Description
Intended User
Features
User Interface Mocks
Screen 1
Screen 2
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.
Describe how you will implement Google Play Services.
Next Steps: Required Tasks
Task 1: Project Setup
Task 2: Implement UI for Each Activity and Fragment
Task 3: Your Next Task
Task 4: Your Next Task
Task 5: Your Next Task

GitHub Username: piotrprus

My storage place

Description

The application will store the data about items that we store in boxes in our storages like basements, wardrobes, garages etc

This app solve the problem of searching the whole basement and opening all the boxes to find one precious item.

Intended User

For all people with basements, garages and attics. For everyone that want to keep order of their stuff

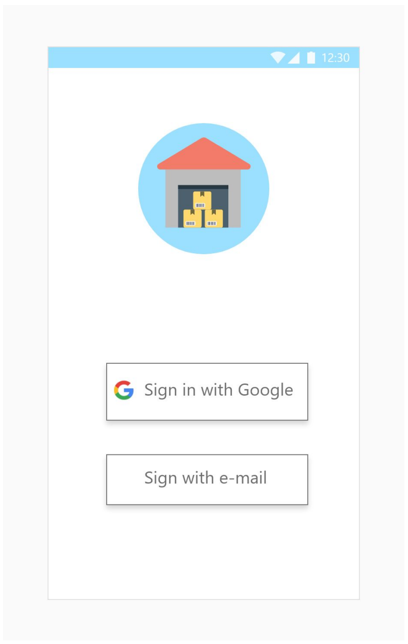
Features

- Store the data in cloud
- Takes pictures

- Scan the QR code/number

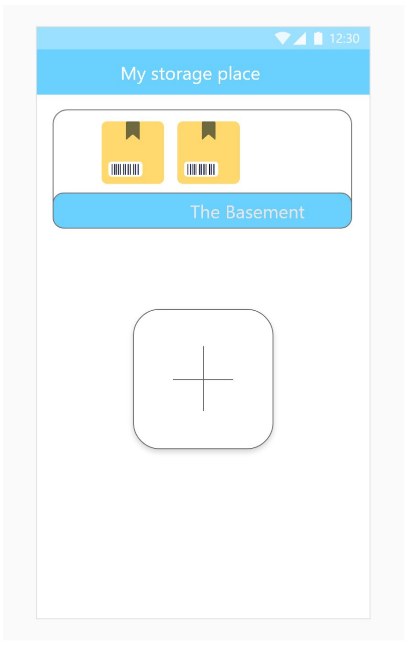
User Interface Mocks

Screen 1



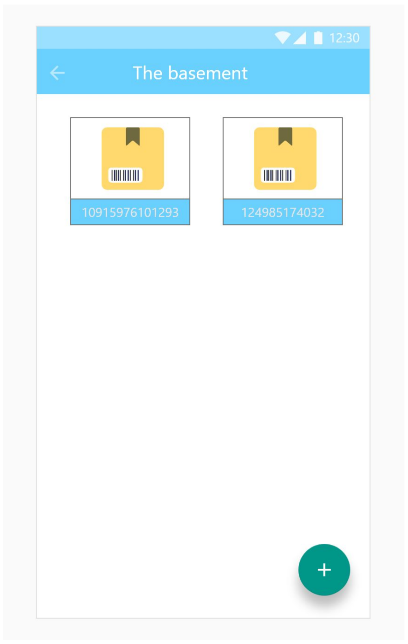
Sign-in screen with app logo and google sign-in button

Screen 2



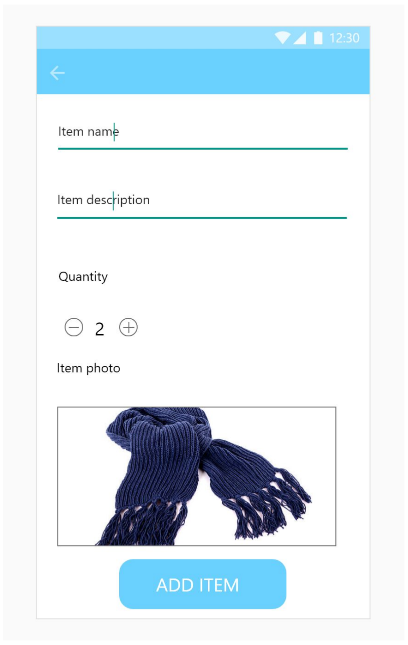
Main screen with list of places where user stores their boxes. Each listItem will have label and visual information about number of boxes included

Screen 3



This detail view will present grid with all the boxes saved in user storage. There is a label with unique number on each box to identify it in the future. After clicking on the box the new screen will appear with list of items included in it.

Screen 4



Add item screen that includes name label, description, quantity and image. After hitting the ADD button user will see the item in the list that includes every item in the box.

Key Considerations

How will your app handle data persistence?

I plan to use Firebase Realtime Database

Describe any edge or corner cases in the UX.

User will have a possibility to add a new item to his predefined box using Floating Action Button. Additionally there will be possibility to take a picture of this item.
I would like to make a QR code/barcode scanner to make search of boxes easier in the real life.

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

- Picasso to handle to load the pictures and cache them
- Lifecycle to handle lifecycle-aware components
- DataBinding to bind xml data and ViewModel class
- RxJava to handle Asynchronous actions in application
- Google vision library to handle QR scan

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

Firebase analytics service to analyse the user experience
Google Play Services to integrate google sign-in

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

- App is written solely in the Java Programming Language
- App utilizes stable release versions of all libraries, Gradle, and Android Studio

Task 2: Implement UI for Each Activity and Fragment

List the subtasks.

- Build UI for SplashScreen
- Build UI for sign-in screen
- Build UI for MainActivity
- Build UI for Add item screen
- Build UI for Add new box screen

Task 3: Implement Firebase Database

- Implement firebase database to add/delete boxes/items
- make a service to retrieve informations after login

Task 4: Implement logic for adding new item

- Make an adding new item/box possible in app
- Create layout for adding item
- handle all cases of no text input, no quantity input etc
- handle back button event, ask about saving item or close without saving

Task 5: Introduce QR/barcode scanner

- Make scanning possible with use of google vision library
- check if scanned code is recognized
- make a qr code generator for each box in “my storage” 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**”