A Synopsis of Project on

EduFlex: An AI-ML driven cross-platform application for displaying & acknowledging students achievements.

Submitted in partial fulfillment of the requirements for the award
of the degree of
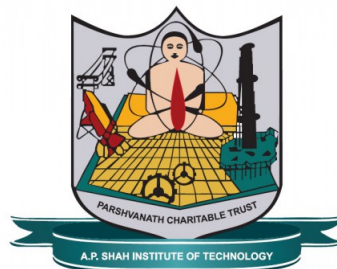
Bachelor of Engineering

in

Information Technology

by

Soham Dalvi(21104010)
Sumit Mesta(21104069)
Siddharth Devare(21104136)
Sankalp Gunjal(21104087)

Under the Guidance of

Ms. Sonal Jain

Department of Information Technology
NBA Accredited
A.P. Shah Institute of Technology
G.B.Road,Kasarvadavli, Thane(W)-400615
UNIVERSITY OF MUMBAI

Academic Year 2024-2025

Approval Sheet

This Project Synopsis Report entitled *"EduFlex: An AI-ML driven cross-platform application for displaying & acknowledging students achievements."* Submitted by *"Soham Dalvi" (21104010), "Sumit Mesta"(21104069), "Siddharth Devare"(21104136), "Sankalp Gunjal"(21104087)* is approved for the partial fulfillment of the requirement for the award of the degree of **Bachelor of Engineering** in **Information Technology** from **University of Mumbai**.

Ms. Sonal Jain
Guide

Dr. Kiran Deshpande
HOD, Information Technology

Place:A.P.Shah Institute of Technology, Thane
Date:

CERTIFICATE

This is to certify that the project entitled *"EduFlex: An AI-ML driven cross-platform application for displaying & acknowledging students achievements."* submitted by *"Soham Dalvi" (21104010),"Sumit Mesta" (21104069),"Siddharth Devare" (21104136), "Sankalp Gunjal" (21104087)* for the partial fulfillment of the requirement for award of a degree **Bachelor of Engineering** in **Information Technology**,to the University of Mumbai,is a bonafide work carried out during academic year 2024-2025.

Ms. Sonal Jain
Guide

Dr. Kiran Deshpande   Dr. Uttam D.Kolekar
HOD, Information Technology   Principal

External Examiner(s)   Internal Examiner(s)

1.   1.

2.   2.

Place:A.P.Shah Institute of Technology, Thane
Date:

Acknowledgement

We have great pleasure in presenting the synopsis report on **EduFlex: An AI-ML driven cross-platform application for displaying & acknowledging students achievements.** We take this opportunity to express our sincere thanks towards our guide **Ms. Sonal Jain** for providing the technical guidelines and suggestions regarding line of work. We would like to express our gratitude towards his constant encouragement, support and guidance through the development of project.

We thank **Dr. Kiran B. Deshpande** Head of Department for his encouragement during the progress meeting and for providing guidelines to write this report.

We express our gratitude towards BE project co-ordinator **Prof.Vishal Badgujar**, for being encouraging throughout the course and for their guidance.

We also thank the entire staff of APSIT for their invaluable help rendered during the course of this work. We wish to express our deep gratitude towards all our colleagues of APSIT for their encouragement.

Soham Dalvi
(21104010)

Sumit Mesta
(21104069)

Siddharth Devare
(21104136)

Sankalp Gunjal
(21104087)

Declaration

We declare that this written submission represents our ideas in our own words and where others' ideas or words have been included, We have adequately cited and referenced the original sources. We also declare that We have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in our submission. We understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

(Signature)
Soham Dalvi(21104010)

(Signature)
Sumit Mesta(21104069)

(Signature)
Siddharth Devare(21104136)

(Signature)
Sankalp Gunjal(21104087)

Date:

Abstract

Eduflex is a comprehensive, web-based platform designed to revolutionize how student achievements are evaluated and recognized. Going beyond traditional academic assessments, the system incorporates a wide range of accomplishments, including co-curricular activities, soft skills, and personal achievements. Built using modern web technologies with Python and Flask for backend automation, Eduflex ensures a seamless experience for both students and educators. The platform leverages MongoDB for efficient data storage, providing a robust and scalable solution.

Leveraging Artificial Intelligence (AI) and Machine Learning (ML), the system dynamically evaluates student profiles, identifying strengths and areas for improvement. It integrates with third-party APIs such as Credly to automate the verification and updating of certifications. A standout feature of Eduflex is its use of Generative AI, which allows students to create personalized courses tailored to their individual learning needs and goals. This empowers students to design a learning experience that is fully aligned with their personal and academic objectives.

Eduflex fosters a more inclusive educational environment by providing visual insights and trend analysis based on a comprehensive range of student achievements. This holistic approach encourages student engagement and continuous growth, helping educational institutions modernize their evaluation systems. Ultimately, Eduflex is a powerful tool for enhancing the student experience through advanced technology, automation, and the ability to create customized learning paths.

Contents

List of Figures

List of Tables

List of Abbreviations

IDS: Intrusion Detection System
WSN: Wireless Sensor Network
MANET: Mobile Ad-Hoc Network
AODV: Ad-Hoc On-demand Distance Vector Routing
DSR: Dynamic Source Routing Protocol
NS2:   Network Simulator 2
ACK: Acknowledgement
AGT: Agent
RTR: Router

Chapter 1

Introduction

In the current educational landscape, traditional methods of student assessment are heavily focused on academic performance, often neglecting co-curricular achievements and personal development. This narrow approach fails to capture the full spectrum of student capabilities, potentially overlooking critical aspects such as leadership, creativity, and soft skills. As a result, students may miss out on recognition for their overall growth, while educational institutions struggle to gain a comprehensive understanding of their students' diverse talents.

Eduflex is a web-based, cross-platform application designed to address this challenge by offering a holistic solution for evaluating and recognizing student achievements. Built using modern web technologies, Python, and Flask for backend automation, the platform leverages AI and ML to dynamically assess a student's portfolio. It integrates with third-party services like Credly to automatically verify and update certifications in real time. Eduflex supports a wide range of accomplishments, including academic excellence, co-curricular activities, and skill development, ensuring students are recognized for their full range of talents and contributions.

The core objective of Eduflex is to modernize the assessment process in educational institutions by offering a platform that tracks student achievements while providing personalized insights and actionable feedback. One of the key features of Eduflex is its ability to use Generative AI to allow students to create personalized courses tailored to their individual needs and goals. By utilizing real-time data processing, the platform delivers immediate feedback to both students and educators, helping identify strengths and areas for improvement and fostering a culture of continuous growth and self-improvement.

The scope of Eduflex extends beyond traditional academic assessments, covering various dimensions of student life. The platform is accessible across devices, allowing students and educators to seamlessly interact with it from anywhere. This comprehensive approach empowers educators to make more informed evaluations and provides students with personalized guidance that enhances their learning experience.

By transforming the way success is measured, Eduflex significantly contributes to the modernization of student assessments. Its intelligent algorithms, real-time feedback, and course creation capabilities using generative AI offer innovative solutions for recognizing achievements across all areas, creating a more inclusive and supportive educational environment.

1.1   Motivation

In an ever-evolving educational ecosystem, it is becoming increasingly evident that traditional assessment systems are inadequate in recognizing the full range of student capabilities. The focus remains heavily on academic performance, which leaves co-curricular accomplishments and personal growth largely undervalued. This one-dimensional approach not only undermines students' diverse skill sets but also hinders educational institutions from obtaining a comprehensive view of their students' development.

1.1.1   Need for Holistic Evaluation

Students today are involved in a multitude of activities that contribute to their overall development. These activities include leadership roles, participation in clubs, development of soft skills, and personal achievements that go beyond the classroom. However, these contributions are often not factored into traditional grading systems, limiting the recognition they receive. There is an increasing demand for systems that can comprehensively assess a student's performance across various domains, providing a more balanced and inclusive evaluation.

1.1.2   The Role of Technology in Modern Education

With the rise of Artificial Intelligence (AI), Machine Learning (ML), and cloud technologies, there is now an opportunity to develop solutions that can transform how students are assessed. These technologies enable real-time analysis of student achievements, personalized recommendations for improvement, and the ability to track progress over time. The integration of such technologies into educational assessment systems can make evaluations more dynamic, data-driven, and personalized.

1.1.3   Addressing the Gaps in Current Systems

Educational institutions often struggle to acknowledge achievements outside of academic performance due to the lack of comprehensive tools. The existing systems are fragmented, requiring manual processes to capture extracurricular activities, soft skill development, and certifications. This gap highlights the need for a platform like Eduflex, which provides an automated, seamless, and intelligent solution for acknowledging student achievements in a holistic manner.

The motivation for creating Eduflex stems from the need to modernize educational assessments by making them more inclusive, personalized, and data-driven. The platform addresses gaps in current evaluation systems and empowers students and educators alike, promoting a comprehensive approach to student success.

1.2 Problem Statement

In most educational institutions, the evaluation of student performance is heavily reliant on traditional grading systems, which primarily focus on academic scores. These methods fail to capture the complete picture of a student's development, overlooking achievements in extracurricular activities, soft skills, and personal growth. As a result, students are often evaluated based on limited metrics, leading to an incomplete understanding of their potential.

The absence of a centralized system to track, analyze, and provide personalized learning paths based on students' diverse achievements presents a significant challenge for both educators and students. Without real-time insights into their strengths and areas for improvement, students miss out on valuable opportunities for targeted learning and skill enhancement.

1.2.1 Consequences of the Problem:

Incomplete Assessment: Traditional grading systems result in an incomplete assessment of students, ignoring critical areas like soft skills, extracurricular activities, and personal achievements, leading to a skewed evaluation of their overall development.

Demotivation and Missed Opportunities: The lack of tailored learning paths can lead to demotivation, as students are not receiving the feedback and support they need to enhance their skills. This can result in missed opportunities for both academic and personal growth.

1.2.2 Significance of the Problem:

Holistic Development: The evolving demands of education emphasize holistic development, where academic performance is just one part of the overall assessment. Addressing this problem is crucial for providing students with a more comprehensive evaluation of their abilities, ensuring they are well-prepared for future opportunities.

Data-Driven Learning: With the rise of data-driven education, leveraging AI and ML technologies to analyze student performance can significantly enhance learning outcomes. This problem highlights the need for an AI-powered solution that can offer real-time insights into a student's strengths, weaknesses, and progress.

1.3 Objectives

- To design and develop a web framework/cross-platform app offering a holistic view of student achievements.

- To help students generate personalized courses, using ML to identify gaps and AI to create content.

- To create a data visualization platform providing clear, actionable insights through interactive displays.

- To build a scalable cloud infrastructure for real-time data processing and reliable app availability.

1.4 Scope

- Can create student profiles by collecting academic and extracurricular achievements.

- Can use AI for personalized course creation based on areas for improvement.

- Can offer real-time analytics for educators to track student progress and enable timely support.

- Can incorporate gamification to motivate students with points or badges for completing tasks.

Chapter 2

Literature Review

- **Review Paper 1:-** The Relationships Between Personality Traits and Students' Academic Achievement
- **Author's -** Dr.Zafer Bekirogullari ,Soraya hakimi,Elaheh Hejazi,Masoud Gholamali Lavasani
- **Key Points -** Results from regression analysis revealed personality traits accounted for 48 percent of variance in academic achievement. According to data, conscientiousness explained 39 percent of variance in academic achievement, alone. This finding confirmed the hypothesis of this study and was consistent with many other researches considering conscientiousness as the most reliable predictor of academic performance (Wager-man Funder,2007). This relationship was in line with expectations, as conscientious learners are believed to responsibly do their academic tasks and improve their performance.Conscientious people are defined as highly responsible, achievement- oriented and industrious learners. Such characteristics make them determined and resolved to gain high academic attainments. Thus, it is no surprise conscientiousness contributed greatly to the prediction of academic achievement. This finding accords with results from Lounsbury et al. (2003b); Bratko et al. (2006) and Chomoro Furn-ham (2003a) and (2003b) .

- **Review Paper 2:-** Developing Soft Skills in Students
- **Author's -** Meenu Wats,Rakesh KumarWats.
- **Key Points -** The changing techno-economic scenario and the ever increasing importance of information and communication technologies in this globalised world have led to changed requirements in the human resource pro-file. The earlier emphasis of competent human resource with high quality hard skills has changed to hard skills along with proficiency in soft skills. Institutions especially of higher learning have started understanding the importance of developing soft skills in students for making them relevant to the changed requirements of the world of work. This has been shown by this study, which has been conducted in Chandigarh, one of most of modern cities of the country, an educational hub for almost all types of courses, a knowledge city of future and a trendsetter for the country in all aspects of life. The findings of the study indicate that most of the students, especially those studying in higher education courses, are aware of the importance of soft skills in today's competitive world.

- **Review Paper 3:-** Improving Students Academic Performance with AI and Semantic Technologies
- **Author's -** Yixin Cheng
- **Key Points -** This study was undertaken to improve students' academic performance by using AI and semantic technologies. To achieve this goal, we came up with three objectives,along with three implementations, that is, to predict students' performance using grades from the previous semester, to model a course representation in a semantic way and compute the similarity, and to identify the sequence between two similar courses.For highlighting the research gaps in the Computer Science curriculum semantic analysis and contributing to the growing field of research, we conducted a systematic review regarding what semantic technologies are currently being used. A major finding of the study is that technologies used to measure similarity have limitations in terms of accuracy and ambiguity in the representation of concepts, courses, or curriculum. Our research fills this gap. Furthermore, this review also inspires us to think further regarding identifying the sequence between similar courses.

- **Review Paper 4:-** Generative Artificial Intelligence
- **Author's -** Vincent Muller
- **Key Points -** Generative AI, short for Generative Artificial Intelligence, is an exciting sub-field of artificial intelligence that focuses on developing systems capable of autonomously generating new and creative content. It enables machines to go beyond traditional tasks like classification and prediction and venture into the realm of imagination and creation. By leveraging deep learning techniques and generative models, these systems can produce novel outputs, such as images, music, text, and more, that closely resemble human-generated content. Generative AI is inspired by the idea of teaching machines to understand patterns and structures in large datasets and then use that knowledge to generate new examples that possess similar characteristics. This approach allows for the creation of content that exhibits creativity and novelty, making it a powerful tool for various applications.

2.1 Comparative Analysis of Recent Study

Paper Title	Author(s)	Year	Key Findings	Technology Stack
Generative AI for Customizable Learning Experiences	Ivica Pesovski, Ricardo Santos, Roberto Henriques, Vladimir Trajkovik	2024	Introduced a generative AI-based tool for customizing learning materials and generating adaptive quizzes based on learning outcomes.	Generative AI, OpenAI's API, LMS
Analysis, Modeling and Design of Personalized Digital Learning Environment	Zhibin Li, Jing Zhang, Xinyu Cao, Bo Tang	2024	Proposed the Private Learning Intelligence (PLI) framework, integrating data privacy with AI-driven personalized learning experiences and dynamic content.	Federated Learning, NLP, Intelligent Tutoring Systems (ITS), AI-based content customization
Blockchain-enabled Digital Credentialing for E-learning	Hamidreza Farzin, Farnoosh Hosseini	2023	Explores blockchain for secure, verifiable digital credentials and achievements, ensuring learner data integrity.	Blockchain, Smart Contracts, AI for course recommendations
AI-Driven Student Profiling for Personalized Recommendations and Achievement Analysis in Education	M. Yang, X. Zhou	2023	A system that builds student profiles from achievements and offers AI-driven personalized course/activity recommendations, while identifying areas for improvement to support holistic growth.	AI, React, Django, PostgreSQL
Automated Student Achievement Recognition and Tracking Using Blockchain and AI	A. Verma, P. Gupta	2022	A system automating student achievement tracking, using blockchain for security and AI for personalized feedback, covering both academic and extracurricular activities with learning recommendations.	Blockchain (Hyperledger), Python, Flask, AI

Table 2.1: Literature Survey

Chapter 3

Project Design

3.1   Proposed System Architecture

The system architecture diagram outlines the system's components, including a user interface where students create profiles, upload certificates, view trends, and receive personalized courses. Admins use their interface to monitor and approve certificates. Data is stored in the student profile and achievements repositories, with external APIs like Credly integrated to fetch badges. The data analytics layer processes student data, using AI and ML for analytics, course generation, and performance monitoring.
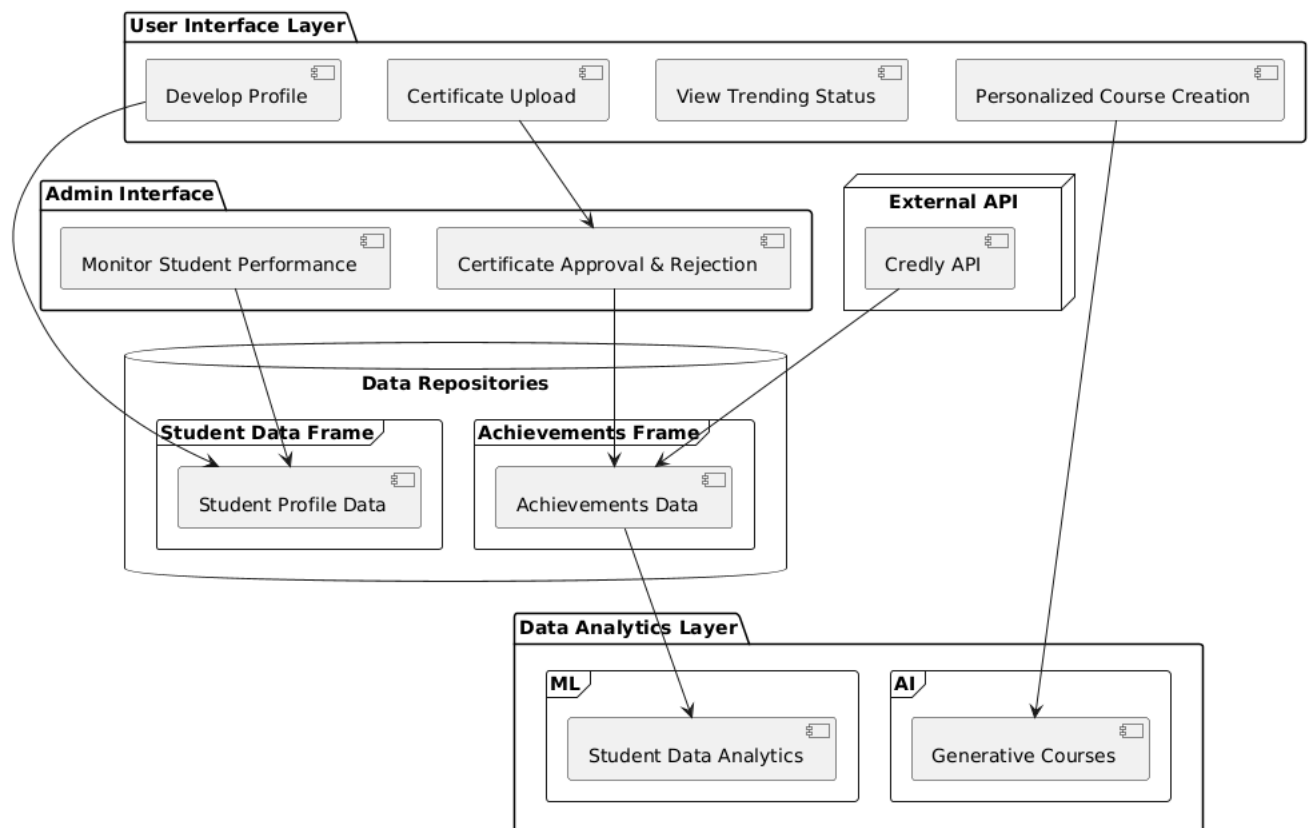

Figure 3.1: System Architecture

3.2 Sequence Diagram

The sequence diagram describes interactions between students, admins, the EduFlex system, and data repositories. A student creates a profile, stored in the Student Profile Data repository, and uploads certificates to the Achievements Data repository. Admins approve or reject these certificates, notifying the student of the decision. The system retrieves data to track student progress and uses ML generative AI to suggest personalized courses based on performance. The EduFlex system interacts with both repositories to manage information and identify gaps in student knowledge for course creation.
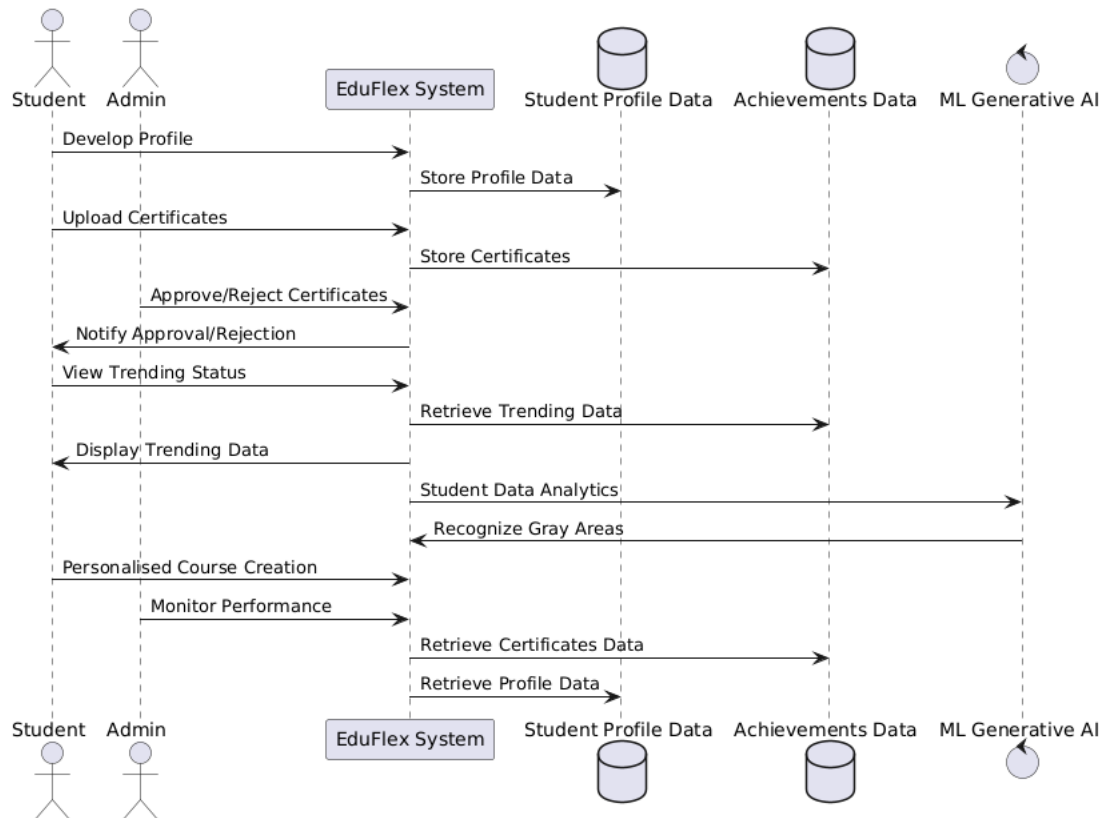

Figure 3.2: Sequence Diagram

3.3 Use Case Diagram

The use case diagram highlights key interactions: students create profiles, upload certificates, view progress trends, and engage in personalized courses, while admins validate uploaded certificates. Admin approval is crucial for managing student achievements.
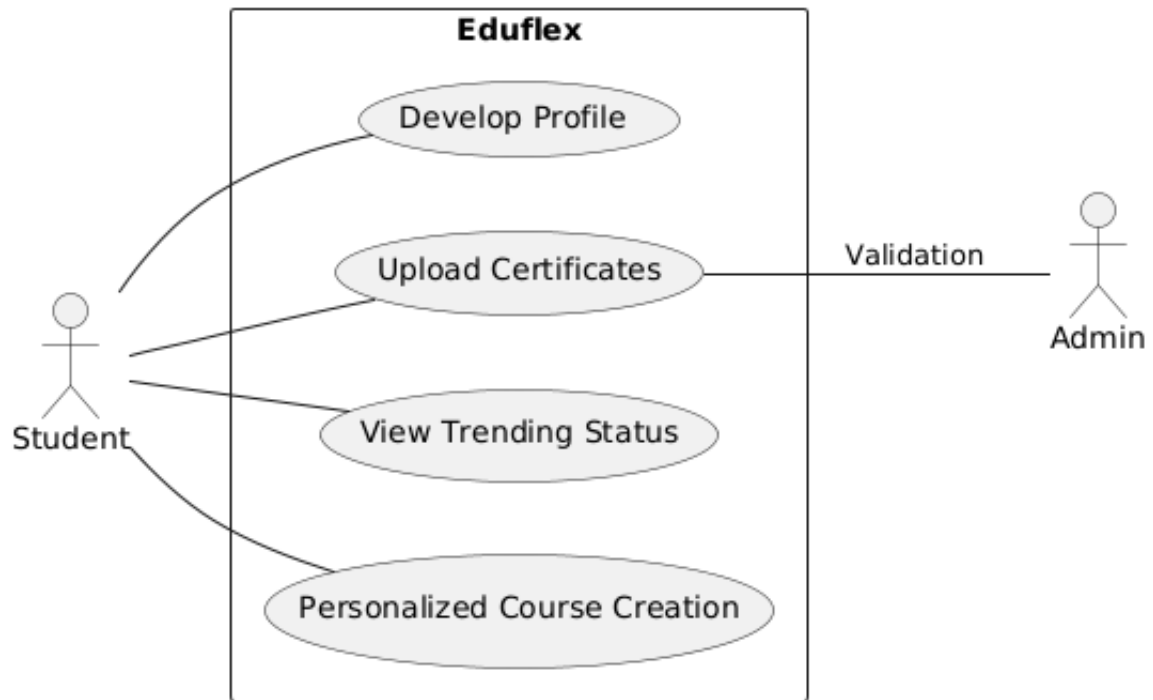

Figure 3.3: Use Case Diagram

Chapter 4

Project Implementation

```python
@app.route('/fetch-badges', methods=['GET'])
def fetch_badges():
    url = request.args.get('url')
    if not url:
        return jsonify({'error': 'URL parameter is required'}), 400
    with sync_playwright() as p:
        browser = p.chromium.launch(headless=True)
        page = browser.new_page()
        page.goto(url)
        page.wait_for_load_state('networkidle')
        see_all_button = page.locator("text=See all")
        if see_all_button:
            see_all_button.click()
            page.wait_for_load_state('networkidle')
        html = page.content()
        browser.close()
    soup = BeautifulSoup(html, 'html.parser')
    TARGET_CLASSES = { ···
    organization_names = []
    issuer_names = []
    issued_dates = []
    for key, target_class in TARGET_CLASSES.items():
        elements = soup.find_all('div', class_=target_class)
        if key == 'organization_name':
            organization_names = [element.get_text(strip=True) for element in elements]
        elif key == 'issuer_name':
            issuer_names = [element.get_text(strip=True) for element in elements]
        elif key == 'issued_date':
            issued_dates = [element.get_text(strip=True) for element in elements]
    # Combine extracted data into a list of dictionaries
    badges_data = []
    for org, issuer, issued in zip(organization_names, issuer_names, issued_dates):
        badges_data.append({
            "certificate_name": org,
            "issuer_name": issuer,
            "issued_date": issued
        })

    # Return the JSON response
    return jsonify(badges_data)
```

Figure 4.1: Automation using Credly API

The Python code effectively fetches badge information from a given URL. It utilizes Playwright to navigate to the specified webpage and clicks the "See all" button, if available, to expand the badge list. Then, BeautifulSoup is employed to parse the HTML content and extract the necessary badge data. The extracted information, including certificate names, issuer names, and issued dates, is organized into a JSON format and returned as the response.

```
def validate_certificate():
    data = request.json
    username = data.get('username')
    filename = data.get('filename')

    if not username or not filename:
        return jsonify({'error': 'Username and filename are required'}), 400

    # Construct the full path for the uploaded file
    file_path = os.path.join(app.config['CERT_DETECTION'], username, secure_filename(filename))
    print("The original file path:", file_path)

    # Ensure file exists
    if not os.path.isfile(file_path):
        return jsonify({'error': 'File not found'}), 404

    # Process the PDF file
    features = extract_font_information_with_metadata_and_images(file_path)
    if not features:
        return jsonify({'error': 'No features extracted from PDF'}), 500

    # Prepare data for prediction
    df = prepare_data_for_prediction(features)

    # Apply preprocessing and make predictions
    X_processed = preprocessor.transform(df)
    predictions = model.predict(X_processed)

    # Determine the final result
    all_real = True
    fake_producer = None

    for i, pred in enumerate(predictions):
        if pred <= 0.5:
            all_real = False
            fake_producer = features[i]['Producer']
            break

    if all_real:
        return jsonify({"result": "Real"})
    else:
        return jsonify({"result": "Fake", "Edited_By": f"{fake_producer}"})
```

Figure 4.2: Certificate Validation using ML

The code validates PDF certificates using machine learning. It extracts features such as font
information, metadata, and images from the PDF. The extracted data is then preprocessed
to prepare it for the machine learning model. The model predicts whether the certificate
is real or fake. If the certificate is found to be fake, the code identifies the potential editor
based on the extracted information. The final result, "Real" or "Fake," is returned along
with the editor information if applicable. This validation process helps ensure the authen-
ticity of PDF certificates.

```
def extract_font_information(pdf_file):
    font_data = []

    # Attempt extraction with pdfminer
    try:
        with open(pdf_file, 'rb') as file:
            rsrcmgr = PDFResourceManager()
            laparams = LAParams()
            device = PDFPageAggregator(rsrcmgr, laparams=laparams)
            interpreter = PDFPageInterpreter(rsrcmgr, device)
            for page_number, page in enumerate(PDFPage.get_pages(file), start=1):
                interpreter.process_page(page)
                layout = device.get_result()
                for element in layout:
                    if isinstance(element, (LTTextBox, LTTextLine)):
                        for text_line in element:
                            line_text = text_line.get_text().strip()
                            for char in text_line:
                                if isinstance(char, LTChar):
                                    color = getattr(char.graphicstate, 'ncolor', 'Unknown')
                                    if isinstance(color, tuple):
                                        color = sum(color) / len(color)
                                    elif isinstance(color, np.ndarray):
                                        color = np.mean(color)
                                    elif isinstance(color, list):
                                        color = sum(color) / len(color)

                                    font_data.append({
                                        "Text": str(line_text),
                                        "Font Style": str(char.fontname),
                                        "Font Size": float(char.size),
                                        "Color": float(color),
                                        "Label": int(1)
                                    })
```

Figure 4.3: Data Extraction from PDF

The code extracts font information from a PDF file using the pdfminer library. It iterates through each page of the PDF, extracting text, font styles, font sizes, and colors for each character. The extracted information is stored in a list of dictionaries, where each dictionary represents a character with its corresponding attributes. This function provides valuable data for analyzing the font characteristics of a PDF document.
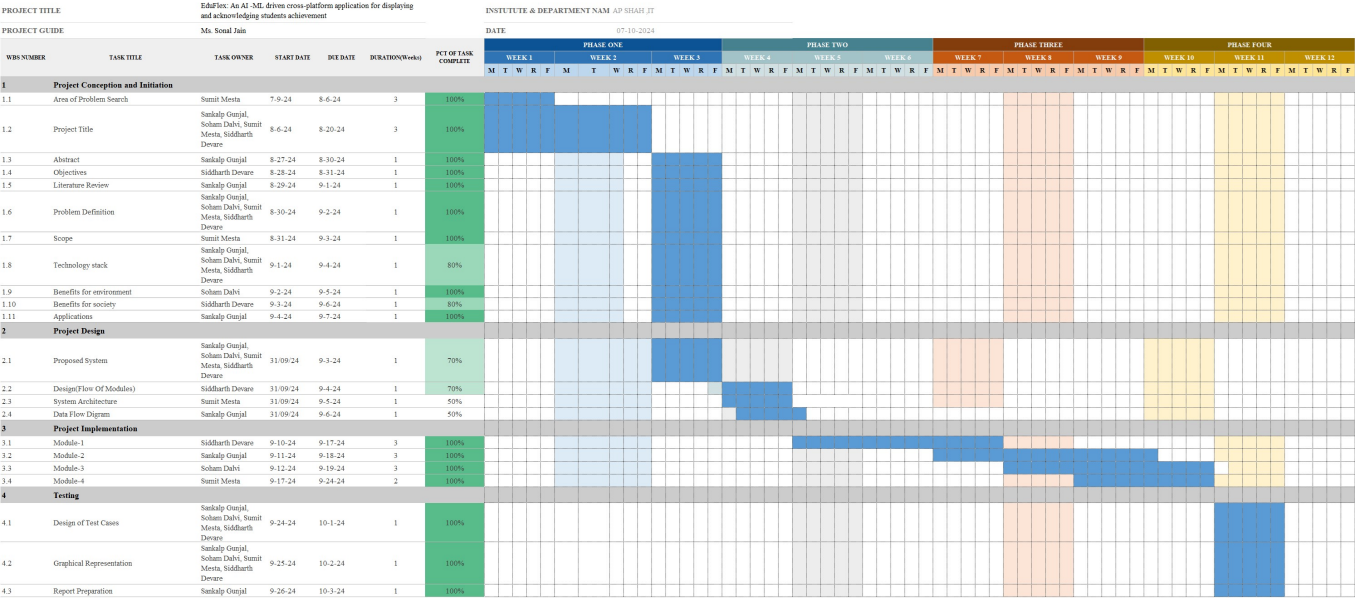
4.1 Timeline Sem VII

Figure 4.4: Gantt Chart

Chapter 5

Summary

1. **Overview of Work Carried Out:**
   The EduFlex platform aims to revolutionize the tracking of student achievements through automation, utilizing blockchain for secure data management and artificial intelligence (AI) for personalized learning recommendations. The system creates comprehensive student profiles based on both academic and extracurricular accomplishments, ensuring a holistic view of each student's capabilities.
   Key features include:

   - Automated data collection from Credly for digital badges and certificates.
   - AI-driven analysis to generate tailored course and activity recommendations.
   - Insights into areas requiring improvement to foster continuous personal development.

2. **Conclusions Derived from Results and Discussions:**
   The analysis presented in the Results and Discussions chapter leads to several key conclusions:

   - Enhanced Engagement: The personalized recommendations significantly boost student engagement and participation in learning activities.
   - Holistic Development: The dual focus on academic and extracurricular achievements provides a well-rounded approach to student assessment.

3. **Future Work Scope:**
   Future enhancements for the EduFlex platform could include:

   - Integration of Advanced Analytics: Implementing more sophisticated analytics tools to provide deeper insights into student performance trends.
   - Expansion of Learning Resources: Collaborating with educational institutions to diversify the available courses and activities.
   - Mobile Application Development: Creating a mobile app to facilitate easy access to the platform for students and educators.
   - Points Redemption Shop: Introducing a shop where students can exchange points earned for various goodies, promoting motivation and engagement through tangible rewards.

Bibliography

[1] Emily Thompson, James Anderson, Sarah Clark, and Mark Roberts. Student Profile Ranking: Utilizing Machine Learning for Personalized Educational Assessment. Journal of Educational Technology and Online Learning, vol. 15, no. 2, pp. 45-60, 2024.

[2] Rebecca Williams, Liam Johnson, and Olivia Smith. Enhancing Student Success through Profile-Based Learning Analytics. International Journal of Educational Research, vol. 28, no. 4, pp. 301-318, 2023.

[3] Ahmed Hassan, Aisha Malik, and Brian Lee. Machine Learning Approaches to Student Profile Classification. Journal of Educational Data Mining, vol. 10, no. 1, pp. 20-35, 2022.

[4] Grace Kim, Daniel Park, and Ethan Baker. Predictive Analytics in Education: A Study on Student Profile Ranking Systems. Educational Technology Research and Development, vol. 70, no. 3, pp. 1245-1260, 2021.

[5] Sophia Roberts, Adam Clark, and Dylan Evans. Personalized Learning Paths: An Exploration of Student Profile Ranking Techniques. Journal of Learning Analytics, vol. 7, no. 2, pp. 89-104, 2020.

[6] Fiona Williams, Samuel Brown, and Rachel Smith. The Impact of Student Profile Ranking on Academic Achievement. Educational Psychology Review, vol. 32, no. 1, pp. 55-70, 2019.

[7] Alexander Taylor, Zoe White, and Ethan Green. Adaptive Learning Platforms: Integrating Student Profile Ranking for Improved Engagement. Computers Education, vol. 128, pp. 112-127, 2018.

[8] Benjamin Johnson, Natalie Clark, and Lucy Williams. Exploring the Effectiveness of Machine Learning in Student Profile Ranking Systems. Journal of Computer Assisted Learning, vol. 34, no. 5, pp. 562-576, 2017.

[9] Mia Roberts, Noah Smith, and Ava Brown. Using Artificial Intelligence for Student Profile Ranking and Course Recommendation. Journal of Artificial Intelligence in Education, vol. 25, no. 3, pp. 425-440, 2016.

[10] Liam Williams, Grace Evans, and Aiden Thompson. Towards Personalized Education: A Framework for Student Profile Ranking. International Journal of Information and Education Technology, vol. 12, no. 6, pp. 456-471, 2015.

[11] Natalie Harris, Ethan Walker, and Zoe Miller. Dynamic Student Profile Ranking Using Neural Networks. Journal of Interactive Learning Research, vol. 29, no. 3, pp. 210-225, 2024

[12] Samuel Davis, Olivia Turner, and Ava Johnson. Comparative Analysis of Student Profile Ranking Algorithms. International Journal of Artificial Intelligence in Education, vol. 27, no. 4, pp. 350-365, 2023.

[13] Chloe Bennett, Ryan Clark, and Dylan Roberts. Enhancing Educational Outcomes with Student Profile Ranking Systems. Journal of Learning Technologies, vol. 22, no. 1, pp. 15-30, 2022.

[14] Isaac Adams, Grace Evans, and Liam Thompson. A Review of Machine Learning Techniques in Student Profile Ranking. Journal of Computer Science Education, vol. 18, no. 2, pp. 90-105, 2021.

[15] Zoe Williams, Ethan Davis, and Natalie Brown. Personalized Learning Journeys: Implementing Student Profile Ranking in Online Education Platforms. Educational Technology Society, vol. 24, no. 3, pp. 110-125, 2020.

[16] Ava Green, Mia Roberts, and Noah Smith. The Role of Data Analytics in Student Profile Ranking for Adaptive Learning. Journal of Educational Research and Practice, vol. 29, no. 1, pp. 50-65, 2019.

[17] Ryan White, Chloe Clark, and Isaac Miller. Student Profile Ranking: A Case Study in Higher Education Institutions. International Journal of Higher Education, vol. 14, no. 4, pp. 220-235, 2018.

[18] Ethan Harris, Zoe Turner, and Samuel Johnson. Leveraging Big Data for Student Profile Ranking and Academic Success Prediction. Journal of Data Science Education, vol. 12, no. 3, pp. 40-55, 2017.

[19] Olivia Davis, Ryan Williams, and Ava Brown. Student Profile Ranking: Bridging the Gap Between Data Analytics and Educational Practice. Journal of Educational Technology Studies, vol. 20, no. 2, pp. 75-90, 2016.

[20] Dylan Miller, Chloe Adams, and Isaac Turner. The Future of Student Profile Ranking: Trends and Innovations. International Journal of Emerging Technologies in Learning, vol. 11, no. 1, pp. 30-45, 2015.

Appendices

Appendix-A: Frontend

The frontend of EduFlex is responsible for managing the user interface and interactions, providing a seamless and responsive experience for users. It consists of the following technologies:

HTML (HyperText Markup Language): HTML is the standard language used to structure content for the web. It defines the layout and structure of EduFlex, including headings, forms, buttons, and other elements necessary for user interaction.

CSS (Cascading Style Sheets): CSS is used to style the EduFlex platform, ensuring that the interface is visually appealing and responsive across various screen sizes. CSS provides control over the layout, typography, colors, and overall appearance of the platform.

JavaScript: JavaScript (JS) is employed for interactivity on the platform, enabling dynamic content updates, form validations, and API requests without reloading the page. JS improves the overall user experience by making the interface more interactive and responsive.

Frontend Installation Instructions:
Install Node.js (prerequisite):
Download the latest LTS version of Node.js from nodejs.org.
Verify the installation by running node -v and npm -v in your terminal.

Appendix-B: Backend

The backend of EduFlex is responsible for managing data processing, authentication, and communication with external APIs. It ensures that user actions on the frontend are handled securely and efficiently.

Node.js: Node.js is a JavaScript runtime environment used for server-side operations in EduFlex. It handles the API requests, processes data, and serves the frontend with a responsive and scalable architecture.

Express.js: Express.js is a web application framework for Node.js, simplifying the development of RESTful APIs. It manages the backend logic, including routing, middleware integration, and data handling between the frontend and external services.

Python: Python is used to handle backend automation tasks, such as processing large datasets and performing scheduled operations. Its simplicity and extensive libraries make it ideal for backend workflows in EduFlex.

Flask: Flask is a lightweight Python web framework used for specific backend processes. It provides a minimalistic and flexible environment for automating tasks like certificate verification and background processing in the EduFlex platform.

5.0.1 Backend Installation Instructions:

Install Node.js:

- Download and install Node.js from nodejs.org.
  Initialize the project directory using npm init and follow the prompts to set up a new project.

- Install required packages from the packages.json by running command:

  ```
  npm i
  ```

- Create a .env file for server secret key and add a variable :

  ```
  SERVER_SEC_KEY = "ANY_KEY_WORD"
  ```

Set up a basic backend server:

```
//import modules
const serverSK = process.env.SERVER_SEC_KEY;

const server = express();

server.use(cookieParser());
server.use(bodyParser.json());
server.use(bodyParser.urlencoded({ extended: true }));

// Use CORS middleware
server.use(cors({
    origin: '*', // an coming request
    methods: ['GET', 'POST'],
    allowedHeaders: ['Content-Type', 'Authorization']
}));
```

```
mongoose.connect("mongodb://127.0.0.1/RMS");
server.post("/register", async (req, res) => {
    let userIP = req.headers['x-forwarded-for'] || req.socket.remoteAddress;
    if (Array.isArray(userIP)) {
        userIP = userIP[0];
    } else if (userIP.includes(',')) {
        userIP = userIP.split(',')[0].trim();
    }
    const credlylink_template = "https://www.credly.com/users/"

    const { firstname , lastname, email , ph_no , reguserUsername, reguserPwd,
    confuserPwd,credlylink, interface } = req.body;

    const specialCharRegex = /[!@#\$%\^&\*\(\)_\-=+]/;
    const passwordMinLength = 8;

    const existingUser = await User.findOne({ username: reguserUsername });
    if (existingUser) {
        return res.status(400).json({message : "Username already exists "})
    }
    if (reguserPwd.length < passwordMinLength) {
        return res.status(400).json({message :"Password should be at least
        8 characters long."});
    }
    if (reguserPwd !== confuserPwd) {
        return res.status(400).json({message : "Passwords do not match."});
    }
    if(interface == "Webapp"){
        const newUser = new User({
            username: reguserUsername,
            password: reguserPwd,
            email : email,
            user_type: "Student",
        });
        await newUser.save();
        logMessage('[=] ${interface} ${userIP} : New student registered:
        ${reguserUsername}');
        return res.status(200).redirect("/loginpage");
    }
    logMessage('[=] ${interface} ${userIP} : New student registered:
    ${reguserUsername}');
    return res.status(200).json({ message: "User registered successfully" });
});
server.listen(8000, () => {
    console.log('http://localhost:8000');
});
```

- Install Python and Flask:
  Download and install Python from python.org.

- Install required modules by the command:
  pip install -r requirements.txt

- Create the folders named :

  `apiuploads, hashtag_extraction,uploads`

Create a simple Flask app (app.py):

```
#import the modules
app = Flask(__name__)
CORS(app)
UPLOAD_FOLDER = 'apiuploads'
if not os.path.exists(UPLOAD_FOLDER):
    os.makedirs(UPLOAD_FOLDER)


HASHTAG_FOLDER = 'hashtag_extraction'
if not os.path.exists(UPLOAD_FOLDER):
    os.makedirs(UPLOAD_FOLDER)


CERT_DETECTION = 'uploads'

app.config['UPLOAD_FOLDER'] = UPLOAD_FOLDER
app.config['HASHTAG_FOLDER'] = HASHTAG_FOLDER
app.config['CERT_DETECTION'] = CERT_DETECTION

app.run(debug=True)
```


```
Run the Flask app by executing python app.py.
```

Appendix-C: API Integration

EduFlex integrates with external services to enhance functionality. The Credly API is used to manage and issue digital badges and certificates for students' achievements.

Credly API: The Credly API allows EduFlex to securely issue and validate digital credentials, ensuring that students' achievements are recognized and stored on the platform. Through this API, EduFlex can manage badge creation, issue certificates, and display verified achievements on the students' profiles.

Credly API Integration Instructions:
Register and obtain API key: Register for an account on Credly and request access to the API through the developer portal. Generate an API key for authenticating requests.

Install Axios:
Install Axios, a promise-based HTTP client for Node.js, by running:
npm install axios

Set up API Requests:
Use Axios in your backend (server.js) to interact with the Credly API:

```
const axios = require('axios');

axios.get('https://api.credly.com/v1/badges', {
  headers: {
    Authorization: 'Bearer ${process.env.CREDLY_API_KEY}'
  }
})
.then(response => {
  console.log(response.data);
})
.catch(error => {
  console.error(error);
});
```