INFO 6350 Spring 2020
Rishi Jatania NUID: 001449023

Roomies App

Summary

I have developed an application for students that share an apartment together to resolve the informal communication and monetary problem between roommates by integrating the following features: -

- Add/ Maintain to-buy Lists: - Like maintain a shopping list.
- Add /Maintain / Assign ad-hoc to-do task between flat mates
- Create/ Maintain weekly task schedules related to daily chores like cooking, cleaning etc.
- Create a swap board to exchange tasks previously assigned.

Approach

Stack

Backend:

- Java
- Spring Boot
- Maven
- Hibernate
- JPA
- Spring Security
- JavaX Mail
- Log4j

Client:

- IOS App (Swift)

Database:

- MySQL

Bug Reporting Tool

- Slack

Object Model

1. **ROLE**
    a. Type
2. **Group**
    a. Id
    b. Group Name
    c. Group Description
    d. Users <**User**>
3. **User**
    a. Id
    b. Username
    c. Email
    d. Password
    e. Roles <**Role**>
    f. First name
    g. Last name
    h. Phone no
    i. Group <**Group**>
    j. Tasks <**TaskImpL**>
4. TaskImpl
    a. Task Id
    b. Task Name
    c. Task Description
    d. Creation Date
    e. Completion Date
    f. Task Type
    g. Status
    h. Added by User <**User**>
    i. User in Charge <**User**>
    j. Group <**Group**>
    k. Task up for Swap
5. Item <**TaskImpl**>
    a. Item Name
    b. Item Price
    c. Shared Users <User>
    d. Purchased on Date
    e. Bought By user <User>

Token Based User Authentication

Token-based authentication is a security technique that authenticates the users who attempt to log in to a server, a network, or some other secure system, using a security **token** provided by the server. Some Key benefits of token-based authentication are –

1. **Cross-domain / CORS:** Makes Authentication available across different domains as cookies don't work across domains.
2. **Stateless:** There is no need to keep a session store, the token is a self-contained entity that conveys all the user information
3. **Mobile ready:** When working with Mobile platforms cookies are ideal for session management.

Design Patterns used:

1. Singleton Design Pattern
2. Delegate Pattern

Flow:

Functionality: -

- **Register**: -
    - A user can register in the application by providing user's first name, last name, phone no
    - A user also enters password which later gets saved in the data base in an encrypted format using the Bcrypt library.

- **Login**: -
    - A registered user can only log into the system via the login screen
    - Once logged in the user info is saved in the session.

- **Group Creation/joining Screen**: -
    - A page where user can join or create and own a group
    - User can send invites to other user to join the group via mail

- **Group Dashboard**
    - Priority Board page showing filtered scheduled tasks divided among all users.

- **Swap Board**
    - A Dashboard showing all the task that users among the group have put up for swap. So, if a User is having some how busy during the assigned task, he can raise a swap request and other users in the swap board can swap their tasks with the one on the swap board

- **Shopping List:**

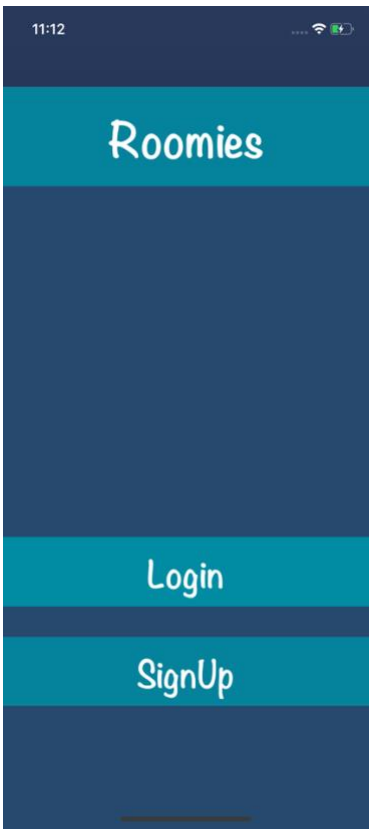
- Priority Board page showing filtered shopping items planned and completed.

- **Task List:**
  - Screen where user can find scheduled task assigned to them
  - Users can mark task complete or edit them on this though this tab.

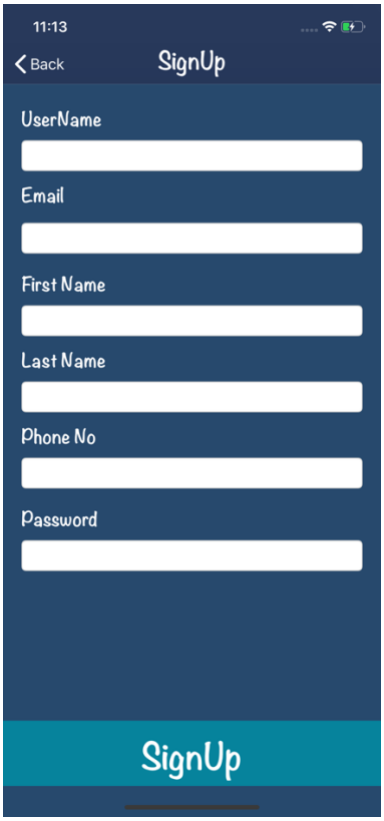
- **Account**
  - **Profile**
    - User is able view and update his profile information from this screen.
    - Users wishing to logout of the application can logout from this screen.
  - **Group Info**
    - Users can view group information like group name, description and the list of users who are part of the group.
    - Here You can also invite Users to join the group via email.
  - **Bug**
    - Since the App is in its initial stages and it is always difficult to track client-side bugs. Bug reporting feature was added so a user can report any issue faced directly to a Slack private slack channel which can subscribed by the developers working on the app to fix the issues at the earliest.

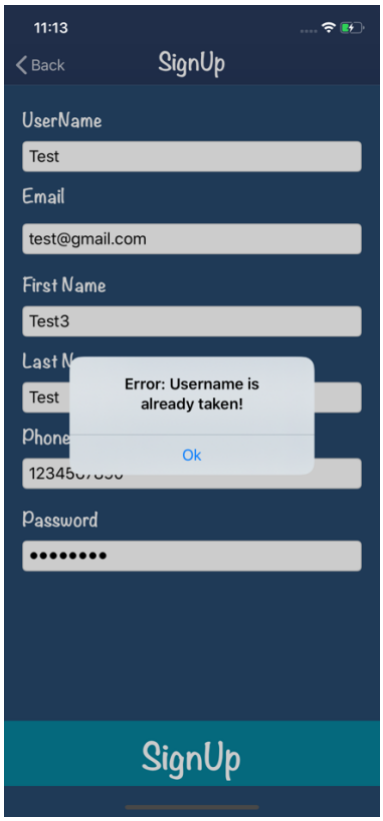
Screens

1. **HOME**

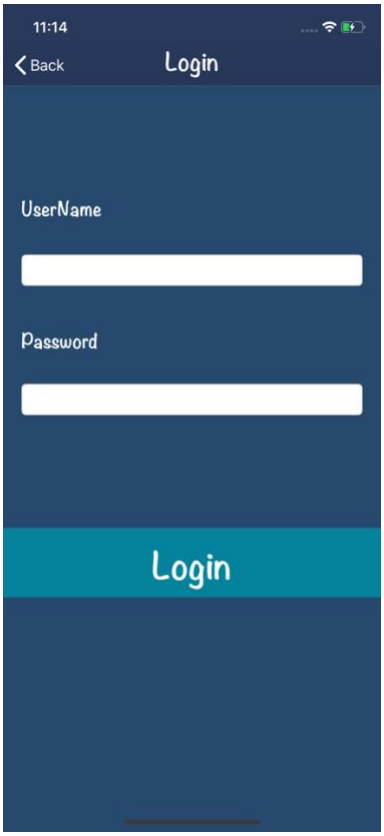


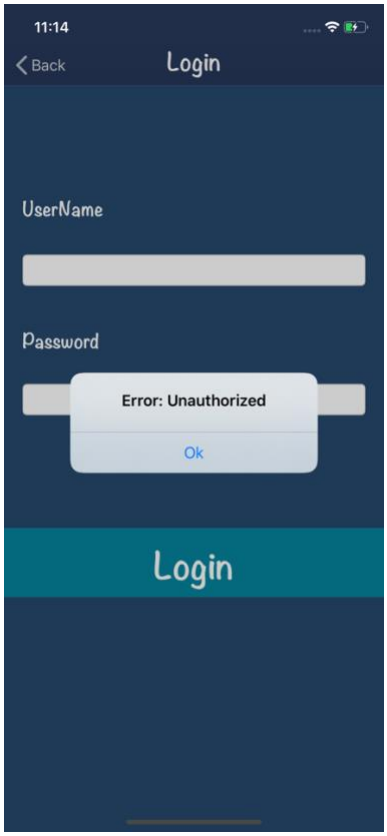
2. **SIGN UP**


11:13
Back
SignUp
UserName
Email
First Name
Last Name
Phone No
Password
SignUp

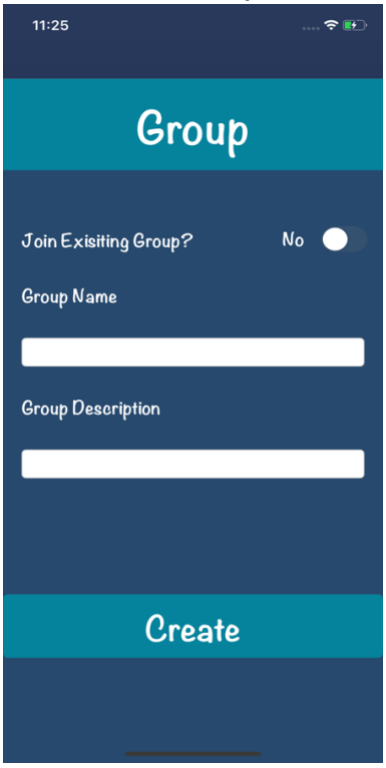

11:13
Back
SignUp
UserName
Test
Email
test@gmail.com
First Name
Test3
Error: Username is already taken!
Ok
Password
SignUp

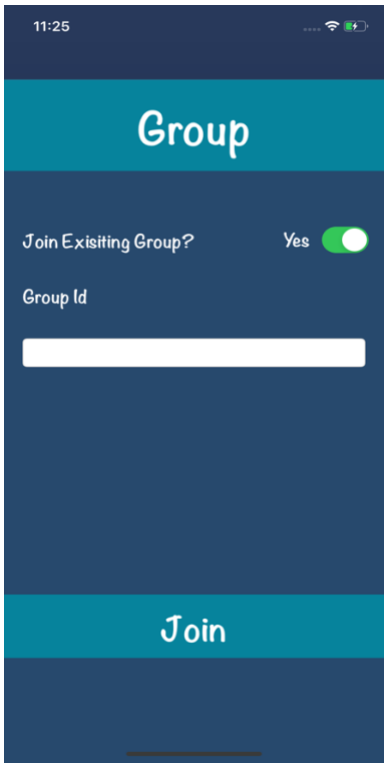
3. LOGIN


11:14
Back
Login
UserName
Password
Login

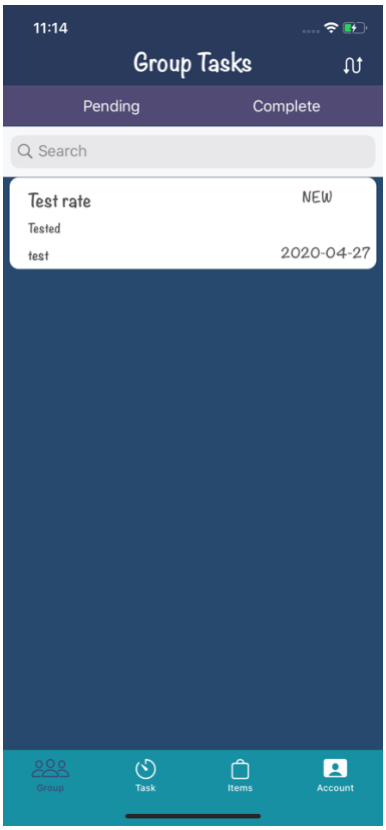

11:14
Back
Login
UserName
Password
Error: Unauthorized
Ok
Login

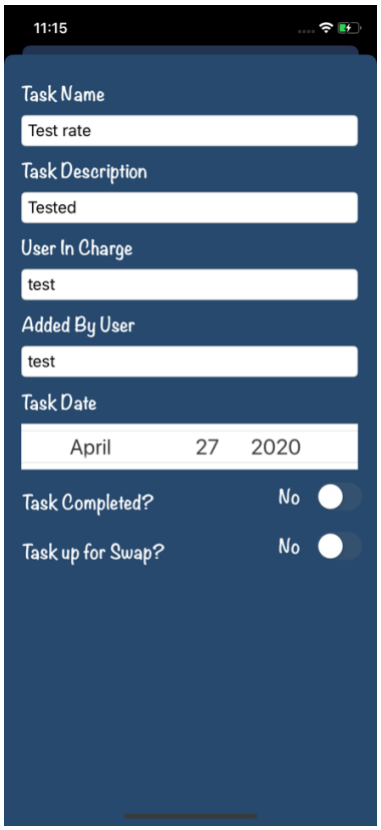
4. GROUP CREATION/JOINING



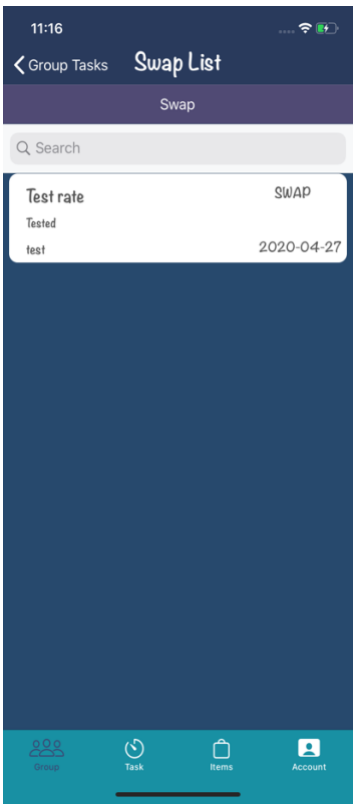


5. GROUP DASHBOARD

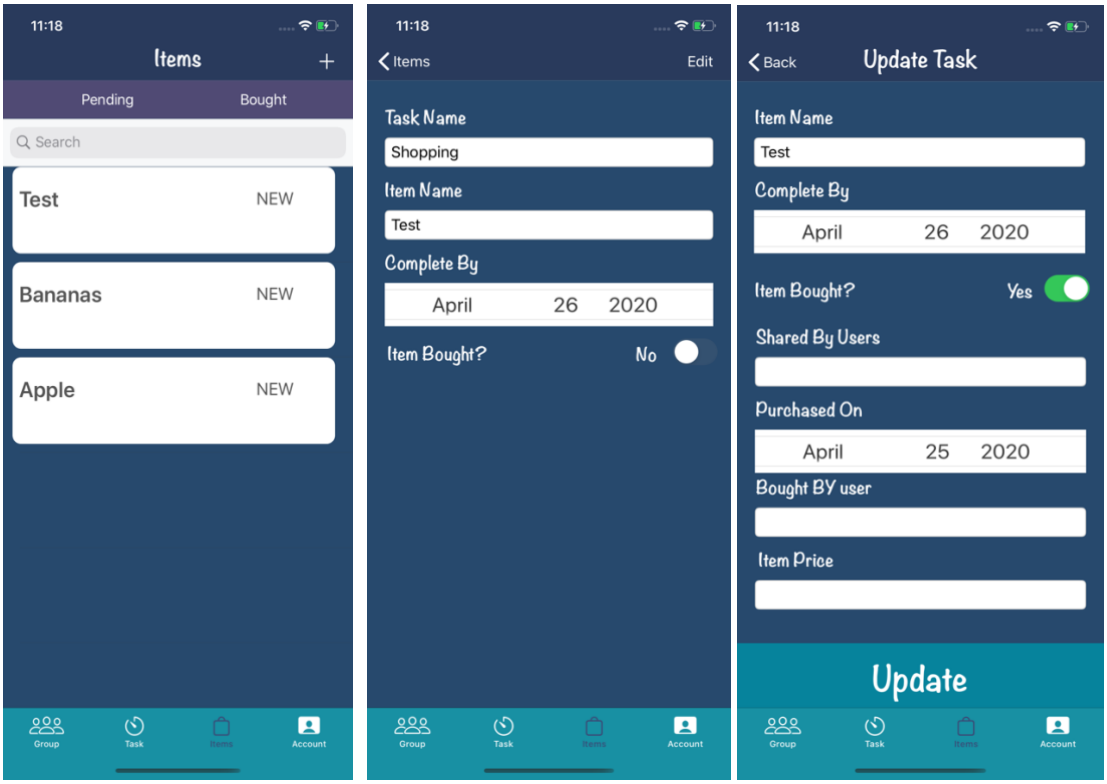




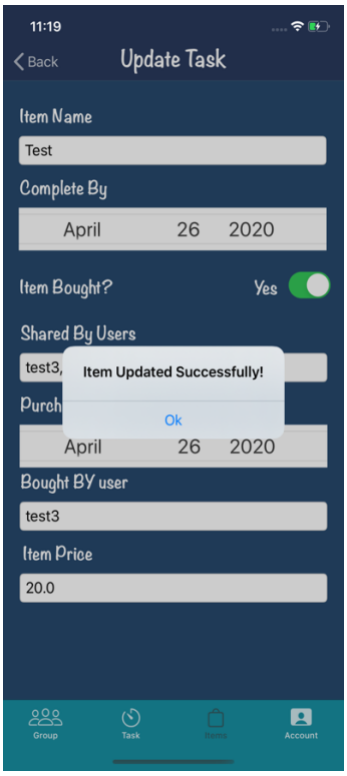
6. SWAP DASHBOARD

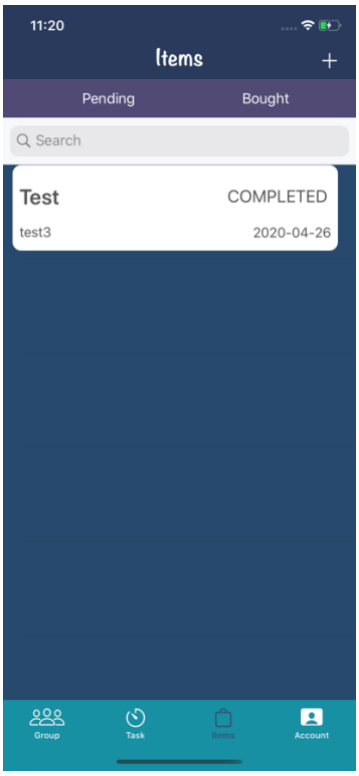


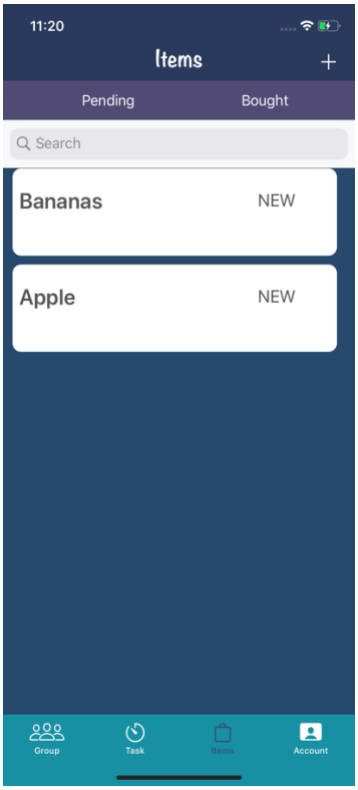
7. SHOPPING LIST




11:19
Update Task
Search
test GRADMIN
test@test.com Test Test
test3 USER
test3@test.com Test3 3
test2 USER
test@!test.com Test Test
Done
Group
Task
Items
Account


11:19
Back
Update Task
Item Name
Test
Complete By
April 26 2020
Item Bought?
Yes
Shared By Users
test3,
Item Updated Successfully!
Ok
Purch
April 26 2020
Bought BY user
test3
Item Price
20.0
Group
Task
Items
Account


11:20
Items
Pending
Bought
Search
Test COMPLETED
test3 2020-04-26
Group
Task
Items
Account

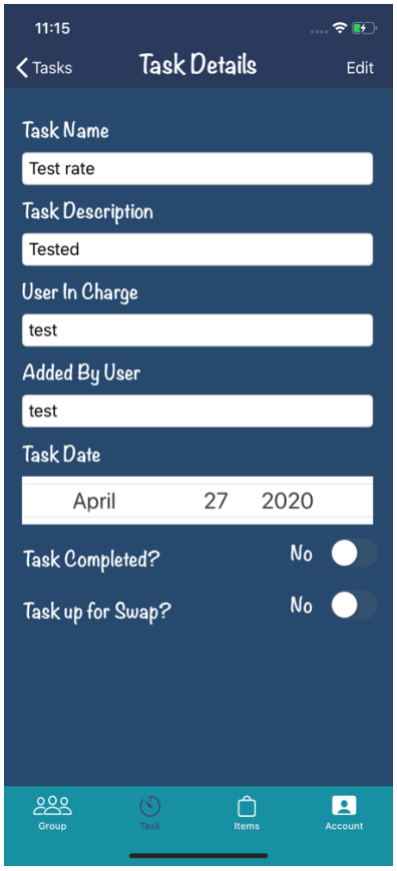

11:20
Items
Pending
Bought
Search
Bananas NEW
Apple NEW
Group
Task
Items
Account

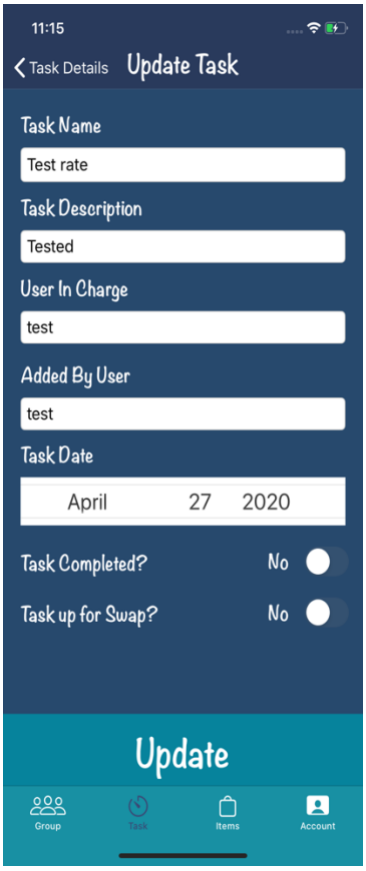

11:18
Items
Create Item
Item Name
Apple
Complete By
April 25 2020
Item Bought?
No
Create
Group
Task
Items
Account

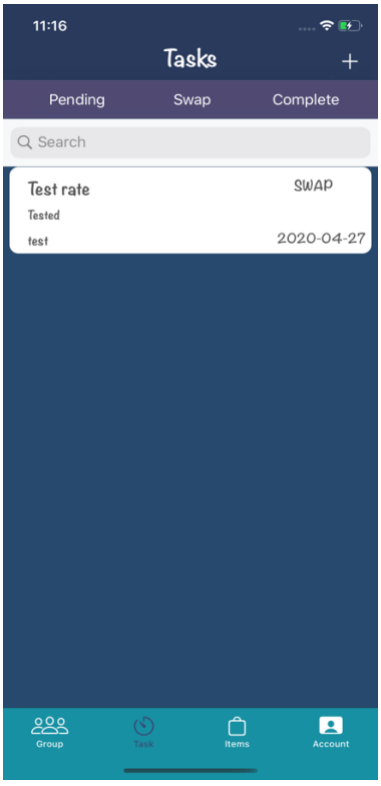

11:17
Items
Pending
Bought
Search
NEW
Delete
Group
Task
Items
Account

8. TASK LIST


11:15
Tasks
Pending
Swap
Complete
Search
Test rate
NEW
Tested
test
2020-04-27
Group
Task
Items
Account

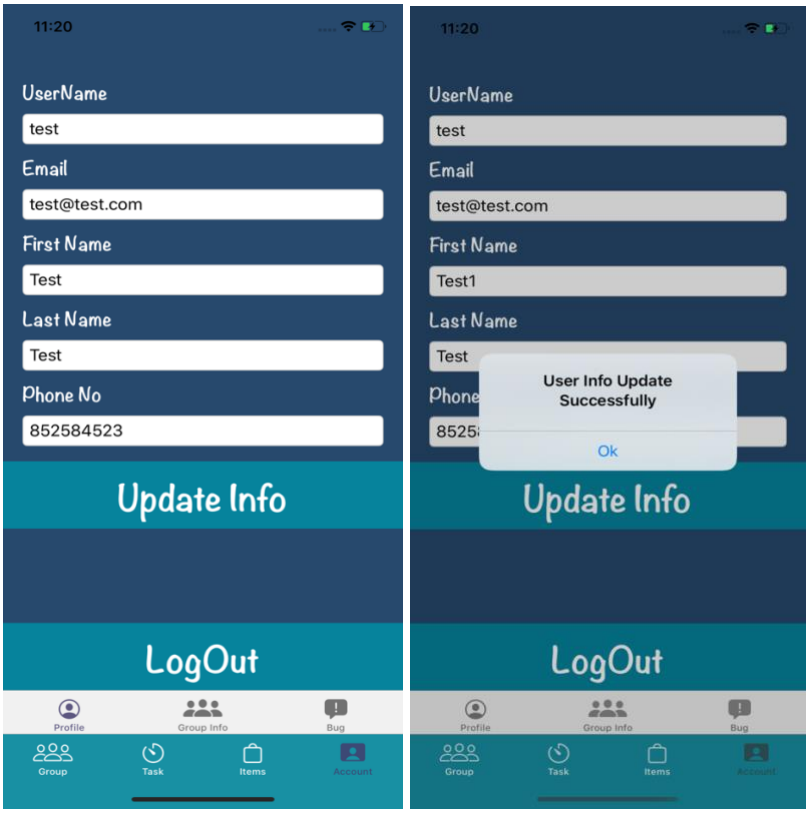

11:15
Tasks
Task Details
Edit
Task Name
Test rate
Task Description
Tested
User In Charge
test
Added By User
test
Task Date
April
27
2020
Task Completed?
No
Task up for Swap?
No
Group
Task
Items
Account


11:15
Task Details
Update Task
Task Name
Test rate
Task Description
Tested
User In Charge
test
Added By User
test
Task Date
April
27
2020
Task Completed?
No
Task up for Swap?
No
Update
Group
Task
Items
Account

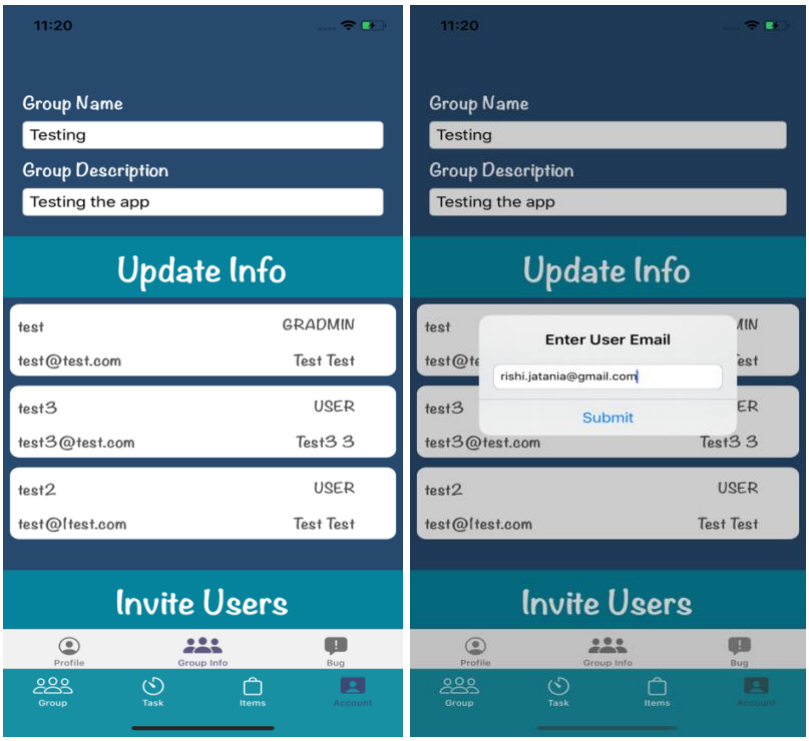

11:16
Tasks
Pending
Swap
Complete
Search
Test rate
SWAP
Tested
test
2020-04-27
Group
Task
Items
Account

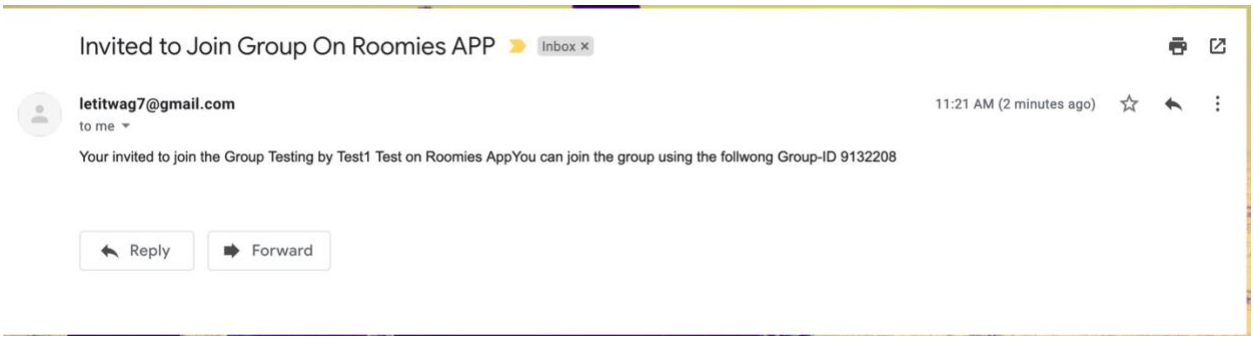
9. ACCOUNT

a. PROFILE

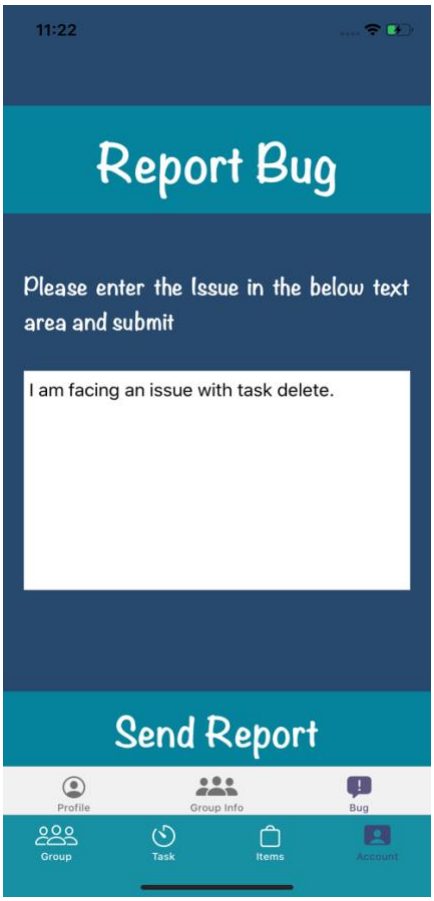


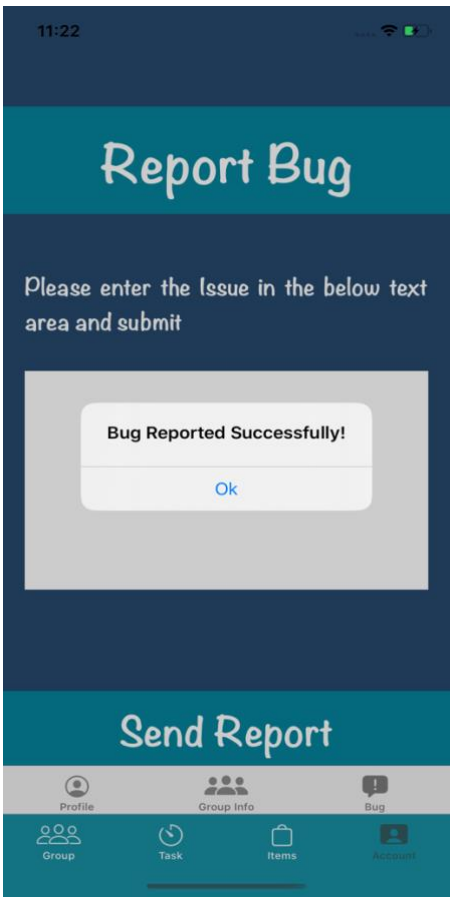
b. GROUP INFO

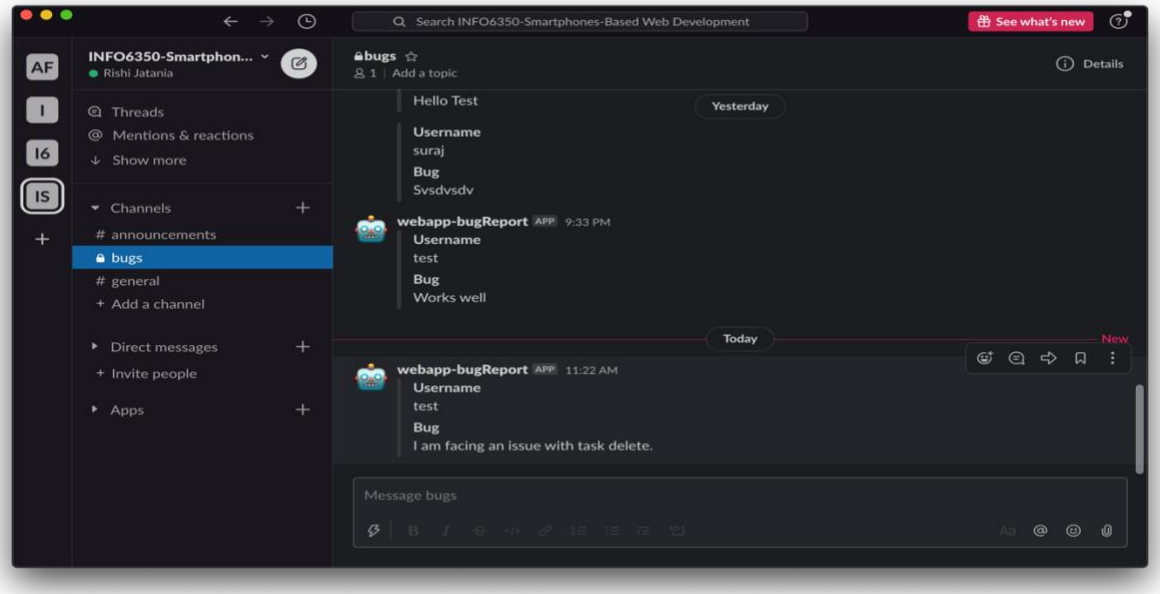



Invited to Join Group On Roomies APP
Inbox ×
letitwag7@gmail.com
to me
11:21 AM (2 minutes ago)
Your invited to join the Group Testing by Test1 Test on Roomies AppYou can join the group using the follwong Group-ID 9132208
Reply
Forward

c. BUG REPORT


11:22
Report Bug
Please enter the Issue in the below text area and submit
I am facing an issue with task delete.
Send Report
Profile
Group Info
Bug
Group
Task
Items
Account


11:22
Report Bug
Please enter the Issue in the below text area and submit
Bug Reported Successfully!
Ok
Send Report
Profile
Group Info
Bug
Group
Task
Items
Account



Future Scope

For the future scope of the Project following ideas can be implemented.

- Add a shared items expense tracker and way to settle the expenses and manage finances This can be done by coupling mobile payment app like Venmo for settling up expenses
- Enable Push Notifications for any and all activity within the group and its users.
  (I was unable to implement this feature without having a paid Apple developer account.)
- Simplify adding shopping items by scanning barcode and getting the product information thought a simple google search.
  (I was unable to implement this feature due to the restrictions of the simulator.)
- Implementing Single Sign On feature to integrate Federation sign in using Oauth2 which uses the same principle of Token based Authentication currently implement in the project.