

[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: [aya-salama1989](#)

JobEase

Description

JobEase is a simple way to find and post small temporary jobs by community members all around the world. If you are a traveller, student or someone who's looking for extra simple jobs to gain extra money, or you are a business owner who needs to hire temporary workers for small tasks every now and then. JobEas is the solution for your needs.

Problem

Actually there is two problems, The first one was my brother's, he is a traveller who likes to wander all around the world every now and then, Sometimes he ran out of money and he needed any nearby small temporary job to give him extra pocket money to complete his journey. The second problem was my best friends' problems, They have a couple of private businesses where they sometimes need workers to sort goods or help them clean stores.

Proposed Solution

The solution was a mobile application, that posts possible jobs and notify the nearest available worker to propose for them.


Intended User

Anyone who needs to hire a worker for small task, or who might need a temporary task that could get him some money.

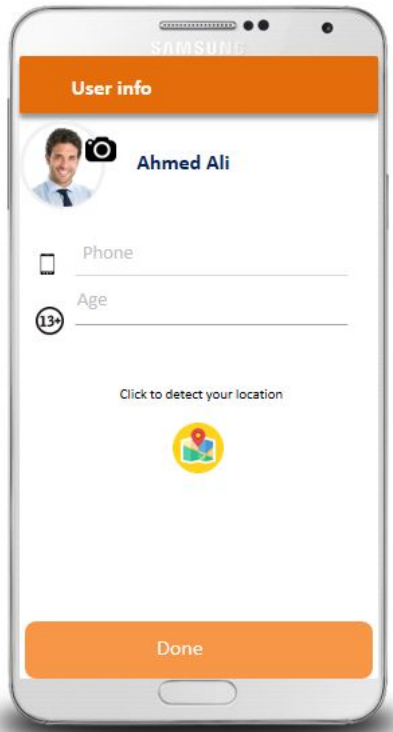
Features

- Post jobs provided by users
- Displays active nearby jobs
- Notify users with nearby jobs
- Determine user's' current Location
- Provide contacts and personal information between users

Mobile User Interface Mocks

Screen 1: Facebook Login	
	This screen will help user login the app using Facebook SDK

Screen 2: Fill user data

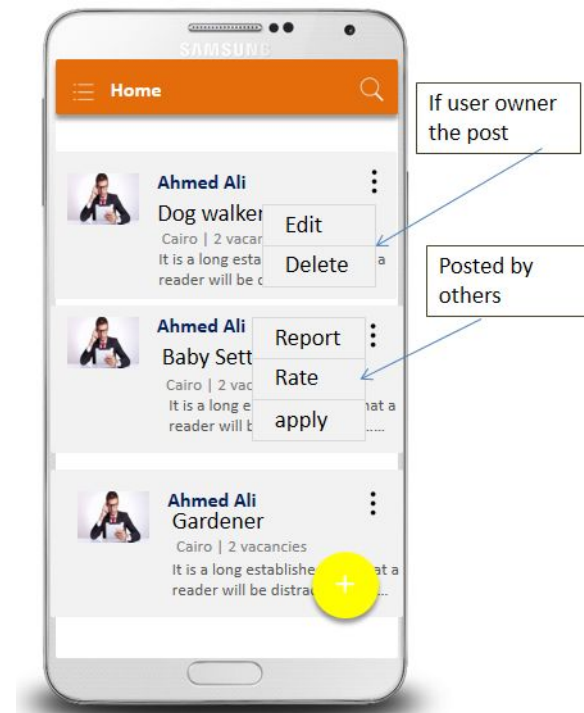


-This Screen appears only if required data wasn't provided by Facebook Login.

-User must provide;

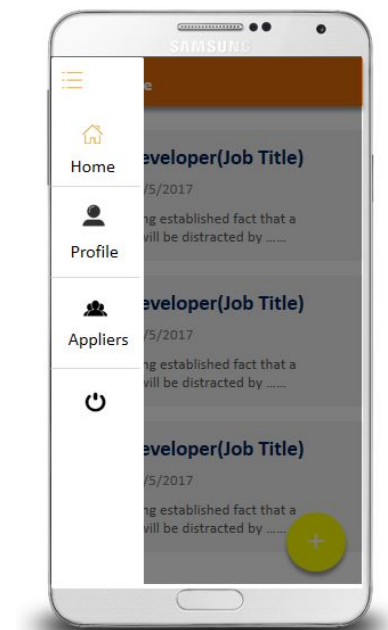
1. name,
2. Age,
3. Phone,
4. Location,
5. an image showing full face,
6. User must be 13+ years old.

Screen 3: Home Screen



Through home screen user can do the following:

- 1- Navigate to Add New Job screen through the floating button
- 2- Open Navigation Drawer
- 3- delete a job post that he owns
- 4- apply to, rate or flag a job post
- 5- scroll the news feed to scan newly posted jobs



User will use navigation bar in home page to navigate to:

1. home screen,
2. Profile Screen,
3. Appliers Screen,
4. Or log-out the application.

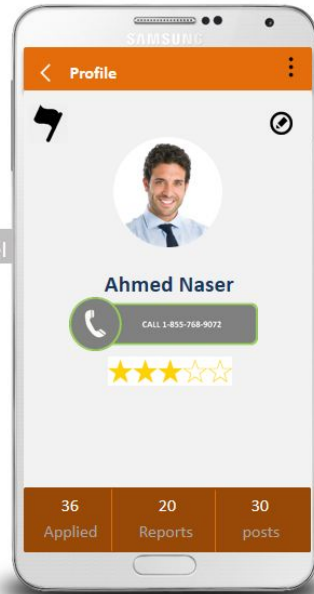
Screen 4: Profile Screen

Report

Write your
report...

Add

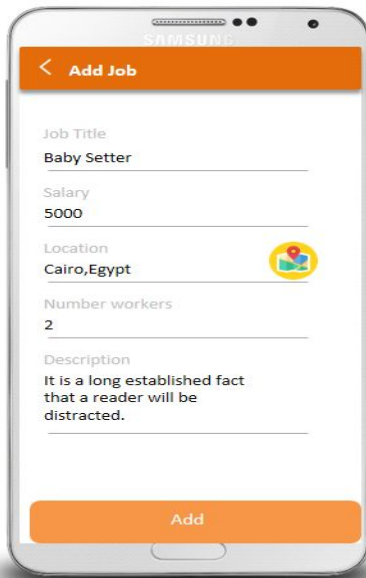
Cancel



In this screen user can:

- 1- view and edit his own profile data.
- 2- view, rate, flag or call other users.

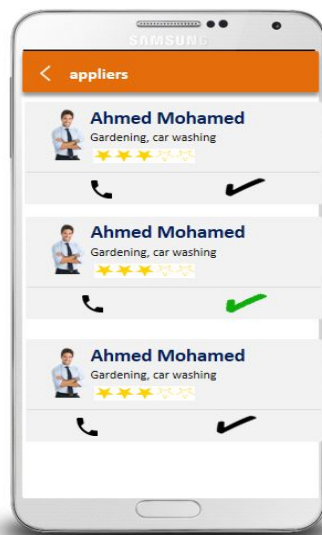
Screen 5: Add New Job



In this screen user can add a new job and provide its,

- 1- location,
- 2- salary,
- 3- brief description about the task.

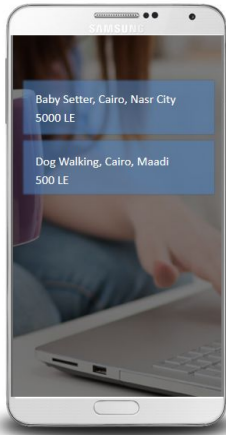
Screen 6: Appliers Screen



In this screen user can view nearby workers who applied to his job, also user can:

- 1- Call appliers
- 2- Accept them for the job

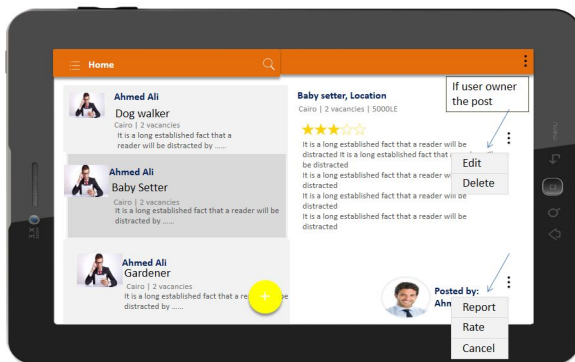
Screen 7: Widget UI



Widgets will show latest jobs, it will update every 6 hours and every item will redirect user to job post detail

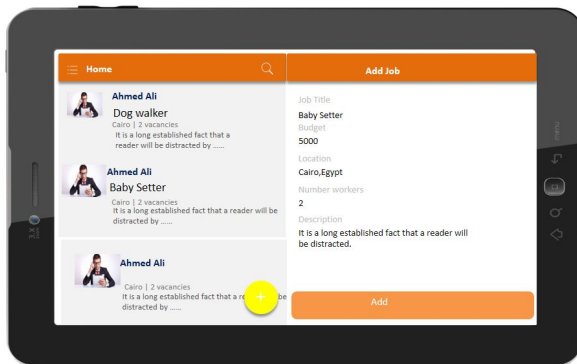
Tablet User Interface Mocks

Screen 1: Home screen and Add post



Combined features of both screens with mild changes

Screen 2: Home screen and post details



Combined features of both screens with mild changes

Key Considerations

How will your app handle data persistence?

I will get user's data from Facebook and following UserData Screen, and save it to SharedPreferences.

I will use Firebase Database to store new jobs.

I will use IntentService to update Widgets daily.

Describe any edge or corner cases in the UX.

Whenever a user clicks on a widget item it will redirect him to the specified job post details.

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

- 1- FaceBook SDK to handle a passwordless login
- 2- ButterKnife for easier data binding
- 3- support design to use material design components
- 4- Picasso to handle images loading and caching
- 5- Google location services to handle location detection
- 6-Google AdMob to handle google ads in Home screen

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

1- Location detection services, I will use it to:

A- detect worker location to provide it in the UI

B- detect job location to provide it in the UI

2- Google AdMob , I will View a banner Ad in the bottom of home page for both UI Mobile and Tablet

Next Steps: Required Tasks

Task 1: Project Setup

At this phase I will:

- Create GitHub repository
- Add application style, colors and drawables.
- Configure previously mentioned libraries
- Add a couple of utility classes and create data models
- Decide if I am going to use an online back-end service or use a locale alternative

Task 2: Implement UI for Each Activity and Fragment

- Build UI for Passwordless Login Screen
- Build UI for Fill User Data
- Build UI for Home Screen
- Build UI for profile Screen
- Build UI for Add new job Screen
- Build UI for Appliers Screen
- Build UI for Tablet Screens
- Test UI Navigation, Animations and Action-items interaction completely

Task 3: Implement and Test Facebook login/ user data collection

- Implement Facebook login.
- Populate and save mandatory user data if available.
- View Fill User Data Screen to if mandatory data wasn't supported completely.
- Save collected data in database using content provider.

Task 4: Use back-end APIs to retrieve data

SubTasks

- Get data from online APIs or a solution mimicking the same function ex: GCE.
- Parse and populate data in the specified views.
- Test the app UI interaction with the retrieved data.

Task 5: Implement Widgets

- Implement a widget that shows users the latest 3 posts on news feed

Task 6: Test and Finalize the Application

- Test the application functionality and UI
- If there's any possible enhancements to the applications' performance and UI I will execute it.