Test Plan for Team U

Group - V

Team Member Names
Mugdha Atul Kulakarni
Anirudha Kapileshwari
Vijay Krishna Raghav Kesanapalli
Naga Sai Shyam Suhas Komaravolu
Thirumalesh Kurukundha

Spring 2024 Software Engineering I (CS-487-02)
24 February 2024

Section 1

Proposed strategy for testing

Testing techniques	Test cases	Test case scenario	Test case Description	Test case objective
Boundary Value Analysis	User Registration	Register a new user with minimal and maximal input values for each field	Verify the system accepts the minimum and maximum valid lengths for username, password, and email.	Ensure the registration form handles boundary values correctly for all input fields.
Integration Testing	Data Integration with University Systems	Sync user data from the university system.	Verify that the system correctly integrates and updates user data from the university's database.	Ensure seamless data integration and consistency between the system and university databases.
Functional Testing	Accessing Personalized Calendar	Access the personalized calendar after logging in.	Check if the logged-in user can view their personalized calendar with the correct events and dates.	Verify that the personalized calendar displays accurately for users.
	Successful Direct Messaging	Send a direct message to another user.	Check if a user can successfully send and receive direct messages.	Ensure the direct messaging feature works correctly for all users.

Usability Testing	Goal Adjustment Mechanism	Adjust a previously set goal.	Evaluate the ease with which a user can modify an existing goal through the interface.	Ensure that the goal adjustment mechanism is user-friendly and functional.
Security Testing	Discussion Board Posting	Post a message on the discussion board.	Assess if the message posting feature prevents SQL injection and cross-site scripting	Ensure the discussion board is secure against common web vulnerabilities.
Compatibility Testing	Successful Access to Course Information	Access course information on various devices and browsers.	Verify that course information is accessible and displays correctly across different platforms.	Ensure cross-platform and cross-browser compatibility for accessing course information.
End-to-End Testing	Feedback Submission	Submit feedback through the system.	Validate the complete process of submitting feedback, from entry to acknowledgment.	Confirm that feedback submission workflow operates as intended.
Regression Testing	Successful Personal Goal Adjustment	Modify a personal goal after an application update.	Ensure that users can still adjust their personal goals following a system update.	Confirm that recent updates do not adversely affect the goal adjustment

Load Testing	Successful Submission of Feedback	Submit feedback under high server load.	Test the feedback submission feature during peak usage times to ensure it handles high traffic.	Guarantee that the system can handle multiple feedback submissions simultaneously without

Section 2

Test Case

Test Cases:

1. Test Case: User Registration

- Initial State: User is not registered.
- Inputs/Actions: User navigates to the registration page, enters required information, and submits.
- Expected End State: User is registered and logged into the application.

2. Test Case: Accessing Personalized Calendar

- Initial State: User logged in and has classes and deadlines.
- Inputs/Actions: User navigates to the calendar feature.
- Expected End State: User views a personalized calendar with their classes and deadlines.

3. Test Case: Goal Adjustment Mechanism

- Initial State: User with set academic goals.
- Inputs/Actions: User modifies their goals through the goal adjustment feature.

- Expected End State: User's academic goals are updated in their profile.

4. Test Case: Data Integration with University Systems

- Initial State: New assignment posted on the university's system.
- Inputs/Actions: Data integration checks for updates.
- Expected End State: New assignment appears in the user's EduConnect account.

5. Test Case: Feedback Submission

- Initial State: User has feedback to provide.
- Inputs/Actions: User navigates to the feedback section and submits their feedback.
- Expected End State: Feedback is recorded and acknowledgment is received.

6. Test Case: Successful Direct Messaging

- Objective: Verify successful direct messaging between users.
- Starting State: Users are logged into the app and on the messaging interface.
- Execution Steps:
  1. User A sends a direct message to User B.
  2. User B receives the message and can view it in their inbox.
- Expected Final State: User A's message is successfully delivered to User B.

7. Test Case: Discussion Board Posting

- Objective: Verify successful posting on a discussion board.
- Starting State: Users are logged into the app and on the discussion board interface.
- Execution Steps:
  1. User creates a new discussion topic and posts it on the board.
  2. Other users can see the new discussion topic and post replies.

- Expected Final State: The new discussion topic is displayed on the board, and users can interact with it by posting replies.

8. Test Case: Successful Personal Goal Adjustment

- Objective: Verify successful adjustment of personal goals.
- Starting State: User is logged into the app and on the goal adjustment interface.
- Execution Steps:

  1. User edits an existing personal goal, changing the description or target date.
  2. User saves the changes.

- Expected Final State: The user's personal goal is updated with the new information.

9. Test Case: Successful Access to Course Information

- Objective: Verify successful access to course information from university databases.
- Starting State: User is logged into the app and on the course information interface.
- Execution Steps:

  1. User selects a course to view detailed information.
  2. The app retrieves and displays information such as course description, schedule, and instructor.

- Expected Final State: The user can view accurate and up-to-date information about the selected course.

10. Test Case: Successful Submission of Feedback

- Objective: Verify successful submission of feedback through the app.
- Starting State: User is logged into the app and on the feedback submission interface.
- Execution Steps:

  1. User enters feedback comments in the provided text field.

2. User submits the feedback.

- Expected Final State: The feedback is successfully submitted and recorded by the app.

11. Test Case: Successful Event Addition

- Objective: Verify successful addition of events to the personalized calendar.
- Starting State: User is logged into the app and on the calendar interface.
- Execution Steps:

1. User adds a new event, providing details such as title, date, and time.

2. User saves the event.

- Expected Final State: The new event is added to the user's calendar and displayed on the specified date and time.

Section 3

Recommendations

To improve the analysis report, the team could focus on refining user categories by identifying specific needs within each group, clarifying functionalities for different users, detailing technical and non-functional requirements, including performance metrics, and outlining a feedback collection mechanism. Incorporating market and competitor analysis, adding application mock-ups for a clearer visual understanding, discussing long-term impacts, and addressing ethical considerations around data privacy and accessibility would also enhance the report's comprehensiveness and depth.

Enhancing the report requires focusing on detailed user needs, pinpointing functionalities per user group, and elaborating on technical

specifics and non-functional expectations. A stronger emphasis on user feedback mechanisms, alongside a thorough market and competitor insight, could significantly enrich the report. Visual prototypes would aid in visualizing the user interface, while discussions on ethical concerns and the app's long-term educational impact would provide a holistic view of EduConnect's potential.