KASPER – ID4 PROJECT DOCUMENTATION

SOFTWARE DEVELOPMENT TEAM:

Project Manager: Tushita Patel

Dev Lead: Kristof Mercier, Dylan Prefontaine

Test Lead: Jeremy Liau

Build Manager: Christopher Mykota-Reid (ChrisMR)

Developers: Gaurav Arora, Haotian (Justin) Ma, Melody (Tian) Zhao

Test Team: Christopher May (Chris May), Ryan Tetland

Documentation: Arianne Butler

Contents

[0.0 Important Links 3](#_Toc477707105)

[1.0 Requirements Document ID4 3](#_Toc477707106)

[1.1 Front-end Requirements 3](#_Toc477707107)

[1.2 Back-end Requirements 3](#_Toc477707108)

[1.3 Mini Milestones for ID4 4](#_Toc477707109)

[2.0 Time Estimations 5](#_Toc477707110)

[3.0 Design 6](#_Toc477707111)

[3.1 API Document 6](#_Toc477707112)

[3.2 Development Unit Tests 7](#_Toc477707113)

[4.0 Testing Document 7](#_Toc477707114)

[4.1 Test Plan 7](#_Toc477707115)

[4.2 Manual Test Report 11](#_Toc477707116)

[4.3 Defect Report 12](#_Toc477707117)

[4.4 State Transition Diagrams 17](#_Toc477707118)

[4.5 Critical Use Cases 19](#_Toc477707119)

[5.0 Coding Style Guide 24](#_Toc477707120)

[6.0 Build Report 25](#_Toc477707121)

[7.0 Upcoming Requirements 27](#_Toc477707122)

[7.1 ID5 Requirements (Quality Assurance) 27](#_Toc477707123)

[7.2 Incomplete Requirements 27](#_Toc477707124)

[8.0 Triage Meeting ID4 28](#_Toc477707125)

# Important Links

**Slack**: <https://cmpt371group1.slack.com/messages/C3N24Q5EC/>

**Trello**:

* Front-end: <https://trello.com/b/S2XiqLAm/dev-id4-front-end>
* Back-end: <https://trello.com/b/m9OtfvS1/dev-id4-back-end>

**Activity Log**: <https://docs.google.com/spreadsheets/d/1g3CfvlI1erdg2zaHZlVyh6uQ-rYQ6L5Cay0YIn2UETk/edit#gid=0>

* Click on the bottom right tab labelled **Individual Activity Log** or the bottom right tab **Activity Log Totals**, for a break down of the activities

**GitHub Project**: <https://github.com/CMPT371Team1/Project>

**GitHub ID4 Documentation**: <https://github.com/CMPT371Team1/Documentation>

# Requirements Document ID4

## 1.1 Front-end Requirements

* Form control validation for all remaining pages
* Fix description box in Add Listings
* Display currently added images during upload image process
* Make Favourites page the same style as Browse page
* Get front-end fully integrated with back-end (done, except for Sign Out, Edit Listing, Edit User, and Sign in with Facebook which were moved to ID5)
* Finalize and complete UI flow

## 1.2 Back-end Requirements

* Email verification
* Logout
* Contact Seller (done in back end only)
* Sign in / Sign up using Facebook (done in back-end only)
* Delete Listing
* Get the front-end fully integrated with the back-end for all API calls (done except message API)
* Get messages/edit messages/contact seller (done in the back-end only)

## 1.3 Mini Milestones for ID4

**Development**:

* Integrate the front and back-end ✔
  + Not done for Sign Out, Edit Listing, Edit User, and Sign-In with Facebook ✘
* Finish messaging in the back-end ✔
* Finish all back-end API calls ✔

**Testing**:

* Do Peer Review board tasks (as many as possible) ✔
* Manual testing (Browser, Android) ✔
* Remove unused and out-of-date test files ✔
* Automate regression tests ✔
* Create separate configuration files for Firefox and Chrome testing ✔
* Get backend developers to create a test database for testing ✔
* Fix high priority bugs before Bug Party ✔
* Create new Use Cases for ID4, update old ones ✔
* Refactor and update E2E tests ✔
* Testing results and defect report ✔
* Fix smoke tests ✔
* Test matrix update ✔
* Path diagram update ✔
* Testing plan document update ✔
* Hallway Testing ✘
* Acceptance Testing ✘

**Build**:

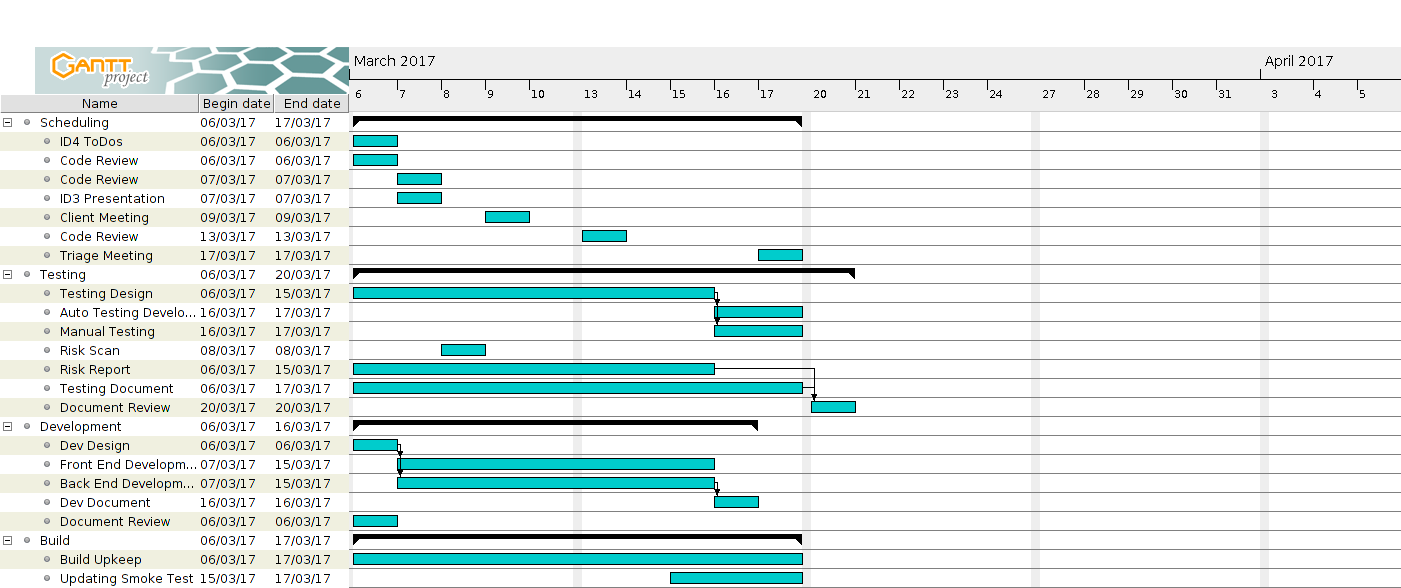
* Get broken smoke tests working ✔
* Integrate test server with the build ✔
* Make the build more automated ✔
* Add GitHub release automation ✘
* Add style checking ✘

**Documentation**:

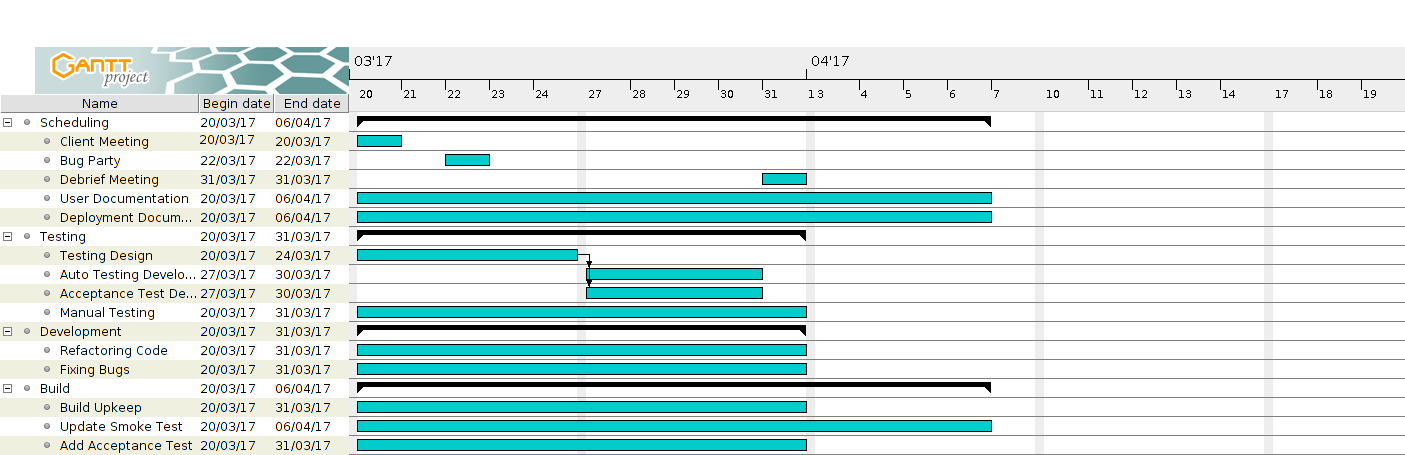
* Organize, compile, and edit all ID4 documentation ✔

# 2.0 Time Estimations

Gantt Diagram ID4:



Gantt Diagram ID5:



To see time estimations on all individual tasks, please follow Activity Log link in section 0.0 of this document, and click on the bottom tab right labelled **Individual Activity Log**.

Trello cards contain a coloured label indicating the approximate estimation of each task. These estimates were approximated by the dev team.

The following image shows the colour coded time estimation scale with which each trello card will be rated.



# 3.0 Design

## 3.1 API Document

The API document outlines the client server communication of our system. Our project contains a total of 18 API calls (calls to the server), 14 of which are integrated with the front-end. The API calls not yet linked with the front-end are the calls related to Facebook and messages.

Our API calls send a JSON body with relevant data. The server receives, parses, and processes the JSON using the database. When finished, the server will reply to the sender with a token containing the reply status and a JSON body with the requested information. The following link contains our updated ID4 API document and details on the various calls to our database.

Note: We are currently experiencing difficulty with this link. Please copy and paste it into your web browser.

[https://docs.google.com/document/d/1N4jt1\_PgxPhXwdc1TcT7TBjFNOZqYO5L10ha3bpO5M8/edit#heading=h.1sskatsa28we](https://docs.google.com/document/d/1N4jt1_PgxPhXwdc1TcT7TBjFNOZqYO5L10ha3bpO5M8/edit%23heading=h.1sskatsa28we)

## 3.2 Development Unit Tests

The procedure to be followed by all developers for writing both front and back-end unit tests respectively, can be found at the following links:

Front-end Guide for Unit Testing:

<https://github.com/CMPT371Team1/Project/wiki/Coding-Style-Example-(JavaScript)>

Back-end Guide for Unit Testing:

<https://github.com/CMPT371Team1/Project/wiki/Coding-Style-Guide-(Python)>

# 4.0 Testing Document

## 4.1 Test Plan

4.1.1 Test Strategy

**Critical Use-Cases ID4:**

* Register New User, Filter Listings, Add Listing

**Test Matrix:**

* A better color system and key were formed for improved readability. A better description was also added to explain the challenges the test team encountered while testing the application both manually and automatically.

**Manual Testing (Android and Browser):**

* Testing the application on phone devices took priority over testing in the browser, since these are the main platforms our app will run on. Browser testing is mostly limited to End-to-End tests. A log of the manual tests done during ID4 is included in this document.

**End-to-end testing:**

* Many builds failed this ID, but not due to failing tests. The smoke test was being mistakenly run on the main database, rather than the test database as planned. This took a few days to figure out, and was eventually fixed.

**Automated Regression tests:**

* Regression tests were automated to check if bugs from previous ID’s were re-appearing. So far, only 1 bug has appeared in the regression test: Square feet information is not being saved when saving a Listing without publishing.

**Second Bug Party:**

* The second bug party is planned for March 22nd /17. We’ll be using this as an opportunity to find bugs that were missed by the test team in ID4, and to give an estimation of the remainder of bugs left in the application.

**Hallway Testing:**

* Testing with random volunteers is being planned for ID5. We’ll set up a station and invite people to test our application in order to find bugs that were overlooked. Hallway testing will take place after most of the ID4 bugs have been fixed, and after the second bug party. Our goal is to have this test complete on March 29th /17. This will also help with our Acceptance testing for the external acceptance (Beta) testing of our application.

**Acceptance Testing:**

* Internal acceptance (Alpha) testing was done by the test team in ID4. Many of the features had major bugs that required fixing, so our goal is to fix these bugs, as well as hold our second bug party to find any others, before attempting the external acceptance test.

4.1.2 Training Requirements

* Testers learned how to run and reset the test database, change static data for easier end-to-end and manual testing, and decrease/increase randomization of users and Listings.
* Code refactoring was a high priority in ID4, so a large amount of time was spent reducing duplicate code using functions, adding better logging messages for easier debugging, and adding expectation statements (Jasmine’s version of assertions).
* Testers learned how to install the Jasmine Logger to their local machines, to help them create better logging messages, and to have a better idea of where the code is breaking. This logger was also installed on the continuous integration system.

4.1.3 Documentation

**Test Matrix:**

**Key:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Features** | **Unimplemented features** | **Tests** | **Tests access Features** | **Tests don’t access Features** |

The Facebook Sign-In feature is currently implemented in the back-end, but not connected to the front-end. Therefore, the testers were unable to test its functionality. Tests shown in orange are high level descriptions. They test both the main success scenario of the functionality, as well as variations in an attempt to break them. A large amount of tests have been automated, but some of them (Browse page swiping, publishing Listings and filtering in the Browse filter page) have been done manually. Automated regression tests have been created to check for previous bugs re-appearing in the application.

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **App Sign-In** | **Facebook Sign-In/Sign-up** | **Sign-Up** | **Browse Listings Filter** | **Publish New Listing** | **Save W/O Publishing Listing** | **My Profile/Change Password/Sign out** | **Browse** | **Favourites** | **My Listings** |
| **Sign In with existing user** |  |  |  |  |  |  |  |  |  |  |
| **Add a**  **Listing (Signed In)** |  |  |  |  |  |  |  |  |  |  |
| **Registering a new user** |  |  |  |  |  |  |  |  |  |  |
| **Cycle listings (Signed In)** |  |  |  |  |  |  |  |  |  |  |
| **Edit account information** |  |  |  |  |  |  |  |  |  |  |
| **Register an existing user** |  |  |  |  |  |  |  |  |  |  |
| **Invalid Sign-In credentials** |  |  |  |  |  |  |  |  |  |  |
| **Change account password** |  |  |  |  |  |  |  |  |  |  |
| **Like/Dislike a listing(own listing, or other user’s listing)** |  |  |  |  |  |  |  |  |  |  |

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **App Sign-In** | **Facebook Sign-In/Sign-up** | **Sign-Up** | **Browse Listings Filter** | **Publish New Listing** | **Save W/O Publishing Listing** | **My Profile/Change Password/Sign out** | **Browse** | **Favourites** | **My Listings** |
| **Weak password given** |  |  |  |  |  |  |  |  |  |  |
| **Navigation while signed-in** |  |  |  |  |  |  |  |  |  |  |
| **Navigation while not signed-in** |  |  |  |  |  |  |  |  |  |  |
| **Edit Favourites** |  |  |  |  |  |  |  |  |  |  |
| **Delete Listing** |  |  |  |  |  |  |  |  |  |  |

4.1.4 Schedule

**ID1:**

Testing was mostly done manually on the front end of the application. The test team began creation of automated tests using Protractor. Tests and functionality were recorded in the test matrix, and all bugs were listed in detail in the defect report.

**ID2:**

The Test Team attempted to integrate automated end-to-end tests with TravisCI, updated the defect report, updated the flow charts to better reflect the new requirements/current state of the system, and re-created, simplified, and updated the test matrix.

**ID3:**

The first bug party was held. Continued updating of the test matrix, defect report, and flow charts was practiced. Testing on Android devices was successful, and brought new defects to the defect report which were unique to device use. The test matrix was updated much more thoroughly, with a better description of the state of our tests. End-to-end tests had to be updated once again, due to the changes in the front-end, and other requirements changes. The flow charts also changed significantly, showing access to new functionality and areas of the program not previously implemented. Protractor was successfully integrated with TravisCI, which was an issue we were having in ID2.

**ID4:**

Large amounts of testing have been done on the features that were previously unimplemented/bugged in ID3. Favourites page, Like/Dislike functionality, browsing, filtering, and publishing Listings were the main focus of testing. Many bugs were found, and the high priority ones were fixed quickly by collaborating with the developers. Some bugs still remain in the application, and will continue to be fixed throughout ID5. A better logging system was installed onto the project for easier debugging. A test database was created to replace testing on the actual database. This test database resets whenever we require it to, which helps with recreating bugs and cleaning out the system after playing around with user registration, listing creation, and saving information to a user, such as liked/disliked listings and profile information.

**ID5:**

**-** The second bug party has been planned for March 22nd

**-** Hallway testing is planned for March 29th

- Regression tests will be updated to ensure no bugs are re-introduced

- Polishing the application will be a high priority for this ID. Our goal is to eliminate as many bugs as possible, starting with the most critical ones

## 4.2 Manual Test Report

**March 6th – 13th (Random manual tests during development)**

**Testers**: Jeremy, Ryan, Chris May

**Bugs Found**:

* Liking/Disliking a listing by swiping on Browse page breaks cycling of listings on Listing details page
* Cannot remove listings from Favourites page
* Save button on My Profile page not functioning
* Sign-out button on My Profile page not functioning
* [App design] Add Edit button for Favourites/My Listings to display edit/delete listing options

**March 14th (Bug fixing with developers):**

**Testers**: Chris May

**Bugs found**:

* Province abbreviation is not being displayed on listing details
* Using filters while not signed in will result in an userID error

**March 16th (During test team meeting after development phase):**

**Testers**: Jeremy, Ryan, Chris May

**Bugs found**:

* App is creating a white screen, HTML code has an error
* Liking/Disliking a listing from the detailed listing page that is already in your favourites list gives a runtime error
* Bathroom filter stops functionality of other filters
* Add Listing page: Cannot publish listings
* Disliking a listing results to the listing being added to the user's favourites list
* Square feet information is not being saved when saving a listing without publishing
* Browse Page: Liking/Disliking should reset the listing to the default position when an error message is given
* Browse page should not be displaying listings already liked/disliked by the user. A refresh is required
* No warning is given when attempting to publish a listing without an image

**March 17th (While fixing smoke tests)**

**Testers**: Jeremy, ChrisMR

**Bugs found**:

* Immediately after registering a new user, pressing the My Profile button will redirect to Sign-in page
* Smoke tests aren’t using the test database. Must switch from main database to test database.
* End-to-end tests are using main database users, which do not exist in the test database. Changing tests to use test users.

**March 18th (During bug fixing session):**

**Testers**: Jeremy, Tushita

**Bugs found**:

* Cannot delete published listing and saved unpublished listing by user
* Cannot publish a saved unpublished listing
* Cannot save edited changes to a saved unpublished listing
* [App design] Need a deletion confirmation when attempting to delete a listing (in Favourites and My Listings pages)
* [Regression] Square feet information is not being saved when saving a listing without publishing

## 4.3 Defect Report

**Bugs which were not fixed during ID3:**

|  |  |
| --- | --- |
| **Bug:** | Like/Dislike on “Browse” page breaks cycling of Listings on details page |
| **Reproducibility:** | 100% |
| **Steps taken:** | 1. Like/Dislike any listing by swiping on the “Browse” page 2. View detailed information of any listing, and attempt to cycle listings with arrow buttons |
| **Variations:** | None |
| **Resolved:** | **Yes** |
| **Screenshot:** | None |

|  |  |
| --- | --- |
| **Bug:** | Pop-up saying a list was liked or disliked pops up multiple times per swipe |
| **Reproducibility:** | 100% |
| **Steps taken:** | 1. Like/Dislike a listing on Browse page by swiping, either direction |
| **Variations:** | Swipe left for reproducing dislike pop-up  Swipe right for reproducing like pop-up |
| **Resolved:** | **Yes** |
| **Screenshot:** | None |

|  |  |
| --- | --- |
| **Bug:** | While signed in to any account, being on the “My Profile” tab and selecting “My Profile” tab again will show the sign-in screen. |
| **Reproducibility:** | 100% |
| **Steps taken:** | 1. Sign in on any account 2. Navigate to “My Profile” tab 3. Select “My Profile” tab |
| **Variations:** | None |
| **Resolved:** | **Yes** |
| **Screenshot:** | None |

|  |  |
| --- | --- |
| **Bug:** | Liking/Disliking a listing on the Listing’s detailed info page doesn’t remove it from the list |
| **Reproducibility:** | 100% |
| **Steps taken:** | 1. Navigate to detailed info page of any listing 2. Like/Dislike the listing |
| **Variations:** | Liking/Disliking on the “Browse” page correctly removes the listing. |
| **Resolved:** | **Yes** |
| **Screenshot:** | None |

|  |  |
| --- | --- |
| **Bug:** | Selecting a listing on the “Browse” page allows the user to navigate to the “My Listings tab” without being required to sign in first |
| **Reproducibility:** | 100% |
| **Steps taken:** | 1. Select any listing on the “Browse” page to navigate to the detailed info page 2. Navigate to “My Listings” tab |
| **Variations:** | None |
| **Resolved:** | **Yes** |
| **Screenshot:** | None |

|  |  |
| --- | --- |
| **Bug:** | No error prompt when login information is incorrect |
| **Reproducibility:** | 100% |
| **Steps taken:** | 1. Navigate to “Sign In” page 2. Attempt to sign in with invalid credentials |
| **Variations:** | None |
| **Resolved:** | **Yes** |
| **Screenshot:** | None |

|  |  |
| --- | --- |
| **Bug:** | Provinces in My Profile not properly aligned |
| **Reproducibility:** | 100% |
| **Steps taken:** | 1. Navigate to My Profile tab while signed in |
| **Variations:** | None |
| **Resolved:** | **Yes** |
| **Screenshot:** | None |

|  |  |
| --- | --- |
| **Bug:** | [Android] In “My Listings”, Add Listing button is incorrectly on top right corner instead of top left |
| **Reproducibility:** | 100% |
| **Steps taken:** | 1. Select “Register Here” on “My Profile” tab 2. Select on any input field |
| **Variations:** | None |
| **Resolved:** | **Yes** |
| **Screenshot:** | None |

|  |  |
| --- | --- |
| **Bug:** | [Android] While registering new user, input fields scroll out of view while typing |
| **Reproducibility:** | 100% |
| **Steps taken:** | 1. Select “Register Here” on “My Profile” tab 2. Select on any input field |
| **Variations:** | None |
| **Resolved:** | **No** |
| **Screenshot:** | None |

|  |  |
| --- | --- |
| **Bug:** | Slide bars change size when switching from numbers to “No max” or “No min” |
| **Reproducibility:** | 100% |
| **Steps taken:** | 1. Navigate to filter screen and move slide bars to boundaries |
| **Variations:** | None |
| **Resolved:** | **No** |
| **Screenshot:** | None |

**Bugs found during ID4:**

|  |  |
| --- | --- |
| **Bug:** | Browse Page doesn't repopulate listings after disliking/liking |
| **Reproducibility:** | 100% |
| **Steps taken:** | 1. Like/Dislike listings on the Browse page (Listings must not already be in Favourites list of user) |
| **Variations:** | None |
| **Resolved:** | **Yes** |
| **Screenshot:** | None |

|  |  |
| --- | --- |
| **Bug:** | Immediately after registering a new user, pressing the My Profile button will redirect to Sign in page |
| **Reproducibility:** | 100% |
| **Steps taken:** | 1. Register a new user 2. Press the My Profile tab |
| **Variations:** | None |
| **Resolved:** | **Yes** |
| **Screenshot:** | None |

|  |  |
| --- | --- |
| **Bug:** | Browse Page: Liking/Disliking should reset the listing to the default position when an error message is given |
| **Reproducibility:** | 100% |
| **Steps taken:** | 1. Swipe left or right on any listing in the Browse page while not signed in. |
| **Variations:** | Attempt to like/dislike own listing while signed in (Should not be allowable in the first place, but still accessible from My Listings page) |
| **Resolved:** | **Yes** |
| **Screenshot:** | None |

|  |  |
| --- | --- |
| **Bug:** | Browse page should not be displaying listings already liked/disliked by the user. A refresh is required. |
| **Reproducibility:** | 100% |
| **Steps taken:** | 1. Sign in, and like/dislike listings on a user. 2. "Sign out" (refresh the page). Sign back in to the same user. Browse page will contain listings already disliked/liked. |
| **Variations:** | None |
| **Resolved:** | **Yes** |
| **Screenshot:** | None |

|  |  |
| --- | --- |
| **Bug:** | Browse page should not be displaying listings already liked/disliked by the user. A refresh is required. |
| **Reproducibility:** | 100% |
| **Steps taken:** | 1. Sign in, and like/dislike listings on a user. 2. "Sign out" (refresh the page). 3. Sign back in to the same user. 4. Browse page will contain listings already disliked/liked. |
| **Variations:** | None |
| **Resolved:** | **Yes** |
| **Screenshot:** | None |

|  |  |
| --- | --- |
| **Bug:** | Cannot remove listings from Favourites page |
| **Reproducibility:** | 100% |
| **Steps taken:** | 1. Attempt to remove any listing from favourite’s page. 2. Leave the favourites page. 3. Re-navigate to the favourite’s page. 4. The listing will re-appear. |
| **Variations:** | None |
| **Resolved:** | **Yes** |
| **Screenshot:** | None |

|  |  |
| --- | --- |
| **Bug:** | Liking/Disliking a listing by swiping on Browse page breaks cycling of listings on Listing details page |
| **Reproducibility:** | 100% |
| **Steps taken:** | 1. Swipe a listing on the Browse page in any direction. 2. Move to details page of any listing. 3. Attempt to cycle listings using arrow buttons. |
| **Variations:** | None |
| **Resolved:** | **Yes** |
| **Screenshot:** | None |

|  |  |
| --- | --- |
| **Bug:** | Disliking a listing results to the listing being added to the user's favourites list |
| **Reproducibility:** | 100% |
| **Steps taken:** | 1. Dislike any listing. 2. Navigate to Favourite’s list to find it there. |
| **Variations:** | None |
| **Resolved:** | **Yes** |
| **Screenshot:** | None |

|  |  |
| --- | --- |
| **Bug:** | Province abbreviation is not being displayed on Listing details page |
| **Reproducibility:** | 100% |
| **Steps taken:** | 1. Sign in to any account with a listing (or create one). 2. Navigate to the listing details page. 3. Abbreviation info is missing. |
| **Variations:** | None |
| **Resolved:** | **Yes** |
| **Screenshot:** | None |

|  |  |
| --- | --- |
| **Bug:** | Bathroom filter stops functionality of other filters |
| **Reproducibility:** | 100% |
| **Steps taken:** | 1. Set bathroom filter min and max to anything. 2. Apply the filter. |
| **Variations:** | 1. Before using the bathroom filter, all other filter functionality works. 2. After using the bathroom filter, other filter functionality stops working. 3. After applying the bathroom filter, any previous filter that had been used stays used until the bathroom filter is reset. |
| **Resolved:** | **Yes** |
| **Screenshot:** | None |

|  |  |
| --- | --- |
| **Bug:** | Add Listing page: Cannot publish listings |
| **Reproducibility:** | 100% |
| **Steps taken:** | 1. Attempt to publish a listing |
| **Variations:** | None |
| **Resolved:** | **Yes** |
| **Screenshot:** | None |

|  |  |
| --- | --- |
| **Bug:** | Square feet information is not being saved when saving a listing without publishing |
| **Reproducibility:** | 100% |
| **Steps taken:** | 1. Save a listing without publishing, including the square feet information. 2. Navigate to the detailed info page of the listing. Square feet is always 0. |
| **Variations:** | None |
| **Resolved:** | **No** |
| **Screenshot:** | None |

|  |  |
| --- | --- |
| **Bug:** | Liking/Disliking a listing from the detailed listing page that is already in your favourites list gives a runtime error |
| **Reproducibility:** | 100% |
| **Steps taken:** | 1. Like/Dislike any listing that is already in your favourites list. |
| **Variations:** | None |
| **Resolved:** | **No** |
| **Screenshot:** | None |

|  |  |
| --- | --- |
| **Bug:** | Save and sign-out buttons on My Profile page are not functioning |
| **Reproducibility:** | 100% |
| **Steps taken:** | 1. Attempt to save any changes made to user info while logged in, attempt to sign out the user |
| **Variations:** | None |
| **Resolved:** | **No** |
| **Screenshot:** | None |

|  |  |
| --- | --- |
| **Bug:** | No warning is given when attempting to publish a listing without an image |
| **Reproducibility:** | 100% |
| **Steps taken:** | 1. Attempt to publish a listing without an image |
| **Variations:** | None |
| **Resolved:** | **No** |
| **Screenshot:** | None |

|  |  |
| --- | --- |
| **Bug:** | Cannot delete published, and saved unpublished listings by user |
| **Reproducibility:** | 100% |
| **Steps taken:** | 1. Create a new listing 2. Attempt to delete the listing made by the user |
| **Variations:** | None |
| **Resolved:** | **No** |
| **Screenshot:** | None |

|  |  |
| --- | --- |
| **Bug:** | Cannot publish a saved unpublished listing |
| **Reproducibility:** | 100% |
| **Steps taken:** | 1. Create a new listing, save but don't publish it. 2. Edit the listing, and attempt to publish it |
| **Variations:** | None |
| **Resolved:** | **No** |
| **Screenshot:** | None |

|  |  |
| --- | --- |
| **Bug:** | Cannot save edited changes to a saved unpublished listing |
| **Reproducibility:** | 100% |
| **Steps taken:** | 1. Edit an unpublished listing 2. Change any information about the listing 3. Attempt to save without publishing |
| **Variations:** | None |
| **Resolved:** | **No** |
| **Screenshot:** | None |

## State Transition Diagrams

4.4.1 Filter, Browse and Favourites screen diagram. Google Maps has been cut from implementation by the development team due to time constraints. Deletion confirmation will be added during ID5.

Diag1Rough

4.4.2 Sign-In, Sign-Up, My Profile, My Listings screen diagram. Sign-out and Save user information changes are currently bugged. Front-end of Facebook log-in has not been connected with back-end implementation, but may be feasible for ID5. Confirmation of deletion of own listings will be added in ID5.

Diag2Rough

## 4.5 Critical Use Cases

The following critical use cases outline the low-level details of the most important features of our application.

|  |  |
| --- | --- |
| **Use Case Component** | **Description** |
| Use Case Name | ID4: Attempt to register a new user |
| Primary Actor | New User |
| Pre-conditions | At entry point of app, on main browse screen |
| Success End Conditions | Either the new user is added to the database or it is identified that the e-mail is already in the database |
| Failed End Conditions | User is not registered |
| Main Success Scenario  A: Actor  S: System | 1. A: Selects “My Profile” tab 2. A: Selects “Register Here” button 3. A: Inputs “testEmail@email.ca” in email field 4. A: Inputs “Password123” in password field 5. A: Inputs “Password” in confirm password field 6. A: Selects “Next” button 7. S: Moves to next info screen 8. A: Inputs “John” in first name field 9. A: Selects “Smith” in last name field 10. A: Selects “3065551234” in phone number field 11. A: Selects “Next” button 12. S: Moves to next info screen 13. A: Selects "Province" drop down list and selects "Alberta" 14. A: Selects "OK" button 15. A: Inputs “Edmonton” into city field 16. A: Clicks "finish" button 17. S: Adds the new user to database if it is not already |
| Extensions | III. A: Inputs “wrongFormatEmail” into email field. Should fail.  IV. A: Inputs "weakpassword" into password field. Should fail.  IV. A: Inputs “NotMatchingPassword123” into password field  A: Inputs “DifferentPassword123” into confirm password field. Should fail.  X. A: Inputs “306555” into phoneNumber field. Should fail. |
| Priority | Critical |
| Response Time | < 1 second |
| Notes | The user is will only be registered if the specific test email has not already been used. If it has been used, it will be tested in the next part of the tests to make sure it can be logged into. |

|  |  |
| --- | --- |
| **Use Case Component** | **Description** |
| Use Case Name | ID4: Filter Listings |
| Primary Actor | Browser (Can be signed in, or not signed in) |
| Pre-conditions | On “Browse” page |
| Success End Conditions | Listings on “Browse” page are filtered to specifications provided |
| Failed End Conditions | Listings on “Browse” page are not filtered to specifications provided |
| Main Success Scenario  A: Actor  S: System | 1. A: Selects “Filter” button 2. A: Selects “Provinces” drop down 3. A: Selects “Yukon” and selects the “OK” button 4. A: Drags the “Price” slider between “100 000 and 1 300 000” 5. A: Drags the “Square feet” slider between “100 and 10 000” 6. A: Drags the “Bedrooms” slider between “2 and 8” 7. A: Drags the “Bathrooms” slider between “3 and 7” 8. A: Selects the “Apply Filter” button 9. S: Displays the listings from the database within the filter criteria |
| Extensions | I. - VIII. A: Selects Cancel Button  I. – VIII. A: Clicks outside of Filter pop up  I. – VIII. A: Apply filters using no specifications |
| Priority | Critical |
| Response Time | < 1 second |
| Notes |  |

|  |  |
| --- | --- |
| **Use Case Component** | **Description** |
| Use Case Name | ID4: Add Trailer Listing in Saskatchewan |
| Primary Actor | Seller |
| Pre-conditions | User is logged in |
| Success End Conditions | Listing is added to “Browse” page |
| Failed End Conditions | Listing is not added to “Browse” page |
| Main Success Scenario  A: Actor  S: System | 1. A: Selects “My Listings” tab 2. A: Selects “+” button to add a listing 3. A: Selects “Province” dropdown list 4. A: Selects “Saskatchewan” province field 5. A: Selects “OK” button 6. A: Inputs “Saskatoon” in city field 7. A: Inputs “123 First Street East“ 8. A: Inputs “fls 9u8“ into Postal Code field 9. A: Inputs “130000“ into price field 10. Inputs “600” in Sq. Feet field 11. A: Inputs “5” in Bed Rooms field 12. A: Inputs “2” in Bath Rooms field 13. A: Inputs “Nice trailer with a little dust. Fixer upper.” 14. A: Adds 1 Image to images 15. A: Selects “Publish” 16. S: Attempts to Adds the listing to database |
| Extensions |  |
| Priority | Critical |
| Response Time | < 1 second |
| Notes | Disallow < 0 in Price, Bedroom, Bathroom, Sq. Ft fields.  Uploading images can only be done on mobile device. |

|  |  |
| --- | --- |
| **Regression Test** | **Description** |
| Regression Test Name | ID4: Regression test: Functionality while not signed in: Using Browse and Filter functionality allows a non-logged-in user to use “My Profile”, “My Listings”, or “Favourites pages”. |
| Primary Actor | Browser |
| Pre-conditions | User is not signed in, on the Browse page |
| Success End Conditions | The user is unable to access sign-in required functionality |
| Failed End Conditions | The user has access to sign-in only functionality |
| Main Success Scenario  A: Actor  S: System | 1. A: Accesses a Listing on the “Browse” page, going to the detailed listing page 2. A: Moves to “My Listings” 3. S: The page notifies the user that signing-in is required to access the “My Listings” functionality |
| Extensions | Upon failure, the user has full access to sign-in only functionality. |
| Priority | Critical |
| Response Time | < 1 second |
| Notes |  |

|  |  |
| --- | --- |
| **Regression Test** | **Description** |
| Regression Test Name | ID4: Regression test: Functionality while not signed in: “Confirm Password” field on register page allows incorrect matches. |
| Primary Actor | Browser |
| Pre-conditions | User is not signed in |
| Success End Conditions | The user is not allowed to get past the registration page without first specifying a valid password match. |
| Failed End Conditions | The user is able to continue registration |
| Main Success Scenario  A: Actor  S: System | 1. A: Selects “My Profile” page 2. A: Selects “Register” button 3. A: Enters “[test@test.com](mailto:test@test.com)” into email field 4. A: Enters “Password123” into password field 5. A: Enters “Password1234” into confirm password field 6. S: Prompts the user that the password and confirm password fields do not match |
| Extensions | Upon failure, the user is capable of registering a user that has incorrectly confirmed their password. |
| Priority | Critical |
| Response Time | < 1 second |
| Notes |  |

|  |  |
| --- | --- |
| **Regression Test** | **Description** |
| Regression Test Name | ID4: Regression test: Functionality while not signed in: Multiple “My Profile” tab clicks removes all UI from the page |
| Primary Actor | Browser |
| Pre-conditions | User is not signed in |
| Success End Conditions | The “My Profile” page still has functionality, is not white/black screened, and DOM elements are not missing. |
| Failed End Conditions | The user sees a white/black screen on the “My Profile” page |
| Main Success Scenario  A: Actor  S: System | 1. A: Repeatedly presses the “My Profile” global tab 2. S: The system does nothing. |
| Extensions | Upon failure, the “My Profile” page turns white/black, losing all functionality. The user is still able to traverse to other tabs, which retain their functionality. No error is given to the user. The only way to revert the problem is to refresh the page. |
| Priority | Critical |
| Response Time | < 1 second |
| Notes |  |

|  |  |
| --- | --- |
| **Regression Test** | **Description** |
| Regression Test Name | ID4: Regression test: Functionality while signing in: Test if a prompt is given after an incorrect login |
| Primary Actor | Browser |
| Pre-conditions | User is not signed in, on the Sign-In page |
| Success End Conditions | The user is given a prompt that the information provided is invalid for logging in |
| Failed End Conditions | The prompt does not display |
| Main Success Scenario  A: Actor  S: System | 1. A: Inputs “[test1@test.com](mailto:test1@test.com)” into email field 2. A: Inputs “WrongPassword” into password field 3. S: Displays prompt to user, explaining invalid log in failure |
| Extensions | Upon failure, the prompt is not displayed. |
| Priority | Critical |
| Response Time | < 1 second |
| Notes |  |

|  |  |
| --- | --- |
| **Regression Test** | **Description** |
| Regression Test Name | ID4: Regression test: Functionality while signed in: Test if multiple “My Profile” tab clicks while signed in returns the user to the Sign-In screen |
| Primary Actor | Browser |
| Pre-conditions | User is signed in, on the “My Profile” page |
| Success End Conditions | The user stays on the “My Profile” page |
| Failed End Conditions | The user is redirected to the “Sign-in” page |
| Main Success Scenario  A: Actor  S: System | 1. A: Repeatedly presses the “My Profile” global tab 2. S: The system does nothing |
| Extensions | Upon failure, the user is redirect to the Sign-In page, but still has access to sign-in only functionality. |
| Priority | Critical |
| Response Time | < 1 second |
| Notes |  |

# 5.0 Coding Style Guide

The development team has put together a set of guidelines to serve our purposes. These guidelines highlight the salient features of the coding style to be followed by developers. Useful examples are provided for quick referencing.

For front-end development using Ionic, the coding guide and sample can be found at the following wiki page:

<https://github.com/CMPT371Team1/Project/wiki/Coding-Style-Example-(JavaScript)>

For back-end development using Python, the coding guide and sample can be found at the following wiki page:

<https://github.com/CMPT371Team1/Project/wiki/Coding-Style-Guide-(Python)>

# 6.0 Build Report

Smoke Test Status:

Front-end tests are implemented using protractor and are currently passing. Back-end smoke tests are also implemented and passing.

Build Status:

The builds for iOS and Android are undergoing light smoke tests, ensuring compilation. We are experiencing difficulty with setting up the build for these platforms, and are focusing on other priorities before the code freeze. Both the Linux build and the remote server are undergoing smoke tests. The server is being updated from our repository after successful builds. Integration of the test server was implemented in ID4, which involves refreshing the test server before running the end-to-end tests, as well as updating after successful builds. End-to-end smoke tests are being conducted using Protractor on the Firefox browser. The back-end is being tested with Python scripts. Thus far, all testing has been done on the browser version of our app; smoke tests are currently not in place for other platforms. Upon successful completion of the smoke tests, the build should push the source code to a branch dedicated to more rigorous forms of manual testing.

SDKs, Packages, and Tools:

All SDKs, packages, and tools employed in our build, as well as their version number, are subject to change. These frameworks are still in question due to lack of experience. These decisions will be made final once the build manager has a firm understanding of automated testing, deployment, server builds, and system builds.

Current list of SDK’s, Packages, and Tools:

* Cordova CLI: 6.5.0
* Ionic Framework Version: 2.0.0-rc.5
* Ionic CLI Version: 2.2.1
* Ionic App Lib Version: 2.2.0
* Ionic App Scripts Version: 1.0.0
* npm: 3.10.10
* jdk: 1.8.0\_121
* nvm: 0.32.0
* node: 6.9.4
* packages listed in package.json
* plus ~400 other Ionic dependency packages
* Android:
* SDK Platform Android 7.1.1, API 25, revision 3
* Android SDK Tools, revision 25.2.5
* Android SDK Build-tools, revision 25.0.1
* Android SDK Platform-tools, revision 25.0.3
* Google Repository, revision 42
* Android Support Repository, revision 42
* iOS:
  + OS: OS X El Capitan
  + Xcode version: Xcode 7.3.1 Build version 7D1014Server:
* Server:
  + Python 2.7
  + Google Cloud sdk v143.0.1
  + Python Extension for google cloud v1.9.50
  + Python Extension (Extra Libs) v1.9.49
  + Gulp v3.9.1
  + OpenSSH v6.6.1p1
  + Sshpass v1.05
* End-to-End Tests
  + Protractor v5.0.0
  + Firefox v47.0.1
  + Selenium v3.1
  + Jasmine v2.5.3

Corodova 6.5.0 requires both jdk 1.8 (or higher), as well as npm v2.2.1 and node v4.0.0. We will be using the most recent version of node and npm to reduce version conflicts. Google Cloud is required for the Google App engine; the platform our servers are built upon. The server uses Python 2.7, the most recent version of Google Cloud SDK, and Python extensions for Google Cloud SDK. Our end-to-end tests are driven by Protractor, which sends to Ionic through the Selenium server and is driven by Jasmine. We are currently testing our system on Firefox v47.0.1, because of compatibility issues with newer versions. We are using Gulp to create a config file which sets the database of our app to the test database, instead of the actual one, if the app is undergoing tests. This is to prevent filling the actual database with test data. Sshpass is used to provide a password and ssh on the command line, allowing us to automatically login to the database server and update the files. All developers and testers have been set up with the latest versions of the required tools. We are using Xcode 7.3.1 for the time being. The Ionic dependencies are extensive, and can be viewed in further detail at the following link:

[https://ionicframework.com/docs/](https://ionicframework.com/docs/%20)

Releases for our build can be found here:

<https://github.com/CMPT371Team1/Project/releases>

# 7.0 Upcoming Requirements

## 7.1 ID5 Requirements (Quality Assurance)

* Replace descriptive text with icons
* Get a legitimate Kasper icon
* Thorough review of test cases for all back-end code

## 7.2 Incomplete Requirements

This section outlines unfinished requirements. These requirements will not be implemented this term and will be documented and passed on to future programmers in Programmer Documentation.

* Contact Seller (front-end only)
* Sign in / Sign up using Facebook (front-end only)
* Get messages/edit messages/contact seller (front-end only)
* Push notifications if something changes regarding a Listing saved in Favourites (a change in the database triggers a notification to the User’s who have favourited that Listing)
* Integration with Google Maps (future)
* Add Google Maps section to Detail page (removed – future)
* Forgot password (future)
* Implement the message system in the front-end and integrate it with the backend (removed – future)
* Finalize method of ordering images
* Send a message to the seller including contact information
* Users can sign up to receive “hot list” notifications
  + Feature Listing (paid for by Sellers)
  + Newly added Listings
  + Price changes on Favourites
  + Based on previous search history
* Sellers receive notifications regarding personal Listings:
  + When a Listing is saved to Favourites
  + When someone requests a viewing
  + When someone sets a price watch
* Super Admin User
  + Log in as Super Admin
  + Add new Listing under any user
  + Edit any Listing
  + Remove any Listing

# 8.0 Triage Meeting ID4

**Date**: Friday, March 17th /17

**Start Time**: 1:30pm

**End Time**: 2:25pm

**Location**: Spinks320

**Members Present**: Tushita Patel (Project Lead), Kristof Mercier (Dev lead), Jeremy Liau (Test Lead), Chris Mykota-Reid (Build Manager)

**Development:**

All Trello tasks have been implemented, except for:

1. Google maps integration with front-end and back-end (but is setup for the database)
2. Buyer to Seller messaging on front-end (this API is implemented in the backend)
3. “Forgot Password” feature in both front and back-end

Due to the ID5 code freeze, the team has decided to document and abandon these requirements in favour of producing better quality code for the finished areas. The above requirements will not be delivered to the client.

**Testing:**

1. All ID4 tests will be performed (manual, end-to-end, etc.)
2. The test team has finalized plans for acceptance and hallway testing

**Bugs / Defects:**High priority bugs to be fixed by the end of ID4 (Sunday, March 19th, /17)

Medium priority bugs to be fixed by the Bug Party (Wednesday, March 22nd, /17)

Low priority bugs will be fixed by the end of ID5

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **STI** | **Decision** | **Severity** | **Priority** | **Fixed as of 03/18?** |
| 1. | Bathroom filter stops functionality of the other filters | This STI is probably appearing due to lack of back-end testing. It could be tricky to fix. Fixing this bug should be postponed until ID5. | Low | Low | ✔ |
| 2. | Liking and Disliking while not signed in returns a runtime error on details page | Although fixing this bug is important, we estimate that fixing it will be too time consuming to prioritize during ID4. | High | Low | ✔ |
| 3. | When a user is not signed in and they attempt to swipe right or left on a Listing, the Listing turns red or green | This is a front-end bug and could be easily fixed. It is a frequently occurring bug that should be prioritized. | High | High | ✔ |
| 4. | Save button on My Profile page is not working while signed-in. | Although fixing this bug is of high importance, this STI could be postponed to early next week because there are other, higher priority bugs. | High | Medium | ✔ |
| 5. | Sign-out button on My Profile page is not working while signed-in. | Although fixing this bug is of high importance, this STI could be postponed to early next week because there are other, higher priority bugs. | High | Medium |  |
| 6. | Sign-in using Facebook is not implemented on the front-end. | This issue might not be too tricky to fix, but is important to the client. We should consider this bug of medium priority. | Low | Medium |  |
| 7. | Browse page does not add more Listings to the cycle when you Like/Dislike a Listing while signed in | This is an important issue because it is a major feature. It should be fixed as soon as possible. | High | High | ✔ |
| 8. | No refresh after every sign in. | Might be an easy fix. Not an important issue, comparatively. | Low | Low | ✔ |
| 9. | Disliking on the Browse Page still moves the Listing to Favourites. | Might be an easy fix. Although severe, it is of lower priority, comparatively. | High | Medium | ✔ |
| 10. | Add Listing is unable to publish Listings. | Might be the same problem as postal code. Quite severe. Should be fixed as soon as possible. | High | High | ✔ |
| 11. | There should be a “required image” pop-up when attempting to publish a Listing without at least one image. Currently not displayed. | Might be an easy fix. Not too severe. | Low | Low | ✔ |
| 12. | My Listings and My Favourites page should have an Edit button on the top right of the app | Swiping left to get Delete/Edit options for a listing is not intuitive. This is a potential design change. Not a priority right now. | Low | Low |  |
| 13. | “Missing Postal code” even when it’s been added. | Fixing of this STI depends on 9. This may be the same bug as 9, or a bug resulting from 9. This bug will be tracked after 9 is fixed. | Low | Unknown | ✔ |
| 14. | Cannot remove Listings from Favourites list. Reappears. | Might be the same bug as 8, or a bug being caused by 8. | High | Medium | ✔ |
| 15. | When registering a user on android, the text fields scroll all the way up. | This bug was identified at the end of ID and has not yet been fixed. It should be prioritized soon. | High | Medium |  |

**Build:**

1. Troubleshooting issues and regressions
2. Added test servers to implementation
3. Reset the test servers before running end-to-end smoke tests on the front-end
4. Shared build knowledge with Gaurav during pair programming sessions
5. Chrome smoke tests removed due to increasing complexity and bugs
6. Added installation of Jasmine Spec Reporter to the build
7. Added if-statements to the build for less manual modifications on release
8. Still looking into automated Style Checking and automated GitHub release