

## Project 7: Capstone, Stage 1 – Design: “TripTop”

### **1 - Description**

### **2 - Intended User**

### **3 - Features**

### **4 - User Interface Mocks**

*Screen 1: Main Screen*

*Screen 2: Drawer menu*

*Screen 3: By Types*

*Screen 4: Countries*

*Screen 5: Alert Login*

*Screen 6: Add Trip*

*Screen 7: Add Photo*

*Screen 8: Add Types*

*Screen 9: Add Position*

*Screen 10: Recommends*

*Screen 11: Add Route*

*Screen 12: Add POI*

*Screen 13: Detail Screen*

*Screen 14: Detail Screen: Recommends*

*Screen 15: Detail Screen: Routes*

*Screen 16: Detail Screen: Vote*

*Screen 17: Favorites*

*Screen 18: Login*

*Screen 19: Register*

*Screen 20: Follow Us*

*Screen 21: Detail Screen for Route*

*Screen 22: Detail Screen Tablet*

*Screen 23: Countries Tablet*

### **5 - Key Considerations**

*How will your app handle data persistence?*

*Describe any corner cases in the UX.*

*Describe any libraries you'll be using and share your reasoning for including them.*

## **6 - Next Steps: Required Tasks**

*Task 1: Project Setup*

*Task 2: Implement UI for Each Activity and Fragment*

*Task 3: Implement UI for Countries Screen & GoogleMaps/Analytics*

*Task 4: Implement UI for AddTrip Screen*

*Task 5: Implement UI for Detail Screen*

*Task 6: Implement UI for Favorites Screen & Search View*

*Task 7: Implement UI for Login Screen & Follow Us*

*Task 8: Adapt the views to the different devices*

*Task 9: Create a Widget on the main screen of the device*

*Task 10: Testing the Login & AddTrip*

*Task 11: Creation of keystore, signed and version release*

## **7 – Schemes**

*Architecture*

*Structure DB*

[GitHub Username: mblancomu](#)

## 1 - Description:

Triptop app is a social tourism, which helps to find routes and trips made by other users, and shared through this app informally, without being associated with any organization, company or sponsor.

It is about sharing travel experiences of all kinds, in order to help other users to choose a destination, as well as everything that has an associated travel (food, lodging, routes, etc). TripTop about becoming a social network or bulletin board, providing an informal traveler view, the most important of this app is that the traveler informs the traveler, this is the key point of this new social environment. Unlike other apps tourism or travel, where only an automated search of accommodations, restaurants, tours, etc ... with hardly takes a few feedbacks travelers, here the information is provided by the traveler himself, recommending their sites and experience, away from the business world who pursue other apps.

In this app a lot of types of tourism are provided, to reach all travelers, hard to find on other sites or specialized apps. The traveler can describe their experience, value it, recommending places and even upload photos. They may find experiences around the world, ideal for deciding what to do next weekend or the next summer vacation.

The app provides updated and cataloged by geographical location, type of tourism, last added routes, ranking of the most valuable, information etc. And a map with the different routes and added places.

This app is solved the problem of knowing plan a trip or find something that is not what you promised, for travelers inform travelers worldwide.

## 2 - Intended User:

This app is suitable especially for undecided tourists and travelers who need a push to decide the perfect place. **Travelers**.

## 3 - Features

The List of the main features of this app are:

- Saves local information
- Takes pictures
- Geolocation
- Share information
- Uses a REST service
- Upload data

## 4 - User Interface Mocks

Then the screens that make up the app. They are only one project and could suffer any changes during the project.

For the realization of this design has been used open source software Pencil Project (<http://pencil.evolus.vn/>) and the official iconography of Android (<https://design.google.com/icons/index.html>) for Material Icons.



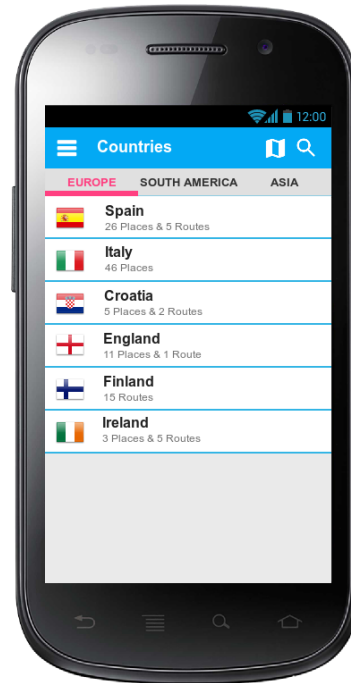
Screen 1. Main Screen with the trips order, by different options.



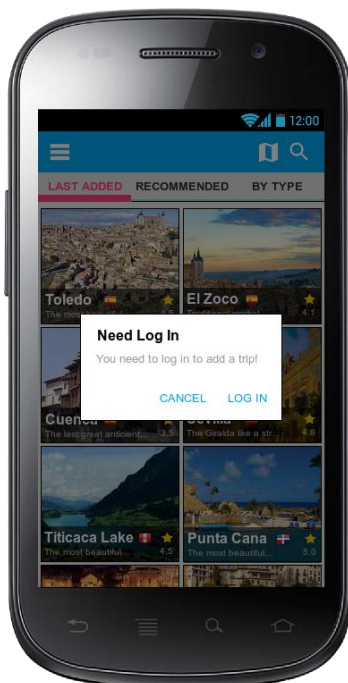
Screen 2. Drawer menu with the different options.



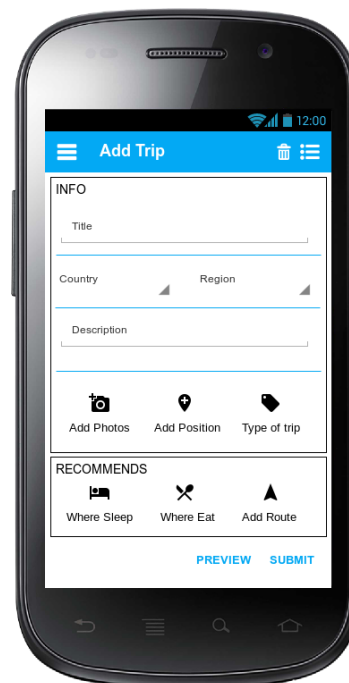
Screen 3. Main Screen with the trips classified by type. The user can select the type of trip that want.



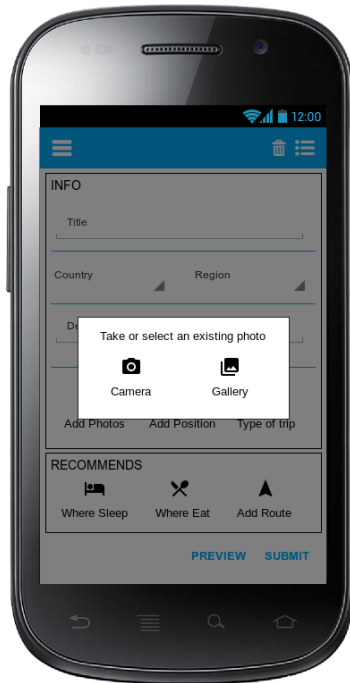
Screen 4. Trips grouped by country. The users can select the desired country.



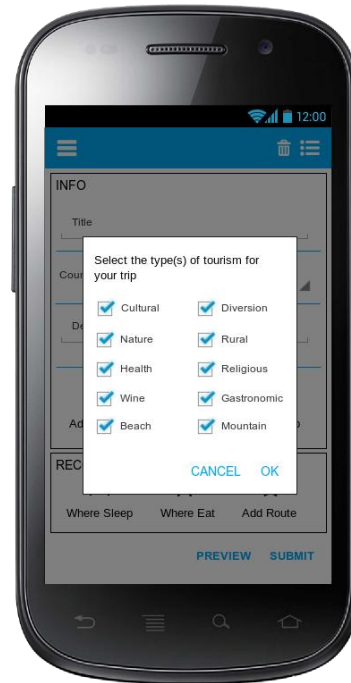
Screen 5. If the user wants to add a trip from the drawer menu, he must be logged before.



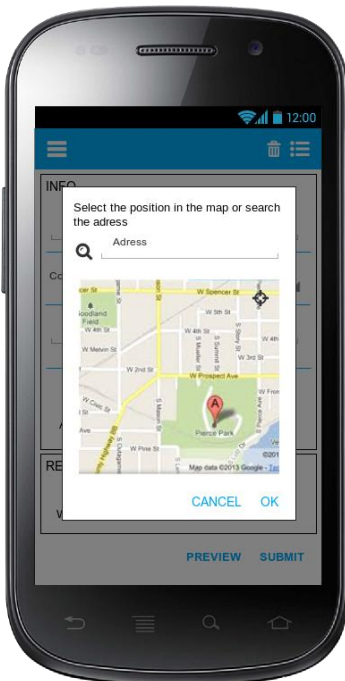
Screen 6. When the user is logged in, he can add a trip, as seen in this screen.



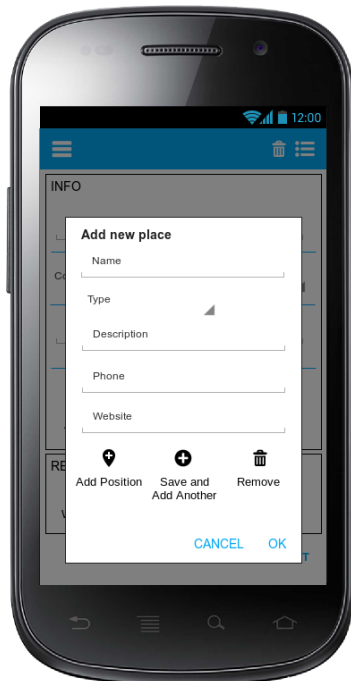
Screen 7. Under section add trip, the user can add photos from the gallery or take a picture with the camera.



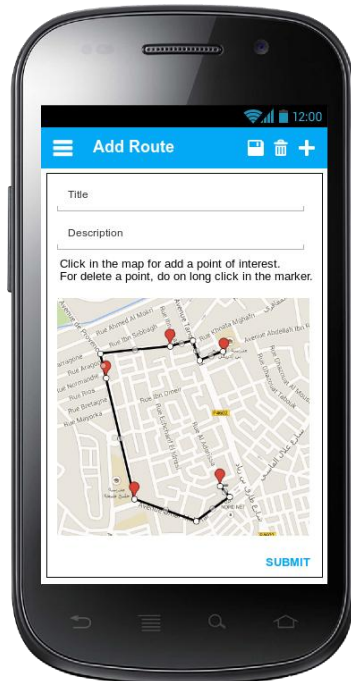
Screen 8. The user must add the types of tourism that best fit his trip, for more later filtering by type.



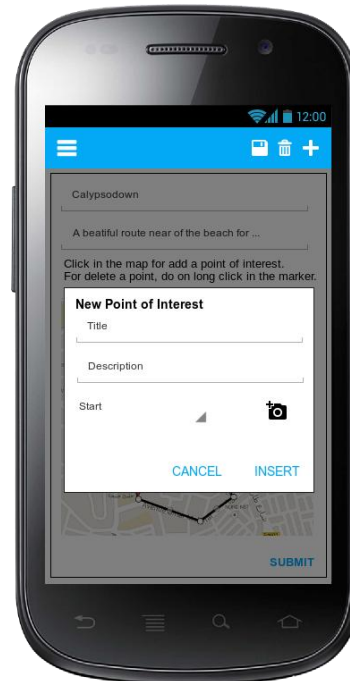
Screen 9. The user must add the position of the place to be geolocated. Along with the above actions, this information is required to add a trip.



Screen 10. The user may recommend places to eat or sleep. This information is not required to add a trip, it is only a recommendation.



Screen 11. Also, the user can add a route with several points of interest along this trip. This information is not mandatory, but can be very useful.



Screen 12. To draw the route, the user click on the map for add points of interest, associating different information for each point.

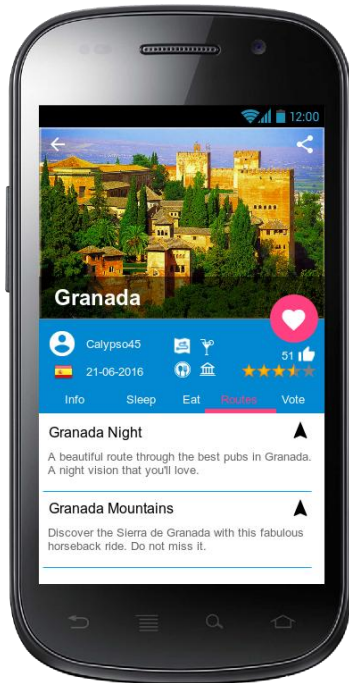


Screen 13. When the user selects a trip, a detail screen with all the travel information (entered by another user before) is displayed.



Screen 14. There are different tabs, the information provided by another user. In this screen we see such a list of recommended accommodations.

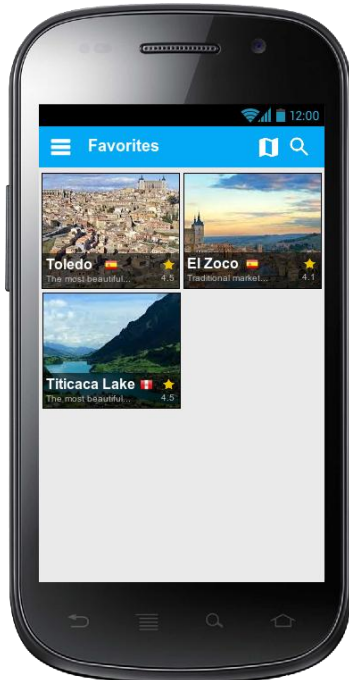




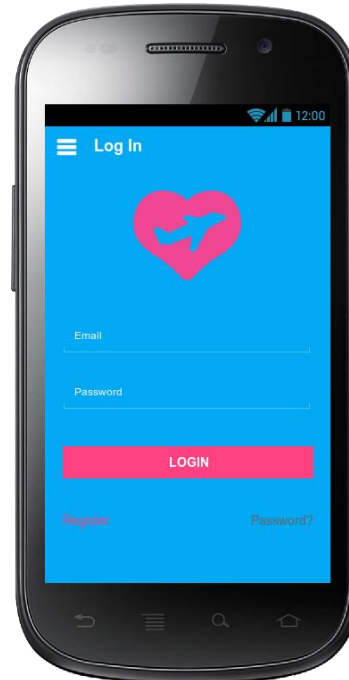
Screen 15. Another tabs that the user can find in the detail screen, is the route. Although this information is not mandatory.



Screen 16. Finally, the user can vote and say that likes a particular trip. Thus, we can perform a ranking with the registered votes.

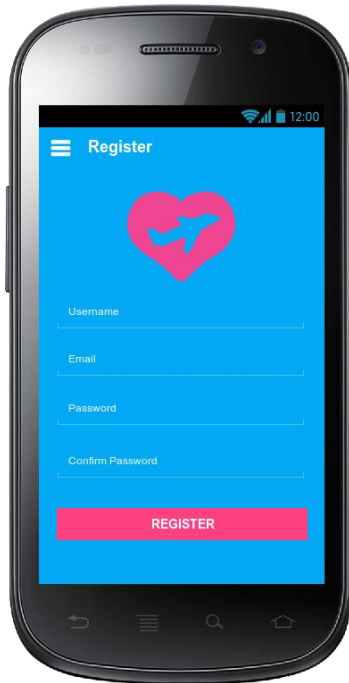


Screen 17. Another section of the menu, is the favorites. Here the user will be stored his trips, through the floating button on the screen detail of a trip.

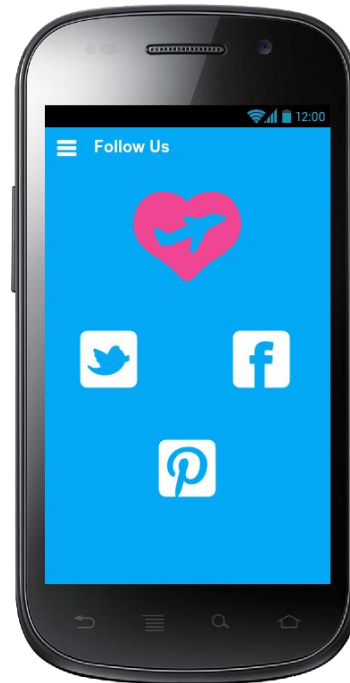


Screen 18. On the menu we also have login section, necessary for users who want to add a trip.

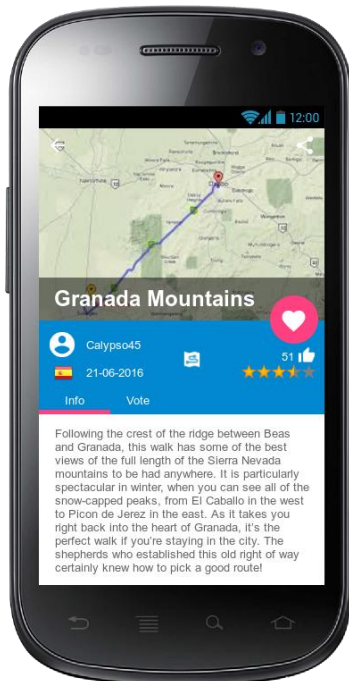




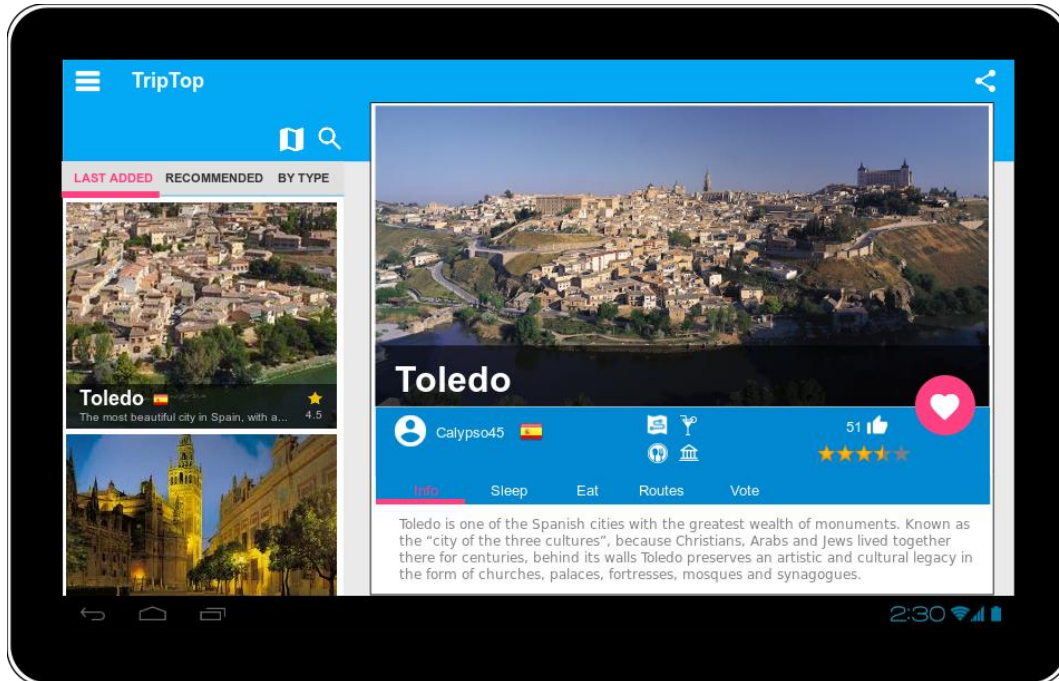
Screen 19. If the user haven't an account, can do the register from the login section.



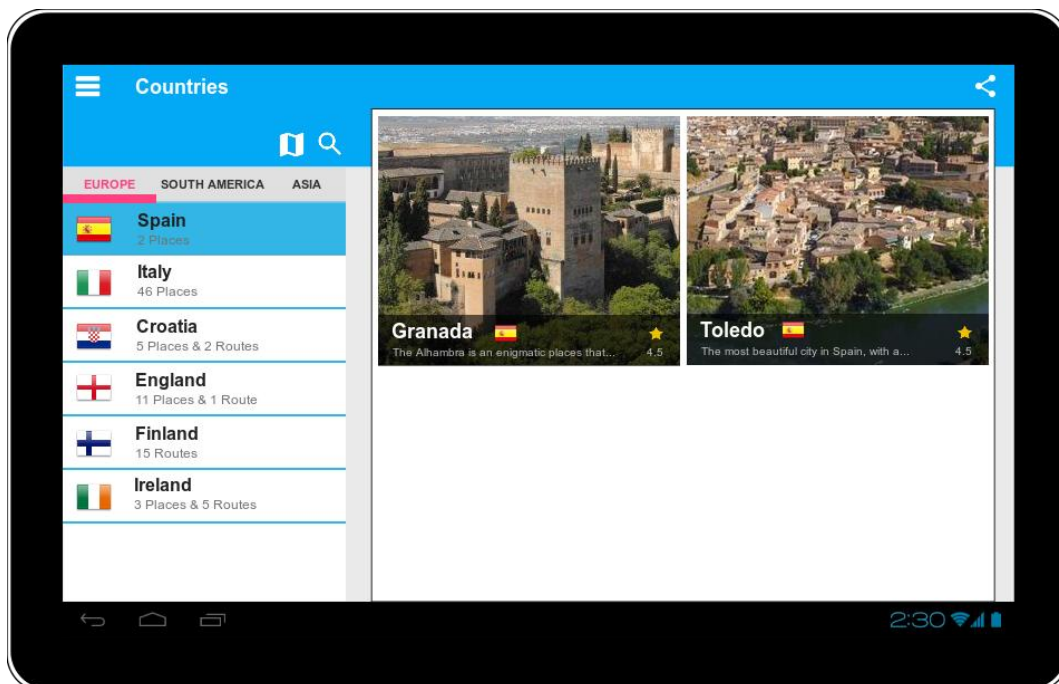
Screen 20. In order to share information and follow the evolution of the app, there is a section of social networks on the menu, called "Follow Us".



Screen 21. Detail screen for a route. The user can set the map to full screen to display the route.



Screen 22. Screen detail of a trip for a tablet in landscape. It is selected by default the first item in the list.



Screen 23. For “Countries” section in tablet, we had a list with all the countries on the left, and the trips on right for each country. The remaining sections in the drawer menu will be similar to these screens for tablets.

## 5 - Key Considerations

### How will your app handle data persistence?

The app will get the data through a REST service using an external library, Volley, for it. This data is stored in the app, in a local DB (SQLite), and access it through a Content Provider. The data will be drawn on the view through a Loader, optimizing the updating of views with the new data recorded.

For synchronizing the data app with the service REST, we will be used a SyncAdapter, which will optimize the service and the access to the remote data.

### Describe any corner cases in the UX.

The user will see a main screen with different tabs, depending on the latest added trips, types and ranking. Each tab contains a grid with different options. In the case of the latest added trips and ranking, clicking on one, the user will see a detail screen with location data, photo, fab button, recommendations, comments, etc. When the user rotate the screen, the information will be displayed differently to take advantage of the width of the device.

Clicking on a type of tourism, the user will access a list with the results, and if you click on one of them, can see the same screen detail discussed above, with the same information and behavior.

In the drawer menu, the user can Access to the following sections: Home, Countries, Add Trip, Follow Us, My Favorites and LogIn.

In the icon map the user will see a full-screen map with all routes and trips, focusing on his current location. The user can interact with the map, clicking on a route or point of interest and accessing to the detail information. Also can to access a list of sites sorted by distance to the user. When you rotate the screen, the map will be redrawn to take advantage of the width of the device.

In the Follow Us button, the user can access by web to the different social networks, for know the latest news about the app, and what the users say about the different trips.

### Describe any libraries you'll be using and share your reasoning for including them.

The libraries used in this project are:

- **Picasso**: for get images from the service.
- **Butterknife**: for inject views efficiently.

- **Volley:** for the calls to the backend service.
- **Backendless:** For manage the remote service and store the information in backend.
- **Google Play Services:** For work with *Google Maps & Google Analytics*. It's not sure, but will be to use also *Google Maps Directions & Places*.
- Other libraries for some components like animations or custom views.
- Of course, we also use the Android compatibility libraries as well as design, palettes, etc.

## 6 - Required Tasks

Then I will describe the different tasks for the project, many of them divided into subtasks.

### Task 1: Project Setup.

In this task the different android design guidelines and libraries to use are analyzed.

The first is to create a repository on GitHub, on what will work in the future. Then, we create an Android project, on which work, and where will be to add in the file build.gradle, the different libraries to use in the project, which we discussed in the previous section.

Between the different subtasks that will do to continuation are:

- Create accounts and projects in the different services to use, in my case: GoogleMaps, Analitics and Backendless (backend service).
- After obtaining the different API keys, the empty project will set up in order to work with them.
- Create different tables that configure the backend service. Which fields compose and create the structure and relationship between them will be analyzed. It is also analyze which tables will stay local and what the backend service only.
- Preparation of necessary design resources, such as icons, colors, fonts, etc. Material Design Guidelines will be consulted and there will be obtained the iconography and color palette.
- Define the launcher icon for the app, and prepare it for different resolutions.
- Define the user's role within the app, limiting the actions that he will be to perform and the characteristics of the files that the user can upload to the service (text, images size, etc).

## Task 2: Implement UI for Main Screen & DrawerMenu.

In this task, I'll create the different activities and fragments, focusing on the main screen and the drawer menu. The list of subtasks that make up this section are:

- BaseActivity & BaseFragment (From this classes shall inherit all the activities and fragments).
- MainActivity, MainFragment & MainPresenter.
- DrawerActivity.
- MainGridAdapter.
- MainPagerAdapter.
- BackendProvider (logic for the GET service).
- SqliteManager.
- Create the ContentProvider , SyncAdapter and Loader in the Main Activity.
- Creating the layouts for this activities and fragments:
  - activity\_main
  - fragment\_main
  - item\_maingrid
- Creating the App Theme and the different styles.
- Creating files for the strings,colors, integers, etc for this classes.

### **Task 3: Implement UI for Countries Screen & GoogleMaps/Analytics.**

Implement the UI for Countries Activity & the Google Play Services to use in this app, in this case: GoogleMaps & Google Analytics. access to the backend service to consume the information is also controlled.

It will initialise Google Analytics in the class ApplicationController, that extends of Application. Here also instanciarán by first and only time, the database, location and GoogleMaps.

They will control the errors for these implementations and will incorporate the API keys to the Manifest. The UI to make in this task are:

- Create CountriesActivity, CountriesFragment & CountriesPresenter.
- Create MapActivity, MapFragment & MapPresenter
- Create the CountriesAdapter.
- Create LocationProvider.
- Creating the layouts for this activities and fragments:
  - activity\_countries
  - fragment\_countries
  - item\_countries
  - activity\_map
  - fragment\_map
- Add the different resources into the files: strings, colors, integers, etc.
- Creation of the classes of support: Constants, Utils, Errors, etc.

### **Task 4: Implement UI for AddTrip Screen.**

In this task, perhaps the most complex and long, will create the UI for Add trips, and will control the access to the login, so that only it execute this activity if the user is logged. It adds the logic to access to the camera, map and service backend for upload the user information.

The subtasks that compose this task are:

- Create AddTripActivity, AddTripFragment & AddTripPresenter.
- BackendProvider (Logic for the POST service).
- Create TakePhotoFragment & TakePhotoDialog.
- Create AddPositionFragment & AddPositionDialog.
- Create AddEatSleepDialog.
- Create AddRouteFragment.
- Create AddPOIDialog.
- Create the layouts:
  - activity\_add\_trip
  - fragment\_add\_trip
  - fragment\_take\_photo
  - dialog\_add\_position
  - dialog\_add\_types
  - dialog\_add\_eat\_sleep
  - fragment\_add\_route
  - dialog\_add\_poi
- Add the different resources into the files: strings, colors, integers, etc.

## **Task 5: Implement UI for Detail Screen.**

Creation of the detail screen for the different lists, like trips and routes. It will allocate quite time in this activity to create the sight of detail, with material components design, like the rest of seen of the app.



In this task will make the following subtasks:

- Create DetailActivity, DetailFragment & DetailPresenter.
- Create DetailPagerAdapter.
- Create DetailListAdapter.
- Create the layouts:
  - activity\_detail
  - fragment\_detail
  - item\_eat\_sleep
  - item\_routes
- Incorporate strings, colors, etc. to his respective files.
- BackendProvider (Prepare the service backend to visualise images of different sizes).
- AnimationsUtils : It will contain the different animations that will create in this screen and other parts of the app.
- CustomViewsProvider: Incorporate here the different custom views.

## **Task 6: Implement UI for Favorites Screen & Search View.**

In this task will create the interface for the favorites screen, as well as the options of research. This task includes the creation of the table of favourites in the database, as well as the class that will control these queries through a content provider and loader.

The subtasks that compose this point are:

- Create FavoritesActivity, FavoritesFragment & FavoritesPresenter.
- Create FavoritesDB: table that will contain the information of the favourites of the user.
- Create DBProvider: it will contain the actions of dates it base.
- Create SearchActivity, SearchFragment & SearchPresenter.

- Create UpdateResults with the methods to update the researches.
- Create the layouts:
  - activity\_favorites
  - fragment\_favorites
  - activity\_search
  - fragment\_search
- Incorporate strings, colors, etc. to his respective files.
- Incorporate different methods and utilities to the classes Constants, Utils, etc.

## **Task 7: Implement UI for Login Screen & Follow Us.**

It will implement the logic of the login in this task, so much in the service backend as inside the app. This functionality allows the access to parts of the app, as add trip and favorites.

Besides, will be create the views and activities for the section “Follow Us”, launching an external link through a webview, where the user can visit the social networks associated to this app.

Between the subtasks of this point have:

- Create LoginActivity, LoginFragment & LoginPresenter
- Create RegisterFragment
- Create LoginDialog
- Create WebViewActivity & WebViewFragment
- Create FollowUsActivity & FollowUsFragment
- Create the layouts:
  - activity\_login
  - fragment\_login
  - fragment\_register
  - dialog\_login

- fragment\_webview
- activity\_webview

### **Task 8: Adapt the views to the different devices.**

In this task will verify the correct visualisation in the different devices for any screen, especially the ones of detail, that are those more different. Between the actions to make in this task are:

- Adapt layouts for tablet: Main, Detail, Favorites, Countries & AddTrip.
- Create the navigation, adapted so much for smartphones like tablet.
- Create TabletFragment for the different screens that visualise in the tablet, like screen of detail.
- Create logic for the different devices.
- Verify the correct visualisation of the icons and images for the different devices (Size and densities).

### **Task 9: Create a Widget on the main screen of the device.**

Create a widget that visualise the latest trips added, with basic information in a cardview.

In this task will make the following subtasks:

- Create WidgetProvider to control this component.
- Create WidgetDetail with the view and logical of this component.
- Create the layout:
  - widget\_detail

### **Task 10: Testing the Login & AddTrip.**

Verify the correct operation of the login and add trip. When being the most delicate parts of the app, as we have to control the errors associated to the service backend and the control of users. A task is intended exclusively to testing these components.

### **Task 11: Creation of keystore, signed and version release.**

The last task will be to create a keystore, with the digital signature for this project, put in a path relative inside the app.

Created and signed the app, generate an apk of release, the which should be testing for his correct operation, before finalizing completely the app, and can upload it later to the Play Store.

## **7 - Schemes**

In the following pages include a series of graphs describing the Project. These are:

- The App Architecture (the app name is TripTop).
- The data base structure for the app.

## TripTop Architecture



## TRIPTOP DATA BASE STRUCTURE

