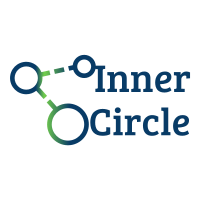
**CS673 Software Engineering** 

**Team 1 - Project Name**

**Tests Report**

|  |  |  |  |
| --- | --- | --- | --- |
| Team Member | Role(s) | Signature | Date |
| Wasupol | Team Leader | *WT.* | 09/23/2020 |
| Peter | QA Leader | *PL* | 11/12/2020 |
| Jian Ma | Backup Leader | *JM* | 11/3/2020 |
| Ananya | Requirement Leader | *AS* | 11/11/2020 |
| Michael | Implementation Leader | *MR* | 11/13/2020 |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**Revision history**

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Author** | **Date** | **Change** |
| **1.0** | **Team 1** | **11/13/20** | **11/13/20** |
|  |  |  |  |

[Introduction](#_87t9hln2vjz0)

[Test Summary](#_sm5odwyvuk3j)

[Tests Reports](#_pqso2mbjyzx4)

[Testing Metrics](#_mtfbusfb0eq3)

[References](#_15tmymhipvdv)

[Glossary](#_8n34lvocupub)

# Introduction

This document serves as a record of testing performed by all team members. This consists of testing summaries of acceptance tests, unit tests and UI tests that have been completed. In some cases,

# Test Summary

This section of the document summarizes the testing reports by test type or, in the absence of specific test cases, provides a status report of testing for that test type. This will be updated at the end of each iteration.

* Acceptance Tests: As of the end of iteration 3, we have completed acceptance tests for the core features of the application including logging in, initiating chats, searching for users, viewing profiles, initiating chat rooms and receiving notifications. Summaries of these tests are written in the tests reports section of this document. This includes tests of the firebase user creation processes and checks for existing users.
* Unit Tests: At the end of iteration 3 there have been a handful of unit tests written for the backend code.
* Integration Tests: There have been no reported tests of this type.
* UI Tests: UI tests have been written and recorded using Espresso. Summaries of these tests are listed in the test reports section.

# Tests Reports

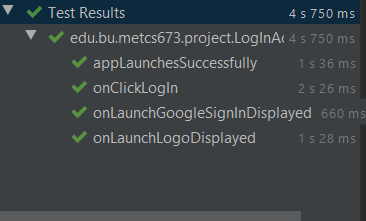
This section provides detailed descriptions of completed tests, organized by test type. Defects found through these tests will require generation of a bug report in pivotal tracker, that story should be linked here.

Acceptance Tests

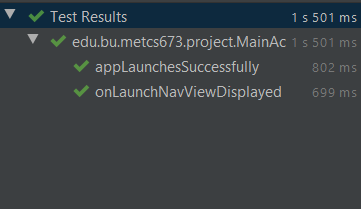
* App Startup
  + Test name: App initialization (first time)
  + New or old: Old
  + Test items: Initial login display
  + Test priority: High
  + Dependencies: None
  + Preconditions: User has installed the app and has not logged in before.
  + Input data: User clicks app icon from home screen
  + Test steps:
    - Built Gradle
    - Run emulator
    - Open app from home screen
  + Postconditions: The user is presented with the app logo and clickable box with the google logo and “Sign In” text.
  + Expected output: None
  + Actual output: N/A
  + Pass or Fail: Pass
  + Bug id/link: N/A
  + Additional notes:
* Login:
  + Test name: Login Test
  + New or old: Old
  + Test items: Google authorization and integration with app
  + Test priority: High
  + Dependencies: None
  + Preconditions: User’s android debug SHA-1 or SHA -256 key needs to be uploaded to the firebase instance.
  + Input data: Click the “Sign In” button
  + Test steps:
    - When google login view appears,
      * If first time: Provide gmail address and password
      * Subsequent tests: Either select account credentials added in prior test or add another account.
  + Postconditions: If the login is successful you view changes to the home feed.
  + Expected output: If the login is successful your view changes to the home feed If the login is unsuccessful (google authorization fails) text appears stating the failure.
  + Actual output: “”
  + Pass or Fail: Pass
  + Bug id/link: N/A
  + Additional notes:
* **Logoff:**
  + Test name: Logoff
  + New or old: Old
  + Test items: Ending user session
  + Test priority (high/medium/low): Medium
  + Dependencies (to other test case/requirement if any): Dependent on login test
  + Preconditions: User’s android debug SHA-1 or SHA -256 key needs to be uploaded to the firebase instance, user has logged in successfully
  + Input data: Click hamburger menu on app banner, click “Sign - Out”
  + Test steps: Login successfully, from feed view click the hamburger menu (three vertical bullet points in top right banner), from pop-up menu, click “Sign - Out.”
  + Postconditions: If the logout action is successful the login page should be displayed to the user.
  + Expected output: After clicking “Sign - Out” the user’s view changes to the login page.
  + Actual output: “”
  + Pass or Fail: Pass
  + Bug id/link: (this should link to your github issue id): N/A
  + Additional notes:
* **Create Chat Room:**
  + Test name: Create Chat Room
  + New or old: New
  + Test items: Creating new user chatroom
  + Test priority (high/medium/low): High
  + Dependencies: (to other test case/requirement if any): Dependent on login test
  + Preconditions: (if any): User has to be logged in successfully
  + Input data: Click friends tab, and click registered user to send a new message to
  + Test steps:
    - Login successfully
    - Then from the bottom navigation menu, click “Friends” and then click on the desired user to send message to.
    - Click “Send Message”
  + Postconditions: If the create chatroom session is successful, then a new chat page should be created between the user and intended recipient
  + Expected output: After clicking on a specific user when on the “Friends” fragment, the view changes to a chat display that allows the user to type chat messages.
  + Actual output:””
  + Pass or Fail:Pass
  + Bug id/link: (this should link to your github issue id) N/A
  + Additional notes:
* **Visit User Profile:**
  + Test name: Visit User Profile
  + New or old: New
  + Test items: Visit user profile page
  + Test priority (high/medium/low): medium
  + Dependencies: (to other test case/requirement if any): Dependent on login test and friends page test
  + Preconditions: (if any): User has to be logged in successfully; The friend page shows the list of users correctly.
  + Input data: Click friends tab, and click a registered user
  + Test steps:
    - Login successfully
    - Then from the bottom navigation menu, click “Friends” and then click on desired user
    - Click “view user’s profile”
  + Postconditions: N/A
  + Expected output: If the profile page shows smoothly, the user’s profile photo and name will appear.
  + Actual output:N/A
  + Pass or Fail:Pass
  + Bug id/link: (this should link to your github issue id) N/A
  + Additional notes:N/A
* **New Message Notification:**
  + Test name: Android notification test
  + New or old: New
  + Test items:
    - New message displays on android notification drawer,
    - Message text is displayed
    - Clicking the notification returns view to app.
  + Test priority (high/medium/low): medium
  + Dependencies (to other test case/requirement if any): Dependant on login test, create chatroom test.
  + Preconditions: User has an account, another user sends a message while the user is not actively viewing the app.
  + input data: Incoming message data
  + Test steps:
    - Login to app & switch your view to another app or the android home screen
    - Wait for incoming message
    - Receive and view notification in the notification drawer
    - Click notification and navigate back to the app.
  + Postconditions: View has returned to the app
  + Expected output: N/A
  + Actual output: N/A
  + Pass or Fail:Pass
  + Bug id/link: (this should link to your github issue id)
  + Additional notes:

UI Tests

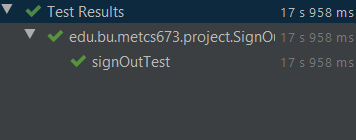
* LogInActivityTests: Tests display of login page UI elements and the functionality of the sign in button



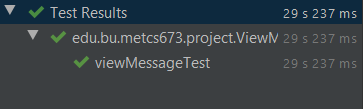
* + MainActivityTest: Tests display of nav menu in dashboard view



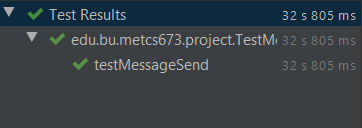
* + SignOutTest: Test functionality of signout menu interaction



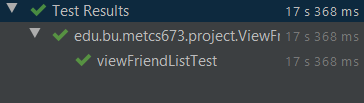
* + ViewMessageTest: Navigates to message view, opens top chatroom.



* + TestMessageSend: Send test message in existing chat room.



* + ViewFriendList: Access FrienListView



# Testing Metrics

* Product metrics:
  + Defects per lines of code: No defects have been reported at this time.
  + Amount of failed chat messages per a given reference number of chat messages (choice for reference number to be determined)
* Process metrics:
  + Test coverage rates by class: Pending unit test commits to our repository.
  + Defects resolved per an iteration: No defects have been identified at this time.
  + Man hours per a defect resolution: No defects have been identified/addressed at this time.

# References

Our project’s pivotal tracker <https://www.pivotaltracker.com/n/projects/2465407>

# Glossary