# 7. System Validation

This chapter coves our system validation, implicating all the testing process of the system. This chapter is divided into two sections. Section 7.1 Subsystem Tests, which covers all the testing done for each subsystem in specific preceding the integration process. Section 7.2 System Tests, covering all the system tests directed after integrating the modules of the system.

## 7.1 Subsystem Tests

|  |  |
| --- | --- |
| Subsystems groups | Test Description |
| Frontend UI (Client Subsystem) and backend Web Services | This tests that all the request generated on the frontend are interpreted in the backend webservice returning the result expected. This will only validate that the string to perform the request is correct and that the return is the json object projected. |
| Backend Web Services and backend Favorite Alarms | This tests that a favorite alarm can be added, removed, or set defined time to trigger through the Web Service, and that favorite alarm triggers a message. This Email message is sent to the user that registered that favorite alarm. |
| Backend Web Services and backend Email Service | This tests checks that Emails generated are being received by the correct recipients with the defined template and attachments if so. |

The system is comprised of four subsystems. Among these four subsystems there were three groups that needed to have subsystem testing done on them. These three groups are as follows, the frontend UI (client subsystem) and the backend Web services, the backend Web services and the backend Favorite Alarms, the backend Web services and backend Email service. We will now identify the subsystem testing that took place for each of these groups.

This project had very little subsystem testing that required to be completed. During the planning stage of designing our application we did it in a way that the exchange of information between subsystems were as minimal as possible. We decided to do this so that we would have very little connection between subsystems. In that way our system will become more consistent.

For the first subsystem group, we created the Web services taking to account that few information will be passed as parameters by the client subsystem to each of them and sent in the form of Post request. JSON format was returned with minimum information possible. Continues testing was accomplished using browser Rest tools, given the fact that the frontend and backend interaction is only through web services requests. We conducted manual testing of the interaction of this method between the two subsystems

For the second subsystem group, we used the Web Service to make possible that a favorite alarm can be added, removed, or set defined time to trigger through the Web Service, and that favorite alarm triggers a message in the form of Post request. JSON format was returned with minimum information possible stating the status of that favorite alarm. To check this first part the testing was accomplished using browser Rest tools like in the first subsystem group testing. The second part was checked that the Email message is sent to the user that registered that favorite alarm during the time defined.

For the third subsystem group, we created the Email services using Gmail as SMTP client so our testing focused in check that Emails generated were received by the correct recipients with the defined template and attachments if the message was generated with them.

## 7.2 System Tests

|  |  |
| --- | --- |
| Test Case ID: | PPM-001-1 Display Hours of Operation |
| Purpose: | To validate that they system displays the hours of operation specified by the client |
| Preconditions: | The user has downloaded the app |
| Inputs | 1. User clicks on ‘More ‘ button 2. User presses the ‘Hours of Operation’ button |
| Expected Output: | The system retrieves the information and displays the hours of operation to the user |
| Actual Output: | Same as expected |

|  |  |
| --- | --- |
| Test Case ID: | PPM-001-2 Display Hours of Operation |
| Purpose: | To validate that they system displays the hours of operation specified by the client |
| Preconditions: | The user has downloaded the app |
| Inputs | 1. User clicks on ‘More ‘ button 2. User select another button 3. User clicks back to ‘More’ menu 4. User presses the ‘Hours of Operation’ button |
| Expected Output: | The system retrieves the information and displays the hours of operation to the user |
| Actual Output: | Same as expected |

|  |  |
| --- | --- |
| Test Case ID: | PPM-001-3 Display Hours of Operation |
| Purpose: | To validate that they system does not display hours of operation if database is not populated |
| Preconditions: | The user has downloaded the app |
| Inputs | 1. User clicks on ‘More ‘ button 2. User select another button 3. User clicks back to ‘More’ menu 4. User presses the ‘Hours of Operation’ button |
| Expected Output: | The system does not show any information because no data is stored in database |
| Actual Output: | Hours of operation are still be shown |

|  |  |
| --- | --- |
| Test Case ID: | PPM-002-1 Trolley Alerts |
| Purpose: | To validate that they system allows the user to see any alerts pertaining to the trolleys |
| Preconditions: | * The user has downloaded the app * The admin has typed in one alert |
| Inputs | 1. User clicks on ‘Trolley Alerts ‘ button |
| Expected Output: | The system retrieves the information from the database and now displays information to the user |
| Actual Output: | Same as expected |

|  |  |
| --- | --- |
| Test Case ID: | PPM-002-2 Trolley Alerts |
| Purpose: | To validate that they system allows the user to see any alerts pertaining to the trolleys |
| Preconditions: | * The user has downloaded the app * The admin has typed in multiple alerts |
| Inputs | 1. User clicks on ‘Trolley Alerts ‘ button |
| Expected Output: | The system retrieves the information from the database and now displays information to the user |
| Actual Output: | Same as expected |

|  |  |
| --- | --- |
| Test Case ID: | PPM-002-3 Trolley Alerts |
| Purpose: | To validate that the system does not display any information if admin has removed data |
| Preconditions: | * The user has downloaded the app * The admin has typed in multiple alerts |
| Inputs | 1. User clicks on ‘Trolley Alerts ‘ button |
| Expected Output: | They system displays blank page and an alert message says there are no alerts |
| Actual Output: | Same as expected |

|  |  |
| --- | --- |
| Test Case ID: | PPM-003-1 Show Estimated Time to Favorite Stop |
| Purpose: | To validate that they system calculates the estimated time from trolley location to set favorite stop |
| Preconditions: | * The user has selected the favorite stop * The trolley is being operated |
| Inputs | 1. User opens up app 2. User clicks on ‘Favorites’ tab |
| Expected Output: | The system retrieves the location of the favorite stop from the database, gets the location of the trolley from the TSO API, calculates the time and displays to user |
| Actual Output: | Same as expected |

|  |  |
| --- | --- |
| Test Case ID: | PPM-003-2 Show Estimated Time to Favorite Stop |
| Purpose: | To validate that they system shows message if trolley has passed favorite stop on the route |
| Preconditions: | * The user has selected the favorite stop * The trolley is being operated |
| Inputs | 1. User opens up app 2. User clicks on ‘Favorites’ tab |
| Expected Output: | The system retrieves the location of the favorite stop from the database, gets the location of the trolley from the TSO API, calculates the time and displays to use a ‘N/A’ message |
| Actual Output: | Same as expected |

|  |  |
| --- | --- |
| Test Case ID: | PPM-003-3 Show Estimated Time to Favorite Stop |
| Purpose: | To validate that the system does not calculate time if trolley is not on the route for that favorite stop |
| Preconditions: | * The user has selected the favorite stop * The trolley is being operated |
| Inputs | 1. User opens up app 2. User clicks on ‘Favorites’ tab |
| Expected Output: | The system acknowledges that stop is not on current route and displays a notification message |
| Actual Output: | The favorite stops continue to show estimated time even when the trolley is not on the current route |

|  |  |
| --- | --- |
| Test Case ID: | PPM-004-1 Set Favorite Stop |
| Purpose: | To validate that they system allows the user to save a favorite stop |
| Preconditions: | * The user has signed in to the app |
| Inputs | 1. User opens up app 2. User clicks on ‘Routes’ tab 3. User selects a route 4. User selects a stop 5. User clicks on save stop |
| Expected Output: | The system retrieves the user’s token and stop information. The system then saves the information in the database |
| Actual Output: | Same as expected |

|  |  |
| --- | --- |
| Test Case ID: | PPM-004-2 Set Favorite Stop |
| Purpose: | To validate that they system allows the user to save multiple favorite stops |
| Preconditions: | * The user has signed in to the app * The user already has a stop saved in the database |
| Inputs | 1. User opens up app 2. User clicks on ‘Routes’ tab 3. User selects a route 4. User selects a stop 5. User clicks on save stop |
| Expected Output: | The system retrieves the user’s token and stop information. The system then saves the information in the database for the second favorite stop |
| Actual Output: | Same as expected |

|  |  |
| --- | --- |
| Test Case ID: | PPM-004-3 Set Favorite Stop |
| Purpose: | To validate that they system does not save information if no token is available |
| Preconditions: | * The user is not signed in to the app |
| Inputs | 1. User opens up app 2. User clicks on ‘Routes’ tab 3. User selects a route 4. User selects a stop 5. User clicks on save stop |
| Expected Output: | The system displays a notification if the user is not signed in |
| Actual Output: | System states for function to work user must be signed in |

|  |  |
| --- | --- |
| Test Case ID: | PPM-005-1 Remove Favorite Stop |
| Purpose: | To validate that they system allows the user to remove a favorite stop |
| Preconditions: | * The user has signed in to the app * User has set favorite stop |
| Inputs | 1. User opens up app 2. User clicks on ‘Favorites’ tab 3. User selects a favorite stop 4. User selects Delete button and confirms it |
| Expected Output: | The system retrieves the user’s token and stop information and removes information from the database. System refreshes view |
| Actual Output: | Same as expected |

|  |  |
| --- | --- |
| Test Case ID: | PPM-005-2 Remove Favorite Stop |
| Purpose: | To validate that they system allows the user to remove a favorite stop |
| Preconditions: | * The user has signed in to the app * User has more than one favorite stop |
| Inputs | 1. User opens up app 2. User clicks on ‘Favorites’ tab 3. User selects a favorite stop 4. User selects Delete button and confirms it. Does it twice |
| Expected Output: | The system retrieves the user’s token and stop information and removes information from the database. System refreshes view |
| Actual Output: | Same as expected |

|  |  |
| --- | --- |
| Test Case ID: | PPM-005-3 Remove Favorite Stop |
| Purpose: | To validate that the system does not remove stop if user does not confirm |
| Preconditions: | * The user has signed in to the app * User has saved favorite stop |
| Inputs | 1. User opens up app 2. User clicks on ‘Favorites’ tab 3. User selects a favorite stop 4. User selects Delete button but does not confirm it. |
| Expected Output: | The system acknowledges user did not confirm and keeps information stored |
| Actual Output: | Same as expected |

|  |  |
| --- | --- |
| Test Case ID: | PPM-006-1 List Stops |
| Purpose: | To validate that the system displays all the stops for a given route |
| Preconditions: | * The user has opened the application |
| Inputs | 1. User clicks on ‘Routes’ tab 2. User selects a route |
| Expected Output: | The system retrieves the stops from that given route from the database and displays to user as a list |
| Actual Output: | Same as expected |

|  |  |
| --- | --- |
| Test Case ID: | PPM-006-2 List Stops |
| Purpose: | To validate that the system displays all the stops for a given route |
| Preconditions: | * The user has opened the application |
| Inputs | 1. User clicks on ‘Routes’ tab 2. User selects a route 3. User clicks back and selects another route |
| Expected Output: | The system retrieves the stops from that given route from the database and displays to user as a list |
| Actual Output: | Same as expected |

|  |  |
| --- | --- |
| Test Case ID: | PPM-006-3 List Stops |
| Purpose: | To validate that the system hides the list route when user clicks back button |
| Preconditions: | * The user has opened the application |
| Inputs | 1. User clicks on ‘Routes’ tab 2. User selects a route 3. User clicks back |
| Expected Output: | The system acknowledges view change and hides route lst |
| Actual Output: | Same as expected |

|  |  |
| --- | --- |
| Test Case ID: | PPM-007-1 Map Routes |
| Purpose: | To validate that the system displays all of the routes for the trolley |
| Preconditions: | * The user has opened the application |
| Inputs |  |
| Expected Output: | The system retrieves the waypoints from the database for each route and displays in color coated fashion representing two different routes |
| Actual Output: | Same as expected |

|  |  |
| --- | --- |
| Test Case ID: | PPM-007-2 Map Routes |
| Purpose: | To validate that the system displays all of the routes for the trolley |
| Preconditions: | * The user has opened the application |
| Inputs | 1. User clicks on ‘More’ tab 2. User clicks on ‘Map’ tab |
| Expected Output: | The system retrieves the waypoints from the database for each route and displays in color coated fashion representing two different routes |
| Actual Output: | Same as expected |

|  |  |
| --- | --- |
| Test Case ID: | PPM-007-3 Map Routes |
| Purpose: | To validate that the system does not display route if no information is stored in database |
| Preconditions: | * The user has opened the application |
| Inputs |  |
| Expected Output: | The system only displays route that information is present in database |
| Actual Output: | Same as expected |

|  |  |
| --- | --- |
| Test Case ID: | PPM-008-1 Show Estimated Time to Stop |
| Purpose: | To validate that the system displays estimated time for a clicked stop |
| Preconditions: | * The user has opened the application |
| Inputs | 1. User has clicked on a stop |
| Expected Output: | The system retrieves the stop information and the trolley location. System calculates and displays estimated time |
| Actual Output: | Same as expected |

|  |  |
| --- | --- |
| Test Case ID: | PPM-008-2 Show Estimated Time to Stop |
| Purpose: | To validate that the system displays notification that trolley has passed particular stop |
| Preconditions: | * The user has opened the application |
| Inputs | 1. User has clicked on a stop |
| Expected Output: | The system retrieves the stop information and the trolley location. System calculates and displays a ‘N/A’ for estimated time |
| Actual Output: | Same as expected |

|  |  |
| --- | --- |
| Test Case ID: | PPM-008-3 Show Estimated Time to Stop |
| Purpose: | To validate that the system displays a notification for estimated time for the stop not on route |
| Preconditions: | * The user has opened the application |
| Inputs | 1. User has clicked on a stop |
| Expected Output: | The system retrieves the stop information and the trolley location. System sends message indicating stop is not on current route |
| Actual Output: | System calculates distance from trolley location |

|  |  |
| --- | --- |
| Test Case ID: | PPM-009-1 Find Nearest Stop |
| Purpose: | To validate that the system retrieves the user’s location and returns the nearest stop given the route. |
| Preconditions: | Requires the GPS setting enabled on the user’s device or accept localization when using a desktop web browser |
| Inputs | 1. User clicks on Map view 2. User presses the ‘Near Me’ button 3. User selects route: **Palmetto Middle North Route** |
| Expected Output: | System finds current location of user and returns: 9398 Ludlam Road stop as closest stop |
| Actual Output: | Same as expected |

|  |  |
| --- | --- |
| Test Case ID: | PPM-009–2 Find Nearest Stop |
| Purpose: | To validate that the system retrieves the user’s location and returns the nearest stop given the route. |
| Preconditions: | Requires the GPS setting enabled on the user’s device or accept localization when using a desktop web browser |
| Inputs | 1. User clicks on Map view 2. User presses the ‘Near Me’ button 3. User selects route: **Palmetto Middle North Route** |
| Expected Output: | System finds current location of user and returns: 9800 SW 73rd Ct as closest stop |
| Actual Output: | Same as expected |

|  |  |
| --- | --- |
| Test Case ID: | PPM-009–3 Find Nearest Stop |
| Purpose: | To validate that the system does not show closest stop when the user doesn’t select a route |
| Preconditions: | Requires the GPS setting enabled on the user’s device or accept localization when using a desktop web browser |
| Inputs | 1. User clicks on Map view 2. User presses the ‘Near Me’ button 3. User presses outside the popup menu |
| Expected Output: | System returns to the map view with no new marker indicating closest stop. |
| Actual Output: | Same as expected |

|  |  |
| --- | --- |
| Test Case ID: | PPM-010-1 Arrived to Location Alert |
| Purpose: | To validate that the system sends an email to the user the trolley has arrived to their favorite stop |
| Preconditions: | The user has set a alert time notification for a favorite stop |
| Inputs |  |
| Expected Output: | System retrieves trolley and favorite stop information. System sends the user an email indicating trolley has arrived |
| Actual Output: | Same as expected |

|  |  |
| --- | --- |
| Test Case ID: | PPM-010-2 Arrived to Location Alert |
| Purpose: | To validate that the system sends an email to the user the trolley has arrived to their multiple favorite stop |
| Preconditions: | The user has set a alert time notification for a favorite stop |
| Inputs |  |
| Expected Output: | System retrieves trolley and favorite stop information. System sends the user an email indicating trolley has arrived |
| Actual Output: | Same as expected |

|  |  |
| --- | --- |
| Test Case ID: | PPM-010-3 Arrived to Location Alert |
| Purpose: | To validate that the system does not send an email to the user the trolley has arrived to their favorite stop if not alert was select |
| Preconditions: | The user has saved a favorite stop |
| Inputs |  |
| Expected Output: | System does not send email |
| Actual Output: | System sent the email |

|  |  |
| --- | --- |
| Test Case ID: | PPM-011-1 Login |
| Purpose: | To validate that the system allows the user to login with correct credentials |
| Preconditions: | User has already registered an account |
| Inputs | 1. User clicks on ‘More’ tab 2. User selects ‘Login’ button 3. User types in username: [mauricepruna@gmail.com](mailto:mauricepruna@gmail.com) 4. User types in password: Cuba1234 |
| Expected Output: | System acknowledges user credentials and logins with favorites information |
| Actual Output: | Same as expected |

|  |  |
| --- | --- |
| Test Case ID: | PPM-011-2 Login |
| Purpose: | To validate that the system allows the user to login with correct credentials |
| Preconditions: | User has already registered an account |
| Inputs | 1. User clicks on ‘More’ tab 2. User selects ‘Login’ button 3. User types in username: [rmart071@fiu.edu](mailto:mauricepruna@gmail.com) 4. User types in password: Admin1234 |
| Expected Output: | System acknowledges user credentials and logins with favorites information |
| Actual Output: | Same as expected |

|  |  |
| --- | --- |
| Test Case ID: | PPM-011-3 Login |
| Purpose: | To validate that the system does not allows the user to login with incorrect credentials |
| Preconditions: | User has already registered an account |
| Inputs | 1. User clicks on ‘More’ tab 2. User selects ‘Login’ button 3. User types in username: masoud@fiu.edu 4. User types in password: SeniorProject |
| Expected Output: | System acknowledges user credentials are incorrect, displays message, and does not let user sign in |
| Actual Output: | Same as expected |

|  |  |
| --- | --- |
| Test Case ID: | PPM-012-1 Register User |
| Purpose: | To validate that the system stores user information when user registers with the application |
| Preconditions: | User has not created an account |
| Inputs | 1. User clicks on ‘ More’ tab 2. User clicks on ‘Register’ button 3. User types in user name: gabrielawilson@gmail.com 4. User types in password: lovepinecrest |
| Expected Output: | System stores credentials in database and system logins in user |
| Actual Output: | Same as expected |

|  |  |
| --- | --- |
| Test Case ID: | PPM-012-2 Register User |
| Purpose: | To validate that the system stores user information when user registers with the application |
| Preconditions: | User has not created an account |
| Inputs | 1. User clicks on ‘ More’ tab 2. User clicks on ‘Register’ button 3. User types in user name: pepe@gmail.com 4. User types in password: billete |
| Expected Output: | System stores credentials in database and system logins in user |
| Actual Output: | Same as expected |

|  |  |
| --- | --- |
| Test Case ID: | PPM-012-2 Register User |
| Purpose: | To validate that the system does not let user register if account is already preset |
| Preconditions: |  |
| Inputs | 1. User clicks on ‘ More’ tab 2. User clicks on ‘Register’ button 3. User types in user name: [mauricepruna@gmail.com](mailto:mauricepruna@gmail.com) 4. User types in password: Cuba1234 |
| Expected Output: | System shows notification and does not save information |
| Actual Output: | Same as expected |

|  |  |
| --- | --- |
| Test Case ID: | PPM-013-1 Request a stop |
| Purpose: | To validate that the system allows the user to send an email to request for a new stop |
| Preconditions: | User has opened app |
| Inputs | 1. User clicks on ‘More’ tab 2. User clicks on ‘Request a Stop’ button 3. User Enters name: Maurice 4. User enters email: [mauricepruna@gmail.com](mailto:mauricepruna@gmail.com) 5. User enters location: 1111 SW 111 St 6. User clicks send |
| Expected Output: | Email is sent to system admin with text information |
| Actual Output: | Same as expected |

|  |  |
| --- | --- |
| Test Case ID: | PPM-013-2 Request a stop |
| Purpose: | To validate that the system allows the user to send an email to request for a new stop |
| Preconditions: | User has opened app |
| Inputs | 1. User clicks on ‘More’ tab 2. User clicks on ‘Request a Stop’ button 3. User enters name: Ricky 4. User enters email: [rmart071@fiu.edu](mailto:rmart071@fiu.edu) 5. User enters location: 9400 SW 63 Ct 6. User clicks send |
| Expected Output: | Email is sent to system admin with text information |
| Actual Output: | Same as expected |

|  |  |
| --- | --- |
| Test Case ID: | PPM-013-3 Request a stop |
| Purpose: | To validate that the system recognizes that email input is correct format |
| Preconditions: | User has opened app |
| Inputs | 1. User clicks on ‘More’ tab 2. User clicks on ‘Request a Stop’ button 3. User enters Name: test 4. User enters email: [select \* from users](mailto:rmart071@fiu.edu) 5. User enters location: 1111 SW 111 St 6. User clicks send |
| Expected Output: | System acknowledges email is incorrect format and does not send email |
| Actual Output: |  |

|  |  |
| --- | --- |
| Test Case ID: | PPM-014-1 Report a Problem |
| Purpose: | To validate that the system allows the user to send an email to report a problem |
| Preconditions: | User has opened app |
| Inputs | 1. User clicks on ‘More’ tab 2. User clicks on ‘Report a Problem’ button 3. User enters their email: [mauricepruna@gmail.com](mailto:mauricepruna@gmail.com) 4. User enters problem: tree is in the middle of the road 5. User clicks send |
| Expected Output: | Email is sent with problem in body |
| Actual Output: | Same as expected |

|  |  |
| --- | --- |
| Test Case ID: | PPM-014-2 Report a Problem |
| Purpose: | To validate that the system allows the user to send an email to report a problem |
| Preconditions: | User has opened app |
| Inputs | 1. User clicks on ‘More’ tab 2. User clicks on ‘Report a Problem’ button 3. User enters their email: [mauricepruna@gmail.com](mailto:mauricepruna@gmail.com) 4. User takes picture 5. User enters problem: tree is in the middle of the road 6. User clicks send |
| Expected Output: | Email is sent with problem in body along with photo as attachement |
| Actual Output: | Same as expected |

|  |  |
| --- | --- |
| Test Case ID: | PPM-014-3 Report a Problem |
| Purpose: | To validate that the system does not allow the user to send an email to report a problem if all blank |
| Preconditions: | User has opened app |
| Inputs | 1. User clicks on ‘More’ tab 2. User clicks on ‘Report a Problem’ button 3. User leaves information blank 4. User clicks send |
| Expected Output: | System acknowledges fields are blank and does not send email |
| Actual Output: | Same as expected |

|  |  |
| --- | --- |
| Test Case ID: | PPM-015-1 Send Feedback |
| Purpose: | To validate that the system allows the user to send an email to send feedback |
| Preconditions: | User has opened app |
| Inputs | 1. User clicks on ‘More’ tab 2. User clicks on ‘Send Feedback’ button 3. User types in name: Maurice 4. User types in email: [mauricepruna@gmail.com](mailto:mauricepruna@gmail.com) 5. User types in: This app is the best!!!! 6. User clicks send |
| Expected Output: | System sends email to system admin with text added by the user the body |
| Actual Output: | Same as expected |

|  |  |
| --- | --- |
| Test Case ID: | PPM-015-2 Send Feedback |
| Purpose: | To validate that the system allows the user to send an email to send feedback |
| Preconditions: | User has opened app |
| Inputs | 1. User clicks on ‘More’ tab 2. User clicks on ‘Send Feedback’ button 3. User enters name: Ricky 4. User types in email: rmart071@fiu.edu 5. User types in: This app has a lot of meat! 6. User clicks send |
| Expected Output: | System sends email to system admin with text added by the user the body |
| Actual Output: | Same as expected |

|  |  |
| --- | --- |
| Test Case ID: | PPM-015-3 Send Feedback |
| Purpose: | To validate that the system does not allow the user to send an email to send feedback if incorrect email format |
| Preconditions: | User has opened app |
| Inputs | 1. User clicks on ‘More’ tab 2. User clicks on ‘Enter Feedback’ button 3. User types in username: Ricky 4. User enters email: Ricky 5. User types in: This app has a lot of meat! 6. User clicks send |
| Expected Output: | System acknowledges incorrect format and does not send email |
| Actual Output: | Same as expected |

## 7.3 Evaluation of Tests

### Subsystem Test Results

|  |  |  |
| --- | --- | --- |
| Test Case ID | Description | Result |
|  | Fetch Known Key | PASSED |
| M011 | Fetch Unknown Key | PASSED |
| M012 | ValidCMSUpdate | PASSED |
| M013 | InValidCMSUpdate | PASSED |
| M014 | SendValidEmail | PASSED |
| M015 | DontSendInvalidEmail | PASSED |

### System Testing – Sunny Day

|  |  |
| --- | --- |
| Test Case ID | Test Result |
| PPM-001-1 | PASSED |
| PPM-001-2 | PASSED |
| PPM-002-1 | PASSED |
| PPM-002-2 | PASSED |
| PPM-003-1 | PASSED |
| PPM-003-2 | PASSED |
| PPM-004-1 | PASSED |
| PPM-004-2 | PASSED |
| PPM-005-1 | PASSED |
| PPM-005-2 | PASSED |
| PPM-006-1 | PASSED |
| PPM-006-2 | PASSED |
| PPM-007-1 | PASSED |
| PPM-007-2 | PASSED |
| PPM-008-1 | PASSED |
| PPM-008-2 | PASSED |
| PPM-009-1 | PASSED |
| PPM-009-2 | PASSED |
| PPM-010-1 | PASSED |
| PPM-010-2 | PASSED |
| PPM-011-1 | PASSED |
| PPM-011-2 | PASSED |
| PPM-012-1 | PASSED |
| PPM-012-2 | PASSED |
| PPM-013-1 | PASSED |
| PPM-013-2 | PASSED |
| PPM-014-1 | PASSED |
| PPM-014-2 | PASSED |
| PPM-015-1 | PASSED |
| PPM-015-2 | PASSED |

#### System Testing - Rainy Day

|  |  |
| --- | --- |
| Test Case ID | Test Result |
| PPM-001-3 | FAILED |
| PPM-002-3 | PASSED |
| PPM-003-3 | FAILED |
| PPM-004-3 | PASSED |
| PPM-005-3 | PASSED |
| PPM-006-3 | PASSED |
| PPM-007-3 | PASSED |
| PPM-008-3 | FAILED |
| PPM-009-3 | PASSED |
| PPM-010-3 | PASSED |
| PPM-011-3 | PASSED |
| PPM-012-3 | PASSED |
| PPM-013-3 | PASSED |
| PPM-014-3 | PASSED |
| PPM-015-3 | PASSED |

## 7.3. Evaluation of Tests