Smart Commute: Test Report 11 mar 2019

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Project Name: Smart Commute**  Report Test 1: Event creation | | | | | | |
| Module Name: | | Calendar Engine / event creation – script: app.js | | | | |
| Module description: | | The event creation is comprised of several functions from the moment the user triggers the event submission (functions called in the same order): | | | | |
| Type of Test: | | End to end test | | | | |
| Test Software: | | Google Chrome Console | | | | |
| Test Description: | | Tests the event creation from a user request | | | | |
| Test Date: | | 09 March 2019 | | | | |
|  | | | | | | |
| Test Scenario: | | * The user is logged in * The user clicks a day in the month * Fills the event popup form * Clicks submit event | | | | |
|  | | | | | | |
| Test case 1: submit event with null title | | | | | | |
| Pre-Conditions: | | * User must be logged in * User must click a day from the calendar month in month view from Calendar GUI | | | | |
| **Step #** | **Step Details** | **Test Data** | **Expected Result** | **Actual Result** | **Status (Pass/Fail)** | **Notes** |
| 1 | Provide an event title | <empty> | Title can be entered | Expected | Pass |  |
| 2 | Provide an event location | “Piazza Leonardo Da Vinci, 32” | Location can be entered | Expected | Pass | Location should match a result from the here maps addresses database |
| 3 | Provide start time | 13:00 | Start time can be entered | Expected | Pass |  |
| 4 | Provide end time | 15:00 | End time can be entered | Expected | Pass |  |
| 5 | User clicks event submit |  | Alert: please insert a title | Alert: please insert a title | Pass |  |
| Post-Condition | | App retains the popup open waiting for correct user input | | | | |
|  | | | | | | |
| Test case 2: submit event with incorrect start and end times | | | | | | |
| Pre-Conditions: | | * User must be logged in * User must click a day from the calendar month in month view from Calendar GUI | | | | |
| **Step #** | **Step Details** | **Test Data** | **Expected Result** | **Actual Result** | **Status (Pass/Fail)** | **Notes** |
| 1 | Provide event title | “Meeting with professor” | Title can be entered | Expected | Pass |  |
| 2 | Provide an event location | “Piazza Leonardo Da Vinci, 32” | Location can be entered | Expected | Pass | Location should match a result from the here maps addresses database |
| 3 | Provide start time | 14:00 | Start time can be entered | Expected | Pass |  |
| 4 | Provide end time | 12:00 | End time can be entered | Expected | Pass |  |
| 5 | User clicks event submit |  | Alert: event times are invalid. Please correct | Alert: event times are invalid. Please correct | Pass |  |
| Post-condition | | App retains the popup open waiting for correct user input | | | | |
|  | | | | | | |
| Test case 3: user submits event successfully | | | | | | |
| Pre-Conditions: | | * User must be logged in * User must click a day from the calendar month in month view from Calendar GUI | | | | |
| **Step #** | **Step Details** | **Test Data** | **Expected Result** | **Actual Result** | **Status (Pass/Fail)** | **Notes** |
| 1 | Provide event title | “Meeting with professor” | Title can be entered | Expected | Pass |  |
| 2 | Provide an event location | “Piazza Leonardo Da Vinci, 32” | Location can be entered | Expected | Pass | Location should match a result from the here maps addresses database |
| 3 | Provide start time | 13:00 | Start time can be entered | Expected | Pass |  |
| 4 | Provide end time | 14:30 | End time can be entered | Expected | Pass |  |
| 5 | User clicks event submit |  | Event is submitted | Event is submitted | Pass | Event is rendered onto the calendar and saved in Firebase DB |
| Post-condition | | Popup closes and shows event on the calendar GUI | | | | |
|  | | | | | | |
| Test case 4: user submits a second event which overlaps with the first event’s duration | | | | | | |
| Pre-Conditions: | | * User must be logged in * User has successfully created an event (test case 3) * User must click a day from the calendar month in month view from Calendar GUI | | | | |
| **Step #** | **Step Details** | **Test Data** | **Expected Result** | **Actual Result** | **Status (Pass/Fail)** | **Notes** |
| 1 | Provide event title | “Shopping in C. Buenos Aires” | Title can be entered | Expected | Pass |  |
| 2 | Provide an event location | “Corso Buenos Aires, Milano” | Location can be entered | Expected | Pass | Location should match a result from the here maps addresses database |
| 3 | Provide start time | 14:00 | Start time can be entered | Expected | Pass |  |
| 4 | Provide end time | 16:00 | End time can be entered | Expected | Pass |  |
| 5 | User clicks event submit |  | Alert: one or more events overlapping. Try changing the time. | Alert: one or more events overlapping. Try changing the time. | Pass | Event time is compared with all event times and event created (test case 3) overlaps with this one. |
| Post-condition | | App retains the popup open waiting for adequate time input | | | | |
|  | | | | | | |
| Test case 5: user submits a second event whose travel time is larger than the time available | | | | | | |
| Pre-Conditions: | | * User must be logged in * User has successfully created an event (test case 3) * User must click a day from the calendar month in month view from Calendar GUI | | | | |
| **Step #** | **Step Details** | **Test Data** | **Expected Result** | **Actual Result** | **Status (Pass/Fail)** | **Notes** |
| 1 | Provide event title | “Friends reunion in Eindhoven” | Title can be entered | Expected | Pass |  |
| 2 | Provide an event location | “Eindhoven, Netherlands” | Location can be entered | Expected | Pass | Location should match a result from the here maps addresses database |
| 3 | Provide start time | 15:30 | Start time can be entered | Expected | Pass |  |
| 4 | Provide end time | 19:00 | End time can be entered | Expected | Pass |  |
| 5 | User clicks event submit |  | Alert: you’re either short on time to get there or you’d be unable to get to the following appointment! | Alert: you’re either short on time to get there or you’d be unable to get to the following appointment! | Pass | Event travel time is compared with time gap between the end of first event (test case 3) and its start time and results in a lack of time to travel. |
| Post-condition | | App retains the popup open waiting for adequate input | | | | |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Project Name: Smart Commute**  Report Test 2: Event modification | | | | | | |
| Module Name: | | Calendar Engine / event modification – script: app.js | | | | |
| Module description: | | The event modification is comprised of several functions from the moment the user triggers the event update (functions called in the same order): | | | | |
| Type of Test: | | End to end test | | | | |
| Test Software: | | Google Chrome Console | | | | |
| Test Description: | | Tests the event modification from a user request | | | | |
| Test Date: | | 09 March 2019 | | | | |
|  | | | | | | |
| Test Scenario: | | * The user is logged in * The user clicks an existing event previously created (during current session or previous sessions) * Fills the event popup form * Clicks update event | | | | |
|  | | | | | | |
| Test case 1: update event with null title | | | | | | |
| Pre-Conditions: | | * User must be logged in * User must click a day from the calendar month in month view from Calendar GUI | | | | |
| **Step #** | **Step Details** | **Test Data** | **Expected Result** | **Actual Result** | **Status (Pass/Fail)** | **Notes** |
| 1 | Provide an event title | <empty> | Title can be entered | Expected | Pass |  |
| 2 | Provide an event location | “Piazza Leonardo Da Vinci, 32” | Location can be entered | Expected | Pass | Location should match a result from the here maps addresses database |
| 3 | Provide start time | 13:00 | Start time can be entered | Expected | Pass |  |
| 4 | Provide end time | 15:00 | End time can be entered | Expected | Pass |  |
| 5 | User clicks update event |  | Alert: please insert a title | Alert: please insert a title | Pass |  |
| Post-Condition | | App retains the popup open waiting for correct user input | | | | |
|  | | | | | | |
| Test case 2: update event with incorrect start and end times | | | | | | |
| Pre-Conditions: | | * User must be logged in * User must click a day from the calendar month in month view from Calendar GUI | | | | |
| **Step #** | **Step Details** | **Test Data** | **Expected Result** | **Actual Result** | **Status (Pass/Fail)** | **Notes** |
| 1 | Provide event title | “Meeting with professor” | Title can be entered | Expected | Pass |  |
| 2 | Provide an event location | “Piazza Leonardo Da Vinci, 32” | Location can be entered | Expected | Pass | Location should match a result from the here maps addresses database |
| 3 | Provide start time | 14:00 | Start time can be entered | Expected | Pass |  |
| 4 | Provide end time | 12:00 | End time can be entered | Expected | Pass |  |
| 5 | User clicks update event |  | Alert: event times are invalid. Please correct | Alert: event times are invalid. Please correct | Pass |  |
| Post-condition | | App retains the popup open waiting for correct user input | | | | |
|  | | | | | | |
| Test case 3: user submits event successfully | | | | | | |
| Pre-Conditions: | | * User must be logged in * User must click a day from the calendar month in month view from Calendar GUI | | | | |
| **Step #** | **Step Details** | **Test Data** | **Expected Result** | **Actual Result** | **Status (Pass/Fail)** | **Notes** |
| 1 | Provide event title | “Meeting in B11” | Title can be entered | Expected | Pass |  |
| 2 | Provide an event location | “Piazza Leonardo Da Vinci, 32” | Location can be entered | Expected | Pass | Location should match a result from the here maps addresses database |
| 3 | Provide start time | 15:00 | Start time can be entered | Expected | Pass |  |
| 4 | Provide end time | 16:30 | End time can be entered | Expected | Pass |  |
| 5 | User clicks update event |  | Event is submitted | Event is submitted | Pass | Event is rendered onto the calendar and saved in Firebase DB |
| Post-condition | | Popup closes and shows event on the calendar GUI | | | | |
|  | | | | | | |
| Test case 4: user submits a second event which overlaps with the first event’s duration | | | | | | |
| Pre-Conditions: | | * User must be logged in * User has successfully created an event (test case 3) * User must click a day from the calendar month in month view from Calendar GUI | | | | |
| **Step #** | **Step Details** | **Test Data** | **Expected Result** | **Actual Result** | **Status (Pass/Fail)** | **Notes** |
| 1 | Provide event title | “Shopping in C. Buenos Aires” | Title can be entered | Expected | Pass |  |
| 2 | Provide an event location | “Corso Buenos Aires, Milano” | Location can be entered | Expected | Pass | Location should match a result from the here maps addresses database |
| 3 | Provide start time | 16:00 | Start time can be entered | Expected | Pass |  |
| 4 | Provide end time | 19:00 | End time can be entered | Expected | Pass |  |
| 5 | User clicks update event |  | Alert: one or more events overlapping. Try changing the time. | Alert: one or more events overlapping. Try changing the time. | Pass | Event time is compared with all event times and event created (test case 3) overlaps with this one. |
| Post-condition | | App retains the popup open waiting for adequate time input | | | | |
|  | | | | | | |
| Test case 5: user submits a second event whose travel time is larger than the time available | | | | | | |
| Pre-Conditions: | | * User must be logged in * User has successfully created an event (test case 3) * User must click a day from the calendar month in month view from Calendar GUI | | | | |
| **Step #** | **Step Details** | **Test Data** | **Expected Result** | **Actual Result** | **Status (Pass/Fail)** | **Notes** |
| 1 | Provide event title | “Friends reunion in Eindhoven” | Title can be entered | Expected | Pass |  |
| 2 | Provide an event location | “Eindhoven, Netherlands” | Location can be entered | Expected | Pass | Location should match a result from the here maps addresses database |
| 3 | Provide start time | 17:00 | Start time can be entered | Expected | Pass |  |
| 4 | Provide end time | 19:00 | End time can be entered | Expected | Pass |  |
| 5 | User clicks update event |  | Alert: you’re either short on time to get there or you’d be unable to get to the following appointment! | Alert: you’re either short on time to get there or you’d be unable to get to the following appointment! | Pass | Event travel time is compared with time gap between the end of first event (test case 3) and its start time and results in a lack of time to travel. |
| Post-condition | | App retains the popup open waiting for adequate input | | | | |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Project Name: Smart Commute**  Report Test 3: Event deletion | | | | | | |
| Module Name: | | Calendar Engine / event deletion – script: app.js | | | | |
| Module description: | | The event deletion is comprised of several functions from the moment the user triggers delete event (functions called in the same order):   * ; | | | | |
| Type of Test: | | End to end test | | | | |
| Test Software: | | Google Chrome Console | | | | |
| Test Description: | | Tests the event modification from a user request | | | | |
| Test Date: | | 09 March 2019 | | | | |
|  | | | | | | |
| Test Scenario: | | * The user is logged in * The user clicks an existing event previously created (during current session or previous sessions) * Clicks delete event | | | | |
|  | | | | | | |
| Test case 1: delete any preexisting event | | | | | | |
| Pre-Conditions: | | * User must be logged in * User must click a day from the calendar month in month view from Calendar GUI | | | | |
| **Step #** | **Step Details** | **Test Data** | **Expected Result** | **Actual Result** | **Status (Pass/Fail)** | **Notes** |
| 1 | Click “delete button” |  | Event is deleted | Event is deleted | Pass | Event disappears from the GUI and is deleted from the Database |
| Post-Condition | | App closes the popup and goes back to the calendar GUI | | | | |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Project Name: Smart Commute**  Report Test 4: Roadmap GUI | | | | | | |
| Module Name: | | Roadmap Engine – script: maps.js | | | | |
| Module description: | | The event creation is comprised of several functions from the moment the user triggers the event submission (functions called in the same order): | | | | |
| Type of Test: | | End to end test | | | | |
| Test Software: | | Google Chrome Console | | | | |
| Test Description: | | Tests the event and route rendering on the roadmap upon user access | | | | |
| Test Date: | | 09 March 2019 | | | | |
|  | | | | | | |
| Test Scenario: | | * The user is logged in * The user clicks the roadmap button * The events and routes are then sequentially rendered onto the map and the right-side daily calendar | | | | |
|  | | | | | | |
| Test case 1: User enters the roadmap | | | | | | |
| Pre-Conditions: | | * User must be logged in * User must click the button roadmap from the main page (calendar GUI) | | | | |
| **Step #** | **Step Details** | **Test Data** | **Expected Result** | **Actual Result** | **Status (Pass/Fail)** | **Notes** |
| 1 | Click roadmap in main GUI | <empty> | Switch to Roadmap GUI | Expected | Pass |  |
| Post-Condition | | Roadmap GUI is loaded with all events and routes rendered respectively. | | | | |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Project Name: Smart Commute**  Report Test 5: User changes settings | | | | | | |
| Module Name: | | Settings – script: settings.js | | | | |
| Module description: | | The event creation is comprised of several functions from the moment the user triggers the event submission (functions called in the same order): | | | | |
| Type of Test: | | End to end test | | | | |
| Test Software: | | Google Chrome Console | | | | |
| Test Description: | | Tests the change of settings by the user | | | | |
| Test Date: | | 09 March 2019 | | | | |
|  | | | | | | |
| Test Scenario: | | * The user is logged in * The user clicks the settings button * The user modifies the settings at will * The user clicks “save & exit” | | | | |
|  | | | | | | |
| Test case 1: User enters the roadmap | | | | | | |
| Pre-Conditions: | | * User must be logged in * User must click the button settings from the main page (calendar GUI) | | | | |
| **Step #** | **Step Details** | **Test Data** | **Expected Result** | **Actual Result** | **Status (Pass/Fail)** | **Notes** |
| 1 | Click settings on main GUI | <empty> | Switch to Settings GUI | Switch to Settings GUI | Pass |  |
| 2 | User selects preferred transport | “metro” | Dropdown can be accessed | Expected | Pass |  |
| 3 | User checks available transports | [true, true, true, true, true, true] (selects all transports as available) | Round checks can be used | Expected | Pass | Each element of the array in test data indicates whether the transport is checked or not, for the 6 types of transport. Available entries: true, false. |
| 4 | User checks lunch time | 13:00 – 14:00 | Lunch time can be entered | Expected | Pass |  |
| 5 | User checks break time | 17:15 – 17:30 | Break time can be entered | Expected | Pass |  |
| 6 | User clicks “save & exit” |  | App redirects back to calendar GUI | App redirects back to calendar GUI | Pass | Lunch and break times are not yet implemented and haven’t been saved in the Database. |
| Post-Condition | | Settings preloaded respectively onto the GUI and saved in the Database | | | | |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Report Test of Smart Commute Web App  Report test 5: settings unit test | | | | | | |
| Module Test: | | Settings.js | | | | |
| Module description: | | Contains the main functionalities of the settings and is responsible for the settings interface in overall | | | | |
| Type of Test: | | Unit test | | | | |
| Test Software: | | Google Chrome developer tools console | | | | |
| Test Date: | | 07 March 2019 | | | | |
| Test Set: | | 1. New user, no settings established: 2. Available transports: null 3. Preferred transport: null 4. Existing user, previous settings defined: 5. Available transports: false, false, true, true, true, false 6. Preferred transport: bike | | | | |
| No. Test | Function Name | Description / Context | Input | Output | Expected Output | Pass/ fail test |
| 01 |  | Retrieves the user ID to connect to firebase database and gets the user settings object. Context: |  | : String | : String | Passed |
| 02 |  | Context: new user, first settings configuration (test set 1).  Firebase DB user settings data object is null, then creates a static default settings value. | “data” = | *{“availableTransports”: [false, true, true, true, true, true],*  *“preferredTransport”: ‘bus’};* | *{“availableTransports”: [false, true, true, true, true, true],*  *“preferredTransport”: ‘bus’};* | Passed |
| 03 |  | Context: existing user has configured some settings in a previous session (test set 2).  Firebase DB user settings data object is not null and contains previously saved settings. | “data” = firebase object | *{“availableTransports”: [false, false, true, true, true, false],*  *“preferredTransport”: ‘bike’};* | *{“availableTransports”: [false, false, true, true, true, false],*  *“preferredTransport”: ‘bike’};* | Passed |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Report Test of Smart Commute Web App  Report test 6: login/logout unit test | | | | | | | |
| Module Test: | | | login.js | | | | |
| Module description: | | | Contains the functions that allow to login or sign up Smart Commute clients | | | | |
| Type of Test: | | | Unit test | | | | |
| Test Software: | | | Google Chrome developer tools console | | | | |
| Test Date: | | | 07 March 2019 | | | | |
| Test set: | | | * 1. (correct password, incorrect username)   2. (Incorrect password, incorrect username)   3. (Incorrect password, correct username)   4. (Correct password, correct username) | | | | |
| No. Test | Function Name | | Description / Context | Input | Output | Expected Output | Pass/ fail test |
| 01 |  | | Initialization of firebase authentication service. Start with loading of index.html |  |  |  | Passed |
| 02 |  | | Returns true if user successfully signs in | (test sets 1, 2 or 3) | False | False | Passed |
| 02 |  | Returns true if user successfully signs in | | Test set 4 | True | True | Passed |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Report Test of Smart Commute Web App  Report test 7: event management unit test | | | | | | |
| Module Test: | | app.js | | | | |
| Module description: | | Contains the functions that allow to validate an event addition or update | | | | |
| Type of Test: | | Unit test | | | | |
| Test Software: | | Google Chrome developer tools console | | | | |
| Test Date: | | 07 March 2019 | | | | |
| Test set: | | 1. New user, no events created:   1. Day clicked: 04 March 2019 2. Title: my event. Start: 13:00. End: 15:00. Location: Via Ampere 18. Description: trial of event. 3. Existing user, one event created:   (title: last event. Start: 07 March 2019, 12:00. End: 07 March 2019, 14:00. Location: Piazza Leonardo Da Vinci, 32. Description: null)   1. Day clicked: 07 March 2019 2. Title: my event. Start: 09:00. End: | | | | |
| No. Test | Function Name | Description / Context | Input | Output | Expected Output | Pass/ fail test |
| 01 |  | Context: user clicks a day in the month, enters title: my event / start time: 13:00 / end time: 15:00 / description: trial of event  Description: the start and end dates are parsed from the calendar method | Start: 04-03-2019  End: 04-03-2019 |  |  | Passed |
| 02 |  | Rearranges the event nodes (data elements with keys of start time, end time and location) in recency order |  |  |  | Passed |
| 03 |  | Generates the route parameters to input into the function that provides the route objects |  |  |  |  |
| 04 |  |  |  |  |  |  |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Report Test of Smart Commute Web App  Report test 8: roadmap unit test | | | | | | |
| Module Test: | | maps.js | | | | |
| Module description: | | Contains the functions that permit the map initialization, events and routes rendering onto the map and day calendar. | | | | |
| Type of Test: | | Unit test | | | | |
| Test Software: | | Google Chrome developer tools console | | | | |
| Test Date: | | 07 March 2019 | | | | |
| No. Test | Function Name | Description / Context | Input | Output | Expected Output | Pass/ fail test |
| 01 |  | Retrieves the user ID to connect to firebase database and gets the user settings object. Context: |  | : String | : String | Passed |