



CS 442: Mobile Applications Development

Assignment 4 – Inspiration Rewards (400 pts)

Uses: Multi-Activity, Location Services, Geocoding, Internet, APIs, Images, Camera, Gallery

A) App Highlights:

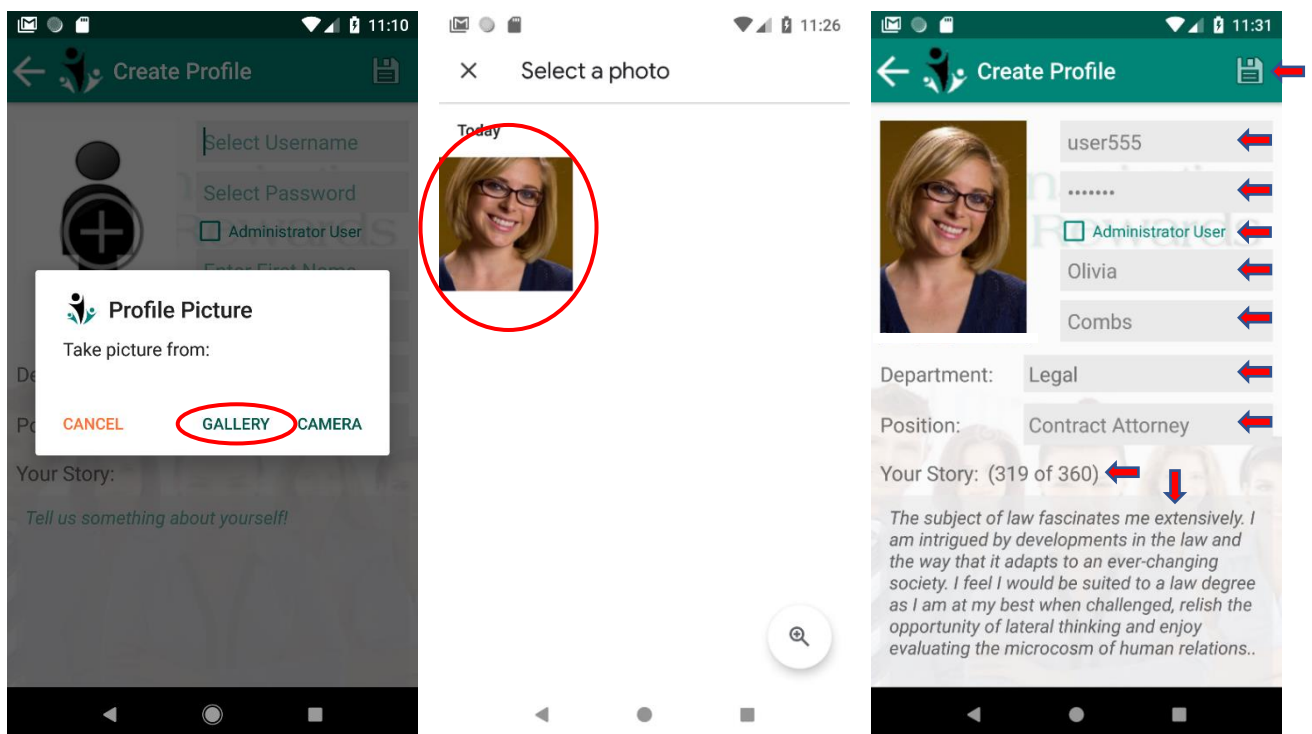
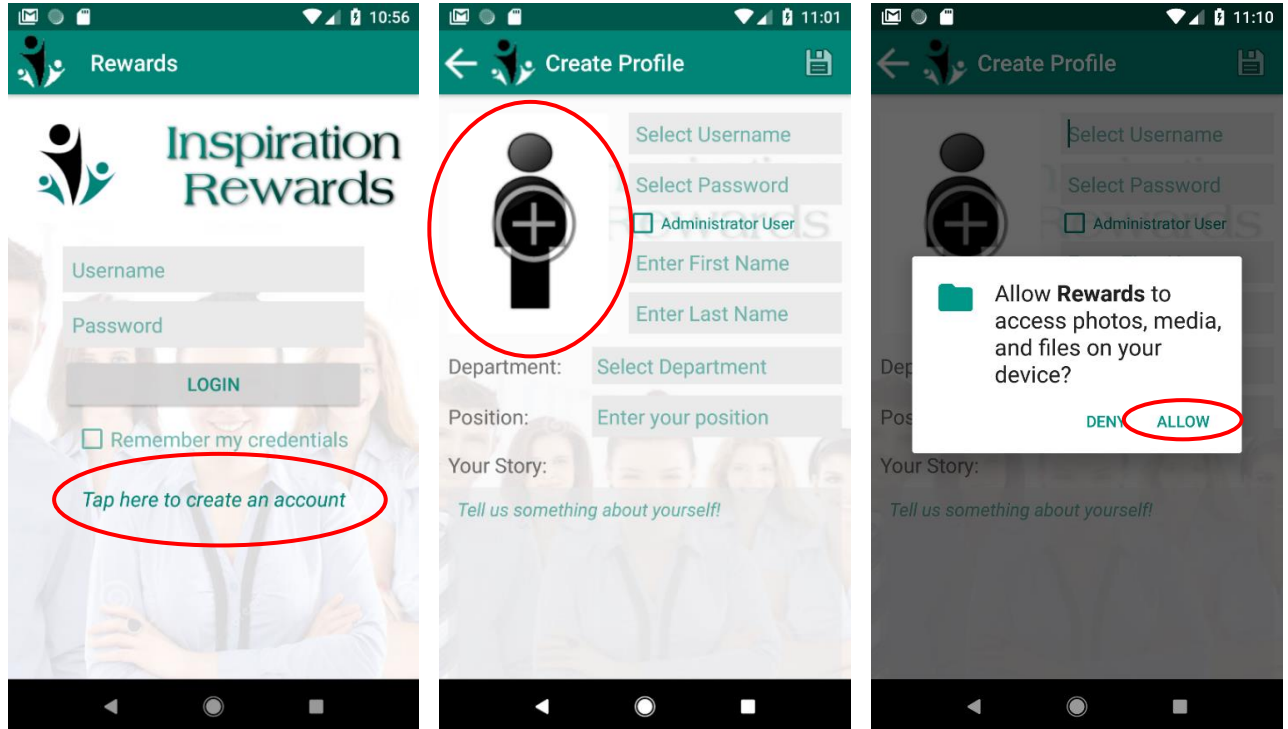
- This app allows users within a company or organization to award each other “reward points” (and comments) as a thank you or a commendation for their performance on a task.
- Users can create a profile for themselves and interact with the other users. Users can later edit/update their profile data
- Users start with a fixed amount of points that they can award to other users.
- When a user awards points to another, the point value is deducted from the point total available to give to others. Once a user is out of points to give, they can no longer add new rewards until their points are renewed.
- Assume the process of resetting the points-to-award value (and the actual act of rewarding the employee) are out of scope for this project.
- You will need to create classes to represent the user profiles and the reward assignments.
- We will be using an Amazon Web Services (AWS) based API to as the “back end” which holds user and reward data.

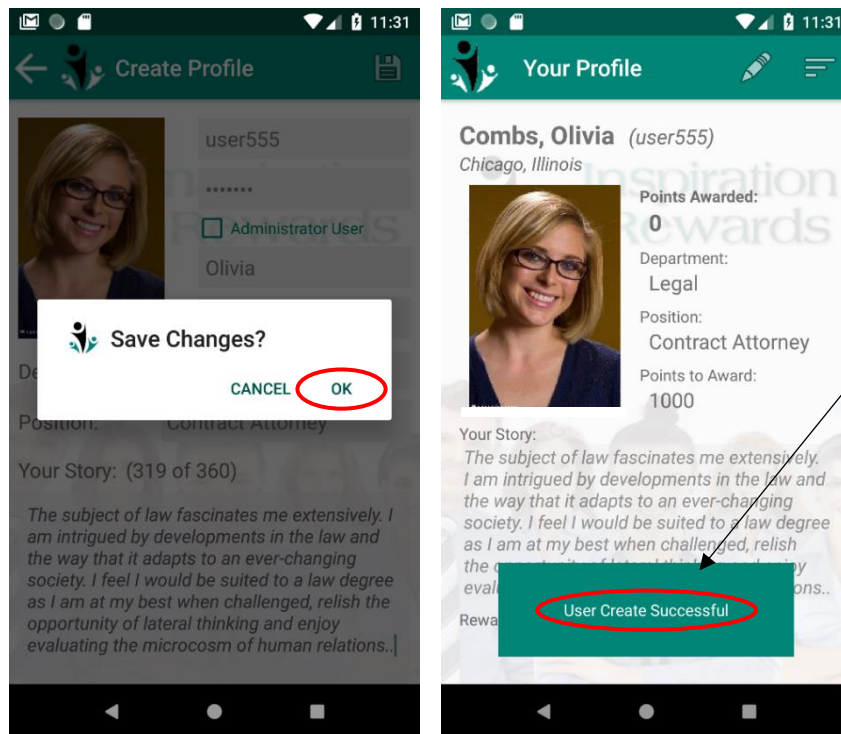
B) Behavior Use Cases and Activities: *Activity/behavior diagrams follow (in section C)*

- Users can create a new profile (via the Create Profile Activity) with data including a username, password, first & last name, administrator flag, department, position, a short self-summary, and a photo. Successful account creation logs in the user and displays the Profile Activity. The Create Profile Activity is displayed by clicking a link on the Login Activity. See diagram “a” in the next section.
- Users login (via the Login Activity) by providing their username and password. A checkbox can be checked to save the credentials so that they are auto-populated whenever the app is started. See diagram “b” in the next section. Upon login, the User Profile Activity is displayed.
- The data displayed in the Profile Activity includes first & last name, username, location, awarded point total, department, position, points available to award, a short self-summary, a photo, and a list of awarded points. See diagram “b” in the next section.
- From the User Profile Activity, the user can open the Edit Profile Activity via Options Menu. The Edit Profile Activity allows the user to edit the password, administrator flag, first & last name, department, position, the short self-summary, and the photo. See diagram “c” in the next section.
- From the User Profile Activity, the user can open the Leaderboard Activity (showing all other users, sorted by awarded point total) via Options Menu. From the Leaderboard Activity, an individual user can be selected and then displayed in the Award Activity. From there, rewards points (and a comment) can be awarded to the selected user. See diagram “d” in the next section.

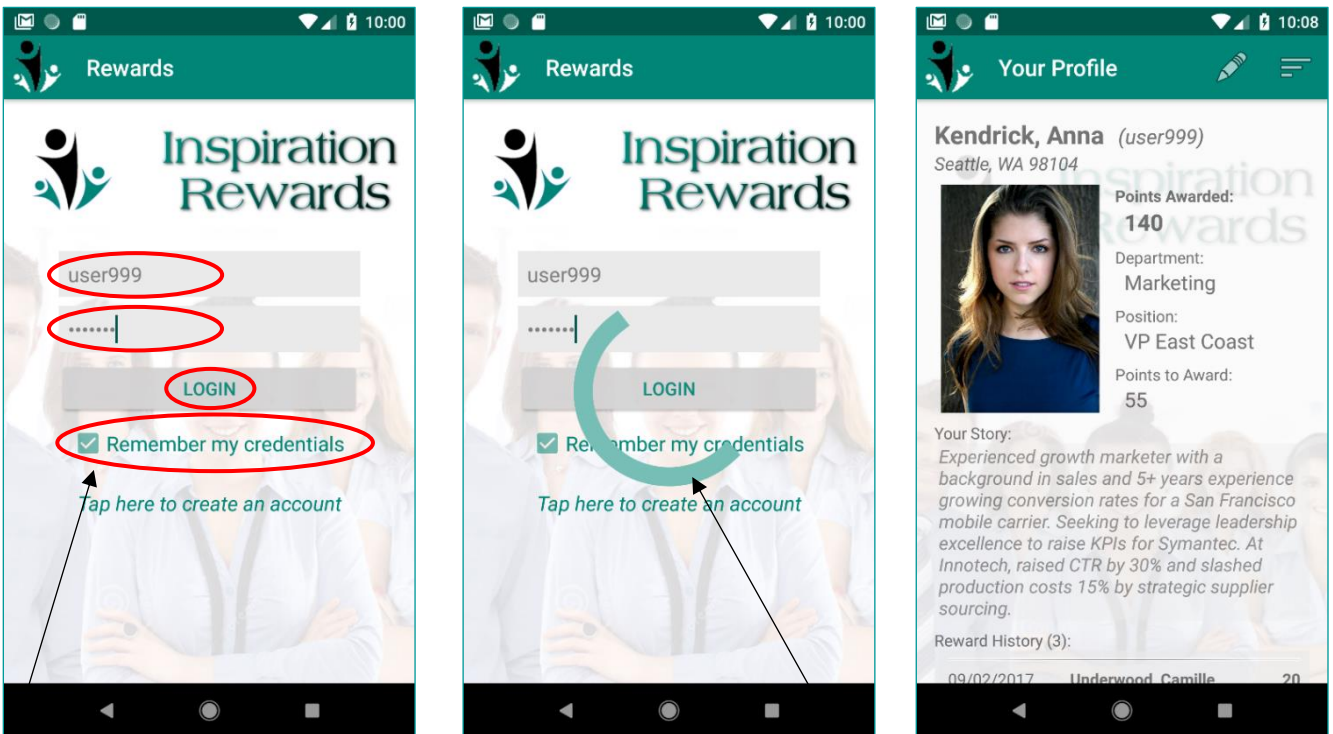
C) Application Behavior Diagrams:

a. Create Profile (Note, the first time the app runs, it will ask for file and location permissions):



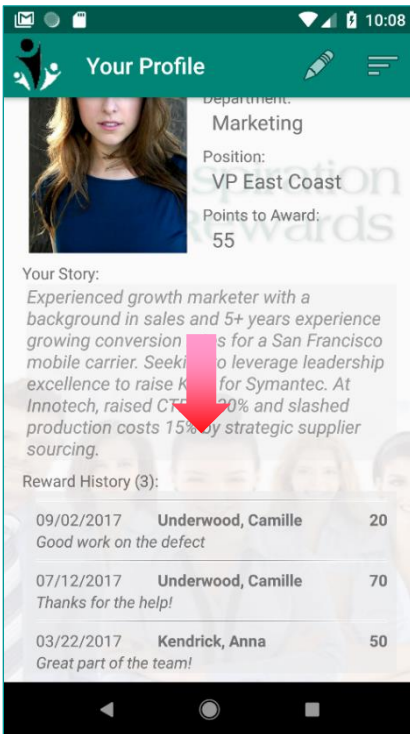


b. **User Login** (Note, the first time the app runs, it will ask for file & location permissions):



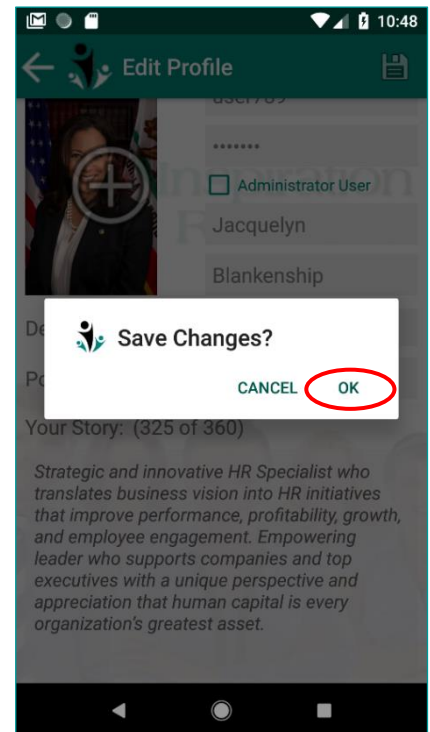
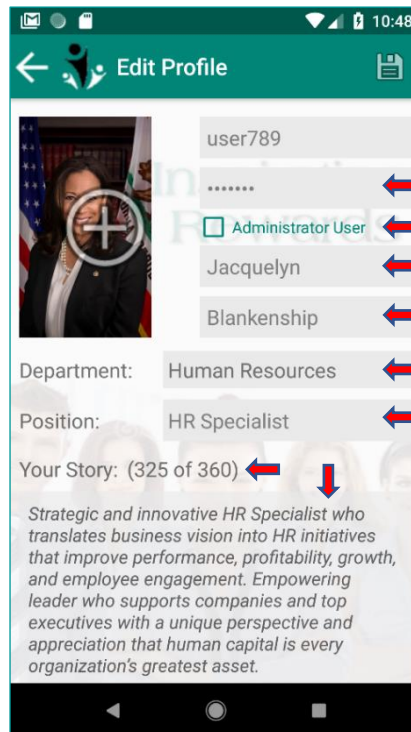
Use SharedPreferences to save this choice, and the user name and password values when checked.

This is an indeterminate ProgressBar



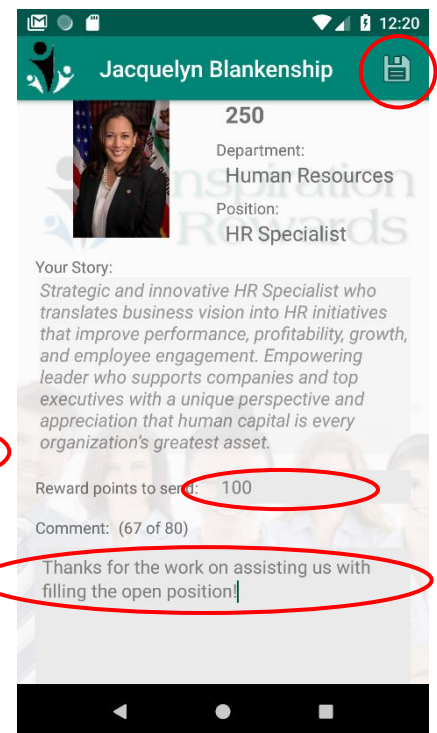
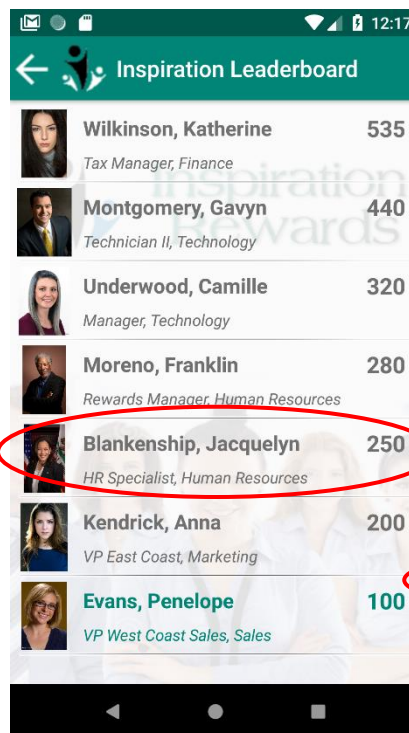
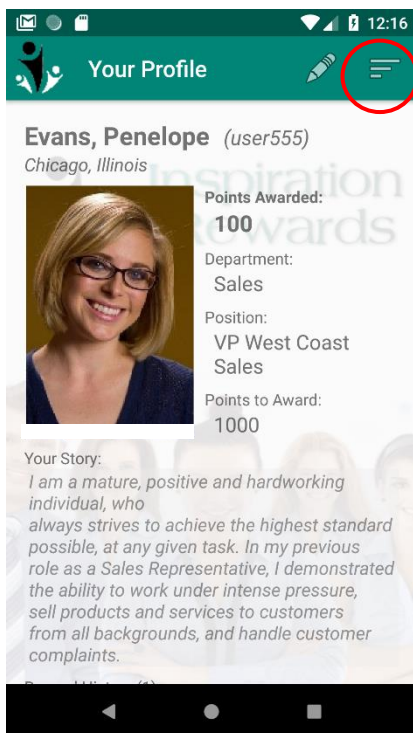
This whole Activity is scrollable

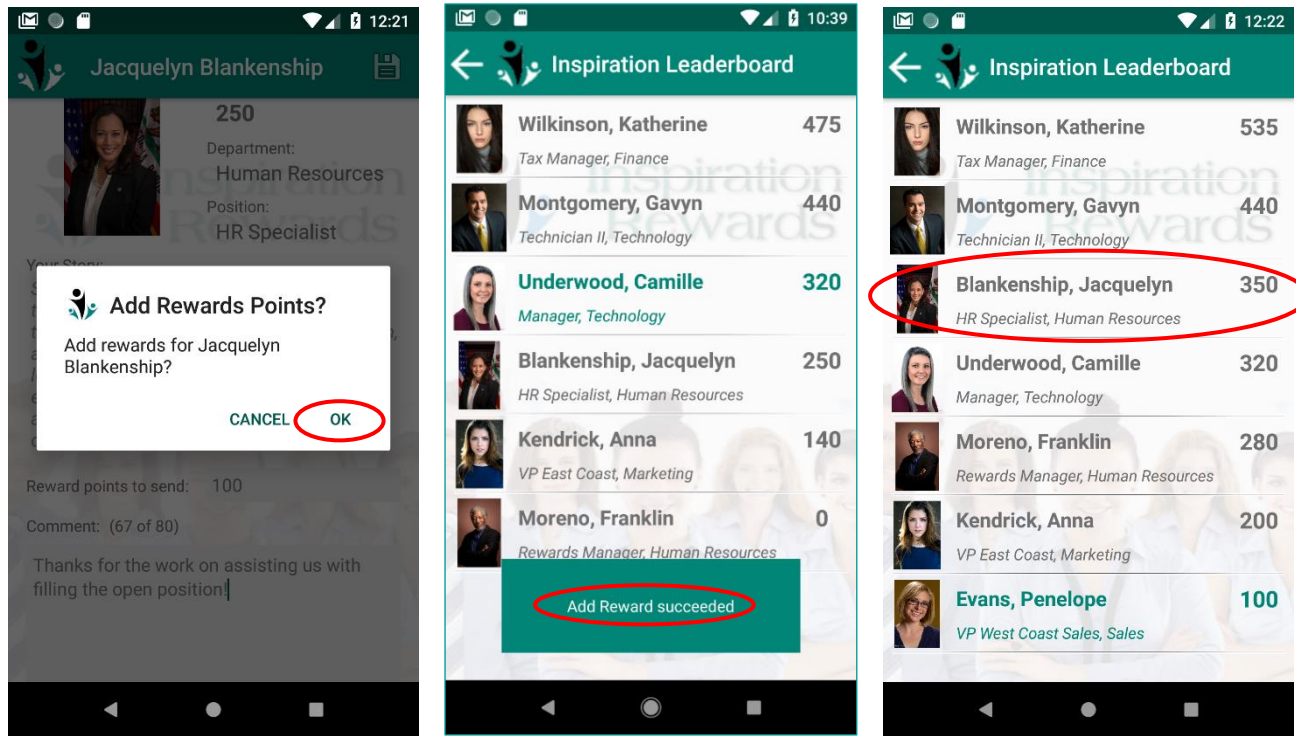
c. Edit Profile





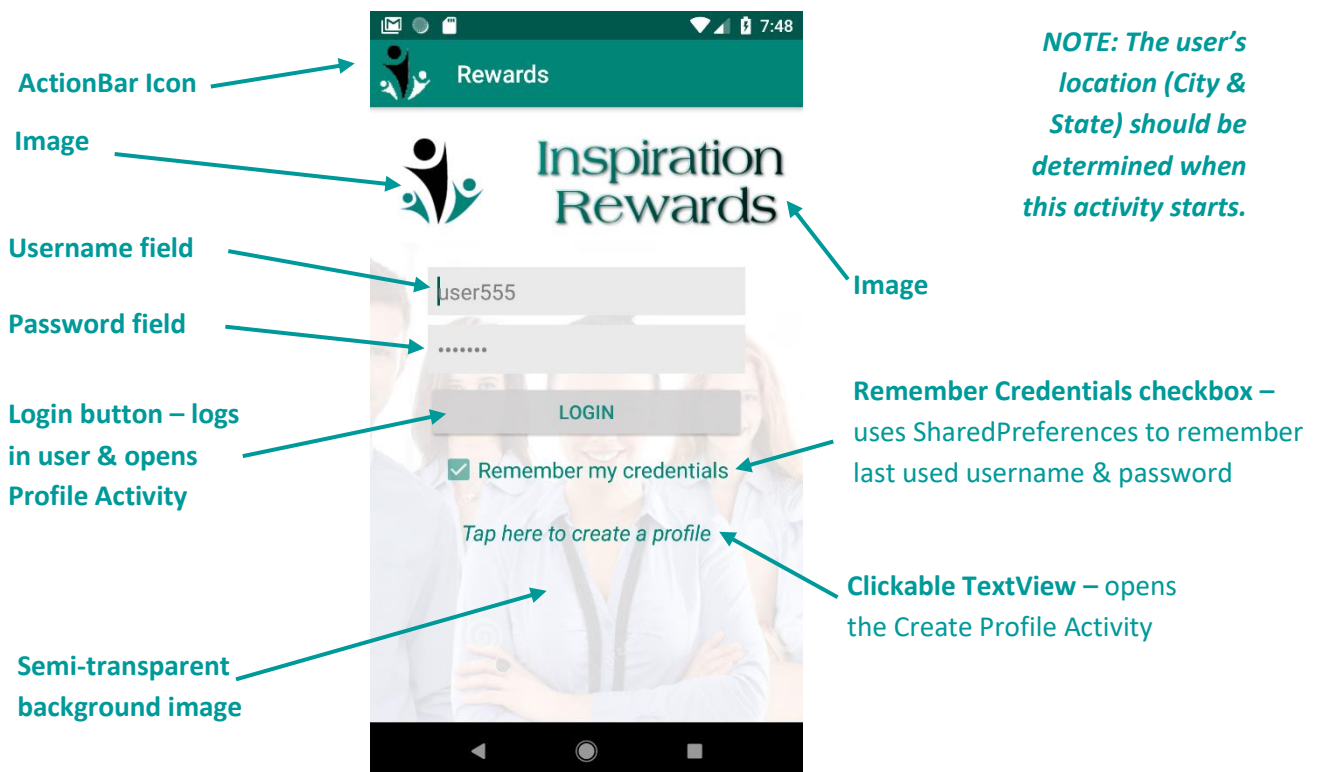
d. Add Rewards



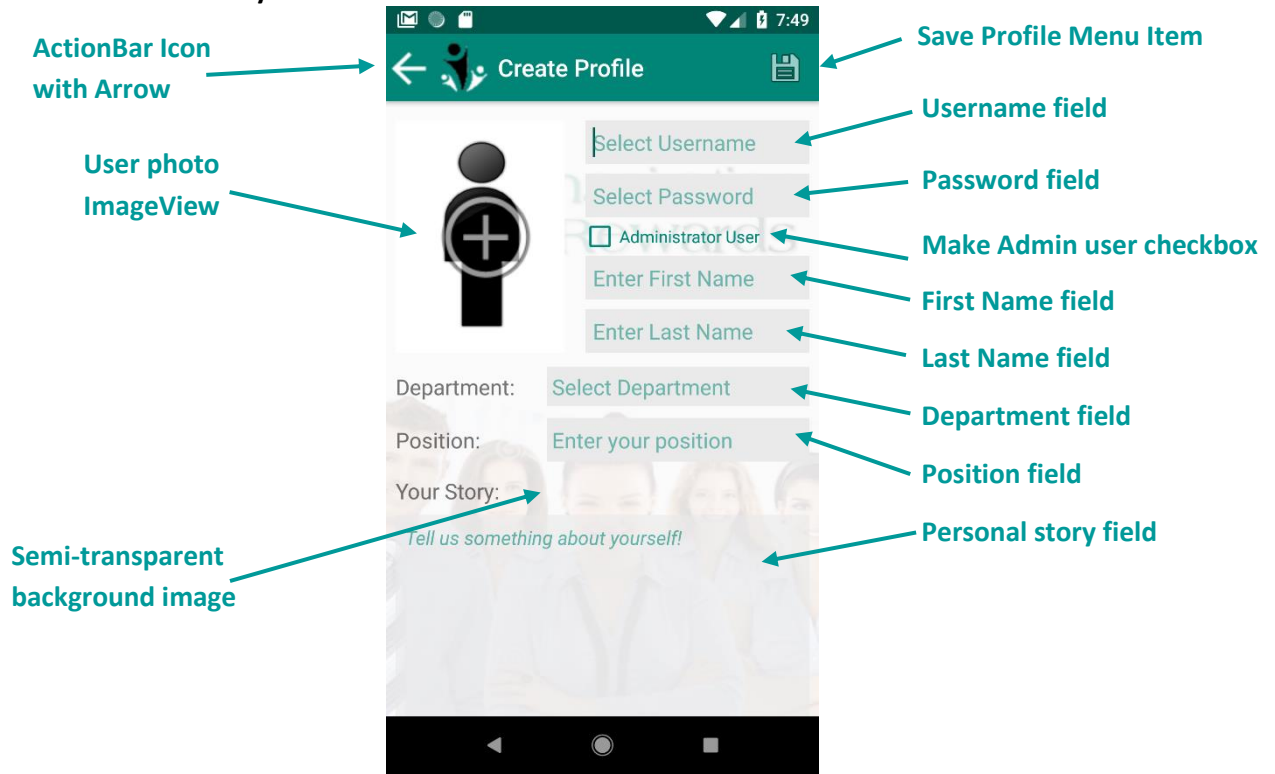


D) Activity Details

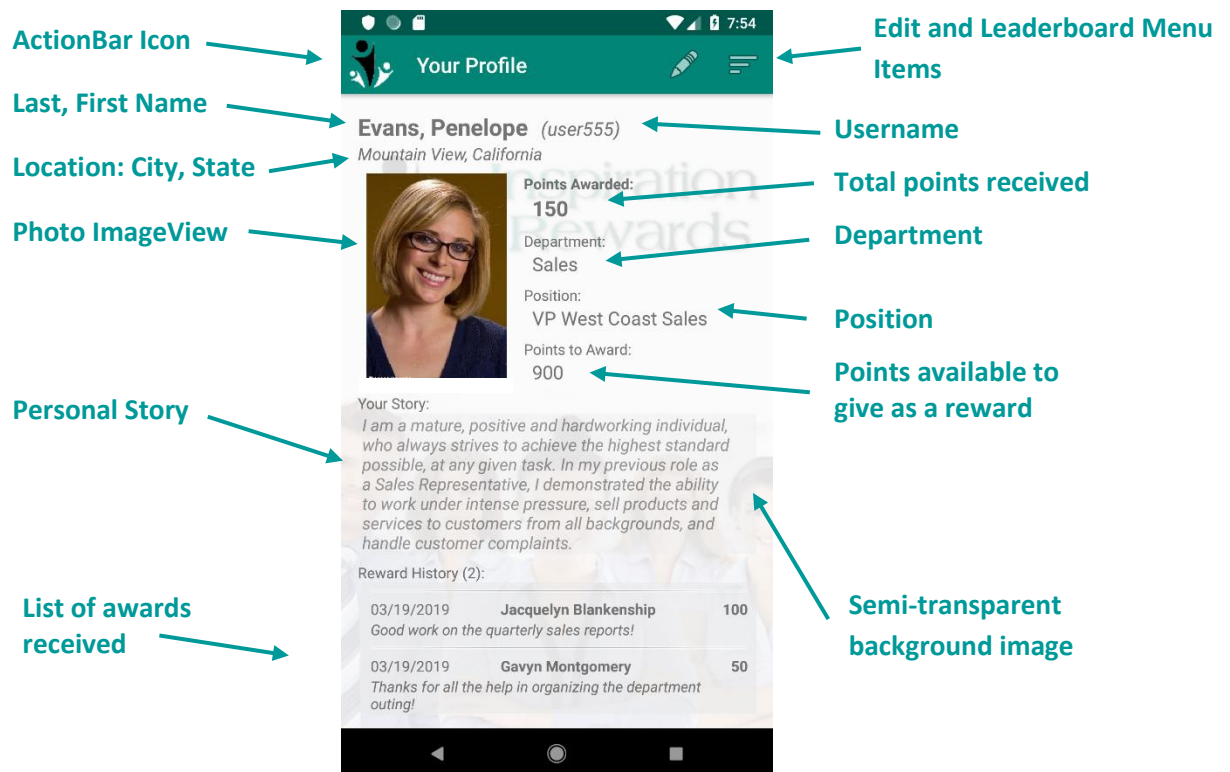
a. Login Activity



b. Create Profile Activity



c. Profile Activity



d. Edit Profile Activity

ActionBar Icon
with Arrow

User photo
ImageView

Semi-transparent
background image

Save Profile Menu Item

Username field
(not editable)

Password field

Make Admin user
checkbox

First Name field

Last Name field

Department field

Position field

Available character count

Personal story field

e. Leaderboard Activity

ActionBar Icon
with Arrow

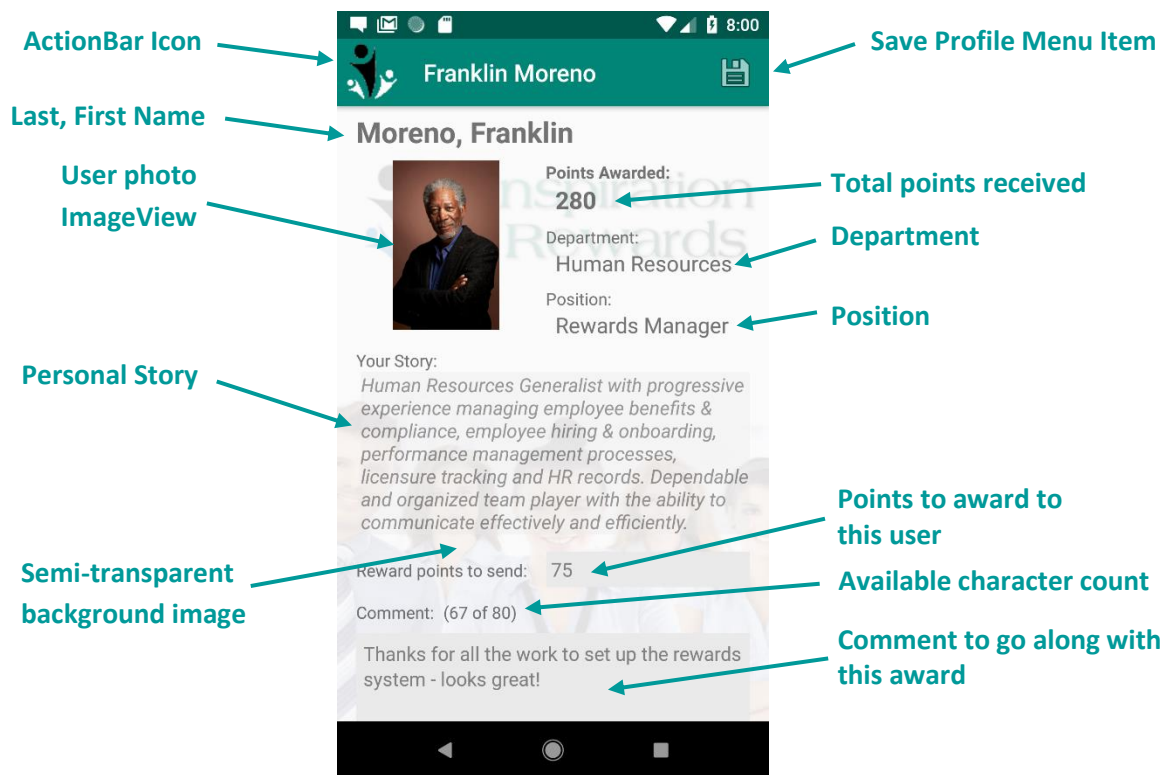
List of users, sorted
by total awards
points received

User Photo	Name	Points
	Wilkinson, Katherine	535
	Montgomery, Gavyn	440
	Blankenship, Jacquelyn	350
	Underwood, Camille	320
	Moreno, Franklin	280
	Kendrick, Anna	200
	Evans, Penelope	150

Each user records contains:
Last, First Name
Position, Department
and Points

Semi-transparent
background image

f. Award Activity



E) Inspiration Rewards API

Data can be created, read, updated, etc. via the Inspiration Rewards API. The root endpoint for the Inspiration Rewards API is:

Base URL = <http://inspirationrewardsapi-env.6mmagpm2pv.us-east-2.elasticbeanstalk.com>

Note: the API calls detailed below require you to provide your IIT student ID number (your "A" number) as a part of the data the API required. You MUST use your own student ID. *Your student ID number begins with an "A" and is followed by 8 numbers.*

The API:

1) Create User Profile:

Base URL + "/profiles"

Connection:

Request Method: POST

Request Property: "Content-Type": "application/json; charset=UTF-8"

Request Property: "Accept", "application/json"

JSON to Send:

```
{
  "studentId": "A00000008",
  "username": "user888",
  "password": "pass888",
  "firstName": "Roman",
  "lastName": "Walters",
  "pointsToAward": 1000,
  "department": "Finance",
  "story": "Chartered Financial Auditor with 5+ years' experience,
    seeking to leverage proven cost, revenue, and budget
    maximization skills for Capital One. Saved HUD A Inc. $2.7
    million by identifying low-margin projects. Improved pricing
    scheme at MRI International. Grew customer retention 32%.",
  "position": "Sr. Auditor",
  "admin": false,
  "location": "Mountain View, California", ← From Location Services
  "imageBytes": "\\9j\\4AAQSkZJBQA.....K/hfdrDCF\\//2=\\n", ← Very Long
  "rewardRecords": []
}
```

Returns (Profile as JSON):

URLConnection.HTTP_OK:

```
{
  "studentId": "A00000008",
  "firstName": "Roman",
  "lastName": "Walters",
  "username": "user888",
  "department": "Finance",
  "story": "Chartered Financial Auditor with 5+ years' experience,
    seeking to leverage proven cost, revenue, and budget
    maximization skills for Capital One. Saved HUD A Inc. $2.7
```

```

        million by identifying low-margin projects. Improved pricing
        scheme at MRI International. Grew customer retention 32%.",
        "position": "Sr. Auditor",
        "password": "pass888",
        "oldPassword": null,
        "pointsToAward": 1000,
        "admin": false,
        "imageBytes": "/9j/4AAQSkZJRgABA...../hfdrDCF//2Q==\n", ← Very Long
        "location": "Mountain View, California",
        "rewards": null
    }
}
Error: (i.e., HttpURLConnection.HTTP_BAD_REQUEST, HttpURLConnection.HTTP_FORBIDDEN, etc)
{
    "errordetails": {
        "timestamp": "10-03-2019 06:27:43",
        "status": "BAD_REQUEST", ← This varies by error type, not needed
        "message": "Invalid input - a profile with this username already
                    exists.", ← Use as a message to user
        "subErrors": [] ← Not used for this call/response
    }
}

```

2) Login:

Base URL + "/login"

Connection:

Request Method: POST

Request Property: "Content-Type": "application/json; charset=UTF-8"

Request Property: "Accept", "application/json"

JSON to Send:

```

{
    "studentId": IIT A-Number (ID Number),
    "username": User Name,
    "password": Password
}

```

Example:

```

{
    "studentId": "A12345678",
    "username": "user555",
    "password": "pass555"
}

```

Returns (Profile as JSON):

```

HttpURLConnection.HTTP_OK:
{
    "studentId": "A00000008",
    "firstName": "Olivia",
    "lastName": "Combs",
}

```



```
"username": "user555",
"department": "Legal",
"story": "The subject of law fascinates me extensively. I am intrigued
        by developments in the law and the way that it adapts to an
        ever-changing society.",
"position": "Contract Attorney ",
"password": "pass555",
"oldPassword": null, ← This field is not used in this call/return
"pointsToAward": 1000,
"admin": false,
"imageBytes": "/9j/4AAQSkZJRgABAQ.....AAQABAAD/ \nDgE6v//Z\n", ← Very Long
"location": "Chicago, Illinois",
"rewards": [{
    "studentId": "A00000008",
    "username": "user789",
    "name": "Kendrick, Anna",
    "date": "03/22/2017",
    "notes": "Great part of the team!",
    "value": 150
  }, {
    "studentId": "A00000008",
    "username": "user789",
    "name": "Wilkinson, Katherine",
    "date": "01/11/2017",
    "notes": "Thanks for the late hours",
    "value": 100
  }]
}

Other: (i.e., HttpURLConnection.HTTP_BAD_REQUEST, HttpURLConnection.HTTP_UNAUTHORIZED , etc)
{
  "errordetails": {
    "timestamp": "10-03-2019 06:38:25",
    "status": "UNAUTHORIZED", ← This can vary by error type
    "message": "Invalid username/password combination.", ← Use as a
    "subErrors": [] ← Not used for this call/response message to user
  }
}
```

3) Update User Profile:

Base URL + "/profiles"

Connection:

Request Method: PUT

Request Property: "Content-Type": "application/json; charset=UTF-8"

Request Property: "Accept", "application/json"

JSON to Send: Profile as JSON


```
{
  "studentId": "A00000008",
  "username": "user555",
  "password": "pass555",
  "firstName": "Penelope",
  "lastName": "Evans",
  "pointsToAward": 825,
  "department": "Sales",
  "story": "I am a mature, positive and hardworking individual, who
    always strives to achieve the highest standard possible, at
    any given task. In my previous role as a Sales
    Representative, I demonstrated the ability to work under
    intense pressure, sell products and services to customers
    from all backgrounds, and handle customer complaints.",
  "position": "VP West Coast Sales",
  "admin": true,
  "location": "Mountain View, California", ← From Location Services
  "imageBytes": "\\9j\\4AAQSkZJRgAB.....\\nDgE6v\\//Z\\n", ← Very Long
  "rewardRecords": [{
    "name": "Jacquelyn Blankenship",
    "date": "03\\19\\2019",
    "notes": "Good work on the quarterly sales reports!",
    "value": 100
  }, {
    "name": "Gavyn Montgomery",
    "date": "03\\19\\2019",
    "notes": "Thanks for all the help in organizing the outing!",
    "value": 50
  }]
}
```

Returns:

URLConnection.HTTP_OK:

Profile Updated Successfully ← This is text, not JSON

Other: (i.e., HttpURLConnection.HTTP_BAD_REQUEST, HttpURLConnection.UNAUTHORIZED, etc)

```
{
  "errordetails": {
    "timestamp": "10-03-2019 06:38:25",
    "status": "UNAUTHORIZED", ← This can vary by error type
    "message": "Invalid username/password combination.", ← Use as a
    "subErrors": [] ← Not used for this call/response message to user
  }
}
```

4) Get All Profiles:

Base URL + "/allprofiles"

Connection:

Request Method: POST

Request Property: "Content-Type": "application/json; charset=UTF-8"

Request Property: "Accept", "application/json"

JSON to Send:

```
{
    "studentId": IIT A-Number (ID Number),
    "username": User Name,
    "password": Password
}
```

Example:

```
{
    "studentId": "A12345678",
    "username": "user555",
    "password": "pass555"
}
```

Returns:

URLConnection.HTTP_OK:

```
[{
    "studentId": "A00000008",
    "firstName": "Katherine",
    "lastName": "Wilkinson",
    "username": "user000",
    "department": "Finance",
    "story": "Chartered Financial Analyst with 5+ years' experience,
        seeking to leverage proven cost, revenue, and budget
        maximization skills for Capital One. Saved HUD Inc. $2.7
        million by identifying low-margin projects. Improved pricing
        scheme at MRI International. Grew customer retention 32%.",
    "position": "Tax Manager",
    "password": null, ← Not used for this call/response
    "oldPassword": null, ← Not used for this call/response
    "pointsToAward": 160,
    "admin": false,
    "imageBytes": "/9j/wQrsOyx1KDVMq/z.....Dlx9o4avxfY80xLD=\n", ← Very Long
    "location": "Hollywood, CA 90210",
    "rewards": [{
        "studentId": "A00000008",
        "username": "user000",
        "name": "Kendrick, Anna",
        "date": "03/22/2017",
        "notes": "Great part of the team!",
        "value": 150
    }, {
        "studentId": "A00000008",
```

```

        "username": "user000",
        "name": "Kendrick, Anna",
        "date": "01/22/2017",
        "notes": "Good work!",
        "value": 75
    }
  ],
  {
    "studentId": "A000000008",
    "firstName": "Penelope",
    "lastName": "Evans",
    "username": "user002",
    "department": "Sales",
    "story": "I am a mature, positive and hardworking individual, who
              always strives to achieve the highest standard possible, at
              any given task. In my previous role as a Sales
              Representative, I demonstrated the ability to work under
              intense pressure, sell products and services to customers
              from all backgrounds, and handle customer complaints.",
    "position": "VP West Coast Sales",
    "password": null,
    "oldPassword": null,
    "pointsToAward": 1000,
    "admin": false,
    "imageBytes": "/9j/75MGeJrR507.....5MG6MM\nB3+g4J1f/9k=\n",
    "location": "Chicago, Illinois",
    "rewards": null
  },
  {
    ...
  },
  {
    ...
  }
]

```

Other: (i.e., HttpURLConnection.HTTP_BAD_REQUEST, HttpURLConnection.HTTP_FORBIDDEN, etc)

```

{
  "errordetails": {
    "timestamp": "10-03-2019 06:38:25",
    "status": "UNAUTHORIZED",
    "message": "Invalid username/password combination.",
    "subErrors": []
  }
}

```

5) **Add Rewards:**

Base URL + "/rewards"

Connection:

Request Method: POST

Request Property: "Content-Type": "application/json; charset=UTF-8"

Request Property: "Accept", "application/json"

JSON to Send:

```
{
  "target": { ← This is the user who gets the award
    "studentId": "A00000008",
    "username": "user456",
    "name": "Olivia Combs",
    "date": "03\10\2019", ← Use current date
    "notes": "Thanks for the help!",
    "value": 100
  },
  "source": { ← This is the user who gives the award
    "studentId": "A00000008",
    "username": "user555",
    "password": "pass555"
  }
}
```

Returns:

URLConnection.HTTP_OK:

Reward Added Successfully ← This is text, not JSON

Other: (i.e., HttpURLConnection.HTTP_BAD_REQUEST, HttpURLConnection.HTTP_FORBIDDEN, etc)

```
{
  "errordetails": {
    "timestamp": "10-03-2019 06:38:25",
    "status": "UNAUTHORIZED", ← This can vary by error type
    "message": "Invalid username/password combination.", ← Use as a
    "subErrors": [] ← Not used for this call/response message to user
  }
}
```


F) Additional Utility API Commands *(not required, but useful when testing)*

The below three API calls are *NOT* needed by the app, but might be useful for you when developing and testing your app. These 3 calls are 1) Reset all Rewards *(this will remove all rewards given to other users and reset all user's points-to-give totals)*. 2) Delete User *(this will delete the specified user)*. 3) Delete All Users *(this call deletes all users for the specified user id)*.

1) Reset All Rewards

Base URL + `"/resetprofiles"`

Connection:

Request Method: POST

Request Property: "Content-Type": "application/json; charset=UTF-8"

Request Property: "Accept", "application/json"

JSON to Send:

```
{
    "studentId": IIT A-Number (ID Number),
    "username": User Name,
    "password": Password
}
```

Example:

```
{
    "studentId": "A12345678",
    "username": "user555",
    "password": "pass555"
}
```

Returns:

URLConnection.HTTP_OK:

Profiles Reset Successfully ← This is text, not JSON

Other: (i.e., HttpURLConnection.HTTP_BAD_REQUEST, HttpURLConnection.HTTP_FORBIDDEN, etc)

```
{
    "errordetails": {
        "timestamp": "20-03-2019 06:38:25",
        "status": "UNAUTHORIZED", ← This can vary by error type
        "message": "Invalid username/password combination.",
        "subErrors": [] ← Not used for this call/response
    }
}
```

2) Delete User

Base URL + "/profiles"

Connection:

Request Method: DELETE

Request Property: "Content-Type": "application/json; charset=UTF-8"

Request Property: "Accept", "application/json"

JSON to Send:

```
{
    "admin": {
        "studentId": "A00000008",
        "username": "user911",
        "password": "pass911"
    },
    "username": "user888"
}
```

Returns:

URLConnection.HTTP_OK:

Profile Deleted Successfully ← This is text, not JSON

Other: (i.e., HttpURLConnection.HTTP_BAD_REQUEST, HttpURLConnection.HTTP_FORBIDDEN, etc)

```
{
    "errordetails": {
        "timestamp": "20-03-2019 06:38:25",
        "status": "UNAUTHORIZED", ← This can vary by error type
        "message": "Invalid username/password combination.",
        "subErrors": [] ← Not used for this call/response
    }
}
```

3) Delete All Users

Base URL + "/allprofiles"

Connection:

Request Method: DELETE

Request Property: "Content-Type": "application/json; charset=UTF-8"

Request Property: "Accept", "application/json"

JSON to Send:

```
{
    "admin": {
        "studentId": "A00000008",
        "username": "user911",
        "password": "pass911"
    },
    "username": "user888"
}
```

Returns:

URLConnection.HTTP_OK:









Profiles Deleted Successfully ← This is text, not JSON

Other: (i.e., HttpURLConnection.HTTP_BAD_REQUEST, HttpURLConnection.HTTP_FORBIDDEN, etc)

```
{
  "errordetails": {
    "timestamp": "20-03-2019 06:38:25",
    "status": "UNAUTHORIZED", ← This can vary by error type
    "message": "Invalid username/password combination.",
    "subErrors": [] ← Not used for this call/response
  }
}
```


G. Provided Images

The following image assets are provided for you to use with this project:

arrow_with_logo.png		icon.png	
background.png		login_people.jpg	
default_photo.png		logo.png	 (also used as launcher icon)
		name_image.png	
		separator.png	

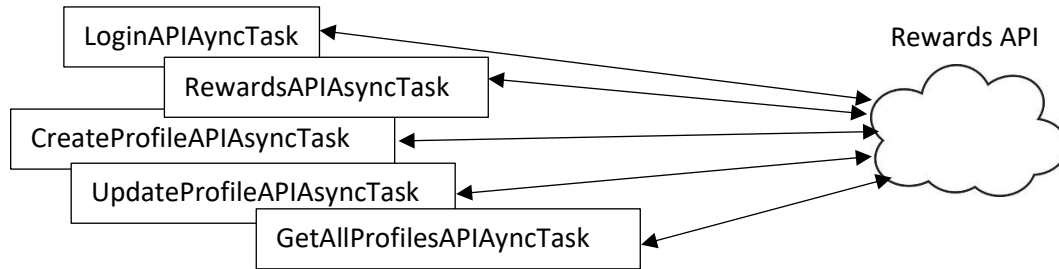
H. Menu Icons

For the required menu icons, you can use the build-in android menu icons:

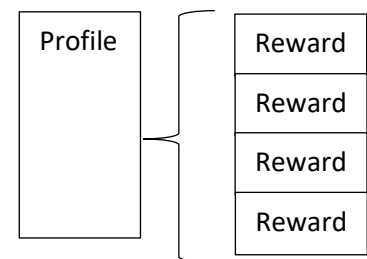
ic_menu_edit		ic_menu_sort_by_size	
ic_menu_save			

I. Other Classes

- You will need AsyncTasks to execute the API calls and to receive their returned data.



- You will need to create a class to represent the user profiles
- You will need to create a class to represent the rewards a user receives.



Assignment Assistance

The TAs for our course is available to assist you with your assignment if needed. Questions on assignment requirements and course concepts can be sent to the instructor.

Submissions & Grading

- Submissions must consist of your zipped project folder (*please execute Build =>Clean Project before generating the zip file*).
- Submissions should reflect the concepts and practices we cover in class, and the requirements specified in this document.
- Grading will be based upon the presence and proper functionality of all features and behaviors described in this document.

NOTE

This assignment is worth 400 points. This means (for example) that if you get 89% on this assignment, your recorded score will be:

$$(89\% * 400 \text{ points} = 356 \text{ points})$$

If you do not understand anything in this handout, please ask.

Otherwise the assumption is that you understand the content.

Unsure? Ask!