

# Design Proposal

---

Author on Github: yahiasalem94 – PlanIt

## Table of Content

---

Description .....	2
Intended User .....	2
Features.....	2
User Interface Mocks .....	3
Key Considerations .....	4
How will your app handle data persistence? .....	4
Describe any edge or corner cases in the UX? .....	4
Describe any libraries you'll be using and share your reasoning for including them? .....	4
Describe how you will implement Google Play Services or other external services? .....	5
Next Steps: Required Tasks.....	5
Task 1: Project Setup .....	5
Task 2: Database .....	5
Task 3: Network .....	5
Task 4: Implement UI for each Activity and Fragment.....	5
Task 5: Responsive Design .....	5
Task 6: Pull Data .....	6
Task 7: Handle Error Cases .....	6
Task 8: Implement Google services.....	6
Task 9: Implement Test cases .....	6
Task 10: Notification .....	6

## Description

A handy travel application that would help you organize and plan your trip itinerary. It gives you the opportunity to create different bucket lists for each place you're planning to visit and add all you want to do in the specific place. Not only does it help you plan your trip but it also gives you insights about the top things to do in each city. Finally it helps you to identify monuments through your phone's camera and also sends you a notification when you're near an interest point you were planning on visiting.

## Intended User

This application is aimed to travelers who want to organize their own itineraries and discover new places in each city

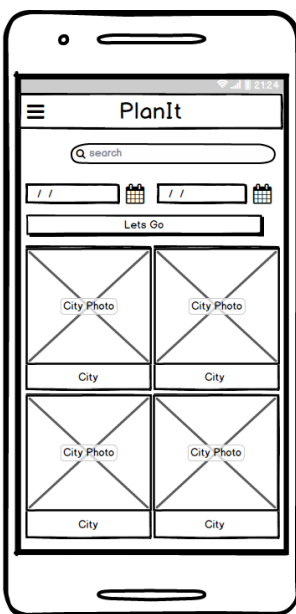
## Features

This application will be written solely in Java and conforming all common standards. Android Studio 3.4.1 and Gradle 5.1.1 will be used.

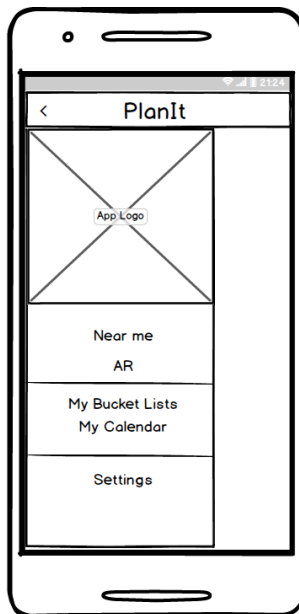
- Browse the most popular destinations
- Browse top things to do
- Details about monuments and interesting activities to do in each city
- Create bucket lists for each city planned to visit
- Organize your bucket lists in a calendar
- AR to identify monuments through the camera
- Get notification when the user is near a point of interest

# User Interface Mocks

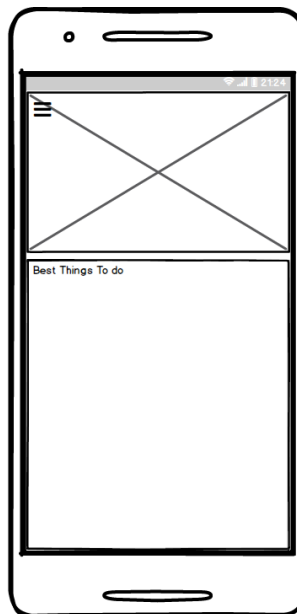
Screen 1



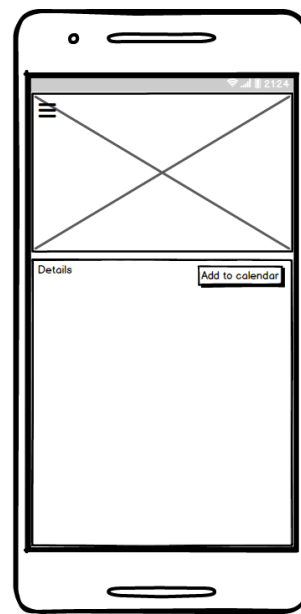
Screen 2



Screen 3



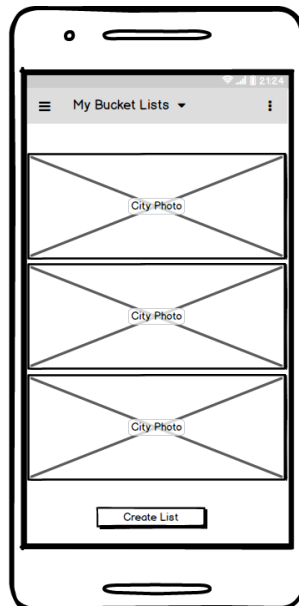
Screen 4



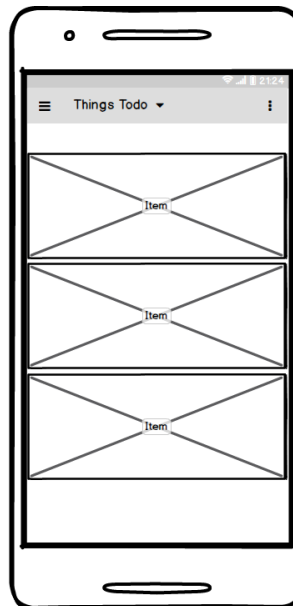
Screen 5



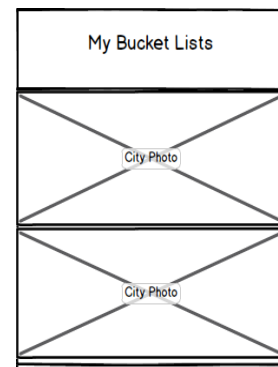
Screen 6



Screen 7



Widget



Screen 1: Main Activity shows most popular destinations  
Screen 2: Displays main navigation bar with all features  
Screen 3: Best things to do in a specific city  
Screen 4: Details of specific activity  
Screen 5: Calendar activity to manage the itinerary  
Screen 6: Bucket List activity to view all activities stored in a bucket list or create a new bucket list  
Screen 7: List of items in a bucket list  
Widget: Home Screen widget to show all created bucket lists

## Key Considerations

### How will your app handle data persistence?

- 1- Use Room and LiveData (when necessary) to store bucket lists information to view it offline
- 2- Use Shared preferences to store apps settings

### Describe any edge or corner cases in the UX?

- Unstable or missed network connection
  - Adding a broadcast receiver to be triggered when connectivity is changed
  - On disconnection stop pulling from servers and add an error message
- Device orientation
  - Save UI state while device is being rotated
- Interstitial Ads

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

- **Retrofit 2.3.0:** For network API requests and JSON parsing
- **Material Intro Screen 0.0.5:** For Intro screen
- **Picasso 2.71828:** For Image loading

**Describe how you will implement Google Play Services or other external services?**

- 1- **Firestore:** for authentication, crash reporting and analytics
- 2- **Google AdMob:** for showing adds in the app
- 3- **Google Vision:** to detect landmarks using camera
- 4- **Google Place:** to get nearby places using AsyncTask

## Next Steps: Required Tasks

### Task 1: Project Setup

- Configure libraries
- Create a debug and release variant
- Setup Gradle dependencies

### Task 2: Database

- Database (Room)
- Data Models

### Task 3: Network

Implement a service which provides all necessary network API requests.

- Data parsing
- Data syncing service

### Task 4: Implement UI for each Activity and Fragment

- Setup basic UI structure, Navigation Drawer
- Build top part of each activity/fragment (AppBar)
- String and Image resources
- Setting Navigation Graph
- Gestures/Transitions
- Build UI element

### Task 5: Responsive Design

Adapt layout for Tablet support

### **Task 6: Pull Data**

Integrate Retrofit to pull data from different servers

### **Task 7: Handle Error Cases**

- Empty views
- Error message
- No Internet Connectivity

### **Task 8: Implement Google services**

- Google Places
- Google Mobile Vision
- Google AdMob
- Firebase authentication
- Firebase Crash Reports
- Firebase Analytics

### **Task 9: Implement Test cases**

- Write Unit tests
- Write Espresso tests for UI

### **Task 10: Notification**

Implement notification system which notify users when they are near a monument/landmark that they want to visit

### **Task 11: Others**

- Home Screen Widget (To show bucket lists)
- Accessibility
- RTL

### **Task 9: Final Testing**

- Rotation
- Phone vs Tablet (Multiple Devices)
- Performance

### **Task 10: Generate and Deploy**

- Generate app flavor keys

- Create Google Play Account
- Follow steps from Android Developers launch checklist
- Create APK
- Create Google Play image and text
- Push APK