Capstone Stage 1

GitHub Username: Mohamed.aly.karim

EGY BANKS TEST

Description

This App Helps Egyptian Graduated People to Pass The Local Banks Selection Exams (Firstly it will contains tests of two banks - Banquemisr and Alahly). It Simulate These Tests And Contains Some Quiz In IQ, English And Banking ..etc

- App is written solely in the Java Programming Language
- App utilizes stable release versions of all libraries, Gradle, and Android Studio:
    - Android studio Version 3.1.3
    - `Picasso:2.71828`
    - Firebase-storage:16.0.1
    - Firebase-auth:16.0.2
    - Firebase-database:16.0.1

Intended User

Egyptian Graduated who looking to work on local bank

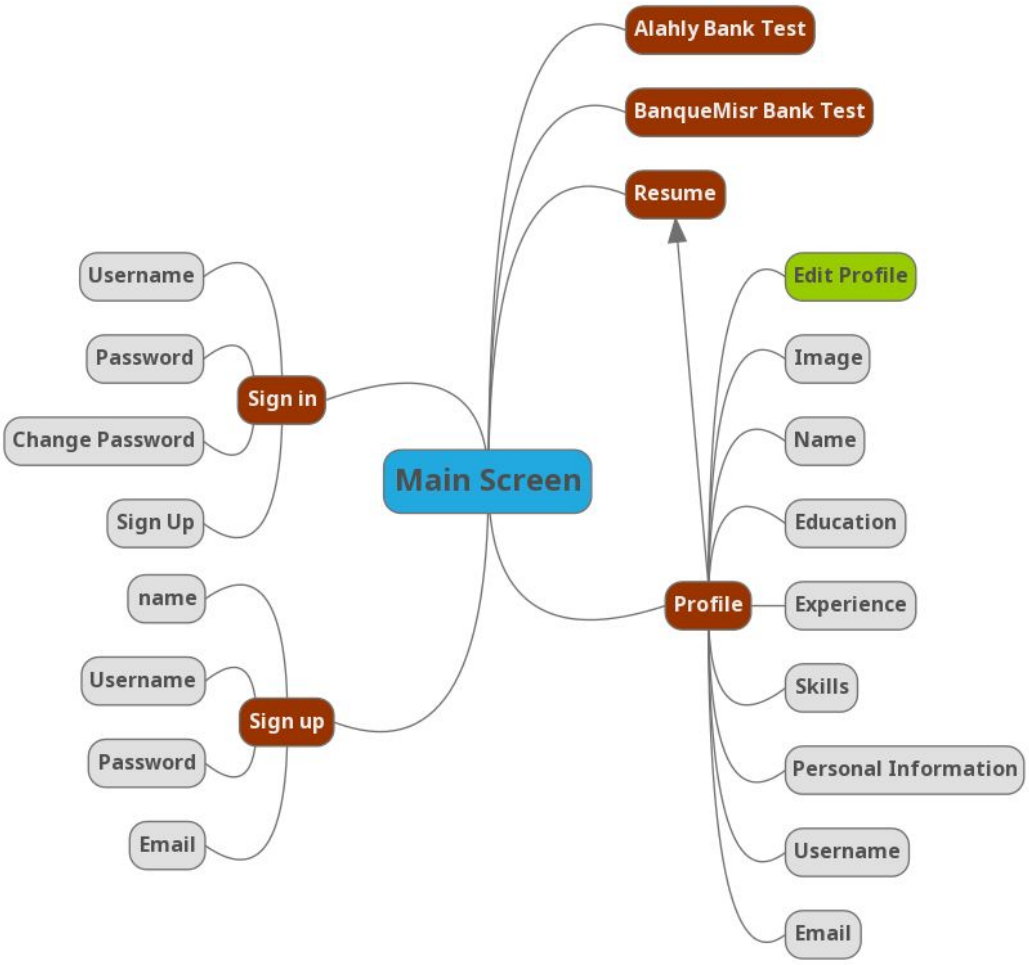
Features

- Simulate Bank Misr and Al ahly Selection Test.
- Contain Resume for Intended User.
- Contain Login And Registration.
- Can Show Other people User Resume.

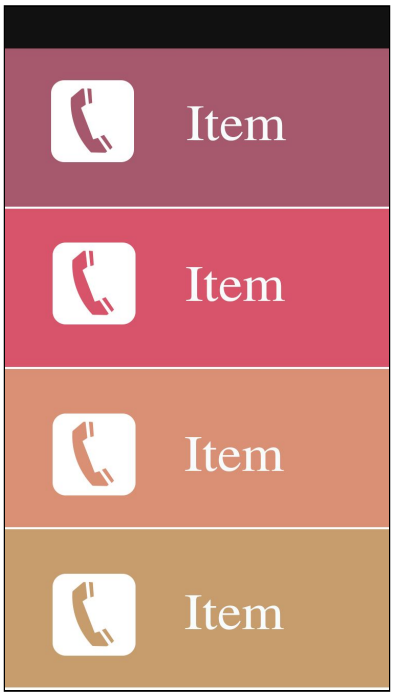
User Interface Mocks

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.

App Chart:

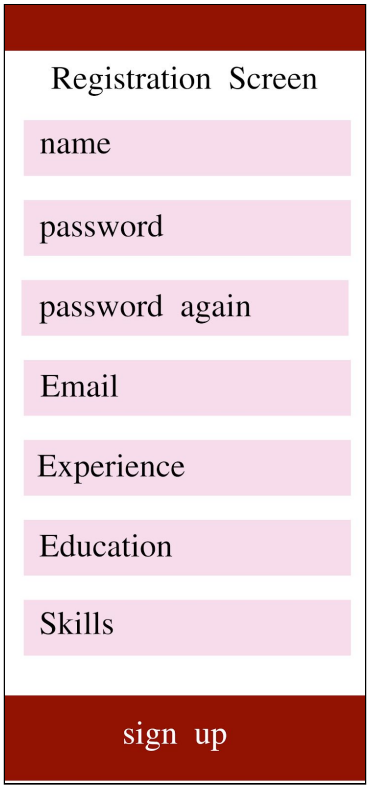

Alahly Bank Test
BanqueMisr Bank Test
Resume
Username
Password
Sign in
Change Password
Sign Up
Main Screen
Edit Profile
Image
Name
Education
Profile
Experience
Skills
Personal Information
Username
Email
name
Username
Sign up
Password
Email

Screen 1 - Main Screen



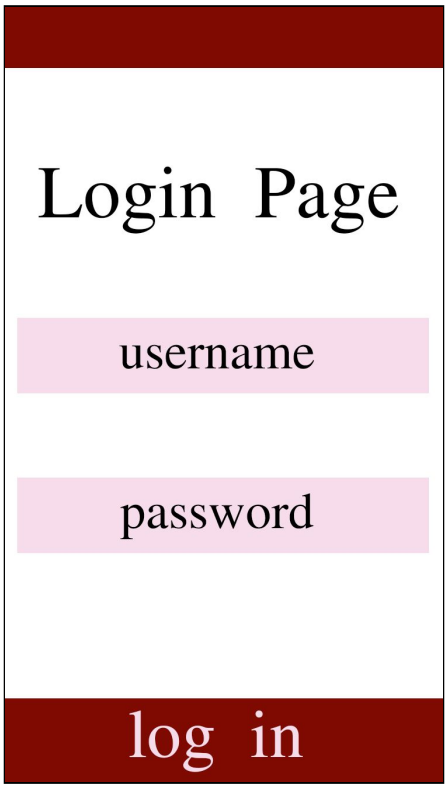
the main menu:
It contains items that the user navigate to it like Resume, Login , Logout, Specific Quiz.

Screen 2 - Sign up



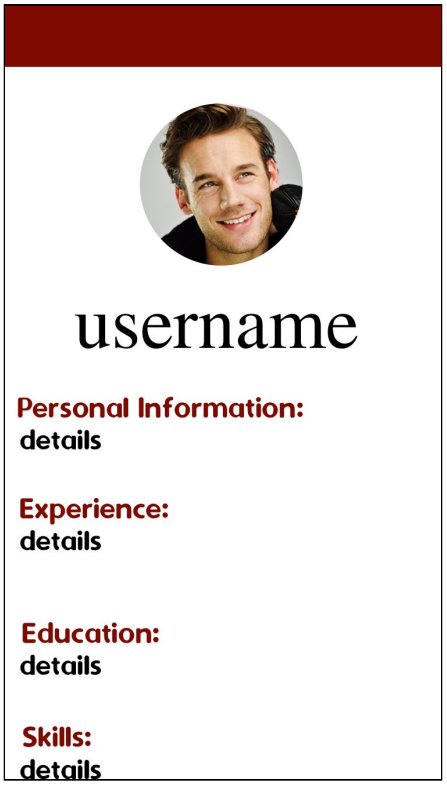
Sign up Page

Screen 3 - Login Page

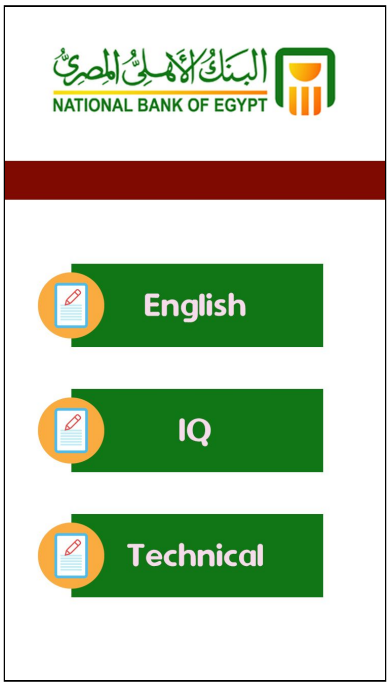


Login Page

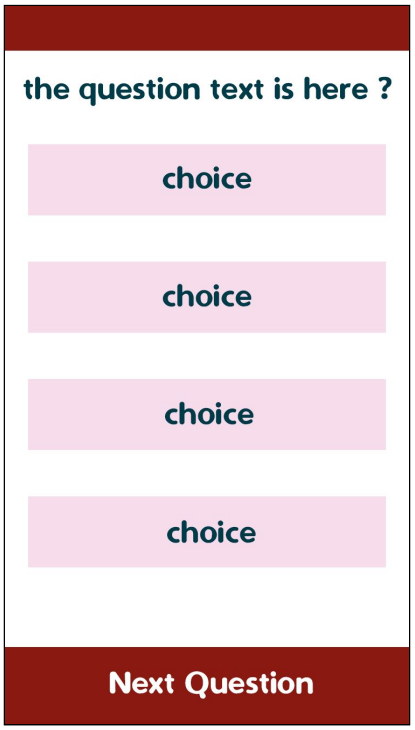
Screen 4 - Resume and profile page



Screen 5 - Bank Test Page

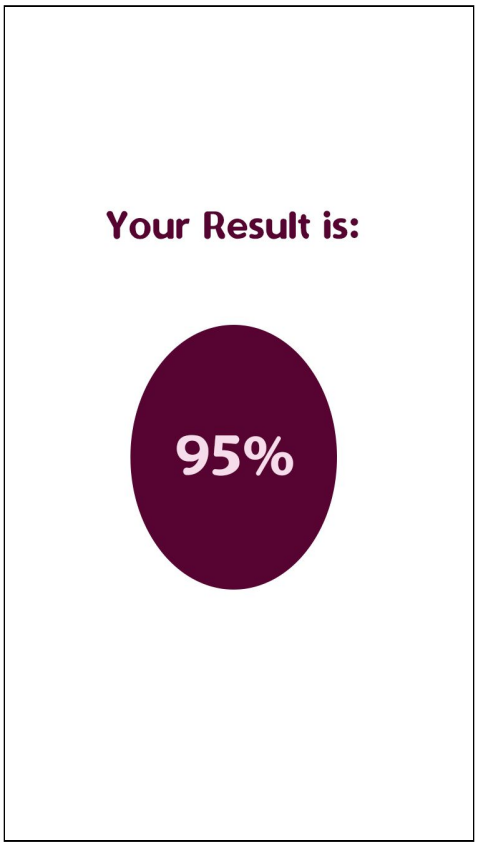


Screen 6 - Quiz Page

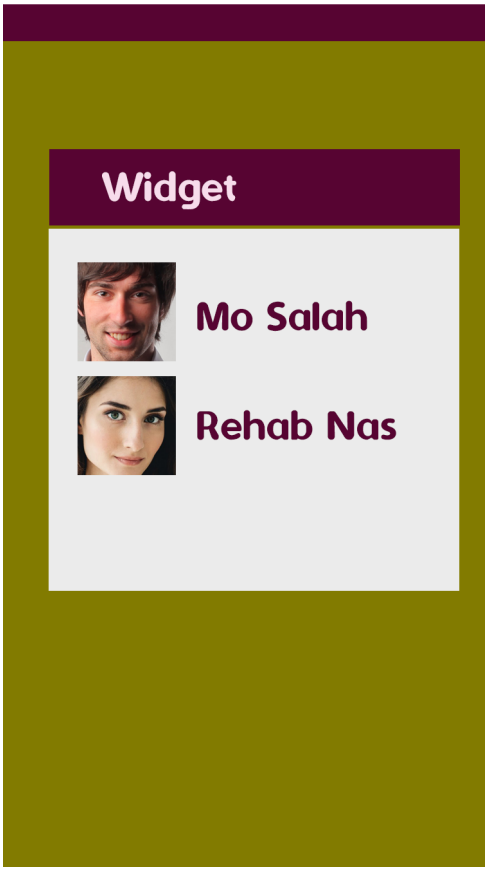


this screen shows the simulation test

Screen 7 - Result Screen



Screen 8 - Widget Screen



Key Considerations

How will your app handle data persistence?

the app contains quiz session that will saved in SQLite database by content provider. the quiz questions and answers, authentication and user profile will saved in firebase database.

Describe any edge or corner cases in the UX.

- Since the app will use sessions with quiz, The Quiz screen need to continue the session when the user restart the phone.

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

Picasso: It's good to use a great library to retrieve photo from the internet. I will use it on retrieve and show user profile image.

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

I will use Firebase services:

- **Auth**: to make sign in and sign up in the app
- **realtime database**: to store data like questions and resume info.
- **cloud hosting**: to save users profile images.

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

- Start new Android studio Project.
- configure style and color.
- look for good font add it to assets.
- add required dependencies.

- App theme extends `AppCompat`
- `All app dependencies are managed by Gradle`

Task 2: Implement UI for Each Activity and Fragment

- Build UI for MainActivity.
- Build UI for EditProfileFragment and ShowProfileFragment.
- Build UI for ShowResume Fragment.
- Build UI for QuizFragment
- Build UI for singin and signup
- Build Widget UI

Task 3: Create Adapter

- Create MainScreen Recyclerview Adapter

Task 4: Create Database Classes

- Create All wanted classes for database
- Create Content Provider
- use loader to move its data to its views

Task 5: Create the fragments

- Create EditProfileFragmen.
- Create ShowProfileFragment.
- Create ShowResume
- Create QuizFragment

Task 6: Initialize the firebase services

- handle login and logout (Auth).
- handle realtime database.
- handle cloud storage.

Task 7: Create and initialize Bank Test Page

- create and initialize bank test page

Task 8: Create and initialize Result Activity

- Create and Initialize ResultActivity

Task 9: Create And Initialize Widget

- Create and initialize widget.
- Create IntentService of the widget

Task 10: Make it accessible

- Incorporate content description.
- Enable RTL.
- Group Content

Task 11: Finish the project

- Finish remaining steps.

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**"