## Back Office Client

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test ID** | 1 | | | |
| **Test Name** | Back Office Login | | | |
| **Test Description** | Tests the Back Office application’s login form validity. | | | |
| **Input** | | **System status** | **Oracle** | **Output** |
| The user enters correct username and password | | The server is reachable | The application shows the main screen of the application |  |
| The user enters a wrong combination of username and/or password | | The server is reachable | The application shows an error message telling that he entered bad credentials |  |
| The user leaves a blank username and/or password field | | The server is reachable | The application shows an error message telling that one of the two fields is empty |  |
| The user doesn’t leave empty fields | | The server is not reachable | The application shows an error message telling that the server is not available |  |
| The user presses the “Close window” button | |  | The application closes |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test ID** | 2 | | | |
| **Test Name** | Payment Order Management | | | |
| **Test Description** | Tests the validity of the screen that lets the user manage the payment orders. | | | |
| **Input** | | **System status** | **Oracle** | **Output** |
| The user clicks on a row whose status is “NOTIFIED” | |  | Only “Create”, “Save as suspended” and “Save as paid” button are enabled |  |
| The user clicks on a row whose status is “NOTISSUED” | |  | Only “Create”, “Issue” and “Delete” button are enabled |  |
| The user clicks on a row whose status is “SUSPENDED” | |  | Only “Create”, “Reissue” and “Save as not pertinent” button are enabled |  |
| The user presses “Save as suspended” button and confirms the operation | | The server is reachable and the operation is successful | The status of the previously selected row changes to “SUSPENDED” and all buttons (“Create” excluded) are disabled |  |
| The user presses “Save as paid” button and confirms the operation | | The server is reachable and the operation is successful | The previously selected row is removed and all buttons (“Create” excluded) are disabled |  |
| The user presses “Issue” button and confirms the operation | | The server is reachable and the operation is successful | The status of the previously selected row changes to “ISSUED” and all buttons (“Create” excluded) are disabled |  |
| The user presses “Delete” button and confirms the operation | | The server is reachable and the operation is successful | The previously selected row is removed and all buttons (“Create” excluded) are disabled |  |
| The user presses “Reissue” button and confirms the operation | | The server is reachable and the operation is successful | The status of the previously selected row changes to “ISSUED” and all buttons (“Create” excluded) are disabled |  |
| The user fills some filtering fields | |  | Only the rows that satisfy the filtering criteria are left shown in the table |  |
| The user clicks on a operation button(“Create” excluded) and confirms it | | The server is not reachable | The application shows an error message |  |
| The user clicks on a operation button(“Create” excluded) and confirms it | | The server is reachable but the access token has expired | The application shows an error message and then the login screen |  |
| The user presses the “Close window” button | |  | The application shows the main screen |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test ID** | 3 | | | |
| **Test Name** | Bill form | | | |
| **Test Description** | Tests the frame where the user can choose a bill to create a payment order | | | |
| **Input** | | **System status** | **Oracle** | **Output** |
| The user selects a row from the table | |  | The “Create payment order” button is enabled |  |
| The user presses “Create payment order” button | | The server is not reachable | The application shows an error message |  |
| The user presses “Create payment order” button | | The server is reachable and the server-side operation is successful | The application shows the Payment Order management frame with the new payment order |  |
| The user presses “Create payment order” button | | The server is reachable but the server-side operation is not successful | The application shows an error message |  |
| The user presses “Create payment order” button | | The server is reachable but the access token has expired | The application shows an error message and then shows the login screen |  |
| The user presses the “Close window” button | |  | The application shows the Payment Order management frame |  |

## Mobile Application

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test ID** | 4 | | | |
| **Test Name** | Readings Operator Login | | | |
| **Test Description** | Tests the Readings Operator mobile application’s login form validity | | | |
| **Input** | | **System status** | **Oracle** | **Output** |
| The user enters correct operator id and password | | The server is reachable | The application shows the main screen of the application |  |
| The user enters a wrong combination of operator id and/or password | | The server is reachable | The application shows an error message telling that he entered bad credentials |  |
| The user leaves a blank operator id and/or password field | |  | “Login” button is disabled |  |
| The user doesn’t leave empty fields | | The server is not reachable | The application shows an error message telling that the server is not available |  |
| The user presses the back button | |  | The application ends |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test ID** | 5 | | | |
| **Test Name** | Readings Operator application | | | |
| **Test Description** | Tests the behaviour of the mobile application for Readings Operators | | | |
| **Input** | | **System status** | **Oracle** | **Output** |
| The user selects an assignment from the table | |  | The “Save reading” button is enabled |  |
| The user presses the menu button and then “Logout” button, and confirms the operation | |  | The application shows the login screen. |  |
| The user presses “Update” button | | The server is reachable | The application shows in the table the assignment of the operator, except the ones already completed but not sended |  |
| User presses “Save reading” button, enters the consumption and presses “Save” | |  | The assignment previously selected is removed from the table and the “Send readings” button is enabled |  |
| User presses “Send readings” button | | The server is reachable and the server-side operation is successful | “Send readings” button is disabled |  |
| The user presses “Send readings” or “Update” button | | The server is not reachable | The application shows an error message |  |
| The user presses “Send readings” or “Update” button | | The server is reachable but the server-side operation fails | The application shows an error message |  |
| The user presses “Send readings” or “Update” button | | The server is reachable but the access token has expired | The application shows an error message and then the login screen |  |
| The user presses back button | |  | The application goes in background |  |

## Unit testing

|  |  |  |  |
| --- | --- | --- | --- |
| **Parameters domain** | **Description** | **Equivalence classes** | **Values** |
| A |  | A1 | Not null |
|  |  | A2 | Null |
| B |  | B1 | Collections of Reading instances |
|  |  | B2 | Missing parameter in request |
|  |  | B3 | Well-formed JSON strings but not a collection of Reading instances |
|  |  | B4 | Malformed JSON strings |
| C |  | C1 | TRUE |
|  |  | C2 | FALSE |

A = Session  
A1 = { not null }  
A2 = { null }

B = Request parameter “readings”  
B1 = { a collection of Reading instances }  
B2 = {null} (not in parameters list)  
B3 = { not a collection of Reading instances } (well-formed json but not a collection of instances of Reading class)  
B4 = { malformed } (not a json well-formed string }

C = DAO method return //TODO  
C1 = { TRUE }  
C2 1= { FALSE }

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test case** | **A** | **B** | **C** | **Response code** |
| SE 1 | A1 | B1 | C1 | 200 |
| SE 2 | A1 | B1 | C2 | 500 |
| SE 3 | A1 | B2 | C1 | 463 |
| SE 4 | A1 | B2 | C2 | 463 |
| SE 5 | A1 | B3 | C1 | 464 |
| SE 6 | A1 | B3 | C2 | 464 |
| SE 7 | A1 | B4 | C1 | 464 |
| SE 8 | A1 | B4 | C2 | 464 |
| SE 9 | A2 | B1 | C1 | 461 |
| SE 10 | A2 | B1 | C2 | 461 |
| SE 11 | A2 | B2 | C1 | 461 |
| SE 12 | A2 | B2 | C2 | 461 |
| SE 13 | A2 | B3 | C1 | 461 |
| SE 14 | A2 | B3 | C2 | 461 |
| SE 15 | A2 | B4 | C1 | 461 |
| SE 16 | A2 | B4 | C2 | 461 |