

# FACHADASAPP

17 de Octubre de 2014

Descripción de la metodología de trabajo

Oscar Pulido Rodríguez

Diego Martínez Botero

## Tabla de Contenidos

<b>1.</b>	<b>Introducción .....</b>	<b>3</b>
1.1	<i>Propósito de este documento .....</i>	<i>3</i>
1.2	<i>Alcance.....</i>	<i>3</i>
<b>2.</b>	<b>Descripción Del proceso.....</b>	<b>4</b>
2.1	<i>En qué se ha trabajado .....</i>	<i>4</i>
2.2	<i>Avances.....</i>	<i>4</i>
<b>3.</b>	<b>Requerimientos .....</b>	<b>5</b>
3.1	<i>Codigo fuente.....</i>	<i>5</i>
3.2	<i>Librerías utilizadas .....</i>	<i>5</i>
3.3	<i>Disponibilidad .....</i>	<i>6</i>
<b>4.</b>	<b>Comparativo .....</b>	<b>7</b>
4.1	<i>Planeación vs Avance.....</i>	<i>7</i>
<b>5.</b>	<b>Problemas Presentados .....</b>	<b>8</b>
2.1	<i>Problemas encontrados .....</i>	<i>8</i>
2.2	<i>Atraso.....</i>	<i>8</i>
<b>6.</b>	<b>Planeacion .....</b>	<b>9</b>
2.1	<i>¿Qué falta? .....</i>	<i>9</i>
2.2	<i>Plan para continuar y completar el proyecto .....</i>	<i>9</i>

## **1. Introducción**

Este documento describe el proceso y la planificación de trabajo proyectado en la metodología Scrum para la clase de Ingeniería de Software II, segundo semestre de 2014, para la gestión del desarrollo del proyecto FachadasApp.

### **1.1 Propósito de este documento**

Facilitar la información de referencia necesaria a las personas implicadas en el desarrollo del sistema FachadasApp.

### **1.2 Alcance**

Personas y procedimientos implicados en el desarrollo del sistema FachadasApp.

## **2. Descripción Del Proceso**

Aquí se describe el resultado del proceso hasta ahora, así como lo que se ha logrado hasta el momento y lo que falta para concluir el proyecto.

### **2.1 En qué se ha trabajado**

El proyecto para dispositivos Android, ha sido implementado en un sistema operativo Windows 8.1 utilizando Java y el SDK de Android. Ha sido desarrollado en Eclipse versión 23.0.2.1259578, utilizando el SDK de Android revisión 23.0.2.

El servidor ha sido implementado en un sistema operativo Linux Fedora y se ha trabajado en python, utilizando un servicio RESTful y una base de datos no relacional MongoDB.

### **2.2 Avances**

Se ha trabajado las piezas por separado como servicios independientes y luego se han armado para completar su funcionalidad. Las funcionalidades por separado del sistema que actualmente se han logrado son:

- El sistema es capaz de conectarse al servidor y enviarle la información para ser guardada en la base de datos.
- El sistema es capaz de solicitar la información al servidor, recibirla y mostrarla en la aplicación.
- El sistema es capaz de organizar la información y mostrarla en una lista, con imagen, nombre y descripción, acorde al diseño planteado para la aplicación.
- El sistema es capaz de permitirle al usuario tomar una foto de una fachada y agregarle su nombre y descripción y almacenar la información para enviarla al servidor.

Por aparte se ha trabajado el servidor, el cual responde acorde a las peticiones de la aplicación y es capaz de:

- Recibir la información, convertirla a formato Json y guardarla en la base de datos de la manera acordada.
- Recibir la petición de una fachada y devolver la información almacenada en la base de datos de esa fachada específica.

### 3. Requerimientos

Se listan los requerimientos necesarios para que la aplicación pueda ser ejecutada en cualquier máquina e instalar la aplicación en cualquier celular. El celular requiere contar con una cámara integrada para poder tomar las fotos de las fachadas y acceso a internet para conectarse con el servidor.

#### 3.1 Código fuente

El código fuente de la aplicación se encuentra disponible en Github en el repositorio de nombre FachadasApp.

<https://github.com/diegomtz/FachadasApp>

<https://github.com/oscarpr/FachadasAPP>

#### 3.2 Librerías utilizadas

```
import java.io.BufferedReader;
import java.io.IOException;
import java.io.InputStream;
import java.io.InputStreamReader;
import java.util.ArrayList;
import java.util.List;

import org.apache.http.HttpEntity;
import org.apache.http.HttpResponse;
import org.apache.http.NameValuePair;
import org.apache.http.client.HttpClient;
import org.apache.http.client.entity.UrlEncodedFormEntity;
import org.apache.http.client.methods.HttpGet;
import org.apache.http.client.methods.HttpPost;
import org.apache.http.entity.BufferedHttpEntity;
import org.apache.http.impl.client.DefaultHttpClient;
import org.apache.http.message.BasicNameValuePair;
import org.apache.http.util.EntityUtils;

import android.os.Bundle;
import android.os.StrictMode;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Toast;

import java.net.HttpURLConnection;
import java.net.URL;
```

### **3.3 Disponibilidad**

El código en Github es de acceso público y puede ser descargado y compilado utilizando Eclipse y Android SDK. El servidor no ha sido implementado como servicio por lo que toca iniciarlo manualmente y no se encuentra corriendo permanentemente.

#### **4. Comparativo**

Se compara el avance logrado hasta el momento con el presentado en el plan del proyecto

##### **4.1 Planeación Vs Avance**

En nuestra planeación del proyecto acordamos que las semanas que transcurrimos, serían de desarrollo verificación y corrección en un ciclo continuo. Sin embargo para esta entrega, esperábamos contar con las funcionalidades que hemos logrado hasta el momento pero funcionando de manera conjunta y no por separado como lo hace hasta ahora. Aunque si hay algunos servicios funcionando en conjunto, todavía no funciona en su totalidad y no se logra apreciar el funcionamiento óptimo de la aplicación como esperábamos.

## **5. Problemas presentados**

En todo proyecto existen unos riesgos establecidos y algunos imprevistos, que se pueden presentar durante el desarrollo del mismo. Causas que conllevan muy probablemente al atraso de este proceso.

### **5.1 Problemas encontrados**

- Problemas con los computadores utilizados: Uno de los computadores utilizados siempre presentó problemas para la ejecución del entorno de desarrollo (Eclipse y Android Studio). Lo cual incurrió en un trabajo más lento y muchas veces unilateral en el desarrollo de la aplicación.
- Falta de Drivers y conocimiento: Uno de los principales atrasos era la falta de capacidad para probar la aplicación en el celular, ya que el pc en el que se estaba desarrollando no reconocía el celular. Esto se solucionó instalando los drivers USB de Android que no se encontraban instalados en este pc y no nos habíamos percatado de la falla.
- Descuido y falta de permisos: Otro atraso importante que nos retrasó el desarrollo de la aplicación consistió en realizar la conexión con el servidor desde la aplicación, ya que en el módulo individual creado inicialmente funcionaba, y al momento de conectarlo con el sistema conjunto no. Este problema consistió en el descuido al momento de crear el nuevo proyecto, de no implementar los permisos necesarios en el archivo manifiesto de la aplicación, para conectarse a internet.
- Descuido y falta de tiempo: Por supuesto en nuestra vida diaria el tiempo siempre es un riesgo inminente en el desarrollo de cualquier proyecto y mucho más aún si no se administra y organiza debidamente. En nuestro caso nos ha faltado un poco de organización para trabajar y más aún cuando sabemos que los dos desarrolladores del proyecto contamos con diversas actividades y horarios disponibles. Nada que no se pueda solucionar con un poquito más de organización y disposición.

### **5.2 Atraso**

A pesar de que el atraso no es mucho a nuestro parecer, pues aun creemos que podemos concluir el proyecto en el plazo estipulado y que contamos con casi todo lo estipulado para esta entrega, esto no se puede apreciar de una manera evidente. Sin embargo los problemas encontrados han sido la causa de este atraso.



## **6. Planeación**

En esta etapa del proyecto nos hemos replanteado la organización y hemos reestructurado el plan para poder cumplir con la entrega completa del proyecto en el tiempo estimado.

### **6.1 ¿Qué falta?**

Algunas partes no las hemos considerado fundamentales en el desarrollo del proceso hasta ahora, puesto que no eran imprescindibles para el producto mínimo viable esperado. Estas partes son tales como el login de usuarios, la ubicación en el mapa y no nos preocupamos por la presentación visual de la aplicación. Sin embargo si serán tenidas en cuenta en su gran mayoría para esta nueva etapa y la entrega final.

Las partes fundamentales que han hecho falta para esta entrega son:

- El sistema funciona correctamente en conjunto con todos los módulos desarrollados de manera independiente.
- El sistema sea capaz de enviar las fotos tomadas al servidor para ser almacenadas.
- Algunas correcciones de funcionalidad en cambios realizados.

### **6.2 Plan para continuar y completar el proyecto**

Para poder concluir satisfactoriamente el proyecto es necesario principalmente mejorar la organización del tiempo. Para esto hemos decidido cumplir con las reuniones diarias estrictamente, dedicándole entre 5 y 15 minutos a la planeación del día y de la semana.

Con esto pretendemos lograr una pequeña meta o corrección diaria mínimo y no dejar que se acumulen para el final de la semana.

De esta misma manera pretendemos gestionar mejor el tiempo de trabajo y avanzar mucho más rápido en lo que hace falta para completar el producto mínimo viable y luego poder proceder con los siguientes aspectos que hagan falta.