Academic Year: 2023_24

Experiment No. 7

Team Members :

Riddhi Shah-60004210161

Dhruvin Chawda-60004210159

Aim: Design test scenarios and test cases for your SRS.

Implementation :

1. Test Scenario :

Column #1: Test scenario ID

Every entity in our testing process has to be uniquely identifiable. So, every test scenario has to be assigned an ID. The rules to follow while assigning this ID have to be defined. For the sake of this article we are going to follow the naming convention as: TS(prefix that stands for Test Scenario) followed by '_' , module name MI(my Info module of the Orange HRM project) followed by '_' and then the sub section (eg: MIM for My info module, P for photograph and so on)followed by a serial number. An example would be: "TS_MI_MIM_01".

Column #2: Requirement

It helps that when we create a test scenario we should be able to map it back to the section of the SRS document where we picked it from. If the requirements have IDs we could use that. If not section numbers or even page numbers of the SRS document from where we identified a testable requirement will do.

Column #3: Test scenario description

A one liner specifying 'what to test'. We would also refer to it as test objective.

Column #4: Importance

This is to give an idea about how important certain functionality is for the AUT. Values like high, medium and low can be assigned to this field. You could also choose a point system, like 1-5, 5 being most important, 1 being less important. Whatever the value this field can take, it has to be pre-decided.

Column #5: No. of Test cases

A rough estimate on how many individual test cases we might end up writing that one test scenario. For example: To test the login- we include these situations: Correct username and password. Correct username and wrong password. Correct password and wrong username. So, validating the login functionality will result in 3 test cases.

Academic Year: 2023_24

Project Name	Virtual Connect
Reference Document	Software Requirement Specification , Version 1
Created by	Riddhi Shah, Dhruvin Chawda
Date of creation	04-Mar-24
Date of review	10-Mar-24

Test scenario ID	Requirement-reference document index	Test scenario description	Impor-tance	No. of test cases
TS_A_001	3.2.1	Check that users can create accounts securely with email/password or SSO methods.	High	2
TS_A_002	3.2.1	Check if the user is able to see the "Personal Details" on logging in the first time	Medium	1
TS_A_003	3.2.1	validate if the user is able to edit the fields other than the following and save changes Personal Details	High	1
TS_M_001	3.2.2	Validate if the user(host) can create/ schedule and manage virtual meetings effectively.	High	3
TS_M_002	3.2.3	Validate the quality and functionality of audio and video conferencing.	High	4
TS_M_003	3.2.4	Validate the software has seamless screen sharing and collaboration features.	Medium	3
TS_M_004	3.2.5	Validate the software has Chat and Messaging Features	Medium	3
TS_M_005	3.2.6	Validate if all the participant can joined, also remove a participant if required	High	3
TS_M_006	3.2.7	Validate meeting recording and playback functionality.	High	2
TS_M_007	3.2.8	validate if Integration with Calendar and Productivity	Medium	2

		Tools is working properly		
TS_S_001	3.2.9	Validate security measures implemented within the meeting software.	High	3
TS_AR_001	3.2.10	Validate the generation and accuracy of user engagement and meeting performance reports.	Medium	2

2. Test Cases :

Test case ID: Unique ID for each test case. Follow some convention to indicate types of test. E.g. 'TC_UI_1' indicating 'user interface test case #1'.

Test Objective: Describe what the test case is actually going to test.

Pre-condition: Any prerequisite that must be fulfilled before execution of this test case. List all pre-conditions in order to successfully execute this test case.

Dependencies: Mention any dependencies on other test cases or test requirement.

Test Steps: List all test execution steps in detail. Write test steps in the order in which these should be executed. Make sure to provide as much details as you can. **Tip** – to efficiently manage test case with lesser number of fields use this field to describe test conditions, test data and user roles for running test.

Test Data: Use of test data as an input for this test case. You can provide different data sets with exact values to be used as an input.

Expected Result: What should be the system output after test execution? Describe the expected result in detail including message/error that should be displayed on screen.

Post-condition: What should be the state of the system after executing this test case?

Actual result: Actual test result should be filled after test execution. Describe system behavior after test execution.

Project Name	Virtual Connect
Reference Document	Software Requirement Specification , Version 1
Created by	Riddhi Shah, Dhruvin Chawda
Date of creation	04-Mar-14
Date of review	10-Mar-14

Academic Year: 2023_24

Test case ID	Test Objective	Precondition	Steps:	Test data	Expected result	Post-condition
TC_Auth_001	Verify user can create an account with email/password credentials.	User is on the registration page.	1. Enter valid email address and password. 2. Click on the "Create Account" button.	Valid email address and password.	User account is successfully created, and user is redirected to the login page.	User can log in using the created credentials.
TC_Auth_002	Verify user can create an account using Single Sign-On (SSO) method.	User is on the registration page.	1. Click on the SSO provider (e.g., Google, Microsoft). 2. Authenticate with the SSO provider.	N/A	User account is successfully created using SSO authentication, and user is redirected to the login page.	User can log in using SSO credentials
TC_Auth_003	Login in with credentials	User is on the Login page	1. Enter valid credentials or click on proper SSO method 2. Authenticate the user	valid Credentials	User is logged on to the software and use it	
TC_Auth_004	Error message on unsuccessful login	User is on the Login page	`1. Enter invalid credentials or click on unproper SSO method 2. user Authentication failed	"A valid username"	An Error message is displayed and the user is not logged in to the .	
TC_Meeting_001	Verify user(host) can schedule a one-time meeting.	User is logged in and on the meetings dashboard.	1. Click on the "Schedule Meeting" button. 2. Fill in meeting details such as title, date, time, and participants. 3.Click on the "Schedule" button.	Meeting details including title, date, time, and participants.	Meeting is successfully scheduled and appears on the user's calendar.	User can see the scheduled meeting on the dashboard and calendar.
TC_Meeting_002	Verify user(host) can edit a scheduled meeting.	User has scheduled meetings.	1. Select the scheduled meeting to be edited. 2. Click on the "Edit" button. 3. Modify meeting details. 4. Click on the "Save Changes" button.	Modified meeting details.	Meeting details are successfully updated, and changes are reflected in the schedule.	Updated meeting details are saved and visible to participants.
TC_Meeting_003	Verify user can join the meeting	User has invitation or meeting link/ID for the meeting	1. Click on the meeting link or enter the meeting ID 2. Click on join meeting	Meeting ID	User joins the meeting	
TC_Meeting_004	Verify user can add participants to a meeting.	User is in a meeting with permission to add participants.	1. Click on the "Add Participants" button. 2. Enter the email addresses or User ID of participants to be added. 3. Click on the "Add" button.	Email addresses or user ID of participants to be added.	Participants are successfully added to the meeting, and they receive invitations.	Added participants can join the meeting.

Academic Year: 2023_24

TC_Conferencing_001	Verify audio and video are clear and synchronized during a meeting.	User is in an ongoing meeting with audio and video enabled.	1. Join the meeting. 2. Enable audio and video. 3. Check audio and video quality.	N/A	Audio and video are clear, synchronized, and without distortion.	Users can effectively communicate using audio and video features.
TC_Sharing_001	Verify user can share screen with multiple participants.	User is in a meeting with screen sharing enabled.	1. Click on the "Share Screen" button. 2. Select the screen/application/window to share. 3. Click on the "Share" button.	N/A	User's screen is successfully shared with all meeting participants.	Other participants can view the shared screen without issues.
TC_Sharing_002	Verify collaborative document editing feature.	User is in a meeting with document collaboration enabled.	1. Open a document for collaboration. 2. Allow other participants to edit the document. 3. Make edits to the document.	Collaborative document.	All participants can simultaneously edit the document, and changes are synced in real-time.	Document edits are saved and visible to all participants.
TC_Chat_001	Verify in-meeting chat functionality.	User is in an ongoing meeting.	1. Open the chat window. 2. Send a message to all participants.	Message to be sent.	Message is sent successfully and visible to all meeting participants.	Chat history is updated with the sent message.
TC_Meeting_005	Verify user can leave the meeting .	User is in an ongoing meeting.	1. Click on the leave button	N/A	user leaves the meeting successfully	

Conclusion:

Thus, we are able to create test scenarios and test cases for our case study.