**WhatsCamp Program Manual**

|  |  |  |
| --- | --- | --- |
| **Date** | **Name** | **Description** |
| 2 November 2017 | Joseph | Change base format |
| 4 November 2017 | Jessica | Add Database detail |
| 15 November 2017 | Joseph | Add detail description and API description |
| 23 November 2017 | Joseph | Add figure and link reference list |
| 28 November 2017 | Clarissa | Add Flowchart and Workflow |
| 1 Desember 2017 | Jessica | Edit Database detail |

**Figure List**

[Figure 1 String.xml 5](#_Toc499660354)

[Figure 2 ic\_launcer.png 6](#_Toc499660355)

[Figure 3 colors.xml 7](#_Toc499660356)

[Figure 4 style.xml 7](#_Toc499660357)

[Figure 5 Database Configuration 9](#_Toc499660358)

[Figure 6 WhatsCamp PhpMyAdmin 12](#_Toc499660359)

[Figure 7 Facebook API 16](#_Toc499660360)

[Figure 8 Google API 17](#_Toc499660361)

[Figure 9 API in PHP file 17](#_Toc499660362)

[Figure 10 Config.java 19](#_Toc499660363)

[Figure 11 Build.gradle 21](#_Toc499660364)

[Figure 12 Workflow 22](#_Toc499660365)

[Figure 13 Major Flowchart 22](#_Toc499660366)

[Figure 14 Main Menu 23](#_Toc499660367)

[Figure 15 Search Event 23](#_Toc499660368)

[Figure 16 Add Event 24](#_Toc499660369)

[Figure 17 Add Event (Database) 24](#_Toc499660370)

[Figure 18 Join Event 25](#_Toc499660371)

[Figure 19 Join Event (Database) 25](#_Toc499660372)

[Figure 20 Edit Event 26](#_Toc499660373)

[Figure 21 Edit Event (Database) 26](#_Toc499660374)

[Figure 22 Delete Event 27](#_Toc499660375)

[Figure 23 Delete Event (Database) 27](#_Toc499660376)

[Figure 24 Find Ticket 28](#_Toc499660377)

[Figure 25 Favorite Event 28](#_Toc499660378)

[Figure 26 Favorite Event (Database) 29](#_Toc499660379)

[Figure 27 Share Event 29](#_Toc499660380)

[Figure 28 Give Comment 30](#_Toc499660381)

[Figure 29 Give Comment (Database) 30](#_Toc499660382)

[Figure 30 Login 31](#_Toc499660383)

[Figure 31 Logout 31](#_Toc499660384)

**Android SDK & Android Studio**

This chapter describes how to install Android SDK and Android Studio. Although the project can be built without Android Studio, using Gradle and Android SDK, it much better if Android Studio is used.

1. Install [Java JDK] (http://www.oracle.com/technetwork/java/javase/downloads/index.html)
2. Install [Android SDK](https://developer.android.com/sdk/index.html)
3. Run Android SDK Manager and [download necessary SDK packages] (https://developer.android.com/sdk/installing/adding-packages.html), make sure that you have installed Android SDK Tools, Android SDK Platform-tools, Android SDK Build-tools, Android Support Repository, Android Support Library and Google Play services
4. Install [Android Studio](https://developer.android.com/sdk/index.html)
5. Now you should be able to open/edit the Android project and build APK

**Project template**

This project used a template taken from program source code EventFinder ver.1.0 (http://www.codelist.cc/mobile/2271-event-finder-full-android-application-v10.html) as the base template. The program get modification such features and logo change and replace icons to fulfill the project requirement.

**Project structure**

Project has the following structure (directories are marked by square braces):

* [doc] - documentation
* [php] - contains phpfiles
* [mysql] – contain .mysql data for import to phpmyadmin
* [gradle]
* [gradle]/[wrapper] - Gradle Wrapper
* [apps] - main module
* [apps]/[libs] - contains 3rd party libraries
* [apps]/[src] - contains source code
* [apps]/[src]/[main]
* [apps]/[src]/[main]/[java] - java sources
* [apps]/[src]/[main]/[res] - xml resources, drawables
* [apps]/[src]/[main]/AndroidManifest.xml - manifest file
* [apps]/build.gradle - main build script
* [apps]/proguard-rules.pro - Proguard config (not used)
* build.gradle - parent build script
* gradle.properties - build script properties containing path to keystore
* gradlew - Gradle Wrapper (Unix)
* gradlew.bat - Gradle Wrapper (Windows)
* README.md - readme file
* settings.gradle - build settings containing list of modules

Java packages:

* com - contains application class and main config class
* com.application - contains content for Google network
* com.config - contains configuration and final variables
* com.db - contains database helper and queries
* com.libraries – extra class for interface and features
* com.models - database models representing SQL tables
* com.projects – contain activities needed for feature
* com.projects.activites – main features activity
* com.projects.events – contain activity for show event
* com.projects.fragments – contain fragment of activity
* com.projects.posts – activity for adding comments
* com.projects.users – contain activity login
* com.projects.whatscamp – contain MainActivity and SplashActivity

**Configuration**

This chapter describes how to configure the project to be ready for running. All these steps are important!

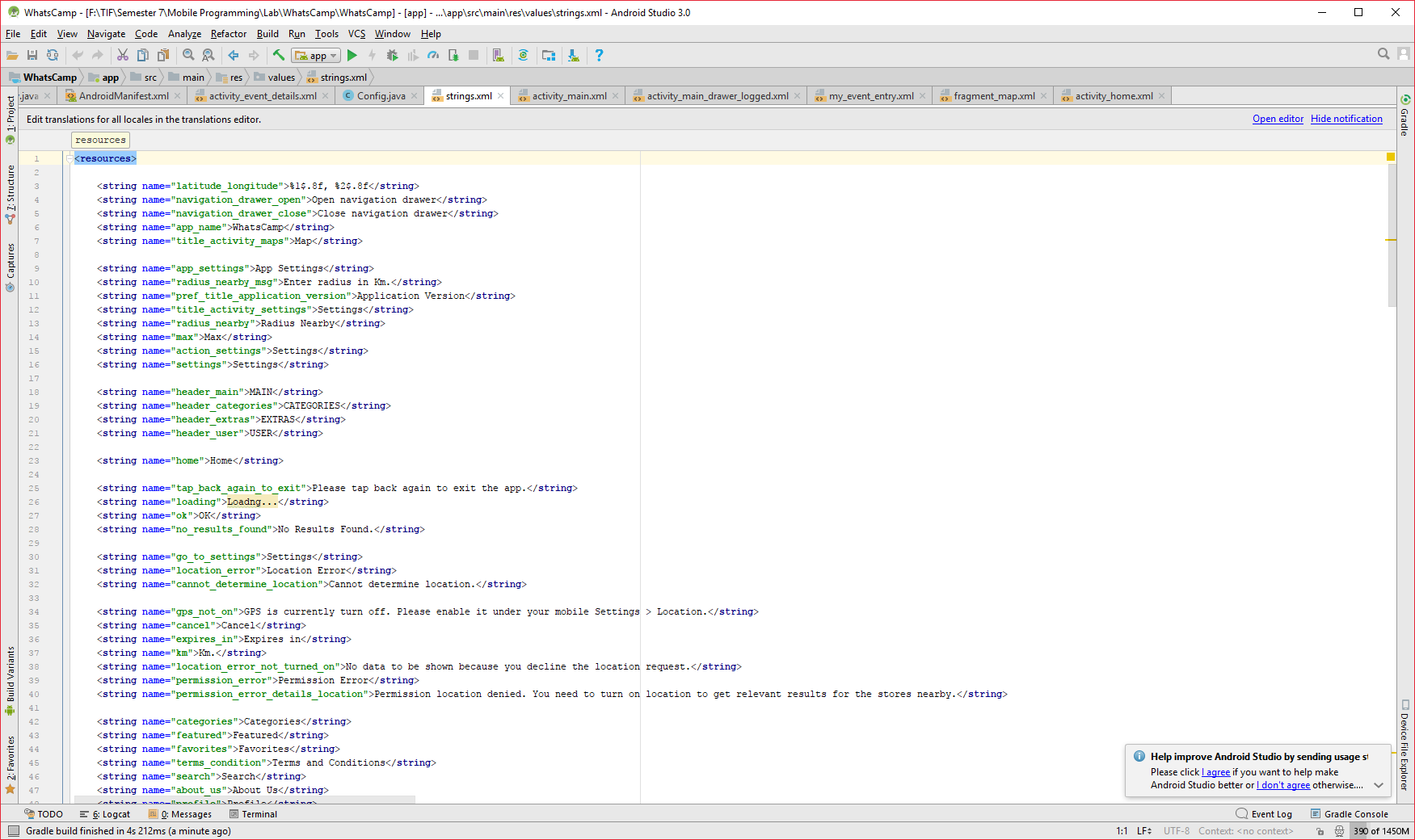
* + - 1. **Import**

Unzip the package and import/open the project in Android Studio. Choose "Import project" on Quick Start screen and select "whatscamp" directory.

* + - 1. **Rename package name**

1. In Project Pane, click one the gear icon
2. Un-check the “Compact Empty Middle Packages” option
3. Select the directory you want to change
4. Right Click -> Select Refactor -> Click on Rename
5. In the Pop-up dialog, click “Rename Package” instead of “Rename Driectory”
6. Enter new name and click “Refactor”
7. Click “Do Refactor in the bottom”
8. Note: When renaming “com” in Android Studio, it might give a warning, Select “Rename All”
   * + 1. **Rename application name**

Open \_mobile/src/main/res/values/strings.xml\_ and change "WhatsCamp" to your own name. Change app\_name, email\_subject and email\_subject\_company



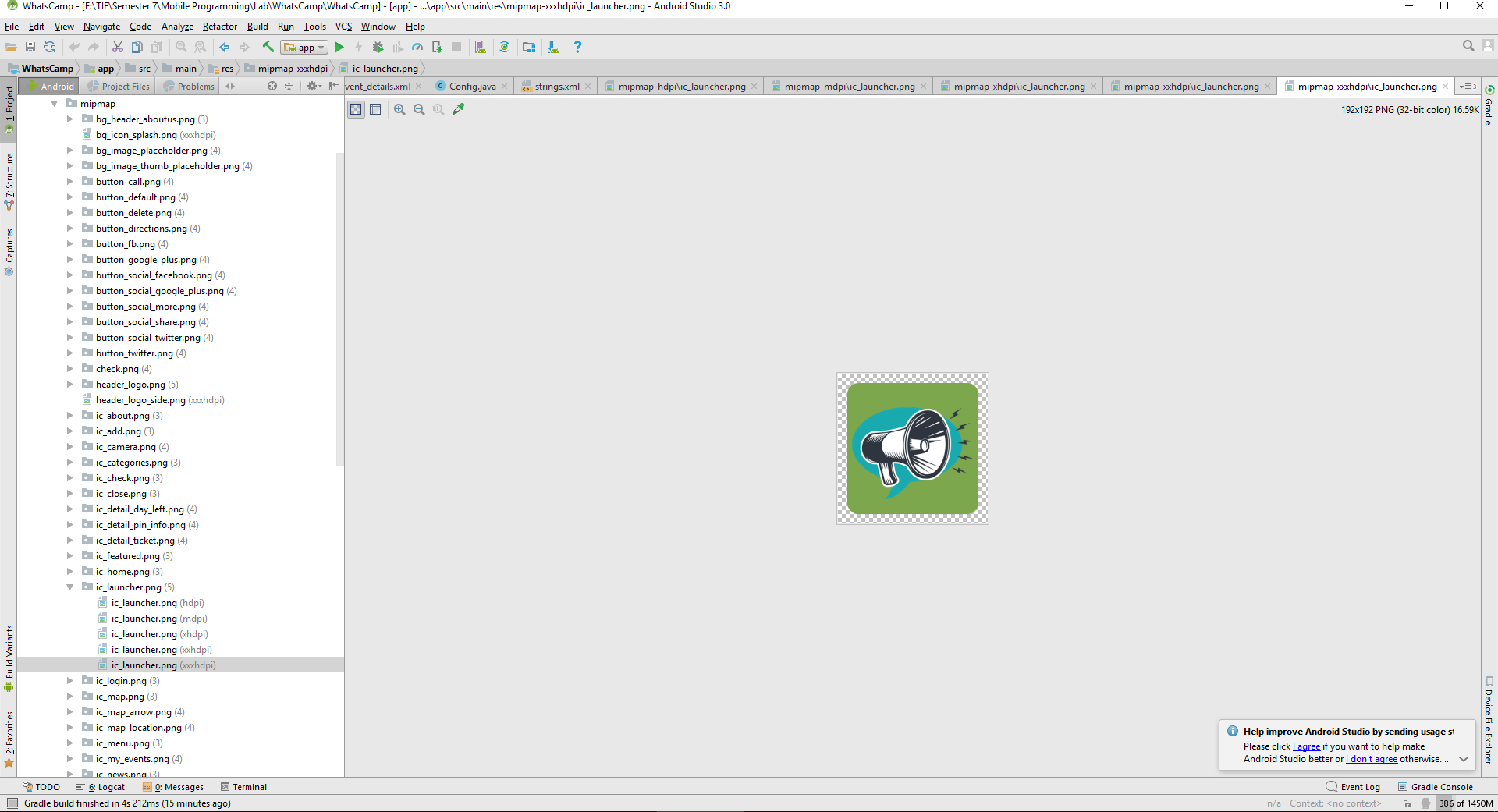
**Figure 1** String.xml

**Sumber:** app/src/main/res/values/strings.xml

* + - 1. **Create launcher icon**

Right click on app/src/main/res directory -> New -> Image Asset -> Asset type Launcher Icons, Resource name "ic\_launcher", create the icon -> Next -> Finish.

You can also change the icon replacing ic\_launcher.png file in mipmap-mdpi, mipmap-hdpi, mipmap-xhdpi, mipmap-xxhdpi, mipmap-xxxhdpi directories. See [Android Cheatsheet for Graphic Designers](http://petrnohejl.github.io/Android-Cheatsheet-For-Graphic-Designers/#screen-densities-and-icon-dimensions) for correct launcher icon dimensions.

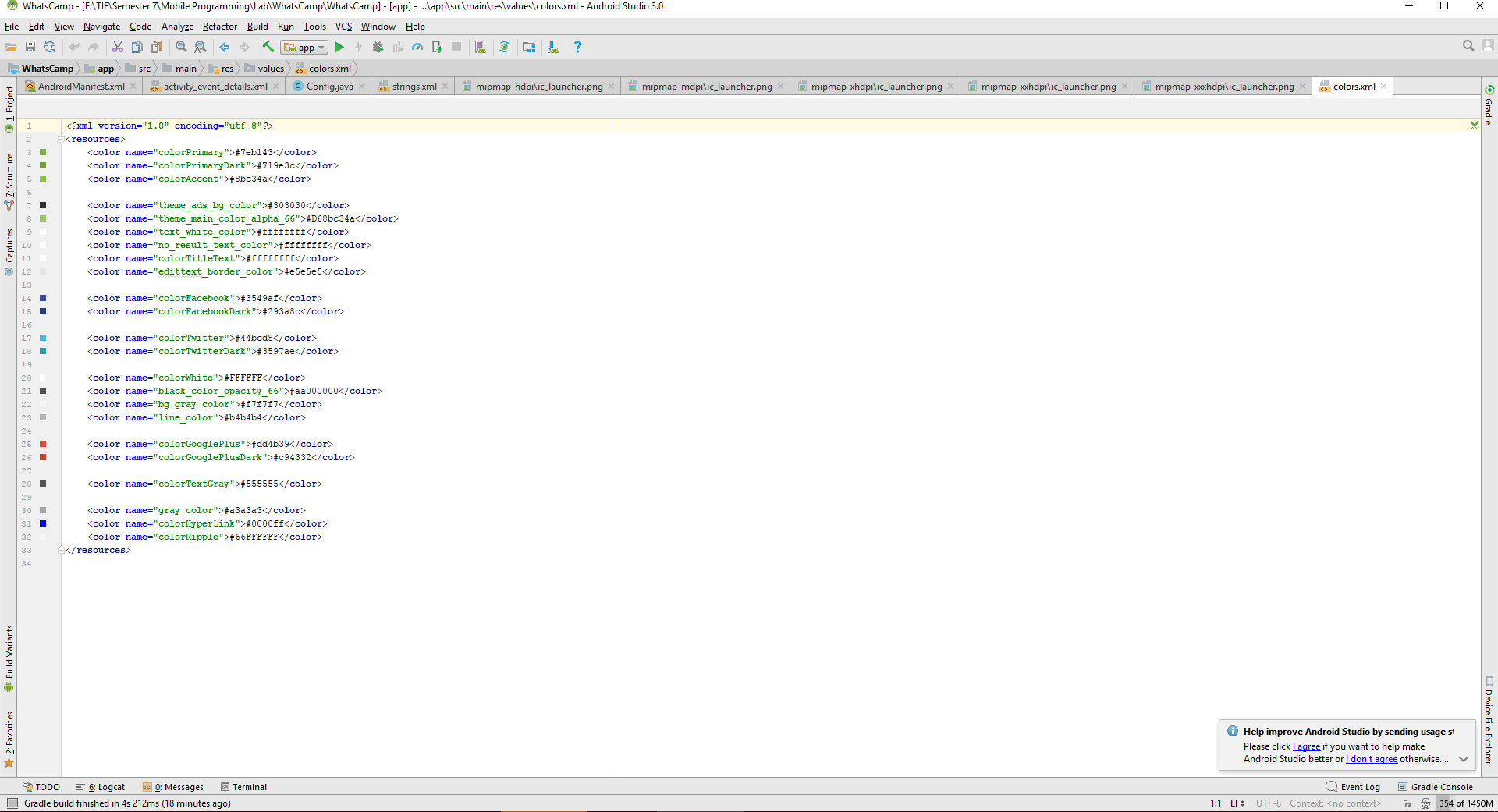
Another possibility is to create launcher icons using [Android Asset Studio] (http://romannurik.github.io/AndroidAssetStudio/icons-launcher.htm)

**Figure 2** ic\_launcer.png

**Sumber:** app/src/main/res/mipmap-xxxhdpi/ic\_launcher.png

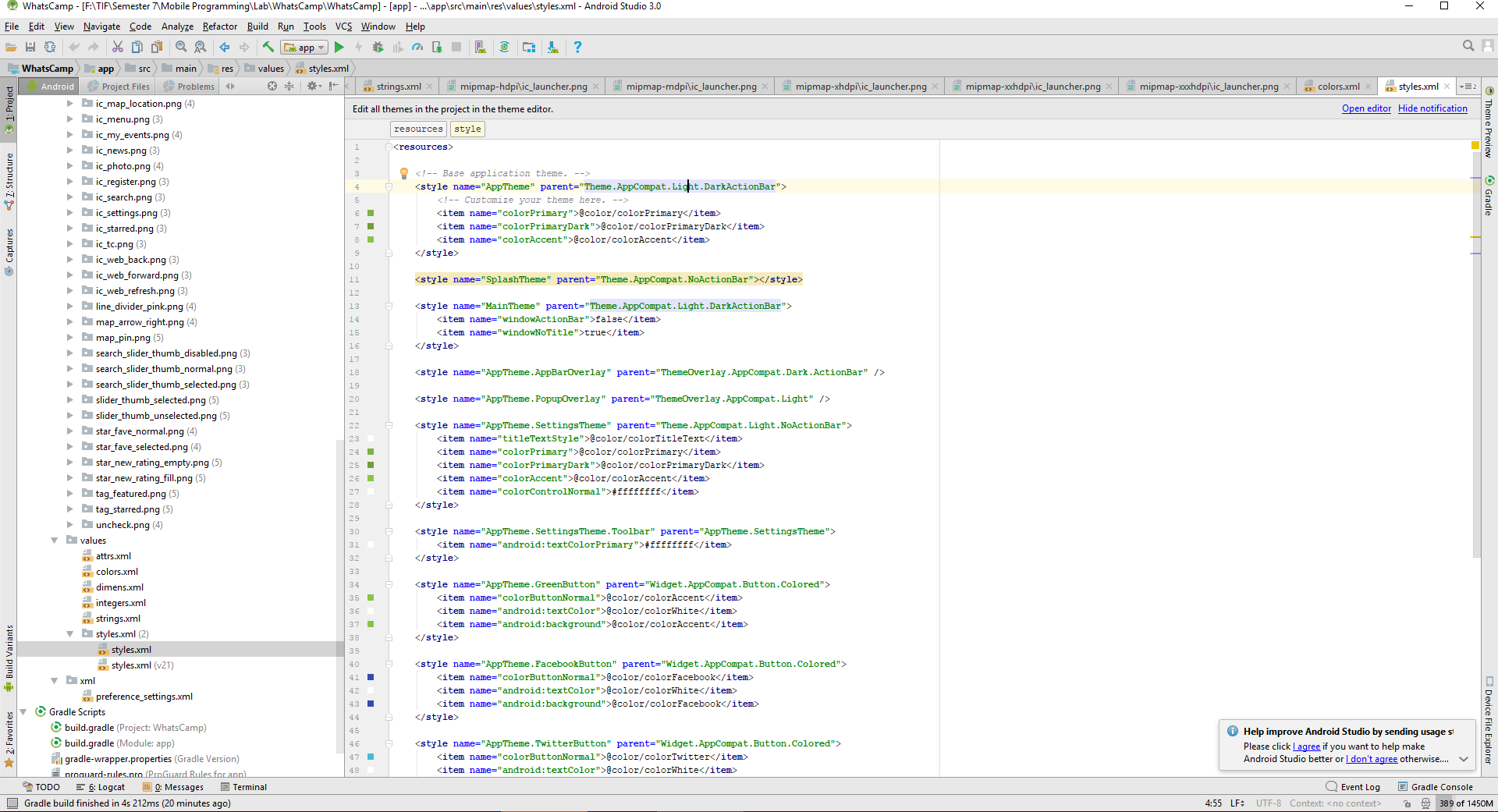
1. **Change color theme**

Open \_mobile /src/main/res/values/style.xml and change items in style named “AppTheme”. You can see and change lists of colours that you can use in /src/main/res/values/colors.



**Figure 3** colors.xml

**Sumber:** app/src/main/res/values/colors.xml



**Figure 4** style.xml

**Sumber:** app/src/main/res/values/style.xml

Example:

<**item name="colorPrimary"**>@color/colorPrimary</**item**>

Change into:

<**item name="colorPrimary"**>@color/colorPrimaryDark</**item**>

1. **Prepare database**

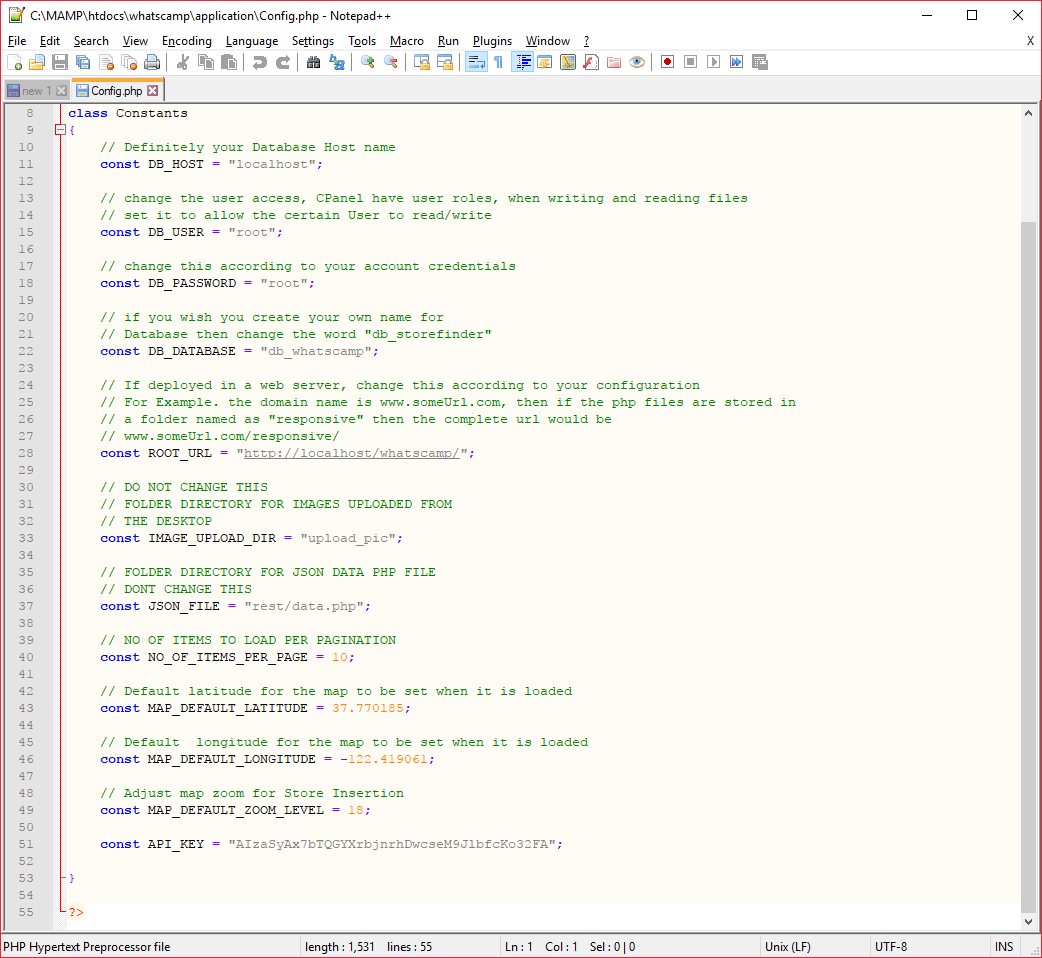
Before importing the database, we need to install software used for hosting the database server. For this application, we use MAMP [MAMP](https://www.mamp.info/en/). After installing MAMP, copy folder “whatscamp” from directory “php” to /MAMP/htdocs.

After copy the php files, run MAMP server and open /localhost/MAMP/phpmyadmin and create database db\_whatscamp or db\_companyname. In the database which just created, import sql file from directory “sql”. The database will create all tables needed for the program to work and the php files used for connecting the database and the program. Do not forget to change the ROOT\_URL in Application/config.php with format http://localhost/[your\_filenames\_in\_htdocs]/.

In the Android project, open config/Config.java and change the BASE\_URL with format http://[your\_ipaddress]/[your\_filenames\_in\_htdocs]/ and change DB\_NAME in db/Dbhelper.java with your database name.

If you want to change the data tables name, don’t forget to change all data table name inside all php files in /htdocs/[your\_filename\_in\_htdocs]/controller.

For dabase connection, we use “root” as user and “root” as password with “localhost” as the host of database. You can see the configuration in the PHP file inside MAMP/htdocs/whatscamp/application/Config.php if you already put the PHP file inside MAMP



**Figure 5** Database Configuration

**Sumber:** app/src/main/java/com/config/Config.java

Data is stored in local PhpMyAdmin database. Data is accessed from php script that is called from the application, so that there are several php files that stored in shared folder (htdocs). Database is in SQLite 3.0 format and has the following structure (SQL script):

**SQL:**

* + - 1. Table structure for table `tbl\_whatscamp\_attendees`

CREATE TABLE `tbl\_whatscamp\_attendees` (

`attendee\_id` int(11) NOT NULL,

`event\_id` int(11) NOT NULL,

`is\_going` int(11) NOT NULL DEFAULT '-1',

`is\_interested` int(11) NOT NULL DEFAULT '-1',

`is\_invited` int(11) NOT NULL DEFAULT '-1',

`user\_id` int(11) NOT NULL DEFAULT '-1',

`updated\_at` int(11) NOT NULL DEFAULT '0',

`created\_at` int(11) NOT NULL DEFAULT '0',

`is\_deleted` int(11) NOT NULL DEFAULT '0'

);

* + - 1. Table structure for table `tbl\_whatscamp\_categories`

CREATE TABLE `tbl\_whatscamp\_categories` (

`category\_id` int(11) NOT NULL,

`category` text NOT NULL,

`category\_icon` text NOT NULL,

`created\_at` int(11) NOT NULL DEFAULT '0',

`updated\_at` int(11) NOT NULL DEFAULT '0',

`is\_deleted` int(11) NOT NULL DEFAULT '0'

) ;

* + - 1. Table structure for table `tbl\_whatscamp\_events`

CREATE TABLE `tbl\_whatscamp\_events` (

`event\_id` int(11) NOT NULL,

`address` text NOT NULL,

`event\_desc` text NOT NULL,

`gmt\_date\_set` datetime NOT NULL,

`is\_ticket\_available` int(11) NOT NULL DEFAULT '-1',

`lat` text NOT NULL,

`lon` text NOT NULL,

`ticket\_url` text NOT NULL,

`email\_address` text NOT NULL,

`contact\_no` text NOT NULL,

`title` text NOT NULL,

`user\_id` int(11) NOT NULL DEFAULT '-1',

`is\_featured` int(11) NOT NULL DEFAULT '0',

`photo\_url` text NOT NULL,

`created\_at` int(11) NOT NULL DEFAULT '0',

`updated\_at` int(11) NOT NULL DEFAULT '0',

`is\_deleted` int(11) NOT NULL DEFAULT '0'

) ;

* + - 1. Table structure for table `tbl\_whatscamp\_event\_categories`

CREATE TABLE `tbl\_whatscamp\_event\_categories` (

`event\_category\_id` int(11) NOT NULL,

`event\_id` int(11) NOT NULL DEFAULT '0',

`category\_id` int(11) NOT NULL DEFAULT '0',

`created\_at` int(11) NOT NULL DEFAULT '0',

`updated\_at` int(11) NOT NULL DEFAULT '0',

`is\_deleted` int(11) NOT NULL DEFAULT '0'

) ;

* + - 1. Table structure for table `tbl\_whatscamp\_posts`

CREATE TABLE `tbl\_whatscamp\_posts` (

`post\_id` int(11) NOT NULL,

`event\_id` int(11) NOT NULL DEFAULT '-1',

`post` text NOT NULL,

`user\_id` int(11) NOT NULL DEFAULT '-1',

`created\_at` int(11) NOT NULL DEFAULT '0',

`updated\_at` int(11) NOT NULL DEFAULT '0',

`is\_deleted` int(11) NOT NULL DEFAULT '0',

`gmt\_date\_added` datetime NOT NULL

);

* + - 1. Table structure for table `tbl\_whatscamp\_users`

CREATE TABLE `tbl\_whatscamp\_users` (

`user\_id` int(11) NOT NULL,

`full\_name` text NOT NULL,

`login\_hash` text NOT NULL,

`facebook\_id` text NOT NULL,

`twitter\_id` text NOT NULL,

`google\_id` text NOT NULL,

`email` text NOT NULL,

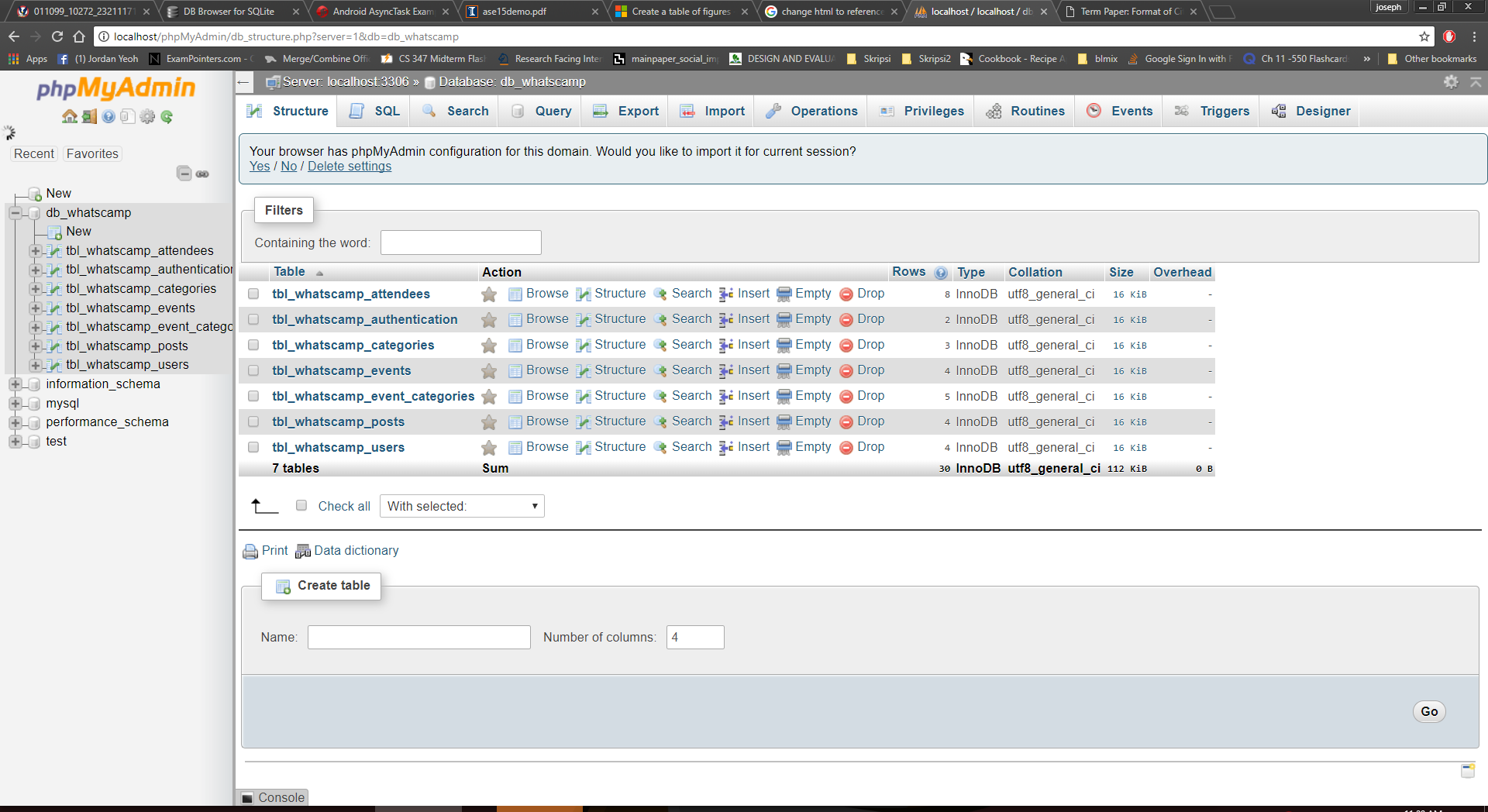
`deny\_access` int(11) NOT NULL,

`thumb\_url` text NOT NULL

)

As you can see, there are 6 SQL tables: tbl\_whatscamp\_attendees, tbl\_whatscamp\_categories, tbl\_whatscamp\_events, tbl\_whatscamp\_event\_categories, tbl\_whatscamp\_posts, tbl\_whatscamp\_users. Models can be find in MAMP/htdocs/whatscamp/models in php files.

This app contains prepopulated database with demo data that could be found in db\_whatscamp.sql. To modify the database, user need to import make new database in PhpMyAdmin named `db\_whatscamp` and import the file db\_whatscamp.sql to db\_whatscamp database. By having access to database, admin could add/remove/edit user, events, post, and category. Admin may freely do edit data in database, but do not modify the structure of the database



**Figure 6** WhatsCamp PhpMyAdmin

**Sumber:** app/Build.gradle

Below are the details of each tables:

tbl\_whatscamp\_attendees

* attendee\_id (integer) – Unique primary key
* event\_id (integer) – Foreign key, store the number of event that user\_id going to attend
* is\_going (integer) – number stored in is\_going by default is -1, and will change into 1 if user is going to attend
* is\_interested (integer) – number stored in is\_interested by default is -1, and will change into 1 if user is interested
* is\_invited (integer) – number stored in is\_invited by default is -1, and will change into 1 if user is invited
* user\_id (integer) – Foreign key, user ID
* updated\_at (integer) – timestamp when entry is updated
* created\_at (integer) – timestamp when entry is created
* is\_deleted (integer) – by default 0, and when user choose to cancel the attendencies, is\_deleted whanged into 1.

tbl\_whatscamp\_categories

* category\_id (integer) – unique primary key
* category (text) – name of the category
* Category\_icon (text) – store URL of the category image (nullable)
* created\_at (integer) – timestamp when entry is created
* updated\_at (integer) – timestamp when entry is updated
* is\_deleted (integer) – by default 0, and will change into 1 if category is deleted

tbl\_whatscamp\_events

* event\_id (integer) – unique primary key
* address (text) – address of the event
* event\_desc (text) – description of the event
* gmt\_date\_set (date time) – date and time of event
* is\_ticket\_available (int) – check if ticket available for the events.
* lat (text) – latitude of the event (for maps)
* lon (text) – longitute of the event (for maps)
* ticket\_url (text) – URL to book ticket
* email\_address (text) – email of the event provider
* contact\_no (text) – contact number of the event provider
* title (text) – title of the event
* user\_id (int) – Forein key, ID of user who post the event
* is\_featured (int) – to determine whether event is featured or not
* photo\_url (text) – URL of the image
* created\_at (integer) – timestamp when event is created
* updated\_at (integer) – timestamp when event is updated
* is\_deleted (integer) - by default 0, and will change into 1 if event is deleted

tbl\_whatscamp\_event\_categories

* event\_category\_id (integer) – Unique primary key
* event\_id – Foreign key, ID of the event
* category\_id – Foreign key, ID of the category that event belongs to
* created\_at (integer) – timestamp when entry is created
* updated\_at (integer) – timestamp when entry is updated
* is\_deleted (integer) - by default 0, and will change into 1 if relation between event and category is deleted

tbl\_whatscamp\_posts

* post\_id (integer) – Unique primary key
* event\_id (integer) – Foreign key, ID of the event that is related to the post
* post (text) – post content
* user\_id (integer) – Foreign key, ID of the user that posted a post/comment
* created\_at (integer) – timestamp when entry is created
* updated\_at (integer) – timestamp when entry is updated
* is\_deleted (integer) - by default 0, and will change into 1 if relation between event and category is deleted
* gmt\_date\_added (time date) – time and date when post is posted by a user

tbl\_whatscamp\_users

* user\_id (integer) – unique primary key
* full\_name (text) – name of the user
* login\_hash (text) – store data so that user could easily login to whatscamp
* facebook\_id (text) – store id of facebook account
* twitter\_id (text) – store id of the twitter account
* google\_id (text) – store id of the google account
* e-mail (text) – store user’s e-mail
* deny\_access (int) – store whether user’s access is denied
* thumb\_url (text) – user’s thumb url

In this application, device that connect to the host’s network could receive update from database. Therefore, when other user submit new event or modify event, user could receive the new data.

The name of prepopulated database is defined in MAMP/whatscamp/application/Config.php (const DB\_DATABASE) and WhatsCamp/app/src/main/java/com/db (static final String DB\_NAME)

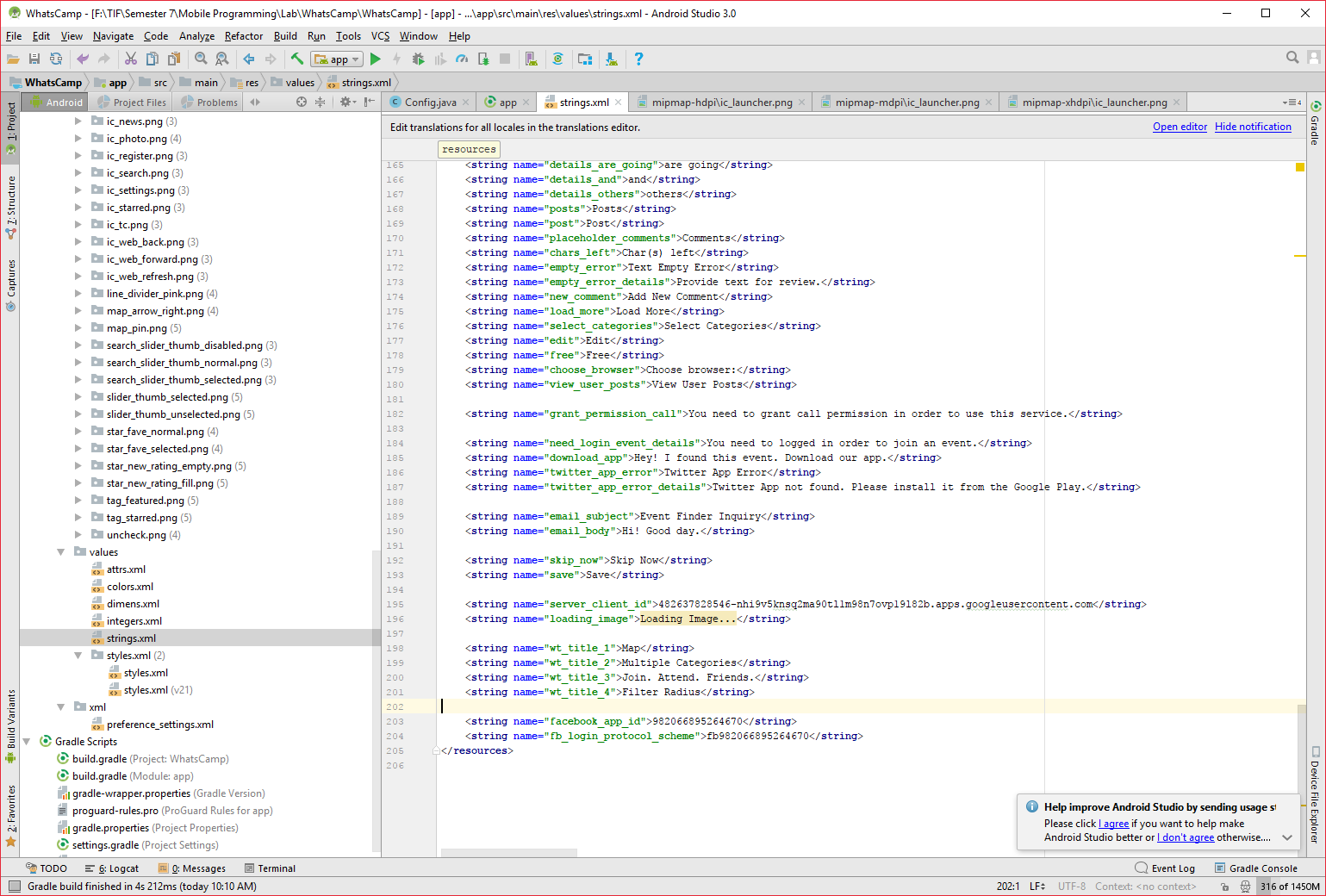
**Customization**

This chapter describes some optional customizations of the app.

**API setup**

This project used Google+ and Facebook API for login purposes. To get or generate your own API follow instruction in Google Developer (https://developers.google.com/+/web/signin/) and Facebook Developer (<https://developers.facebook.com/>)

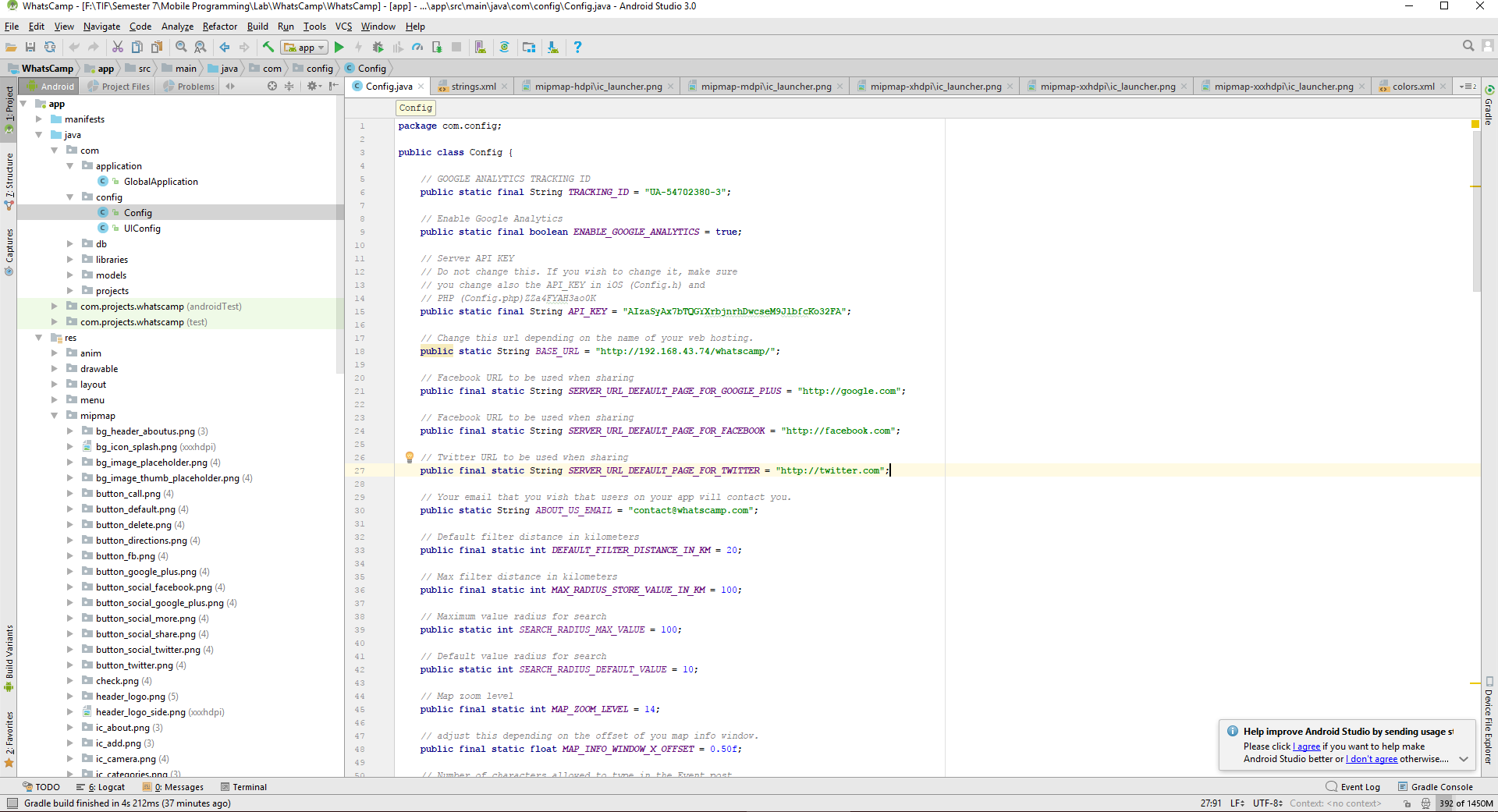
After receive your own API, change the “facebook\_api\_id” in /apps/src/main/res/values/string.xml file with your own app\_id and “fb\_login\_protocol\_scheme” with your API.



**Figure 7** Facebook API

**Sumber:** app/src/main/res/values/string.xml

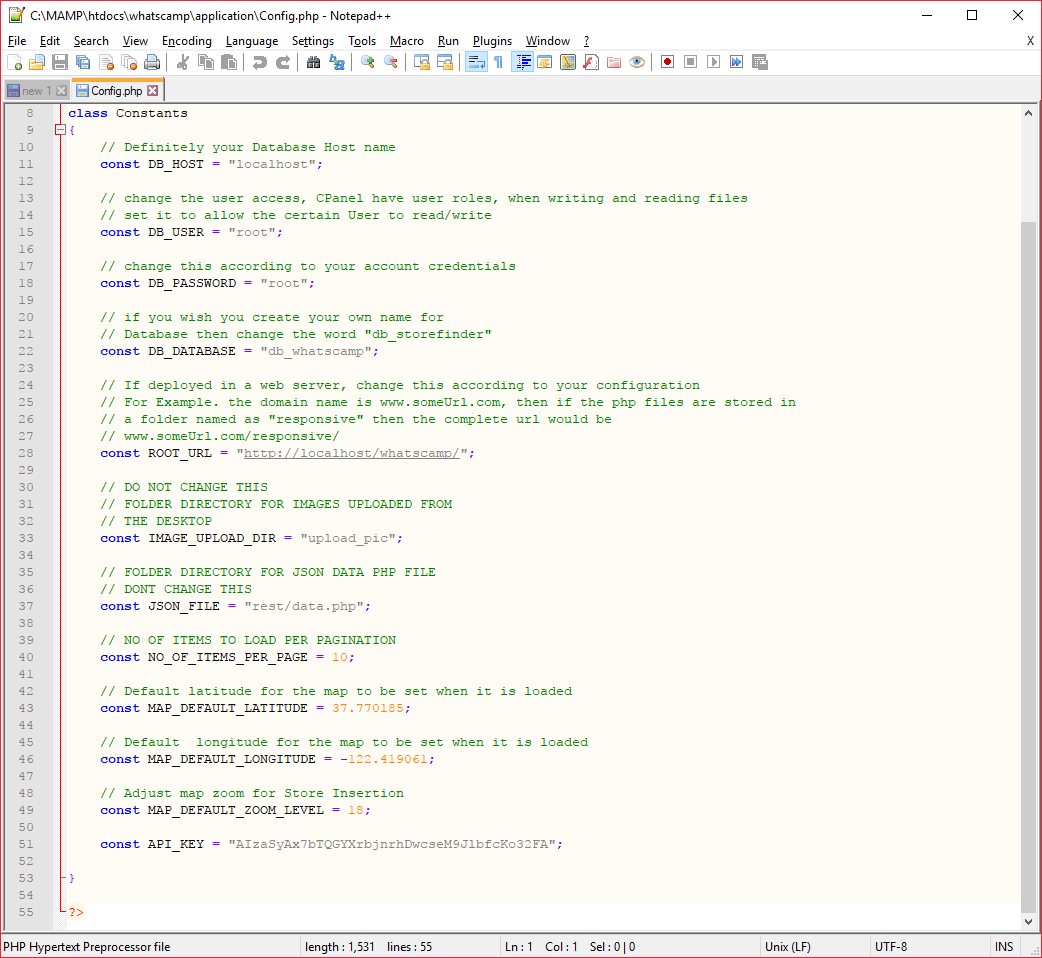
For Google+ API, you can change the Google+ API in API\_KEY inside app/src/main/java/com/config/Config.java



**Figure 8** Google API

**Sumber:** app/src/main/java/com/config/Config.java

For addition, change the API key inside PHP file in /MAMP/htdocs/whatscamp/ application/Config.php with Google API key



**Figure 9** API in PHP file

**Sumber:** MAMP/htdocs/whatscamp/application/Config.php

**Custom colors and icons**

You can customize colors in mobile/src/main/res/values/colors.xml (see Figure 3).

If you need to create the icon for the category, it is recommended to use [Android Asset Studio](http://romannurik.github.io/AndroidAssetStudio/index.html) or you can look at [Flaticon](https://www.flaticon.com). See [Android Cheatsheet for Graphic Designers](http://petrnohejl.github.io/Android-Cheatsheet-For-Graphic-Designers/#screen-densities-and-icon-dimensions) for correct icon dimensions. Use the icons with highest DPI.

**Custom logo**

You can easily change logo, icon or any image by replacing the file in drawable-xxhdpi directory.

**Building**

This chapter describes how to build APK with Gradle and prepare app for launch. Android Studio uses Gradle for building Android applications.

You don't need to install Gradle on your system, because there is a [Gradle Wrapper] (<http://www.gradle.org/docs/current/userguide/gradle_wrapper.html>) built in Android Studio. When you start a Gradle build via the wrapper, Gradle will be automatically downloaded and used to run the build.

1. Open the project in Android Studio
2. Open configuration file /src/main/java/com/config/Config.java and set constants as required (see below for more info)
3. Build and Run the project
4. APK should be available in /mobile/build/outputs/apk directory

**Note**: You will also need a "local.properties" file to set the location of the SDK in the same way that the existing SDK requires, using the "sdk.dir" property. Example of "local.properties" on Windows: “sdk.dir=C:\\adt-bundle-windows\\sdk”. Alternatively, you can set an environment variable called "ANDROID\\_HOME".

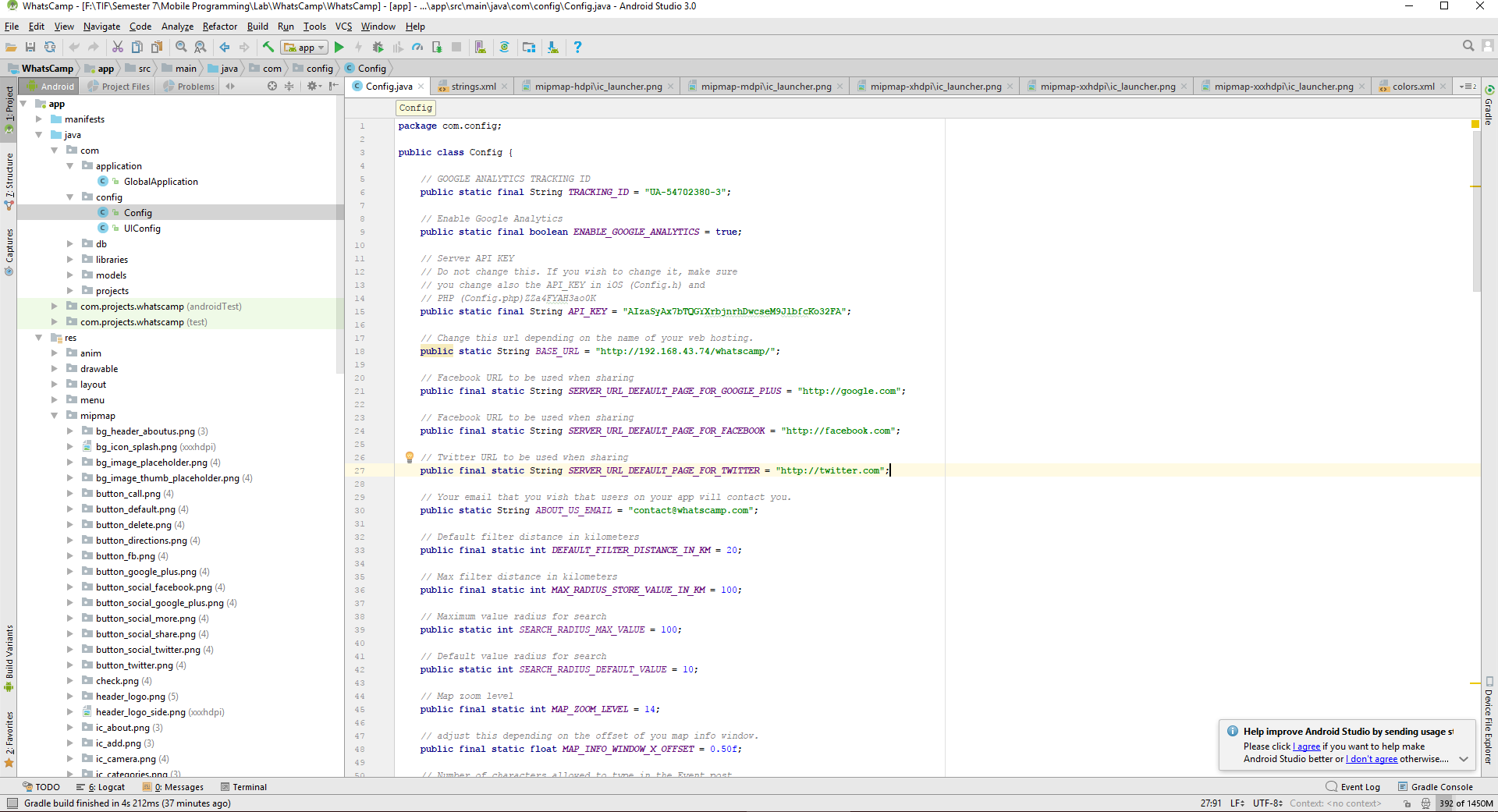
**Tip**: Command “gradlew assemble” builds both - debug and release APK. You can use “gradlew assembleDebug” to build debug APK. You can use “gradlew assembleRelease” to build release APK. Debug APK is signed by debug keystore. Release APK is signed by own keystore, stored in \_/extras/keystore\_ directory.

**Signing process**: Keystore passwords are automatically loaded from property file during building the release APK. Path to this file is defined in "keystore.properties" property in "gradle.properties" file. If this property or the file does not exist, user is asked for passwords explicitly.

**Config.java**

This is the main configuration file. There are some configurations that need to be changed for the application run correctly when building the APK.

* API\_KEY – used for Google Login
* ENABLE\_GOOGLE\_ANALYTICS - true for enabling Google Analytics
* BASE\_URL – define database location
* SERVER\_URL\_DEFAULT\_PAGE – URL used when sharing events
* ABOUT\_US\_EMAIL – email used for users to contact the developer

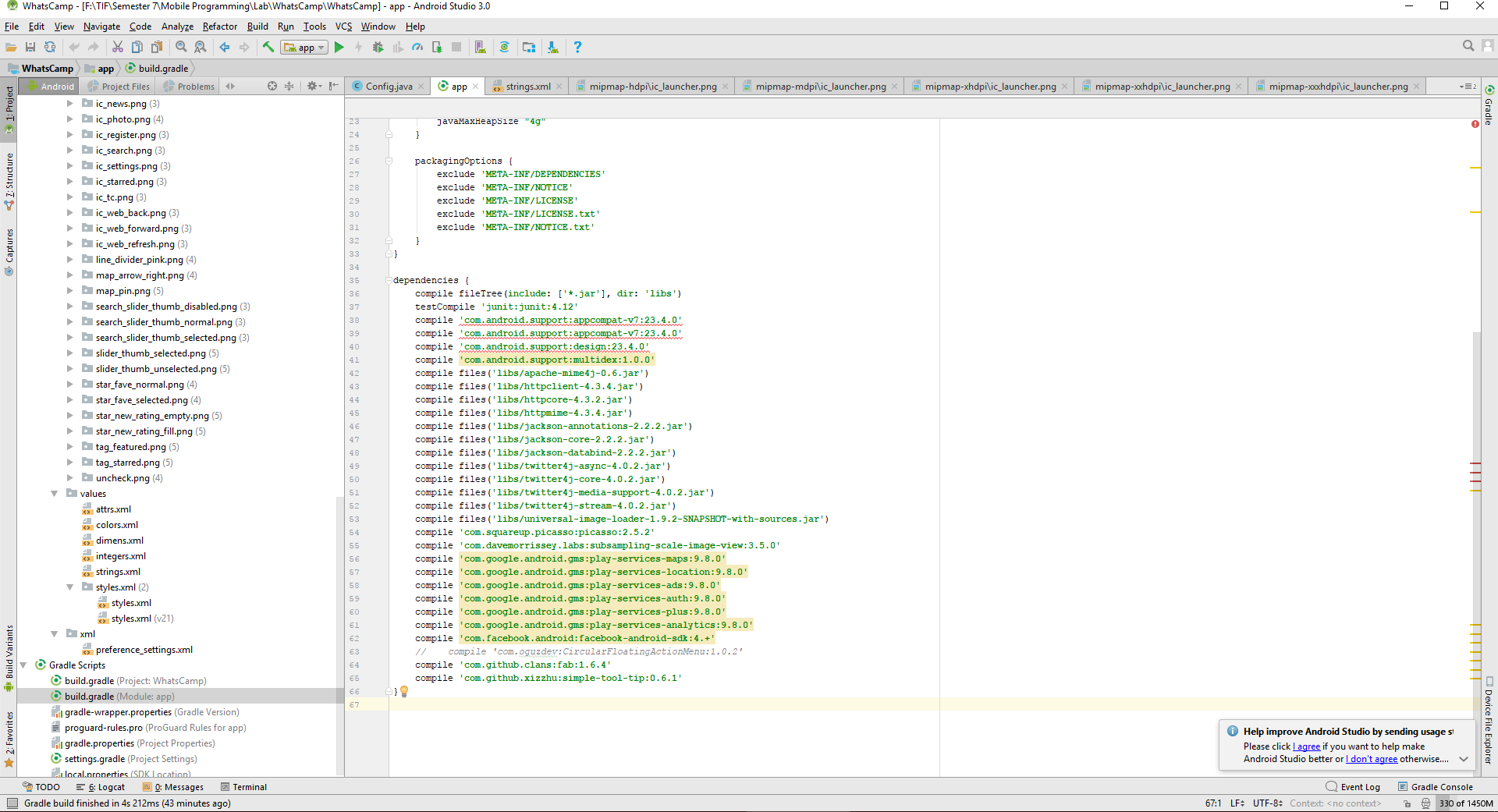


**Figure 10** Config.java

**Sumber:** app/src/main/java/com/config/Config.java

**Dependencies**

* [Design](https://developer.android.com/training/material/design-library.html)
* [Multidex](https://developer.android.com/studio/build/multidex.html)
* [AppCompat](https://developer.android.com/reference/android/support/v7/appcompat/package-summary.html)
* [ApacheMime] (https://gist.github.com/asakasinsky/6126409)
* [HttpClient](http://hc.apache.org/httpcomponents-client-4.3.x/android-port.html)
* [HttpCore](https://hc.apache.org/httpcomponents-core-ga/index.html)
* [JacksonDatabind](http://www.java2s.com/Code/Jar/j/jackson-databind.htm)
* [JacksonCore](http://www.java2s.com/Code/Jar/j/jackson-core.htm)
* [JacksonAnnotation](http://www.java2s.com/Code/Jar/j/jackson-annotations.htm)
* [TwitterAsync](http://www.java2s.com/Code/Jar/t/Downloadtwitter4jjar.htm)
* [UniversalImageLoader](https://mvnrepository.com/artifact/com.nostra13.universalimageloader/universal-image-loader/1.9.2)
* [TwitterMediaSupport](http://www.java2s.com/Code/Jar/t/Downloadtwitter4jjar.htm)
* [TwitterStream](http://www.java2s.com/Code/Jar/t/Downloadtwitter4jjar.htm)
* [GooglePlayServices](http://developer.android.com/google/play-services/index.html)
* [TwitterCore](http://www.java2s.com/Code/Jar/t/Downloadtwitter4jjar.htm)
* [SubSampling](https://github.com/davemorrissey/subsampling-scale-image-view)
* [Picasso](https://github.com/square/picasso)
* [FacebookAndroidSDK](https://developers.facebook.com/docs/android/getting-started/)
* [Tooltip] (https://github.com/ViHtarb/Tooltip)



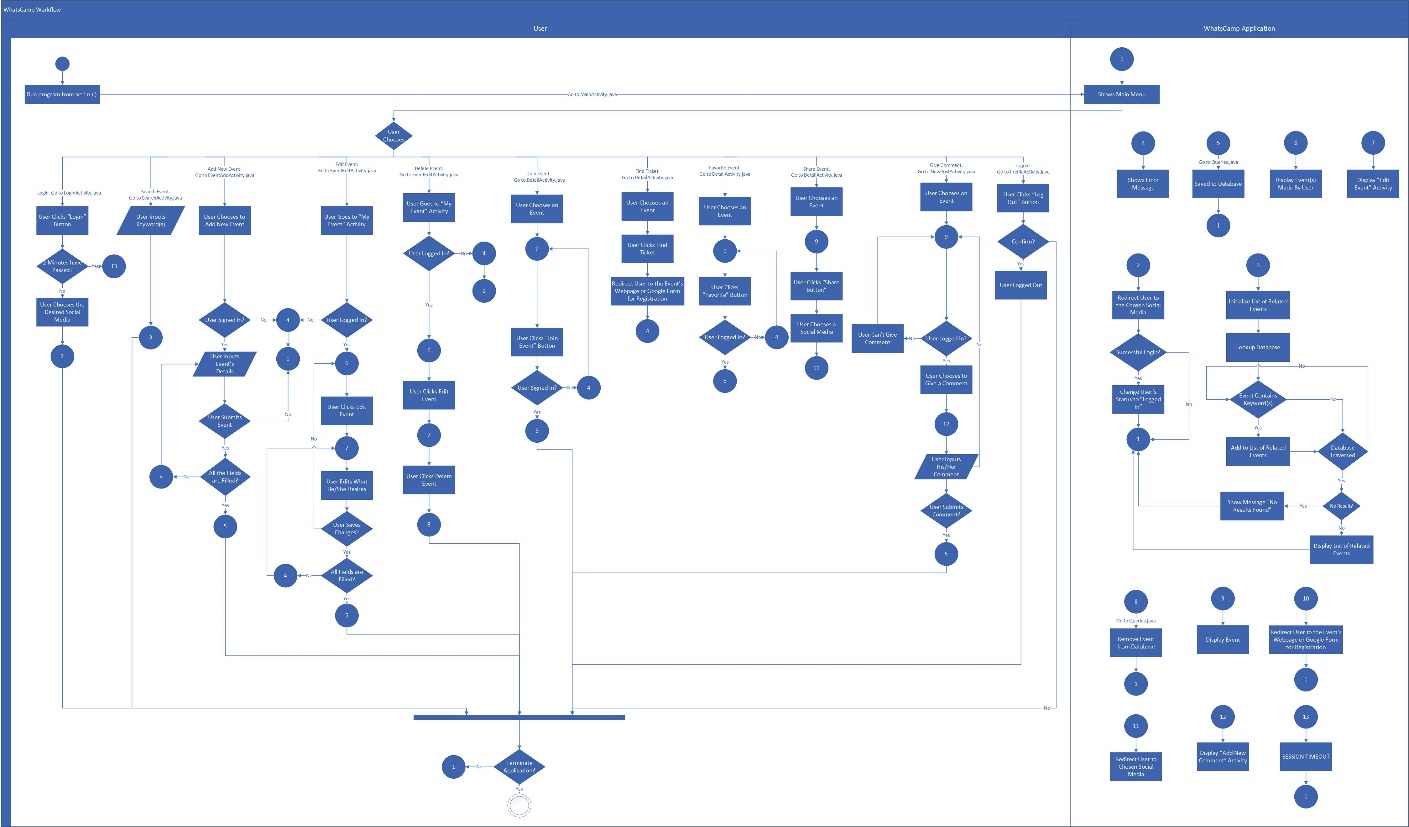
**Figure 11** Build.gradle

**Sumber:** app/Build.gradle

**Changelog**

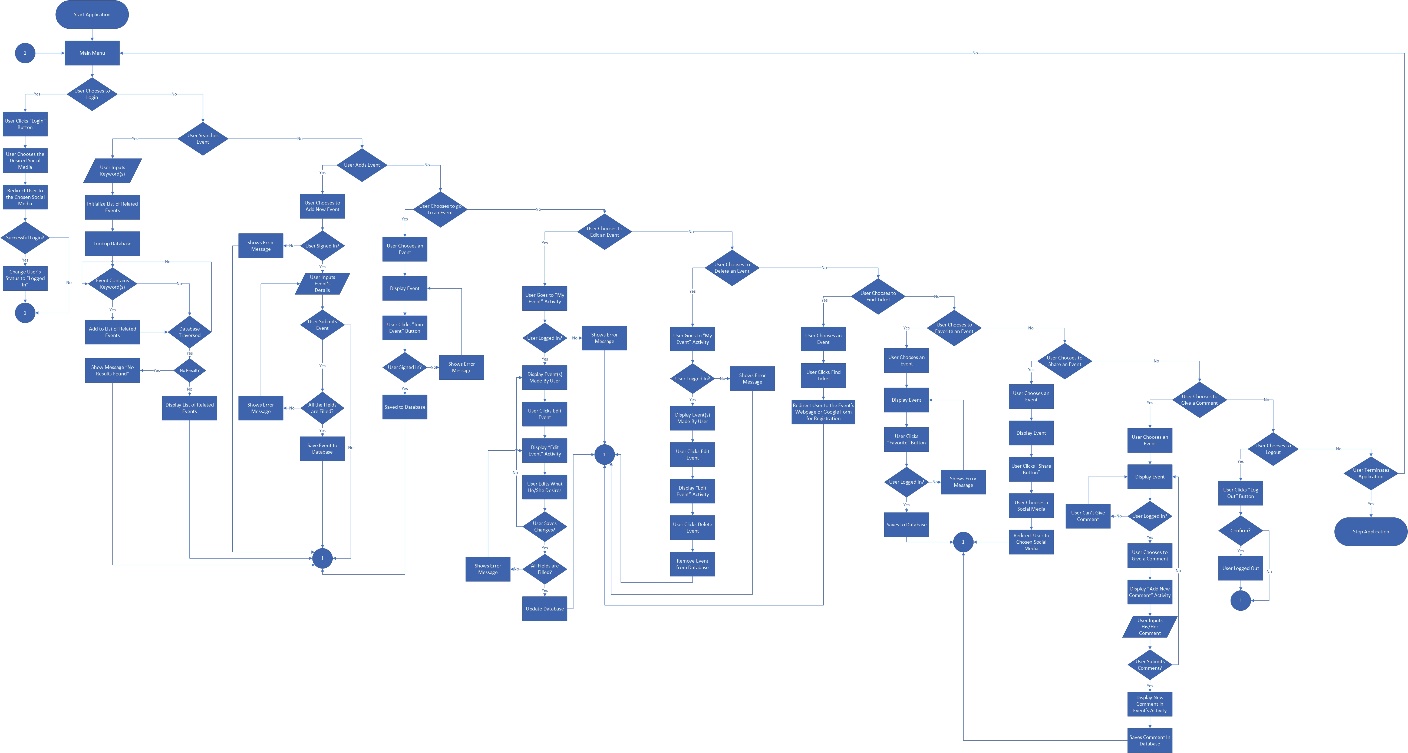
* Version 1.0.0
  + Initial release

**Workflow**

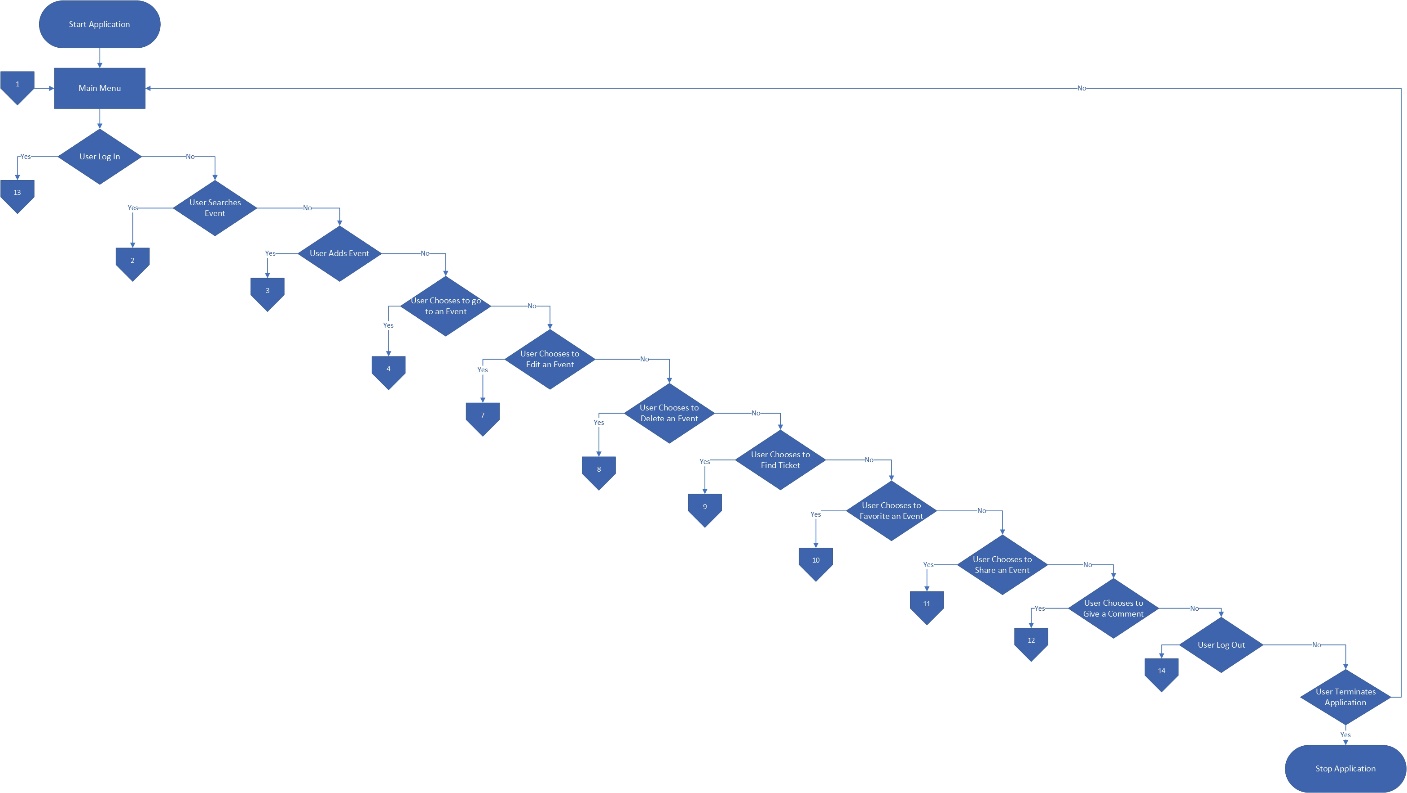
****

**Figure 12** Workflow

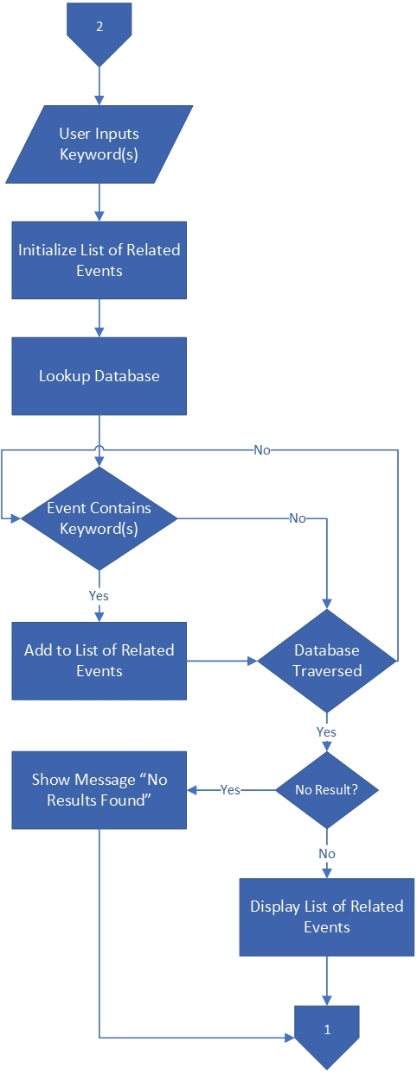
**Major Flowchart**

****

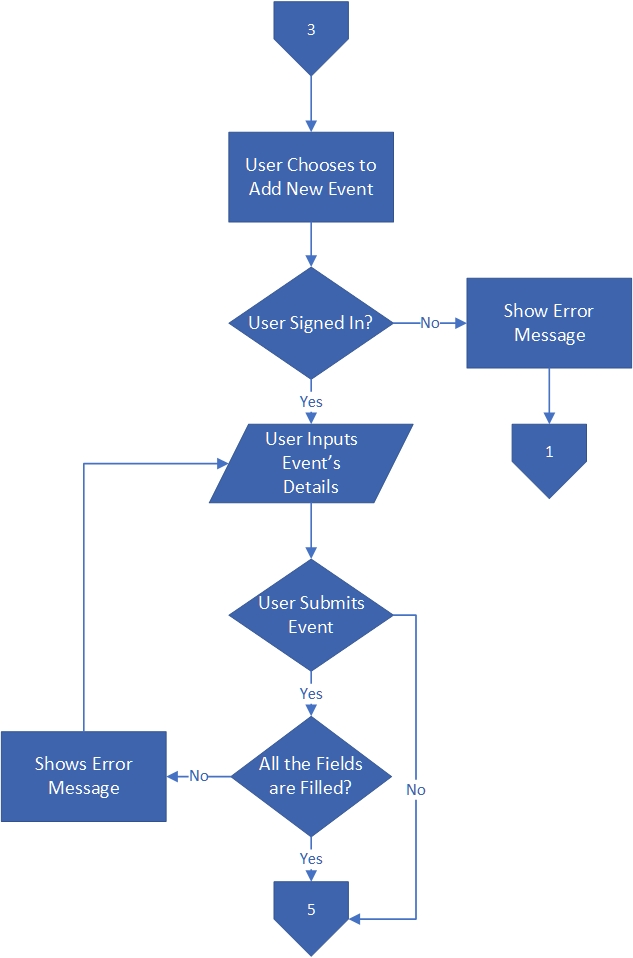
**Figure 13** Major Flowchart

**Detailed Flowchar**

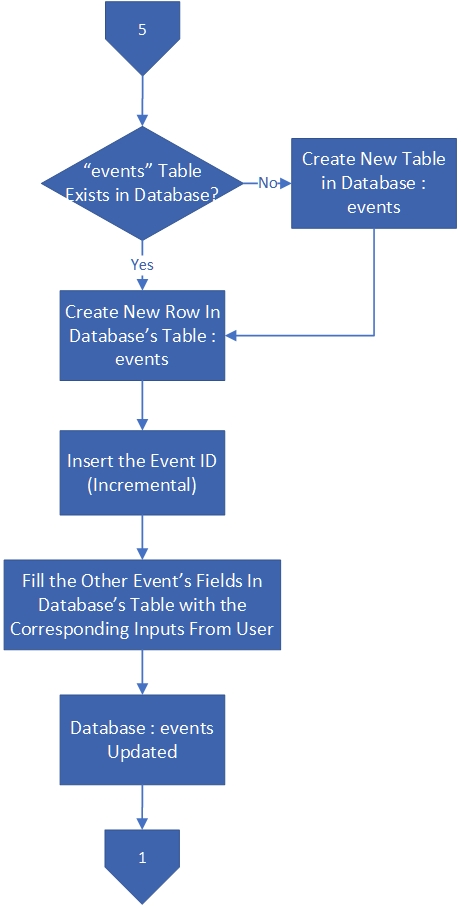
**Figure 14** Main Menu

****

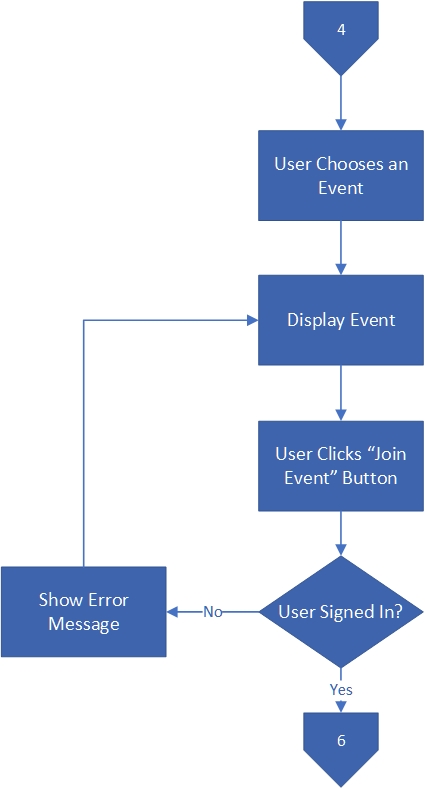
**Figure 15** Search Event

****

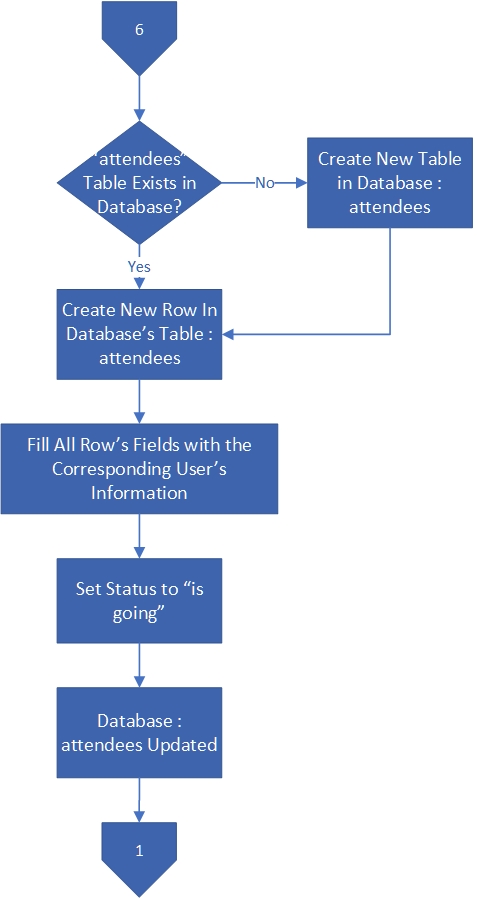
**Figure 16** Add Event

****

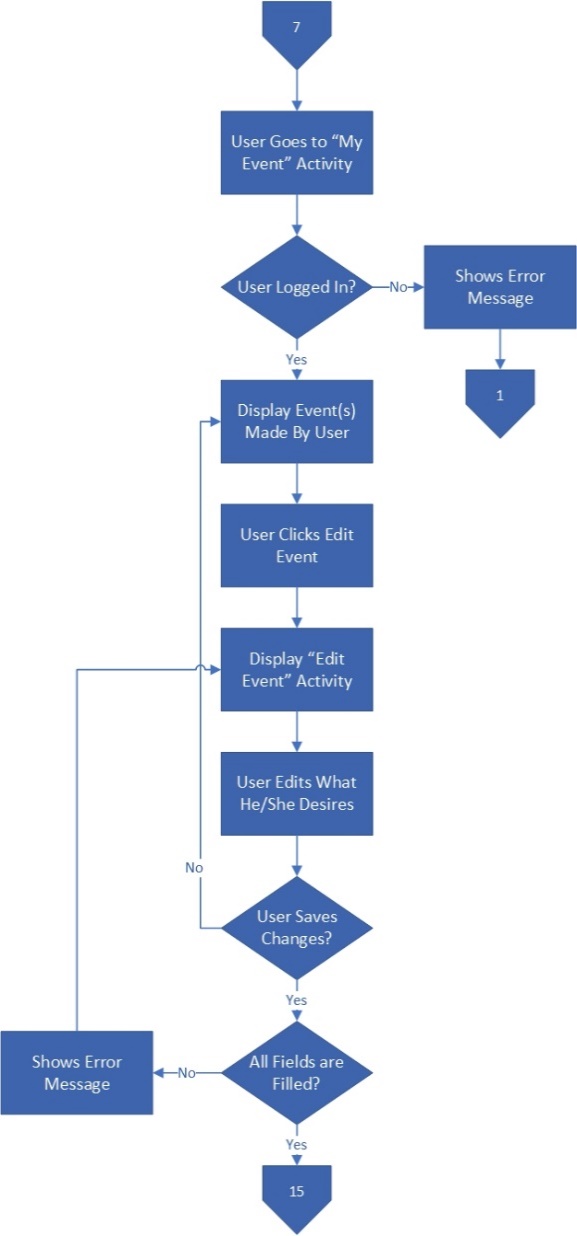
**Figure 17** Add Event (Database)

****

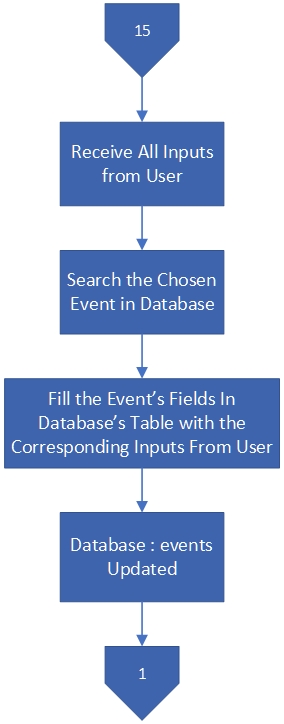
**Figure 18** Join Event

****

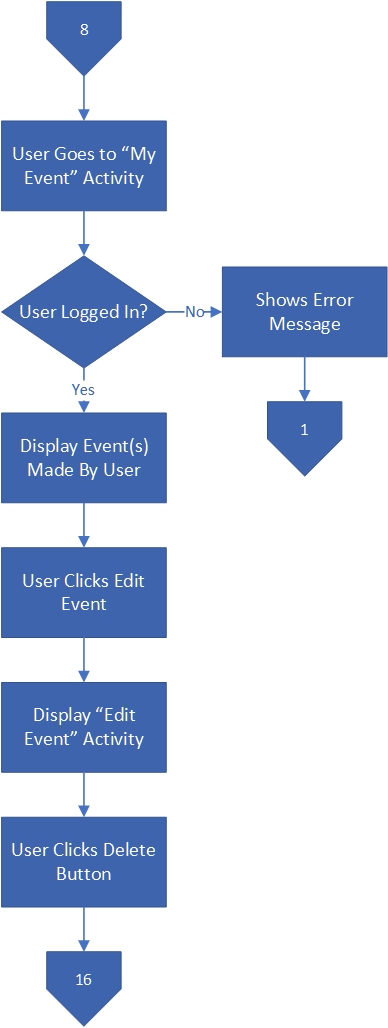
**Figure 19** Join Event (Database)

****

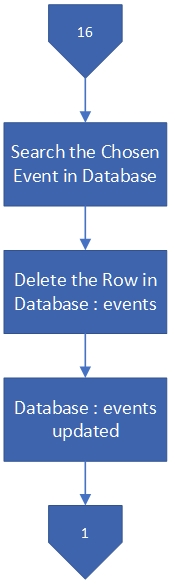
**Figure 20** Edit Event

****

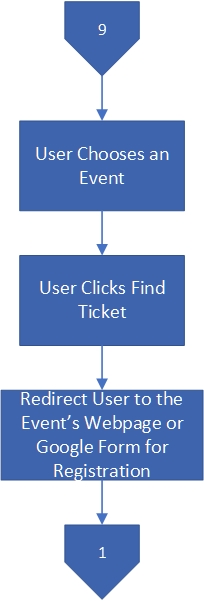
**Figure 21** Edit Event (Database)

****

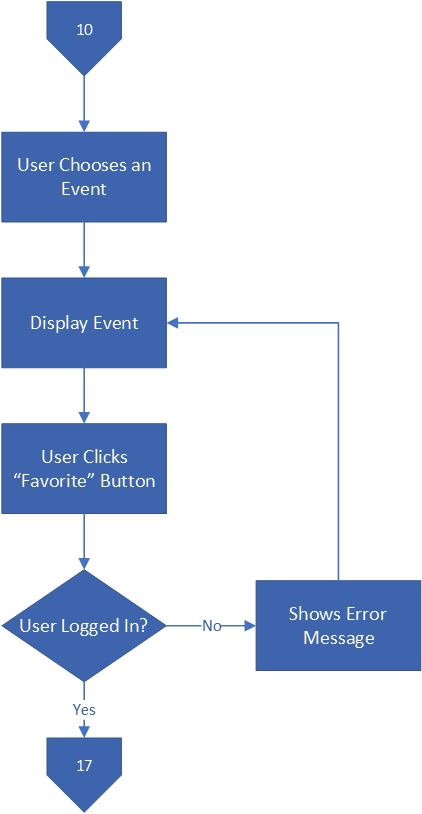
**Figure 22** Delete Event

****

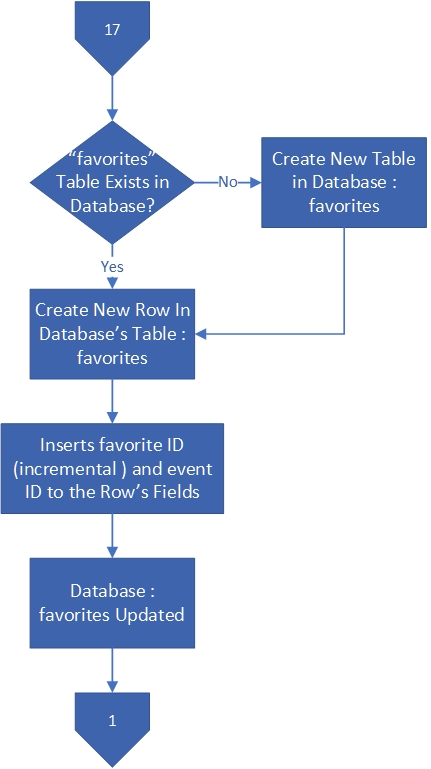
**Figure 23** Delete Event (Database)

****

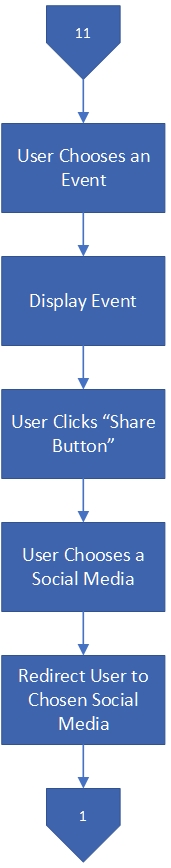
**Figure 24** Find Ticket

****

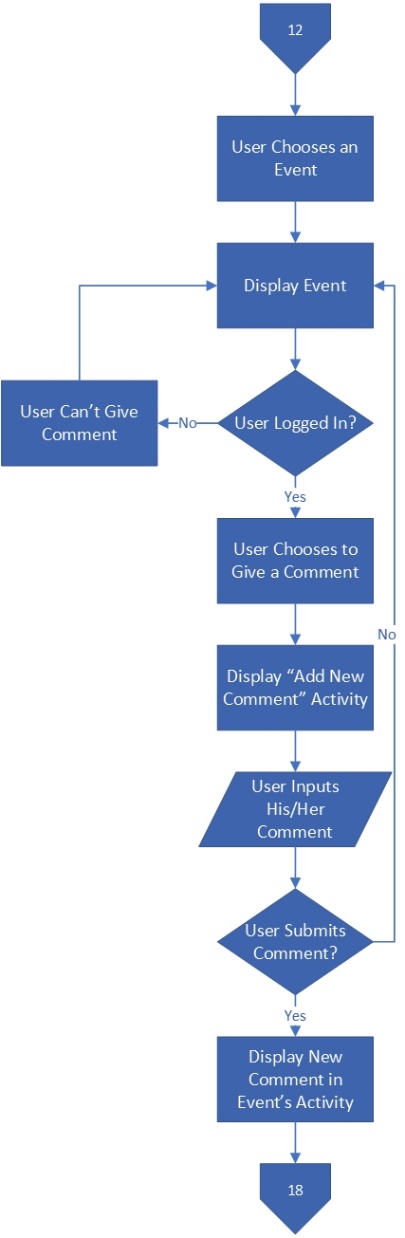
**Figure 25** Favorite Event

****

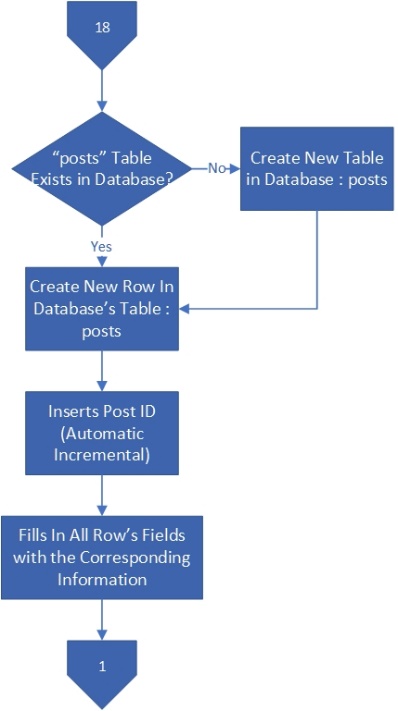
**Figure 26** Favorite Event (Database)

****

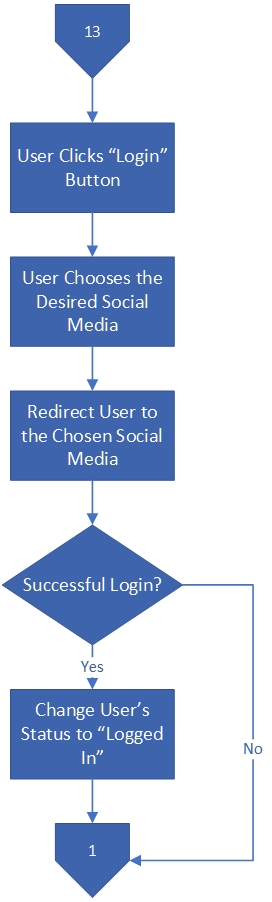
**Figure 27** Share Event

****

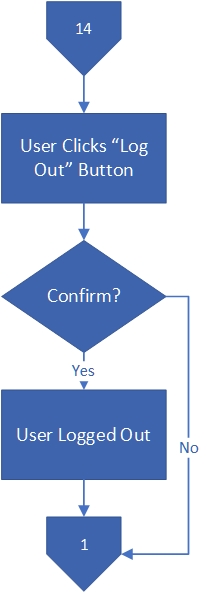
**Figure 28** Give Comment

****

**Figure 29** Give Comment (Database)

****

**Figure 30** Login

****

**Figure 31** Logout

**Link Reference** **List**

“Android - Facebook Login.” 2017. Accessed November 10. https://developers.facebook.com/docs/facebook-login/android.

“Sign In Users  |  Google+ Platform for Web  |  Google Developers.” 2017. Accessed November 10. https://developers.google.com/+/web/signin/.

“The Gradle Wrapper - Gradle User Guide Version 4.3.1.” 2017. Accessed November 2. https://docs.gradle.org/current/userguide/gradle\_wrapper.html.

“Free Vector Icons - SVG, PSD, PNG, EPS &amp; Icon Font - Thousands of Free Icons.” 2017. Accessed November 2. https://www.flaticon.com/.

“Android Asset Studio.” 2017. Accessed November 2. http://romannurik.github.io/AndroidAssetStudio/index.html.

“MAMP &amp; MAMP PRO.” 2017. Accessed October 15. https://www.mamp.info/en/.

“Android Asset Studio - Launcher Icon Generator.” 2017. Accessed November 2. http://romannurik.github.io/AndroidAssetStudio/icons-launcher.html#foreground.type=clipart&foreground.clipart=android&foreground.space.trim=1&foreground.space.pad=0.25&foreColor=rgba(96%2C 125%2C 139%2C 0)&backColor=rgb(68%2C 138%2C 255)&crop=0&.

“Android Cheatsheet for Graphic Designers.” 2017. Accessed November 4. http://petrnohejl.github.io/Android-Cheatsheet-For-Graphic-Designers/#screen-densities-and-icon-dimensions.

“Event Finder Full Android Application v1.0 » Download Free Premium Scripts, Wordpress Plugins, Mobile Games and Apps.” 2017. Accessed October 15. http://www.codelist.cc/mobile/2271-event-finder-full-android-application-v10.html.

“Update the IDE and SDK Tools | Android Studio.” 2017. Accessed November 14. https://developer.android.com/studio/intro/update.html.

“Download Android Studio and SDK Tools | Android Studio.” 2017. Accessed November 14. https://developer.android.com/studio/index.html.

“Java SE - Downloads | Oracle Technology Network | Oracle.” 2017. Accessed November 14. http://www.oracle.com/technetwork/java/javase/downloads/index.html.