Testing:

The Testing of the project was done in all the phases of the project development, after each sprint completion the testing was done, and the potential errors were resolved by the team of developers. The unit tests were also written to test the internal functionality of the code. In this way the code was thoroughly tested using white box techniques.

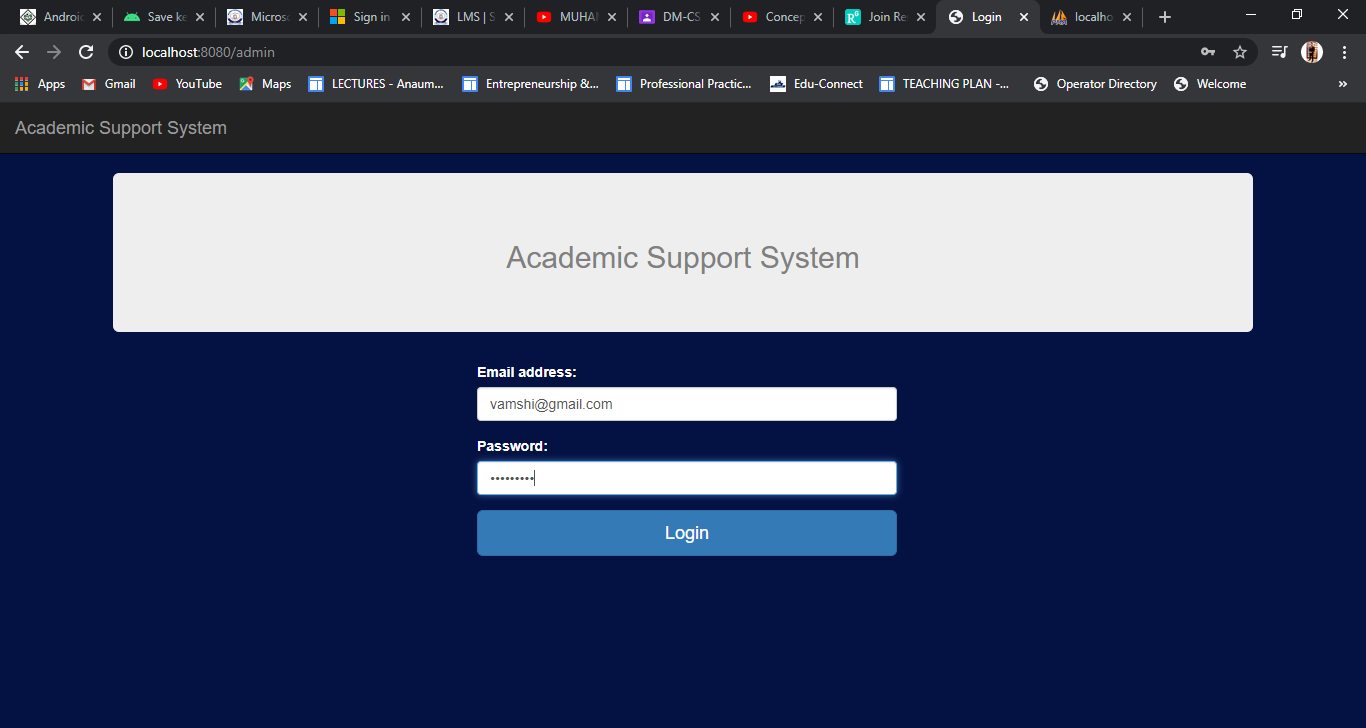
After completion of each phase the product is tested by several users using the black box techniques to check the functionality of code by using the application. Then the errors were resolved after each phase to make the client satisfied and check the acceptance criteria along with it, whether it is fulfilling the criteria or not.

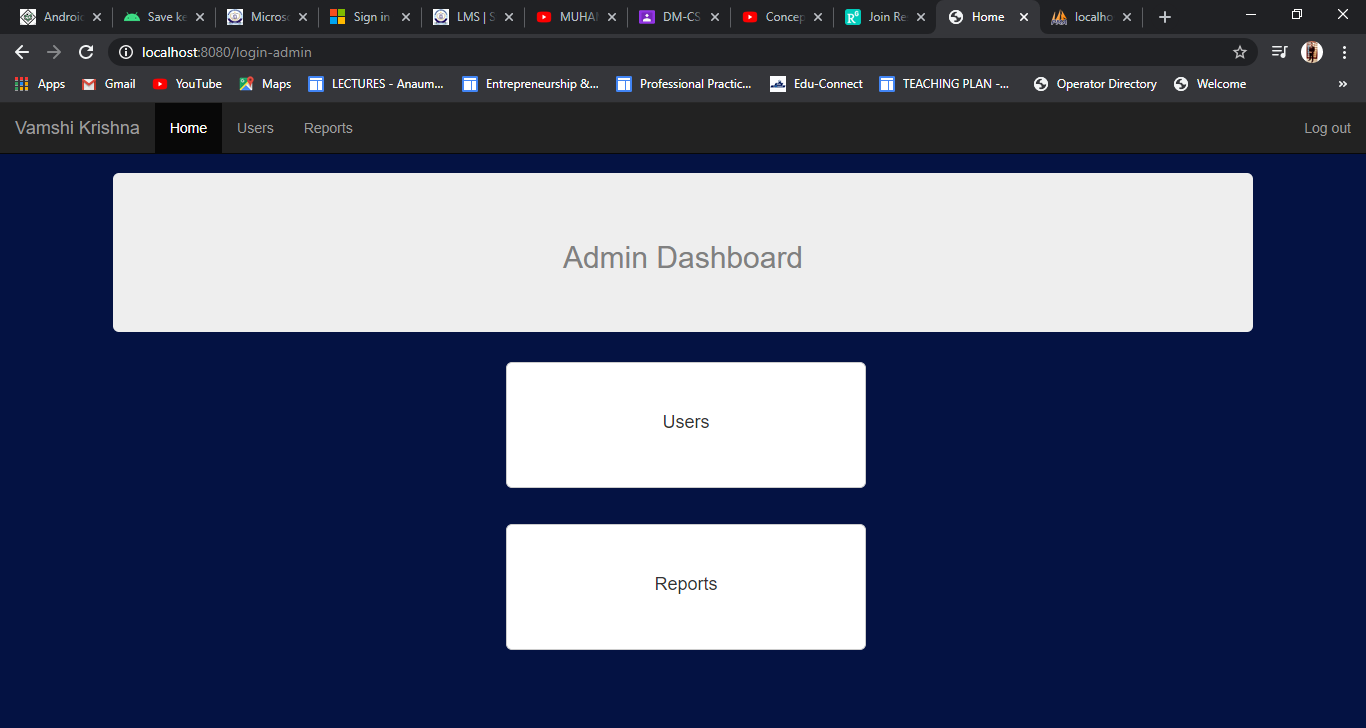
In this way the whole application was tested by the team and the potential errors were resolved when they were found.

1. **Test Admin Panel Login:**

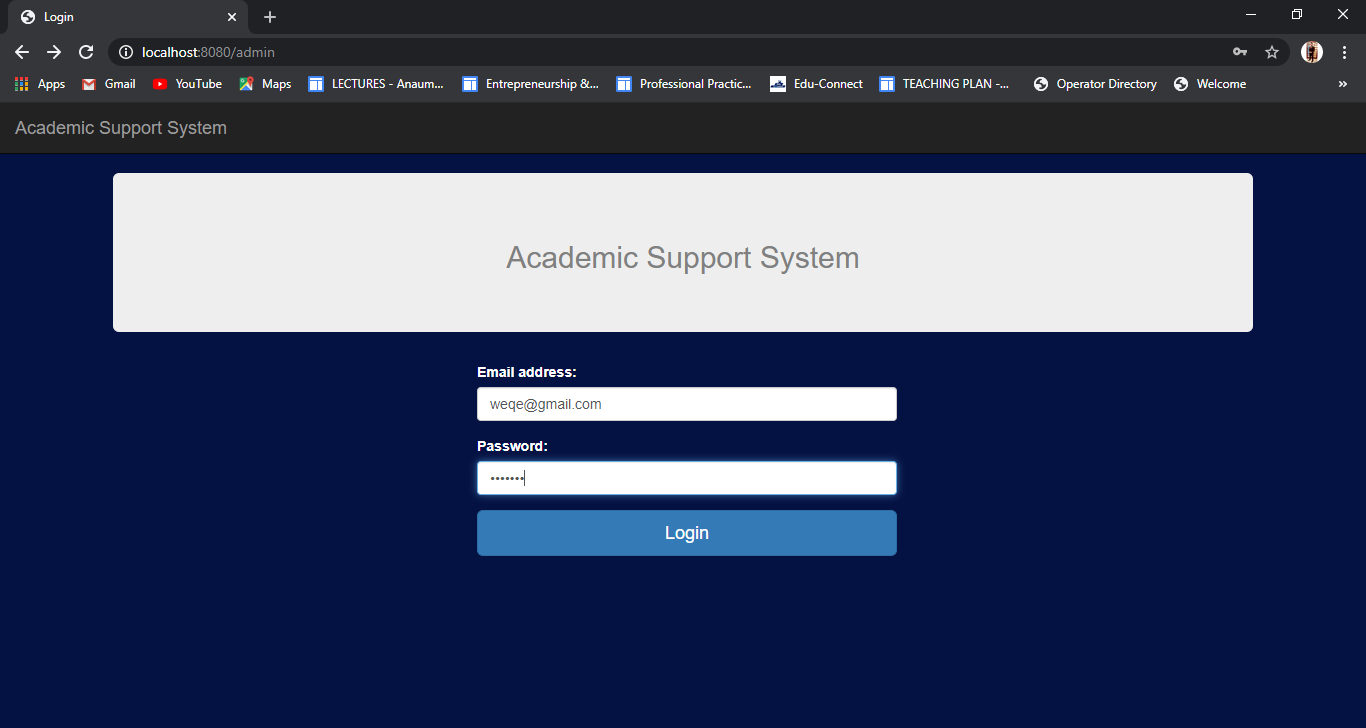
|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| ID | Scenario | Preconditions | Steps involved | Expected Result | Actual Result | Status |
| 1 | Admin can login successfully | Admin already have the email and password | Enter correct password and email.  Then click on Login button. | Admin login to the system. | Admin login to the system. | **Passed** |
| 2 | User except admin tries to login to the system | User doesn’t have the email or password of the admin account. | Enter invalid email or password.  Then click on Login button. | Username/Password/Role incorrect | Username/Password/Role incorrect | **Passed** |

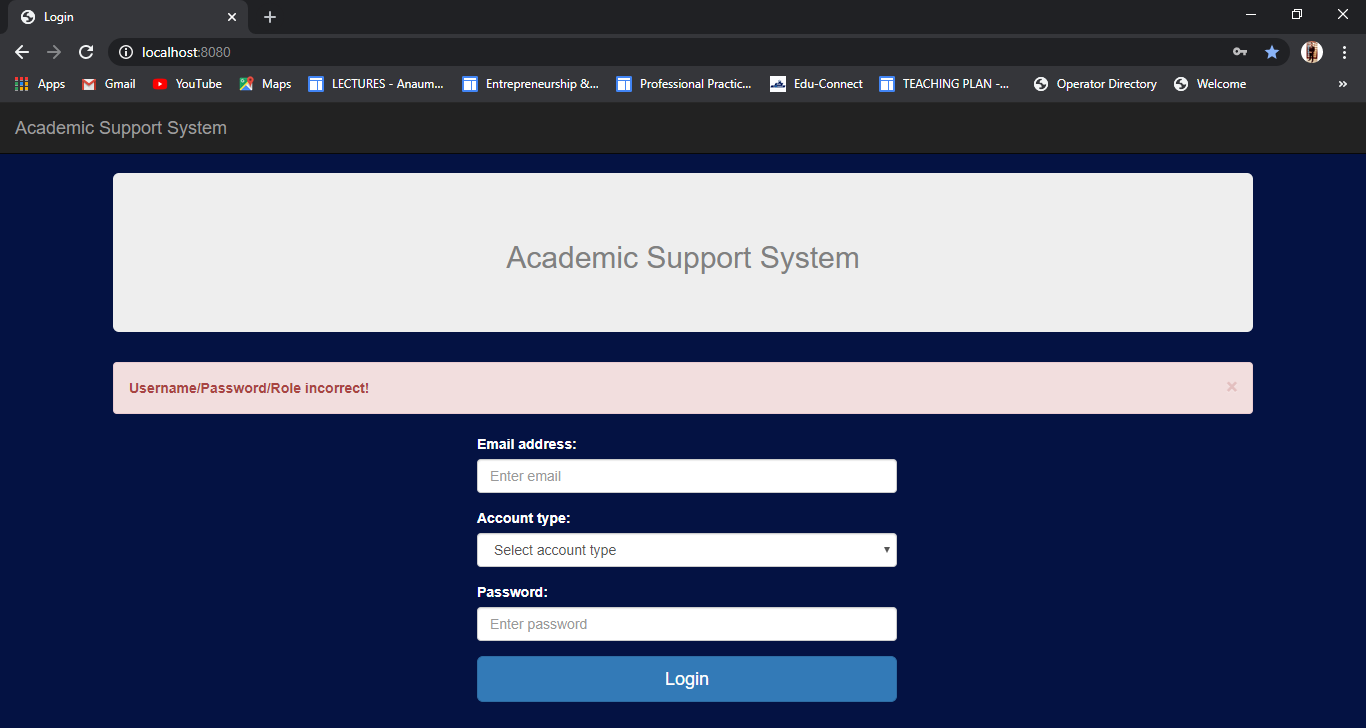
1. **Admin can login successfully:**





**2) User except admin tries to login:**

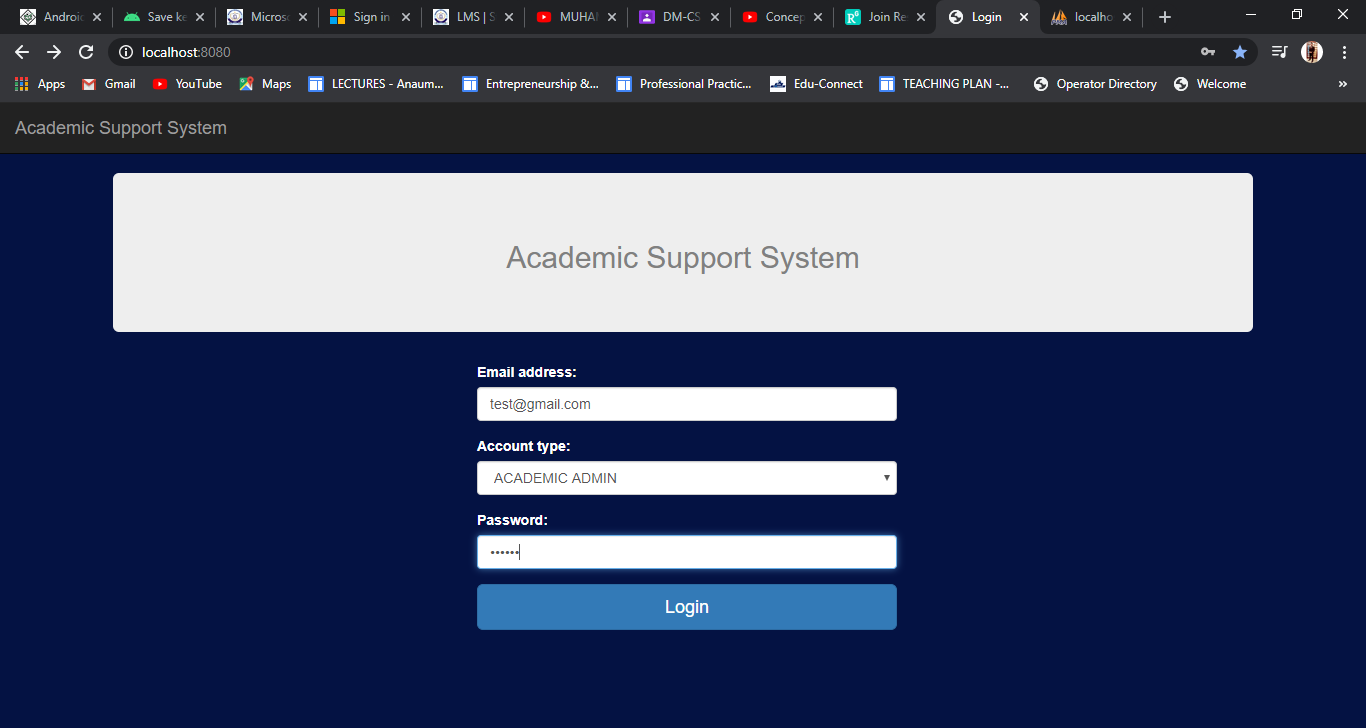


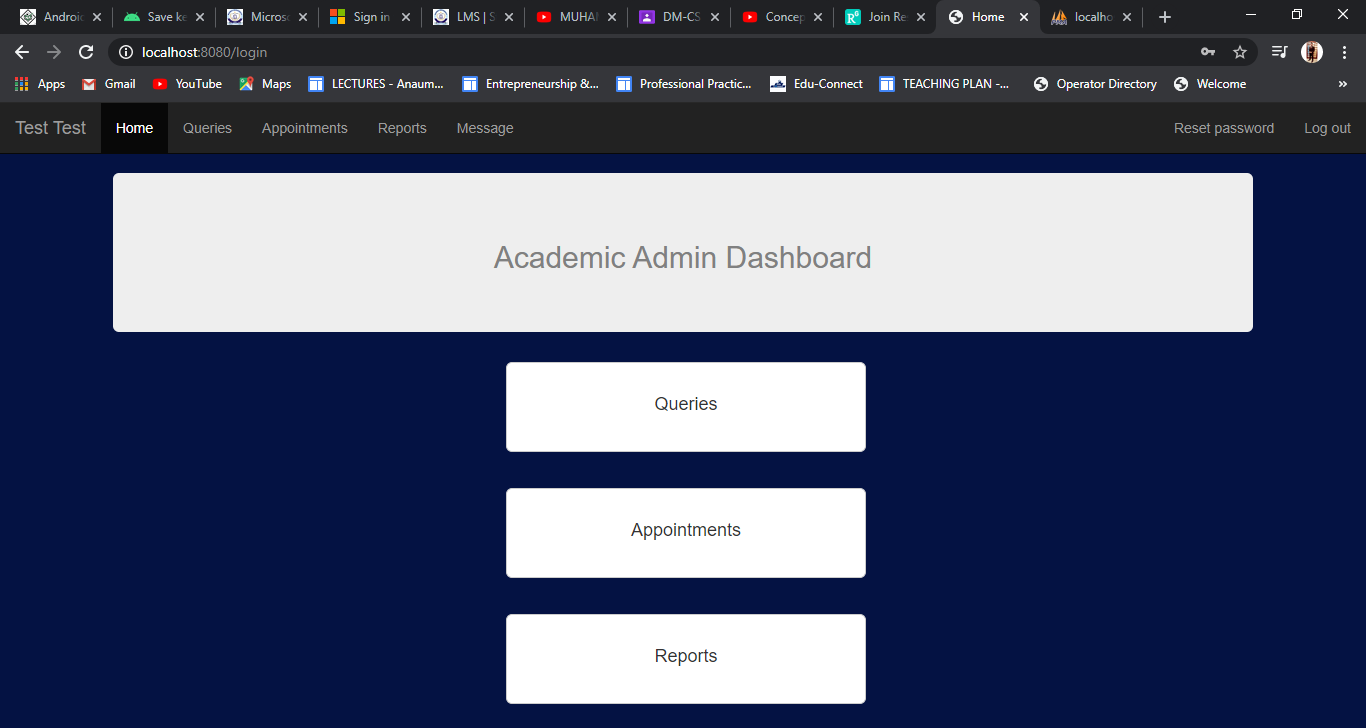


**2) Test Academic admin login:**

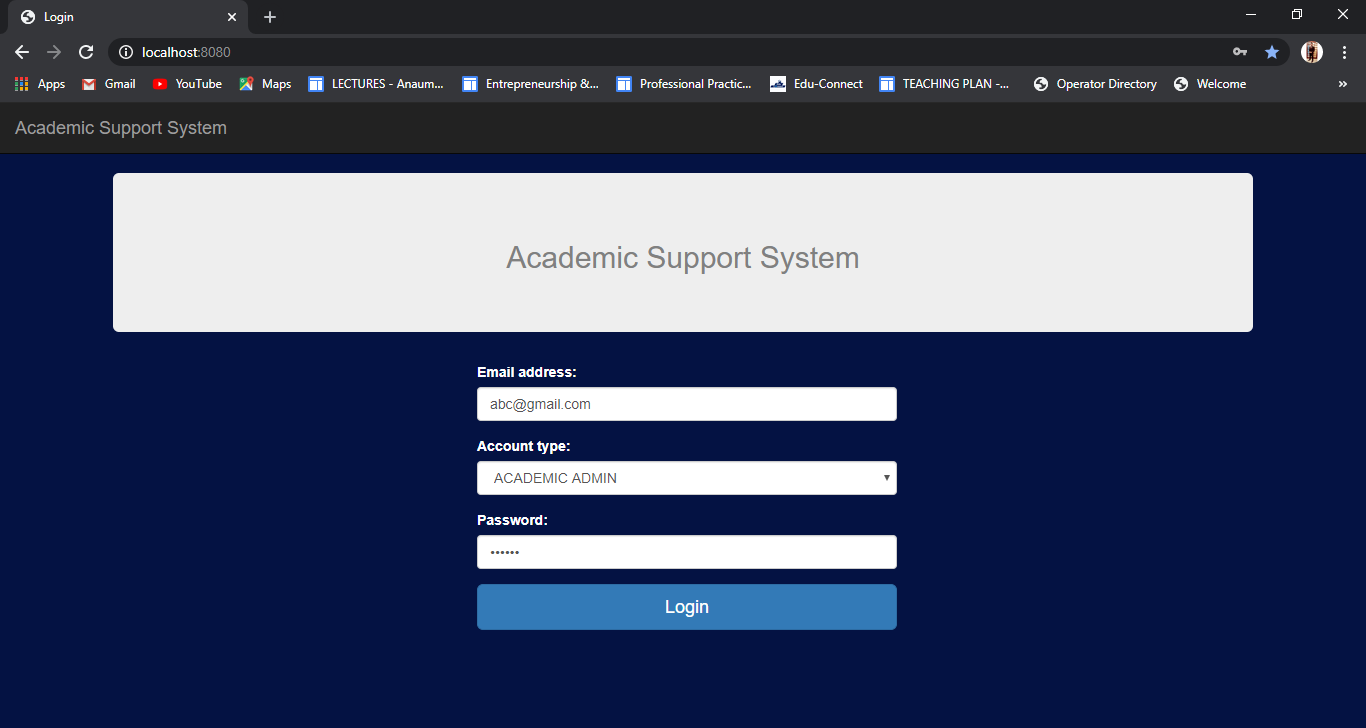
|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| ID | Scenario | Preconditions | Steps involved | Expected Result | Actual Result | Status |
| 1 | Academic Admin can login successfully. | Academic Admin already have the email and password | Enter correct password and email.  Then click on Login button. | Academic Admin login to the system. | Academic Admin login to the system. | **Passed** |
| 2 | User except academic admin tries to login to the system. | User doesn’t have the email or password of the admin account. | Enter invalid email or password.  Then click on Login button. | Username/Password/Role incorrect | Username/Password/Role incorrect | **Passed** |

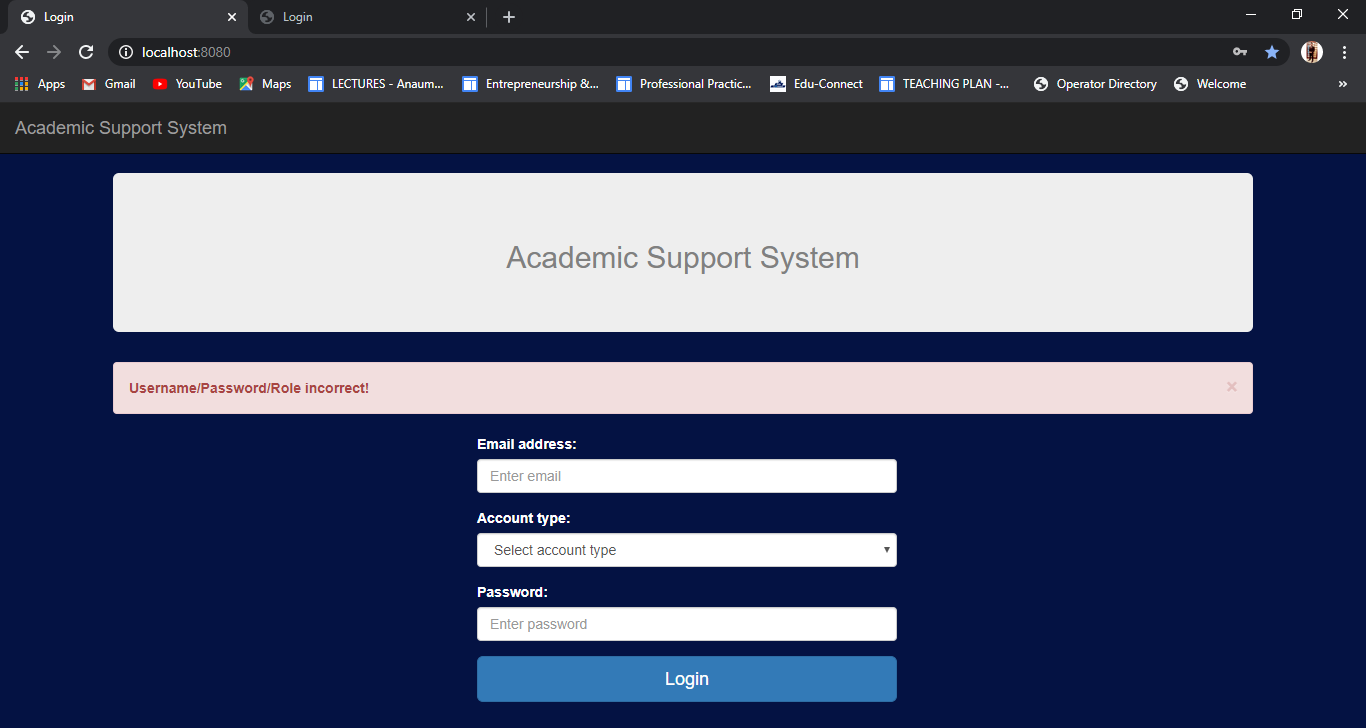
1. **Academic admin can login successfully:**





1. **User except academic admin tries to login:**

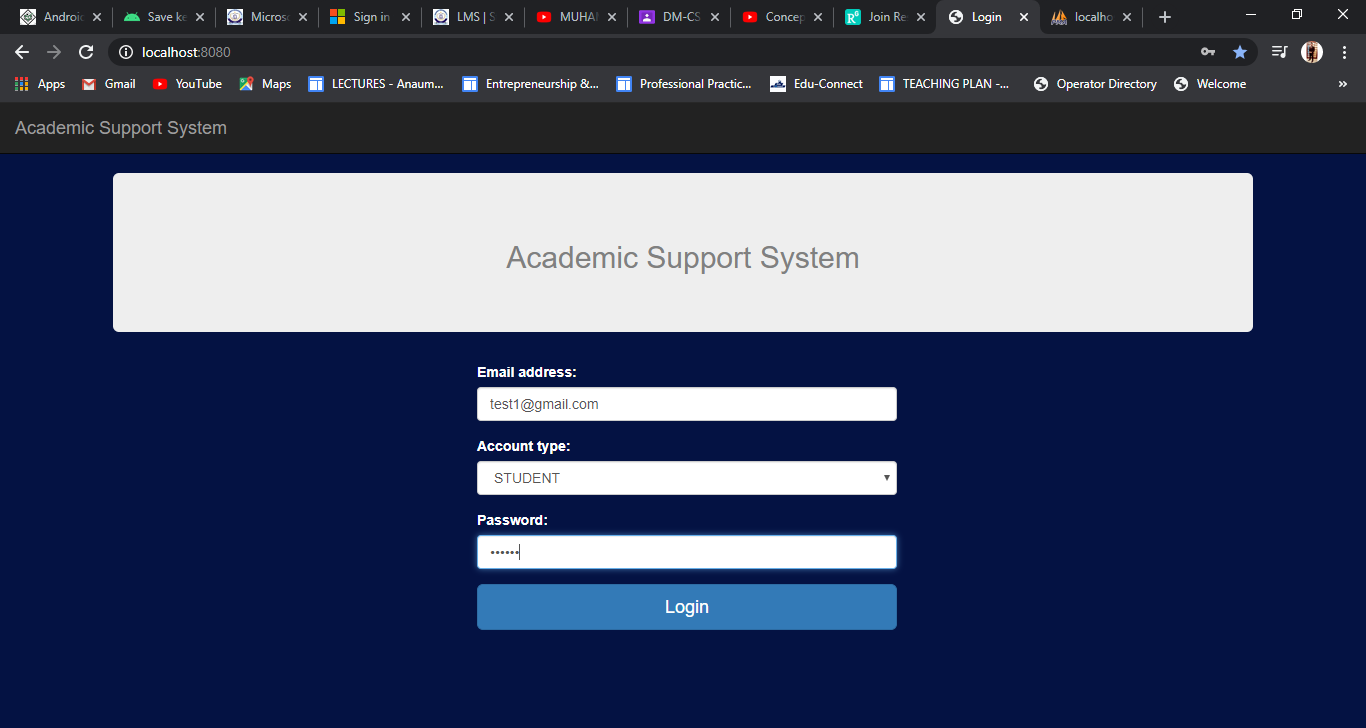


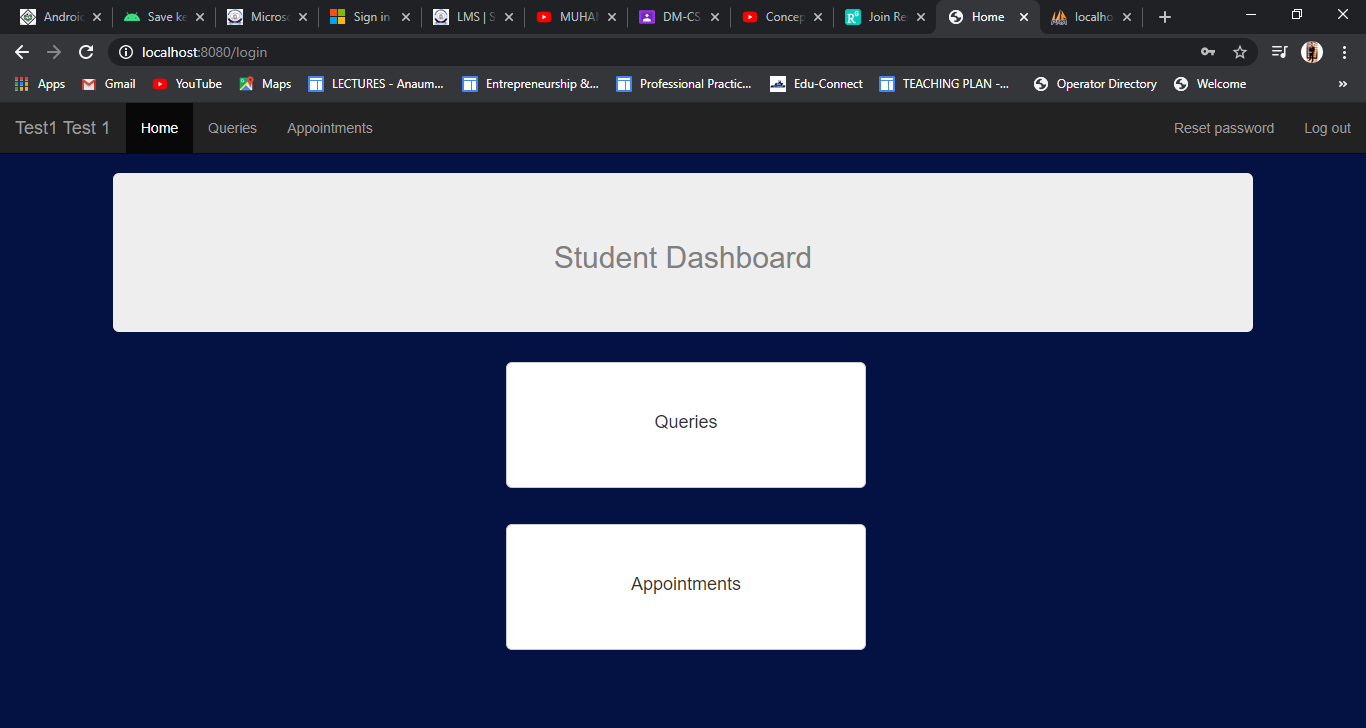


**3) Test Student login:**

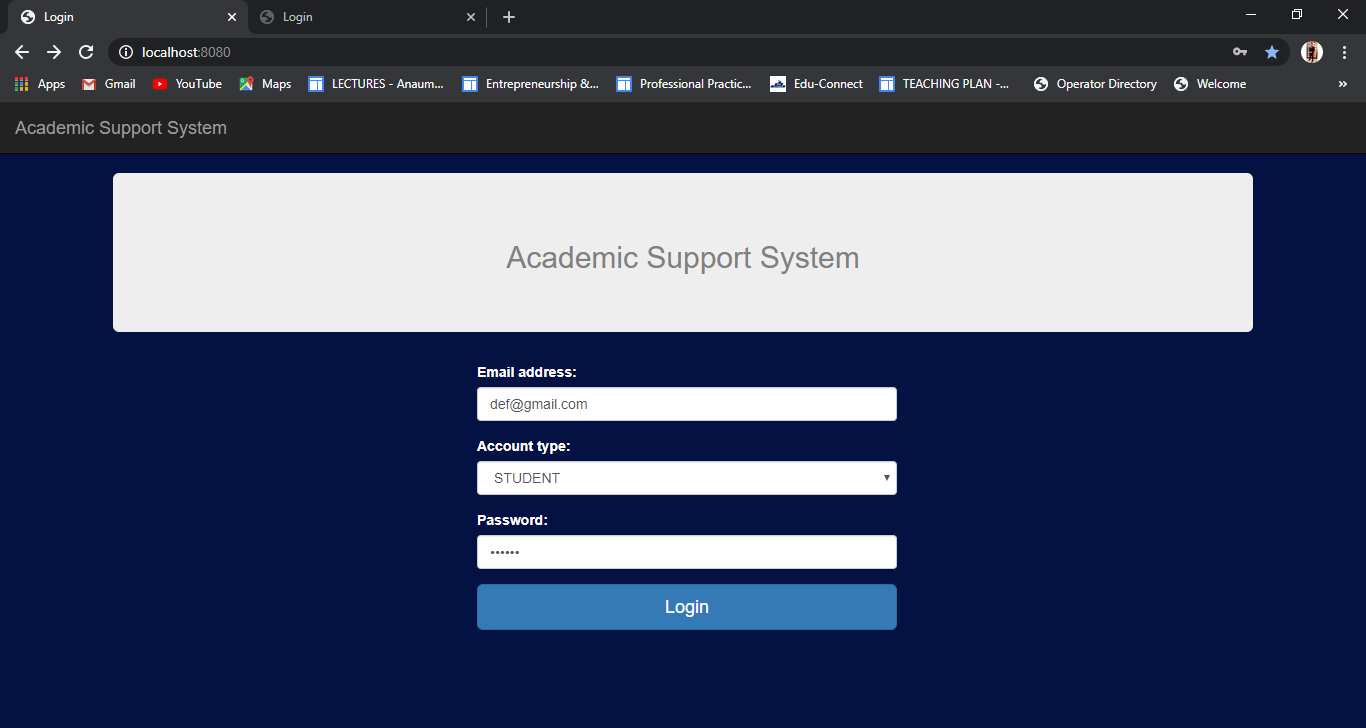
|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| ID | Scenario | Preconditions | Steps involved | Expected Result | Actual Result | Status |
| 1 | Student can login successfully | Student already have the email and password | Enter correct password and email.  Then click on Login button. | Student login to the system. | Student login to the system. | **Passed** |
| 2 | User except Student tries to login to the system | User doesn’t have the email or password of the admin account. | Enter invalid email or password.  Then click on Login button. | Username/Password/Role incorrect | Username/Password/Role incorrect | **Passed** |

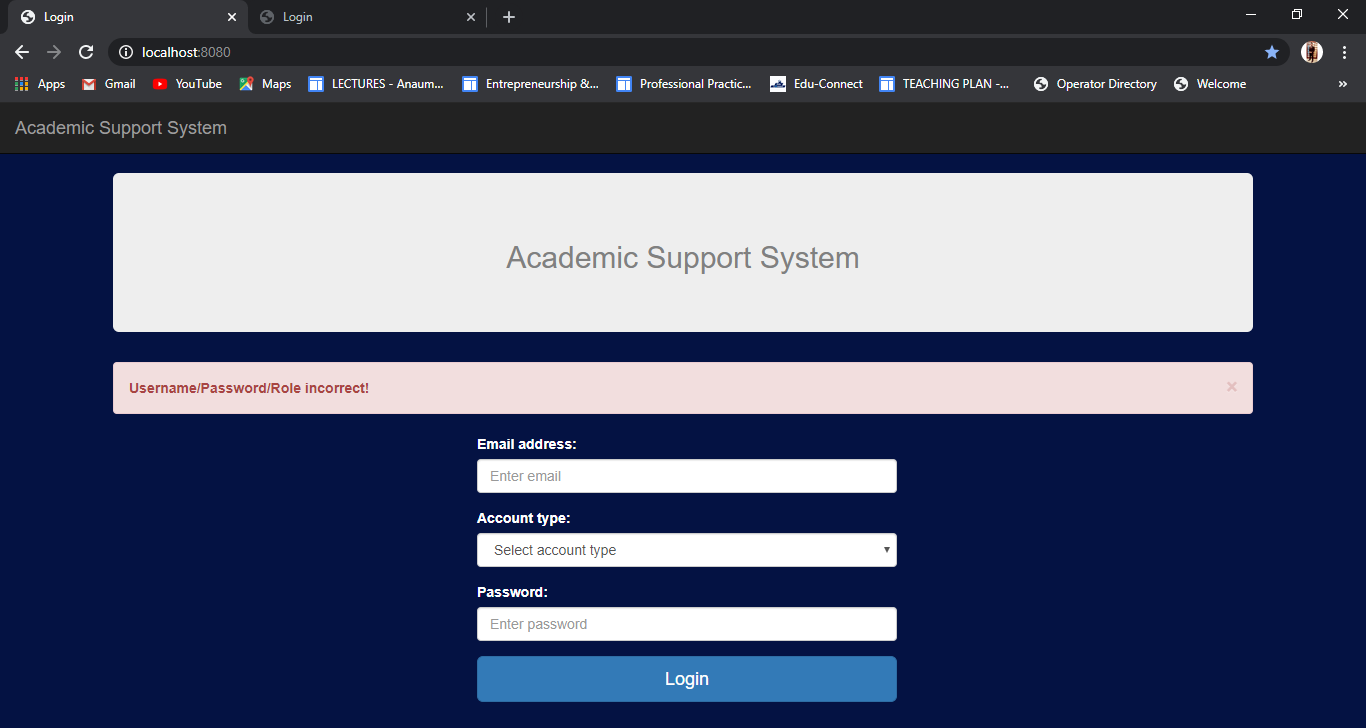
1. **Student can login successfully:**





1. **User except Student tries to login:**

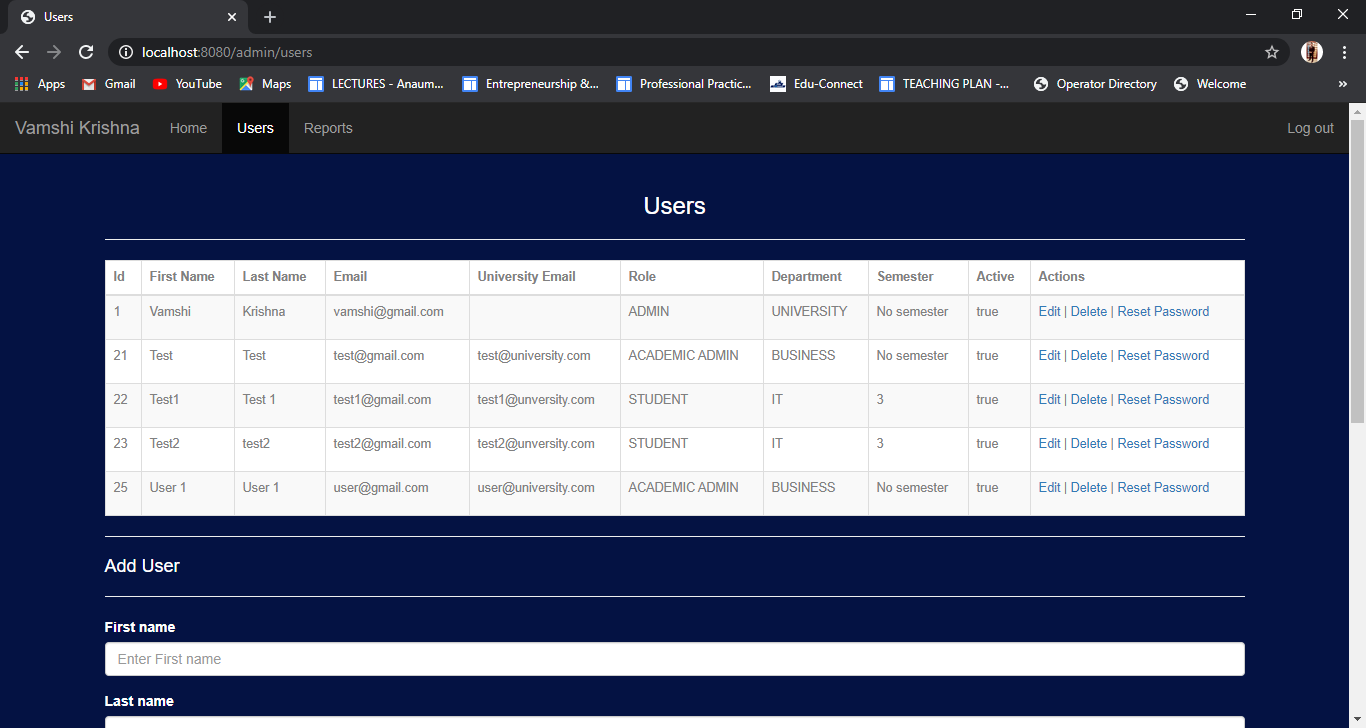




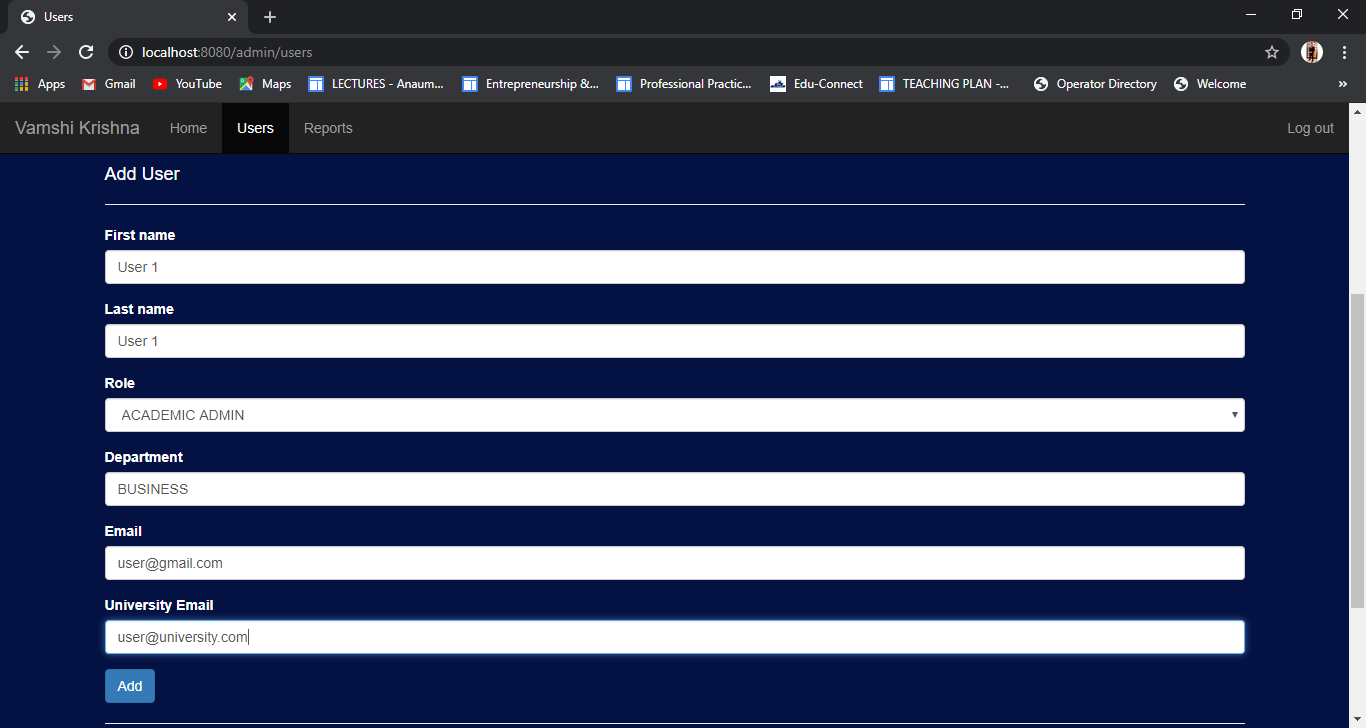
**4) Test Users:**

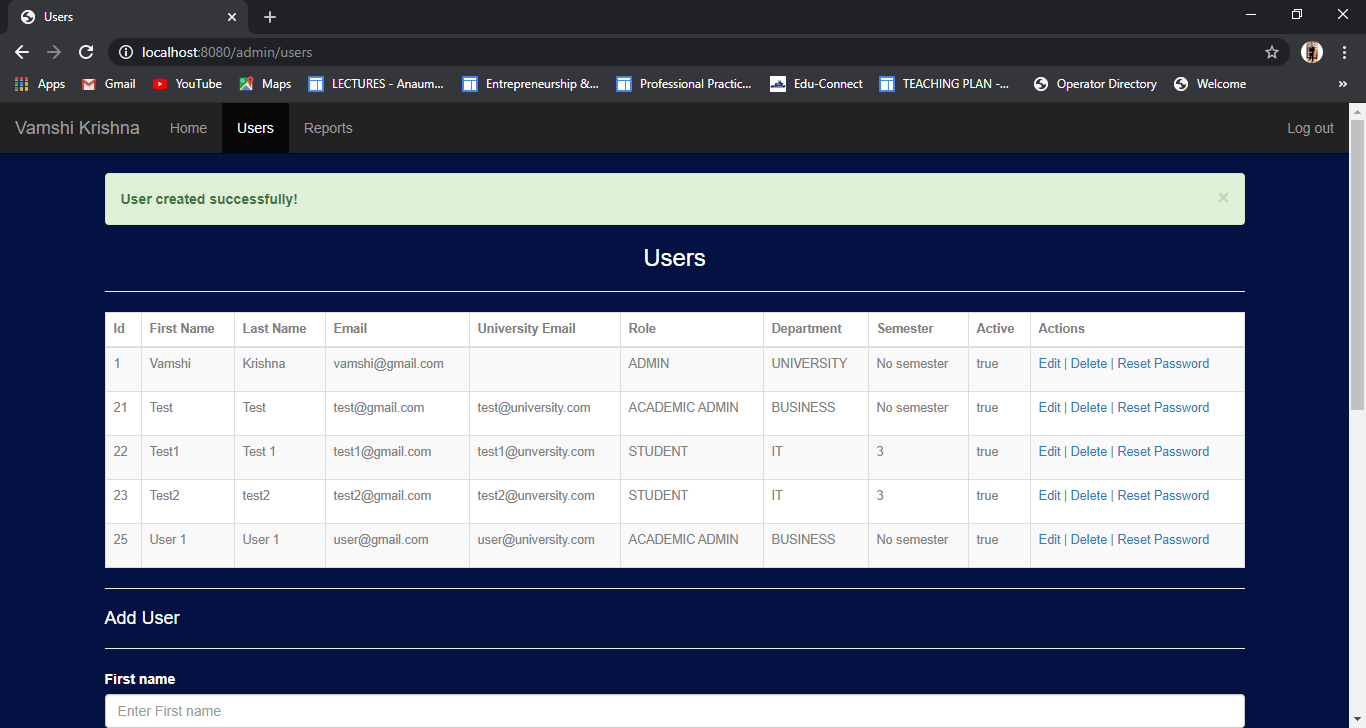
|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| ID | Scenario | Preconditions | Steps involved | Expected Result | Actual Result | Status |
| 1 | Admin can view all the users | Admin is logged in to the admin dashboard. | Login ad admin.  Go to users page. | All Users visible to admin. | All Users visible to admin. | **Passed** |
| 2 | Admin can add a single user. | Admin is logged in to the admin dashboard. | Login as admin.  Add user details.  Click on the add button. | User created successfully. | User created successfully | **Passed** |
| 3 | Admin can add multiple users. | Admin is logged in to the admin dashboard. | Login as admin.  Select the csv file with proper multiple user details.  Click on the upload button. | User created successfully. | Username/Password/Role incorrect | **Passed** |
| 4 | Admin can delete a user | Admin is logged in to the admin dashboard. | Login as admin.  Click on delete button for a user in users table. | User deleted! | User deleted! | **Passed** |

1. **Admin can view all users**

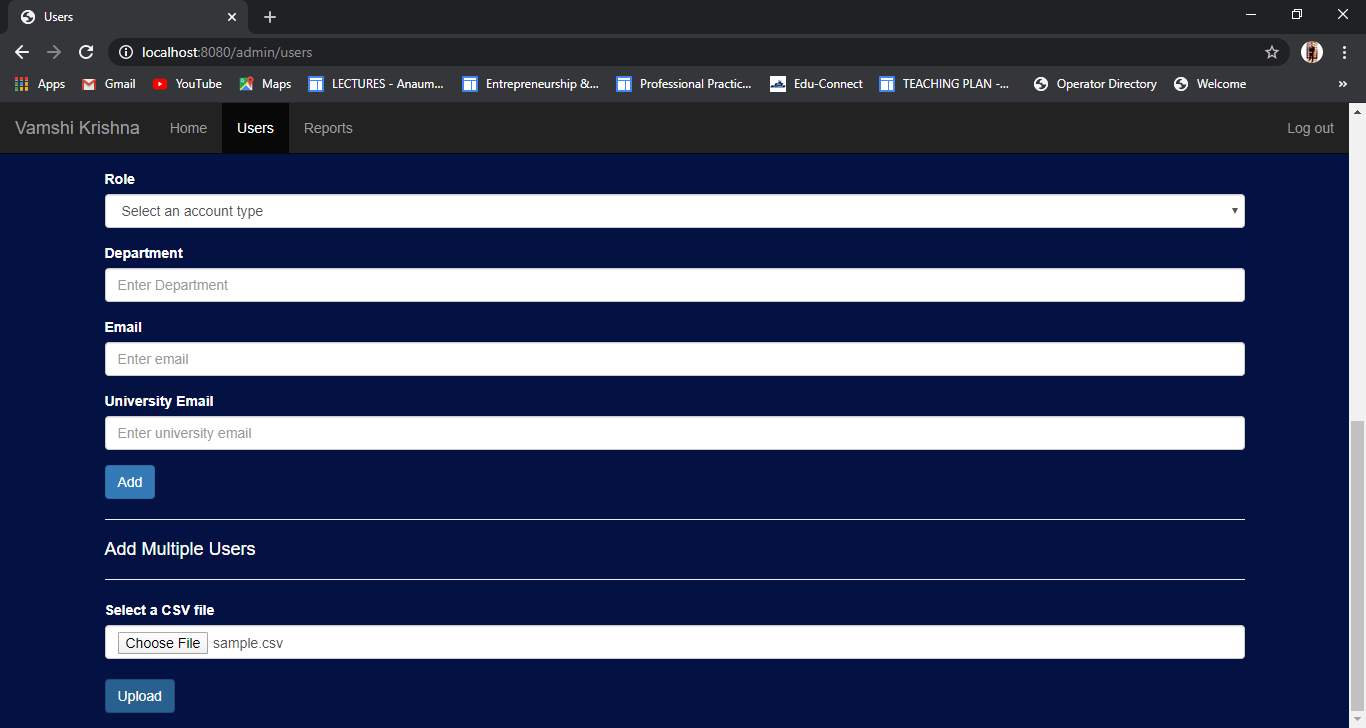


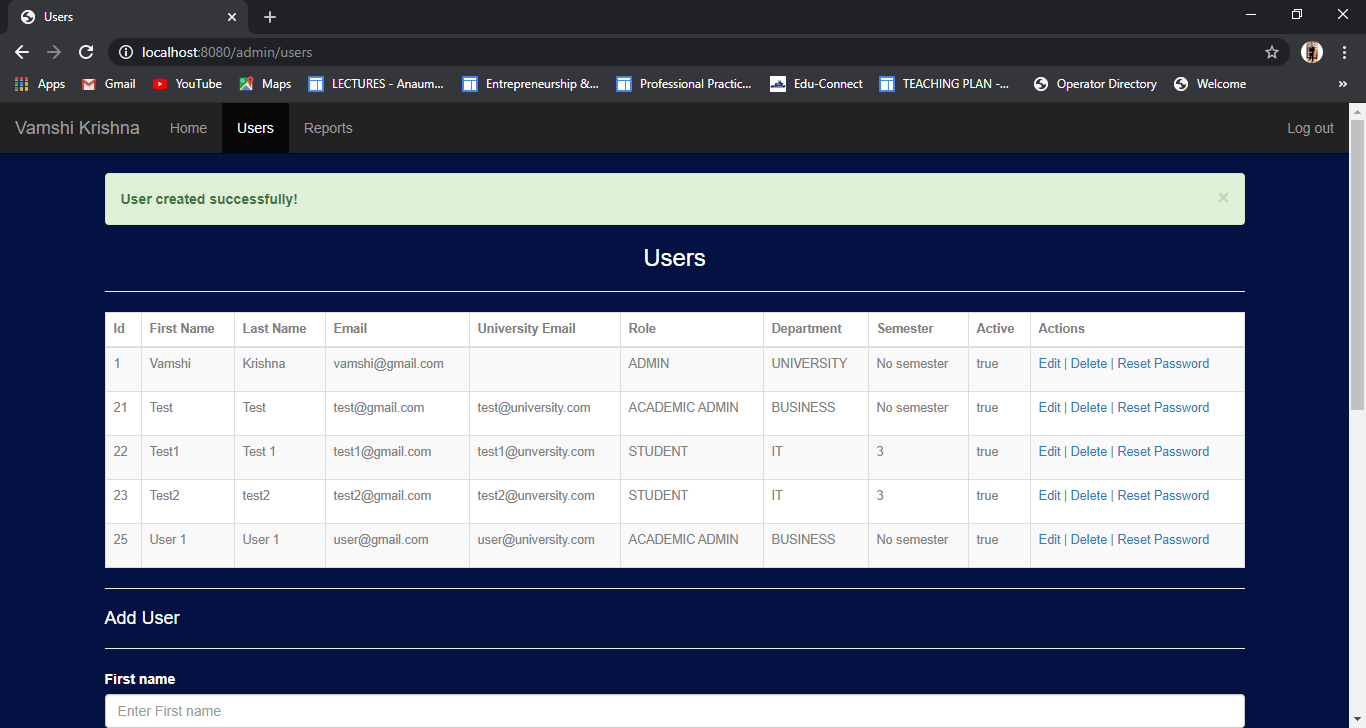
1. **Admin can add a single user**



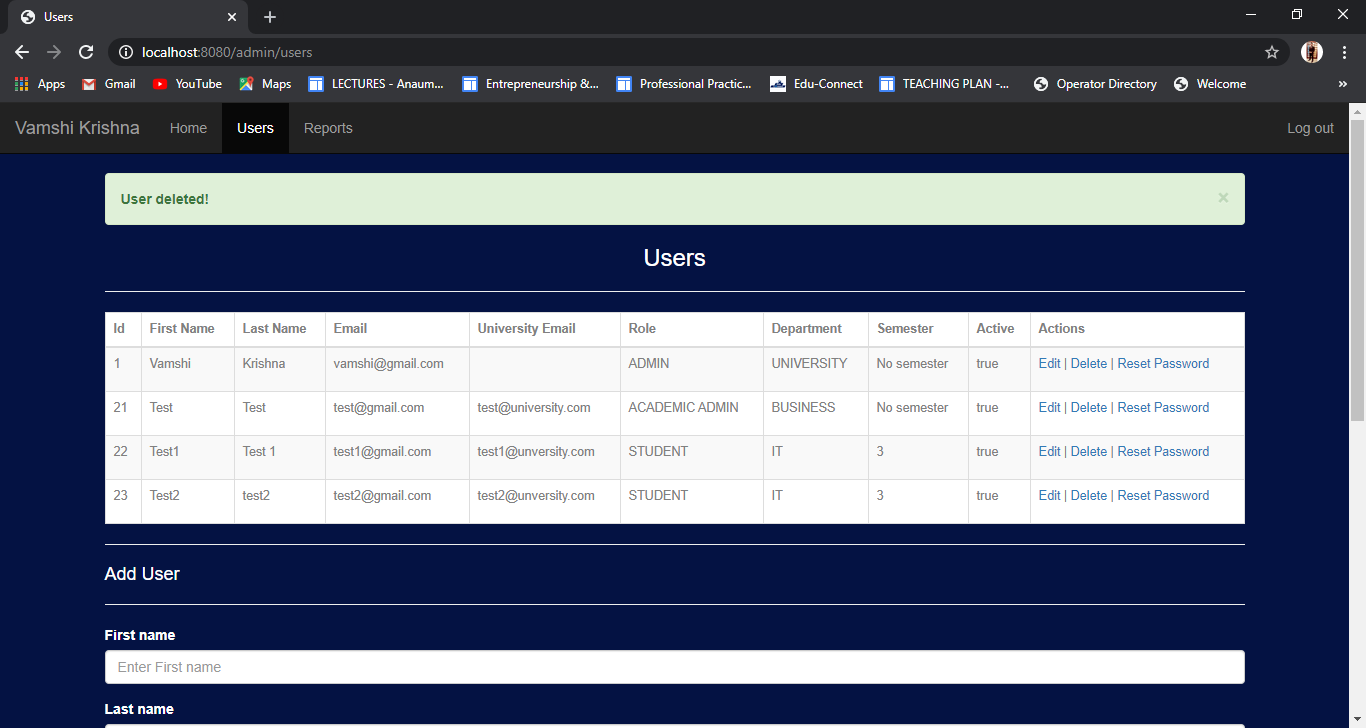


1. **Admin can add multiple users**





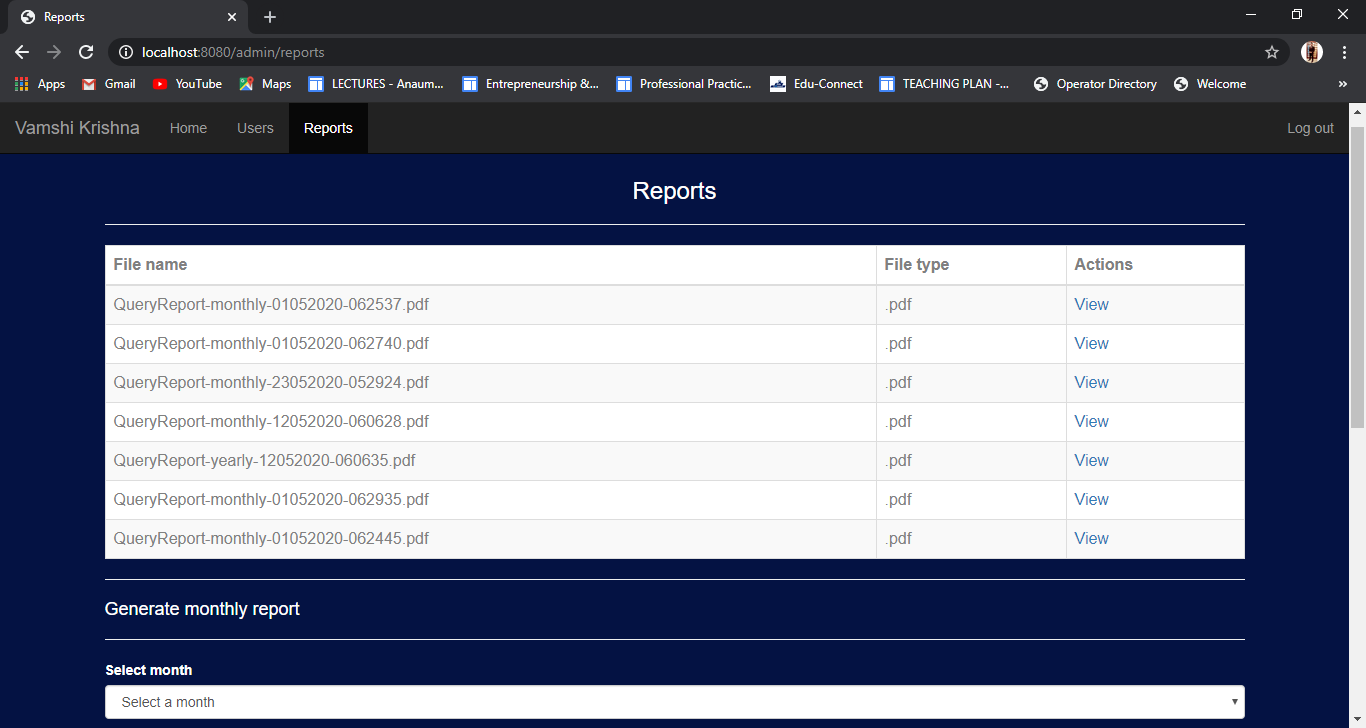
1. **Admin can delete users**



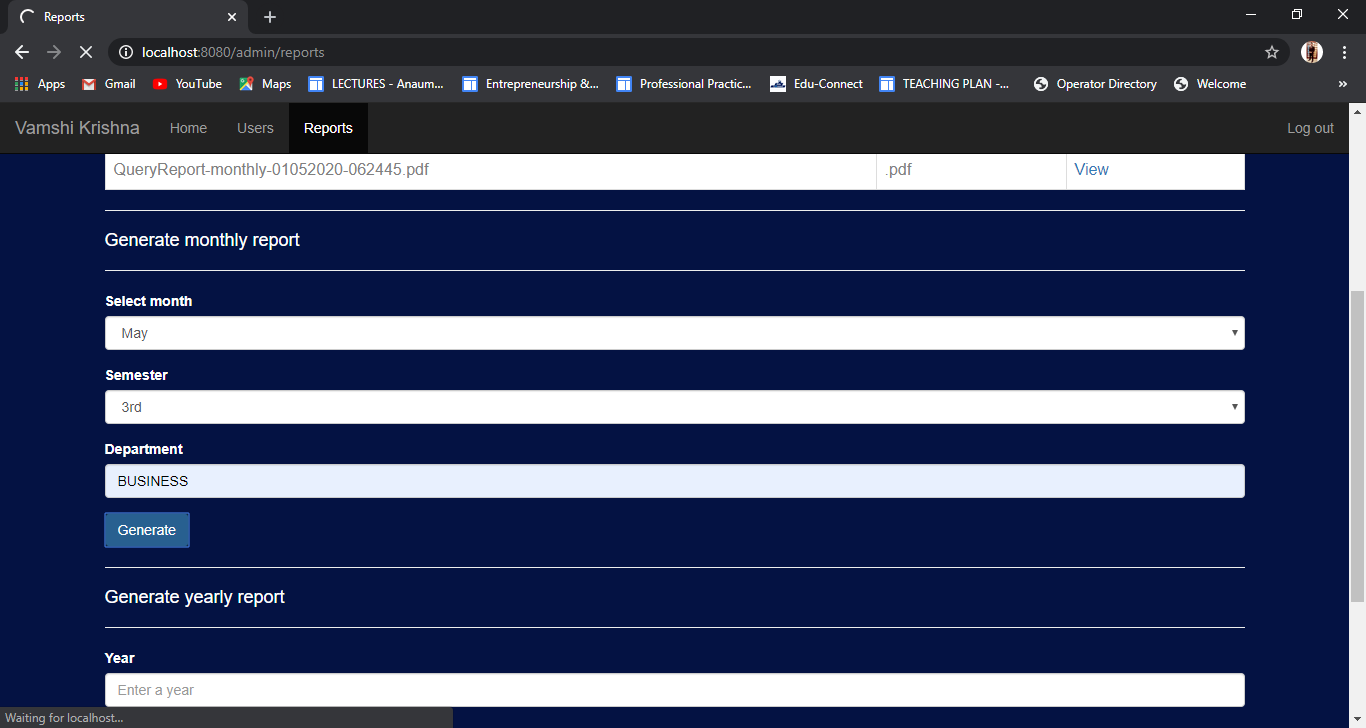
1. **Test Admin Reports:**

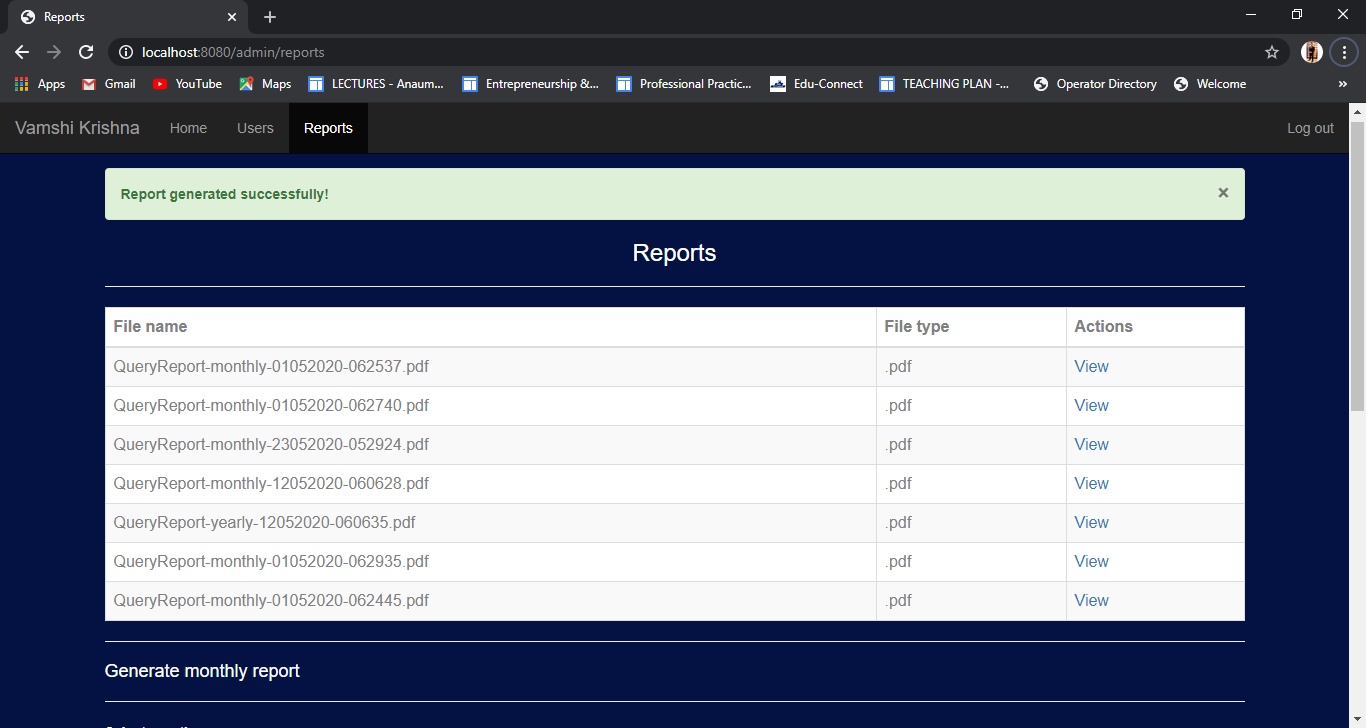
|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| ID | Scenario | Preconditions | Steps involved | Expected Result | Actual Result | Status |
| 1 | Admin can view previous generated reports. | Admin is logged in to the system. | Login as admin to the system.  Go to Reports page. | Admin can view all reports. | Admin can view all reports. | **Passed** |
| 2 | Admin can generate monthly query report. | Admin is logged in to the system. | Login as admin to the system.  Select the semester, department and month.  Click on generate button. | Report generated successfully. | Report generated successfully | **Passed** |
| 3 | Admin can generate yearly query report. | Admin is logged in to the system. | Login as admin to the system.  Select the semester, department and year.  Click on generate button. | Report generated successfully. | Report generated successfully. | **Passed** |

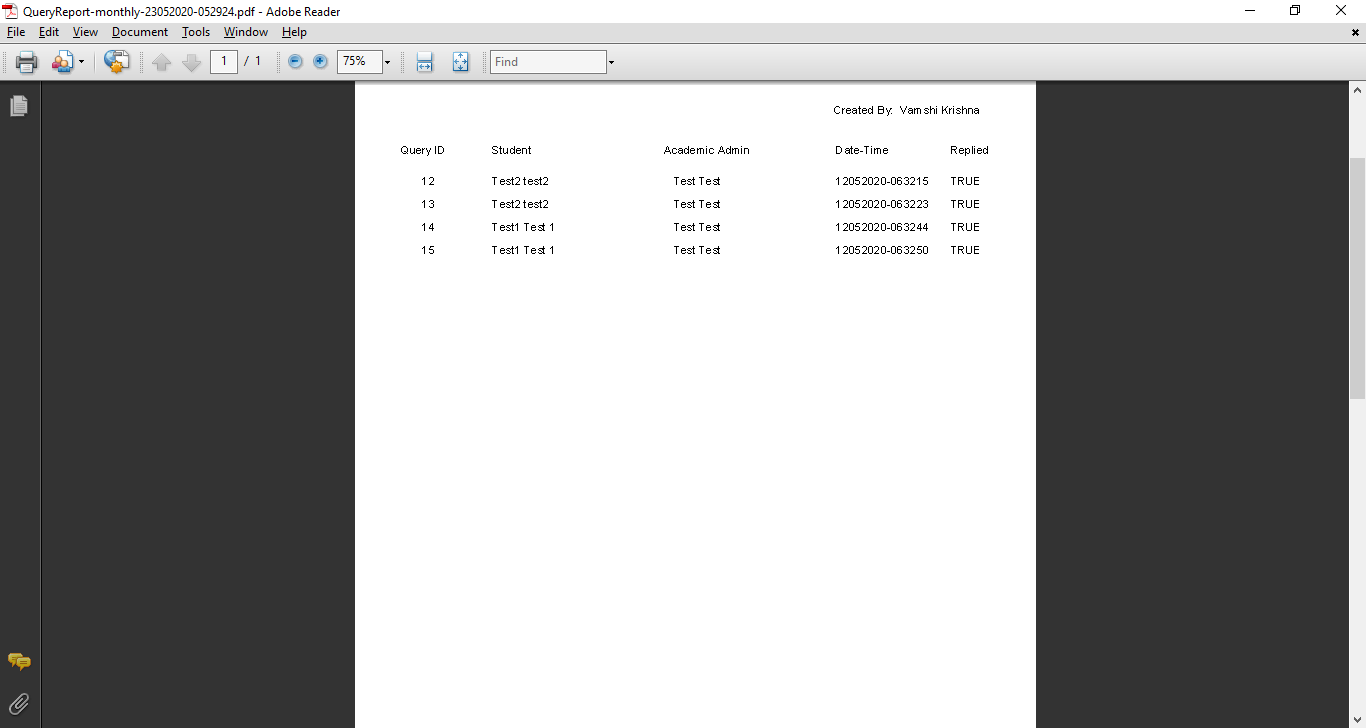
1. **Admin can view previous generated reports**



1. **Admin can generate monthly report**







1. **Admin can generate yearly report**

