Consider the sprint task #11 – Develop GUI to show the details about the agents in each node in graphical view

Some of the test cases for this task are as follows:

|  |  |  |  |
| --- | --- | --- | --- |
| Test case #sprint 11 | Scenario | Inputs | Expect Output |
| 1 | Single click on a node which has been visited | Mouse click | All agents which have visited this node is shown on a board. |
| 2 | Single click on a node which has not been visited | Mouse click | None |
| 3 | double click on a node which has been visited | Mouse click | All agents which have visited this node is shown on a board. |
| 4 | double click on a node which has not been visited | Mouse click | None |
| 5 | Mouse hovering on a node which has been visited | None | None |
| 6 | Mouse hovering on a node which has not been visited | None | None |
| 7 | Single click or double click on a particular node which has been visited | Mouse click | Show the correct corresponding detail information of that particular node |

Consider the sprint task #20 – Develop DB access code for storing and retrieving run information

Some of the test cases for this task are as follows:

|  |  |  |  |
| --- | --- | --- | --- |
| Test case #sprint 20 | Scenario | Inputs | Expect Output |
| 1 | Save a run information | JSON data of a run | Server return message that save successful |
| 2 | Retrieve a history run using start date and end date | Start date and end date | Matched history run is retrieved from the table |
| 3 | Retrieve a history run just using start date | Start date | Server return error message that must include end date |
| 4 | Retrieve a history run just using start date | End date | Server return error message that must include start date |
| 5 | Retrieve a history run using description | Description | Matched history run is retrieved from the table |
| 6 | Retrieve history runs between two timestamps | 1488592090200 and 1489024090200 | All history runs are retrieved whose ID is between 1488592090200 and 1489024090200 |
|  |  |  |  |

Consider the sprint task #21 – Develop GUI to show all run information

Some of the test cases for this task are as follows:

|  |  |  |  |
| --- | --- | --- | --- |
| Test case #sprint 21 | Scenario | Inputs | Expect Output |
| 1 | Display all history runs | None | All history runs are displayed on the screen |
| 2 | Display all history runs; there is no history run in DB | None | Nothing will be displayed on the screen |

Consider the sprint task #22 – Develop GUI to filter history runs

Some of the test cases for this task are as follows:

|  |  |  |  |
| --- | --- | --- | --- |
| Test case #sprint 22 | Scenario | Inputs | Expect Output |
| 1 | Select the start date and end date | Mouse Click | The time selector is shown |
| 2 | Filter the history runs after selecting start and end date | Mouse click on filter button (Selected start date is Feb 27 2017 20:00:00, end date is Feb 27 2017 20:00:00) | All history runs are displayed on the screen |
| 3 | Filter the history runs before selecting start and end date | Mouse click on filter button | Message showing that user must select start and end date before clicking the filter button |
| 4 | Filter the history runs after selecting start and end date; the start date is after the end date | Mouse click on filter button (Selected start date is Mar 5 2017 20:00:00, end date is Feb 27 2017 20:00:00) | Message showing that the start date must be before the end date |