Hajee Mohammad Danesh Science and Technology University,Dinajpur-5200

Department of Computer Science and Engineering
Level : 3, Semester : I

A Report on A Software (Personal Academic Auditor)

Course Code: CSE 305
Course Title: Software Engineering

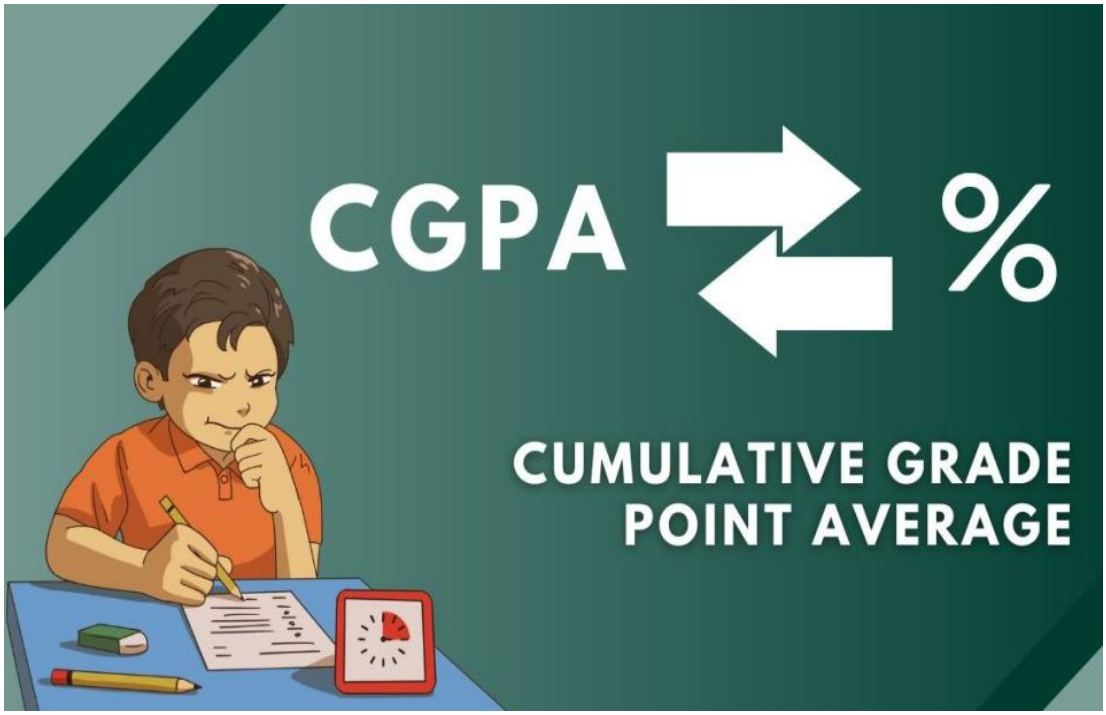
Prepared By
Fariha Tasneem Feya
ID: 2102006

Submitted To
Pankaj Bhowmik
Lecturer
Department of Computer Science and Engineering

Submission Date: 24 November 2024

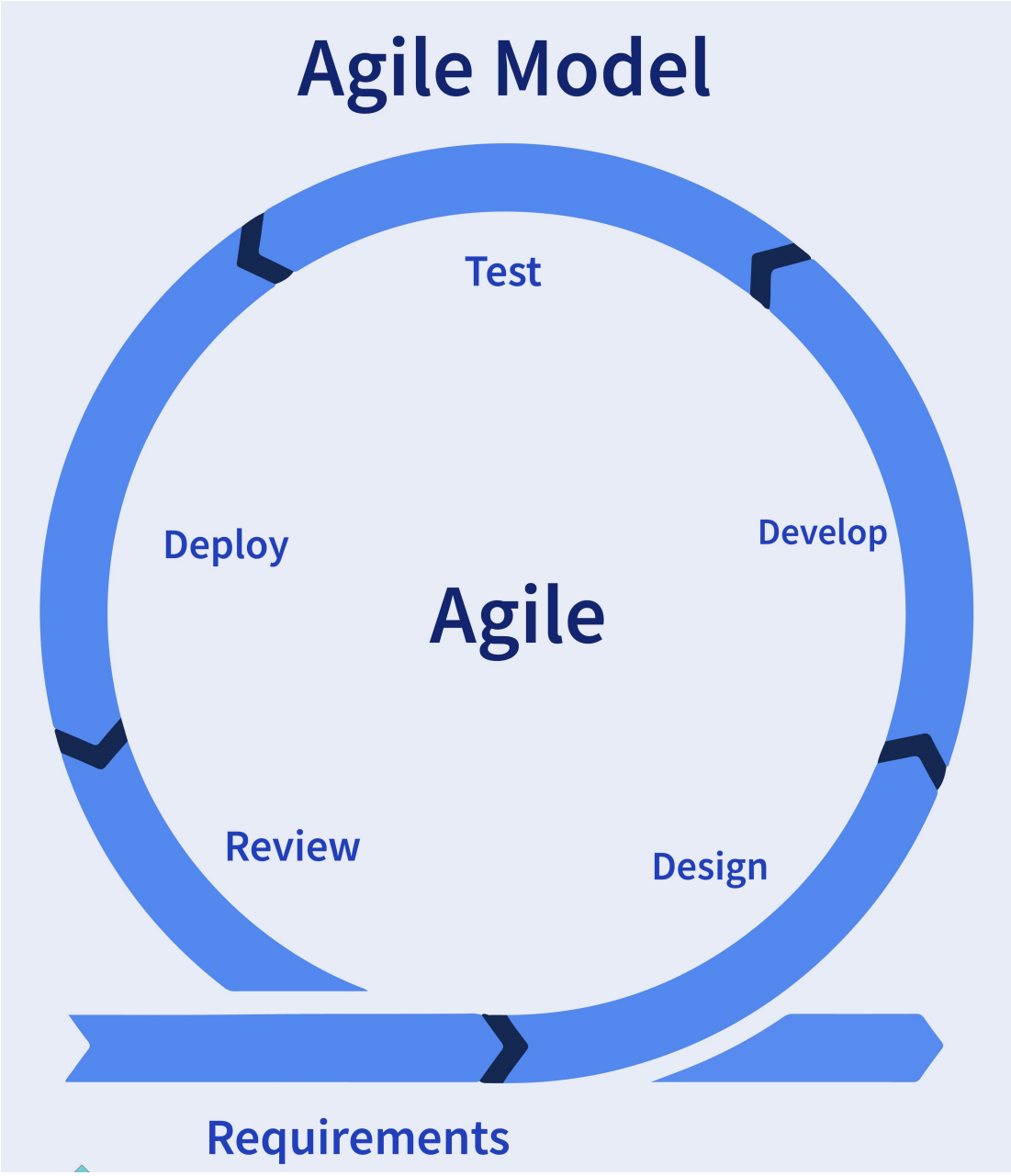
Introduction

The Personal Academic Auditor (PAA) is an innovative software solution designed to empower students in tracking, analyzing, and improving their academic performance. With the increasing emphasis on data-driven insights in education, the PAA system aims to provide students with a comprehensive platform to monitor their academic progress, predict future performance, and receive actionable recommendations to achieve their academic goals. The application facilitates the input of course credits, grades, and CGPA (Cumulative Grade Point Average) while offering advanced features such as performance analysis, peer comparison, and predictive modeling. By leveraging modern software development techniques, the PAA seeks to enhance student engagement, foster goal-oriented learning, and promote academic excellence. This project will be developed using a systematic Software Development Life Cycle (SDLC) process, ensuring a structured and well-documented approach to design, development, testing, deployment, and maintenance.



Agile Model

Agile is highly effective for projects requiring continuous iteration, feedback, and flexibility. Since academic performance tracking may involve evolving user requirements, Agile ensures rapid prototyping and frequent releases. It's continuous engagement with users ensures the product meets expectations. Agile model allows iterative refinements based on feedback and quick adjustments to changes in the requirements.



1. Concept & Requirement Gathering (Project Backlog Creation)

The purpose is to gather and document clear and concise requirements for the PAA system. This ensures that the system meets the expectations of users (students, advisors) and aligns with its objectives: tracking, analyzing, and improving academic performance.

1.1 Functional Requirements

Functional requirements specify the behavior or functionality of the system. These are directly tied to what the system should do.

- **Grade and Course Input**

Students should be able to input grades, course credits, and other academic details.
Allow bulk uploading of academic data via spreadsheets.

- **CGPA Calculation**

The system should calculate the CGPA based on inputted grades and course credits.
Provide real-time updates when grades are modified.

- **Performance Tracking**

Display historical performance trends using visualizations like charts or graphs.
Highlight academic strengths and areas needing improvement.

- **CGPA Prediction**

Predict future CGPA based on current and past performance.
Allow students to input hypothetical grades for "what-if" scenarios.

- **Peer Comparison**

Provide insights into a student's performance relative to their peers.
Anonymize peer data for privacy.

- **Report Generation**

Generate detailed academic performance reports in PDF format.

1.2 Non-Functional Requirements

Non-functional requirements define the quality attributes and constraints of the system.

- **Performance**

The system should handle up to 1,000 concurrent users without performance degradation. Response time for grade calculations and visualizations should not exceed 2 seconds.

- **Usability**

The interface should be intuitive and easy to use for non-technical users.
Provide accessibility features for users with disabilities.

- Scalability

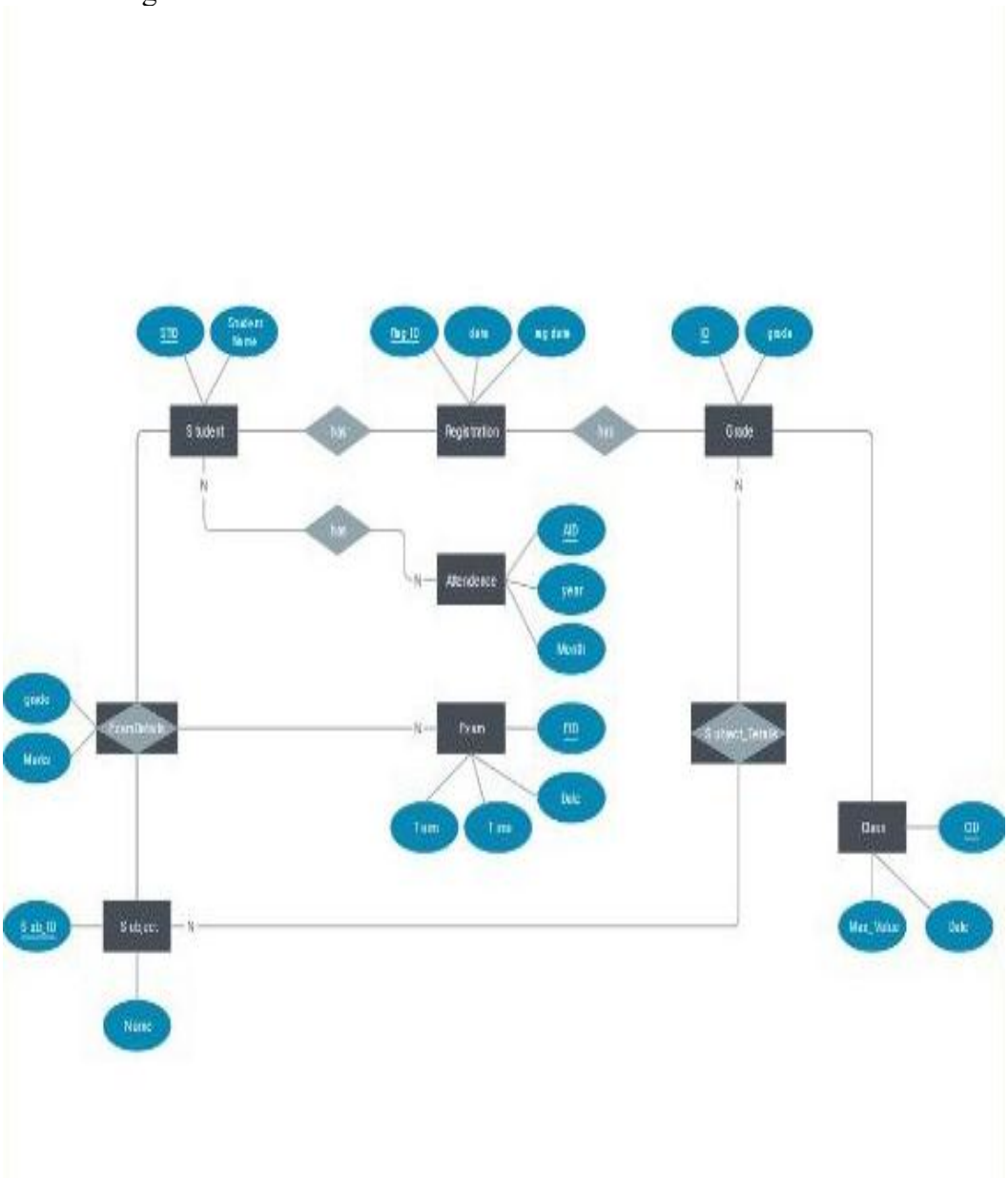
The system should be scalable to support additional features like AI-based recommendations or integration with institutional databases.

- **Maintainability**

The system should follow modular design principles for easier updates and bug fixes.

2. Designing

The ER diagram for the software Personal Academic Auditor



3. Sprint Planning

Divide the product backlog into smaller, manageable pieces called Sprints (time-boxed iterations, typically 2–4 weeks). Plan the tasks for the first sprint, focusing on delivering the highest-priority functionality (e.g., inputting grades and calculating CGPA).

4. Sprint Execution (Development and Testing)

The team works on the sprint tasks to develop and implement the planned features.
Daily stand-up meetings to discuss progress, challenges, and next steps.
Continuous testing (unit testing, integration testing) to ensure quality.

Working software increments with functional features (e.g., basic grade input and CGPA computation).

5. Sprint Review (Demonstration and Feedback)

Present the completed features to stakeholders (students, advisors).
Gather feedback on usability, functionality, and potential improvements.
Stakeholder feedback,updates to the product backlog based on feedback are delivered.

6. Sprint Retrospective (Process Improvement)

Review the sprint process with the team to identify what went well and what can be improved. Discuss challenges faced and brainstorm solutions to improve efficiency in future sprints.

7. Repeat Sprints (Iterative Development)

The team picks the next set of high-priority tasks from the product backlog for the next sprint. Each sprint adds incremental improvements to the PAA, such as peer analysis, performance predictions, or personalized recommendations.

8. Release (Delivery of Functional Software)

After completing a series of sprints and incorporating all critical features, the PAA is ready for a production release. Conduct user acceptance testing (UAT) to ensure the system meets the user's expectations.

9. Continuous Improvement (Post-Release Iterations)

Monitor the performance and usability of the deployed system. Collect additional user feedback and address new requirements or issues through subsequent sprints.

10. Conclusion

The Personal Academic Auditor system offers a comprehensive solution for students to track and improve their academic performance, offering real-time analysis of their CGPA, course credits, and academic standing. By following the SDLC process, the project will ensure that the application meets both functional and non-functional requirements, providing a reliable and scalable solution for students and academic institutions alike.