ADES Test Cases (Website & Mobile)

# Website Test Cases - Success

|  |  |
| --- | --- |
| <Test Case #1> - <1 Live Queue Website Success Process> | |
| Description | Test case for success for the user to view arrival rate of one queue using the chart provided (queue01001) |
| Pre-Condition | Created tables for backend (npm run populate-website)  Backend server for customer working (npm run start-win-customer)  Backend server for company working (npm run start-win-company)  inserting live data for queue01001(npm run single-peak) |
| Test Steps | 1. Press “add another” to add a monitor 2. insert company id = 1000000001 3. press the “search” button 4. select queue01001 from the dropdown box 5. wait for 3 seconds to see that the chart is refreshing with live data |
| Expected Result | Chart for queue01001 will show up with the data being updated every 3 seconds |

# 

|  |  |
| --- | --- |
| <Test Case #2> - <Multiple Live queue Website Success Process> | |
| Description | Test case for success for the user to view arrival rate of multiple queue using the chart provided (queue01001 and queue01002) |
| Pre-Condition | Created tables for backend (npm run populate-website)  Backend server for customer working (npm run start-win-customer)  Backend server for company working (npm run start-win-company)  inserting live data for queue01001(npm run single-peak)  inserting live data for queue01002(npm run double-peak) |
| Test Steps | 1. Press “add another” to add a monitor 2. insert company id = 1000000001 3. press the “search” button 4. select queue01001 from the dropdown box 5. wait for 3 seconds to see that the chart is refreshing with live data 6. Press “add another” to add another monitor 7. insert company id = 1000000001 8. press the “search” button 9. select queue01002 from the dropdown box 10. wait for 3 seconds to see that the chart is refreshing with live data |
| Expected Result | Chart for queue01001 and queue01002 will show up with the data being updated every 3 seconds |

# 

# 

|  |  |
| --- | --- |
| <Test Case #3> - <swapping queues in one monitor Website Success Process> | |
| Description | Test case for success for the user to swap the queue using one monitor |
| Pre-Condition | Created tables for backend (npm run populate-website)  Backend server for customer working (npm run start-win-customer)  Backend server for company working (npm run start-win-company)  inserting live data for queue01001(npm run single-peak)  inserting live data for queue01002(npm run double-peak) |
| Test Steps | 1. Press “add another” to add a monitor 2. insert company id = 1000000001 3. press the “search” button 4. select queue01001 from the dropdown box 5. wait for 3 seconds to see that the chart is refreshing with live data 6. select queue01002 from the same dropdown box 7. wait for 3 seconds to see that the chart is refreshing with live data |
| Expected Result | Chart for queue01001 and queue01002 will show up with the data being updated every 3 seconds |

# 

# 

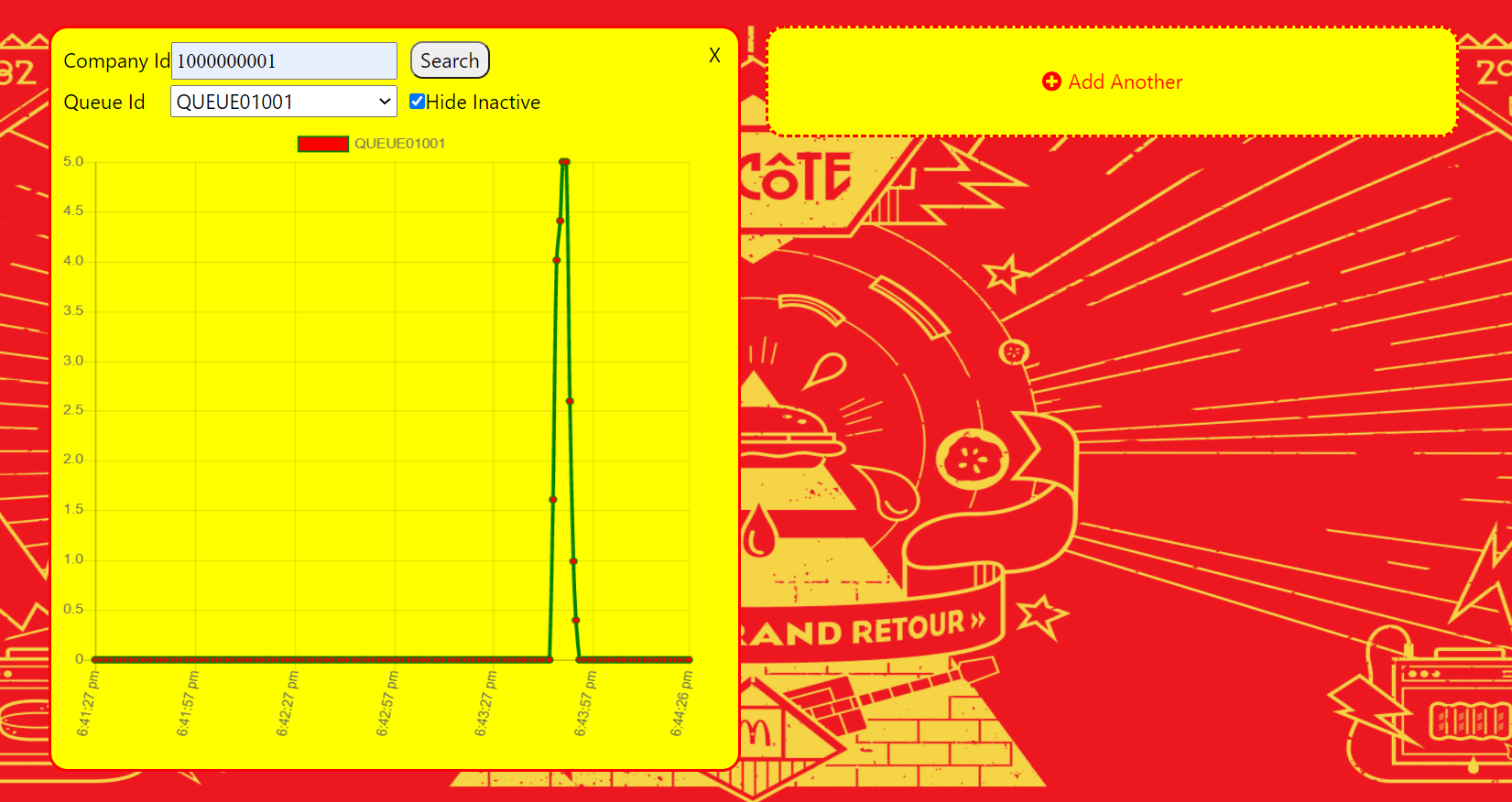
|  |  |
| --- | --- |
| <Test Case #4> - <Hide invalid button works> | |
| Description | Test case for success for when user checks the hide inactive check box |
| Pre-Condition | Created tables for backend (npm run populate-website)  Backend server for customer working (npm run start-win-customer)  Backend server for company working (npm run start-win-company)  inserting live data for queue01001(npm run single-peak)  inserting live data for queue01002(npm run double-peak) |
| Test Steps | 1. Input a company id = 1000000001 2. Press Search button 3. Check Hide Inactive |
| Expected Result | Queue id would show inactive queue as disabled options |

# 

# 

|  |  |
| --- | --- |
| <Test Case #5> - <close monitor Website Success Process> | |
| Description | Test case for success for the user to close a monitor while the queue is running |
| Pre-Condition | Created tables for backend (npm run populate-website)  Backend server for customer working (npm run start-win-customer)  Backend server for company working (npm run start-win-company)  inserting live data for queue01001(npm run single-peak)  inserting live data for queue01002(npm run double-peak) |
| Test Steps | 1. Press “add another” to add a monitor 2. insert company id = 1000000001 3. press the “search” button 4. select queue01001 from the dropdown box 5. wait for 3 seconds to see that the chart is refreshing with live data 6. Press “add another” to add another monitor 7. insert company id = 1000000001 8. press the “search” button 9. select queue01002 from the dropdown box 10. wait for 3 seconds to see that the chart is refreshing with live data 11. press the X button to close a monitor |
| Expected Result | There should be only one monitor left opened |

# 



# Website Test Cases - Errors

# 

|  |  |
| --- | --- |
| <Test Case #6> - <Unknown company id Website Failure Process> | |
| Description | Test case for failure for unknown queue id |
| Pre-Condition | Created tables for backend (npm run populate-website)  Backend server for customer working (npm run start-win-customer)  Backend server for company working (npm run start-win-company) |
| Test Steps | 1. Press “add another” to add a monitor 2. insert company id = 1000000010 3. press the “search” button |
| Expected Result | Alert for unknown company ID will appear |

# 

# 

|  |  |
| --- | --- |
| <Test Case #7> - <invalid customer id Website Failure Process> | |
| Description | Test case for failure for unknown queue id |
| Pre-Condition | Created tables for backend (npm run populate-website)  Backend server for customer working (npm run start-win-customer)  Backend server for company working (npm run start-win-company) |
| Test Steps | 1. Press “add another” to add a monitor 2. insert company id = 100000001 3. press the “search” button |
| Expected Result | Alert for invalid company ID will appear |

# 

# 

|  |  |
| --- | --- |
| <Test Case #8> - <Failed to fetch company Website Failure Process> | |
| Description | Test case for failure to fetch company data |
| Pre-Condition | Created tables for backend (npm run populate-website)  backend have not started |
| Test Steps | 1. Press “add another” to add a monitor 2. insert company id = 1000000001 3. press the “search” button |
| Expected Result | Alert for failing to fetch company ID will appear |

# 

# 

|  |  |
| --- | --- |
| <Test Case #9> - <failed to fetch queue data Website Failure Process> | |
| Description | Test case for failure to fetch queue data while it is refreshing |
| Pre-Condition | Created tables for backend (npm run populate-website)  Backend server for customer working (npm run start-win-customer)  Backend server for company working (npm run start-win-company)  inserting live data for queue01001(npm run single-peak) |
| Test Steps | 1. Press “add another” to add a monitor 2. insert company id = 1000000001 3. press the “search” button 4. select queue01001 from the dropdown box 5. Stop the backend from running through terminal |
| Expected Result | Error icon will appear for the graph |

# 

# Mobile Test Cases - Success

|  |  |
| --- | --- |
| **<Test Case #10> - <Check Queue Success Process>** | |
| Description | Test case for success when user checks a queue |
| Pre-Condition | Created tables for backend (npm run populate-website)  Backend server for customer working (npm run start-win-customer)  Backend server for company working (npm run start-win-company) |
| Test Steps | 1. Input customer id = 1234567890 (Optional) 2. Input queue id = QUEUE01001 3. Press check button once user id and queue id is inputted 4. If customer id is provided, join queue button would show |
| Expected Result | User is able to check queue’s total amount of people |

# 

|  |  |
| --- | --- |
| **<Test Case #11> - <Join Queue Success Process>** | |
| Description | Test case for success when user join queue if they’re not in the queue |
| Pre-Condition | Created tables for backend (npm run populate-website)  Backend server for customer working (npm run start-win-customer)  Backend server for company working (npm run start-win-company) |
| Test Steps | 1. Input customer id = 1234567890 2. Input queue id = QUEUE01001 3. Press check button once user id and queue id is inputted 4. Total should show below the check button, showing the total number of people in the queue, and the join queue button would show if user is not in that queue 5. Press the join queue button 6. Loading animation would show and an alert would show once the user have successfully joined the queue 7. Press Check button again, total and ahead would then show up without the join queue button as user is already in the queue which does not allow user to join again |
| Expected Result | User is able to join queue and an alert would pop-up when customer joined the queue |

# 

# Mobile Test Cases - Errors

|  |  |
| --- | --- |
| **<Test Case #12> - <Invalid Queue Entered>** | |
| Description | Test case for error when user inputs a invalid queue id |
| Pre-Condition | Created tables for backend (npm run populate-website)  Backend server for customer working (npm run start-win-customer)  Backend server for company working (npm run start-win-company) |
| Test Steps | 1. Input a invalid queue id = ABCD 2. Press Check button 3. Error message would show |
| Expected Result | Error Message showing invalid queue id |

# 

# 

|  |  |
| --- | --- |
| **<Test Case #13> - < Queue Not Found>** | |
| Description | Test case for error when user inputs a queue which is not in the database |
| Pre-Condition | Created tables for backend (npm run populate-website)  Backend server for customer working (npm run start-win-customer)  Backend server for company working (npm run start-win-company) |
| Test Steps | 1. Input a invalid queue id = ABCDE12345 2. Press Check button 3. Error message would show |
| Expected Result | Error Message showing that queue is not found |

# 

# 

|  |  |
| --- | --- |
| **<Test Case #14> - < Queue Not Entered>** | |
| Description | Test case for error when user does not input queue id |
| Pre-Condition | Created tables for backend (npm run populate-website)  Backend server for customer working (npm run start-win-customer)  Backend server for company working (npm run start-win-company) |
| Test Steps | 1. Press Check button without inputting any value 2. Error message would show |
| Expected Result | Error Message showing “Please enter queue id” |

# 

# 

|  |  |
| --- | --- |
| **<Test Case #15> - <Invalid customer id Entered>** | |
| Description | Test case for error when user inputs a invalid customer id |
| Pre-Condition | Created tables for backend (npm run populate-website)  Backend server for customer working (npm run start-win-customer)  Backend server for company working (npm run start-win-company) |
| Test Steps | 1. Input a invalid customer id = ABCDEF 2. Press the check button 3. Error message would show |
| Expected Result | Error Message showing “Invalid Customer id” |

# 