This document lists all the test scenarios identified for testing the website.

The document also summarizes the tools/technologies used for automating the test cases.

Due to time constraint, I was able to Automate limited test cases.

|  |  |
| --- | --- |
| **Summary** | |
| **Tools/Technology used** | Selenium, RestAssured, TestNG, Maven |
| **Time spent** | Environment setup and Study of website :2 hours  Drafting test scenarios : 8 hours  Automation : 13 hours |
| **Test Scenarios summary** | No. of test scenarios identified : 52  No. of automated test scenarios : 35 (7 REST API, 28 Selenium automation) |

| 1. **API Test Cases – Add and Remove** | |
| --- | --- |
| **Rest API Test** | **Automated/Manual** |
| Test service status of the API endpoint (only one in this case) | Automated |
| Find how many tasks do not have "category" assigned | Automated |
| Aggregate and print only "task names" | Automated |
| Read API response and print tasks in descending "due date" order | Automated |
| Count and validate the number of tasks | Automated |
| Validate whether all five attributes are returned for (each task) in each JSON array | Automated |
| Aggregate and show all completed tasks | Automated |

| 1. **Tasks – Add and Remove** | | |
| --- | --- | --- |
| **Scenario** | **Test Case** | **Automated/Manual** |
| Add Task with no category and no due date | **Given** that the ‘todo’ Web Page is open  **When** user enters Task Name as ‘testTask1’  **And** clicks the button ‘Add’  **Then** task with name ‘testTask1’ should be added in the list  **And** it should be black in color  **And** Date should be displayed as None to the right of the task name | Automated |
| Add duplicate Task with no category and no due date | **Given** that the ‘todo’ Web Page is open  **When** user enters Task Name as ‘testTask1’  **And** clicks the button ‘Add’ to add ‘testTask1’  **And** user enters Task Name as ‘testTask1’  **And** clicks the button ‘Add’  **Then** page should be navigated to a page which says ‘Sorry that ‘TODO’ item already exists.’ | Automated |
| Add a Task with category and no due date | **Given** that the ‘todo’ Web Page is open and category ‘College’ exists with assigned color yellow  **When** user enters Task Name as ‘testTask1’  **And** select category as ‘College’  **And** clicks the button ‘Add’  **Then** task with name ‘testTask1’ should be added in the list  **And** it should be Yellow in color  **And** Date should be displayed as None to the right of the task name | Automated |
| Add a Task with category and future due date | **Given** that the ‘todo’ Web Page is open and category ‘College’ exists with assigned color yellow  **When** user enters Task Name as ‘testTask1’  **And** selects category as ‘College’  **And** select day as 30 and select month as Dec and select year as 2018  **And** clicks the button ‘Add’  **Then** task with name ‘testTask1’ should be added in the list  **And** it should be Yellow in color  **And** Date should be displayed to the right of task name as ‘30/12/2018’ | Automated |
| Add a Task with no category and future due date | **Given** that the ‘todo’ Web Page is open  **When** user enters Task Name as ‘testTask1’  **And** no category is selected  **And** selects day as 30 and select month as Dec and select year as 2018  **And** clicks the button ‘Add’  **Then** task with name ‘testTask1’ should be added in the list  **And** it should be black in color  **And** Date should be displayed to the right of task name as ‘30/12/2018’ | Automated |
| Add a Task with category with due date in the past | **Given** that the ‘todo’ Web Page is open and category ‘College’ exists with ‘yellow’ color assigned  **When** user enters Task Name as ‘testTask1’  **And** selects category as ‘College’  **And** selects day as 30 and selects month as Mar and selects year as 2018  **And** clicks the button ‘Add’  **Then** task with name ‘testTask1’ should be added in the list  **And** it should be Red in color  **And** Date should be displayed to the right of task name as ‘30/03/2018’ in red color  **And** Category ‘College’ should be displayed to the right of date in yellow color | Automated |
| Add a Task with no category with due date in the past | **Given** that the ‘todo’ Web Page is open  **When** user enters Task Name as ‘testTask1’  **And** no category is selected  **And** selects day as 30 and selects month as Mar and selects year as 2018  **And** clicks the button ‘Add’  **Then** task with name ‘testTask1’ should be added in the list  **And** it should be Red in color  **And** Date should be displayed to the right of task name as ‘30/03/2018’ in red color | Manual |
| Remove a task | **Given** that the ‘todo’ Task Page is open  **And** existing task with name ‘testTask1’  **When** user selects check box on the side of ‘testTask1’  **And** Clicks ‘Remove’ Button  **Then** ‘testTask1’ should be removed | Automated |
| Remove multiple than one task | **Given** that the ‘todo’ Task Page is open  **And** existing tasks with names ‘testTask1’…’testTask4’  **When** user selects check box on the side of ‘testTask1’ …’testTask4’  **And** Clicks ‘Remove’ Button  **Then** ‘testTask1’…’testask4’ should be removed | Automated |
| Add a Task with Invalid due Date selection | **Given** that the ‘todo’ Web Page is open  **When** user enters Task Name as ‘testTask1’  **And** selects day as 30 and selects year as 2018 and selects the month as None  **And** clicks the button ‘Add’  **Then** task with name ‘testTask1’ should be added in the list  **And** Date should be displayed as None to the right of the task name | Automated (Negative Test Case) |
| Add a Task with Invalid due Day selection | **Given** that the ‘todo’ Web Page is open  **When** user enters Task Name as ‘testTask1’  **And** selects day as None and selects year as 2018 and selects the month as 12  **And** clicks the button ‘Add’  **Then** task with name ‘testTask1’ should be added in the list  **And** Date should be displayed as None to the right of the task name | Manual  (Negative Test Case) |

|  |  |  |
| --- | --- | --- |
| 1. **Inspect details of the Tasks** | | |
| **Scenario** | **Test Case** | **Automated/Manual** |
| Inspect Task details (task with Category and Due Date) in the Task Update Page | **Given** that the a task name ‘testTask1’ is already added under category ‘College’ with due date ‘30/12/2018’  **When** user clicks on the link of Task Name ‘testTask1’  **And** navigates to ‘testTask1’ update page  **Then** task name should be displayed as ‘testTask1’  **And** category should be shown as auto populated with value ‘College’  **And** due date should be shown as auto populated with value 30  **And** due month should be shown as auto populated 12  **And** due year should be shown as auto populated 2018 | Automated |
| Inspect Task details (task with Due Date and no Category) in the Task Update Page | **Given** that the a task name ‘testTask1’ is already added with No category and with due date ‘30/12/2018’  **When** user clicks on the link of Task Name ‘testTask1’  **And** navigates to ‘testTask1’ update page  **Then** task name should be displayed as ‘testTask1’  **And** category should be auto-populated with No Value  **And** due date should be shown as auto populated with value 30  **And** due month should be shown as auto populated 12  **And** due year should be shown as auto populated 2018 | Automated |
| Inspect Task details (Task with Category and no Due Date) n the Task Update Page | **Given** that the a task name ‘testTask1’ is already added with category ‘College’ and no due date  **When** user clicks on the link of Task Name ‘testTask1’  **And** navigates to ‘testTask1’ update page  **Then** task name should be displayed as ‘testTask1’  **And** category should be auto-populated with value ‘College’  **And** due date should be shown as auto populated with No Value  **And** due month should be shown as auto populated with No Value  **And** due year should be shown as auto populated with current year | Automated |
| Inspect Task details (Task with no Category and no Due Date) n the Task Update Page | **Given** that the a task name ‘testTask1’ is already added with No category and no due date  **When** user clicks on the link of Task Name ‘testTask1’  **And** navigates to ‘testTask1’ update page  **Then** task name should be displayed as ‘testTask1’  **And** category should be auto-populated with No Value  **And** due date should be shown as auto populated with No Value  **And** due month should be shown as auto populated with No Value  **And** due year should be shown as auto populated with current year | Automated |

|  |  |  |
| --- | --- | --- |
| 1. **Update details of the Tasks** | | |
| **Scenario** | **Test Case** | **Automated/Manual** |
| Update task Name | **Given** that the Update Task Page is open  **When** user renames Task Name ‘testTask1’ to ‘testTask2’  **And** clicks the button ‘Update’  **Then** user should be navigated to‘todo’ main page  **And** task with name ‘testTask1’ should be renamed to ‘testTask2’ | Automated |
| Update category for a task with no due date | **Given** that the Update Task Page is open  **And** a task with name ‘testTask1’ under category ‘College’ and no due date exists.  **And** ‘College’ category is assigned color ‘yellow’ and ‘Play’ category is assigned ‘Gray’  **When** user changed task ‘testTask1’ category from ‘College’ to ‘Play’  **And** clicks the button ‘Update’  **Then** user should be navigated to‘todo’ main page  **And** task with name ‘testTask1’ should be with color ‘Gray’ | Automated |
| Update with past due date for task with no category | **Given** that the Update Task Page is open  **And** a task with name ‘testTask1’ with no category.  **When** user selects day as 30 and selects month as Mar and selects year as 2018  **And** clicks the button ‘Update’  **Then** user should be navigated to‘todo’ main page  **And** task with name ‘testTask1’ should be Red in color with date ‘30/03/2018’ | Automated |
| Update with past due date for task with category | **Given** that the Update Task Page is open and category ‘College’ exists with assigned color yellow  **And** a task with name ‘testTask1’ with category College.  **When** user selects day as 30 and selects month as Mar and selects year as 2018  **And** clicks the button ‘Update’  **Then** user should be navigated to‘todo’ main page  **And** task with name ‘testTask1’ should be Red in color with date ‘30/03/2018’  **And** Category name ‘College’ should be displayed in yellow color to the right of date. | Automated |
| Update with future due date for task with no category | **Given** that the Update Task Page is open  **And** a task with name ‘testTask1’ with no category.  **When** user selects day as 30 and selects month as Dec and selects year as 2018  **And** clicks the button ‘Update’  **Then** user should be navigated to‘todo’ main page  **And** task with name ‘testTask1’ should be Black in color with date ‘30/12/2018’ on its right | Manual |
| Update with future due date for a task with category | **Given** that the Update Task Page is open and category ‘College’ exists with assigned color yellow  **And** a task with name ‘testTask1’ with ‘College’ category.  **When** user selects day as 30 and selects month as Dec and selects year as 2018  **And** clicks the button ‘Update’  **Then** user should be navigated to‘todo’ main page  **And** task with name ‘testTask1’ should be updated to yellow in color with date ‘30/12/2018’ on its right | Automated |
| Update category for a task with future due date | **Given** that the Update Task Page is open  **And** ‘College’ category is assigned color ‘yellow’ and ‘Play’ category is assigned ‘Gray’  **And** a task with name ‘testTask1’ with ‘College’ category.  **When** user changed task ‘testTask1’ category from ‘College’ to ‘Play’  **And** clicks the button ‘Update’  **Then** user should be navigated to‘todo’ main page  **And** task with name ‘testTask1’ should be with color ‘Gray’ with future date on its right | Manual |

|  |  |  |
| --- | --- | --- |
| 1. **Inspect Tasks - Complete, Toggle All** | | |
| **Scenario** | **Test Case** | **Automated/Manual** |
| Inspect details of the overdue task with no category and not complete | **Given** that the ‘todo’ Task Page is open with a task name ‘testTask1’  **And** task ‘testTask1’ with no category and not completed  **When** due date is past for ‘testTask1’  **Then** ‘testTask1’ should be shown in ‘Red’ color with due date to its right in Red color | Manual |
| Inspect details of the overdue task with category and not complete | **Given** that the ‘todo’ Task Page is open with a task name ‘testTask1’  **And** ‘College’ category present with assigned color ‘yellow’  **And** task ‘testTask1’ is assigned to ‘College’ category  **And** task ‘testTask1’ is not complete  **When** due date is past for ‘testTask1’  **Then** ‘testTask1’ should be shown in ‘Red’ color with due date to its right in Red color  **And** ‘College’ in ‘yellow’ color should be displayed on the right of due date | Manual |
| Inspect details of the future due date task with no category and not complete | **Given** that the ‘todo’ Task Page is open with a task name ‘testTask1’  **And** task ‘testTask1’ with no category and not completed  **When** due date is in future for ‘testTask1’  **Then** ‘testTask1’ should be shown in ‘Black’ color with due date to its right in Black color | Manual |
| Inspect details of the future due date task with category and not complete | **Given** that the ‘todo’ Task Page is open with a task name ‘testTask1’  **And** ‘College’ category present with assigned color ‘yellow’  **And** task ‘testTask1’ is assigned to ‘College’ category  **And** task ‘testTask1’ is not complete  **When** due date is in future for ‘testTask1’  **Then** ‘testTask1’ should be shown in ‘Yellow’ color with due date to its right in Black color | Manual |
| Inspect task of the previous/future due date task with no category and complete | **Given** that the ‘todo’ Task Page is open with a task name ‘testTask1’  **And** No category is selected and with past/future due date  **And** task is complete  **When** due date is in future for ‘testTask1’  **Then** ‘testTask1’ should be shown in Black color with a strike with due date to its right in Black color | Automated |
| Inspect details of the overdue/future dated task with category and complete | **Given** that the ‘todo’ Task Page is open with a task name ‘testTask1’  **And** ‘College’ category present with assigned color ‘yellow’  **And** task ‘testTask1’ is assigned to ‘College’ category  **And** task due date is past/future  **When** task ‘testTask1’ is complete  **Then** ‘testTask1’ should be shown in ‘Yellow’ color with a strike and due date to its right in Black color | Automated |
| TOGGLE ALL/UNTOGGLE ALL Tasks | **Given** that the ‘todo’ Task page is open with more than one task  **AND** (Optional) atleast one task checked on in the list  **When** user checks on the checkbox ‘Toggle All’  **Then** all the tasks checkbox should be selected  **Given** that the ‘todo’ Task page is open with more than one task  **AND** (Optional) atleast one task is unchecked on in the list  **When** user unchecks on the checkbox ‘Toggle All’  **Then** all the tasks checkbox should be unselected | Automated |
| Task with name that contains a Hyperlink | **Given** that the ‘todo’ Task page is open with a task which has downloadable link to a file  **When** user clicks on the hyperlink  **Then** a file should be downloaded which is referenced in the hyperlink | Manual |
| Task with name that contains a website address | **Given** that the ‘todo’ Task page is open with a task which has a website link  **When** user clicks on the hyperlink  **Then ‘**todo’ page will be navigated to the new website mentioned in the hyperlink |  |

|  |  |  |
| --- | --- | --- |
| 1. **Add/Delete/Rename Category** | | |
| **Scenario** | **Test Case** | **Automated/Manual** |
| Add a category with a color which is not assigned to other categories | **Given** that the ‘todo’ Task Page is open  **And** color ‘Yellow is not assigned to any existing category  **When** user enters a category text box with name ‘College’  **And** Selects ‘Yellow  **And** Clicks ‘Add Category’  **Then** ‘College’should be added under category list  **And** ‘College’should be styled in yellow color | Automated |
| Add a category with a color which is already assigned to other category (Rename the Category) | **Given** that the ‘todo’ Task Page is open  **And** color ‘Blue’ is assigned to existing category ‘ExistingCategory’  **When** user enters a category text box with name ‘ProposedCategory’  **And** Selects ‘Blue’  **And** Clicks ‘Add Category’  **Then** a clickable link to be available to **rename existing category**  **And** on click replaces ‘ExistingCategory’ with ‘ProposedCategory’ | Automated |
| Add a category with a color which is already assigned to other category (Add the category with this colour) | **Given** that the ‘todo’ Task Page is open  **And** existing category ‘ExistingCategory’ with color ‘Blue’  **When** user enters a category text box with name ‘ProposedCategory’  **And** Selects ‘Blue’  **And** Clicks ‘Add Category’, followed by a click on the link to add the category(i.e., to add ‘ProposedCategory’ in Blue)  **Then** ‘ProposedCategory’ should be assigned blue and added  **And** ‘ExistingCategory’ will be displayed with no color | Automated |
| Add a category with a color which is already assigned to other category (Nevermind) | **Given** that the ‘todo’ Task Page is open  **And** existing category ‘ExistingCategory’ with color ‘Blue’  **When** user enters a category text box with name ‘ProposedCategory’  **And** Selects ‘Blue’  **And** Clicks ‘Add Category’, followed by a click on the link to Nevermind  **Then** categories and tasks are not effected | Automated |
| Add Category - same name, different colour – (Nevermind) | **Given** that the ‘todo’ Task Page is open  **And** there is an existing category ‘ExistingCategory’ with color ‘Blue’ assigned  **When** user adds a category with same name ‘ExistingCategory’ and a un-assigned colour (to any category)  **Then** site navigates to a page where a prompt is shown as –  *The category you want to add already exists: ExistingCategory. Do you want to update its colour to ExistingCategory? With Yes/Nevermind link*  **And** On click of ‘Nevermind’ ,neither the existing categories nor the tasks get affected. | Manual |
| Add Category - same name different color -(Yes) | **Given** that the ‘todo’ Task Page is open  **And** there is an existing category ‘ExistingCategory’ with color ‘Blue’ assigned  **When** user adds a category with same name ‘ExistingCategory’ and a un-assigned colour (to any category)  **Then** site navigates to a page where a prompt is shown as –  *The category you want to add already exists: ExistingCategory. Do you want to update its colour to ExistingCategory? With Yes/NeverMind link*  **And** On click of ‘Yes’ , the color of ‘ExistingCategory’ will turn to new color | Manual |
| Add Category (Same name and same color) | **Given** that the ‘todo’ Task Page is open  **And** existing category ‘ExistingCategory’ with color ‘Blue’  **When** user clicks the category with name ‘ExistingCategory’ with color ‘Blue’  **Then** site navigates to a page where it shows The category you want to add already exists: ExistingCategory’ | Manual |
| Remove a Category(Yes) | **Given** that the ‘todo’ Task Page is open  **And** there is an ‘ExistingCategory’ with color ‘Blue’ assigned  **When** user clicks ‘ExistingCategory’ ,site navigates to a page where a prompt is shown as –  Are you sure you wish to remove Work?  **And** on click on ‘Yes’**Then** the ‘ExistingCategory’ should be removed | Automated |
| Remove a Category (Nevermind) | **Given** that the ‘todo’ Task Page is open  **And** there is an ‘ExistingCategory’ with color ‘Blue’ assigned  **When** user clicks ‘ExistingCategory’ ,site navigates to a page where a prompt is shown as –  Are you sure you wish to remove Work?  **And** on click on ‘Nevermind**Then** the ‘ExistingCategory’ should not be removed | Manual |

|  |  |  |  |
| --- | --- | --- | --- |
| 1. **Tasks and Categories (together)** | | | |
| **Scenario** | **Test Case** | **Automated/Manual** | **Comments** |
| Add a category with a color which is already assigned to other category with tasks (Rename the Category option) | **Given** Test Case ‘**Add a category with a color which is already assigned to other category’**  **And** ‘testTask1’ is already assigned to ‘ExistingCategory’  **Then** ‘testTask1’ will be assigned to‘Proposed Category’ | Automated | This test scenario links to the test case ***Add a category with a color which is already assigned to other category*** under ***Add/Delete/Rename Category***.  This scenario tests how tasks get affected on any updates made to Categories. |
| Add a category with a color which is already assigned to other category with tasks (Add the category with this colour option) | **Given** Test Case **‘*Add a category with a color which is already assigned to other category (Add the category with this colour)’***  **And** ‘testTask1’ is already assigned to ‘ExistingCategory’  **Then** ‘testTask1’ should remain with ‘ExistingCategory’ | Automated | This test scenario links to the test case ***Add a category with a color which is already assigned to other category (Add the category with this colour)*** under ***Add/Delete/Rename Category***.  However, this scenario tests only the end result w.r.t how tasks get affected on any updates made to Categories. |
| Add Category with different name and existing colour assigned to different category (Nevermind) | **Given** that the ‘todo’ Task Page is open  **And** there is an existing category ‘ExistingCategory’ with color ‘Blue’ assigned  **When** user adds a category with different name ‘ProposedCategory’ and assigned color ‘Blue’  **Then** site navigates to a page where a prompt is shown as –  *That colour is already being used by: Work.  Do you want to rename Work to Working?  Or do you want to add Working and remove the colour of 'Work'?* [*Rename the category*](http://localhost:85/nss-todo-automation/todo.php?rename=yes&colour=00FFFF&categorydata=Working)*,*[*Add the category with this colour*](http://localhost:85/nss-todo-automation/todo.php?colourpinchadd=yes&colour=00FFFF&categorydata=Working)*, [Nevermind](javascript:history.back();)*  **And** On click of ‘Nevermind’ ,neither the existing categories nor the tasks get affected | Manual |  |
| Task/Category Names in Upper case | Task / Category Names in Upper cases are treated differently, when compared to lower case. A test scenario can be automated for the same if that is a feature | Manual | Negative Test Case |

|  |  |  |  |
| --- | --- | --- | --- |
| 1. **NA for Automation** | | | |
| **Scenario** | **Test Steps** | **Expected Result** | **Comments** |
| Existing Task with future due date | The due date reached | The color of task along with date should turn to color ‘Red’ irrespective of category assigned or not. | This test case can be automated, but better be tested manually as the automated test case with a hard-coded due date will pass only on a single day in a given year. Instead, it can be manually tested. |