How to Use this Template

1. Create a new document, and copy and paste the text from this template into your new document [Select All → Copy → Paste into new document]
2. Name your document file: “**Capstone_Stage1**”
3. Replace the text in green

GitHub Username: CurtesMalteser

António Bastião

Description

This app uses Google Maps API to find the closest points of interest.

The PingPoinz main goal, is to show active events in the vicinity the user location. Example of this are Summer Festivals, sportif events, museum expositions or restaurants that are doing special offers on their menus.
All this information in only one app.

Intended User

The intended users Travelers are or locals that search for events that usually are spread through different platforms or on maps apps that usually don't show what's happening around. So, somehow are difficult to find.

Features

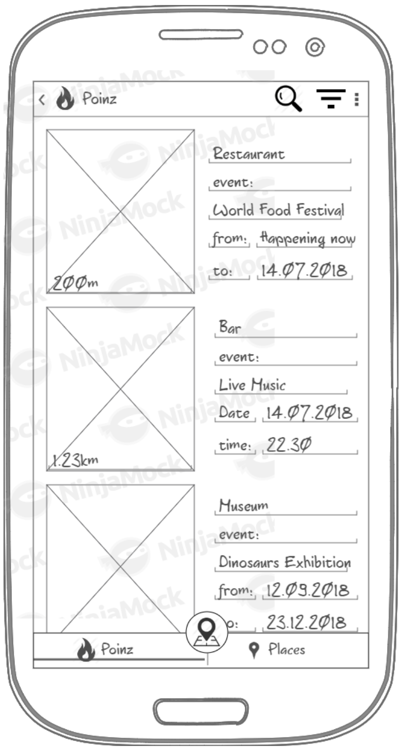
Main features:

- List points of interest for current user location
- Show pictures from this places
- Compiles the information about current events in this places, or occasional that tend to no be list on maps or travel apps.
- Get photos from Instagram for hashtags related to the place name
- Allow to search for specific places
-

User Interface Mocks

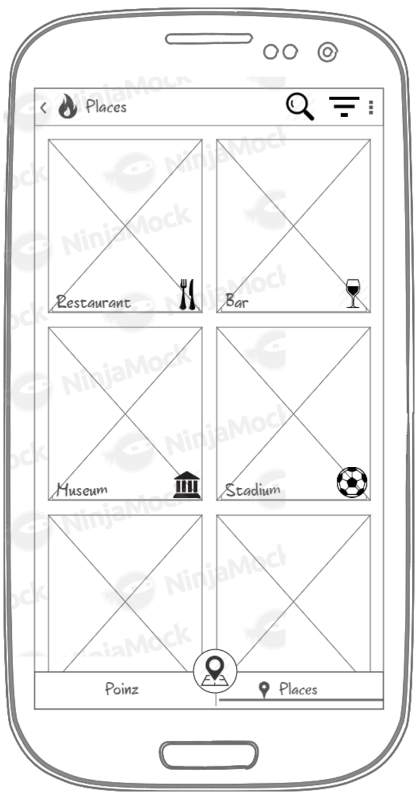
These can be created by hand (take a photo of your drawings and insert them in this flow), or using a program like Google Drawings, www.ninjamock.com, Paper by 53, Photoshop or Balsamiq.

Screen 1



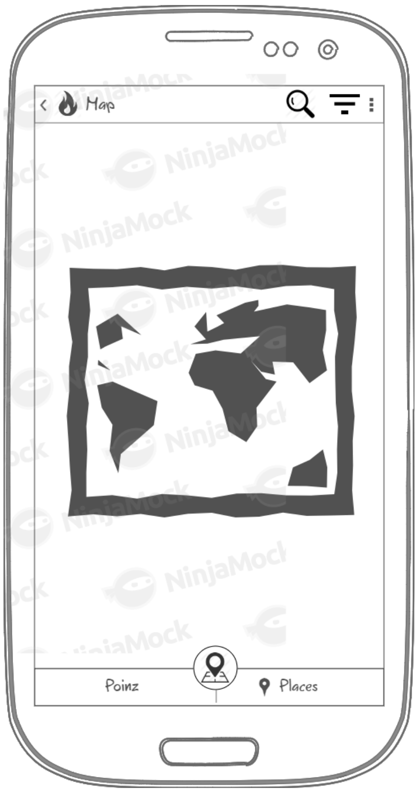
Replace the above image with your own mock [click on the above image, then navigate to Insert → Image…]
Provide descriptive text for each screen

Screen 2



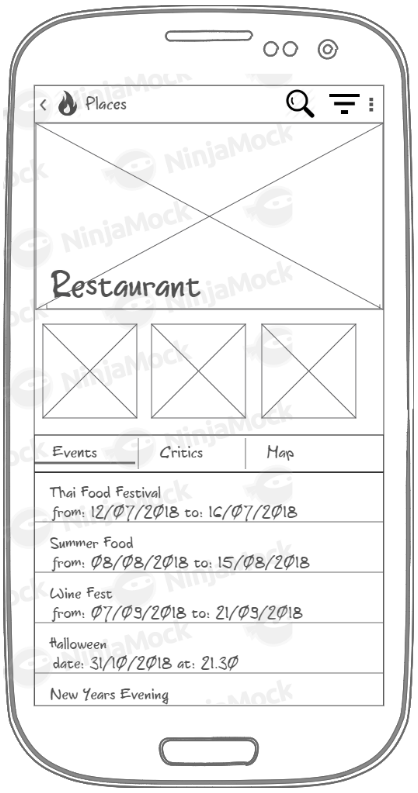
Replace the above image with your own mock [click on the above image, then navigate to Insert → Image…]
Provide descriptive text for each screen

Screen 3



Replace the above image with your own mock [click on the above image, then navigate to Insert → Image…]
Provide descriptive text for each screen

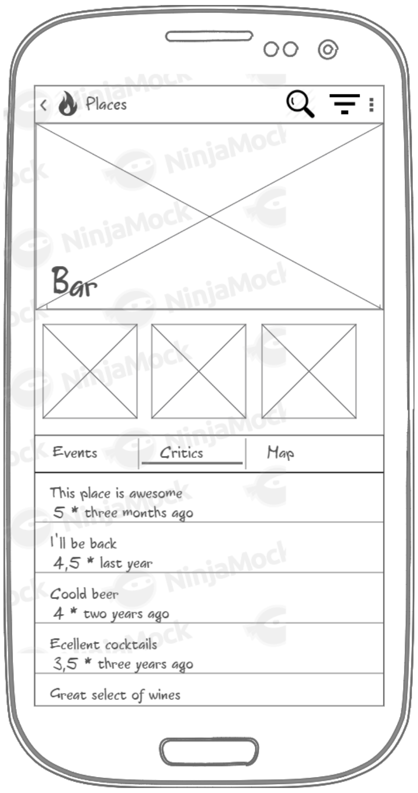
Screen 4



Thai Food Festival
from: 12/07/2018 to: 16/07/2018
Summer Food
from: 08/08/2018 to: 15/08/2018
Wine Fest
from: 07/09/2018 to: 21/09/2018
Halloween
date: 31/10/2018 at: 21.30
New Years Evening

Replace the above image with your own mock [click on the above image, then navigate to Insert → Image…]
Provide descriptive text for each screen

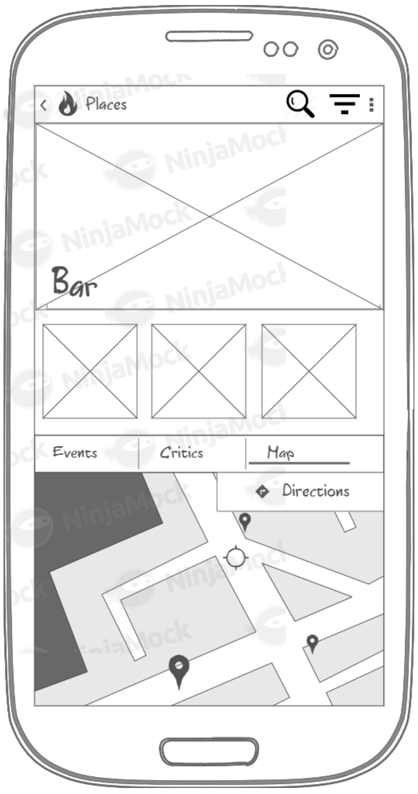
Screen 5



Replace the above image with your own mock [click on the above image, then navigate to Insert → Image…]

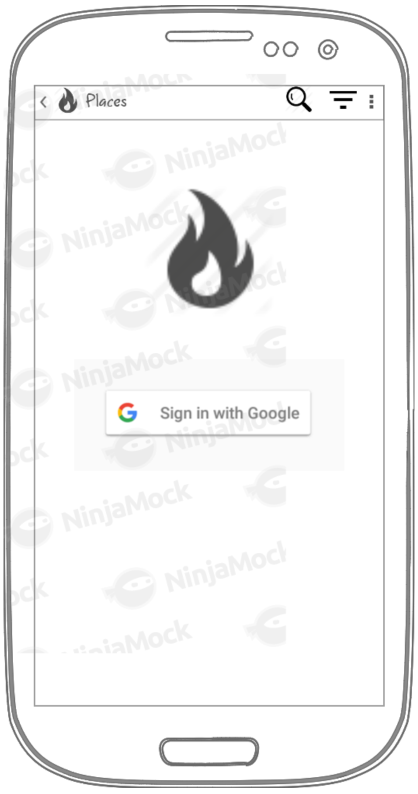
Provide descriptive text for each screen

Screen 6



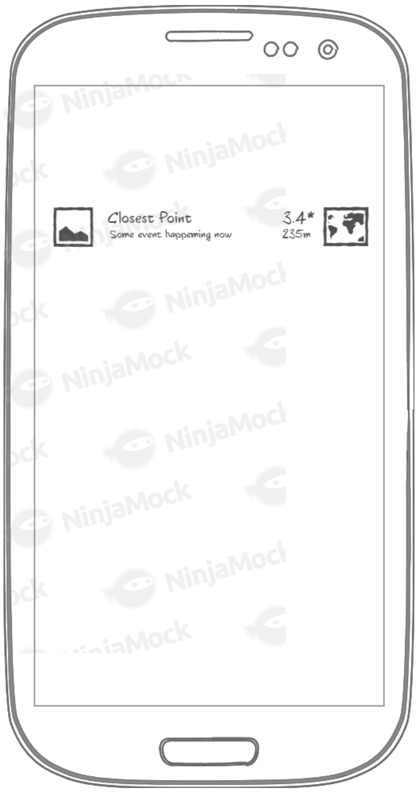
Replace the above image with your own mock [click on the above image, then navigate to Insert → Image…]
Provide descriptive text for each screen

Screen 7



Replace the above image with your own mock [click on the above image, then navigate to Insert → Image…]
Provide descriptive text for each screen

Screen 8



Replace the above image with your own mock [click on the above image, then navigate to Insert → Image…]
Provide descriptive text for each screen

Add as many screens as you need to portray your app’s UI flow.

Key Considerations

How will your app handle data persistence?

Describe how your app with handle data. (For example, will you build a Content Provider or use Firebase Realtime Database?)

Describe any edge or corner cases in the UX.

For example, how does the user return to a Now Playing screen in a media player if they hit the back button?

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

Google Maps API because the app is based on maps and location. Android Architecture Components because to show knowledge on MVVM is a must for a professional Android Developer and that's the goal of this app, to be professional. Retrofit just because is the best library is a REST Client for Android , Dagger 2 because I want to understand how it works and is a must on MVVM or MVP (dependency injection), Butterknife because I like and is more easy to implement than DataBinding from my point of view.
Firebase, because I intend to log users with in the app with FirebaseUI and also make my own API based on Firebase realtime database.

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

Google Maps API to show maps and places.
Firebase that I'll search in the websites from the canton of the Valais in Switzerland where I'm currently living and compile the events for the next two or three months.

Next Steps: Required Tasks

This is the section where you can take the main features of your app (declared above) and break them down into tangible technical tasks that you can complete one at a time until you have a finished app.

Task 1: Project Setup

Write out the steps you will take to setup and/or configure this project. See previous implementation guides for an example.

You may want to list the subtasks. For example:
- Configure libraries
- Something else

If it helps, imagine you are describing these tasks to a friend who wants to follow along and build this app with you.

Task 2: Implement UI for Each Activity and Fragment

List the subtasks. For example:
- Build UI for MainActivity
- Build UI for something else

Task 3: Your Next Task

Describe the next task. For example, “Implement Google Play Services,” or “Handle Error Cases,” or “Create Build Variant.”

Describe the next task. List the subtasks. For example:

- Create layout
- Something else

Task 4: Your Next Task

Describe the next task. List the subtasks. For example:

- Create layout
- Something else

Task 5: Your Next Task

Describe the next task. List the subtasks. For example:

- Create layout
- Something else

Add as many tasks as you need to complete your app.

Submission Instructions

- After you’ve completed all the sections, download this document as a PDF [File → Download as PDF]
    - Make sure the PDF is named “**Capstone_Stage1.pdf**”
- Submit the PDF as a zip or in a GitHub project repo using the project submission portal

If using GitHub:

- Create a new GitHub repo for the capstone. Name it “**Capstone Project**”
- Add this document to your repo. Make sure it’s named “**Capstone_Stage1.pdf**”