

SMALL TASKS AT HAND- INCREMENT 3

Ayinala kaushik (Class ID: 4)

Yaswanth Bonda (Class ID: 6)

Tharkin Vankayala (Class ID: 34)

Ravi Teja Yakkala (Class ID: 38)

OBJECTIVES:

- Every new user can register to be as a part of our application by using our Login page and Register Now options.
- Every Employer can give a feedback on the employee who did his assigned job, how well did he do that.
- User can select the location of the job instead of getting it automatically from the device GPS current location.

EXISTING SERVICES/API:

In this third increment we used the GOOGLE MAPS API to make the application user friendly by giving the option to the user/employer to choose his location where to locate an address of the task he is posting.

```
Example: <meta-data
  android:name="com.google.android.geo.API_KEY"
  android:value="API_KEY"/>
<fragment xmlns:android="http://schemas.android.com/apk/res/android"
  android:name="com.google.android.gms.maps.MapFragment"/>
```

Another API we REST API used is the same as the one we used previous increment provided by the MongoLab to access the MongoDB.

Example: <https://api.mongolab.com/api/1/databases?apiKey=mykey>

DETAIL DESIGN OF SERVICES:

USER STORIES:

So far we developed the job posting of a new user by letting him register for free as a new user, giving the user to select his own location of the task he is posting and also the employer can post us feedback about the employee how he did the task well or not. In the later section we will integrate those parts. The picture below shows the stories and progress of our project at the end of the third iteration.

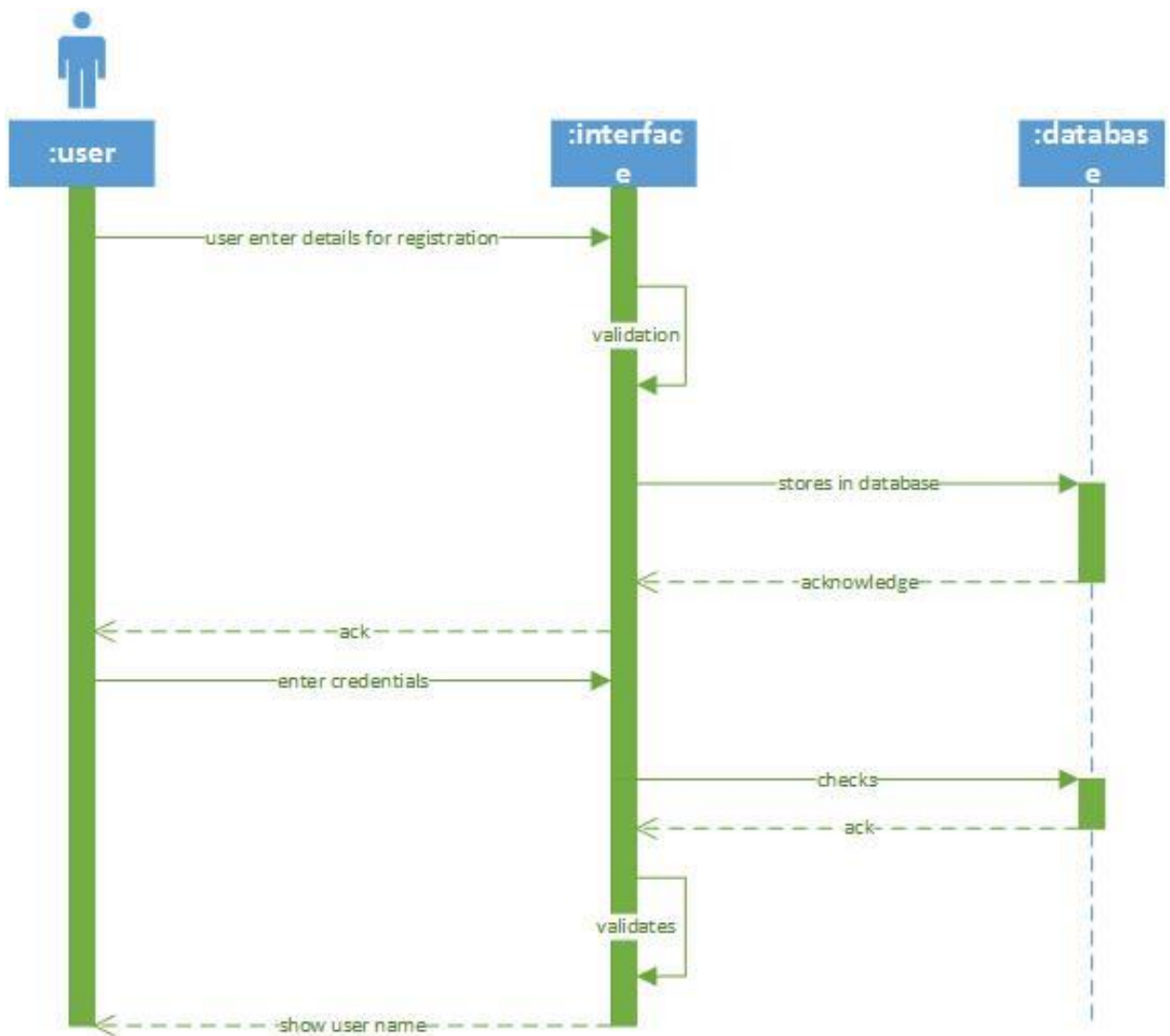
SERVICE DESCRIPTION:

We have developed three services in this increment.

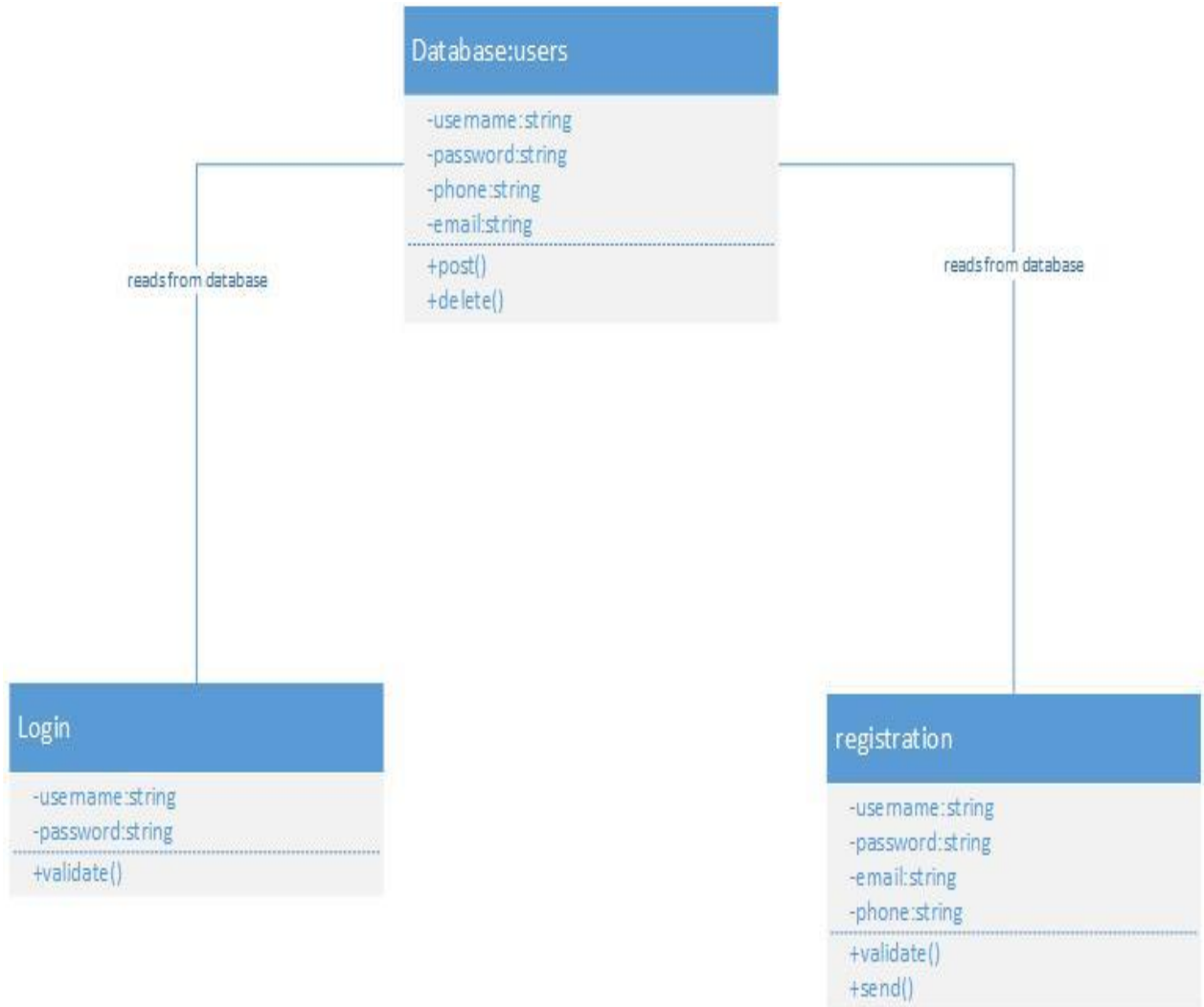
1. LOGIN AND REGISTRATION:

In this service any new person can become a part of our users by registering as a new user in our application. He can fill in all the required details and press submit so that his details get registered with us and give him a new account so that he can login into our application with his given unique ID provided by us.

SEQUENCE DIAGRAM:

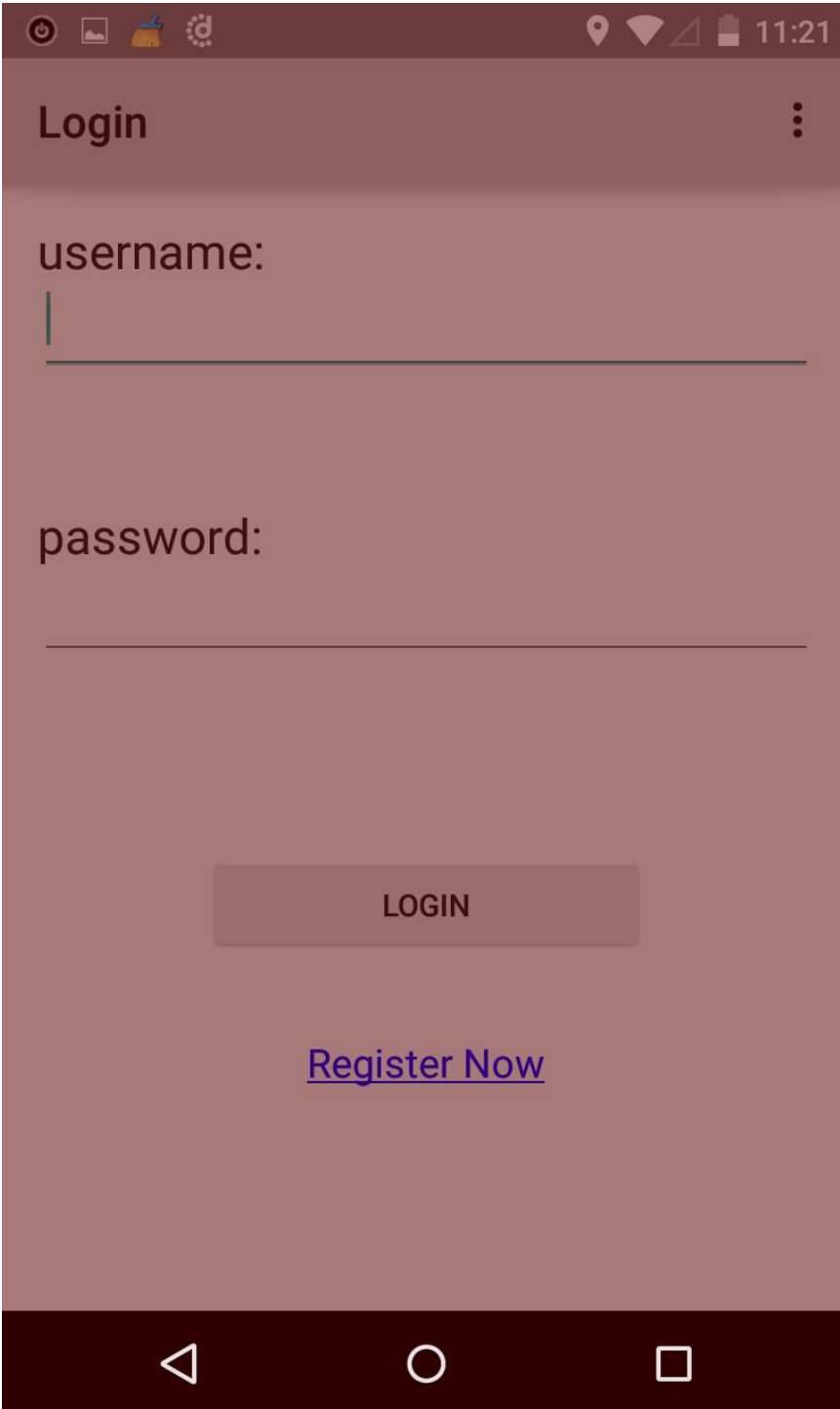


CLASS DIAGRAM:



MOBILE CLIENT INTERFACE:

User can login using the following user interface



The image shows a mobile application interface for a login screen. At the top, there is a status bar with various icons and the time 11:21. Below this is a header bar with the title "Login" and a menu icon. The main area contains two input fields: "username:" and "password:". Below the password field is a "LOGIN" button. At the bottom, there is a link that says "Register Now". The interface is set against a solid blue background.

11:21

Login

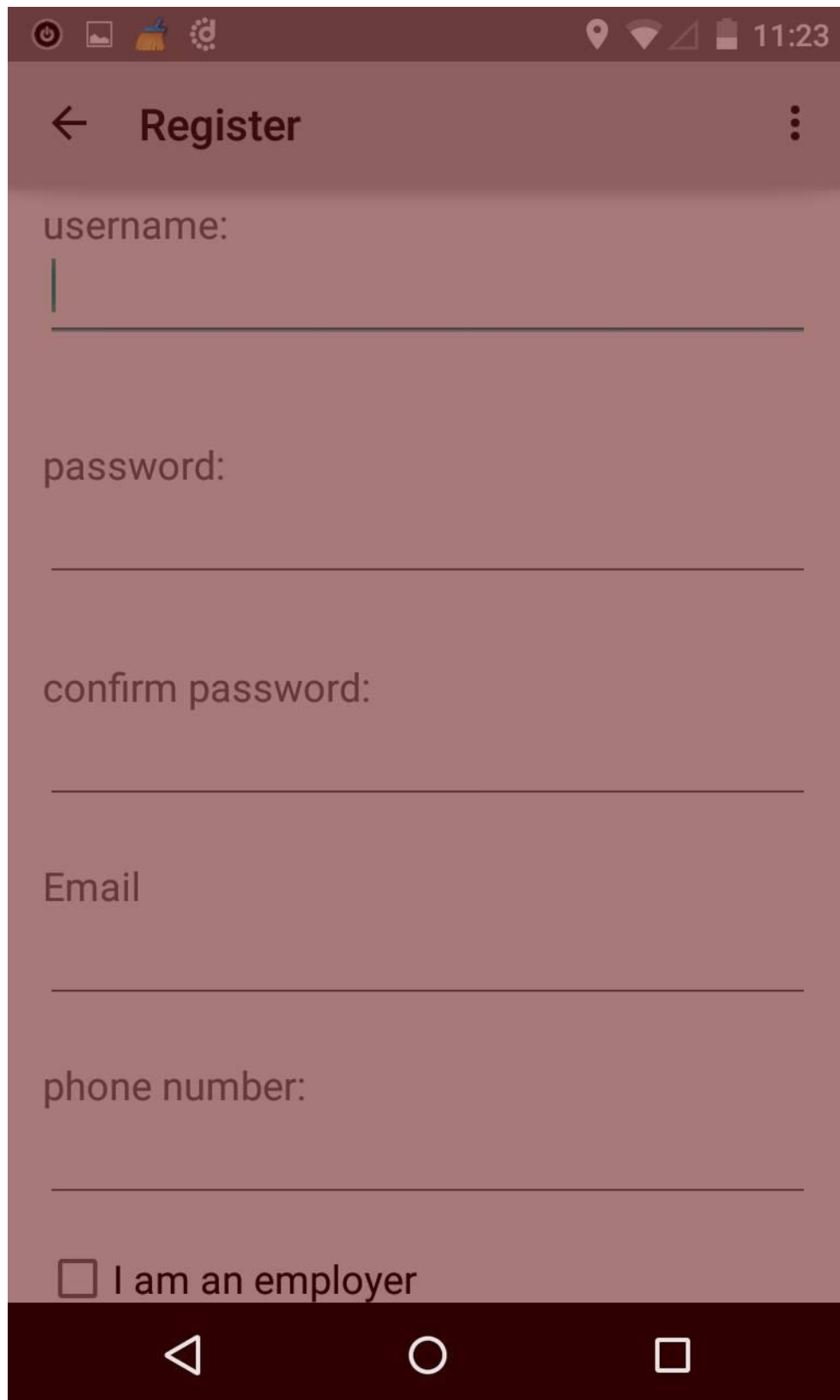
username:

password:

LOGIN

[Register Now](#)

User can register using the below interface

A mobile application registration screen with a dark blue header and a light blue body. The header contains a back arrow, the title 'Register', and a menu icon. The body contains five input fields with labels: 'username:', 'password:', 'confirm password:', 'Email', and 'phone number:'. At the bottom, there is a checkbox labeled 'I am an employer'. The screen is framed by a dark blue status bar at the top and a dark blue navigation bar at the bottom.

Register

username:

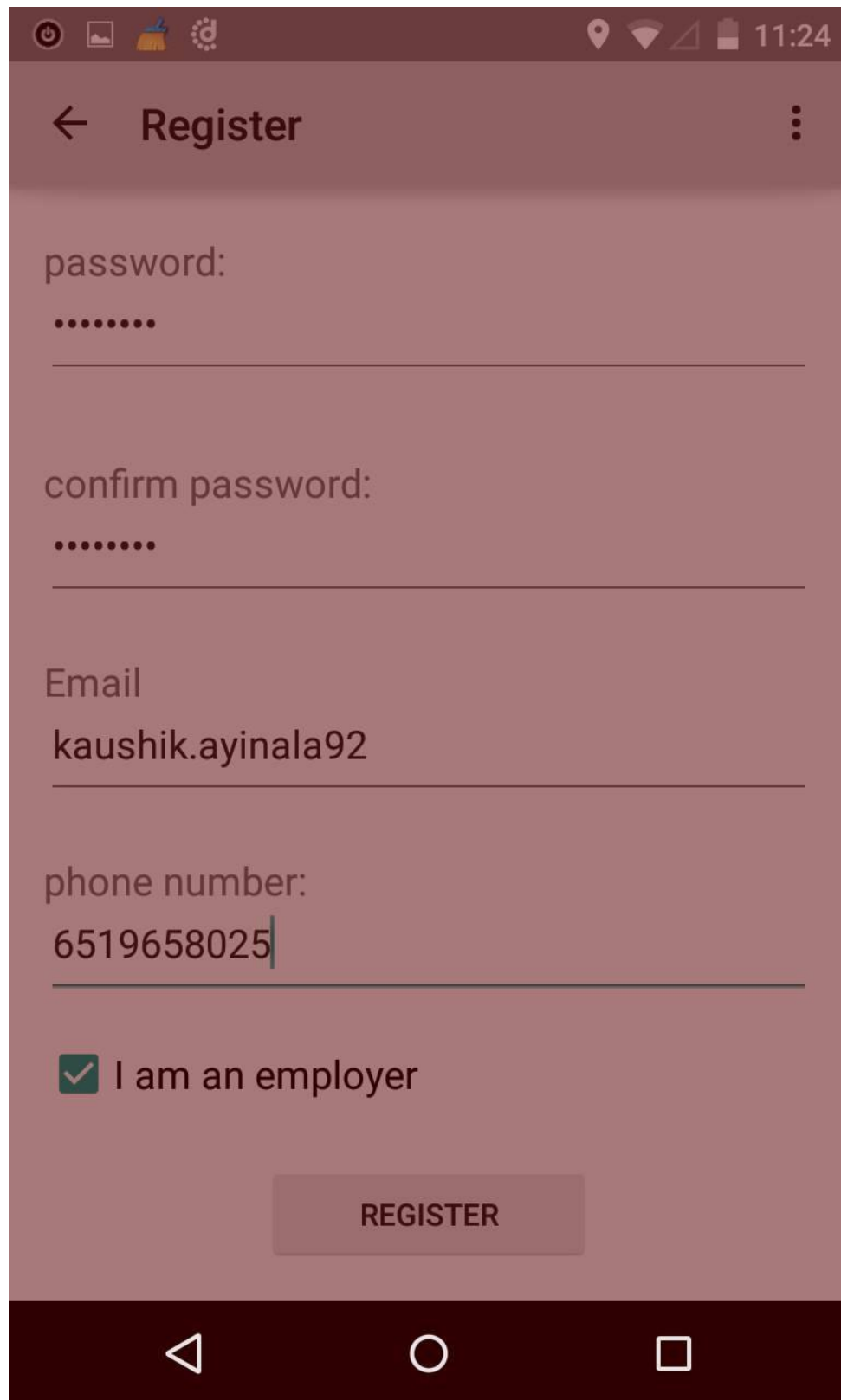
password:

confirm password:

Email

phone number:

☐ I am an employer



A screenshot of a mobile application's registration screen. The screen has a dark blue header with a back arrow, the title "Register", and a menu icon. The background is a light blue gradient. The form contains four input fields: "password:" with a red asterisk, "confirm password:" with a red asterisk, "Email" with the text "kaushik.ayinala92", and "phone number:" with the text "6519658025". Below these is a checkbox labeled "I am an employer" which is checked. At the bottom is a blue "REGISTER" button. The Android navigation bar is at the very bottom.

password:
.....

confirm password:
.....

Email
kaushik.ayinala92

phone number:
6519658025

☒ I am an employer

REGISTER

After user login, we have to integrate modules in next iteration

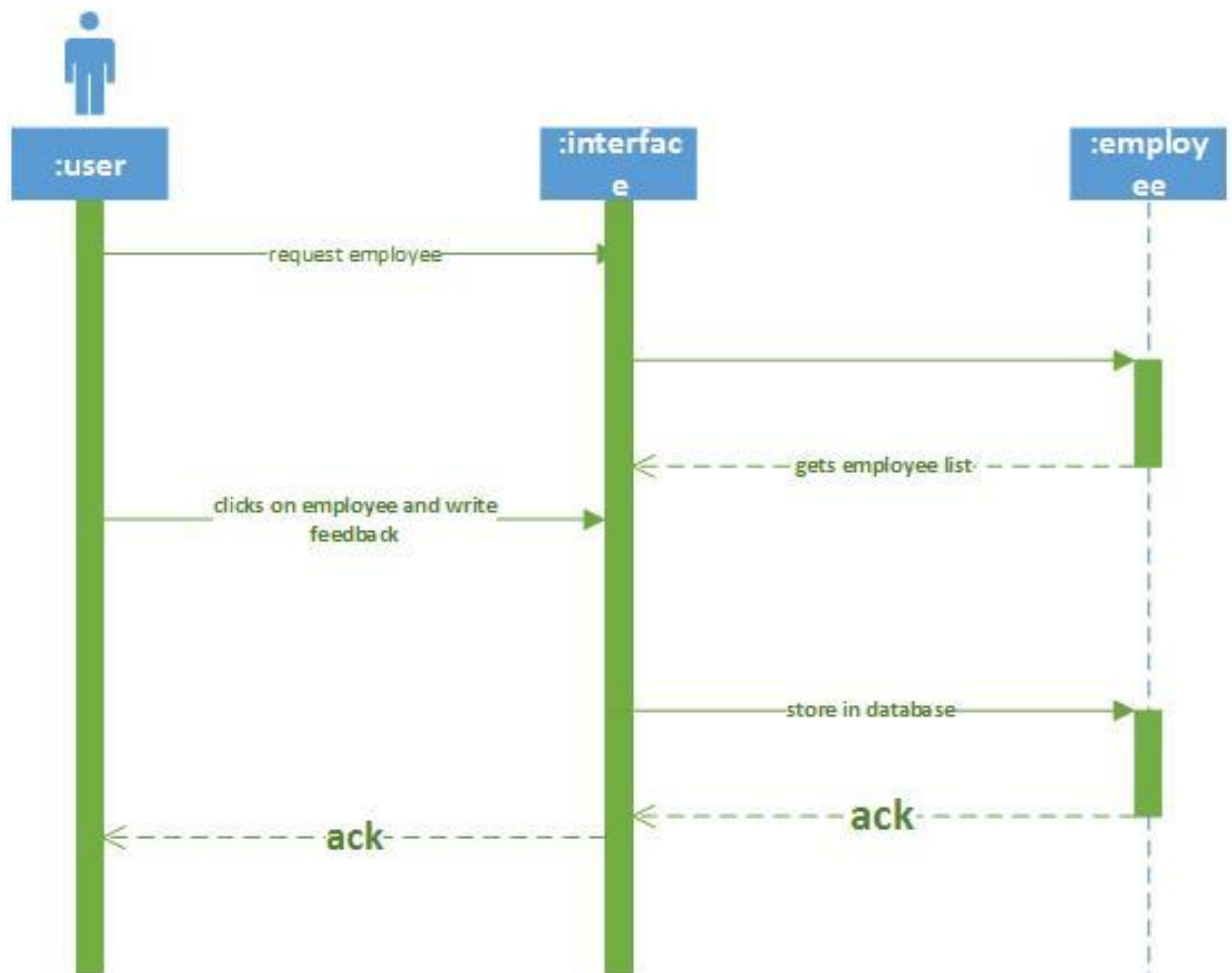


DESIGN OF TEST CASES:

- Wrong credentials for login should be detected
- Test REST api
- Check validation of input fields in registration

2. **FEEDBACK:** By using this service employer can give feedback to employee. Employer can give feedback to employee for those he hired.

SEQUENCE DIAGRAM:

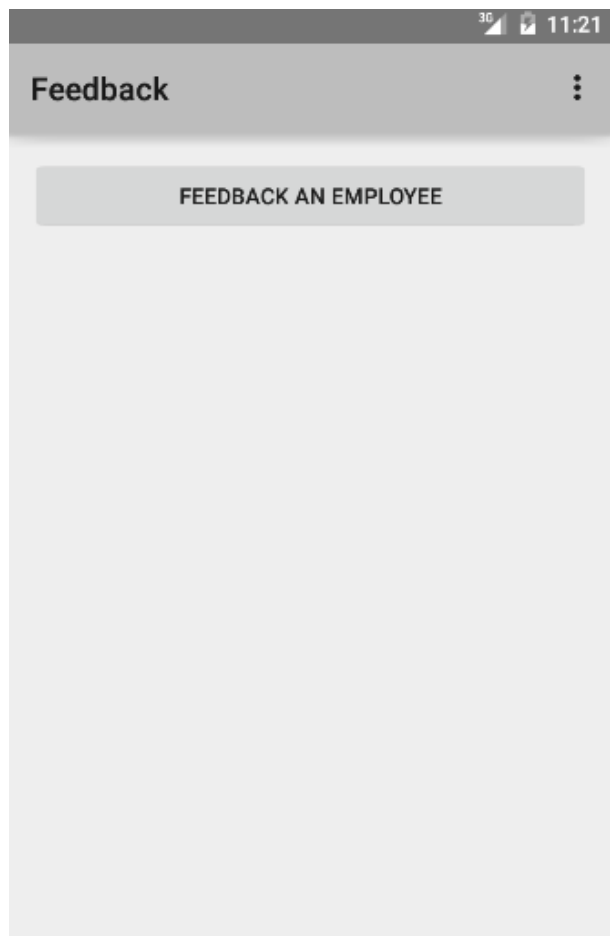


CLASS DIAGRAM:



MOBILE CLIENT INTERFACE:

On pressing the button employer can see list of employee that he/she hired



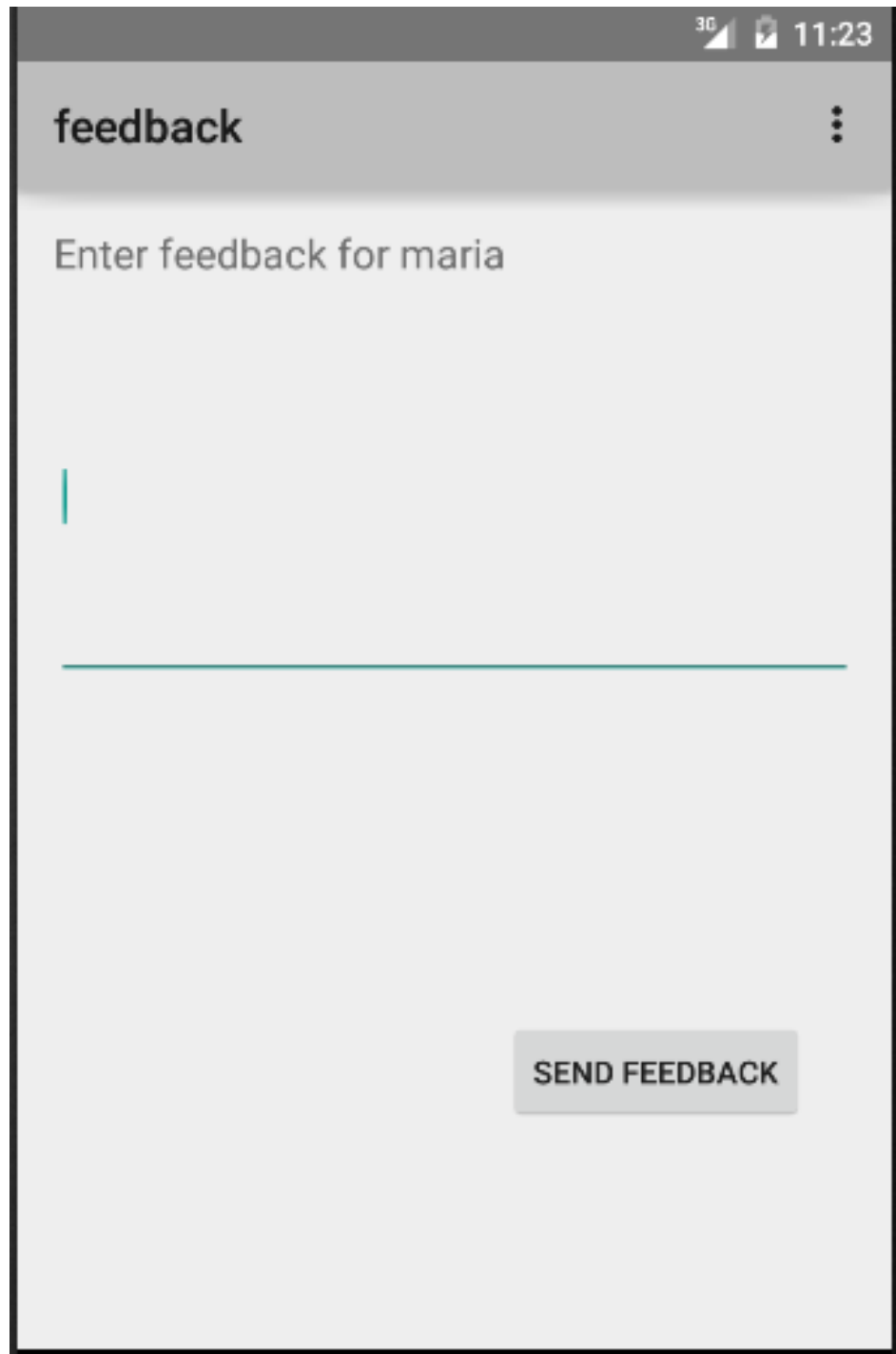
ResultActivity



keith

maria

john



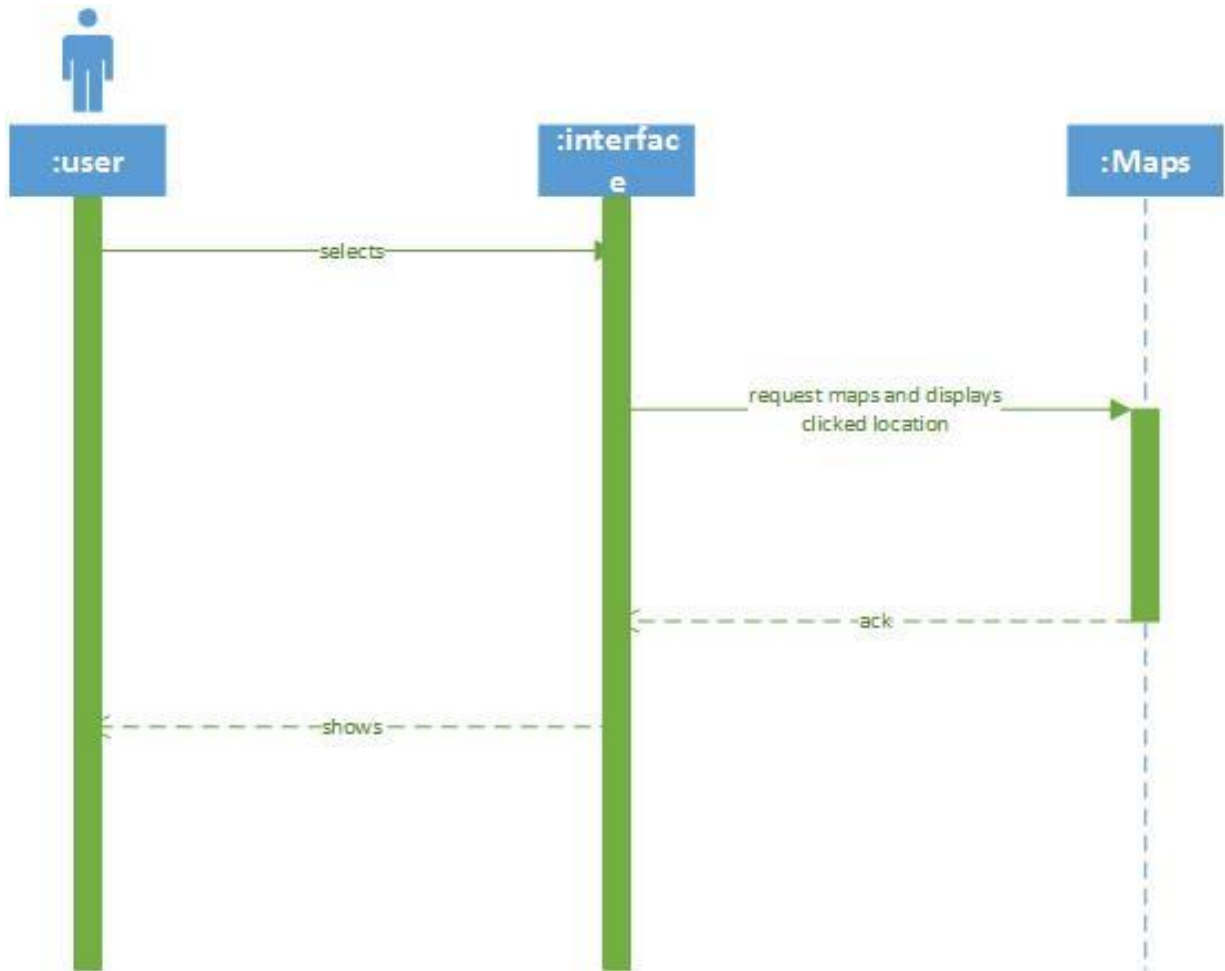
A mobile application interface for providing feedback. At the top, a dark grey header bar contains the word "feedback" in white, a three-dot menu icon, and status icons for 3G, signal strength, and battery level, along with the time 11:23. Below the header, the text "Enter feedback for maria" is displayed in a light grey font. A large, light grey text input area follows, with a thin teal vertical line on the left and a horizontal teal line below. At the bottom right, there is a grey button with the text "SEND FEEDBACK" in black.

DESIGN OF TEST CASES:

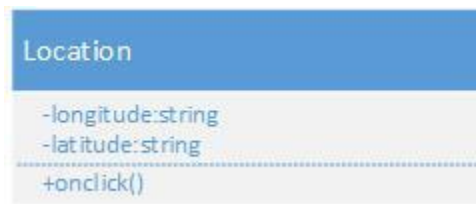
1. Employer can see only the list of employee that he hired
2. Test REST API

3. Location Service: user can select the location of task which he wants to post. This service is integrated to Task Post module

SEQUENCE DIAGRAM:

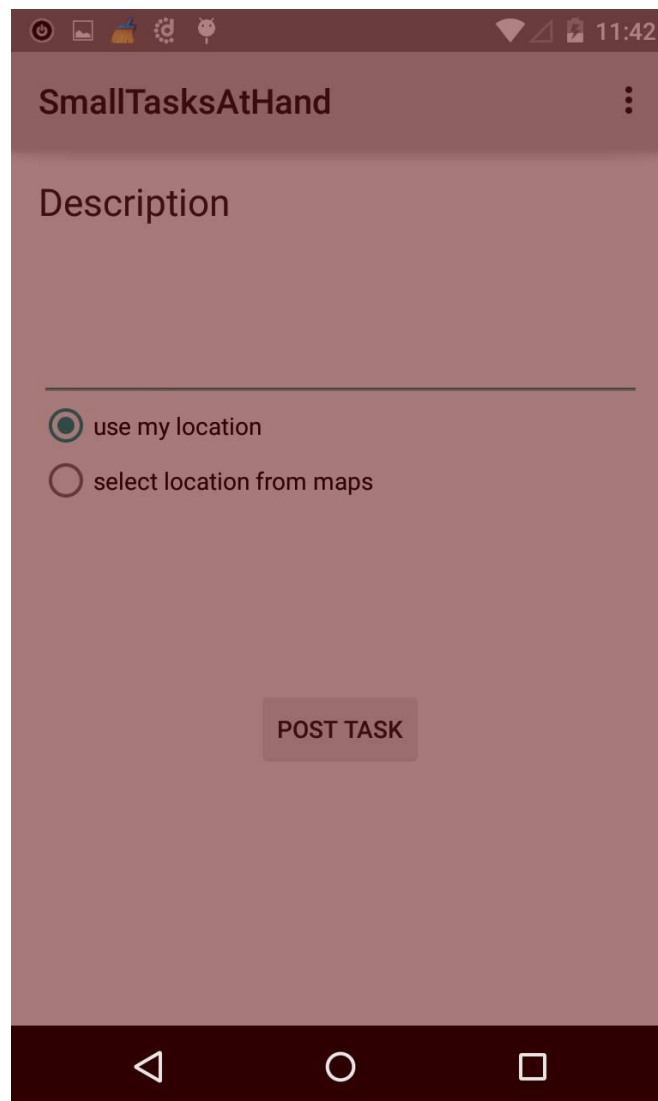


CLASS DIAGRAM:

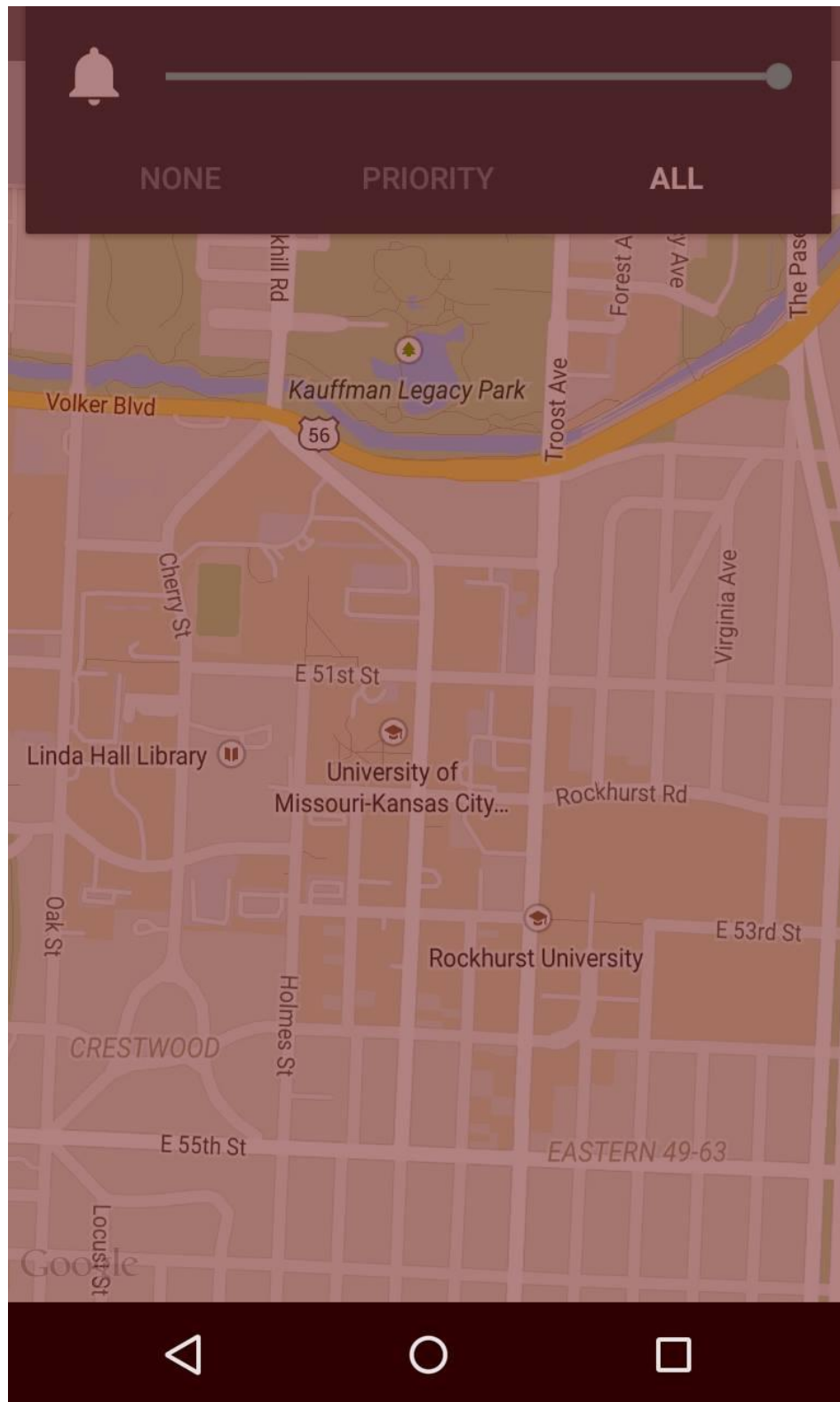


MOBILE CLIENT INTERFACE:

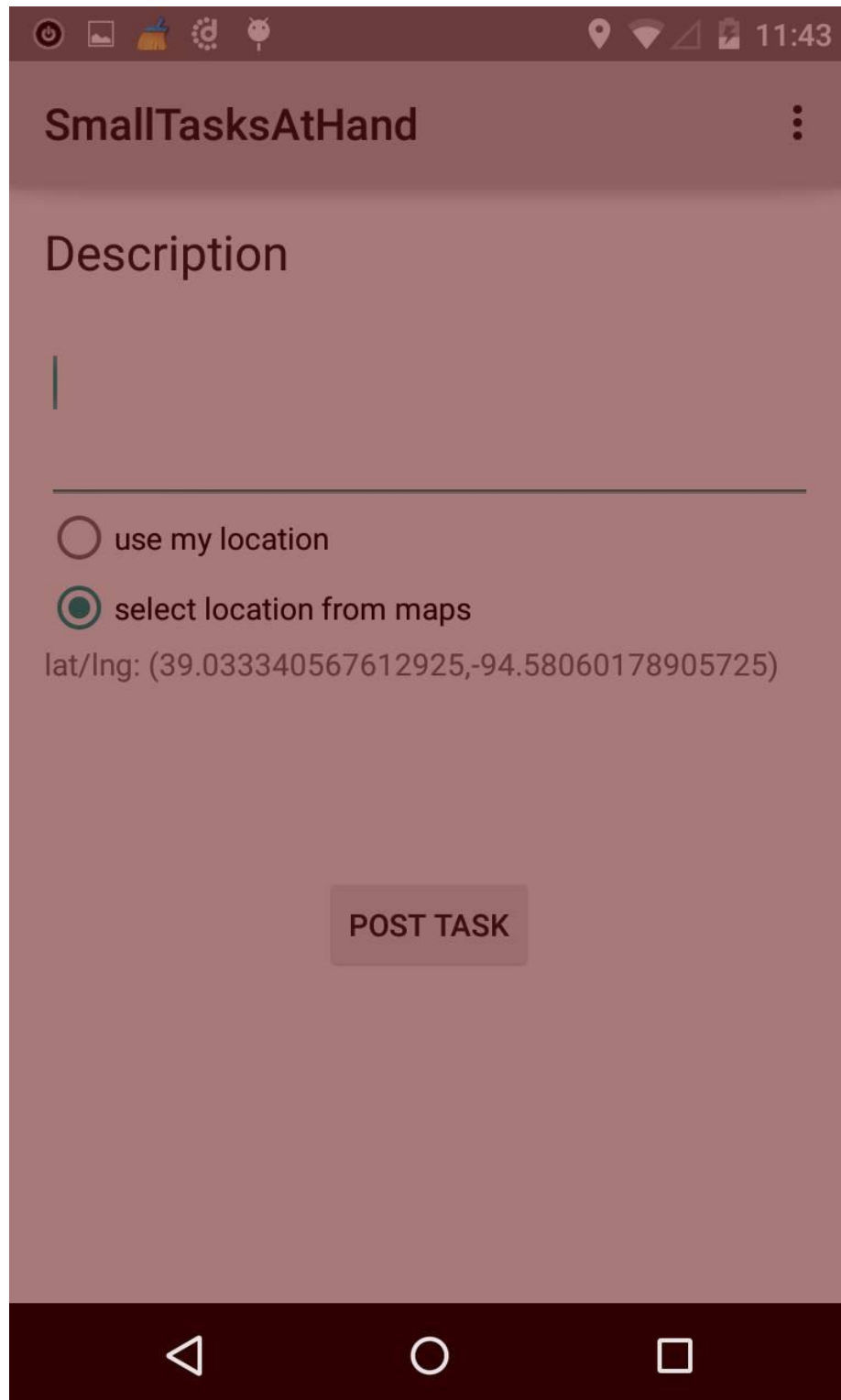
User can select location from maps



After clicking on select from maps user will redirected to activity that contains maps



User can see the location that he selects



DESIGN OF TEST CASES:

1. Maps should return the co-ordinates of the location where user clicks.

IMPLEMENTATION OF USER INTERFACE:

All services will have user interface in android. In this service we developed three services and all are having user interface.

1. Login and Registration:

In this service we have two activities. First if it's a new user then he should register as a new user and then he will receive his unique ID provided by the application. Then secondly he can login into the application by using his received unique provided by the application.

2. FEEDBACK:

In this service we have only one activity that is the employer who posted a task and got it done/completed by an employee of our application then he can give feedback about the employee whether he completed the task in given time and good manner or he did not respond to the calls or messages sent by the employer or did not follow particular instructions given or did not follow the particular order in doing a set of subtasks.

3. SELECTION OF LOCATION IN MAPS:

In this service we give the option of choosing the location of the job by the user. On selecting the option to select the location user will be redirected to maps activity and when user clicks on maps the clicked position will be returned to main activity.

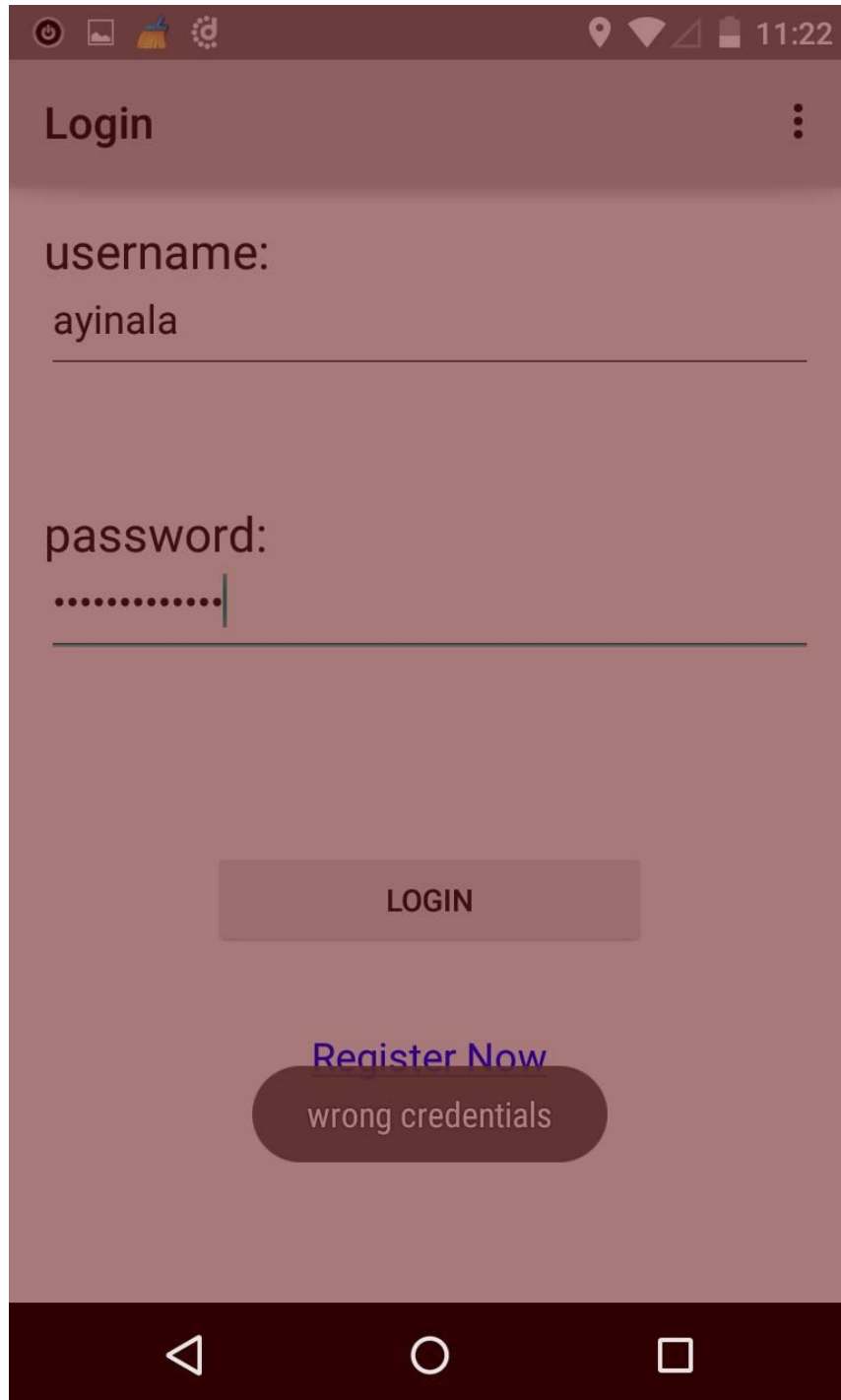
IMPLEMENTATION OF TEST CASES:

- REST API were tested using postman chrome extension and functional testing is done using mongoDB shell. Detailed explanation was given in testing.
- Test cases are tested by using Android Emulator.

TESTING:

FUNCTIONALITY TESTING:

1. Validating the Login – If user enters wrong credentials, then he should be notified.



The screenshot displays a mobile application interface for a login screen. At the top, there is a status bar with various icons and the time 11:22. Below this, the app title 'Login' is visible in the top left corner, and a three-dot menu icon is in the top right corner. The main form area contains two input fields: 'username:' with the text 'ayinala' entered, and 'password:' with masked characters '.....'. Below the password field is a 'LOGIN' button. At the bottom of the form, there is a link 'Register Now' and a red rounded rectangle containing the text 'wrong credentials', indicating an error message.

2. All fields in registration should be validated correctly, if any field is filled in wrong by the user then user should be notified and that particular registration must be cancelled.

Register

username:
ayinala

username already taken





password:
.....





confirm password:
.....


Email
ayinalakaus@gmail.com

phone number:
6519648022


☒ I am an employer



 11:24



Register



password:

.....

confirm password:

.....

Email

kaushik.ayinala92

invalid email address

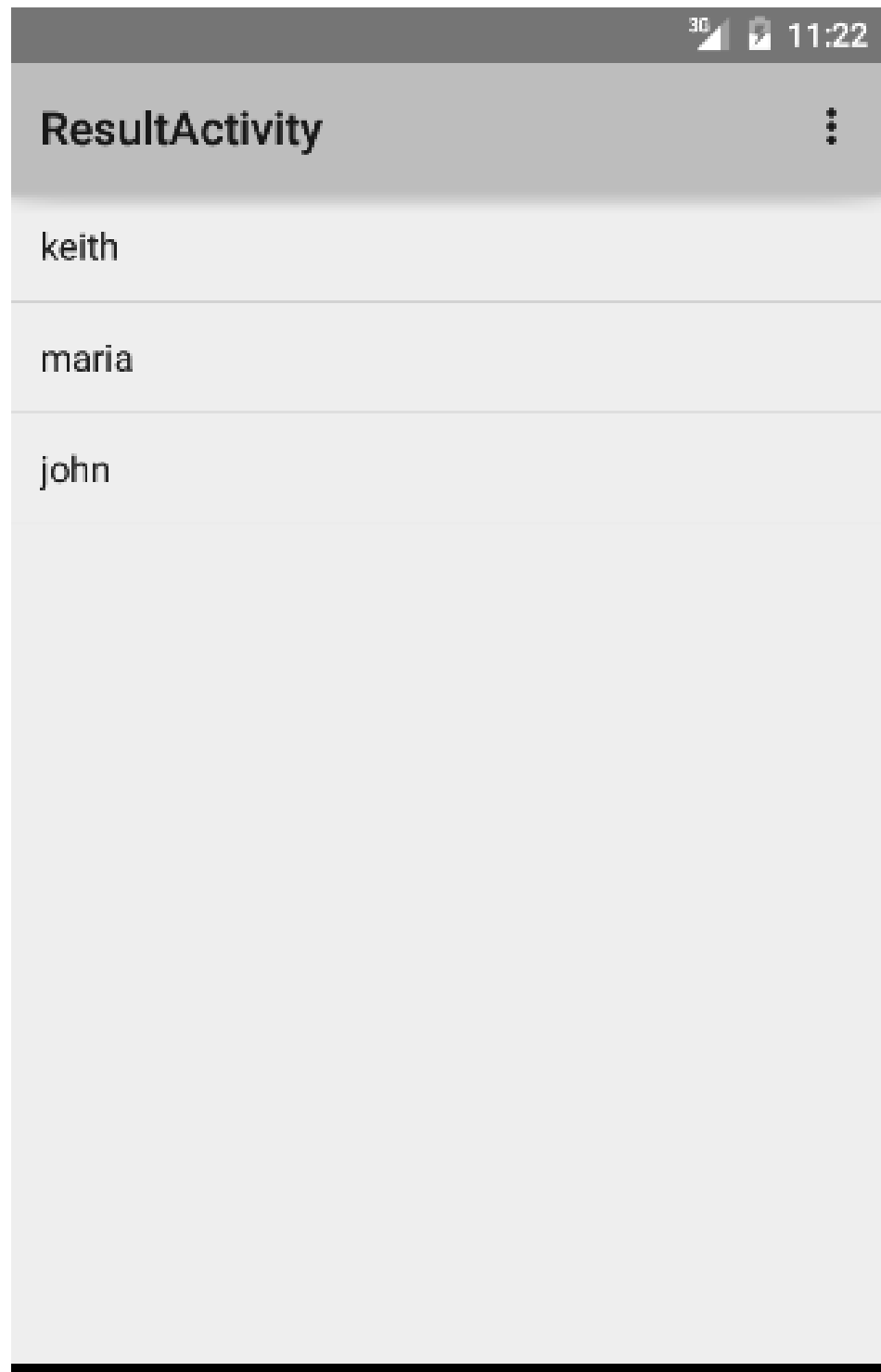
phone number:

6519658025


☒ I am an employer

REGISTER

Employer should be able to give feedback only about the employee that he hired to do his tasks.





All Documents



Display mode: ☒ list ☐ table ([edit table view](#)) 

records / page [1 - 3 of 3]



```
{
  "_id": {
    "$oid": "552b2489e4b0d849d90d7c83"
  },
  "id": "john",
  "description": "Buy stationary"
}
```

```
{
  "_id": {
    "$oid": "552b24d2e4b0d849d90d7c86"
  },
  "id": "maria",
  "description": "drop me at airport"
}
```

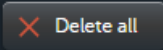
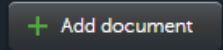
```
{
  "_id": {
    "$oid": "552b2526e4b0d849d90d7c88"
  },
  "id": "keith",
  "description": "buy goods for thanks giving"
}
```


 

records / page [1 - 3 of 3]


3. Check whether feedback is stored in the database or not.

Documents





All Documents



Display mode: ☒ list ☐ table ([edit table view](#)) 

records / page [1 - 2 of 2]

```
{
  "_id": {
    "$oid": "550b727be4b00d60c5cb1e58"
  },
  "id": "12345",
  "tid": 1426813373338,
  "a_id": 1426813444453
}
```

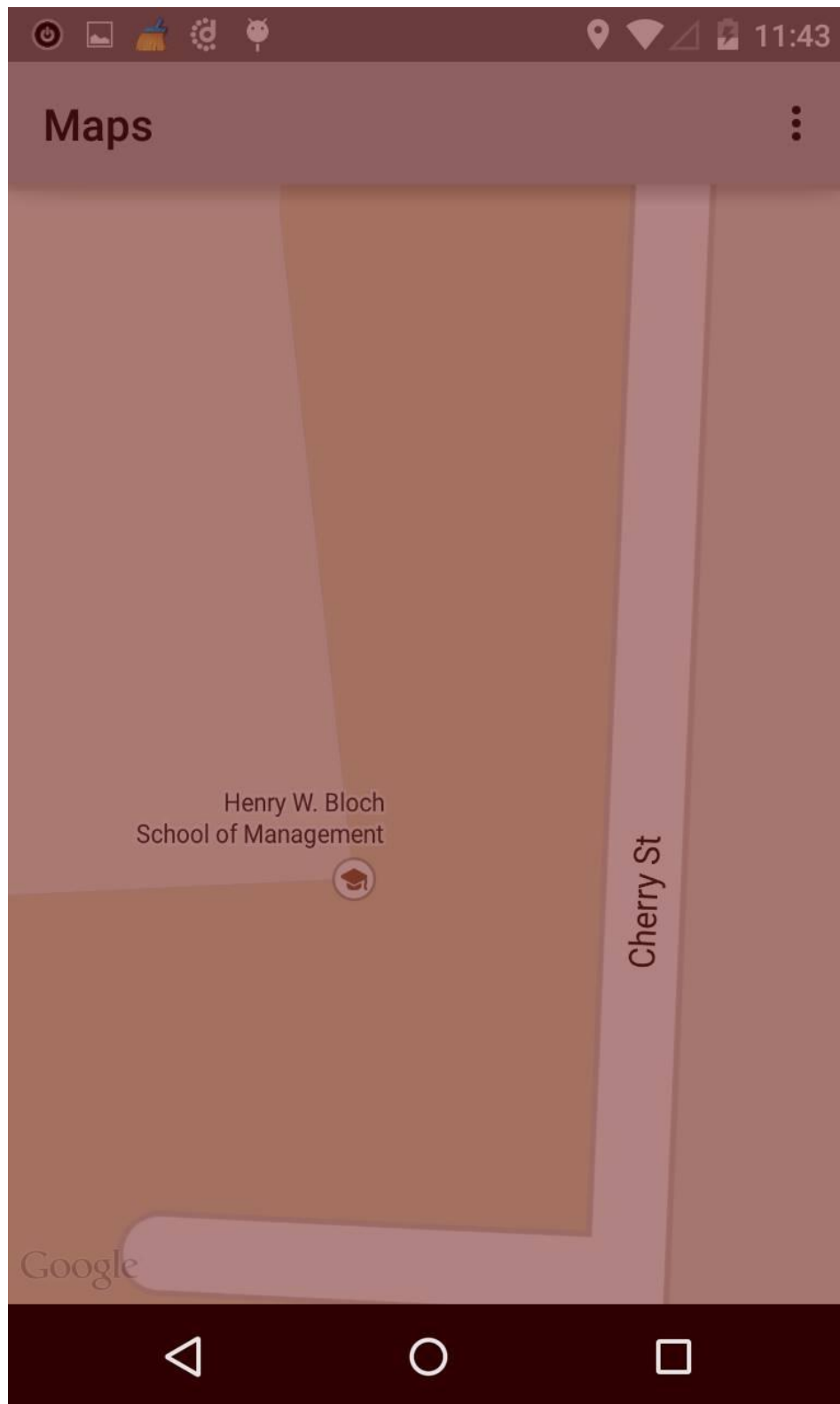
 

```
{
  "_id": {
    "$oid": "552b24d2e4b0d849d90d7c86"
  },
  "id": "maria",
  "description": "drop me at airport",
  "feedback": "She is late"
}
```

records / page [1 - 2 of 2]

4. Maps should return the address of location where user click's.





11:43

SmallTasksAtHand



Description

|

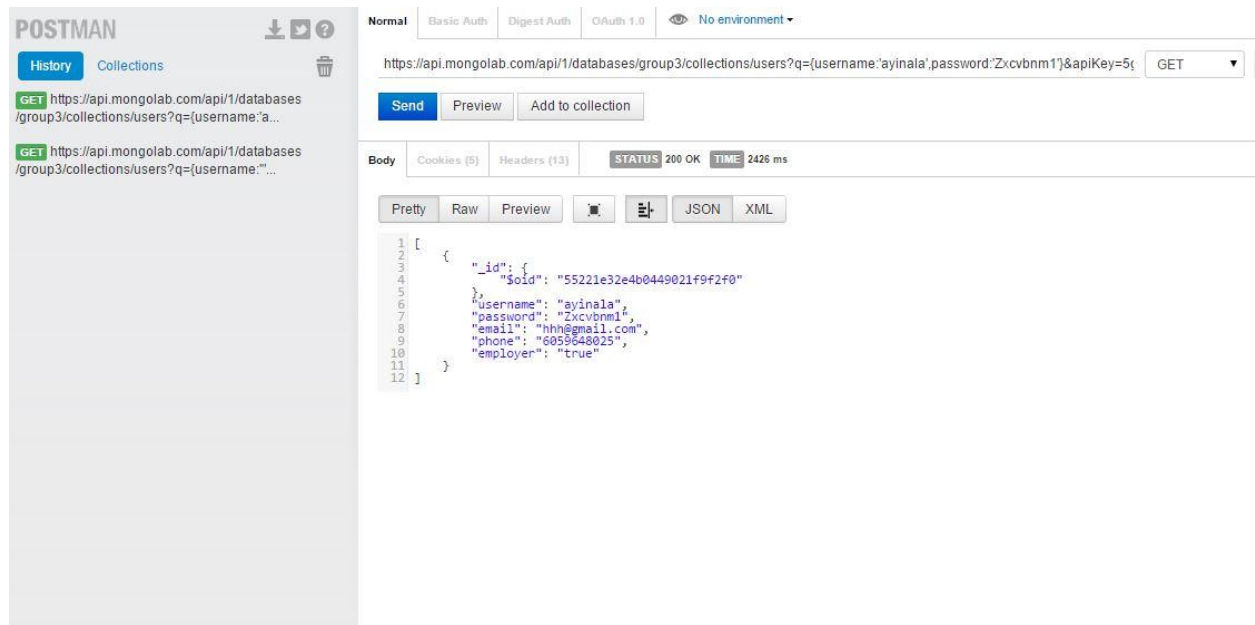
-
- ☐ use my location
- ☒ select location from maps

lat/lng: (39.033340567612925,-94.58060178905725)

POST TASK



REST API TESTING



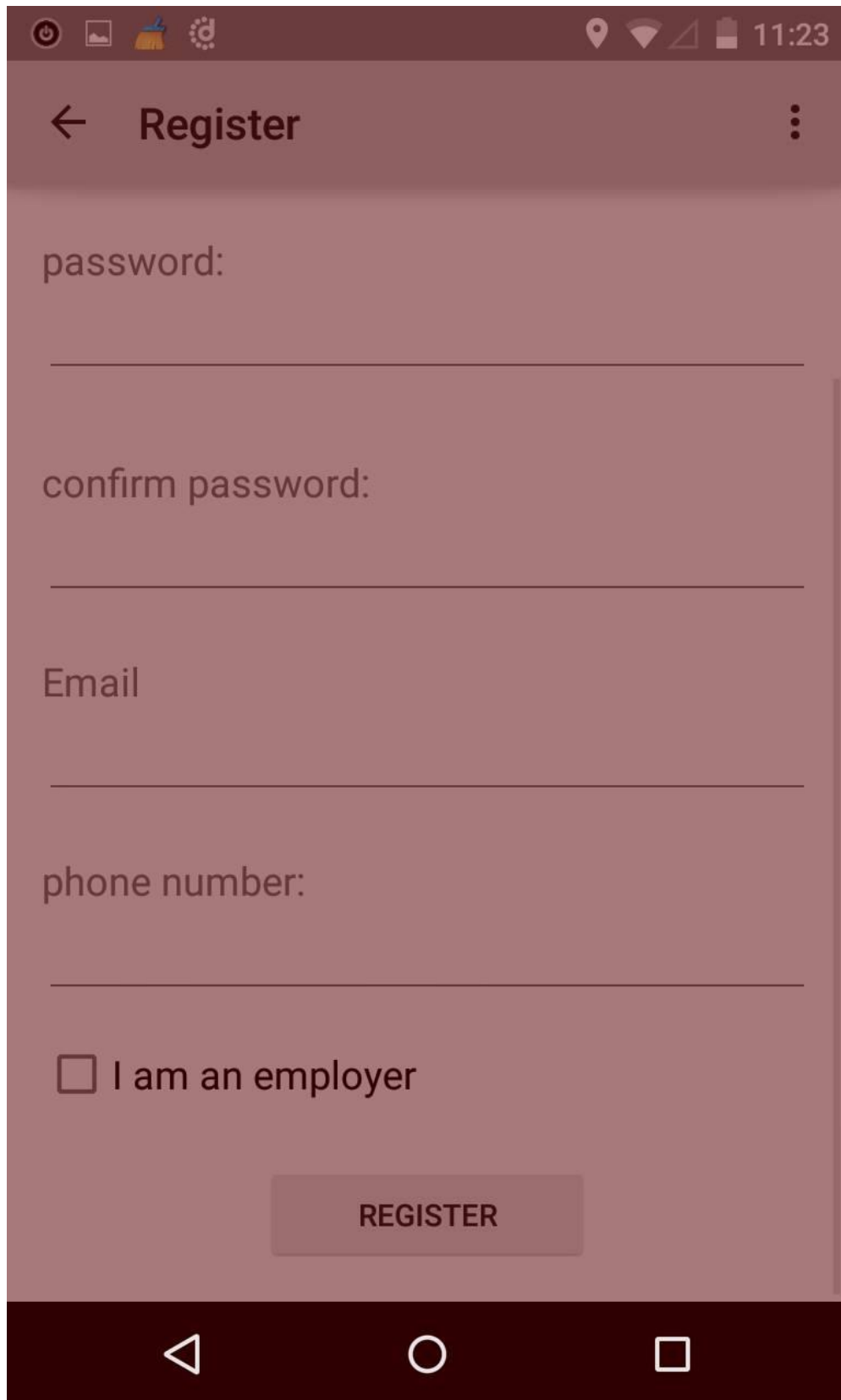
Deployment:

Scrumdo link: <https://www.scrumdo.com/projects/project/smalltasksathand1/iteration/126289#>

GitHub link: <https://github.com/ayinalakaushik/SmallTasksAtHand>

Report:

- At first we have implemented the login service.
- It took some time to implement the scrolling in the registration page.

A mobile application registration screen with a dark blue header and a light blue body. The header contains a back arrow, the title "Register", and a menu icon. The body contains four input fields with labels "password:", "confirm password:", "Email", and "phone number:". Below these is a checkbox labeled "I am an employer". At the bottom is a "REGISTER" button. The screen is framed by a dark blue status bar at the top and a dark blue navigation bar at the bottom.

password:

confirm password:

Email

phone number:

☐ I am an employer

REGISTER

- We have written many cases for the validation of registration.

Login

username:
ayinala

password:
.....

LOGIN

[Register Now](#)

Registration was successful

- We have implemented the feedback services in which employer can give feedback to employee to those who he hired.

feedback

⋮

Enter feedback for maria

She is late

SEND FEEDBACK

ResultActivity

⋮

keith

maria

john

feedback was sent

- Next have implemented the map services which is the difficult service in this situation.
- We have obtained API key from google Maps API

New Project

Project name ?

small-tasks-at-hand

Project ID ?

dynamic-vehicle-91404

Show advanced options...

Create Cancel

Google Developers Console

< Projects

small-tasks-at-hand

Overview

Permissions

Billing & settings

APIs & auth

APIs

Credentials

Consent screen

Push

Monitoring

Source Code

Compute

Networking

Storage

Big Data

Support

Need help?

<<

OAuth

No client IDs found.

OAuth 2.0 allows users to share specific data with you (for example, contact lists) while keeping their usernames, passwords, and other information private.

Learn more

Create new Client ID

Public API access

No keys found.

Use of this key does not require any user action or consent, does not grant access to any account information, and is not used for authorization.

Learn more

Create new Key

ers to share specific data

2,

m

Create an Android key and configure allowed Android applications

This key can be deployed in your Android application.

API requests are sent directly to Google from your client Android device. Google verifies that each request originates from an Android application that matches one of the certificate SHA1 fingerprints and package names listed below. You can discover the SHA1 fingerprint of your developer certificate using the following command:

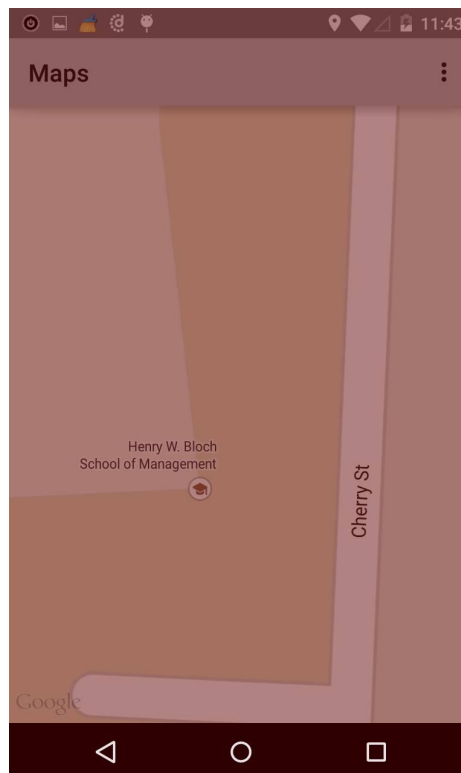
```
keytool -list -v -keystore mystore.keystore
```

[Learn more](#)

Accept requests from an Android application with one of the certificate fingerprints and package names listed below
One SHA1 certificate fingerprint and package name (separated by a semicolon) per line. Example:
45:B5:E4:6F:36:AD:0A:98:94:B4:02:66:2B:12:17:F2:56:26:A0:E0;com.example

Create Cancel

- Using fragments we displayed the map and when the user touches the map we finished the map activity and return the position to main activity.



Works Completed:

We completed three services in this increment.

1. Login and registration for the users
2. Employer gives feedback to employee. (Feedback Service)
3. User can select the location of the task (Map Service)

Team Contributions:

Kaushik - 30%

Yaswanth - 23%

Tharkin - 24%

Ravi Teja - 23%

First Service (Login and reg)	Second Service (Feedback)	Third Service (Maps)
Kaushik – Login and registration UI (Time Taken-10 hours)		Kaushik – Obtaining and integrating API key, play services, maps UI
Yaswanth - Connection to database(Improved) (Time Taken-9 hours)	Yaswanth - UI (Time Take- 2 hours)	Yaswanth - written logic code to return data from maps (Time Taken- 2 hours)
Tharkin - Validation of login and registration (Time Taken- 7 hours)	Tharkin - Database Schema (Time Taken- 2 hours)	Tharkin - connection to maps from MainActivity using fragments (Time Taken- 2 hours)
Ravi Teja - Testing of fields (Time Taken- 4 hours)	Ravi Teja - Logic and testing (Time Taken- 8 hours)	Ravi Teja - Testing (Time Taken- 2 hours)

Work to be completed: We have planned to integrate all the services that we have implemented till now but we are unable to integrate right now due to the construction of classes.