Group: NTG

Hendrik Adriaan Honiball 601476

Manissha Packary 577639

Brandon Alfonso Lemmer 578062

Adolph Jacobus Van Coller 601005

  Francois Myburg 600576

Campuslearn platform Sen 381 assignment 1

Table of Contents

[Functional requirements for CampusLearn platform 2](#_Toc207817375)

[Non-Functional Requirements for CampusLearn platform 3](#_Toc207817376)

[Selected Agile Methodology: Scrum 4](#_Toc207817377)

[Benefits of Using Scrum for Campus Learn 5](#_Toc207817378)

## Functional Requirements for CampusLearn Platform

**1. Student Registration & Profile Management** The system should be able to allow students to register new accounts, update their personal information as well as track their interactions.

**2. Peer Tutor & Topic Management** The system should allow tutors to create topics that they can help in, send alerts to students and upload learning materials.

**3. Data Storage**  The system should be able to organize and store learning materials, ensuring that the materials are accessible across different platforms.

**4. Private Messaging**  The system must provide a secure, one-on-one communication channel for students and tutors to interact privately.

**5. API Notification**  The system must be integrated with an external API to provide real time notifications to students and tutors via email, SMS, or WhatsApp.

**6. Public Forum**  The system needs a public forum that allows for anonymous posting, questions and answers as well as moderation by administrators.

## Non-Functional Requirements for CampusLearn Platform

**1. Security**  The system needs to implement comprehensive strategies and technologies to safeguard user information against various threats such as unauthorised access, cyberattacks, and data breaches. The system can use anti-virus software, firewalls, data encryption, Multi-Factor Authentication (MFA), Intrusion Detection and Prevention Systems (IDS/IPS) and VPNs (Virtual Private Networks). Security incidents should be detected and responded to within ten minutes. Vulnerabilities should be patched within 48 hours. Performing regular software updates helps eliminate vulnerabilities before attackers can exploit them. Performing regular data backups helps reduce the risk of data loss. Enforcing strong passwords, conducting risk assessments, and implementing access control measures also help enhance overall security. Conducting security audits to log and track user actions for compliance is another good security measure.

**2. Reliability**  The system needs to have 99.9% uptime over a month. The system needs to handle unexpected errors and system failures with minimal downtime. Access and submissions should not be interrupted.

**3. Scalability** The system needs to handle an increasing number of users and data without a drop in performance. As the number of students and courses increases, the system needs to maintain its speed and reliability.

**4. Performance**  The system needs to respond to user requests within a suitable time frame, ensuring a smooth and efficient experience.

**5. Usability**  The system needs to be easy to navigate for users. This high usability enhances user satisfaction, encourages frequent platform usage, and reduces the need to learn how to navigate and use the system. User satisfaction should be above 80% and task completion rate should be above 90%.

**6. Maintainability** The system’s software needs to be designed for easy updates and modifications. Bug fixes should be deployed within 48 hours.

**7. Accessibility**  The system should comply with accessibility standards to ensure all users, including those with disabilities, can use the platform effectively. The system can make use of Braille keyboard for blind people, text-to-speech software, voice assistance, adaptive keyboard, and many more.

**8. Data Integrity**  The system needs to make sure the data is accurate, reliable, and consistent throughout its lifecycle.

**9. Interoperability**  Increases overall functionality and allows for a more cohesive environment. The system must have the ability to integrate with other systems. API integration with three external systems, and data exchange formats include JSON and XML.

**10. Compatibility**  The system must be able to run on other devices and work with other applications.

**11. Availability**

The system must be available 24/7. It should have only one hour downtime per month.

**12. Portability**

The system must have the ability to run on different platforms or environments.

## Selected Agile Methodology: Scrum

Scrum is the most appropriate Agile methodology for CampusLearn because of its flexible yet structured approach to iterative development.

Why Scrum Matches the Nature of the Project:

**1. Size of Team** Scrum is ideal because it is meant for small to medium teams (5-9 members), which aligns with the group size on this assignment.

**2. The Scope of Project** Campus Learn has a wide scope with changing requirements. Scrum's sprint-based delivery system allows for distribution of changes in multiple increments.

**3. User Feedback** Scrum supports the need for regular reviews and feedback, which happen during sprint reviews.

**4. Iterative Development** Features are delivered in short cycles (sprints), which allows for regular improvement.

## Benefits of Using Scrum for CampusLearn

**1. Reduced Time-to-Market**  An iterative sprint cycle in Scrum allows you to receive features such as user authentication and course browsing early, and the flexibility allows the Campus Learn platform become usable quickly, receive feedback, and show progress with an actual deliverable to stakeholders.

**2. Improved Collaboration**  Scrum has continuous communication through stand-ups and sprint planning. The emphasis put on the daily stand-up fosters team as a unit (not individuals) to produce a product; this encourages interaction and collaboration and helps align developers, designers, and academic stakeholders to the educational purpose across various disciplines.

**3. Higher Product Quality**  Sprint reviews occur frequently, and continuous integration encourages consistent testing and bug fixing. This produces a Campus Learn system that will be more stable and reliable, leading to better user satisfaction because of the user if there are defects and better usability.

**4. Flexibility for Change** Scrum methodology encourages an iterative approach, which allows the Campus Learn team to react to changing requirements (course formats, regulations, etc.) without derailing the project. This is especially useful for the academic environment which can be unpredictable.

**5. User-driven Development**  By having users present in sprint reviews and in feedback loops, Scrums help to grow Campus Learn based on real students and tutor needs. This will assist in higher adoption and engagement for new graduates, while providing significant features that matter (like the AI Chatbot and private messaging) and using feedback to improve how those features are delivered to students.