Java Institute for Advanced Technology

UNIT NAME: SOFTWARE ENGINEERING II

UNIT ID: HF2W 04

ASSIGNMENT ID: HF2W 04/AS/05

NAME: M.R.P.N.THARUKSHA RAJAPAKSHA

STUDENT ID: 2019/2020/CO/SE/I2/029

SCN NO: 207977608

NIC: 200019401866

BRANCH: JAVA INSTITUTE, COLOMBO

Contents

Q1: Design test cases

1. Traffic Controller System

Project Name	**Traffic Controller System**						
Test Case Number	Test Case Name	Description	Pre-Conditions	Test Data	Step Name	Step Description	Expected Result
1	Correct Login	Login with correct username & password	Need valid username & password to login	username: admin password: 1234	Enter username	Type given username	Successful login
					Enter password	Type correct password	
					Click Login button	Click Login button	
2	Incorrect Login	Login with correct username & incorrect password	Need valid username & incorrect password	username: admin password: 1111	Enter username	Type given username	Fail to login
					Enter password	Type incorrect password	
					Click Login button	Click Login button	
3	Register Existing User	Register with existing username	Need existing username	username: tharuksha password: 4042 name: Tharuksha	Enter username	Enter username	Fail to register
					Enter name	Enter name	
					Enter password	Enter password	
					Click Register button	Click Register button	
4	Register New User	Register user with new username	Unused username	username: testuser password: 4042345 name: Testuser	Enter username	Enter username	Successful register
					Enter name	Enter name	
					Enter password	Enter password	

					Click Register button	Click Register button	
5	Add New Route	Add new route with correct data	admin permission, valid route data	routeId: 154 from: Kiribathgoda to: Agulana	Enter routeId	Enter routeId	Route added successfully & auto updated the table
					Enter from /to data	Enter from /to data	
					Click Add Route button	Click Add Route button	
6	Add Existing Route	Add route with existing id	admin permission, existing route id	routeId: 154 from: Kiribathgoda to: Agulana	Enter routeId	Enter routeId	Fail to add, route already exist
					Enter from /to data	Enter from /to data	
					Click Add Route button	Click Add Route button	
7	Traffic Light Profile Timing	Traffic light profile timing working correctly	new traffic light profile, traffic lights to create a profile , on site tester	traffic lights: 142a,144a,144b	Select traffic lights	Select traffic lights	Timing works correctly
					Enter timing	Enter timing	
					Click Save button	Click Save button	
8	Traffic Light Malfunction Mode	Traffic Light Malfunction mode working correctly	Intentionally malfunctioned traffic light	malfunction traffic light”44b	Create test error	Create test error to check whether yellow light blinking mode (error mode) working in traffic light	Only yellow light blinks. other lights off
					Select traffic lights	Select traffic lights	
					Enter timing	Enter timing	

9	Test Working Pedestrian Button	Pedestrian button successfully working	Press pedestrian button by pedestrian	button value = 1	Press pedestrian button	Pedestrian button worked	Turn on the red light, walking signal and countdown
10	Test Not Working Pedestrian Button	Pedestrian button not working	Press pedestrian button by pedestrian	button value = 0	Press pedestrian button	Pedestrian button not worked	The red light, walking signal and countdown don't work
11	Start Working Countdown For Pedestrian	Countdown working	Press pedestrian button by pedestrian	i = 30s	Press pedestrian button	Pedestrian button worked	Start the countdown & walking signal
12	Start Not Working Countdown For Pedestrian	Countdown not working	Press pedestrian button by pedestrian	i = 30s	Press pedestrian button	Pedestrian button worked	Countdown doesn't start
13	Test Red Bulb Turning On	Turning on red bulb when countdown starts	Press pedestrian button by pedestrian	i = 30s	Press pedestrian button	Pedestrian button worked	Turning on the red bulb
14	Test Red Bulb Not Turning On	Not turning on red bulb when countdown starts	Press pedestrian button by pedestrian	i = 30s	Press pedestrian button	Pedestrian button worked	Red bulb not turning on
15	Test Yellow Bulb Turning On	Turning on yellow bulb when countdown reach 5 s	Starting countdown	i = 5s	i = 5s	Countdown should be 5 seconds	Turn on yellow bulb
16	Test Yellow Bulb Not Turning On	Not turning on yellow bulb when countdown reach 5 s	Starting countdown	i = 5s	i = 5s	Countdown should be 5 seconds	Yellow bulb not turning on
17	Test Green Bulb Turning On	Turning on green bulb when countdown ends	Turning off yellow bulb	i = 0s	Turning off Yellow bulb	Yellow bulb worked	Turning on green bulb
18	Test Green Bulb Not Turning On	Not turning on green bulb when countdown ends	Not turning the yellow bulb	i = 0s	Not turning off yellow bulb	Not turning off yellow bulb	Not turning on green bulb

2. University Management System

Project Name	**University Management System**						
Test Case Number	Test Case Name	Description	Pre-Conditions	Test Data	Step Name	Step Description	Expected Result
1	Correct Login	Login with correct username & password	Need valid username & password to login	username: admin password: 1234	Enter username	Type given username	Successful login
					Enter password	Type correct password	
					Click Login button	Click Login button	
2	Incorrect Login	Login with correct username & incorrect password	Need valid username & incorrect password	username: admin password: 1111	Enter username	Type given username	Fail to login
					Enter password	Type incorrect password	
					Click Login button	Click Login button	
3	Incorrect Login With Incorrect Username	Login with incorrect username & correct password	Invalid username and valid password	username: admin123 password: 1234	Give username	Type invalid username	Fail to login
					Give password	Type valid password	
					Click Login button	Click Login button	
4	Register New Student	Register with non-existing information	Unused username	username: tharuksha email: tharuksha@gmail.com password: 4042	Give username	Type unused username	Successful register
					Give email	Type email	
					Give password	Type valid password	
					Click Register button	Click on Register button	
5	Check Student Login	Login with correct username	Valid username and password	username: tharuksha password: 4042	Give username	Type valid username	Successful login
					Give password	Type valid password	

		and password			Click Login button	Click Login button	
6	Incorrect Student Login With Incorrect Password	Login with correct username and incorrect password	Valid username and invalid password	username: tharuksha password: 232@@	Give username	Type valid username	Fail to login
					Give password	Type invalid password	
					Click login button	Click Login button	
7	Incorrect Student Login With Incorrect Username	Login with incorrect username and correct password	Invalid username and valid password	username: user123 password: 4042	Give username	Type invalid username	Fail to login
					Give password	Type valid password	
					Click Login button	Click Login button	
8	Login With Empty Fields	Try to login with empty password field	Valid username	username: tharuksha password:	Give username	Type username	Fail to login
					Click Login button	Click Login button	
9	Update Student	Update student details	Valid username and password	email: tharu@gmail.com username: tharuksha password: taru1234	Give new email	Type new email	Successful update
					Give new password	Type valid password	
					Click Update button	Click Update button	
10	Delete Student	Delete student Details	Valid username	username: tharuksha	Give username	Type valid username	Successful delete
					Click Delete button	Click Delete button	
11	Register Lecturer With Valid Information	Input lecturer with new information	Unused nic	nic: 200019401866 name: Tharaka password: tharaka12	Give new nic	Type new nic	Successful register
					Give name	Type name	
					Give password	Type valid password	
					Click on register	Click on register button	
12	Register Lecturer With Existing Information	Register with existing nic	used nic	nic: 200019401866 name: Tharaka	Give used nic	Type used nic	Fail to register
					Give name	Type name	
					Give password	Type valid password	

				password: tharaka12	Click Register button	Click Register button	
13	Lecturer Login	Login with correct nic and password	Valid nic and password	nic: 200019401866 password: tharaka12	Give nic	Type valid nic	Successful login
					Give password	Type valid password	
					Click login button	Click login button	
14	Incorrect Lecturer Login With Incorrect Password	Login with correct nic and incorrect password	Valid nic and invalid password	nic: 200019401866 password: tharaa1	Give nic	Type valid nic	Fail to login
					Give password	Type invalid password	
					Click login button	Click login button	
15	Incorrect Lecturer Login With Incorrect NIC	Login with incorrect nic and correct password	Invalid nic and valid password	nic: 200019401v password: tharaka12	Give nic	Type invalid nic	Fail to login
					Give password	Type valid password	
					Click login button	Click login button	
16	Update Lecture	Update lecture details	Valid nic and password	nic: 200019401866 name: Tharaka password: tharaka12	Update password	Type new password	Successful update
					Click update button	Click update button	
17	Login With Empty Fields	Try to login with empty password field	Valid nic	nic: 200019401866 Password:	Give nic	Type nic	Fail to login
					Click login button	Click login button	
18	Delete Lecture	Delete student Details	Valid nic	nic: 200019401866	Give nic	Type valid nic	Successful delete
					Click delete button	Click delete button	

Q2: Explain various types of testing in detail.

1. Alpha Testing

It is the most widely used test in the software industry. The purpose of this test is to identify all possible problems or errors before releasing them to the market or the user. The alpha test will be performed at the end of the software development phase but before the beta test. Still, small design changes can be made as a result of such testing. Alpha testing is performed on the developer's site. This type of testing can create an internal virtual user environment.

2. Back-end Testing

When an input or data is entered in a front-end application, it is stored in a database and such database testing is called Database Testing or Backend Testing. There are various databases such as SQL Server, MySQL, and Oracle. Database inspection includes checking the table structure, programming, stored procedure, data structure, and so on. In back-end testing, the GUI is not connected, testers are connected directly to the database with proper access, and testers can easily verify data by running multiple queries in the database. This backup test can detect issues such as data loss, blockage, and data corruption, and it is important to address these issues before the system goes live in the production environment.

3. Beta Testing

Beta testing is formal software testing performed by the customer. It is done in a real environment before releasing the product to the market for real end-users. Beta testing is done to ensure that there are no major failures in the software or product, which satisfies the business needs from an end-user perspective. The beta test is successful when the customer accepts the software. Typically, this test is usually performed by end-users or others. This is the final test done before issuing the application for commercial purposes. Generally, the beta version of a released software or product is limited to a specific number of users in a specific area. So the end-user actually uses the software and shares feedback with the company. The company takes the necessary action before releasing the software worldwide.

4. Black Box Testing

Internal system designs are not considered in these types of tests. Tests are based on requirements and functionality. Detailed information on the advantages, disadvantages, and types of Black Box testing can be found here.

5. Boundary Value Testing

This type of test checks the behavior of the application at the boundary level. Boundary value testing is performed to check for errors in boundary values. Boundary value testing is used to check a range of different numbers. Each range has an upper and lower boundary and tests are performed on these boundary values. If the test requires a numerical test range from 1 to 500, the boundary value test is performed on 0, 1, 2, 499, 500, and 501 values.

6. Branch Testing

This is a type of white box testing and is performed during unit testing. Branch inspection, the name suggests, is a thorough check of the code as it travels through each branch.

7. Comparison Testing

Comparing a product's strengths and weaknesses with its predecessors or other similar products are called comparison testing.

8. Component Testing

This is largely done by the developers after the unit testing is completed. Component testing involves testing multiple functions as a single code, the purpose of which is to identify whether there is any error after connecting those multiple functions.

9. End-to-End Testing

Similar to system testing, end-to-end testing involves testing a complete application environment in a situation that mimics real-world usage, such as interacting with a database, using network communications, or interacting with other hardware, applications, or systems if suitable.

10. Example Testing

This means real-time testing. Example testing includes real-time instances, which include instances based on the tester's experience.

11. Functional Testing

This type of test ignores internal components and focuses only on the output to test whether it is in accordance with the requirement or not. This is a black box type test directed to the functional requirements of an application.

12. Graphical User Interface (GUI) Testing

The purpose of this GUI testing is to validate the GUI according to business requirements. The desired GUI of the application is specified in the detailed design document and on the GUI mockup screens. The GUI testing includes the size of the buttons and input fields on the screen, as well as the alignment of all text, tables, and contents in the table. It also validates the app menu, validates that the page does not fluctuate after selecting various menus and menu items, and the alignment remains the same after hovering the mouse over the menu or sub-menu.

13. Install/Uninstall Testing

Installation and uninstallation testing is performed on full, partial, or upgraded installation/uninstallation processes of different operating systems under different hardware or software environments.

14. Integration Testing

Testing all integrated modules to verify integrated functionality after integration is called integrated testing. Modules are usually code modules, individual applications, client and server applications on a network, etc. This type of testing is especially relevant for client/server and distributed systems.

15. Load Testing

It is a non-functional type of test and the purpose of load testing is to test how much load or maximum performance a system can handle without degrading performance. Load testing helps to detect any issues that may be causing the maximum capacity of the system and software performance to deteriorate under a specific load. Load testing is done using tools such as JMeter, LoadRunner, WebLoad, Silk performer, etc.

16. Negative Testing

Testers have the mindset of “attitude to break” and validate if a system or application breaks down using negative testing. Negative testing technology is performed using incorrect data, invalid data, or input. If the system causes an error with an invalid input and behaves as expected, it is valid.

17. Non-Functional Testing

This is a type of test that usually has a separate team for each organization, called a non-functional testing team (NFT) or performance team. On-functional testing includes testing of non-functional requirements such as weight testing, stress testing, safety, volume, recovery testing, and so on. The purpose of the NFT test is to ensure that the response time of the software or application is fast enough. Business need. No page or system should take long to load and should be maintained at maximum load.

18. Performance Testing

This term is often exchanged with the 'stress' and 'load' testing. Performance testing is performed to test whether the system meets performance requirements. Different performance and loading tools are used to perform this test.

19. Recovery Testing

It is a type of test that confirms how well the application or system is recovering from crashes or disasters. Recovery testing determines whether a system can continue to function after a disaster. Assume that the application receives data over a network cable and that network is suddenly disconnected. After a while, connect the network cable; then the system should start receiving data from the place where the connection was lost due to disconnection of the network cable.

20. Security Testing

It is a type of test performed by a special inspection team. Any hacking method can penetrate a system. Security testing is done to check how the software, application, or website is protected from internal and external threats. This test includes malicious programs, how much software is protected from viruses, and how secure and robust the approval and verification processes are. It also examines how the software behaves for any hacker attack and malware, and how the software maintains data security after such a hacker attack.

21. System Testing

Under the system testing technique, the entire system is tested according to requirements. It is a black-box type test based on the overall requirements specifications and covers all integrated parts of the system.

22. Unit Testing

Testing a single software component or module is called Unit Testing. It is usually done by the programmer and not by the testers, as it requires detailed knowledge of the internal program design and code. It may also be necessary to develop test drive modules or test harnesses.

23. White Box Testing

The white box testing is based on knowledge of the internal logic of an application's code. It is also known as glass box testing. Internal software and code work must know how to perform these types of tests. Under this, tests are based on covering code statements, branches, paths, conditions, etc.