White Pinnacle

ITERATION 2 TEST CASE

Contents

[View All Case Scenarios 2](#_Toc400809034)

[View all Case Scenarios (Lecturer) 2](#_Toc400809035)

[View all Case Scenarios (Admin) 4](#_Toc400809036)

[Edit and Delete Case 5](#_Toc400809037)

[Edit Case (Lecturer) 5](#_Toc400809038)

[Delete Case (Lecturer) 7](#_Toc400809039)

[Activate Case Scenario 8](#_Toc400809040)

[Activate Case Scenario (Lecturer) 8](#_Toc400809041)

[Activate Case Scenario (Admin) 10](#_Toc400809042)

[View Patient’s Personal Information 11](#_Toc400809043)

[View Patient’s Personal Information (Student) 11](#_Toc400809044)

# View All Case Scenarios

## View all Case Scenarios (Lecturer)

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| S/N | Functionality | Description | Test Inputs | Test Procedure | Expected Results | Actual Results | Pass/ Fail | Remarks |
| 1 | View all scenario (lecturer) | Validate that cases created should appear on the viewScenarioLecturer.jsp | **ScenarioID**: SC10 **ScenrarioName**: QWERTY **ScenarioDescription**: The simulated clinical experience was designed to expose the learner.  **Status**: activated **AdmissionInfo**: 48 year old man met with an minor accident | 1. Create new scenario in phpMyAdmin 2. Key in scenarioID, scenarioName, scenarioDescription, status and admission information | User should be able to all see the created cases |  |  |  |
| 2 | View all scenario (lecturer) | Validate that when cases appear on the viewScenarioLecturer.jsp, the reveal modal will display the information of the case | NA | 1. Click on the image of the case 2. Information of the case will be shown on the reveal modal | User should be able to see the information of the case in the reveal modal |  |  |  |
| 3 | View all scenario (lecturer) | Validate that cases created should appear on the viewScenarioLecturer.jsp | **ScenarioID**: SC11 **ScenrarioName**: QWERTY **ScenarioDescription**: The simulated clinical experience was designed to expose the learner.  **Status**: activated **AdmissionInfo**: 48 year old man met with an minor accident | 1. Create new scenario in phpMyAdmin 2. Key in scenarioID, scenarioName, scenarioDescription, status and admission information | User should be able to view the newly created case with the correct information |  |  |  |
| 4 | View all scenario (lecturer) | Validate that the number of cases displayed is the same as the number of cases created | NA | 1. Check phpMyAdmin for the number of cases created.  2. Check if the number of cases that appear on home page is the same as the number of cases created in the database. | User should be able to view all the cases created |  |  |  |

## View all Case Scenarios (Admin)

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| S/N | Functionality | Description | Test Inputs | Test Procedure | Expected Results | Actual Results | Pass/ Fail | Remarks |
| 1 | View all scenario (admin) | Validate that user is able to view all scenario cases at viewScenarioAdmin.jsp | **ScenarioID**: SC10 **ScenrarioName**: QWERTY **ScenarioDescription**: The simulation clinical experience was designed to expose the learner.  Status: activated **AdmissionInfo**: 48 year old man met with an minor accident | 1. Create new scenario in phpMyAdmin 2. Key in scenarioID, scenarioName, scenarioDescription, status and admission information  3. Check if a new case is create in viewScenarioAdmin.jsp | User should be able to see the newly added case scenario |  |  |  |
| 2 | View all scenario (admin) | Validate that cases created should be displayed | **ScenarioID**: SC10 **ScenrarioName**: QWERTY **ScenarioDescription**: The simulate clicnical experience was designed to expose the learner.  **Status**: activated **AdmissionInfo**: 48 year old man met with an minor accident | 1. Create new scenario in phpMyAdmin 2. Key in scenarioID, scenarioName, scenarioDescription, status and admission information | User should be able to view the newly created case with the correct information |  |  |  |
| 3 | View all scenario (admin) | Validate that the number of cases displayed is the same as the number of cases created | NA | 1. Click on the case management tab 2. Look at the number of cases using the serial number and compare with the total number of cases created in the database | User should be able to view the case created |  |  |  |

# Edit and Delete Case

## Edit Case (Admin)

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| S/N | Functionality | Description | Test Inputs | Test Procedure | Expected Results | Actual Results | Pass/ Fail | Remarks |
| 1 | Edit Case Scenario (Admin) | Validate that user is able to edit and save Activated case scenario information | NA | 1. Login as Admin 2. Click on edit case in viewScenarioAdmin.jsp 3. Click edit button 4. Change the description of the case 5. Click on the "Save" button | Case information should be saved successfully with success message |  |  |  |
| 2 | Edit Case Scenario (Admin) | Validate that user is able to edit and save deactivated case scenario information | NA | 1. Login as Admin 2. Click on edit case in viewScenarioAdmin.jsp 3. Click edit button 4. Change the description of the case 5. Click on the "Save" button | Case information should be saved successfully with success message |  |  |  |
| 3 | Edit Case Scenario (Admin) | Validate that all fields must be filled up | 1. Leave the case name empty 2. fill up the case description and admission information | 1. Click the "Save" button | User should not be able to save the case details and will be prompted to fill up the case name |  |  |  |
| 4 | Edit Case Scenario (Admin) | Validate that all fields must be filled up | 1. Leave the case description empty 2. fill up the case name and admission information | 1. Click the "Save" button | User should not be able to save the case details and will be prompted to fill up the case description |  |  |  |
| 5 | Edit Case Scenario (Admin) | Validate that all fields must be filled up | 1. Leave the admission information empty 2. fill up the case name and case description | 1. Click the "Save" button | User should not be able to save the case details and will be prompted to fill up the admission information |  |  |  |
| 6 | Edit Case Scenario (Admin) | Validate that the edited information are updated | 1. Fill in case name, case description and admission information | 1. Click on the "Edit" button for case 1  2. Edit the information  3. Click the "Save" button | User should be able to see the updated information in viewScenarioAdmin.jsp |  |  |  |
| 7 | Edit Case Scenario (Admin) | Validate that only one case is activated | Na | 1. Click on the "Edit" button for a deactivated case 2. Select the activated button  3. Click the "Save" button | User should be alerted that activating this case would deactivate the other cases |  |  |  |
| 8 | Edit Case Scenario (Admin) | Validate that user is able successfully edit a deactivated case | Na | 1. Click on the "Edit" button for a deactivated case 2. Edit the information  3. Click the "Save" button | User should not be prompted any message |  |  |  |

## Delete Case (Admin)

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| S/N | Functionality | Description | Test Inputs | Test Procedure | Expected Results | Actual Results | Pass/ Fail | Remarks |
| 1 | Delete Case Scenario (Admin) | Validate that user is able to delete case scenarios | Na | 1. Click on the "Delete" button 2. Click on "Ok" on the prompt | Case scenario should be deleted and removed from the page. Success message should be shown |  |  |  |
| 2 | Delete Case Scenario (Admin) | Validate that user is not able to delete case scenarios if cancel is clicked | Na | 1. Click on the "Delete" button 2. Click on "Cancel" on the prompt | Case scenario should not be deleted and still remain on the page. |  |  |  |
| 3 | Delete Case Scenario (Admin) | Validate that deleted case scenarios are deleted from the database | Na | 1. Click on the "Delete" button 2. Click on "Ok" on the prompt 3. Go to phpMyAdmin, check if case scenario is still shown | Case scenario should be removed from the database |  |  |  |

# Activate Case Scenario

## Activate Case Scenario (Lecturer)

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| S/N | Functionality | Description | Test Inputs | Test Procedure | Expected Results | Actual Results | Pass/ Fail | Remarks |
| 1 | Activate Case (Lecturer) | User should be able to activate the cases | NA | 1. Login to NPSTAFF account 2. Click on a deactivated case 3. Click "activate case" button on the reveal modal | User should be able to deactivate case with the success message. |  |  |  |
| 2 | Activate Case (Lecturer) | User should only be able to activate one case | NA | 1. Login to NPSTAFF account 2. Click on a case 2 3. Click "activate case" button on the reveal modal  4. Login to phpMyAdmin  5. Check if case 2 is activated | Case 2 should be activated |  |  |  |
| 3 | Activate Case (Lecturer) | User should be able to activate the cases | NA | 1. Login to NPSTAFF account 2. Deactivate case 1 3. Click on a case 2 4. Click "activate case" button on the reveal modal  5. Login to phpMyAdmin  6. Check if case 2 is activated | Status of case 2 should be activated |  |  |  |
| 4 | Activate Case (Lecturer) | User should be able to deactivate the cases | NA | 1. Login to NPSTAFF account 2. Click on case 2 3. Click "deactivate case" button on the reveal modal | User should be able to deactivate case with the success message |  |  |  |
| 5 | Activate Case (Lecturer) | User should be able to deactivate the cases | NA | 1. Login to NPSTAFF account 2. Click on a case 2 3. Click "deactivate case" button on the reveal modal  4. Login to phpMyAdmin  5. Check if case 2 is deactivated | Status of case 2 should be deactivated |  |  |  |
| 6 | Activate Case (Lecturer) | User should be able to activate the cases | NA | 1. Login to NPSTAFF account 2. Click on a case 2 3. Click "activate case" button on the reveal modal  4. Login to phpMyAdmin  5. Check if case 2 is activated | Status of case 2 should be activated |  |  |  |
| 7 | Activate Case (Lecturer) | Lecturer portal should be in sync with the Admin portal | NA | 1. Login to NPSTAFF account 2. Select "Deactivate case" button on a case 2  3. Login to Admin account  4. Check if case 2 is deactivated | Case 2 should be deactivated in Admin portal |  |  |  |
| 8 | Activate Case (Lecturer) | Lecturer portal should be in sync with the Admin portal | NA | 1. Login to NPSTAFF account 2. Select “activate case" button on a case 2  3. Login to Admin account  4. Check if case 2 is activated | Case 2 should be activated in Admin portal |  |  |  |
| 9 | Activate Case (Lecturer) | Validate that only one case is activated at each time | NA | 1. Login to NPSTAFF account 2. Activate case 1  3. Activate Case 2 | Case 1 should be deactivated when case 2 is activated |  |  |  |

## Activate Case Scenario (Admin)

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| S/N | Functionality | Description | Test Inputs | Test Procedure | Expected Results | Actual Results | Pass/ Fail | Remarks |
| 1 | Activate Case (Admin) | User should be able to activate the cases | NA | 1. Login as Admin 2. Click on "Case management" tab 3. Click on the "Activate" case button on Case 1 to activate case | Case 1 should be activated and status should be changed to "Activated" in green |  |  |  |
| 2 | Activate Case (Admin) | User should be able to activate the cases | NA | 1. Login to Admin account 2. Select "activate case" button on a case 2  3. Login to phpMyAdmin  4. Check if case 2 is activated | Status of case 2 should be activated |  |  |  |
| 3 | Activate Case (Admin) | User should be able to deactivate the cases | NA | 1. Login as Admin 2. Click on "Case management" tab 3. Click on the "Deactivate" case button on Case 1 to deactivate case | Case 1 should be activated and status should be changed to "Deactivated" in red |  |  |  |
| 4 | Activate Case (Admin) | User should be able to deactivate the cases | NA | 1. Login to Admin account 2. Select “Deactivate case" button on a case 2  3. Login to phpMyAdmin  4. Check if case 2 is deactivated | Status of case 2 should be deactivated |  |  |  |
| 5 | Activate Case (Admin) | Lecturer portal should be in sync with the Admin portal | NA | 1. Login to Admin account 2. Select “Deactivate case" button on a case 2  3. Login to NPSTAFF account  4. Check if case 2 is deactivated | Case 2 should be deactivated in NPSTAFF portal |  |  |  |
| 6 | Activate Case (Admin) | Lecturer portal should be in sync with the Admin portal | NA | 1. Login to Admin account 2. Select “Activate case" button on a case 2  3. Login to NPSTAFF account  4. Check if case 2 is Activated | Case 2 should be activated in NPSTAFF portal |  |  |  |
| 9 | Activate Case (Lecturer) | Validate that only one case is activated at each time | NA | 1. Login to admin account 2. Activate case 1  3. Activate Case 2 | Case 1 and all other cases should be deactivated when case 2 is activated |  |  |  |

# View Patient’s Personal Information

## View Patient’s Personal Information (Student)

***Note:***To ensure that each case scenario would display the correct patient information, test cases 4, 5, 6 is a replication of test cases 1, 2, 3 with different sets of test inputs.

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| S/N | Functionality | Description | Test Inputs | Test Procedure | Expected Results | Actual Results | Pass/ Fail | Remarks |
| 1 | View Patient's information | Validate that user can view patient's basic information at the top of each tab | 1. **Name:** James Tan 2. **NRIC**: S9999999B | 1. Add new patient in database 2. Add new scenario in database and assign the new patient to the scenario 3. Log in to nurse account  4. Patient's information should be shown at the top of viewPatientInformaiton.jsp | Patient information which has just been created is shown on viewPatientInformation.jsp |  |  |  |
| 2 | View Patient's information | Validate that patient's information matches the case | 1. **Name**: James Tan 2. **NRIC**: S9999999B | 1. Check the patient NRIC shown on page viewPatientInformation.jsp and compare it with the database | NRIC of patient on the page should be the same as the case's patient's NRIC |  |  |  |
| 3 | View Patient's information | Validate that patient's allergy is correct | 1. **Name**: James Tan 2. **NRIC**: S9999999B  3. **Allergy**: Bactrim | 1. Check the patient's allergy at the patient's information at the top of each page | Patient’s allergy should be the same as what is in the database |  |  |  |
| 4 | View Patient's information | Validate that user can view patient's basic information at the top of each tab | 1. **Name**: Nicole Tan 2. **NRIC**: S1234567B | 1. Add new patient in database 2. Add new scenario in database and assign the new patient to the scenario 3. Log in to nurse account  4. Patient's information should be shown at the top of viewPatientInformaiton.jsp | Patient informsation which has just been created is shown on viewPatientInformation.jsp |  |  |  |
| 5 | View Patient's information | Validate that patient's information matches the case | 1. **Name**: Nicole Tan 2. **NRIC**: S1234567B | 1. Check the patient NRIC shown on page viewPatientInformation.jsp and compare it with the database | NRIC of patient on the page should be the same as the case's patient's NRIC |  |  |  |
| 6 | View Patient's information | Validate that patient's allergy is correct | 1. **Name**: Nicole Tan 2. **NRIC**: S1234567B 3. **Allergy**: Seafood | 1. Check the patient's allergy at the patient's information at the top of each page | Patient’s allergy should be the same as what is in the database |  |  |  |