Submission Four: System Testing

Reynard Etwaroo(USI#1039877), Tyreck Paul(USI#1024096), Zane Bishop(USI#1041471),

Ricardo Narine(USI#1040581), and Shivesh Mohamed(USI#1039116)

Faculty of Natural Sciences

Department of Computer Science, University of Guyana

CSE2101: Software Engineering I

Alicia Layne, Penelope DeFreitas, Juanelle Marks

December 9th, 2022

Table of Contents

Test Case 1

Test case ID:	TC_UI_1
Test priority:	HIGH
Test Title/Name:	Login Verification
Test Summary/Description:	In this test the user will attempt to access the system by using a username and a password associated with that user.
Pre-conditions:	1. Users must have a valid username and password. 2. User credentials should be registered to the system prior to login attempt. 3. Database must be functional and online.
Dependencies:	Assume that the system has already passed the following tests: TC_UI_3: system test to check for database's functionality to appropriately store data. TC_UI_4: system test to check that system's database responds with correct queries. TC_UI_5: system test to check if user buttons and input collection fields correctly collect user imputed data.
Test Steps:	1. Start program application. 2. On the login interface, enter a username in the appropriate input bar. 3. On the login interface, enter a password in the appropriate input bar. 4. Click the 'Login' button. 5. Wait for the system to validate credentials via login. 6. Record system response to inputted

	credentials. 7. Repeat with different username and password.
Test Data:	Input: Username: str, password: str Data Sets: **Valid Admin user credentials:** Username: ‘Rosewater2fa’:, Password: ‘nu4]>7Pn;TZ<B;G’ Username: ‘Agelin131’ Password: ‘!&UNCP*3HvNFw6r’ **Invalid Admin user credentials (invalid username: user does not exist)** Username: ‘Ectos’ Password: ‘|glSl-I>}IWCc_%’ Username: ‘Alan’ Password: ‘*So`"`lf%-|j.WN’ **Invalid Admin use credentials (incorrect password)** Username: ‘Rosewater2fa’ Password: ‘$rp*\,gJPls@r4*’ Username: ‘Agelin131’ Password: ‘!&UNCP*3HvNFw6r’ **Valid end user credentials:** Username: ‘Genos’ Password: ‘z}59rz<T4.F5T}8’ Username: ‘Jesca1234’ Password: ‘^^@>aL-^eU?Cn8e’ **Invalid end user credentials (invalid username: user does not exist):** Username: ‘Starplantinum13’

	Password: ‘q'lNKfR7neE_t#/’ Username: ‘Elene134’ Password: ‘5S^h"N*"nJul6Ud’ **Invalid end user credentials (incorrect password):** Username: ‘Genos’ Password: ‘1Xtm+rS"E)?I~&7’ Username: ‘Agelin131’ Password: ‘9suuG=<n%MX(`~X’
Expected Results:	1. Successful login: If username and password is correct users will be able to access the system, based on user position i.e Administrator or Employee. 2. Unsuccessful login: If the username is invalid the user will be presented with the error message “User does not exist. See create user account”. If the username is valid but the password is incorrect, the user will be presented with the error message “Incorrect password. Forgot your password? See reset password”
Post Condition:	1. Successful login: Program should display the appropriate user interface associated with the user position. 2. Unsuccessful login: Program will remain on the login interface.

Test Case 2

Test case ID:	TC_UI_2
Test priority:	HIGH
Test Title/Name:	Equipment condition checking
Test Summary/Description:	This test validates the system's functionality of correctly analyzing and recording a piece of equipment's current condition.
Pre-conditions:	1. The system's database must be online and functional. 2. User interfaces components are created and functional. 3. A test setup of directly feeding the system's equipment checking algorithm must be created. In the overall system, these images would normally be inputted via security cameras but for this test case, they will be fed directly into the system via tester input. a. This setup can be done by utilizing components of the user interfaces created and adding a dummy input field that collects jpegs, pngs and jpegs. That inputted image will then follow the same logic path that an image inputted via a security camera would follow. 4. Tester should have permissions to query and view the database
Dependencies:	Assume that the system has already passed the following tests: TC_UI_1: test to see login verification is functional.

	TC_UI_3: system test to check for database's functionality to appropriately store data. TC_UI_4: system test to check that system's database responds with correct queries. TC_UI_5: system test to check that the logic for inputting images into the system via security cameras is functional. TC_U1_6: system test to see if admin user's interface functionality of querying and viewing the database is functional.
Test Steps:	1. Open the application. 2. Log in to the administrative view. 3. Using the test setup interface (this is not on the admin view, it is a simple dummy field created separately), input an image of a piece of equipment in the dummy field. 4. Wait for a response from the algorithm.
Test Steps	5. Using the administrative view, query the database and look for the recently added equipment condition log. This should be at the top of the database after invoking a simple select all from table name (equimentconditionlogs) query 6. Record the algorithm's assessment of the equipment: the algorithm should report the equipment condition on a scale of: very good, good, okay, bad, very bad. 7. Record the algorithm's condition response next to the test data's

	assigned condition for that piece of equipment. 8. Repeat with different images from test data.
Test Data:	Images: png, jpg or jpeg This data set will comprise pictures of onsite equipment whose conditions will be labeled as: very good, good, okay, bad, very bad. Admin user credentials: Username: ‘Rosewater2fa’:, Password: ‘nu4]>7Pn;TZ<B;G’ Username: ‘Agelin131’ Password: ‘!&UNCP*3HvNFw6r’
Expected Results:	The system should record the assessed condition of the equipment in the database table ‘EquipmentConditionLogging’ with the appropriate times of entry listed. Each time a piece is inserted, the administrative interface should receive an update and display the newly added log at the top of the view.
Post Condition:	The data of the database’s table will remain changed, without intervention, due to the insertion of the newly inputted equipment condition data.