



SECP3204: Software Engineering WBL

System Test Description (STD)

Vending Machine Inventory Management System

Version 1

Date: 21 July 2022

School of Computing, Faculty of Engineering

Prepared by: Garrison Tech

Revision Page

a. Overview

The System Test Description (STD) component of the system description has been finished in this documentation. This section will outline each test case that is required as well as the testing approach that we will use to evaluate our system. We have documented the specific steps and anticipated outcomes for each test case. To keep track of the test cases, we have utilized Microsoft Excel. The usage of a general template made everything appear uniform. To conduct the tests, we use equivalence partitioning (EP) and boundary value analysis (BVA), which will be discussed in the section on test approach analysis.

b. Target Audience

The owner of MFA Solutions and the lecturers for UTM's 2u2i Program are the intended audience of this system description. They will make use of this system description to comprehend the process and results of the system testing. They will be able to observe the tests' workflow as well.

c. Project Team Members

| Team Member | Task(s) | Status |
|-----------------------|-------------------------------------|-----------|
| Jayaneysanraj Jayaraj | TC001 Test Case & TC002 Test Cases | Completed |
| Alvin Hee Jun Sheung | TC005 Test Cases | Completed |
| Ragu Raju Naidu | TC003 Test Cases & TC004 Test Cases | Completed |
| Jeggean | TC006 Test Cases | Completed |

d. Version Control History

| Version | Primary Author(s) | Description of Version | Date Completed |
|---------|---|--|----------------|
| 1.0 | Jayaneysanraj Jayaraj, Jeggean A/L Rajendran, Ragu Raju Naidu A/L S Palani Raju Naidu, Alvin Hee Jun Sheung | Completed the proposal and produced Use Case Diagram (PR1) | 19/05/2022 |
| | | Completed the SRS - System Requirement Specification (PR2) | 20/06/2022 |
| | | Completed the SDD - System Design Description (PR3) | 12/07/2022 |
| | | Completed the STD - System Test Description (PR4) | 21/07/2022 |

Table of Contents

| | | | |
|---|-------------------|--|----------|
| 8 | Test Cases | | 3 |
| | 8.1 | TC001: Test <User Management> Subsystem: <Login/Logout (UC001)> | 3 |
| | | 8.1.1 TC001_01: Test <User Login/Logout (SD001)> | 3 |
| | 8.2 | TC002: Test <User Management> Subsystem: <Manage Users (UC002)> | 3 |
| | | 8.2.1 TC002_01: Test <Create User (SD002)> | 3 |
| | | 8.2.2 TC002_02: Test <Update User (SD003)> | 4 |
| | | 8.2.3 TC002_03: Test <Delete User (SD004)> | 4 |
| | 8.3 | TC003: Test <Stock Management> Subsystem: <Manage Stock (UC003)> | 5 |
| | | 8.3.1 TC003_01: Test <Create Stock (SD005)> | 5 |
| | | 8.3.2 TC003_02: Test <View Stock (SD006)> | 5 |
| | | 8.3.3 TC003_03: Test <Update Stock (SD007)> | 6 |
| | | 8.3.4 TC003_04: Test <Delete Stock (SD008)> | 6 |
| | 8.4 | TC004: Test <Stock Management> Subsystem: <Manage Vending Machine (UC004)> | 7 |
| | | 8.4.1 TC004_01: Test <Create Vending Machine (SD009)> | 7 |
| | | 8.4.2 TC004_02: Test <View Vending Machine (SD010)> | 7 |
| | | 8.4.3 TC004_03: Test <Update Vending Machine (SD011)> | 8 |
| | | 8.4.4 TC004_04: Test <Delete Vending Machine (SD012)> | 8 |
| | 8.5 | TC005: Test <Sales Management> Subsystem: <Manage Vending Machine Sales (UC005)> | 9 |
| | | 8.5.1 TC005_01: Test <Create Vending Machine Sales (SD013)> | 9 |
| | | 8.5.2 TC005_02: Test <View Vending Machine Sales (SD014)> | 9 |
| | | 8.5.3 TC005_03: Test <Update Vending Machine Sales (SD015)> | 10 |

| | | | | |
|---|---------------------------------|-------|--|-----------|
| | | 8.5.4 | TC005_04: Test <Delete Vending Machine Sales (SD016)> | 10 |
| | 8.6 | | TC006: Test <Sales Management > Subsystem: <Manage Report (UC006)> | 11 |
| | | 8.6.1 | TC006_01: Test <Create Stock History (SD017)> | 11 |
| | | 8.6.2 | TC006_02: Test <Update Stock History (SD018)> | 11 |
| 9 | Test Approach Analysis | | | 12 |
| | Appendices | | | 23 |
| | Appendix A: Traceability Matrix | | | 23 |

8. Test Cases

8.1 TC001: Test <User Management> Subsystem: <Login/Logout (UC001)>

This test contains the following test cases:

(a) TC001_01: Test <User Login/Logout (SD001)>

8.1.1 TC001_01: Test <User Login/Logout (SD001)>

| | | | | | |
|-----------------|---|--|--|--|------|
| Test Case ID | TC001_01 | Test Case Description | Test the functionality of login/logout | | |
| Created By | Jayaneyasauraj Jayaraj | Reviewed By | Jayaneyasauraj Jayaraj | Version | 1.0 |
| QA Tester's Log | None | | | | |
| Tester's Name | Jegan | Date Tested | 12/07/2022 | Test Case (Pass/Fail/Not) | Pass |
| S# | Prerequisites: | | S# | Test Data | |
| 1 | Access to Internet Browser | | 1 | username= admin@gmail.com | |
| 2 | Account already registered on the system | | 2 | password = 1qaz2wsxQ! | |
| 3 | | | | | |
| 4 | | | | | |
| 5 | | | | | |
| Test Scenario | Verify that the user can login and logout | | | | |
| Step # | Step Details | Expected Results | Actual Results | Pass / Fail / Not Executed / Suspended | |
| 1 | Type in Username and Password | No errors and fields are filled properly | | | |
| 2 | Click on the 'Login' button | Redirects user to landing page based on user type | | | |
| 3 | Verify that the landing page is correct | Landing page is correct | | | |
| 4 | Click on name on the top right corner | A pop up is shown with a profile picture and 'Sign out' button | | | |
| 5 | Click on 'Sign out' button | Redirects user back to login page | | | |
| 6 | | | | | |

8.2 TC002: Test <User Management> Subsystem: <Manage Users (UC002)>

This test contains the following test cases:

(a) TC002_01: Test <Create User (SD002)>

(b) TC002_02: Test <Update User (SD003)>

(c) TC002_03: Test <Delete User (SD004)>

8.2.1 TC002_01: Test <Create User (SD002)>

| | | | | | |
|-----------------|---|---|---------------------------------------|--|------|
| Test Case ID | TC002_01 | Test Case Description | Test the functionality of create user | | |
| Created By | Jayaneyasauraj Jayaraj | Reviewed By | Jayaneyasauraj Jayaraj | Version | 1.0 |
| QA Tester's Log | None | | | | |
| Tester's Name | Ragu | Date Tested | 12/07/2022 | Test Case (Pass/Fail/Not) | Pass |
| S# | Prerequisites: | | S# | Test Data | |
| 1 | Access to Internet Browser | | 1 | username= admin@gmail.com | |
| 2 | User logged in as Employer | | 2 | password = 1qaz2wsxQ! | |
| 3 | | | | | |
| 4 | | | | | |
| 5 | | | | | |
| Test Scenario | Verify that the Employer can create a user | | | | |
| Step # | Step Details | Expected Results | Actual Results | Pass / Fail / Not Executed / Suspended | |
| 1 | Click on 'User Management' from side navbar | Redirects to user management page | | | |
| 2 | Click on the 'New User' button | A pop up form is shown | | | |
| 3 | Fill out new user details | Details are correct and no errors displayed on the form | | | |
| 4 | Click on the 'Register' button | The user is created and displayed in the list of users | | | |
| 5 | | | | | |

8.2.2 TC002_02: Test <Update User (SD003)>

| | | | | | |
|-----------------|--|---|---------------------------------------|--|------|
| Test Case ID | TC002_02 | Test Case Description | Test the functionality of update user | | |
| Created By | Jayaneysauraj Jayaraj | Reviewed By | Jayaneysauraj Jayaraj | Version | 1.0 |
| QA Tester's Log | None | | | | |
| Tester's Name | Ragu | Date Tested | 12/07/2022 | Test Case (Pass/Fail/Not) | Pass |
| S# | Prerequisites: | | S# | Test Data | |
| 1 | Access to Internet Browser | | 1 | username= admin@gmail.com | |
| 2 | User logged in as Employer | | 2 | password = 1qaz2wsxQ! | |
| 3 | | | | | |
| 4 | | | | | |
| 5 | | | | | |
| Test Scenario | Verify that the Employer can update a user's details | | | | |
| Step # | Step Details | Expected Results | Actual Results | Pass / Fail / Not Executed / Suspended | |
| 1 | Click on 'User Management' from side navbar | Redirects to user management page | | | |
| 2 | Click on the 'Edit User' button | A pop up form is shown with existing details | | | |
| 3 | Make necessary changes to user details | Details are correct and no errors displayed on the form | | | |
| 4 | Click on the 'Save' button | The user details are updated | | | |
| 5 | | | | | |

8.2.3 TC002_03: Test <Delete User (SD004)>

| | | | | | | | |
|-----------------|---|-----------------------|---------------------------------------|----|---------------------------|---------------------------|--|
| Test Case ID | TC002_03 | Test Case Description | Test the functionality of delete user | | | | |
| Created By | Jayaneysauraj Jayaraj | Reviewed By | Jayaneysauraj Jayaraj | | | Version | 1.0 |
| QA Tester's Log | None | | | | | | |
| Tester's Name | Ragu | Date Tested | 12/07/2022 | | | Test Case (Pass/Fail/Not) | Pass |
| S# | Prerequisites: | | | S# | Test Data | | |
| 1 | Access to Internet Browser | | | 1 | username= admin@gmail.com | | |
| 2 | User logged in as Employer | | | 2 | password = 1qaz2wsxQ! | | |
| 3 | | | | | | | |
| 4 | | | | | | | |
| 5 | | | | | | | |
| Test Scenario | Verify that the Employer can delete a user | | | | | | |
| Step # | Step Details | | Expected Results | | Actual Results | | Pass / Fail / Not Executed / Suspended |
| 1 | Click on 'User Management' from side navbar | | Redirects to user management page | | | | |
| 2 | Click on the 'Delete User' button | | A pop up form is shown | | | | |
| 3 | Click on 'Confirm' | | User is deleted | | | | |
| 4 | | | | | | | |

8.3 TC003: Test <Stock Management> Subsystem: <Manage Stock (UC003)>

This test contains the following test cases:

- (a) TC003_01: Test <Create Stock (SD005)>
- (b) TC003_02: Test <View Stock (SD006)>
- (c) TC003_03: Test <Update Stock (SD007)>
- (d) TC003_04: Test <Delete Stock (SD008)>

8.3.1 TC003_01: Test <Create Stock (SD005)>

| | | | | | | | | | |
|-----------------|--|-----------------------|---|----|---------------------------|---------------------------|--|--|--|
| Test Case ID | TC003_01 | Test Case Description | Test the functionality of Create Stock | | | | | | |
| Created By | Ragu R Naidu | Reviewed By | Jayaneyanraj Jayaraj | | | Version | 1.0 | | |
| QA Tester's Log | None | | | | | | | | |
| Tester's Name | Jeggean | Date Tested | 04/07/2022 | | | Test Case (Pass/Fail/Not) | Pass | | |
| S# | Prerequisites: | | | S# | Test Data | | | | |
| 1 | Access to Internet Browser | | | 1 | username= admin@gmail.com | | | | |
| 2 | User logged in as Employer | | | 2 | password = 1qaz2wsxQ! | | | | |
| 3 | | | | | | | | | |
| 4 | | | | | | | | | |
| 5 | | | | | | | | | |
| Test Scenario | Verify that the Employer/Employee can create a stock | | | | | | | | |
| Step # | Step Details | | Expected Results | | Actual Results | | Pass / Fail / Not Executed / Suspended | | |
| 1 | Click on 'Inventory Stock' from side navbar | | Redirects to stock management page | | | | | | |
| 2 | Click on the 'New Stock' button | | A pop up form is shown | | | | | | |
| 3 | Fill out new stock details | | Details are correct and no errors displayed on the form | | | | | | |
| 4 | Click on the 'Create' button | | The stock is created and displayed in the Inventory | | | | | | |
| 5 | | | | | | | | | |

8.3.2 TC003_02: Test <View Stock (SD006)>

| | | | | | | | | | |
|-----------------|---|-----------------------|--|---------------------------|----------------|---------------------------|--|--|--|
| Test Case ID | TC003_02 | Test Case Description | Test the functionality of View Stock | | | | | | |
| Created By | Ragu R Naidu | Reviewed By | Jayaneysanraj Jayaraj | | | Version | 1.0 | | |
| QA Tester's Log | None | | | | | | | | |
| Tester's Name | Jeggean | Date Tested | 04/07/2022 | | | Test Case (Pass/Fail/Not) | Pass | | |
| S# | Prerequisites: | | S# | Test Data | | | | | |
| 1 | Access to Internet Browser | | 1 | username= admin@gmail.com | | | | | |
| 2 | User logged in as Employer | | 2 | password = 1qaz2wsxQ! | | | | | |
| 3 | | | | | | | | | |
| 4 | | | | | | | | | |
| 5 | | | | | | | | | |
| Test Scenario | Verify that the Employer/Employee can view existing stock | | | | | | | | |
| Step # | Step Details | | Expected Results | | Actual Results | | Pass / Fail / Not Executed / Suspended | | |
| 1 | Click on 'Inventory Stock' from side navbar | | Redirects to stock management page | | | | | | |
| 2 | View Inventory Page | | Inventory Page Displays List of Stocks | | | | | | |
| 3 | | | | | | | | | |

8.3.3 TC003_03: Test <Update Stock (SD007)>

| | | | | | | | | |
|-----------------|--|---|--|---------------------------|---------------------------|--|------|--|
| Test Case ID | TC003_03 | Test Case Description | Test the functionality of Update Stock | | | | | |
| Created By | Ragu R Naidu | Reviewed By | Jayaneysanraj Jayaraj | | Version | | 1.0 | |
| QA Tester's Log | None | | | | | | | |
| Tester's Name | Jeggean | Date Tested | 04/07/2022 | | Test Case (Pass/Fail/Not) | | Pass | |
| S# | Prerequisites: | | S# | Test Data | | | | |
| 1 | Access to Internet Browser | | 1 | username= admin@gmail.com | | | | |
| 2 | User logged in as Employer | | 2 | password = 1qaz2wsxQ! | | | | |
| 3 | | | | | | | | |
| 4 | | | | | | | | |
| 5 | | | | | | | | |
| Test Scenario | Verify that the Employer/Employee can update a stock | | | | | | | |
| Step # | Step Details | Expected Results | Actual Results | | | Pass / Fail / Not Executed / Suspended | | |
| 1 | Click on 'Inventory Stock' from side navbar | Redirects to stock management page | | | | | | |
| 2 | Click on the 'Edit Stock' button | A pop up form is shown | | | | | | |
| 3 | Make necessary changes to Stock Details | Details are correct and no errors displayed on the form | | | | | | |
| 4 | Click on the 'Save' button | The Stock Details are updated | | | | | | |
| 5 | | | | | | | | |

8.3.4 TC003_04: Test <Delete Stock (SD008)>

| | | | | | | | | | |
|-----------------|--|-----------------------|--|--|----------------|---------------------------|--|------|--|
| Test Case ID | TC003_04 | Test Case Description | Test the functionality of Delete Stock | | | | | | |
| Created By | Ragu R Naidu | Reviewed By | Jayaneysanraj Jayaraj | | | Version | | 1.0 | |
| QA Tester's Log | None | | | | | | | | |
| Tester's Name | Jeggean | Date Tested | 04/07/2022 | | | Test Case (Pass/Fail/Not) | | Pass | |
| S# | Prerequisites: | | | | S# | Test Data | | | |
| 1 | Access to Internet Browser | | | | 1 | username= admin@gmail.com | | | |
| 2 | User logged in as Employer | | | | 2 | password = 1qaz2wsxQ! | | | |
| 3 | | | | | | | | | |
| 4 | | | | | | | | | |
| 5 | | | | | | | | | |
| Test Scenario | Verify that the Employer/Employee can delete a stock | | | | | | | | |
| Step # | Step Details | | Expected Results | | Actual Results | | Pass / Fail / Not Executed / Suspended | | |
| 1 | Click on 'Inventory Stock' from side navbar | | Redirects to stock management page | | | | | | |
| 2 | Click on the 'Delete Stock' button | | A pop up form is shown | | | | | | |
| 3 | Click on 'Delete' | | Stock is deleted | | | | | | |
| 4 | | | | | | | | | |

8.4 TC004: Test <Stock Management> Subsystem: <Manage Vending Machine (UC004)>

This test contains the following test cases:

- (a) TC004_01: Test <Create Vending Machine (SD009)>
- (b) TC004_02: Test <View Vending Machine (SD010)>
- (c) TC004_03: Test <Update Vending Machine (SD011)>
- (d) TC004_04: Test <Delete Vending Machine (SD012)>

8.4.1 TC004_01: Test <Create Vending Machine (SD009)>

| | | | | | | |
|-----------------|---|--|--|---------------------------|--|------|
| Test Case ID | TC004_01 | Test Case Description | Test the functionality of Create Vending Machine | | | |
| Created By | Ragu R Naidu | Reviewed By | Jayaneyanraj Jayaraj | | Version | 1.0 |
| QA Tester's Log | None | | | | | |
| Tester's Name | Alvin | Date Tested | 04/07/2022 | | Test Case (Pass/Fail/Not) | Pass |
| S# | Prerequisites: | | S# | Test Data | | |
| 1 | Access to Internet Browser | | 1 | username= admin@gmail.com | | |
| 2 | User logged in as Employer | | 2 | password = 1qaz2wsxQ! | | |
| 3 | | | | | | |
| 4 | | | | | | |
| 5 | | | | | | |
| Test Scenario | Verify that the Employer can create a vending machine | | | | | |
| Step # | Step Details | Expected Results | Actual Results | | Pass / Fail / Not Executed / Suspended | |
| 1 | Click on 'Vending Machine' from side navbar | Redirects to Vending Machine management page | | | | |
| 2 | Click on the 'New VM' button | A pop up form is shown | | | | |
| 3 | Fill out vending machine details | Details are correct and no errors displayed on the form | | | | |
| 4 | Click on the 'Create' button | The Vending Machine is created and displayed in the Vending Machine List | | | | |
| 5 | | | | | | |

8.4.2 TC004_02: Test <View Vending Machine (SD010)>

| | | | | | | | |
|-----------------|--|---|--|---------------------------|--|------|--|
| Test Case ID | TC004_02 | Test Case Description | Test the functionality of View Vending Machine | | | | |
| Created By | Ragu R Naidu | Reviewed By | Jayaneyanraj Jayaraj | Version | | 1.0 | |
| QA Tester's Log | None | | | | | | |
| Tester's Name | Alvin | Date Tested | 04/07/2022 | | Test Case (Pass/Fail/Not) | Pass | |
| S# | Prerequisites: | | S# | Test Data | | | |
| 1 | Access to Internet Browser | | 1 | username= admin@gmail.com | | | |
| 2 | User logged in as Employer | | 2 | password = 1qaz!wsxQ! | | | |
| 3 | | | | | | | |
| 4 | | | | | | | |
| 5 | | | | | | | |
| Test Scenario | Verify that the Employer can view existing vending machine | | | | | | |
| Step # | Step Details | Expected Results | Actual Results | | Pass / Fail / Not Executed / Suspended | | |
| 1 | Click on 'Vending Machine' from side navbar | Redirects to Vending Machine page | | | | | |
| 2 | View Vending Machine Page | Vending Machine Page Displays List of Vending Machine | | | | | |
| 3 | | | | | | | |

8.4.3 TC004_03: Test <Update Vending Machine (SD011)>

| | | | | | | | | | | | | |
|-----------------|---|--------------|--|---|----|--|--|--|--|--|------|--|
| Test Case ID | | TC004_03 | | Test Case Description | | Test the functionality of Update Vending Machine | | | | | | |
| Created By | | Ragu R.Naidu | | Reviewed By | | Jayaneyanraj Jayaraj | | | Version | | 1.0 | |
| QA Tester's Log | | None | | | | | | | | | | |
| Tester's Name | | Alvin | | Date Tested | | 04/07/2022 | | | Test Case (Pass/Fail/Not) | | Pass | |
| S# | Prerequisites: | | | | S# | Test Data | | | | | | |
| 1 | Access to Internet Browser | | | | 1 | username= admin@gmail.com | | | | | | |
| 2 | User logged in as Employer | | | | 2 | password = 1qaz2wsxQ! | | | | | | |
| 3 | | | | | | | | | | | | |
| 4 | | | | | | | | | | | | |
| 5 | | | | | | | | | | | | |
| Test Scenario | Verify that the Employer can update a vending machine | | | | | | | | | | | |
| Step # | Step Details | | | Expected Results | | Actual Results | | | Pass / Fail / Not Executed / Suspended | | | |
| 1 | Click on 'Vending Machine' from side navbar | | | Redirects to Vending Machine management page | | | | | Pass / Fail / Not Executed / Suspended | | | |
| 2 | Click on the 'Edit VM' button | | | A pop up form is shown | | | | | | | | |
| 3 | Make necessary changes to Vending Machine Details | | | Details are correct and no errors displayed on the form | | | | | | | | |
| 4 | Click on the 'Save' button | | | The Vending Machine Details are updated | | | | | | | | |
| 5 | | | | | | | | | | | | |

8.4.4 TC004_04: Test <Delete Vending Machine (SD012)>

| | | | | | | | | | |
|-----------------|---|-----------------------|--|----|---------------------------|---------------------------|--|--|--|
| Test Case ID | TC004_04 | Test Case Description | Test the functionality of Delete Vending Machine | | | | | | |
| Created By | Ragu R Naidu | Reviewed By | Jayaneyanraj Jayaraj | | | Version | 1.0 | | |
| QA Tester's Log | None | | | | | | | | |
| Tester's Name | Alvin | Date Tested | 04/07/2022 | | | Test Case (Pass/Fail/Not) | Pass | | |
| S# | Prerequisites: | | | S# | Test Data | | | | |
| 1 | Access to Internet Browser | | | 1 | username= admin@gmail.com | | | | |
| 2 | User logged in as Employer | | | 2 | password = 1qaz2wsxQ! | | | | |
| 3 | | | | | | | | | |
| 4 | | | | | | | | | |
| 5 | | | | | | | | | |
| Test Scenario | Verify that the Employer can delete a vending machine | | | | | | | | |
| Step # | Step Details | | Expected Results | | Actual Results | | Pass / Fail / Not Executed / Suspended | | |
| 1 | Click on 'Vending Machine' from side navbar | | Redirects to 'Vending Machine management page | | | | | | |
| 2 | Click on the 'Delete VM' button | | A pop up form is shown | | | | | | |
| 3 | Click on 'Delete' | | Vending Machine is deleted | | | | | | |
| 4 | | | | | | | | | |

8.5 TC005: Test <Sales Management> Subsystem: <Manage Vending Machine Sales (UC005)>

This test contains the following test cases:

- (a) TC005_01: Test <Create Vending Machine Sales (SD013)>
- (b) TC005_02: Test <View Vending Machine Sales (SD014)>
- (c) TC005_03: Test <Update Vending Machine Sales (SD015)>
- (d) TC005_04: Test <Delete Vending Machine Sales (SD016)>

8.5.1 TC005_01: Test <Create Vending Machine Sales (SD013)>

| | | | | | | | | | |
|-----------------|--|--|-----------------------|---|--|----------------|---------------------------|--|------|
| Test Case ID | | TC005_01 | Test Case Description | | Test the functionality of create vending machine sales | | | | |
| Created By | | Alvin Hee Jun Sheung | Reviewed By | | Jayaneysanraj Jayaraj | Version | | 1.0 | |
| QA Tester's Log | | None | | | | | | | |
| Tester's Name | | Jegan | Date Tested | | 12/07/2022 | | Test Case (Pass/Fail/Not) | | Pass |
| S# | Prerequisites: | | | S# | Test Data | | | | |
| 1 | Access to Internet Browser | | | 1 | username= admin@gmail.com | | | | |
| 2 | Account already registered on the system | | | 2 | password = 1qaz2wsxQ! | | | | |
| 3 | | | | | | | | | |
| 4 | | | | | | | | | |
| 5 | | | | | | | | | |
| Test Scenario | | Verify that the Employer/Employee can create a vending machine sales | | | | | | | |
| Step # | Step Details | | | Expected Results | | Actual Results | | Pass / Fail / Not Executed / Suspended | |
| 1 | Click on 'Vending machine Sales' from side navbar and select the vending machine | | | Redirects to vending machine page | | | | | |
| 2 | Click on the 'New Vending machine Sale' button | | | A pop up form is shown | | | | | |
| 3 | Fill out new vending machine sales details | | | Details are correct and no errors displayed on the form | | | | | |
| 4 | Click on the 'Create' button | | | The sales is created and displayed in the page | | | | | |

8.5.2 TC005_02: Test <View Vending Machine Sales (SD014)>

| | | | | | | | | | |
|-----------------|--|--|-----------------------|---|--|----------------|--|--|--|
| Test Case ID | | TC005_02 | Test Case Description | | Test the functionality of view vending machine sales | | | | |
| Created By | | Alvin Hee Jun Sheung | Reviewed By | | Jayaneysanraj Jayaraj | Version | | 1.0 | |
| QA Tester's Log | | None | | | | | | | |
| Tester's Name | | Jegan | | Date Tested | | 12/07/2022 | | Test Case (Pass/Fail/Not) | |
| | | | | | | | | Pass | |
| S# | | Prerequisites: | | | | S# | | Test Data | |
| 1 | | Access to Internet Browser | | | | 1 | | username= admin@gmail.com | |
| 2 | | Account already registered on the system | | | | 2 | | password = 1qaz2wsxQ! | |
| 3 | | | | | | | | | |
| 4 | | | | | | | | | |
| 5 | | | | | | | | | |
| Test Scenario | | Verify that the Employer/Employee can view a vending machine sales of the respective vending machine | | | | | | | |
| Step # | | Step Details | | Expected Results | | Actual Results | | Pass / Fail / Not Executed / Suspended | |
| 1 | | Click on 'Vending machine Sale' from side navbar and select the respective vending machine | | Redirects to vending machine sales page | | | | | |
| 2 | | View Vending machine sales page of the selected vending machine | | List of vending machine sales details being displayed | | | | | |
| | | | | | | | | | |

8.5.3 TC005_03: Test <Update Vending Machine Sales (SD015)>

| | | | | | | | | | |
|-----------------|--|--|-----------------------|---|--|----------------|---------------------------|--|--|
| Test Case ID | | TC005_03 | Test Case Description | | Test the functionality of edit vending machine sales | | | | |
| Created By | | Alvin Hee Jun Sheung | Reviewed By | | Jayaneysanraj Jayaraj | Version | | 1.0 | |
| QA Tester's Log | | None | | | | | | | |
| Tester's Name | | Jegan | | Date Tested | | 12/07/2022 | | Test Case (Pass/Fail/Not) | |
| | | | | | | | | Pass | |
| S# | | Prerequisites: | | | S# | | Test Data | | |
| 1 | | Access to Internet Browser | | | 1 | | username= admin@gmail.com | | |
| 2 | | Account already registered on the system | | | 2 | | password = 1qaz2wsxQ! | | |
| 3 | | | | | | | | | |
| 4 | | | | | | | | | |
| 5 | | | | | | | | | |
| Test Scenario | | Verify that the Employer/Employee can edit a vending machine sales of the respective vending machine | | | | | | | |
| Step # | | Step Details | | Expected Results | | Actual Results | | Pass / Fail / Not Executed / Suspended | |
| 1 | | Click on 'Vending Machine Sales' from side navbar and select the vending machine | | Redirects to respective vending machine page | | | | | |
| 2 | | Click on the 'Edit Vending Machine Sales' button | | A pop up form is shown | | | | | |
| 3 | | Make necessary changes to vending machine sales Details | | Details are correct and no errors displayed on the form | | | | | |
| 4 | | Click on the 'Save' button | | The vending machine sales details are updated | | | | | |

8.5.4 TC005_04: Test <Delete Vending Machine Sales (SD016)>

| | | | | | | | | | | |
|-----------------|--|--|-----------------------|-------------|--|------------|---------------------------|---------------------------|--|--|
| Test Case ID | | TC005_04 | Test Case Description | | Test the functionality of delete vending machine sales | | | | | |
| Created By | | Alvin Hee Jun Sheung | Reviewed By | | Jayaneysanraj Jayaraj | Version | | 1.0 | | |
| QA Tester's Log | | None | | | | | | | | |
| Tester's Name | | Jegan | | Date Tested | | 12/07/2022 | | Test Case (Pass/Fail/Not) | | |
| | | | | | | | Pass | | | |
| S# | | Prerequisites: | | | S# | | Test Data | | | |
| 1 | | Access to Internet Browser | | | 1 | | username= admin@gmail.com | | | |
| 2 | | Account already registered on the system | | | 2 | | password = 1qaz2wsxQ! | | | |
| 3 | | | | | | | | | | |
| 4 | | | | | | | | | | |
| 5 | | | | | | | | | | |
| Test Scenario | | Verify that the Employer/Employee can delete a vending machine sales of the respective vending machine | | | | | | | | |
| Step # | | Step Details | | | Expected Results | | Actual Results | | Pass / Fail / Not Executed / Suspended | |
| 1 | | Click on 'Vending Machine Sales' from side navbar and select the vending machine | | | Redirects to respective vending machine page | | | | | |
| 2 | | Click on the 'Delete Vending Machine Sales' button | | | A pop up form is shown | | | | | |
| 3 | | Click on 'Delete' | | | Vending machine Sales is deleted | | | | | |

8.6 TC006: Test <Sales Management> Subsystem: <Manage Report (UC006)>

This test contains the following test cases:

- (a) TC006_01: Test <Create Stock History (SD017)>
- (b) TC006_02: Test <Update Stock History (SD018)>

8.6.1 TC006_01: Test <Create Stock History (SD017)>

| | | | | | | |
|-----------------|--|--|--|--|------|--|
| Test Case ID | TC006_01 | Test Case Description | Test the functionality of create Stock History | | | |
| Created By | Jegan | Reviewed By | Jayaneyanraj Jayaraj | Version | 1.0 | |
| QA Tester's Log | None | | | | | |
| Tester's Name | Jayaneyanraj Jayaraj | Date Tested | 12/07/2022 | Test Case (Pass/Fail/Not) | Pass | |
| S# | Prerequisites: | | S# | Test Data | | |
| 1 | Access to Internet Browser | | 1 | username= admin@gmail.com | | |
| 2 | Account already registered on the system | | 2 | password = 1qaz2wsxQ! | | |
| 3 | | | | | | |
| 4 | | | | | | |
| 5 | | | | | | |
| Test Scenario | Verify that the system can create Stock History | | | | | |
| Step # | Step Details | Expected Results | Actual Results | Pass / Fail / Not Executed / Suspended | | |
| 1 | Click on 'Vending machine Sales' from side navbar and select the vending machine | Redirects to vending machine page | | | | |
| 3 | Fill out new vending machine sales details | Details are correct and no errors displayed on the form | | | | |
| 4 | Click on the 'Create' button | The sales is recorded in stock history page and database | | | | |
| 5 | | | | | | |
| 6 | | | | | | |
| 7 | | | | | | |
| 8 | | | | | | |
| 9 | | | | | | |
| 10 | | | | | | |
| 11 | | | | | | |
| 12 | | | | | | |

8.6.2 TC006_02: Test <Update Stock History (SD018)>

| | | | | | | | |
|-----------------|--|-----------------------|---|----|---------------------------|---------------------------|--|
| Test Case ID | TC006_02 | Test Case Description | Test the functionality of update Stock History | | | | |
| Created By | Jegan | Reviewed By | Jayaneysanraj Jayaraj | | | Version | 1.0 |
| QA Tester's Log | None | | | | | | |
| Tester's Name | Ragu | Date Tested | 12/07/2022 | | | Test Case (Pass/Fail/Not) | Pass |
| S# | Prerequisites: | | | S# | Test Data | | |
| 1 | Access to Internet Browser | | | 1 | username= admin@gmail.com | | |
| 2 | User logged in as Employer | | | 2 | password = 1qaz2wsxQ! | | |
| 3 | | | | | | | |
| 4 | | | | | | | |
| 5 | | | | | | | |
| Test Scenario | Verify that the system can update Stock History | | | | | | |
| Step # | Step Details | | Expected Results | | Actual Results | | Pass / Fail / Not Executed / Suspended |
| 1 | Click on 'Vending Machine Sales' from side navbar and sele | | Redirects to respective vending machine page | | | | |
| 2 | Click on the 'Edit Vending Machine Sales' button | | A pop up form is shown | | | | |
| 3 | Make necessary changes to vending machine sales Details | | Details are correct and no errors displayed on the form | | | | |
| 4 | Click on the 'Save' button | | The vending machine sales are updated in Stock Histry | | | | |
| 5 | | | | | | | |
| 6 | | | | | | | |
| 7 | | | | | | | |
| 8 | | | | | | | |
| 9 | | | | | | | |
| 10 | | | | | | | |
| 11 | | | | | | | |
| 12 | | | | | | | |

9. Test Approach Analysis

Software testing techniques are essentially a set of procedures that aid in raising the overall quality and efficiency of every software development project. It aids in creating more effective test cases, which are the conditions or variables that a tester will use to judge whether the system being tested satisfies the requirements or functions well. To increase the effectiveness of the tests, various testing approaches are used during the testing process.

The testing strategy uses boundary value analysis (BVA) and equivalence partitioning (EP) as part of its black-box testing strategy (BVA). Black-box testing strategy often refers to a system's functional and behavioural testing. Without being aware of how the core code is implemented, the black-box approach focuses on evaluating the system's input and output to ensure that it is operating as intended. The EP and BVA procedures, which primarily aim to reduce the amount of test cases to be executed, are selected among the black-box testing methodologies.

Test Techniques are generally categorised into Five: We are only focusing on the first 2 categories.

- Boundary Value Analysis (BVA)
- Equivalence Class Partitioning
- Decision Table based testing.
- State Transition
- Error Guessing

Equivalence partitioning is one of the black-box testing strategy's approaches. In order to maintain adequate test coverage, it is used to reduce the number of potential test cases to an ideal level. The testing procedure can be made simpler, and the duties can be finished more quickly if the number of test cases is decreased.

The BVA approach, also known as boundary value analysis, is another method of black-box testing. This method, in contrast to EP, concentrates on the test cases' boundary values. The number of test cases can be further decreased by narrowing down the values to be examined at the boundary values. In contrast to the EP technique, which selects items at random from an equivalence class, BVA uses defined rules to choose values for test cases.

9.1 TC001: Test <User Management> Subsystem: <Login/Logout (UC001)>

9.1.1 TC001_01: Test <User Login/Logout (SD001)>

- **Range value**

| Test Case ID | Step# | Input data | Invalid | Valid | Invalid |
|--------------|-------|------------|---|-------------------|--|
| TC001_01 | 1 | Username | Less than 5 characters and not in email format | jameson@gmail.com | More than 70 characters and not in email format |
| TC001_01 | 1 | Password | Less than 5 characters and does not contain at least one capital letter,number and special character. | 1qaz2wsxQ! | More than 50 characters and does not contain at least one capital letter,number and special character. |

- **Boolean/Set value**

No input data.

9.2 TC002: Test <User Management> Subsystem: <Manage Users (UC002)>

9.2.1 TC002_01: Test <Create User (SD002)>

- Range value

| Test Case ID | Step# | Input data | Invalid | Valid | Invalid |
|--------------|-------|------------|---|-------------------|--|
| TC002_01 | 3 | Name | Less than 5 characters | Jameson | More than 50 characters |
| TC002_01 | 3 | Email | Less than 5 characters and not in email format | jameson@gmail.com | More than 70 characters and not in email format |
| TC002_01 | 3 | Password | Less than 5 characters and does not contain at least one capital letter,number and special character. | 1qaz2wsxQ! | More than 50 characters and does not contain at least one capital letter,number and special character. |

- Boolean/Set value

| Test Case ID | Step# | Input data | Valid | Invalid |
|--------------|-------|----------------|------------|---------------------------------------|
| TC002_01 | 3 | Contact Number | 0128976345 | Input is incorrect and not in numbers |

9.2.2 TC002_02: Test <Update User (SD003)>

- Range value

| Test Case ID | Step# | Input data | Invalid | Valid | Invalid |
|--------------|-------|------------|---|------------------------|--|
| TC002_02 | 3 | Name | Less than 5 characters | Jameson David | More than 50 characters |
| TC002_02 | 3 | Email | Less than 5 characters and not in email format | jamesondavid@gmail.com | More than 70 characters and not in email format |
| TC002_02 | 3 | Password | Less than 5 characters and does not contain at least one capital letter,number and special character. | 1wwq2wsxQ! | More than 50 characters and does not contain at least one capital letter,number and special character. |

- Boolean/Set value

| Test Case ID | Step# | Input data | Valid | Invalid |
|--------------|-------|----------------|------------|---------------------------------------|
| TC002_02 | 3 | Contact Number | 0128976369 | Input is incorrect and not in numbers |

9.2.3 TC002_03: Test <Delete User (SD004)>

No input data for this

9.3 TC003: Test <Stock Management> Subsystem: <Manage Stock (UC003)>

9.3.1 TC003_01: Test <Create Stock (SD005)>

- Range value

| Test Case ID | Step# | Input data | Invalid | Valid | Invalid |
|--------------|-------|---|-----------------------------------|---|--|
| TC003_01 | 3 | Stock Cost Price | -0.50 | 40.00 | 40000000.00 (Cost price must be logical) |
| TC003_01 | 3 | Stock Sell Price (Assume original price per tin is Rm1.00) | Selling price per tin < RM1.00 | RM1.00 <= Selling Price per tin <= RM3.00 | Selling Price per tin > RM3.00 (Price cannot be too expensive) |

- Boolean/Set value

| Test Case ID | Step# | Input data | Valid | Invalid |
|--------------|-------|----------------|-------|---|
| TC003_01 | 3 | Stock Name | Pepsi | Input is incorrect and not in alphabets |
| TC003_01 | 3 | Stock Quantity | 50 | Input is incorrect and not in numbers |

9.3.2 TC003_02: Test <View Stock (SD006)>

No input data.

9.3.3 TC003_03: Test <Update Stock (SD007)>

- Range value

| Test Case ID | Step# | Input data | Invalid | Valid | Invalid |
|--------------|-------|---|-----------------------------------|---|--|
| TC003_03 | 3 | Stock Cost Price | -1.00 | 50.00 | 70000000.00 (Cost price must be logical) |
| TC003_03 | 3 | Stock Sell Price (Assume original price per tin is Rm0.80) | Selling price per tin < RM0.80 | RM0.80 <= Selling Price per tin <= RM3.00 | Selling Price per tin > RM3.00 (Price cannot be too expensive) |

- Boolean/Set value

| Test Case ID | Step# | Input data | Valid | Invalid |
|--------------|-------|----------------|---------|---|
| TC003_03 | 3 | Stock Name | Nescafe | Input is incorrect and not in alphabets |
| TC003_03 | 3 | Stock Quantity | 45 | Input is incorrect and not in numbers |

9.3.4 TC003_04: Test <Delete Stock (SD008)>

No input data.

9.4 TC004: Test <Stock Management> Subsystem: <Manage Vending Machine (UC004)>

9.4.1 TC004_01: Test <Create Vending Machine (SD009)>

- Range value

| Test Case ID | Step# | Input data | Invalid | Valid | Invalid |
|--------------|-------|--------------------------|------------------------|-------------------|-------------------------|
| TC004_01 | 3 | Vending Machine Name | Less than 5 characters | Vending Machine 1 | More than 50 characters |
| TC004_01 | 3 | Vending Machine Location | Less than 5 characters | Skudai | More than 50 characters |

- Boolean/Set value

No input data.

9.4.2 TC004_02: Test <View Vending Machine (SD010)>

No input data.

9.4.3 TC004_03: Test <Update Vending Machine (SD011)>

- Range value

| Test Case ID | Step# | Input data | Invalid | Valid | Invalid |
|--------------|-------|--------------------------|------------------------|-------------------|-------------------------|
| TC004_03 | 3 | Vending Machine Name | Less than 5 characters | Vending Machine 2 | More than 50 characters |
| TC004_03 | 3 | Vending Machine Location | Less than 5 characters | Nusajaya | More than 50 characters |

- Boolean/Set value

No input data.

9.4.4 TC004_04: Test <Delete Vending Machine (SD012)>

No input data.

9.5 TC005: Test <Sales Management> Subsystem: <Manage Vending Machine Sales (UC005)>

9.5.1 TC005_01: Test <Create Vending Machine Sales (SD013)>

- Range value

| Test Case ID | Step# | Input data | Invalid | Valid | Invalid |
|--------------|-------|---|-------------------------|------------------------------|-------------------------|
| TC005_01 | 3 | Stock Quantity Sold (Assume existing stock is 50) | Stock Quantity Sold < 0 | 0< stock quantity sold <= 50 | Stock Quantity sold >50 |

- Boolean/Set value

| Test Case ID | Step# | Input data | Valid | Invalid |
|--------------|-------|--------------------|----------------------------|---|
| TC005_01 | 3 | Vending Machine Id | {1,2,3} | Input is incorrect and not in digit |
| TC005_01 | 3 | StockId | {Pepsi,Soya,Sarsi,Redbull} | Input is incorrect and not in alphabets |

9.5.2 TC005_02: Test <View Vending Machine Sales (SD014)>

No input data

9.5.3 TC005_03: Test <Update Vending Machine Sales (SD015)>

- Range value

| Test Case ID | Step# | Input data | Invalid | Valid | Invalid |
|--------------|-------|--|-------------------------|-------------------------------|--------------------------|
| TC005_03 | 3 | Stock Quantity Sold (Assume existing stock is 100) | Stock Quantity Sold < 0 | 0< stock quantity sold <= 100 | Stock Quantity sold >100 |

- **Boolean/Set value**

| Test Case ID | Step# | Input data | Valid | Invalid |
|--------------|-------|--------------------|----------------------------|---|
| TC005_03 | 3 | Vending Machine Id | {1,2,3} | Input is incorrect and not in digit |
| TC005_03 | 3 | StockId | {Pepsi,Soya,Sarsi,Redbull} | Input is incorrect and not in alphabets |

9.5.4 **TC005_04: Test <Delete Vending Machine Sales (SD016)>**

No input data

9.6 TC006: Test <Sales Management> Subsystem: <Manage Report (UC006)>

9.6.1 TC006_01: Test <Create Stock History (SD017)>

- Range value

| Test Case ID | Step# | Input data | Invalid | Valid | Invalid |
|--------------|-------|---|-------------------------|---|-------------------------|
| TC006_01 | 3 | Stock Quantity Sold (Assume existing stock is 50) | Stock Quantity Sold < 0 | 0<= stock quantity sold <= 50 | Stock Quantity sold >50 |
| TC006_01 | 3 | Date | 13/07/1990 | 13/07/2022 or 07/2022 or 2022 | 13/07/2030 |

- Boolean/Set value

| Test Case ID | Step# | Input data | Valid | Invalid |
|--------------|-------|--------------------|---------------|---|
| TC006_01 | 3 | Vending Machine Id | {1,2,3} | Input is incorrect and not in alphabets |
| TC006_01 | 3 | StockId | {Soya,Pepsi } | Input is incorrect and not in alphabets |

9.6.2 TC006_02: Test <Update Stock History (SD018)>

- Range Value

| Test Case ID | Step# | Input data | Invalid | Valid | Invalid |
|--------------|-------|--|------------|---|--|
| TC006_02 | 3 | Stock Quantity Sold (Assume existing stock is 100) | -100 | 30 | 5000 (Quantity should be less than the existing stock quantity) |
| TC006_02 | 3 | Date | 13/07/1990 | 13/07/2022 or 07/2022 or 2022 | 13/07/2030 |

- Boolean/Set value

| Test Case ID | Step# | Input data | Valid | Invalid |
|--------------|-------|--------------------|---------------|---|
| TC006_02 | 3 | Vending Machine Id | {1,2,3} | Input is incorrect and not in alphabets |
| TC006_02 | 3 | StockId | {Soya,Pepsi } | Input is incorrect and not in alphabets |

Appendices

Appendix A: Traceability Matrix

| Test Case ID | Use Case ID/ Sequence Diagram ID | Package ID |
|--|---|---------------|
| TC001 for <User Management> Subsystem <ul style="list-style-type: none"> • TC001_01 | UC001 <ul style="list-style-type: none"> • SD001 | P001 |
| TC002 for <User Management> Subsystem <ul style="list-style-type: none"> • TC002_01 • TC002_02 • TC002_03 | UC002 <ul style="list-style-type: none"> • SD002 • SD003 • SD004 | P001 |
| TC003 for <Stock Management> Subsystem <ul style="list-style-type: none"> • TC003_01 • TC003_02 • TC003_03 • TC003_04 | UC003 <ul style="list-style-type: none"> • SD005 • SD006 • SD007 • SD008 | P002 |
| TC004 for <Stock Management> Subsystem <ul style="list-style-type: none"> • TC004_01 • TC004_02 • TC004_03 • TC004_04 | UC004 <ul style="list-style-type: none"> • SD009 • SD010 • SD011 • SD012 | P002 |
| TC005 for <Sales Management> Subsystem <ul style="list-style-type: none"> • TC005_01 • TC005_02 • TC005_03 • TC005_04 | UC005 <ul style="list-style-type: none"> • SD013 • SD014 • SD015 • SD016 | P003 |
| TC006 for <Sales Management> Subsystem <ul style="list-style-type: none"> • TC006_01 • TC006_02 | UC006 <ul style="list-style-type: none"> • SD017 • SD018 | P003 |