

K.J.Somaiya College Of Engineering
(Autonomous College Affiliated to University of Mumbai)
Progress Report Project B

Project Name	Edumate - Assisting Teachers K.J. Somaiya College of Engineering
Project Members	1. Rushabh Gandhi (1911012) 2. Kritarth Jain (1911020) 3. Nayan Mandliya (1911027) 4. Hussein Motiwala (1911031)
Reporting period	16th January 2023 - 28th March 2023
Section One: Summary	
<p>This was one of the most crucial phases of our project, during which we accomplished the majority of the project's work. During this time period, we have developed key system modules and refined our existing system. During the previous phase of this project, the basic infrastructure of the system with features such as creating classes, assignments, submitting assignments, peer grading, and announcements was implemented. In this phase, we have begun the activity scheduling section, in which teachers can schedule activities. Following this, we developed the intelligent attendance component, which was somewhat challenging as it is one of our primary modules. This module came before the quiz creation and attempt section, in which we gave teachers the ability to create tests and allowed students to take them in a secure environment.</p> <p>After completing the above two major modules, we moved on to the assignment plagiarism check. For this section, we began with an algorithm that was effective at detecting copy cases but took a bit more time, so in order to find a solution, we found another algorithm that reduced the amount of time spent comparing student assignments. This section initiates the process of developing a progressive web app that enables teachers and students to access the system efficiently on mobile devices. This section was not initially proposed, but we have also</p>	

completed it as an additional activity. The final component on which we have worked is the project tracking, where teachers can create projects and groups, and where groups can report their progress so that teachers can easily track the activities performed by students and the project's progress directly on the site. With the modifications provided by our mentor and evaluator, nearly all of the sections presented at the beginning of the project have been completed. The following steps in the project would be research papers, minor fixes, and user interface (UI) enhancements. As our system is now complete, we have tested it in every way to evaluate the project.

Section Two: Activities and Progress

- The project has made significant progress towards achieving its objectives. Some of the key accomplishments include the successful execution of the Smart Attendance system, which allows teachers to take attendance automatically and students to mark attendance using their handheld devices.
- Additionally, the project has implemented a quiz module with cheating detection, which helps teachers monitor student activities during quizzes.
- Another important milestone is the successful implementation of the assignment submission and grading module, which includes plagiarism checking. This helps teachers check for similarities between uploaded assignments by students.
- The grading module has also been implemented, which enables teachers to grade assignments and provide feedback to students.
- Peer-grading and project tracking have also been implemented, providing students with an opportunity to learn about grading systems by grading the assignments of their peers.
- The announcement module and schedule generation have also been successfully implemented, allowing teachers to upload reference materials and make announcements that are accessible to students enrolled in a particular course.
- In addition, we have generated a Progressive Web App and has been working on improving the User Interface of the system to enhance user experience.
- The team has also been carrying out integration and testing of the various modules to

ensure smooth functioning of the entire system.

Overall, the project has made impressive progress towards providing a comprehensive solution to address the challenges faced by teachers and students in educational institutions.

Section Three: Institutional & Project Partner Issues

Not applicable as no external Organisation or project partners are associated with the project.

Section Four: Outputs and Deliverables

The following are the deliverables:

- Software Requirement specification
- Literature Review
- Design Document
- Prototype
- Report
- Presentation
- Testing Document.
- Project Code. (<https://github.com/hussein-hub/Edumate>)

Section Five: Outcomes and Lessons Learned

- Importance of User Feedback: Gathering feedback from users, such as teachers and students, is essential for creating a system that meets their needs and is effective in improving teaching and learning outcomes.
- Need for Collaboration: Creating a teaching assistant system requires collaboration among different stakeholders, including developers, designers, educators, and administrators. Ensuring effective communication and collaboration among team members is crucial to the success of the project.
- Emphasis on Security and Privacy: Ensuring the security and privacy of sensitive data, such as student attendance records and academic performance, is essential for maintaining trust and confidence in the system.

- Importance of Scalability and Sustainability: Creating a teaching assistant system that is scalable and sustainable requires planning and foresight, including the ability to adapt to changing needs and technologies.
- Need for Continuous Improvement: Technology and education are constantly evolving, and creating a teaching assistant system requires a commitment to continuous improvement and innovation, including the ability to adapt to changing needs and trends.
- The selection of appropriate machine learning algorithms: The choice of machine learning algorithm and the model architecture plays a crucial role in the performance of the face detection module.
- The importance of developing a user-friendly web application - developing a user-friendly and intuitive web application for deploying the code summarization/translation model is essential for making it accessible to developers.

Section Six: Usage of Tools

1. Visual Studio Code: It's a source-code editor developed by Microsoft that supports several programming languages and features like debugging, version control, and syntax highlighting. We have used VS Code as our code editor, where we have developed our project from scratch. VS Code gives a good user interface and user experience to develop any project. The support for various languages and libraries in Visual Studio code helped us develop our project faster and more efficiently. Also, the inbuilt GitHub functionality in VS Code helped us manage our source code effectively.
2. Postman: It's a collaboration platform for API development, enabling users to design, test, and document APIs through features such as automated testing, API monitoring, and API mocking.
3. MySQL Workbench: It's a visual database design tool that enables users to create, manage, and visualize MySQL databases, including queries, connections, and server administration. For viewing and accessing our data outside the system, we have used Workbench. As our database is not hosted anywhere, to access the same data on every machine workbench provides a data file that can be shared with others so that they can

also access the same data. This feature of the workbench is used by us to have the same copy of data to work on.

4. Draftable API: It's a document comparison API that allows users to compare files in various formats like PDF, Word, Excel, and PowerPoint, providing an automated and efficient way of comparing documents. We have used draftable Api to enhance the similarity score predictions with visual side by side differences between the pair of documents in a user friendly and informative interface.
5. GitHub: It's a web-based version control repository that enables developers to collaborate and manage code changes through features such as pull requests, code reviews, and project management tools. We have been using GitHub for version control of our project for keeping a track of the code and collaborate with the team members.
6. Adobe PDF Embed API: It's a tool that allows users to embed PDF files on their websites, providing a seamless and interactive PDF viewing experience for users. We have used the adobe API to display the assignments that are submitted by the students in the project.

Software Languages used: Python, JavaScript, HTML, CSS, jQuery.

Libraries used: Django, TensorFlow, Keras, nltk, OpenCV, django-pwa, sklearn, Bootstrap, Datatables.

Section Seven: Evaluation

Edumate is a teaching assisting system that has undergone various testing stages to ensure its functionality and usability. The testing process began with unit testing, where individual models of the system were tested to ensure they worked correctly. This was followed by integration testing, where the system's components were combined and tested to ensure they worked seamlessly together. Finally, the system underwent acceptance testing by reporting to the project mentor to ensure it met the requirements and expectations.

Throughout the testing process, the project was refined and improved to ensure it meets the needs of teachers and students. The system includes features such as smart attendance, quiz and

assignment management, plagiarism checking, peer grading, project tracking, announcement management, and schedule generation. Additionally, the system was transformed into a progressive web app, improving the user interface and making it more accessible to users. The machine learning modules were evaluated with their performances based on the accuracy of predictions of faces and document similarities along with the speed of predictions. Overall, the testing and refinement process of Edumate have led to a reliable and functional system that will assist teachers in their daily tasks and provide a comprehensive learning experience for students.

Section Eight: Dissemination

Not Applicable.

Section Nine: Risks, Issues and Challenges

- For the face detection section in the smart attendance, high number faces in a single image although detected correctly cannot be displayed with visually conceivable sharpness. Thus, to overcome this issue attendance can be taken using images of each row in the classroom. Also, the model which we have used to detect faces is quite heavy and takes considerable time to train.
- Pairwise plagiarism checking for an increased number of documents is time consuming due to comparison checking of each pair using the draftable Api. Our system is also partially dependent on Draftable API, which if stops working for some reason might have a negative impact on our system.
- At the moment assignments are only accepted in pdf format and no other. As we plan to keep the submission pipeline smooth and consistent.
- Quizzes have cheating detection that auto-submits if the person changes the tab three times, assuming a glitch the first two times. The student cannot retake the quiz after a second technical glitch. We notify the user the second time.
- Before the algorithm is run, it is important to verify that all assignments have been submitted in order to ensure the peer-grading module's proper operation. This is because if there are very few assignments, the algorithm might not function properly. Therefore,

prior to starting the peer-grading process, a check is made to ensure that all assignments have been turned in.

Section Ten: Collaboration and Support

To test the smart attendance component of our system, we require permission from our college to capture images of classrooms where the system will be implemented. The images will be used to validate the output generated by our smart attendance unit.

Section Eleven: Next Steps

- Partial implementation of team formation: Team formation is a crucial aspect of any collaborative project or activity. Partial implementation of team formation involves creating a system for grouping students into teams by choice of the instructor of the course. The system also enables communication between the student groups and the instructor with fixed checkpoints serving as status updates.
- Research Paper: Conducting a research paper related to the teaching assistant system can help in identifying and analysing the various challenges and opportunities associated with it. The paper can cover topics such as the impact of technology on teaching and learning and best practices for implementing educational technology systems.
- Minor Fixes: As with any software development project, minor fixes and adjustments may be necessary to ensure that the system is functional, user-friendly, and effective. The team may need to conduct various tests and debugging sessions to identify and fix any issues or bugs in the system.
- Testing: Testing the system is a crucial step in ensuring that it meets the requirements and objectives of the project. The team may need to conduct various tests, such as unit testing, integration testing, and acceptance testing, to ensure that the system is reliable, scalable, and secure.
- Final Report and Presentation: Creating a final report and presentation can help in summarizing the findings, outcomes, and lessons learned from the project. The report can

cover topics such as the project objectives, methodology, results, and recommendations for future improvements. The presentation can be used to showcase the system's features and benefits to stakeholders, including educators, administrators, and students.

Comments by Examiner (s):

Name and Signature of Examiner(s):