para implementar la funcionalidad de búsqueda nuevamente se siguió el esquema de presentando en la figura 5.3. Como en el API no existía un mecanismo para realizar búsquedas por ubicación o coordenadas geográficas, se decidió implementar esta funcionalidad usando solamente palabras claves. Además se integró con el mecanismo de visualizar locales desarrollado en la iteración anterior. El resultado obtenido de esta se muestra en la figura 5.6. El desarrollo necesario para poder realizar las búsquedas dadas una palabra clave y una ubicación o un par de coordenadas se postpuso a una futura iteración.

La funcionalidades pendientes fueron desarrolladas con éxito.

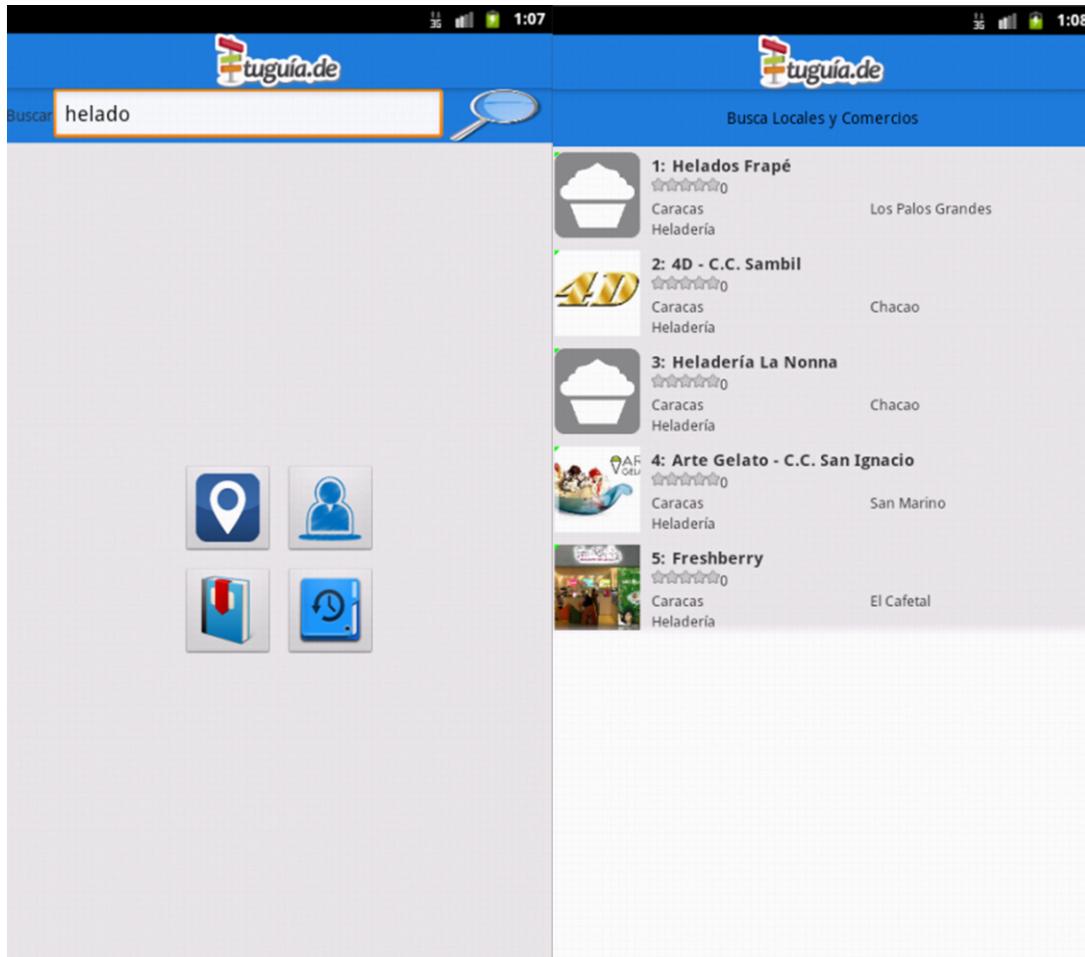


Figura 5.6: Colaje que muestra la funcionalidad de búsqueda dada una palabra clave

5.2.4. Cuarta Iteración

En esta iteración se hizo una pausa sobre el desarrollo de la aplicación móvil y se realizó un análisis de la tecnología ApacheSolr y su integración con Drupal dadas las deficiencias encontradas en el *sprint* anterior, con el objetivo de poder extender la funcionalidad del API.

5.2.4.1. Objetivos Planteados

A continuación se enumeran los objetivos planteados para la cuarta iteración:

- Adquirir conocimientos acerca del funcionamiento Apache Solr.
- Investigar el mecanismo de integración de Drupal y Apache Solr.
- Implementar la funcionalidad necesaria para realizar búsquedas desde el API, dadas una palabra clave y una ubicación.
- Desarrollar los mecanismos necesarios para realizar búsquedas desde el API, dadas una palabra clave, unas coordenada geográfica y un radio.

5.2.4.2. Resultados Alcanzados

En la sección 3.2.2.2 referente a las tecnologías asociadas con Tuguia.de se expone la información recaudada en esta iteración acerca de la herramienta Apache Solr. Luego de analizarla, se determinó la manera de incluir la funcionalidad deseada en su comportamiento.

Para implementar la búsqueda por palabra clave y una ubicación se accedió a la configuración del módulo de Drupal que permite integrar Apache Solr al sistema. Mediante la interfaz administrativa de este modulo se agregó un campo de texto llamado ubicación en el documento de configuración de Apache Solr.

Luego se desarrolló una extensión para este módulo donde se insertó en el campo ubicación, la información contenida, en los atributos ciudad, urbanización y dirección de un local.

Cuando se corre el proceso de generación de indices en Apache Solr mediante la interfaz de configuración observado en la figura 5.7, el contenido de este campo se segmenta en palabras para que cada una sea tomada como un elemento para la búsqueda.

The screenshot shows the Apache Solr search interface on a Drupal site. At the top, there's a navigation bar with links like 'Panel de control', 'Contenido', 'Estructura', 'Apariencia', 'Personas', 'Módulos', 'Configuración' (which is selected), 'Informes', and 'Ayuda'. Below the navigation is a sub-navigation for 'Apache Solr search' with tabs for 'DEFAULT INDEX', 'PAGES/BLOCKS', and 'OPCIONES'. A message at the top says '3663 items successfully processed. 3663 documents successfully sent to Solr.' Below this, a table titled 'localhost server: Search Index Content' provides details about indexed items:

TIPO	VALOR
Indexed	3663 Items (413 sent but not yet processed)
Remaining	0 elementos (100% has been sent to the server)
Schema	drupal-3.0-rc2
Delay	2 mins before updates are processed.
Pending Deletions	0

Below the table, there's a link 'View more details on the search index contents'. Under the 'ACCIONES' section, there are four buttons: 'Index queued content (50)' (Indexes just as many items as 1 cron run would do.), 'Index all queued content' (Could take time and could put an increased load on your server.), 'Queue all content for reindexing', and 'Delete the Search & Solr index' (Useful with a corrupt index or a new schema.xml.). Under the 'CONFIGURACIÓN' section, it says 'Select the entity types and bundles that should be indexed.'

Figura 5.7: Interfaz administrativa del módulo de integración de Drupal con ApacheSolr

Por último fue necesario agregar un nuevo servicio al API. Para lograr esto se desarrolló y configuró un complemento mediante el cual se hizo posible acceder a la nueva funcionalidad de búsqueda dada una palabra clave y una ubicación. Por ejemplo al pasar las palabras *pizza* y *Chacao*, se obtienen todos los locales que en algún atributo tenga la palabra pizza y la palabra Chacao en su dirección, ciudad o urbanización.

Como se determinó en la investigación, Apache Solr posee la capacidad para realizar búsquedas dadas un par de coordenadas geográficas. Sin embargo esta funcionalidad no se había configurado en Tuguia.de. Para realizar búsquedas de este tipo fue necesario editar manualmente los archivos de configuración y los documentos de Apache Solr y de esta manera

agregar un campo del tipo *location*. Esta configuración se hizo de forma manual y no mediante el módulo de integración como en el caso anterior, debido a que este no presenta soporte para este tipo de campos.

El proceso seguido a partir de aquí es igual al caso anterior, se agregó la extensión para el módulo de integración de Apache Solr y se agregó un nuevo servicio al API que permite acceder a la funcionalidad.

5.2.5. Quinta Iteración

Con la nueva funcionalidad disponible en el API de *Tuguia.de*, se retoma el desarrollo de la aplicación móvil con el fin de incluir la búsqueda de locales, elemento fundamental de *Tuguia.de*.

5.2.5.1. Objetivos Planteados

En la quinta iteración se plantearon los siguientes objetivos.

- Modificar la funcionalidad de búsqueda implementada en la tercera iteración para que adicional a la palabra clave, también reciba ubicación.
- Implementar la funcionalidad necesaria para realizar búsquedas desde el API, dadas una palabra clave, unas coordenada geográfica y un radio.
- Desarrollar un mecanismo de paginación para el resultado de las búsquedas.

5.2.5.2. Resultados Alcanzados

Como la funcionalidad de búsqueda dada una palabra clave fue implementada en la tercera iteración, en este caso sólo fue necesario modificar el cliente para que usase el nuevo servicio elaborado en la iteración anterior.

En cuanto a la búsqueda por coordenadas, se siguió el flujo de desarrollo planteado en la segunda iteración mostrado en la figura 5.3, gracias a este esquema se concluyó con éxito la funcionalidad.

La paginación se implementó modificando el mecanismo para listar locales construido en el segundo *sprint*, de tal forma de que se muestra un máximo de veinte locales por página. Si se realiza una búsqueda con más de veinte locales en el resultado, se listan los veinte primeros y aparece un botón de “siguientes” al final de la vista. Al ser pulsado, realiza la petición al API solicitando los veinte siguientes y así sucesivamente hasta mostrar todos los resultados de la búsqueda. Siempre es posible retornar a la página de procedencia pulsando el botón “anteriores” que realiza una petición por los locales que corresponden. Este botón es mostrado sólo si se ha avanzado de la primera página. En la figura 5.8 se ilustra la funcionalidad descrita.

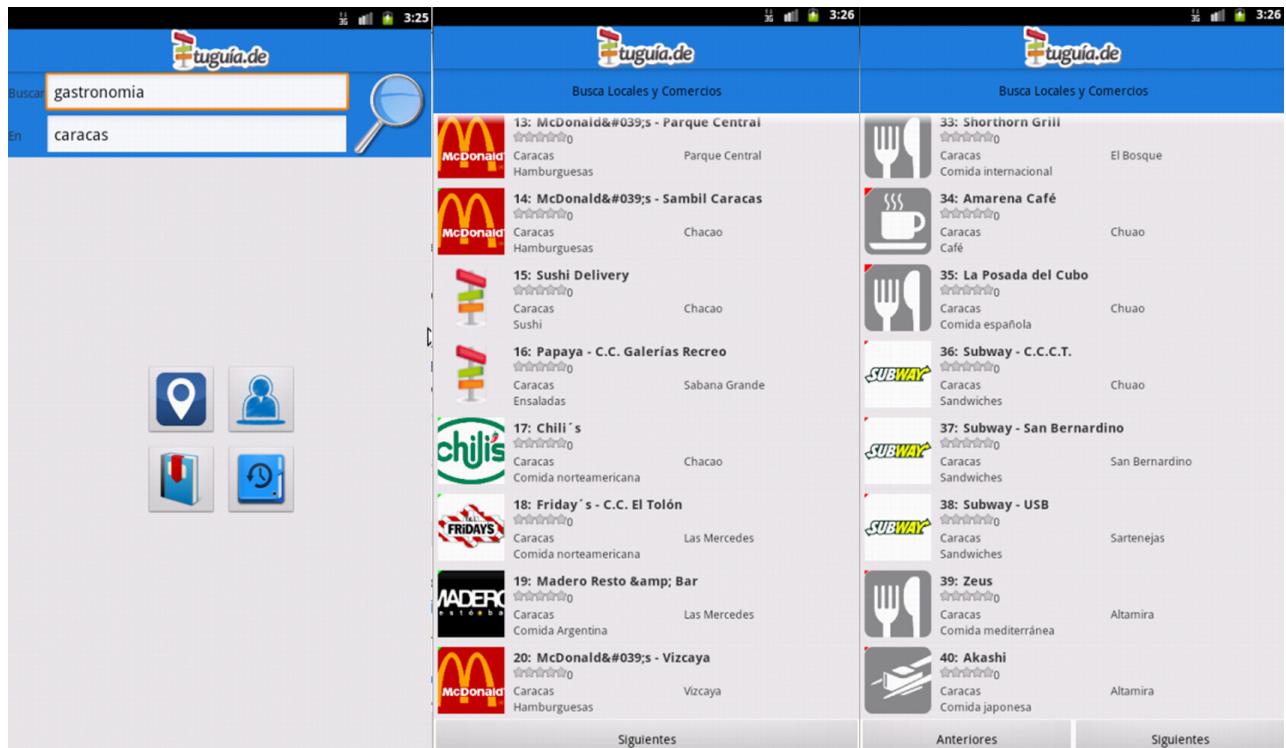


Figura 5.8: Colaje que muestra la funcionalidad de búsqueda y de paginado.

5.2.6. Sexta Iteración

Continuando con la lista de comportamientos, luego de buscar el local de su preferencia usando los diferentes criterios, la siguiente actividad que haría un usuario es ver la información detallada de un local, sus fotografías y los comentarios asociadas a éste. Esas son las tareas que se desarrollaron en esta iteración.

5.2.6.1. Objetivos Planteados

En la sexta iteración se establecieron los siguientes objetivos:

- Implementar una funcionalidad que permite ver información detallada de un local
- Desarrollar un mecanismo que permita mostrar una lista de comentarios.
- Construir la funcionalidad que muestre un comentario completo.

5.2.6.2. Resultados Alcanzados

Para la implementación de estas tres funcionalidades se siguió el flujo de desarrollo empleado en las iteraciones anteriores (figura 5.3) y en los tres casos se lograron desarrollar con éxito. A continuación se detallan los aspectos mas relevantes del desarrollo.

En la vista detallada de un local observada en la figura 5.9, se muestra la siguiente información: nombre, puntuación, número de comentarios, las categorías a las que pertenece, un botón con el número de teléfono que funciona como un acceso directo, al presionarlo lleva directamente a la aplicación usada para hacer llamadas marcando este número y la dirección. Ademas en este *layout* se muestra una lista de cinco comentarios realizados sobre el local y un botón de ver todos los comentarios, que lleva a otra pantalla donde se muestra la lista completa.

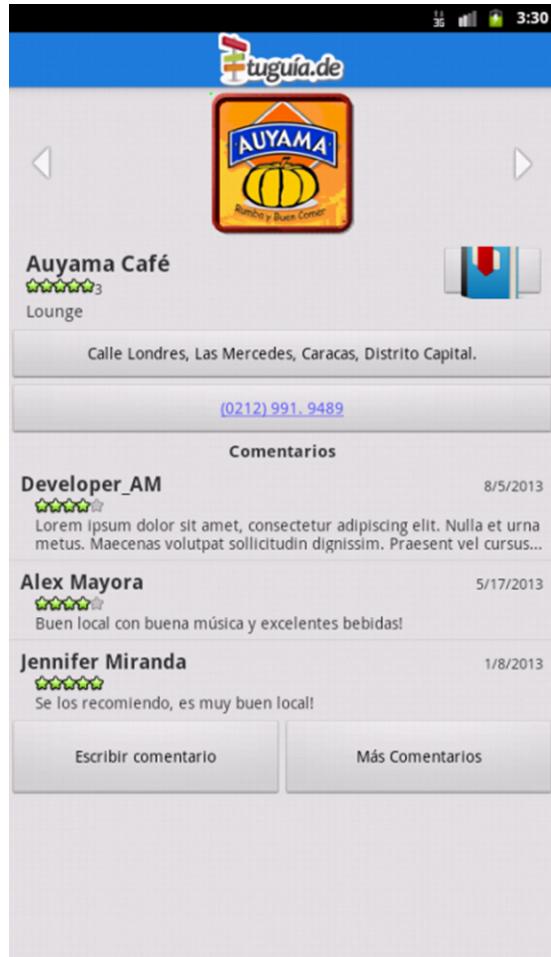


Figura 5.9: Vista detallada de un local en la aplicación móvil.

En Tugua.de no existe un límite para la longitud de la reseña de un usuario, por lo que en las listas de comentarios, se acotó el texto de la descripción a dos líneas. Si se presiona sobre un comentario, se va a una vista que muestra el contenido completo tal y como es mostrado en la figura 5.10.



Figura 5.10: Vista de un comentario completo en la aplicación móvil.

5.2.7. Séptima Iteración

En esta iteración se realizó un desarrollo orientado al manejo de las bases de datos internas de la aplicación, con el fin de registrar los usuarios y los locales vistos. De esta manera poder ofrecer un registro de los locales vistos recientemente por el usuario y la posibilidad de almacenar los locales favoritos, además de un manejo de sesión para el usuario en la aplicación.

5.2.7.1. Objetivos Planteados

A continuación se enumeran los objetivos establecidos:

- Crear el modelo de datos necesario para almacenar locales y usuarios en una base de datos de la aplicación
- Construir la funcionalidad que permita almacenar automáticamente los locales recientemente vistos por el usuario.
- Implementar la funcionalidad para que los usuarios pueden almacenar sus locales favoritos.
- Desarrollar un manejo de sesión para que una instancia de la aplicación pueda ser usada por múltiples usuarios.

5.2.7.2. Resultados Alcanzados

Gracias a la base de datos SQLite embebida dentro del sistema operativo Android, se disponía de todos los elementos necesarios para almacenar los locales. Se tomó la decisión de crear dos tablas independientes, una para almacenar los locales vistos recientemente y la segunda para los favoritos del usuario, a pesar que ambas manejan los mismos atributos.

Esta decisión fue tomada para facilitar el manejo de las operaciones sobre las tablas, ya que a pesar de que manejan la misma información el comportamiento es distinto. Con esta elección se buscaba también minimizar el tamaño de la base de datos. En una sola tabla se hubiese requerido una columna extra que identificase los locales recientes y los favoritos. Esta columna adicional a la larga podría representar mas espacio que el costo por mantener dos tablas separadas.

Cuando se visualiza la información detallada de un local, éste es agregado en la tabla creada para almacenar los locales recientes automáticamente. Ademas en este *layout* se dispuso

de un botón que permite que el usuario identifique un local como favorito. Para que tener acceso a esta funcionalidad los usuarios deben haber sido autenticados pasando exitosamente por el proceso de *login*. De esta forma todos los locales almacenados en la aplicación son marcados con un identificador único correspondiente al usuario autenticado que esté usando la aplicación en ese momento.

Otro aspecto importante de la gestión de sesión que se implementó fue el manejo de *cookies*³ ya que este mecanismo permite que no sólo se esté autenticado en la aplicación móvil, sino que también a través del envío de un identificador único de usuario en un *cookie*, se puede interactuar con el sitio web como si éste se hubiera autenticado.

5.2.8. Octava Iteración

La última iteración se enfocó en mejorar la eficiencia del manejo de recursos en la aplicación y de agregar la funcionalidad que permite realizar comentarios en Tuguia.de.

5.2.8.1. Objetivos Planteados

Los objetivos planteados en esta iteración fueron los siguientes:

- Diseñar e implementar la funcionalidad que permita subir comentarios asociados a un local de Tuguia.de.
- Implementar diferentes mecanismos que permitan que la aplicación realice el menor uso posible de la red de datos

5.2.8.2. Resultados Alcanzados

La funcionalidad correspondiente a los comentarios siguió el esquema planteado y fue desarrollada sin problema. Este comportamiento solo está disponible una vez que el usuario

³*Cookies*: pequeños fragmentos de información enviados al servidor en cada petición realizada

esta autentificado y se puede acceder a él, a través de la vista detallada de un local. Este mecanismo se muestra en la figura 5.11

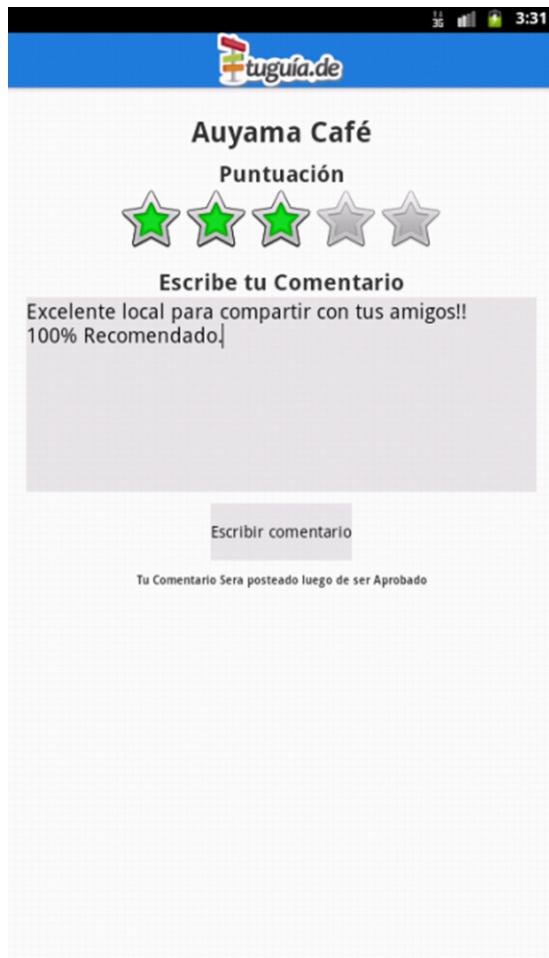


Figura 5.11: Interfaz para escribir comentarios en Tuguia.de.

A lo largo del desarrollo de la aplicación se idearon diversos mecanismos que permitieron optimizar no sólo el uso de la red, si no que también la memoria que consume la aplicación en ejecución.

La primera mejora es la integración de la biblioteca de manejo de imágenes *Picasso*, que ofrece la posibilidad de realizar diversas transformaciones de imágenes. Gracias a esto es posible reducir el tamaño de las imágenes recibidas, de tal forma que el espacio ocupado en

memoria sea el menor. Además provee un mecanismo de soporte a fallas, cada petición es intentada tres veces antes de descartar el recurso. Si el recurso es descartado, se coloca una imagen por defecto y de esta forma no se dejan espacios vacíos en la interfaz. Por último, permite implementar mecanismos de caché en la memoria del teléfono, de esta forma cuando se descarga una imagen, una copia es guardada en la memoria. Si por cualquier razón la aplicación intenta descargar nuevamente la misma imagen se retorna la copia que se encuentra en memoria. Como se mencionó anteriormente, las imágenes manejadas en Tuguia.de son de gran tamaño, la suma de los elementos brindados por ésta librería (ajuste de tamaño, caché y soporte de fallos) representó una cambió considerable en el funcionamiento de la aplicación.

Por otro lado se modificó también el comportamiento del paginado, buscando reducir el uso de la red de datos. Se implementó un mecanismo para guardar en archivos de texto las páginas de resultado. De esta forma al moverse entre las páginas, se verifica si existe un archivo que la contenga, antes de hacer una petición al API para obtenerla. Este enfoque no es solo mas eficiente sino que permite un desplazamiento más fluido dentro de la aplicación.

Como última mejora se editaron las respuestas del API para cada recurso eliminando todos los campos que no eran necesarios en la aplicación móvil. De esta forma en cada petición sólo se obtiene la información que es necesaria para mantener al mínimo la cantidad de información transferida por la red.

5.3. Resultados Generales y Estado Actual

El proyecto se logró completar en un 100% de lo que estaba establecido en el alcance incluso dispone de funcionalidades que aún no se encuentra disponibles en el sitio de web. A pesar de que cuando se visualiza la información de un local en la página web de Tuguia.de se muestra su ubicación sobre el mapa, esto no ocurre cuando se observa un local en la aplicación móvil. Esta funcionalidad quedó fuera del alcance de las pasantía debido a que se

dio prioridad al trabajo realizado con Drupal y Apache Solr que permitió realizar búsquedas más específicas por varios parámetros.

Este tipo de búsqueda aun no está disponible en el portal web de Tugua.de, sin embargo gracias a las configuraciones, módulos y extensiones desarrolladas durante la pasantía, se disponen todos los mecanismos necesarios para la integración de esta funcionalidad en el sitio web. Este trabajo será realizado por el equipo de desarrollo responsable a corto plazo, agregando así un valor adicional gracias a las pasantías.

Si bien una de las ventajas mas importantes de las metodologías basadas en pruebas, es que garantizan la elaboración de productos de calidad, la versión desarrollada se encuentra en fase beta y aunque es bastante estable, debe ser sometida a una serie pruebas mas extensas antes de considerar realizar su despliegue.

Capítulo 6

Conclusiones y Recomendaciones

El desarrollo de las pasantías permitió conocer diferentes aspectos relacionados con el ámbito empresarial y el desarrollo de proyectos de manera efectiva en éste ambiente. Dentro del aprendizaje adquirido resalta el benéfico del contacto directo y constante con la dirección general del proyecto, esto no sólo mejora la calidad del producto generado, sino que también permite aportar directamente al modelo de negocio y su crecimiento.

Ha sido un reto constante para Empresas TD ofrecer soluciones tecnológicas e innovadoras orientadas a mejorar la calidad del comercio electrónico en Venezuela. Este proyecto de pasantía representa la primera aplicación móvil de desarrollo nativo que se realiza en la empresa, por lo que significó la apertura a una nueva forma de acercamiento a sus clientes y la garantía de mantenerse al día con las nuevas tendencias mundiales.

Las metodologías aplicadas al igual que otros procesos de desarrollo ágil dan prioridad a satisfacer al cliente mediante la entrega continua de funcionalidad y la capacidad de que los requisitos cambien, incluso en etapas tardías del desarrollo. Adaptarse a esta estructura de trabajo tan dinámica es la única manera de vivir al día con los avances tecnológicos y las tendencias globales. Las empresas que logran llevar estos dos elementos de la mano, son las que tienen mayor éxito en nuestros días.

Este dinamismo al que estuvo sujeto el proyecto afectó también muy positivamente la experiencia y el aprendizaje adquirido. Si bien el objetivo principal de la pasantía era el desarrollo de una aplicación Android, también se pudo interactuar con otras tecnologías como Drupal y Apache Solr, lo que representó la adquisición de un conocimiento valioso que no estaba planificado.

El proyecto de pasantía cumple con los objetivos planteados y convierte a Tuguia.de en servicio disponible a los usuarios 24 horas durante los 7 de la semanas, en cualquier lugar simplemente usando un teléfono celular Android, lo que representa una avance que puede llegar a marcar la diferencia entre varias empresas ofreciendo el mismo servicio.

A continuación se mencionan algunas recomendaciones que pueden enriquecer el proceso de desarrollo y producir mejoras en la aplicación móvil de Tuguia.de.

- No se realizaron pruebas exhaustivas en el rendimiento de la versión final, por lo que éstas deben ser realizadas cuando antes de que la aplicación sea lanzada al mercado.
- Es necesario integrar a la vista de cada local su ubicación en el mapa. Para ello existen varias alternativas, entre las cuales resalta Google Maps, ya que mediante la integración de un conjunto de bibliotecas disponibles en el sitio oficial, se puede acceder de manera muy sencilla a su funcionalidad desde el entorno Android. Además esta solución ofrece un valor agregado al brindar la capacidad de mostrar en el mapa la forma de llegar desde la ubicación hasta el local.
- Se podría mejorar la experiencia de búsqueda a través de la integración de un mecanismo para filtrar los resultados, no solo por ubicación sino por los atributos asociados a un local. Para un usuario no es suficiente sólo saber cuáles son los locales que existen en un determinado lugar, también sería útil saber por ejemplo cuáles tienen estacionamiento y aceptan tarjeta de crédito. Dado la configuración actual de Apache Solr

esto es posible, sin embargo es necesario realizar un desarrollo en Drupal para hacer que esta funcionalidad este disponible a través del API y luego llevarla a la aplicación móvil.

- Actualmente la interfaz de la aplicación móvil, no se ajusta a la imagen corporativa de Tugua.de, es necesario realizar diseños para cada una de las pantallas y luego llevarlos a los *layouts* de la aplicación.

Bibliografía

- [AAT] Arquitectura de android. <http://columna80.wordpress.com/2011/02/17/arquitectura-de-android/>, consultado el 22 de septiembre de 2013.
- [AND] Android ndk. <http://developer.android.com/tools/sdk/ndk/index.html>, consultado el 22 de septiembre de 2013.
- [APS] Apache solr. <http://lucene.apache.org/solr/>, consultado el 22 de septiembre de 2013.
- [ASD] Get the android sdk. <http://developer.android.com/sdk/index.html>, consultado el 22 de septiembre de 2013.
- [Cen05] Miquel Centelles. Taxonomías para la categorización y la organización de la información en sitios web. *Universitat de Barcelona*, 2005.
- [CTG] El cine restaurant. <http://tugua.de/local/el-cine-restaurant#.UjpGYVFDtKY>, consultado el 22 de septiembre de 2013.
- [DGT] By 2017 there will almost be as many smartphones as literate adults. <http://www.digitaltrends.com/mobile/2017-literate-adults-smartphones/>, consultado el 22 de septiembre de 2013.

- [DLC] Definición de local. <http://www.definicionabc.com/economia/local.php>, consultado el 22 de septiembre de 2013.
- [DPB09] Larman C Deemer P, Benefield G and Vodde B. *The Scrum Primer*. Agile-Spain, 1 edition, 2009.
- [IBD] Introducing bdd. <http://dannorth.net/introducing-bdd/>, consultado el 22 de septiembre de 2013.
- [JSO] Json. <http://www.json.org/>, consultado el 22 de septiembre de 2013.
- [JUN] Junit. <http://junit.org/>, consultado el 22 de septiembre de 2013.
- [LTG] Lealtag. <http://www.lealtag.com/>, consultado el 22 de septiembre de 2013.
- [NTD] Nielsen tops of 2012: Digital. <http://www.nielsen.com/us/en/newswire/2012/nielsen-tops-of-2012-digital.html>, consultado el 22 de septiembre de 2013.
- [PCM] Definition of:smartphone. <http://www.pcmag.com/encyclopedia/term/51537/smartphone>, consultado el 22 de septiembre de 2013.
- [SCC] Sucucho. <http://sucucho.com/ve/sucucho/nosotros>, consultado el 22 de septiembre de 2013.
- [TBS] Tasks and back stack. <http://developer.android.com/guide/components/tasks-and-back-stack.html>, consultado el 22 de septiembre de 2013.
- [TDC] Tudescuenton.com. <http://www.tudescuenton.com/about/us.php>, consultado el 22 de septiembre de 2013.
- [TGD] Tugua.de. <http://tugua.de/preguntas-frecuentes>, consultado el 22 de septiembre de 2013.

[TMM] The mobile movement. <http://www.google.com/think/research-studies/the-mobile-movement.html>, consultado el 22 de septiembre de 2013.

[TV10] T Tomlinson and J VanDyk. *Pro Drupal 7 Development*. Apress, 3 edition, 2010.