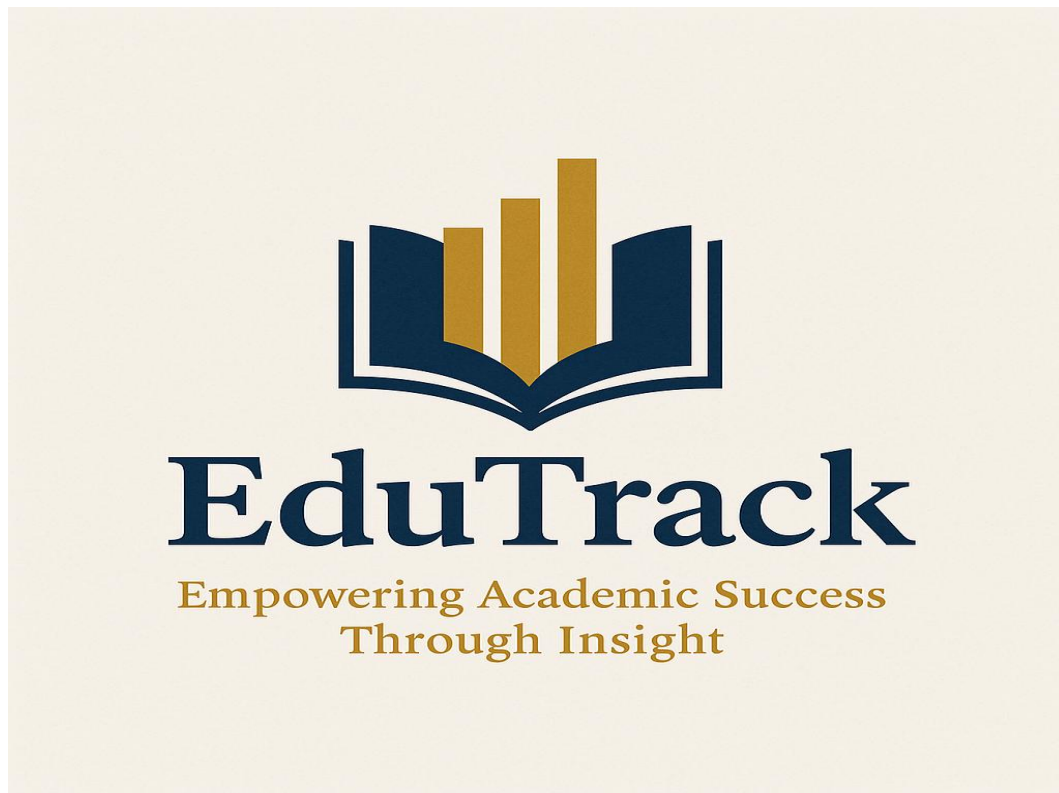


GROUP 1:

EduTrack

COSC 612 / AIT 624

SPRINT 1



Group Members:

Naga Dhanushya Ram Munnanuru

Stephen Aboagye-Ntow

Muhammad Adam

Ayandayo Adeleke

Ravinder Maini

TABLE OF CONTENTS

03 Resume

08 Planning and Scheduling

09 Communication and Collaboration

11 Teamwork Basics and Summary

14 Problem Statement

17 System Requirements and Architecture

22 Appendix

Naga Dhanushya Ram Munnannuru

Towson, MD | nmunnannuru@gmail.com | (704) 910-7707 | LinkedIn | GitHub

EDUCATION

Towson University, Master of Science in Computer Science | Towson, MD Jan. 2025 - Dec. 2026

- **GPA:** 3.77/4.0
- **Relevant Coursework:** Database Management Systems, Operating System Principles, Computer Networks.

Geethanjali College of Engineering and Technology, Bachelor of Technology in Computer Science and Engineering – Data Science | Hyderabad, India Aug. 2020 – Aug. 2024

- **GPA:** 3.0/4.0
- **Relevant Coursework:** Data Structures, Artificial Intelligence, Machine Learning, Object Oriented Programming.

SKILLS

Programming Languages: Java, SQL, Python, HTML.

Databases, Technologies & Tools: SQL Server, Anaconda, Jupyter Notebook, Eclipse IDE.

Operating Systems & Software: Linux, Microsoft Office Suite (Excel, PowerPoint, Word), Google Suite.

WORK EXPERIENCE

Trizula Digital Services, Data Analyst Intern | Hyderabad, India Sept. 2024 – Jan. 2025

- Collaborated with a cross-functional team to analyze and interpret large datasets, demonstrating strong communication and teamwork.
- Developed and delivered actionable insights through data visualizations, improving stakeholder decision-making.

IICL, Data Analyst Intern | Hyderabad, India Feb. 2024 – May 2024

- Led data collection and cleansing initiatives, ensuring high-quality inputs for analysis.
- Engaged with clients to gather requirements and tailored deliverables to meet business objectives.

Cantilever Labs, Intern (Brain Tumor Detection) | Hyderabad, India July 2022 – Sept. 2022

- Implemented a CNN-based pipeline for MRI/CT brain tumor detection using Python (Anaconda) with data normalization and augmentation.
- Documented UML diagrams and testing procedures to ensure reproducibility and audit ability.

PROJECTS

ORBITEl - Job Portal System | Towson University Jan. 2025 – May 2025

- Built a full-stack web application using Spring Boot, Spring Security, JPA (Hibernate), Thymeleaf, MySQL, and H2 for academic evaluation.
- Enabled role-based access control for Admin, Employer, and Job Seeker with CRUD operations and secure authentication.

Satellite Image Time Series Analysis for Crop Mapping | Geethanjali College of Engineering and Technology Dec. 2023 – May 2024

- Conducted crop mapping using U-Net and Sentinel-2 data, achieving 90% accuracy through rigorous data preprocessing.
- Applied problem solving skills to address data imbalances and collaborated with peers to enhance model performance.

Real-Time Image Animation System | Geethanjali College of Engineering and Technology July 2023 – Dec. 2023

- Developed a deep learning-based real-time image animation pipeline targeting video-conferencing, gaming, and AR use-cases.
- Adapted to challenges during implementation, demonstrating problem solving and adaptability in ensuring system efficiency.

Brain Tumor Detection Using ML | Geethanjali College of Engineering and Technology July 2022 – Sept. 2022

- Designed a machine learning model for automated brain tumor detection, reducing reliance on manual diagnostics.
- Communicated findings with clarity to nontechnical stakeholders, showcasing adaptability and collaboration.
- MRI and CT imaging data sets were used, ensuring robust and reliable results.

PUBLICATIONS

- **Co-Author:** *Satellite Image Time Series Analysis for Crop Mapping Using U-Net, Sentinel Dataset*, International Journal of Novel Research and Development (IJNRD), Vol. 9, Issue 4, April 2024.

STEPHEN ABOAGYE-NTOW

Towson, MD |
SaboagyI@students.towson.edu

PROJECTS: Understanding and Addressing the Impacts of Wetland Mowing to Facilitate Meeting the Chesapeake Bay Wetland Enhancement Goals - 2024

PUBLICATION: Location, biophysical and agronomic parameters for croplands in northern Ghana

EXPERIENCE

TEACHING ASSISTANT

Computer and Information Science Department | Towson University

2024 – PRESENT

- Evaluate and Grade assignments, quizzes and Labs with timely and constructive feedback
 - Proctor Quizzes and Exams
-

JUNIOR DATA SCIENTIST

Consortium for Digitalization of Climate Data | Accra, Ghana
2023

- Profiled the entire Climate Data Store of Ghana
- Field Data Collection

RESEARCH TECHNOLOGIST

Ghana Space Science and Technology Inst. | Accra, Ghana
2019 - 2023

- Assisted Research Scientists with data analytics and Map production.
-

EDUCATION

PH. D STUDENT IN INFORMATION TECHNOLOGY

Towson University

JUNE 2024 - PRESENT

- Relevant coursework: Software Engineering and Data Mining
-

SKILLS AND TOOLS

- **Programming Languages:** R, Python, C++, Bash Scripting
 - **Earth Observation and Map production:** QGIS and ArcGIS
 - **Statistical and Machine Learning Modeling**
-

Ayandayo Adeleke

Owings Mills, MD | (703) 459-5202 | ayandayoadeleke@gmail.com | [linkedin.com/in/ayandayo-adeleke](https://www.linkedin.com/in/ayandayo-adeleke)

SUMMARY

PhD student in Information Technology with research interests in human-computer interaction (HCI). Experienced AWS Solutions Architect with a proven track record in designing and implementing scalable cloud solutions that leverage cloud technologies to solve complex challenges.

EXPERIENCE

Amazon Web Services

Arlington, VA

Solution Architect

Jan 2022 – Jan 2025

- Collaborated with sales teams to deliver proof-of-concept architectures and technical workshops, contributing to successful deal closures and expanded customer usage.
- Led 100+ technical sessions, architecture reviews, and modernization strategies, improving customer satisfaction across diverse clients.
- Architected a cost-optimized cloud solution that cut storage costs by 50% and improved reliability by 10%, driving AWS service adoption.
- Designed a semantic search solution using AWS services, reducing support queries by 20% and boosting user engagement
- Implemented disaster recovery solutions that met RTO/RPO requirements, ensuring business continuity.
- Delivered technical workshops and led sessions for underserved communities through AWS Impact Accelerator programs.

Ministry of Finance

Ibadan, Nigeria

Accounting Assistant

Jan 2017 – October 2017

- Reviewed financial records to extract and verify key data for reconciliation reports.
- Ensured accuracy in ledger by identifying discrepancies in bank and financial statements.
- Maintained documentation and data logs to support audits and internal reviews.

LEADERSHIP EXPERIENCE

Vice President, Jerusalem Youth Fellowship

2014 - 2016

- Assisted in the planning and execution of youth events, including social gatherings and outreach programs
- Oversaw program activities, mentoring and volunteer efforts for youth engagement and impact

EDUCATION

Towson University

Towson, MD

PhD Information Technology (In Progress)

Bowie State University

Bowie, MD

M.S. Management Information Systems, GPA: 3.9/4.0

Ekiti State University

Ekiti, Nigeria

B.S Economics

CERTIFICATIONS

AWS Certified Solutions Architect - Professional

Oct 2023

AWS Certified Data Analytics - Specialty Certification

Dec 2023

AWS Developer Associate

Dec 2022

AWS Certified Solutions Architect - Associate

April 2022

SKILLS

Technical Skills: Python, R, SQL, AWS Solutions Architecture, Excel

Soft Skills: Strong Communication, Team Collaboration, Detail Oriented

RAVINDER MAINI

Laurel, MD | (301) 828-0960 | ravindermaini001@gmail.com | linkedin.com/in/raviindermaini

PROFESSIONAL SUMMARY

Experienced Software Test Engineer with proven expertise in designing, testing, and maintaining software systems. Skilled in automation, debugging, and QA process optimization using Selenium and Agile methodologies. Recognized for improving testing efficiency, minimizing downtime, and delivering high-quality, reliable software solutions that enhance user experience and operational performance.

TECHNICAL SKILLS

Languages: Java, Python, HTML, CSS, JavaScript, SQL

Frameworks & Tools: OOP, Flask, MySQL, Selenium, GitHub, JIRA, AWS

Software: Microsoft Office Suite, Visio, Outlook, Excel

Certifications: DHS Trusted Tester, Section 508 Standards

Methodologies: SDLC, STLC, Agile, Scrum, Waterfall

Accessibility Tools: ANDI, Screen Reader

PROFESSIONAL EXPERIENCE

Software Test Engineer | USCIS | Jun 2024 – Present

- Automated manual testing scenarios using Selenium, boosting test coverage and reducing manual effort.
- Partnered with developers and leads to resolve functionality issues and enhance workflow efficiency.
- Designed and maintained Requirement Traceability Matrix (RTM) and detailed test plans ensuring full coverage.
- Conducted regression and integration testing improving stability and overall quality.
- Enhanced UI/UX for improved accessibility and user satisfaction.
- Executed regression test suites validating post-release integrity.

Software Test Engineer Intern | SpecsWorks | May 2023 – Feb 2024

- Applied QA best practices under Agile, Scrum, and Waterfall environments.
- Developed and executed test plans, identifying and documenting defects efficiently.
- Collaborated with developers for debugging and process improvement.
- Enhanced existing application interfaces, improving performance and usability.
- Performed regression and integration testing, ensuring product quality before release.

EDUCATION

Master of Science in Computer Science – Towson University | Expected Dec 2026 | GPA: 4.00/4.00

Bachelor of Science in Computer Science – University of Maryland Global Campus | Aug 2022 | GPA: 3.78/4.00



Muhammad Adam

Place of birth: Nigeria | **Nationality:** Nigerian (Nigeria) | **Phone number:**

(+1) 4104991894 (Mobile) | **Email address:** madam2@students.towson.edu |

Address: 42 Solar Cir, 21234, Baltimore, United States (Home)

WORK EXPERIENCE

 **MATRIX INTERNATIONAL ACADEMY, GOMBE** – GOMBE, NIGERIA

MANAGEMENT INFORMATION OFFICER (MIO) – 01/09/2019 – 25/12/2024

 **CLARITAE VIDERE LIMITED** – LAGOS, NIGERIA

MACHINE LEARNING INTERN - REMOTE – 16/12/2021 – 30/03/2023

EDUCATION AND TRAINING

CURRENT Baltimore, United States

PH.D IN INFORMATION TECHNOLOGY (IN VIEW) Towson University

Website <https://www.towson.edu/>

16/11/2011 – 05/10/2015 Gombe

B.SC (HONS) COMPUTER SCIENCE Gombe State University

Website <https://gsu.edu.ng/home/>

SKILLS

Computer vision | Deep Learning | Machine Learning | Python | Java

RECOMMENDATIONS

Dr. Qingqing Li Assistant Professor

Academic advisor.

Email qingqingli@towson.edu | **Phone** (+1) 4107044532

PROJECTS

Computerized Hospital Billing System.

A research work submitted to the Department of Mathematics, Faculty of Sciences, Gombe State University in partial fulfillment for the award of the Degree in Computer Science.

PUBLICATIONS

2025

[Deep Learning Approaches for Automatic Livestock Detection in UAV Imagery: State-of-the-Art and Future Directions](#)

Adam, M., Song, J., Yu, W., & Li, Q. (2025). Deep Learning Approaches for Automatic Livestock Detection in UAV Imagery: State-of-the-Art and Future Directions. Future Internet, 17(9), 431. <https://doi.org/10.3390/fi17090431>

Planning and Scheduling

Table 1

Assignee	Email	Task	Duration	Dependency	Due Date
Naga Dhanushya Ram Munnannuru (Coordinator)	nmunnan1@students.towson.edu	Github Repo Setup and Problem Statement	4 hours	Meeting Notes	10/02/2025
Muhammad Adam	madam2@students.towson.edu	Systems Requirements & Architecture	5 hours	Meeting Notes and Personal Expertise	10/04/2025
Ayandayo Adeleke	aadelek7@students.towson.edu	Teamwork Basics and Summary	3 hours	Meeting Notes	10/03/2025
Ravinder Maini	rmaini1@students.towson.edu	Resume Collection and Formatting	2 hours	None	10/03/2025
Stephen Aboagye-Ntow	sabaogy1@students.towson.edu	Report Compilation and Final Review	4 hours	All tasks	10/05/2025

Communication and Collaboration

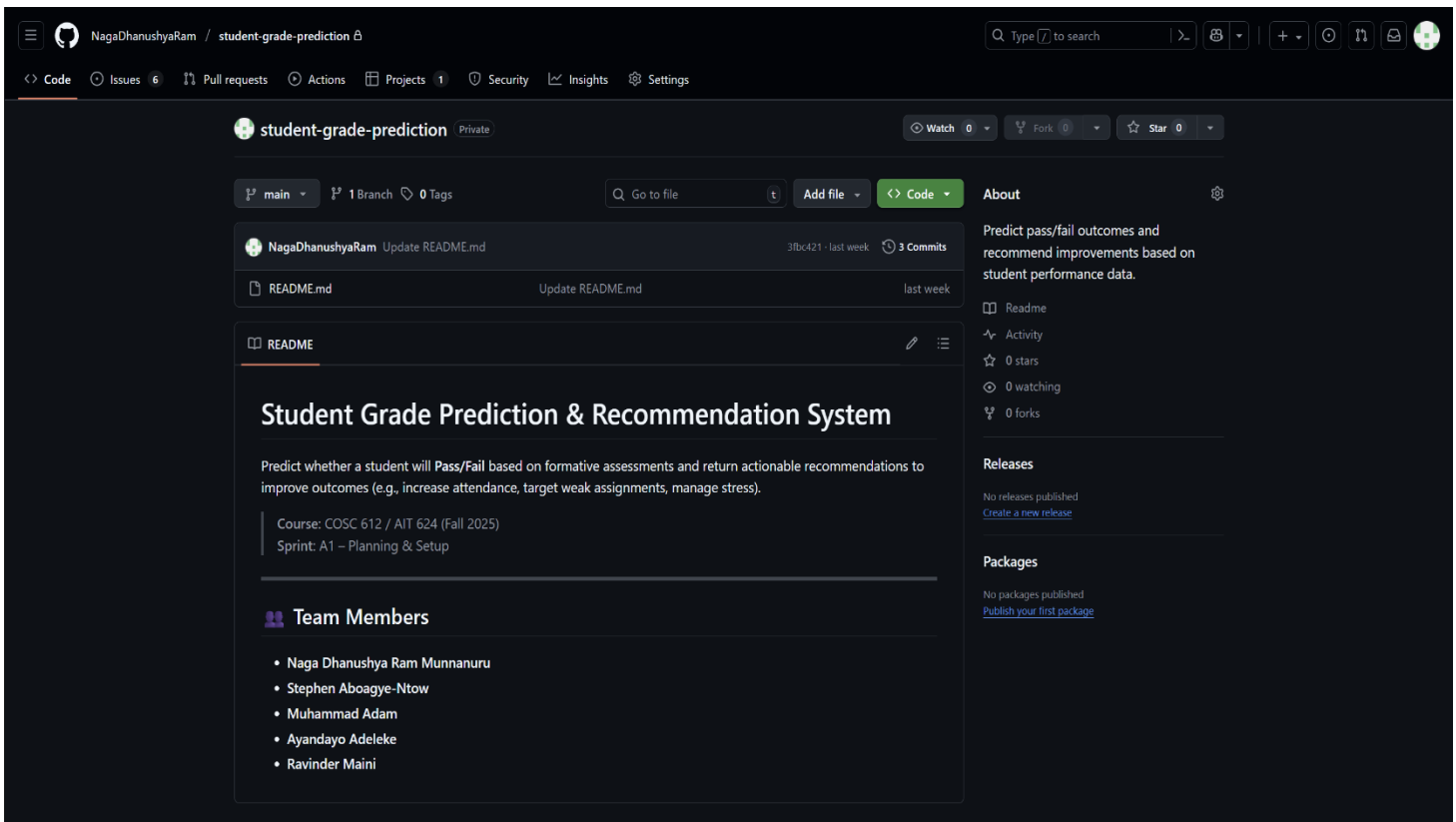


Fig 1

NagaDhanushyaRam / Projects / SWE - Group 1

Q Type to search

>

+

SWE - Group 1

Add status update

Insights

Workflows 4

View 1

+ New view

Q Filter by keyword or by field

Discard

Save

0

Todo

This item hasn't been started

+ Add item

1

In Progress

This is actively being worked on

student-grade-prediction #6

A1 Report Review and Submission

+ Add item

1

Review

It's done but requires reviewing

student-grade-prediction #5

Documents Collection and Assembling the Report

+ Add item

4

Done

It's Reviewed and ready to submit.

student-grade-prediction #1

Github Repository Setup and Add Collaborators

student-grade-prediction #2

Write the Problem Statement after researching the topic in detail

student-grade-prediction #3

Teamwork Basics Summary and Understanding

student-grade-prediction #4

Compile System Requirements and a System Architecture Diagram

+ Add item

Fig 2

1 Teamwork Basics Summary

The Teamwork Basics document provides guidelines for accomplishing team tasks while ensuring higher member satisfaction. To achieve these goals, it is important to understand the team's strengths and weaknesses, set ground rules, and have good communication plans, including strategies to address potential problems within the team.

1.1 Ground Rules

Ground rules provide guidelines for getting things done and ensure that every team member understands the process for completing tasks and achieving set goals. It is important to clearly communicate how work will be completed and task assigned, which would help prevent conflicts of interest. There are five important norms to keep in mind:

- **Work Norms:** This would guide the team in dividing tasks, setting deadlines, assigning responsibilities, reviewing work, resolving disagreements, and managing different work habits. For example, in a previous group project for a Decision Support class during my master's, one member consistently waited until the last minute to complete tasks and was often unresponsive. To avoid delays, another team member took on the responsibility of completing that task, which created an extra burden. This highlights the importance of establishing and communicating clear work norms to ensure fairness and effective teamwork.
- **Facilitator Norms:** A facilitator plays an important role in keeping the team on track, ensuring tasks are completed on time, encouraging participation from team members, and ensuring decisions are made effectively. It is also recommended to rotate the facilitator role. This allows the team to experience different facilitation styles, which can enhance efficiency and collaboration.
- **Communication Norms:** Effective communication is essential to a team's success. A group should agree on communication methods that work best for everyone. It is important to recognize that some members may prefer in-person discussions, some email or text, and others video calls. For example, in a previous database class project, our group primarily used WhatsApp chats and held video calls when necessary, which accommodated everyone's preferences and kept the team coordinated. Understanding and respecting these preferences helps prevent misunderstandings and ensures smooth collaboration.
- **Meeting Norms:** Since team members have different schedules and preferences for meeting times and locations, it is essential to discuss and agree on arrangements that allow everyone to participate fully. For example, in this class group, team members have various commitments such as jobs and family responsibilities. Some can meet on weekends, while others cannot, so finding a time and place that works for everyone is

important. The team should also establish expectations for attendance and punctuality, including consequences for lateness or missing meetings.

- **Consideration Norms:** In a team, members have different personalities and working styles, so it is important to recognize what makes others comfortable or uncomfortable. Some members may dominate discussions, while others may get distracted during meetings. Establishing clear rules about acceptable behavior helps create a respectful and safe environment, and the facilitator should also be prepared to manage such situations effectively.

1.2 Hints for Handling Difficult Behavior

Teams are made up of members with different personalities and perspectives, and difficult behaviors can sometimes hinder productivity and disrupt the successful completion of tasks. To resolve such problems, here are some suggestions:

- **Overly Talkative:** An overly talkative person could be someone who likes to show off their skills, experience, or knowledge, and they often do not give others the opportunity to contribute. A few ways to resolve this include using humor to redirect the conversation, having the facilitator step in to ensure equal participation, or having a team member speak to the person privately in a polite, non-offensive manner.
- **Too Quiet:** A team member may be overly quiet due to shyness, fatigue, boredom, or personal circumstances, which can prevent active participation. The facilitator and other team members can support this person by asking for their opinions, creating a relaxed environment, and appreciating their contributions. For example, in a previous class project, one team member was initially very quiet, rarely participating in messages or meetings. However, by the middle of the semester, the team member became more engaged after being encouraged to share inputs on assigned tasks. Assigning responsibilities can also help quiet members feel more comfortable contributing.
- **Argues:** A team member who argues consistently can slow progress and potentially create conflict in the group. A person could argue constructively, such as critically evaluating ideas, which can be beneficial to the team's productivity and work quality. It is important to assess whether the ideas contribute positively to the team's goals. However, if the arguments are consistently critical of others, it should be addressed and made known to the team member that the behavior is detrimental to the team's progress.
- **Complains:** Constant complaining by a team member can hinder team progress. The team should listen to the concern and evaluate whether it is valid and actionable. The person raising the issue should also be encouraged to suggest solutions on how to address the problem.

1.3 Hints for Handling Group Problems

Working in a team with diverse perspectives can sometimes lead to difficulties which could delay projects and affect productivity. Here are some ways to handle common group problems:

- **Floundering:** A group may flounder when they are struggling to move forward or make decisions. This often happens when the group is new, and members are still getting to know each other. To resolve this, creating a clear task list and defining specific goals can help provide direction and focus.
- **Going Off on Digressions and Tangents:** This occurs when group members get caught up discussing topics unrelated to the project. While little of this can help build relationships, excessive digressions can hinder progress. It is important for the team to recognize when to return to the main agenda. For example, in my previous job, team meetings often included casual chats to break the ice, but the manager always redirected the group back to the main discussion after a few minutes.
- **Making a Decision Too Quickly:** Making rushed decisions can negatively impact the team's progress and may create setbacks. Some members may be more action-oriented and tend to make decisions faster than others, which can pressure the group to move forward without fully evaluating options. The facilitator or other team members can help by asking key questions to ensure decisions are carefully considered and supported by all members of the team.
- **Not Making a Decision:** Just as making decisions too quickly can be problematic, not making a decision can also hinder the team's progress. The best approach is to ensure all team members are in agreement. If the team struggles to reach consensus, each member of the team can vote on their top choices, and the team can proceed with the option that receives the most support.
- **Feuding Between Group Members:** Conflicts between team members can significantly delay progress. To address this, the issue should be discussed, and active listening techniques can be used to help resolve conflicts.
- **Ignoring or Ridiculing Others:** There are possibilities of having factions or subgroups within a group, which can lead to some members being or feeling excluded. It is important to know how to work with all team members, including those we may not feel comfortable with. Every member should make a conscious effort to work effectively with everyone in the group.
- **The Group Member Who Does Not Do His/Her Share of the Work:** A team member may be unresponsive, skip meetings, or fail to complete assigned tasks. Addressing the issue directly by discussing how their behavior affects the group is important, as it can save time, reduce stress, and help the team stay on track.

2 Problem Statement

2.1 What is your product, on a high level?

Our product is a Student Grade Prediction & Recommendation System, a machine learning-based web application that predicts whether a student will pass or fail a course. The system provides students with their chances of passing as well as tailored advice (e.g., attend classes more often, study longer, decrease stress levels) to improve their academic standing.

2.2 Whom is it for?

The system exists for:

- **Students:** Perform predictions and get tailored recommendations to improve their academic standing.
- **Teachers / Advisors:** Monitor student performance, identify at-risk students and if possible, provide personalized guidance to help them succeed.
- **Administrators:** Manage user accounts, control system access, and oversee platform operations.

2.3 What problem does it solve?

The current gradebook systems, which include Blackboard and Canvas show students their actual scores but they lack any system to alert them about their performance or offer help. Students discover their risk status only after completing their final exams which creates limited time for improvement. This system solves the problem by:

- The system enables users to predict student achievement results before the official grades are released.
- Generating recommendations based on the student situation and helping them recognize their potential low points to improve.

2.4 What alternatives are available?

- Learning Management Systems (LMS) like Blackboard and Canvas show performance data but lack predictive analytics and personalized recommendations.
- Blackboard Analytics is commercial, costly, and targets administrators, not students.
- No widely available low-cost system provides student-level explainable predictions with recommendations.

2.5 Why is this project compelling and worth developing?

The system offers:

- **Early Intervention:** helping at-risk students succeed before it's too late.
- **Accessible design:** lightweight web interface (Streamlit GUI).
- **Affordable Deployment:** built entirely with free/open-source tools and deployable on free cloud platforms.
- **Dual Academic & Operational Value:** demonstrates ML research and provides universities with a working tool to improve retention and success.

2.6 Top-level objectives, differentiators, target customers, and scope.

- **Objectives**
 - Assess the potential success of a student.
 - Show the chances of success together with essential risk elements.
 - The program should create individualized plans which help students develop their study methods and create effective daily routines.
- **Differentiators**
 - The system merges machine learning prediction models with rule-based recommendation systems.
 - The system bases its operation on explaining its decision-making steps instead of achieving the highest possible accuracy rates.
 - Users can train models with new data through the system while performing batch predictions from imported CSV files.
- **Target Customers**

University students, advisors, and instructors.
- **Scope**
 - Cloud-hosted web application.
 - The system requires student data entry functions and CRUD operations and an ML-based prediction system.
 - Dashboards for students and advisors.
 - Messaging feature between students and their teachers.

2.7 What are the competitors and what is novel in your approach?

- The competition includes both Learning Management System dashboards (Canvas and Blackboard) and commercial analytics platforms.
- **Novelty**
 - The system integrates explainable AI technology which reveals the specific reasons behind its failure prediction for students.
 - The system implements low-cost deployment through its use of open-source frameworks including scikit-learn and Streamlit and pandas.
 - Provides personalized recommendations that can be customized by advisors.

2.8 Can the system be built with available resources and technology?

Yes. The system operates by:

- **Front-end:** Streamlit
- **Back-end / ML Engine:** Python 3.10+, scikit-learn, pandas, NumPy, joblib.
- **Database:** MongoDB
- **Hosting & Deployment:** Hugging Face Spaces or Streamlit Community Cloud for early versions; Docker + Render / Railway for scalable deployment.
- **Baseline Models:** Decision Trees, expandable to Deep Learning Architectures.

2.9 What is interesting about this project from a technical point of view?

- The high-performance ML models achieve the following results: Accuracy ~92.8%, F1 ~94.1%, ROC-AUC ~98.6%.
- Streamlit GUI provides users with an interactive two-column interface that enables real-time prediction functionality.
- The Recommendation Engine follows two rules which state that students who attend less than 75% of classes need to improve their attendance and students who experience stress levels above 7 should receive stress management strategies.
- The Explainable AI system shows which specific features (attendance, assignments, study hours) affect the prediction results.
- Cloud-ready architecture exists as a deployable system that uses Docker and open-source hosting platforms.

3 System Requirements and Architecture

3.1 Purpose

The purpose of this project is to develop a machine learning–based application that predicts whether a student will **pass or fail**, given their academic, behavioral and socio-economic data. The application also provides **personalized recommendations** to help students improve their performance.

3.2 Scope

- **Users**
Students, teachers, and administrators.
- **Functionality:**
 - Input student academic, behavioral and socio-economic data.
 - Predict pass/fail status.
 - Provide the probability of passing.
 - Recommend actionable steps to improve performance.

3.3 Benefits

Early detection of at-risk students, better academic support, data-driven interventions.

3.4. System Requirements

3.4.1 Functional Requirements

- **Data Input:**
 - Users enter student features (e.g., attendance, study hours, exam scores, stress level).
 - System validates and processes inputs.
- **Prediction:**
 - Predict pass/fail using a trained ML/Deep learning model.
 - Display probability score.
- **Recommendations:**
 - Generate personalized improvement advice based on weak student attributes (e.g., low attendance, high stress).
- **User Interface:**
 - Provide an intuitive GUI for data entry.
 - Display results and recommendations clearly.
- **Extensibility:**
 - Allow retraining of the model with new data.
 - Enable batch predictions from uploaded CSV files (future work).

3.5 Non-Functional Requirements

- **Usability:** Easy-to-use interface with minimal training.
- **Performance:** Predictions should run in <2 seconds per input.
- **Portability:** Works cross-platform (Windows, macOS, Linux).
- **Maintainability:** Well-structured code with modular design.
- **Security:** Exclude personally identifiable information (PII) from the dataset.

3.6 System Environment

- **Language:** Python 3.10+
- **Frameworks:**
 - scikit-learn (ML model training and prediction)
 - streamlit (frontend GUI)
 - pandas & numpy (data preprocessing)
 - joblib (model serialization)
- **Deployment Options:**
 - Run with Streamlit
 - Docker container
 - Cloud (Hugging Face Spaces, Render, Heroku, etc)

3.7 System Design

3.7.1 Architecture Overview

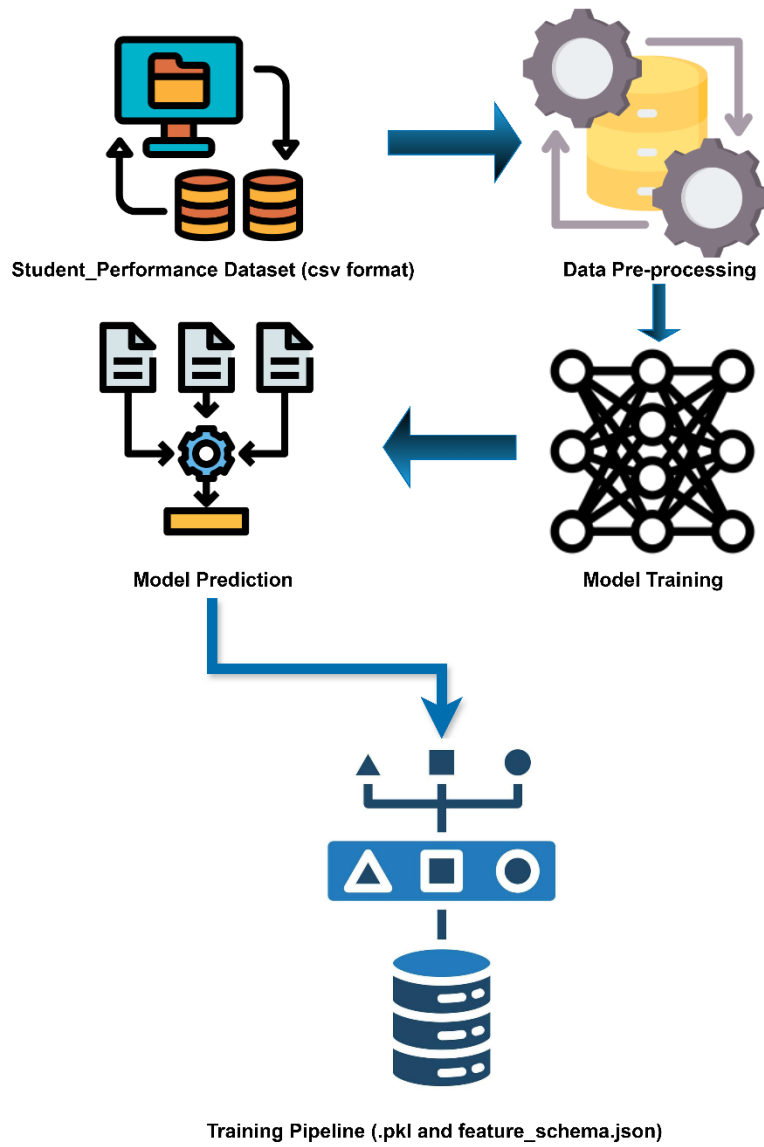


Fig 3 Machine Learning Model Training

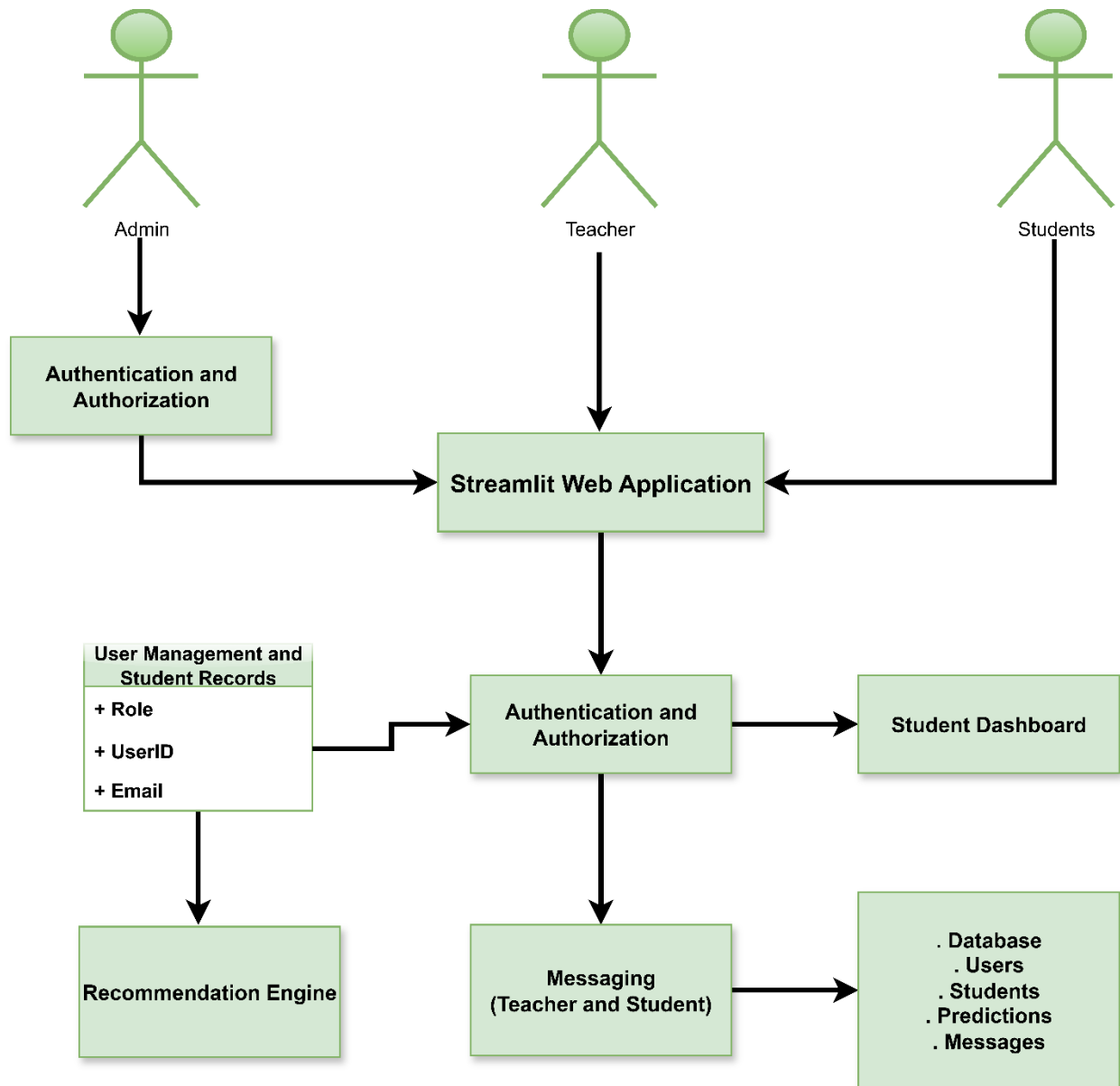


Fig 4 System Architecture

3.7.2 Data Flow

- User enters features → GUI.
- Features packaged into a pandas DataFrame.
- Data passed to the trained Machine learning pipeline.
- Model outputs include pass/fail status and associated probability.
- Recommendations engine generates advice.
- GUI displays prediction + recommendations.

3.8. Dataset

3.8.1 Sources

- **Students Performance Dataset** (provided).
- Features include demographics, study habits, attendance, exam scores, etc.

3.8.2 Preprocessing

- Dropped duplicates and PII (IDs, names, emails).
- Derived label: **Pass if Grade $\in \{A, B, C\}$ or Total_Score ≥ 60** , otherwise Fail.
- Encoded categorical features with OneHotEncoder (is a tool for data pre-processing in ML used to convert categorical data to a numerical data).
- Imputed missing numeric values with **the median**, categorical with **the mode**.

3.9. Implementation

3.9.1 Model Training

- **Algorithm:** We propose either a machine learning or a deep learning model, and then we can choose the best-performing one as the pipeline.
- **Evaluation Metrics:** The following evaluation metrics should be used to see and determine how well our best model performs.
 - Accuracy $\approx 92.8\%$
 - F1 Score $\approx 94.1\%$
 - ROC-AUC $\approx 98.6\%$

3.9.2 GUI Development

- Built with **Streamlit**.
- Two-column layout for balanced feature entry.
- Real-time prediction after clicking **Predict**.

3.9.3 Recommendation Engine

Rule-based system using thresholds:

- Attendance $< 75\% \rightarrow$ Improve attendance.
- Study hours $< 12 \rightarrow$ Increase study hours.
- Participation $< 50 \rightarrow$ Engage more in class.
- Sleep < 7 hours \rightarrow Sleep 7–8 hours daily.
- Stress $\geq 7 \rightarrow$ Stress management strategies.

Appendix

Table 1

Planning and Task Schedule

Fig 1

Github Repository Homepage

Fig 2

Kanban Board on Github

Fig 3

Machine Learning Pipeline

Fig 4

System Architecture