**Acceptance testing**

|  |  |  |  |
| --- | --- | --- | --- |
| Input sequence | Expected result | Current output | Comment |
| Requirement 1: The user should be able to log in with a 6-digit staff number and password | | | |
| Correct login for normal user:  Staff no. = 1234  Password = 1234 | User logs in successfully | A screenshot of a computer  Description automatically generated | As the user has logged in successfully, they have been directed to the dashboard |
| Incorrect login  Staff no. = 5643  Password = wrong | User does not log in successfully |  | As the details are incorrect the page does not re-direct anywhere and the search bar shows “invalid\_credentials”. |
| Requirement 2: The user should be able to create a task | | | |
| Creating a task successfully  Filling in unique details to create a task | The task should appear in the database | A screenshot of a task  Description automatically generated  A screenshot of a black box with white text  Description automatically generated | The task has been created successfully. The three screenshots show the various stages. First filling in the form, receiving confirmation and the last screenshot shows the task in the database. |
| Creating a task unsuccessfully  Attempting to create a task which already exists within the database | The task should not appear in the database | A black screen with white text  Description automatically generated | The task has not been created and the user has received an appropriate message so they can amend task details to retry. |
| Requirement 3: The admin should be able to delete a task | | | |
| Deleting a task successfully  The admin password is 1234 to delete tasks. | The task should be deleted and not exist in the database. |  |  |
|  |  |  |  |

* The user should be able to log in with a 6-digit staff number and password.
* Anyone can create a task
* Only the admin can delete tasks
* Anyone can assign a task
* When signing up the users: email address, staff number, team, first name, surname, rank – should be recorded.
* An option to show who the tasks can be viewed by
* Once a task is created, details should be updateable but not changeable.
* Enable the user to update the task status, due and review dates, and input progress updates to a rolling log.
* No specific requirements for colours/themes/fonts – make sure to cater to neurodivergent individuals too.
* No specific requirement – should be a common language though.
* User should be able to see their assigned tasks.
* Tasks should be ordered by due date.
* Should be able to export PDFs of overdue tasks, all tasks, pending tasks.
* Only a supervisor can say a task is complete.
* Staff members should get an email when a task is assigned
* Should be compatible with open auth.

**Code inspection**

|  |  |  |  |
| --- | --- | --- | --- |
| Inspecting | Expected result | Current output | Comment |
| Correctly naming elements (Example 1)  Task complete button | taskCompleted, completeTask, uponCompletion |  | Input fields and buttons have been correctly named so they can be identified easily. |
| Correctly naming elements  (Example 2)  Navigation bar | menuItems, navigationBar, navBar | A computer screen shot of text  Description automatically generated | Classes for the navigation bar have been correctly named. |
| Using comments | Systematic use of comments throughout code | A screen shot of a computer code  Description automatically generated  A computer screen with text and images  Description automatically generated | Screenshots from three different files, all showing systematic and consistent use of comments throughout. |
| Using correct coding conventions such as camel casing | Correct use of camel casing when naming elements | A computer code with text  Description automatically generated with medium confidence | Systematic and correct use of camel casing throughout code. |
| No harsh use of colours. | Colours are expected to flow well and are easy on the eyes. |  | Appropriate use of colours throughout e.g., green for submit/success and red for failure/delete. Other colours flow well and do not clash with each other or other elements such as font. |
| Appropriate use of font styles. | No elaborate use of styles such as handwriting. |  | Clear and easy to read font styles have been used. |
| Appropriate use of font sizes. | The font sizes should not be too large or too small. |  | Font sizes have been used appropriately, headings large, sub-headings slightly smaller and so on. |
| User experience and useability. | The site should be easy to navigate and self-explanatory. |  | Menu items in the navigation bar are self-explanatory. All elements/buttons have been labelled clearly. |
| Are login details being sent securely to the server. | Login details are not visible in the search bar and are not showing in inspector tools. |  | As we can see only the invalid credential shows in the navbar. When the user enters correct details, they are automatically redirected. |
| Error handling mechanism (Example 1) | Trying to sign up a user which has a staff number that already exists in the database. The user should not be created. |  | Appropriate error message appears when trying to sign up a user which already exists in the database. |
| Error handling mechanism (Example 2) | Trying to add a task which already exists in the database. The task should not be added. |  | Appropriate error message appears when trying to create a task which already exists. |
| Missing input handling | Leaving signup/login fields blank. |  | The user is prompted to enter the missing details. |
| Clearing cache after the user signs out so content is not accessible | Content should not visible after the user signs out as this would violate security. |  | Screenshots show cache is being cleared and session variables being reset and destroyed. This provides security to ensure logged in user cannot see previous users’ data. |

Unit testing

Meeting minutes

Finish acceptance testing

Finish unit testing

Add comments to code

**Peer Review**

All members contributed equally and achieved tasks to a good standard. All members have a score of 10.

17003493 - Sania Bibi

* Completed...

22006776 - Imara Ali

* Completed...

22031400 - Javairia Shahid

* Completed...

20010596 - Amal Abeso Ela

* Completed...

21045121 - Oluwadamilare Falade

* Completed...

22036203 - Shafeeq Shuaib

* Completed...