



Classync - Virtual Classroom Platform

GUI Testing Report

Group - 33

Course: Software Engineering

Professor: Prof. Saurabh Tiwari

Mentor: Shyam Patel

Team Members:

Role	Name (ID)
Team Leader	ATIK VOHRA - 202301447
Member	KANANI TANUJ - 202301474
Member	SANAGADHIYA RADHIKA - 202301184
Member	PATEL JAY - 202301430
Member	CHOVATIYA AYUSH - 202301461
Member	OM PATEL - 202301163
Member	BADRESIYA ANSH - 202301477
Member	PREM KUKADIYA - 202301452
Member	RAMOLIYA SHUBHAM - 202301442
Member	GAJERA MANTHAN - 202301488

Objectives

The objective of GUI testing for the Virtual Classroom Platform is to ensure that all graphical user interface components function as intended and provide a seamless user experience. This includes verifying:

- **Responsiveness:** Ensuring the platform's UI adapts properly to various screen sizes and orientations.
 - **Navigation:** Confirming smooth transitions between screens and accurate functionality of navigation elements like buttons, tabs, and menu items.
 - **Functionality:** Testing the usability of features such as chat, screen sharing, whiteboard, AI-tools, submission, authentication, enrollment.
 - **Visual Design:** Checking for consistent alignment, color schemes, fonts, and overall aesthetic appeal.
 - **Error Handling:** Ensuring that appropriate error messages are displayed for invalid actions.
 - **Performance:** Verifying that the UI responds quickly and without lags under normal and heavy usage scenarios.
-

Major UI Components in the App

- Sign Up
 - Login
 - Forgot Password
 - Join Course
 - Submit Assignment
 - Chat Space
 - Notifications
 - WhiteBoard
 - AI-Tools
 - View Progress
 - Logout
-

Test Scenarios

1) Sign Up

Test Case ID: TC_001

Test Objective: Verify that the sign-up functionality works correctly and displays all necessary validations.

Steps:

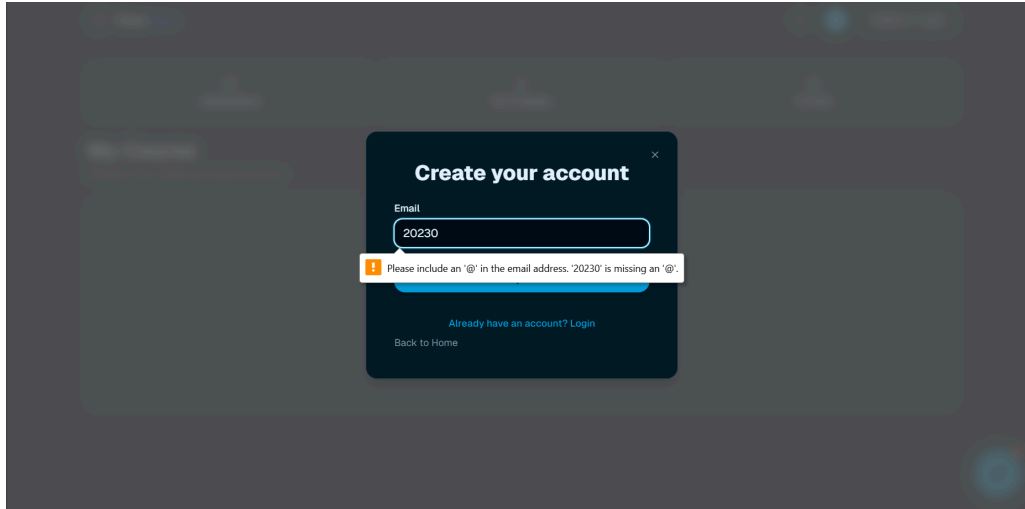
1. Opened App: Navigated to <https://se-virtual-classroom-platform.vercel.app/>
2. Valid Inputs: Entered valid name, email, and password; clicked Sign Up.
3. Invalid Inputs:
 - Empty fields: Error messages displayed.
 - Invalid email: "Invalid email format" error.
4. Button Behavior: Sign Up disabled for invalid inputs; enabled for valid inputs.
5. Navigation: Verified Log In link redirects correctly.
6. Usability: Password toggle works; elements aligned properly.
7. Error Handling: Errors cleared after corrections.

Expected Result:

- On valid input, the user should be successfully registered and redirected to the dashboard or login screen.
- On invalid input, appropriate error messages should be displayed.

Actual Result:

Invalid input response-



Video Link:[link](#)

Pass/Fail: PASS

2) Log in

Test Case ID: TC_002

Test Objective: Verify that the login functionality works correctly with valid and invalid credentials.

Steps:

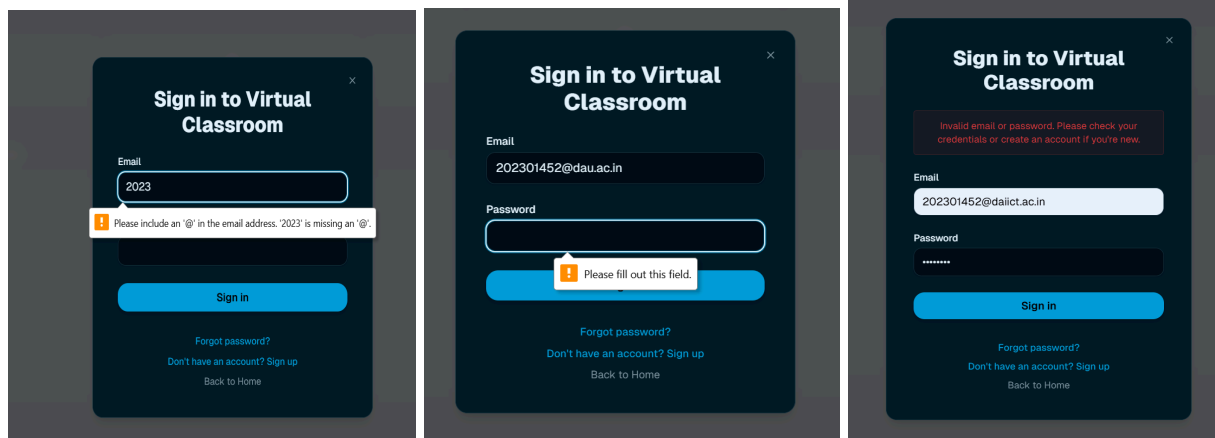
1. Opened App: Navigated to the Login screen.
2. Valid Inputs: Entered valid email and password; clicked Login; if password is correct redirected to Dashboard.
3. Invalid Inputs:
 - Empty fields: Error messages displayed.
 - Invalid email: "Invalid email format" error.
 - Wrong password: "Incorrect password" error.
4. Button Behavior: Login disabled for invalid inputs; enabled for valid inputs.
5. Navigation: Verified Login button redirects correctly.
6. Error Handling: Errors cleared after corrections.

Expected Result:

- On valid input, the user should be successfully logged in and redirected to the dashboard.
- On invalid input, an appropriate error message should be displayed.

Actual Result:

Invalid inputs response-



Video Link:[link](#)

Pass/Fail: PASS

3) Forget Password

Test Case ID: TC_003

Test Objective: Verify that the "Forgot Password" functionality allows users to reset their password correctly via Email.

Steps:

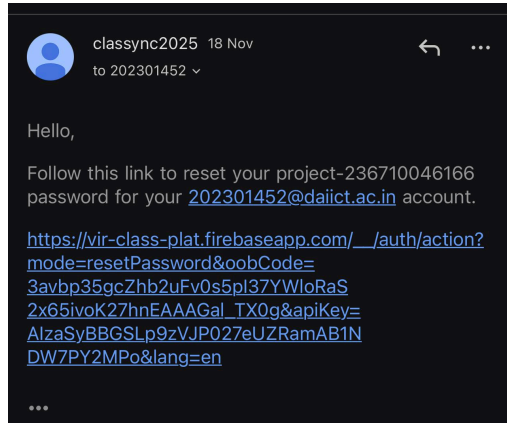
1. Opened App: Navigated to Login screen; clicked "Forgot Password?" and entered a registered email.
2. Valid Inputs: Reset link sent to email; updated password successfully; logged in with the new password.
3. Button Behavior: Submit button disabled for invalid inputs; enabled for valid ones.
4. Navigation: Verified flow from "Forgot Password" to reset and login screens.

Expected Result:

- On valid input, a reset password email should be sent, and a confirmation message (e.g., "Reset link sent to your email") should appear.

Actual Result:

Reset link sent-



Video link:[link](#)

Pass/Fail: PASS

4) View Progress

Test Case ID: TC_004

Test Objective: Verify the functionality, UI elements, and data accuracy on the View Progress page for a student.

Steps:

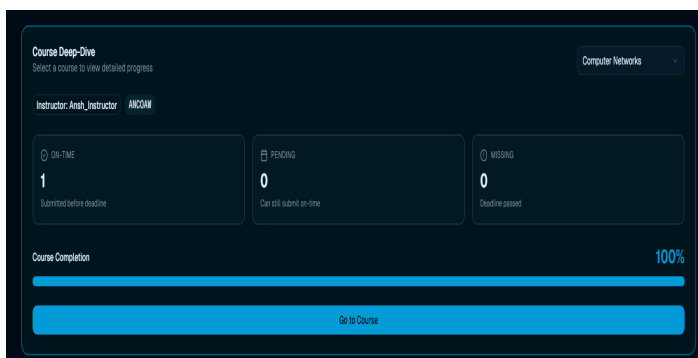
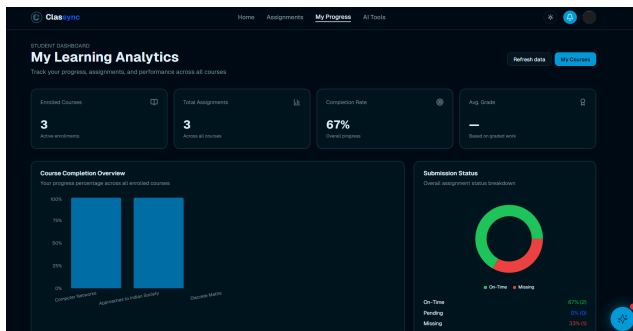
1. Opened App: Navigated to Dashboard; clicked "View Progress".
2. Select Course: Choose a course from the dropdown (e.g., Computer Networks).
3. Load Progress: Verified that course statistics (On-Time, Pending, Missing) load correctly.
4. Instructor Details: Confirmed instructor name and course code are displayed.
5. Progress Bars: Checked course completion percentage and progress bar rendering.
6. Navigate Course: Clicked "Go to Course" and verified navigation to the respective course page.
7. Dynamic Update: Changed course selection and confirmed values update dynamically without page refresh.

Expected Result:

- Progress metrics should load correctly for the selected course.
- Instructor information and course code should be accurate.
- The progress bar should reflect the correct completion percentage.
- "Go to Course" should navigate to the appropriate course.

- Dropdown change should instantly refresh stats without errors.

Actual Result:



Video Link:[link](#)

Pass/Fail:PASS

5) Join Course

Test Case ID: TC_005

Test Objective: Verify the functionality and UI of the "Join Course" screen.

Steps:

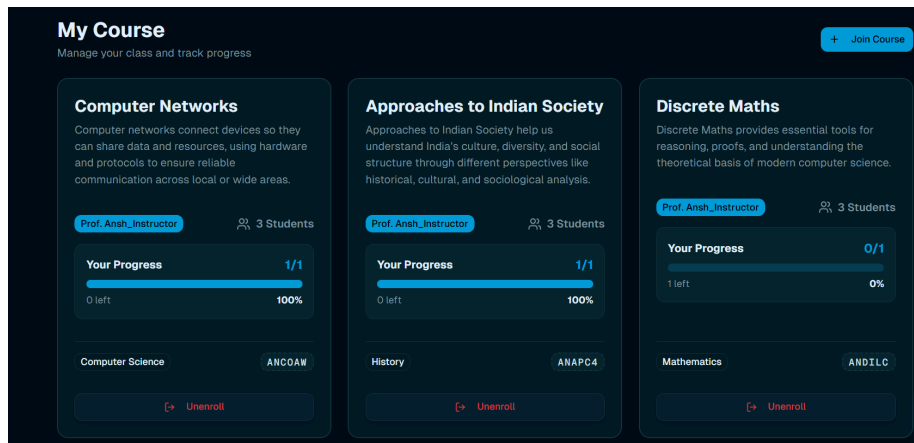
1. Opened App: Navigated to Dashboard; clicked "Join Course".
2. Valid Inputs: Entered classroom id
3. Join Classroom: Clicked "Join" button.
4. VerifyJoin: Confirmed classroom appears in the dashboard classroom list.
5. View Details: Verified ability to view classroom details after joining.
6. Error Handling: Confirmed error messages for invalid inputs (e.g., invalid id).
7. Navigation: Verified smooth navigation from join to classroom view.

Expected Result:

- On valid input, the classroom should be successfully created and appear in the classroom list.

Actual Result:

Showing courses-



Video Link:[link](#)

Unenroll from course-[link](#)

Pass/Fail: PASS

6) AI-Chatbot

Test Case ID: TC_006

Test Objective: Verify the functionality of the "Ask a Question" feature including input fields, submission, and response handling.

Steps:

1. Opened App: Navigated to the "Ask a Question" section from the dashboard.
2. Enter Topic: Typed a valid topic into the topic field (e.g., Physics, Math, History).
3. Enter Question: Enter a detailed question in the description text area.
4. Submit Query: Clicked the "Get Answer" button.
5. Verify Response: Confirmed that the system generates an answer below the form.
6. History Check: Clicked "Show History" to verify that previous questions and answers are displayed.

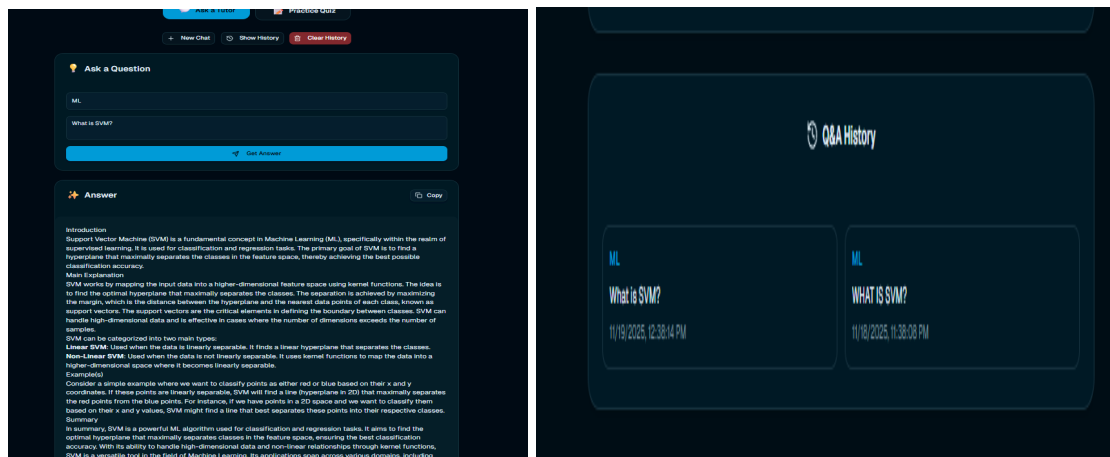
7. Clear History: Clicked "Clear History" and confirmed that history is removed.
8. New Chat: Used the "New Chat" button to reset fields and start a fresh session.
9. Error Handling: Verified that empty fields trigger appropriate validation messages.

Expected Result:

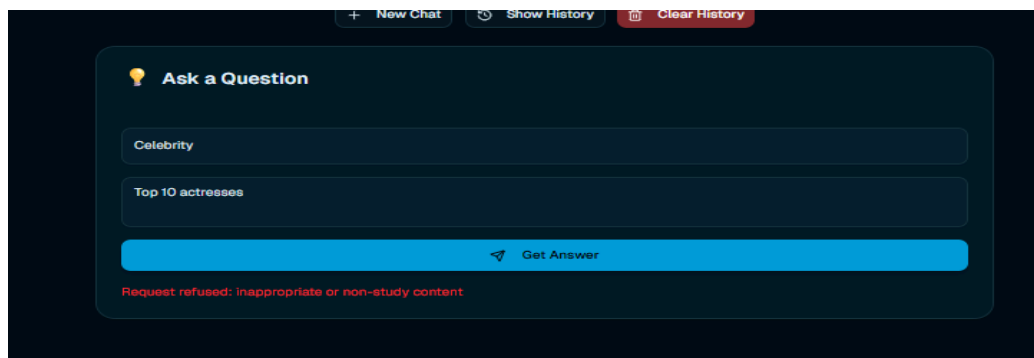
- Users should be able to enter a topic and a question smoothly.
- The "Get Answer" button should generate a valid response.
- History should display all previous queries and answers.

Actual Result:

Minor Issue- **There is a delay in fetching the answer.**



Invalid Input(Like celebrity, drugs turned off) response-



Video link:[link](#)

Pass/Fail:PASS

7) AI-Quiz Generator

Test Case ID: TC_007

Test Objective: Verify the functionality of generating quizzes based on topic, question type, grade level, and number of questions.

Steps:

1. Opened App: Navigated to the "Generate Quiz" section from the dashboard.
2. Enter Topic: Typed a valid topic into the quiz topic field.
3. Select Type: Choose the quiz type (MCQ or True/False).
4. Select Grade: Selected the grade level from the dropdown (e.g., Grade 9).
5. Select Count: Choose the number of questions from the dropdown.
6. Generate Quiz: Clicked the "Generate Quiz" button.
7. Verify Quiz: Confirmed that a quiz is generated with the selected number of questions.
8. History Check: Clicked "Show History" and verified previously generated quizzes are listed.
9. New Quiz: Clicked "New Quiz" to reset inputs for generating another quiz.
10. Error Handling: Verified that missing inputs trigger proper validation messages.

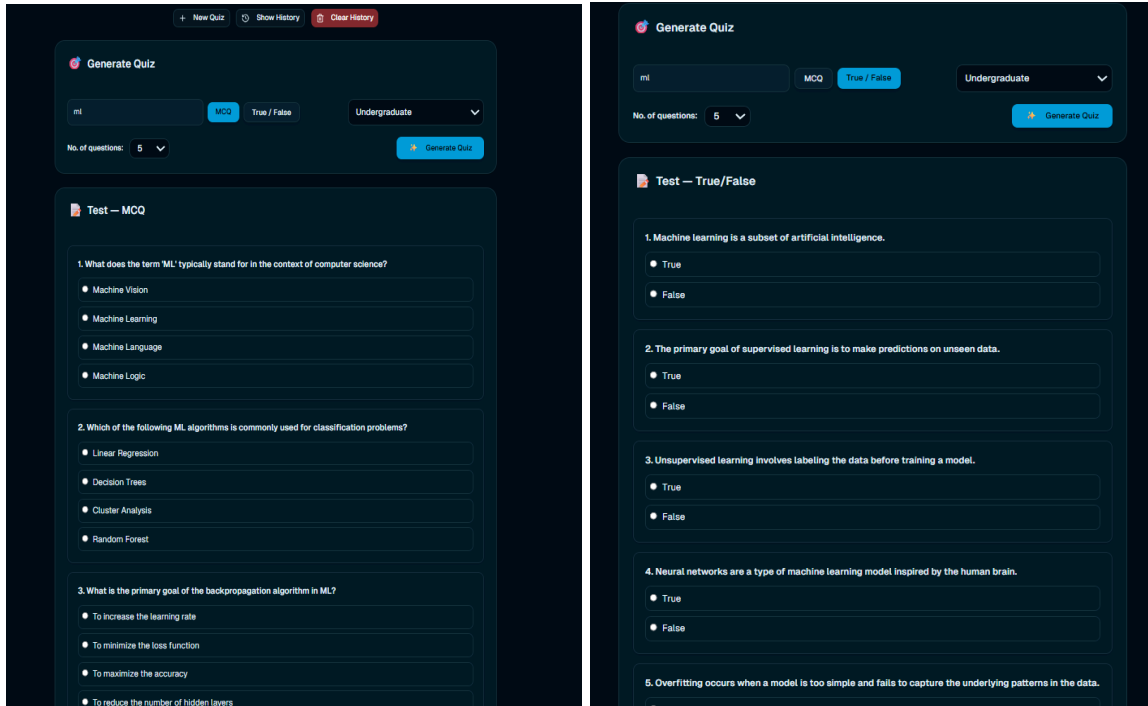
Expected Result:

- Quiz should be generated based on the entered topic, selected type, grade, and number of questions.
- Generated quizzes must contain accurate and complete questions.
- History should correctly show previously generated quizzes.
- "New Quiz" should reset all fields.
- Empty or invalid inputs should show error messages.

Actual Result:

True or False-

MCQ-



Video Link:[link](#)

Pass/Fail: PASS

8) AI Assistant

Test Case ID: TC_008

Test Objective: Verify the functionality of the AI Assistant chat widget, including message input, response generation, and UI behavior.

Steps:

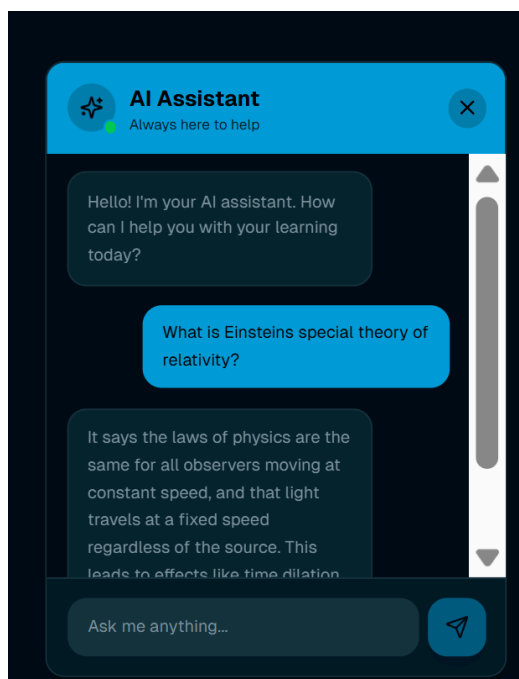
1. Opened App: Navigated to the dashboard and opened the AI Assistant widget.
2. Initial Message: Verified that the assistant displays a welcome message.
3. Enter Query: Typed a valid question into the input field.
4. Send Message: Clicked the send icon to submit the query.
5. Verify Response: Confirmed that the assistant replies with a relevant response.
6. Chat Continuity: Asked follow-up questions and verified conversation consistency.
7. Close Assistant: Clicked the close icon and confirmed the widget closes smoothly.
8. Reopen Assistant: Opened the widget again to verify previous chat state or reset behavior.
9. Error Handling: Verified validation message when attempting to send an empty message.

Expected Result:

- AI Assistant should display a welcome message on launch.
- User inputs should be accepted and processed without delay.
- Responses should appear correctly in the chat window.
- The assistant should maintain conversation flow across multiple queries.
- Close and reopen actions should work smoothly.
- Empty messages should not be allowed.

Actual Result:

Successful short answer displayed-



Video Link:[link](#)

Pass/Fail:PASS

9) AI-Summary Generator

Test Case ID: TC_009

Test Objective: Verify the functionality of the summary generator, including file upload, grade/branch input, summary type selection, and summary generation.

Steps:

1. Opened App: Navigated to the "Generate Summary" section from the dashboard.
2. Upload File: Clicked "Choose File" and selected a valid document (PDF, DOCX, TXT, PPTX).
3. Enter Grade/Branch: Typed the grade or branch details into the input field (e.g., 10th Grade, Biology).
4. Select Summary Type: Choose a summary format from the dropdown (e.g., Quick overview – 5 min read).
5. Generate Summary: Clicked the "Generate Summary" button.
6. Verify Output: Confirmed that the system processes the file and generates a summary.
7. History Check: Verified that previously generated summaries appear in the history (if available).
8. Error Handling: Attempted to submit without uploading a file or without entering required details to confirm proper validation messages.

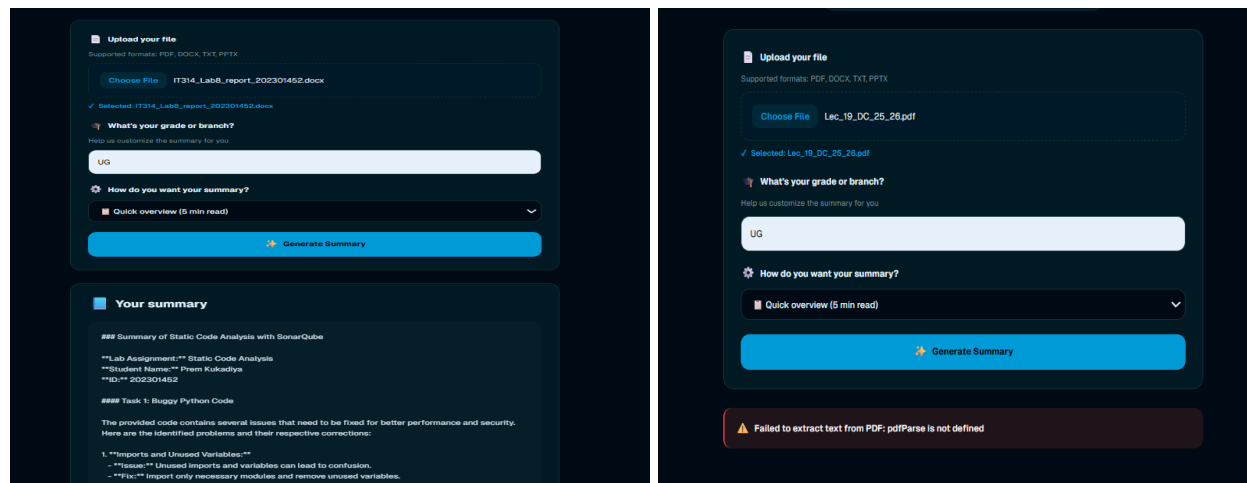
Expected Result:

- The system should accept supported file formats and display the selected filename.
- Summary should be generated based on the uploaded document and selected preferences.
- Dropdown should allow choosing the summary format.
- Validation messages should appear for missing or invalid inputs.
- Output summary should load without errors.

Actual Result: ~~Issue~~ Only generates summary of Word Document.

Successful summary for word document-

Not generating for PDF-



Video Link: [link](#)

Pass/Fail: **FAIL**

10) Assignment Submission

Test Case ID: TC_010

Test Objective: Verify the functionality and UI elements of the Assignment Submission feature for students.

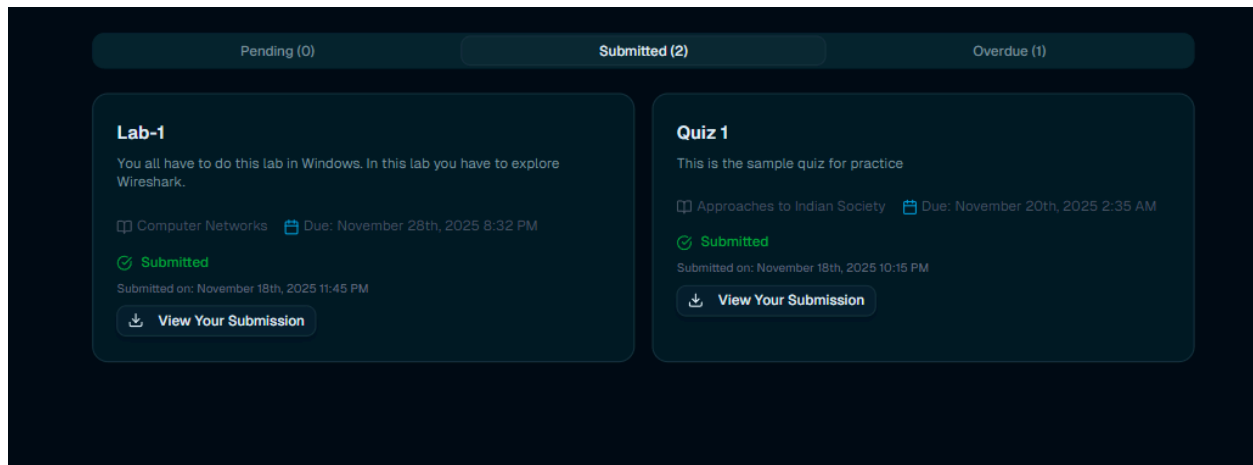
Steps:

1. Open Classroom: Logged into the app → navigated to the classroom → opened the Assignments section.
2. Open Assignment: Selected a specific assignment from the list → viewed assignment details such as title, instructions, and due date.
3. Upload File: Clicked "Upload" → selected a file from the system → verified file name appears in the upload preview area.
4. Submit Assignment: Clicked "Submit" button → confirmed submission popup (if any) → checked if the system shows "Submitted Successfully".
5. Error Handling: Tried uploading unsupported file format / oversized file → validated proper error message appears.
6. View Submission Status: Checked if the submission status updates → displays filename, timestamp, and submission status (Submitted / Pending).

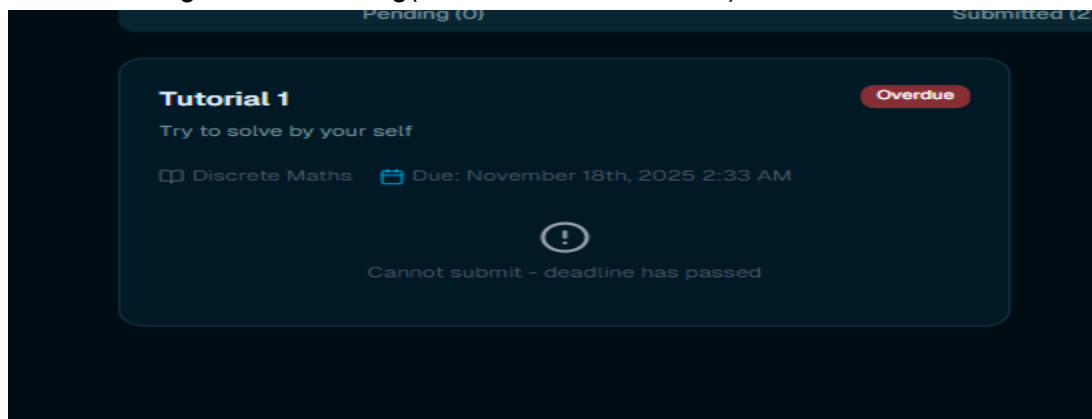
Expected Result:

- Assignment details (title, instructions, due date) load correctly.
- Upload area should allow acceptable file types (PDF/Image/Docs).
- After clicking Submit, the assignment should show the Submitted Successfully message.
- Submitted assignment should display:
 - File name
 - Submission timestamp
 - Status indicator
- Invalid uploads should show clear error messages.

Actual Result: Successfull submission showing-



Overdue Assignments showing(Also cannot submit now)-



Assignment dashboard video-[link](#)

Video Link:[link](#)

Pass/Fail:PASS

11) Chat Space

Test Case ID: TC_011

Test Objective: Verify the functionality and UI elements of the Chat feature during a classroom session.

Steps:

1. Opened App: Joined a classroom; opened chat panel.

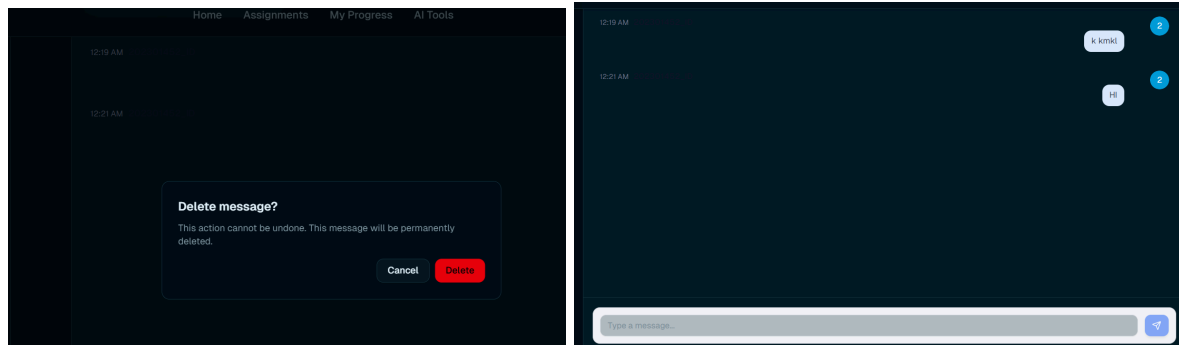
2. Send Message: Typed a message in the chat box; clicked Send; verified message appears in chat window.
3. Error Handling: Make sure messages can be deleted

Expected Result:

- Sent messages should appear instantly, and incoming messages should load dynamically.
- Chat should display sender names and timestamps.
- Delete functionality works.

Actual Result:

Delete Functionality and showing message details-



Video Link:[link](#)

Pass/Fail: PASS

12) Whiteboard Feature

Test Case ID: TC_012

Test Objective: Verify the functionality and UI elements of the Whiteboard feature.

Steps:

1. Opened App: Joined a classroom; opened a whiteboard tool.
2. Drawing Tools:
 - Selected pen tool; drew on whiteboard.
 - Selected shapes (circle, rectangle, line); confirmed they render correctly.
 - Selected text tool; added text.
3. Color Selection: Changed colors for different elements; verified color changes apply.
4. Eraser Tool: Used eraser to remove content; confirmed erasing works.

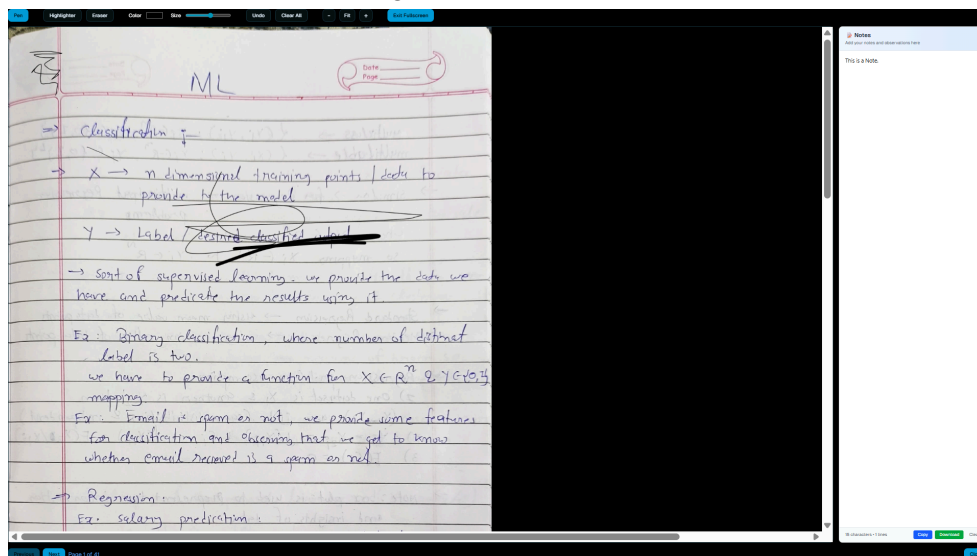
5. Clear Board: Clicked "Clear All" button; verified board cleared completely.
6. Undo/Redo: Tested undo and redo functionality; confirmed actions reverse/restore correctly.
7. Collaboration: Verified multiple users can draw simultaneously; changes appear in real-time.
8. Save/Export: Tested saving or exporting whiteboard content (if applicable).
9. Error Handling: Confirmed smooth performance with multiple elements.
10. Cross-Browser Testing: Tested whiteboard on different browsers.

Expected Result:

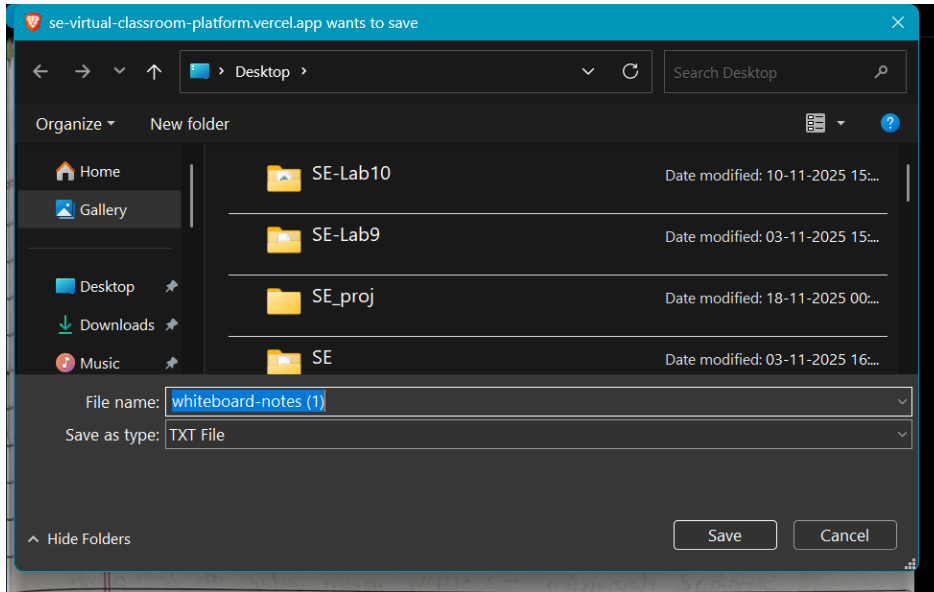
- All drawing tools should work smoothly.
- Multiple users should be able to collaborate in real-time.
- Undo/redo should work correctly.
- The whiteboard should remain responsive with multiple elements.
- Clear board should remove all content immediately.

Actual Result:

Successful whiteboard working-



Notes successfully downloadable-



Video Link:[link](#)

Pass/Fail: PASS

13) User Profile and Update Profile

Test Case ID: TC_013

Test Objective: Verify the functionality and UI elements of the User Profile and Settings screen.

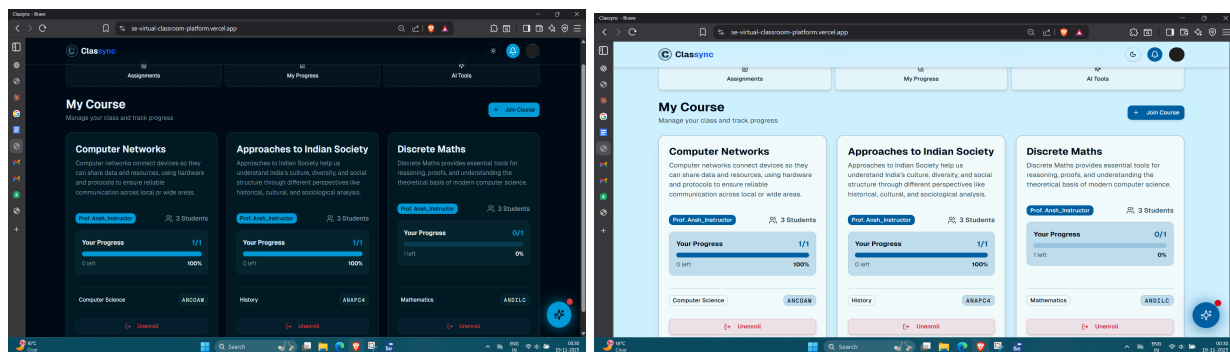
Steps:

1. Opened App: Logged in; navigated to Profile/Settings by clicking profile icon.
2. View Profile: Verified display of user information (name, email, profile picture, bio).
3. Edit Profile:
 - Clicked the "Edit Profile" button.
 - Updated name and bio.
 - Uploaded a new profile picture by selecting from the gallery.
 - Saved changes.
 - Verified updates reflected immediately.
4. Change Password:
 - Navigated to "Change Password".
 - Entered current password and new password.
 - Confirmed password change; verified password updated successfully.
5. Theme Selection: Changed UI theme (light/dark mode); theme applied immediately.
6. Navigation: Verified smooth navigation back to dashboard.

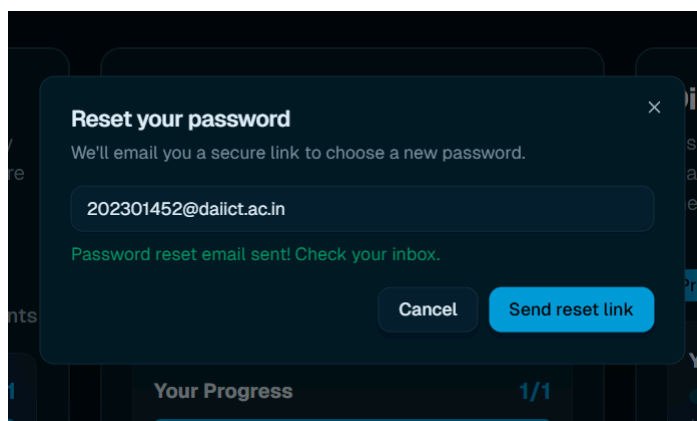
Expected Result:

- The user's profile picture, name, status, and bio should be displayed correctly.
- Clicking on the profile picture should allow changing or viewing the full-size image (if applicable).
- The "Edit Profile" button should allow the user to update their details, and changes should be saved successfully.
- The "Change Password" functionality should navigate to the password change screen and allow for successful password updates.
- Additional options like "Logout," "Delete Account," should work as expected.
- Profile updates (e.g., new picture or bio) should reflect immediately after saving.

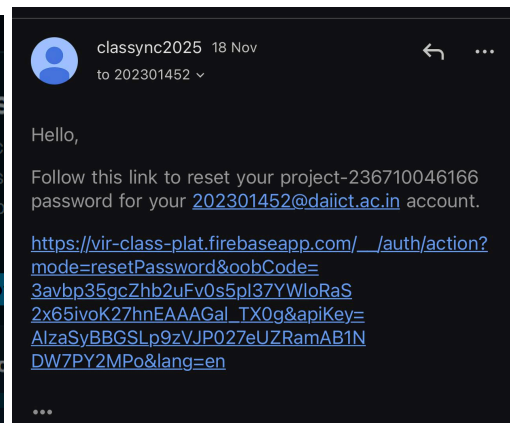
Actual Result: Light mode vs Dark mode -



Change password UI



Reset link sent-



Video Link:

Change username-[link](#)

Add profile picture-[link](#)

Pass/Fail: PASS

14) Logout

Test Case ID: TC_014

Test Objective: Verify that the logout functionality works correctly.

Steps:

1. Opened App: Logged in successfully to the platform.
2. Locate Logout: Found logout button in profile dropdown menu or settings.
3. Click Logout: Clicked logout button.
4. Confirmation: Verified logout confirmation prompt appears (if applicable); clicked "Yes" to confirm.
5. Session Cleared: Confirmed user redirected to login screen.
6. Multiple Tabs: Opened platform in multiple browser tabs; clicked logout in one tab; verified all tabs logged out.
7. Navigation: Verified cannot access protected pages without logging in again.

Expected Result:

- Users should be logged out successfully with the session cleared completely.
- Users should be redirected to the login page immediately.
- Cannot access protected pages without logging in again.
- The browser back button should not allow access to authenticated pages.
- All active sessions should be terminated.

Actual Result:

Successfully log's out of all tab if logging out from one tab- [link](#)

Video Link:[link](#)

Pass/Fail: PASS

Observations and Summary

Test Case ID	Test Case Name	Status	Observations
TC_001	Sign Up	PASS	All form validations functioning properly. Password toggle feature works well. Error messages clear on empty/invalid fields.
TC_002	Log In	PASS	Login redirects to the dashboard on correct credentials. Error handling for wrong password and invalid email format working as expected.
TC_003	Forget Password	PASS	Reset link successfully sent to registered email. Password update and subsequent login with new credentials verified.
TC_004	View Progress	PASS	Course statistics (On-Time, Pending, Missing) displaying correctly. Dropdown updates dynamically without page refresh. Progress bars rendering accurately.
TC_005	Join Course	PASS	Classroom successfully appears in the dashboard after joining. Invalid classroom ID shows appropriate error message. Navigation smooth throughout.
TC_006	AI-Chatbot	PASS	Noticeable delay observed while fetching answers. History feature working correctly. Invalid inputs (celebrity names, drugs) properly restricted.
TC_007	AI-Quiz Generator	PASS	Both MCQ and True/False quiz types generate successfully. Quiz history maintained properly. New quiz button resets all fields correctly.
TC_008	AI Assistant	PASS	Welcome message displays on launch. Response generation working with good conversation flow. Empty message validation in place.
TC_009	AI-Summary Generator	FAIL	PDF files not processing - only Word documents generate summaries successfully. Feature advertises support for PDF, DOCX, TXT, PPTX but PDF functionality not working.

TC_010	Assignment Submission	PASS	File upload and submission working correctly. Overdue assignments clearly marked and submission disabled as expected. Submission status displayed with filename and timestamp.
TC_011	Chat Space	PASS	Messages appear instantly. Delete functionality tested and working. Sender names and timestamps displayed correctly.
TC_012	Whiteboard Feature	PASS	All drawing tools (pen, shapes, text) are functional. Real-time collaboration verified with multiple users. Notes successfully downloadable. Undo/redo working properly.
TC_013	User Profile and Update Profile	PASS	Light/dark mode toggle working instantly. Profile picture and username update reflected immediately. Password change functionality verified with reset link sent.
TC_014	Logout	PASS	Session cleared successfully across all open tabs when logging out from one. Redirect to the login screen immediately. Protected pages inaccessible after logout.

Summary Statistics

- **Total Test Cases:** 14
- **Passed:** 13
- **Failed:** 1
- **Pass Rate:** 92.86%

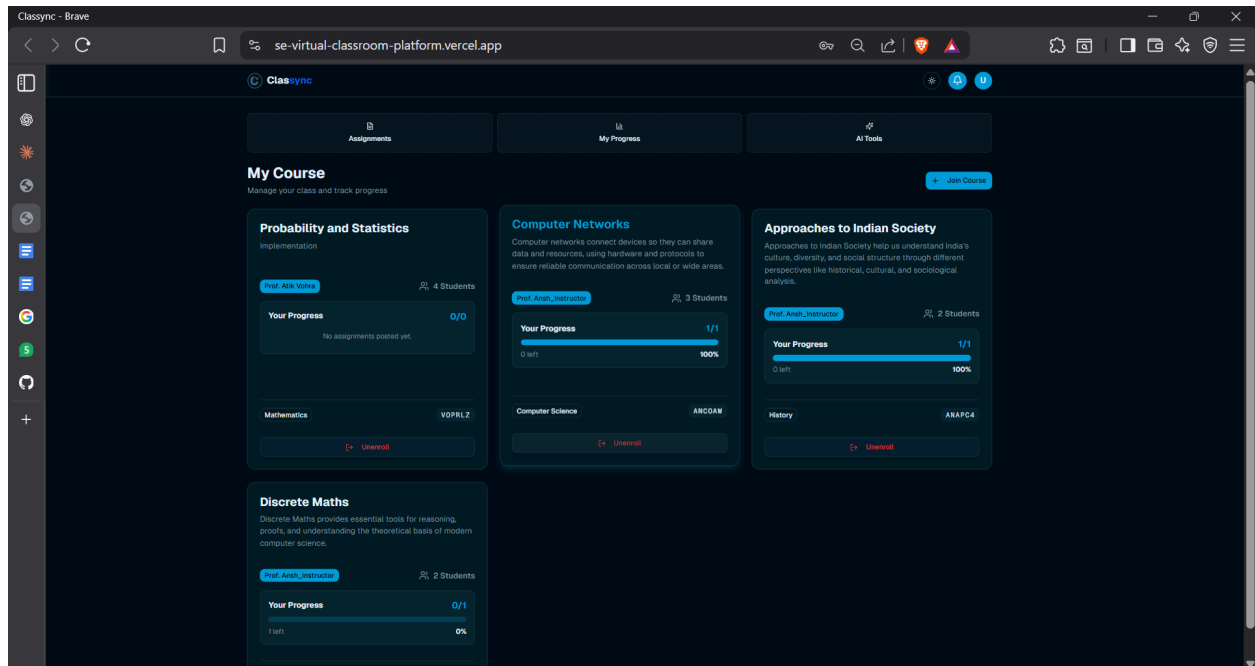
Cross browser working test

In this test case we will test if the deployed website works on various browsers or not.

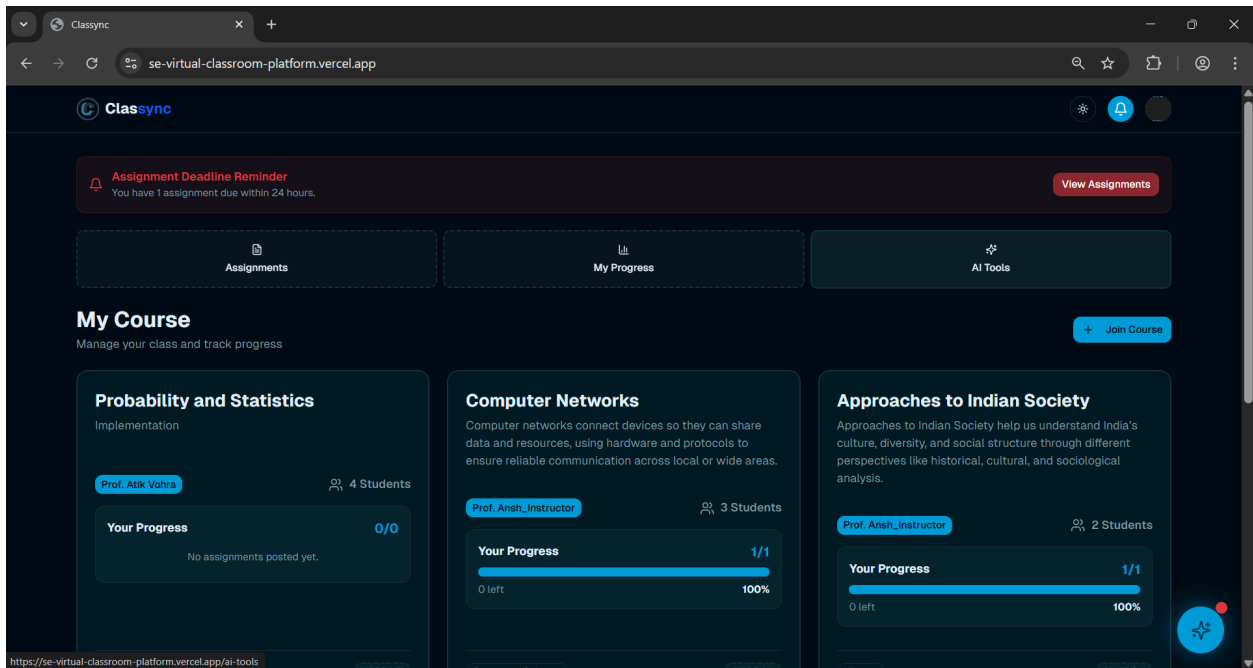
Browsers taken in account-

1. Chrome
2. Brave
3. Edge

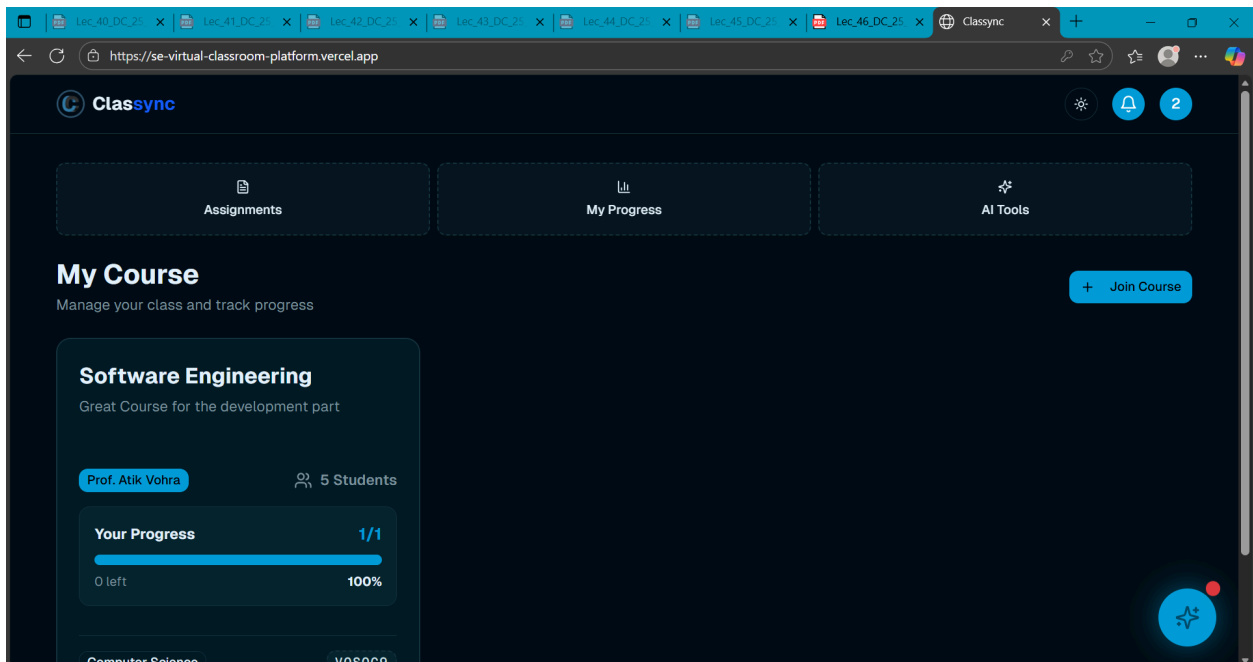
Brave-



Chrome-



Edge-



Summary-

Browser	Pass/Fail
Brave	Pass
Chrome	Pass
Edge	Pass