GRADE-EASE: AI-ASSISTED EVALUATION

Group 17:

1. Sanjo Simon (VAS20CS131)
2. Razin Ahmad Abbas (VAS20CS091)
3. Sandra Mariya Benny (VAS20CS099)
4. Vishnu PR (VAS20CS122)

Project Guide:
Ms. Ramya Raj
Dept CSE, VAST

OVERVIEW

1. Introduction
2. Problem Statement
3. Literature Review
4. Proposed System
5. Objective
6. Architecture
7. System Requirements
8. Methodology
9. Design Formulation
10. Implementation
11. Resource And Tasks
12. Planning And Scheduling
13. Result Analysis
14. Conclusion
15. References

INTRODUCTION

- Examinations have always been part of every educational, and non-educational organization. Examinations can be either descriptive or objective or both.

- Every examination needs evaluation. The majority of competitive exams are objective in structure. They happen on similar machines that have been examined.

- These systems, or any other related methods, offer greater advantages in terms of resource conservation. However, it has been noticed that these systems can only contain multiple-choice questions and cannot be expanded to include subjective questions.

- These methods cannot be used in board exams or university exams where students give subjective answers, hence there is a need for software that will aid in conserving resources.

- It can be seen the amount of pressure that is held on the education system and teachers to evaluate the n number of answer copies of the students.

INTRODUCTION (CONTD....)

- Answer Verifier is required to grade the student after he or she has completed the question paper. The process of evaluating the descriptive answer will save resources and huge amount of time.

- It will also help to speed up the overall educational system because students will not have to wait for a long time. The evaluation of subjective answers has long been a challenge for teachers.

- Grade-ease, powered by AI, has emerged as a solution to this challenge. These can analyse and evaluate subjective responses.

PROBLEM STATEMENT

Traditional methods of grading and assessment can be time-consuming. As a result, organizations may struggle to accurately assess the answers provided by employees or students, leading to suboptimal performance and decreased productivity. Traditional grading methods are susceptible to human bias and subjectivity. Different examiners may interpret answers differently, leading to inconsistent grading and potential unfairness.

PROPOSED SYSTEM

Grade-ease, a dynamic web application developed with Django, redefines the evaluation process for online tests by seamlessly integrating advanced pretrained models like BERT for semantic similarity assessment. It offers educators a comprehensive platform where they can create, administer, and, if necessary, manually edit marks for assessments.

Educators using Grade-ease benefit from its intuitive interface, empowering them to effortlessly generate online tests tailored to their curriculum requirements. Students can conveniently submit their answer papers digitally, triggering the automated evaluation process driven by sophisticated pretrained models.
While the automated evaluation provides a reliable baseline, teachers retain full control over the final grading, ensuring fairness and accuracy in assessment.

PROPOSED SYSTEM

Grade-ease acknowledges the importance of human intervention in refining the grading process. By providing educators with the ability to edit marks, Grade-ease fosters a collaborative approach to assessment, where technology complements rather than replaces human judgment.

Grade-ease represents a paradigm shift in educational technology, offering educators a powerful yet flexible tool to enhance their teaching practices. By combining the robustness of Django with advanced AI capabilities, Grade-ease not only streamlines assessment processes but also empowers educators to deliver personalized and impactful feedback to students.

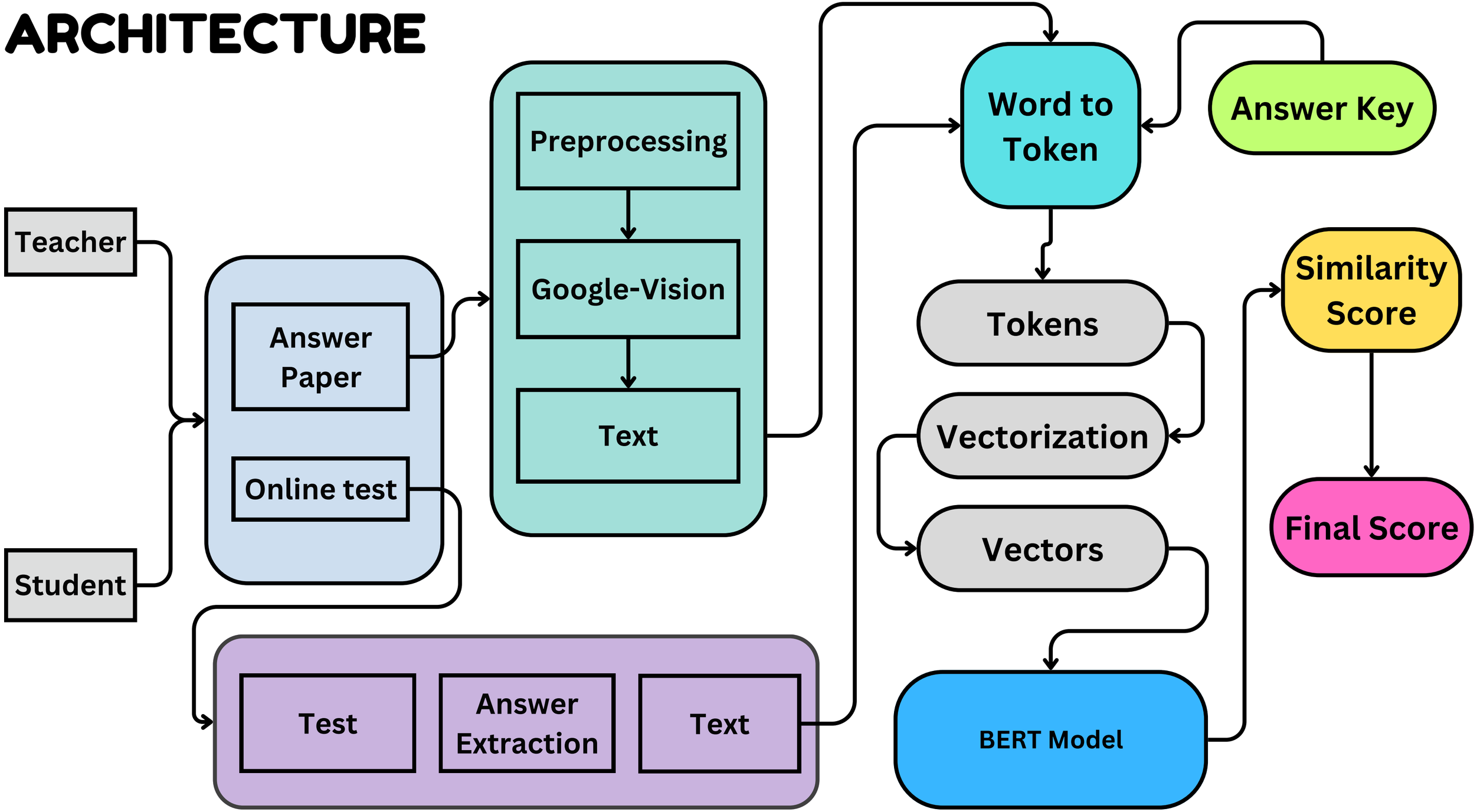
OBJECTIVE

- The objective of a subjective answer checking model is to automate the process of evaluating and grading subjective answers given by students to open-ended questions in educational assessments. The model should be able to assess factors such as relevance, coherence, clarity, and overall understanding of the topic.

- The main goals of a subjective answer checking model are:

1. To reduce the time and effort to evaluate large volumes of subjective answers, while maintaining the accuracy and fairness of the grading process.
2. To provide consistent and reliable grading of subjective answers, reducing the risk of human bias and variability.
3. To improve the quality of educational assessments by ensuring that subjective answers are graded based on standardized criteria and objective measures.

ARCHITECTURE



SYSTEM REQUIREMENTS

HARDWARE

Processor: i3 10 generation
Ram: 4GB
OS: Linux, Windows, Mac
Graphics: NVIDIA GeForce GTX 1650
Operating System Architecture: 64 bits

SOFTWARE

Google Colab
Anaconda version 3
VS Code

LIBRARIES

Python Version 3.9
Numpy
Pandas
Keras
Tensorflow

METHODOLOGY

DATASET:

- The SNLI corpus [9] (version 1.0) is a collection of 570k human-written English sentence pairs manually labeled for balanced classification with the labels entailment, contradiction, and neutral, supporting the task of natural language inference (NLI), also known as recognizing textual entailment (RTE).
- We aim for it to serve both as a benchmark for evaluating representational systems for text, especially including those induced by representation learning methods, as well as a resource for developing NLP models of any kind.

METHODOLOGY

MODEL BUILDING:

- Image text to raw text:
  Convert the handwritten text in a image to raw text by using google-vision

- Tokenize the input data:
  Convert the raw text input into numerical inputs that can be fed into the model.
  Hugging Face provides a tokenizer for each pre-trained model, which converts the text input into numerical input sequences.

- Bert, pre-trained model:
  Hugging Face offers a range of pre-trained models for different NLP tasks.We choose BERT model for the evaluvation

METHODOLOGY

MODEL TRAINING:

- Fine-tune the model:
  Fine-tuning involves training the pre-trained model on specific task using given dataset. Hugging Face provides a range of utilities for fine-tuning the pre-trained models, including trainers, optimizers, and schedulers.

- Evaluate the model:
  Evaluate the performance of the model on a validation set to determine how well it generalizes to new data. Hugging Face provides tools for evaluating the model's performance on a range of metrics, such as accuracy.

METHODOLOGY

MODEL DEPLOYING:

- Once the model has been trained and evaluated, it can be deployed using HTML, CSS, JavaScript in front-end and Django in back-end. Hugging Face provides tools for deploying the model as a web application.

DESIGN FORMULATION

- Python is used to develop Grade-ease. Python's extensive library ecosystem provides the tools and frameworks to build Grade-ease system, from natural language processing and machine learning models to web development and deployment.

- Google-vision model is the model used for analyzing images and videos. It extracts the handwritten text from the images.

- Additionally, Python frameworks such as Django is used to build web applications that provide a user-friendly interface for submitting and viewing answers.

DESIGN FORMULATION

- TensorFlow is the open-source library used for building and training machine learning models. TensorFlow provides a high-level API, Keras, that allows to build and train deep learning models for a variety of tasks, including natural language processing.

- Using TensorFlow, we train the models on large datasets of answers and their corresponding scores or probabilities and use those models to evaluate new answers submitted by users.

DESIGN FORMULATION

- Django is a Python web framework that is used to build web applications -Grade-ease. Using Django, Grade-ease can be built as a web application. The Flask application processes the answer using a pre-trained model or algorithm and returns a score or probability for the correctness of the answer.

- Django ability to handle HTTP requests and responses makes it a powerful tool for building Grade ease, which can be accessed through a web browser.

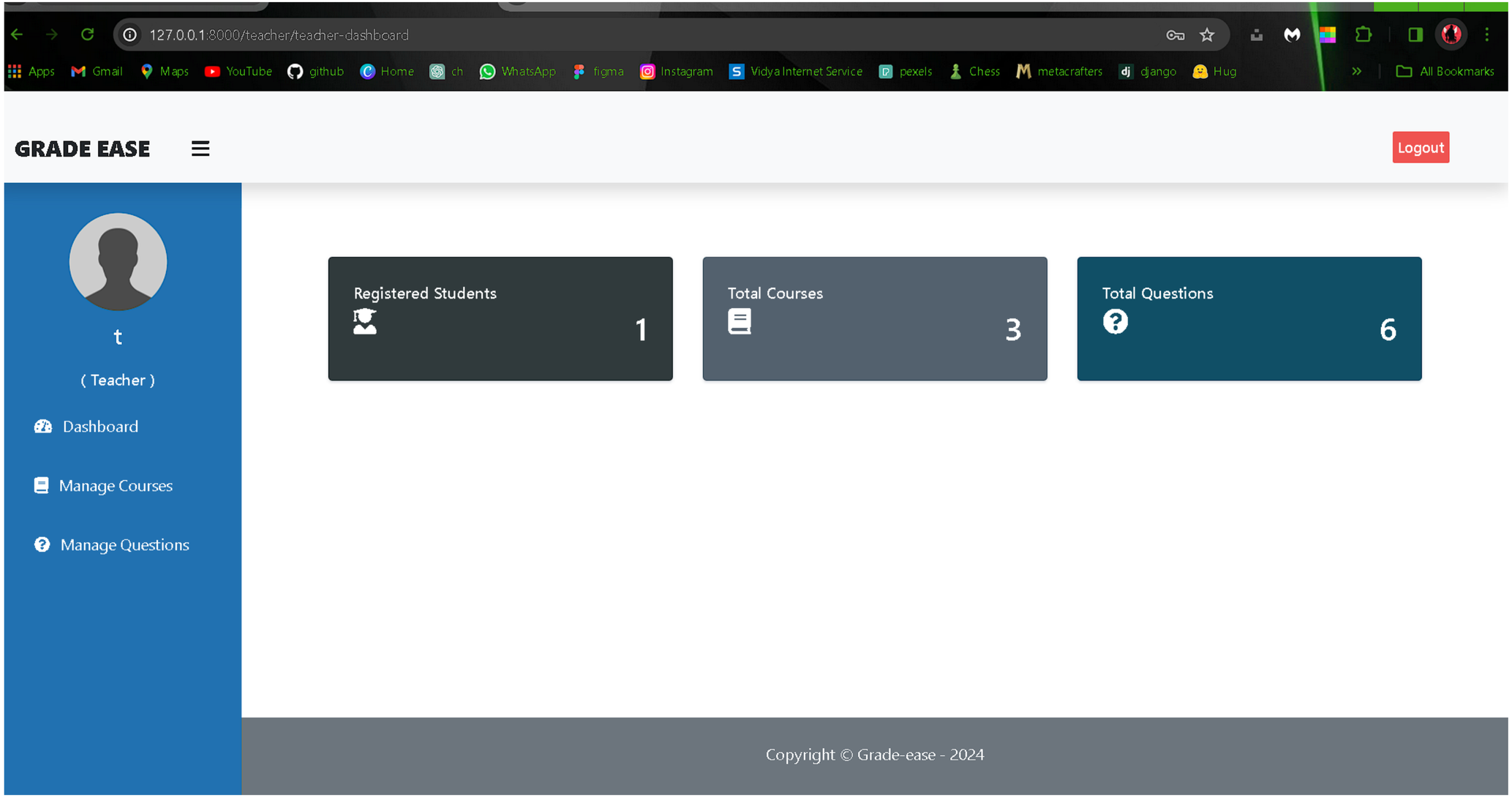
- Hugging Face Transformers is the open-source library that provides pre-trained models for natural language processing tasks such as text classification, question answering, and language 20 translation. It can be utilized as a Grade ease by comparing a student's response to a given prompt to the pre-existing knowledge within the model.

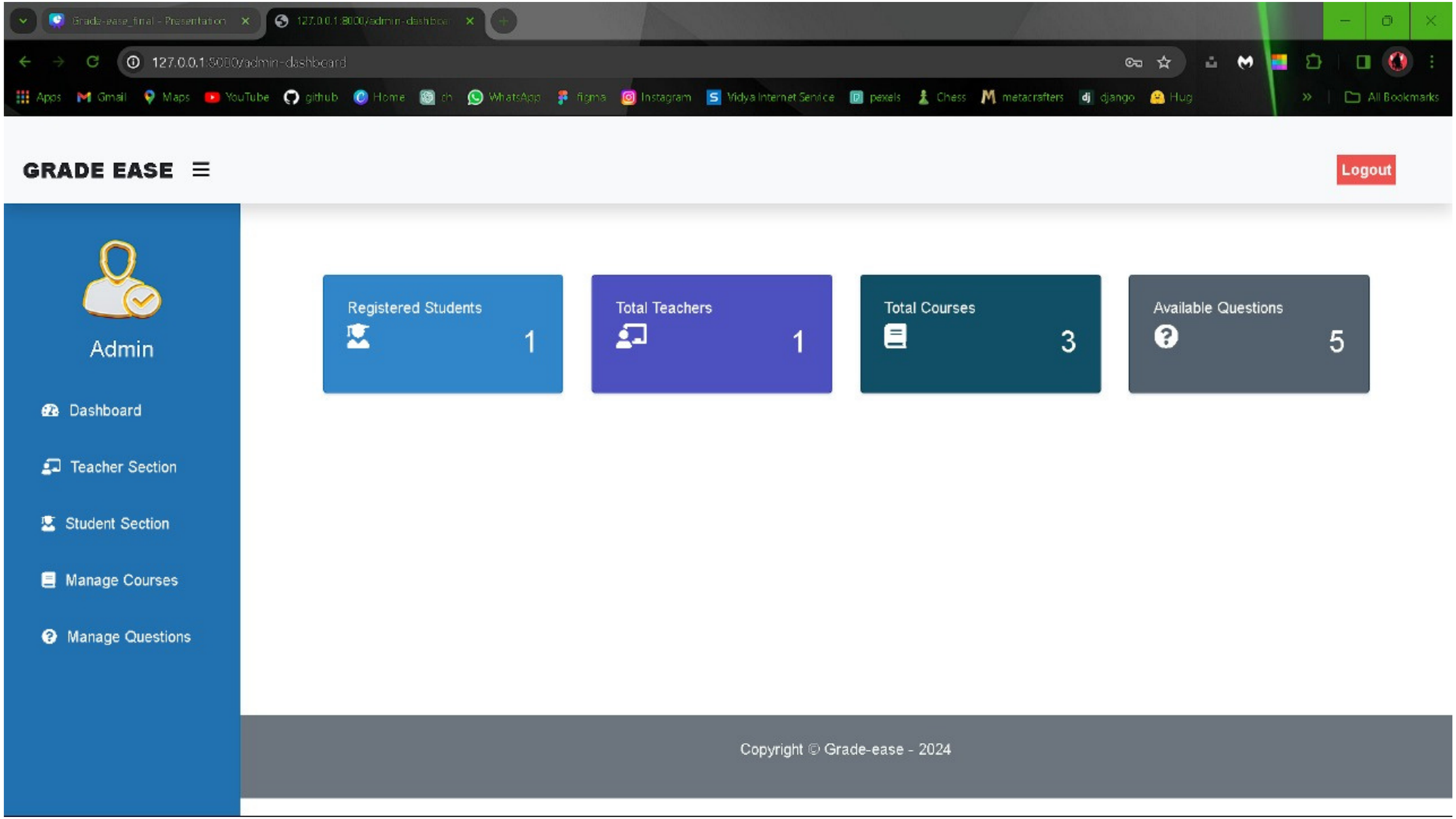
DESIGN FORMULATION

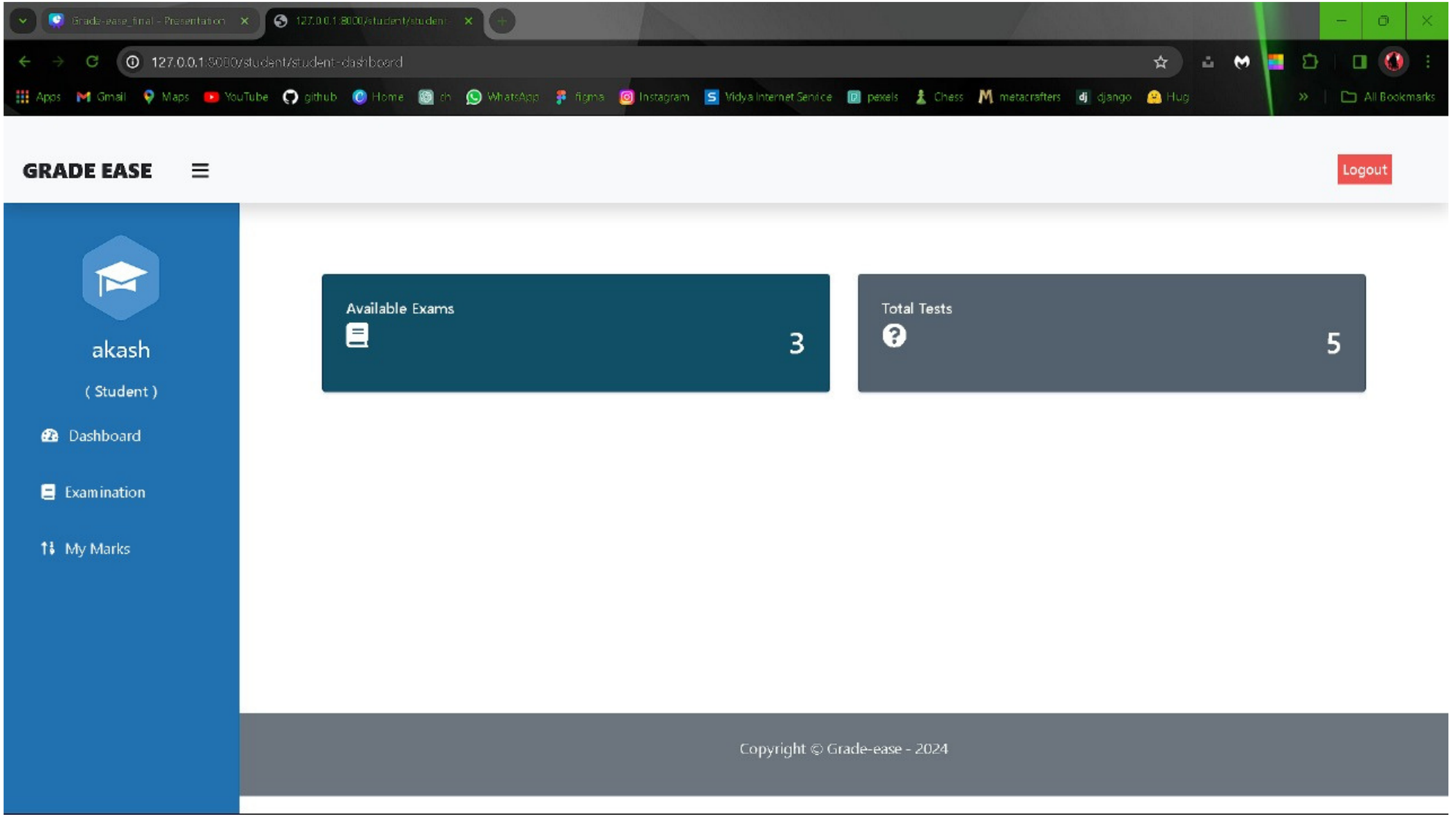
- HTML and CSS can be used to build the front-end of Grade-ease, providing a user-friendly interface for submitting and viewing answers. HTML provides the structure and content of the webpage, including forms for users to submit their answers and CSS used to style the page and create an intuitive layout for users to navigate.

- This front-end interface can be connected to back-end technologies such as Django, trained models to process and evaluate submitted answers, providing an end-to-end solution for subjective answer checking.

IMPLEMENTATION



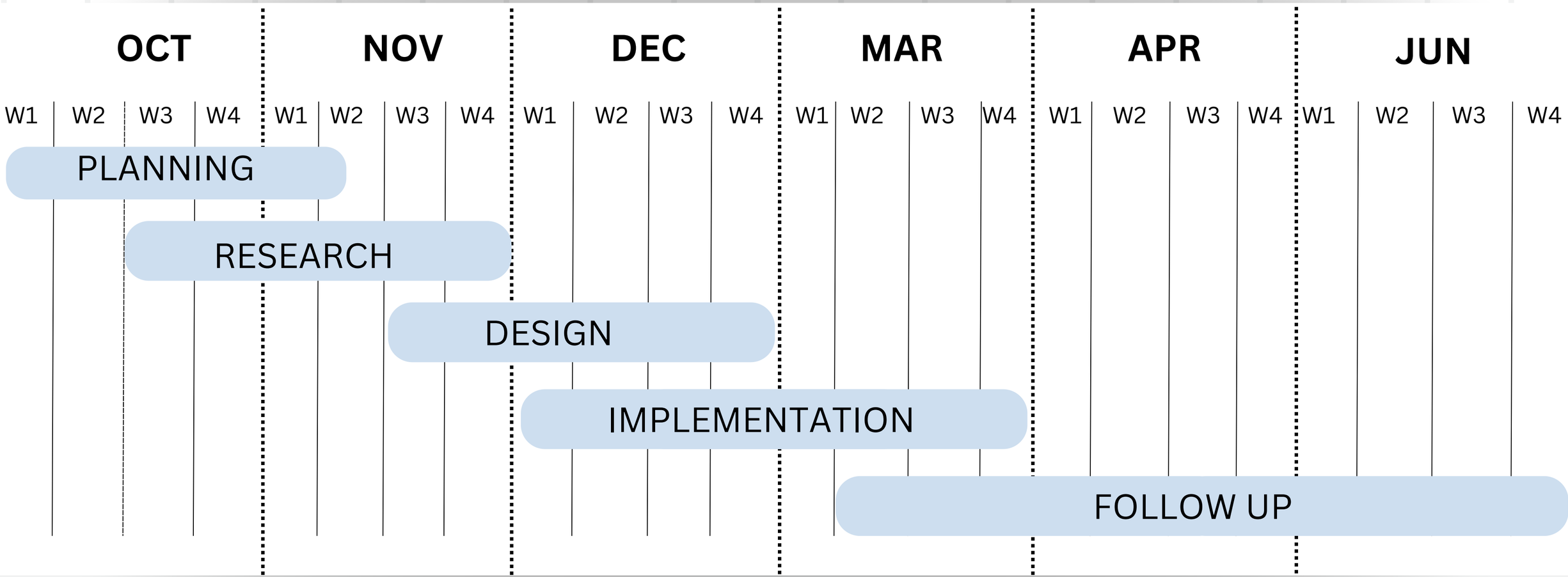

Grade-ease_final - Presentation
127.0.0.1:8000/admin-dashboard
127.0.0.1:8000/admin-dashboard
Apps
Gmail
Maps
YouTube
github
Home
ch
WhatsApp
figma
Instagram
Vidya Internet Service
pexels
Chess
metacrafters
django
Hug
All Bookmarks
GRADE EASE
Logout
Admin
Dashboard
Teacher Section
Student Section
Manage Courses
Manage Questions
Registered Students
1
Total Teachers
1
Total Courses
3
Available Questions
5
Copyright © Grade-ease - 2024


Grade-ease_final - Presentation
127.0.0.1:8000/student/student
127.0.0.1:8000/student/student-dashboard
Apps
Gmail
Maps
YouTube
github
Home
ch
WhatsApp
figma
Instagram
Vidya Internet Service
pexels
Chess
metacrafters
django
Hug
All Bookmarks
GRADE EASE
Logout
akash
(Student)
Dashboard
Examination
My Marks
Available Exams
3
Total Tests
5
Copyright © Grade-ease - 2024

RESOURCE AND TASKS

Sanjo Simon	• Bert model training • Backend
Sandra Mariya Benny	• Handwritten text recognistion
Razin Ahmad Abbas	• Frontend
Vishnu PR	• Frontend • Database

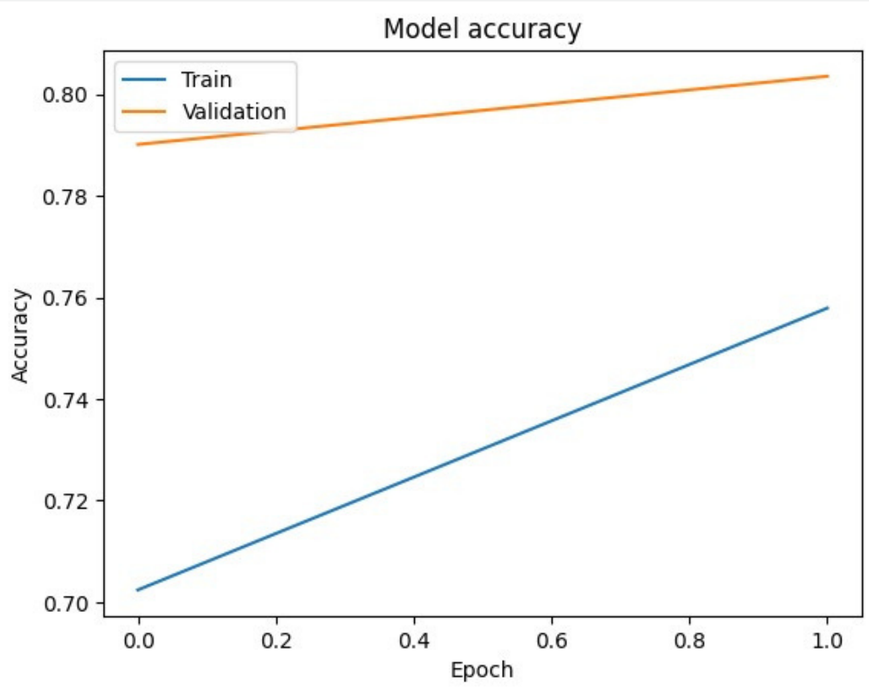
PLANNING

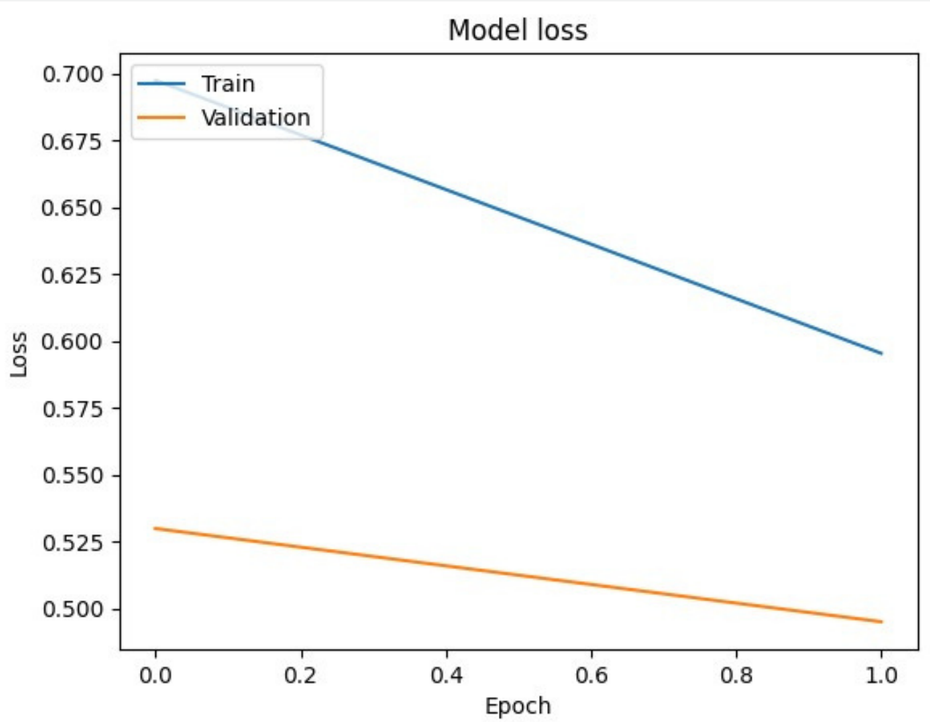

OCT
NOV
DEC
MAR
APR
JUN
W1 W2 W3 W4 W1 W2 W3 W4 W1 W2 W3 W4 W1 W2 W3 W4 W1 W2 W3 W4 W1 W2 W3 W4
PLANNING
RESEARCH
DESIGN
IMPLEMENTATION
FOLLOW UP

RESULT ANALYSIS

Performence metrics of the model

- **Accuracy Metrics: 0.80**
- **Model Loss: 0.70**

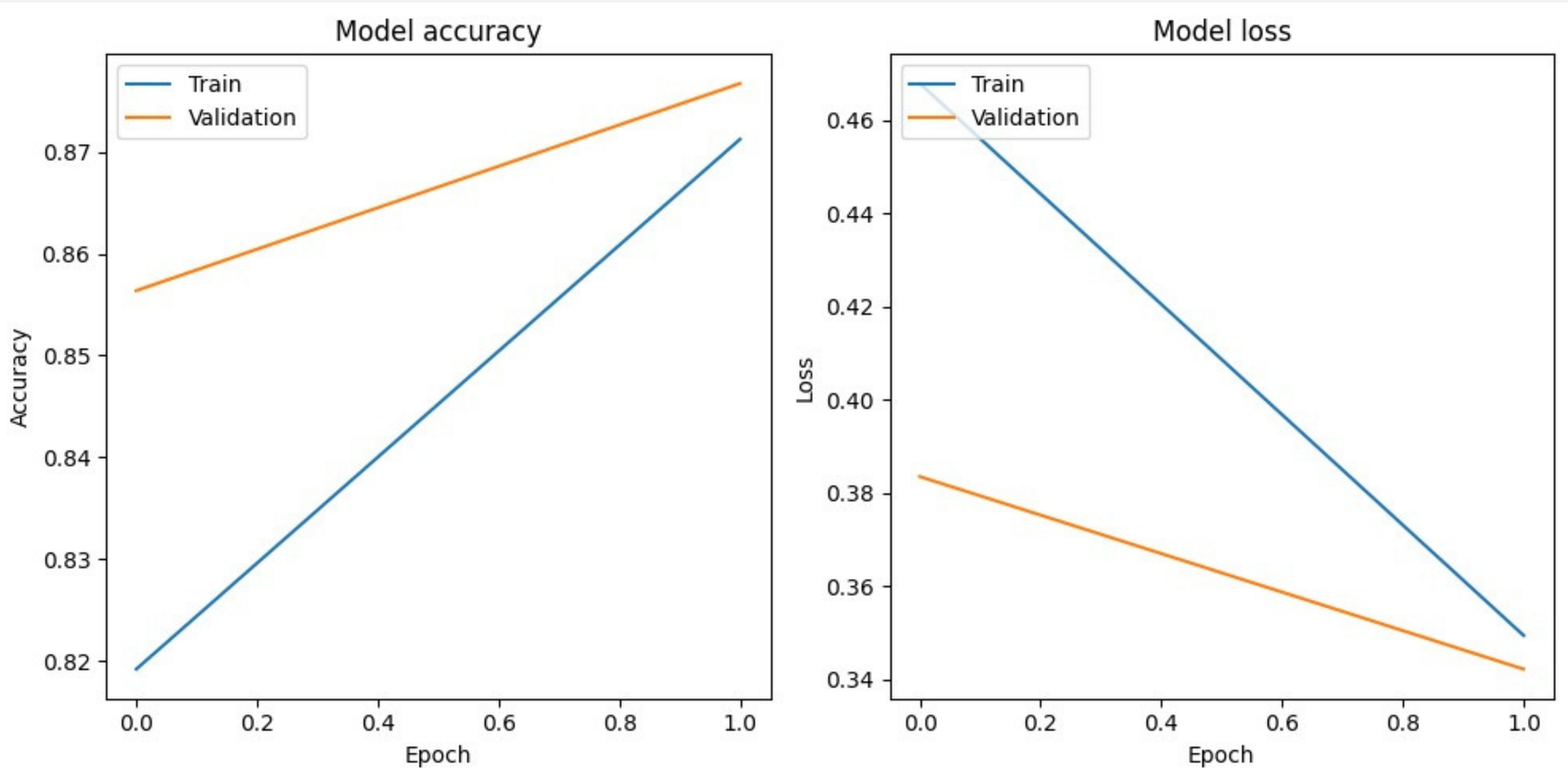




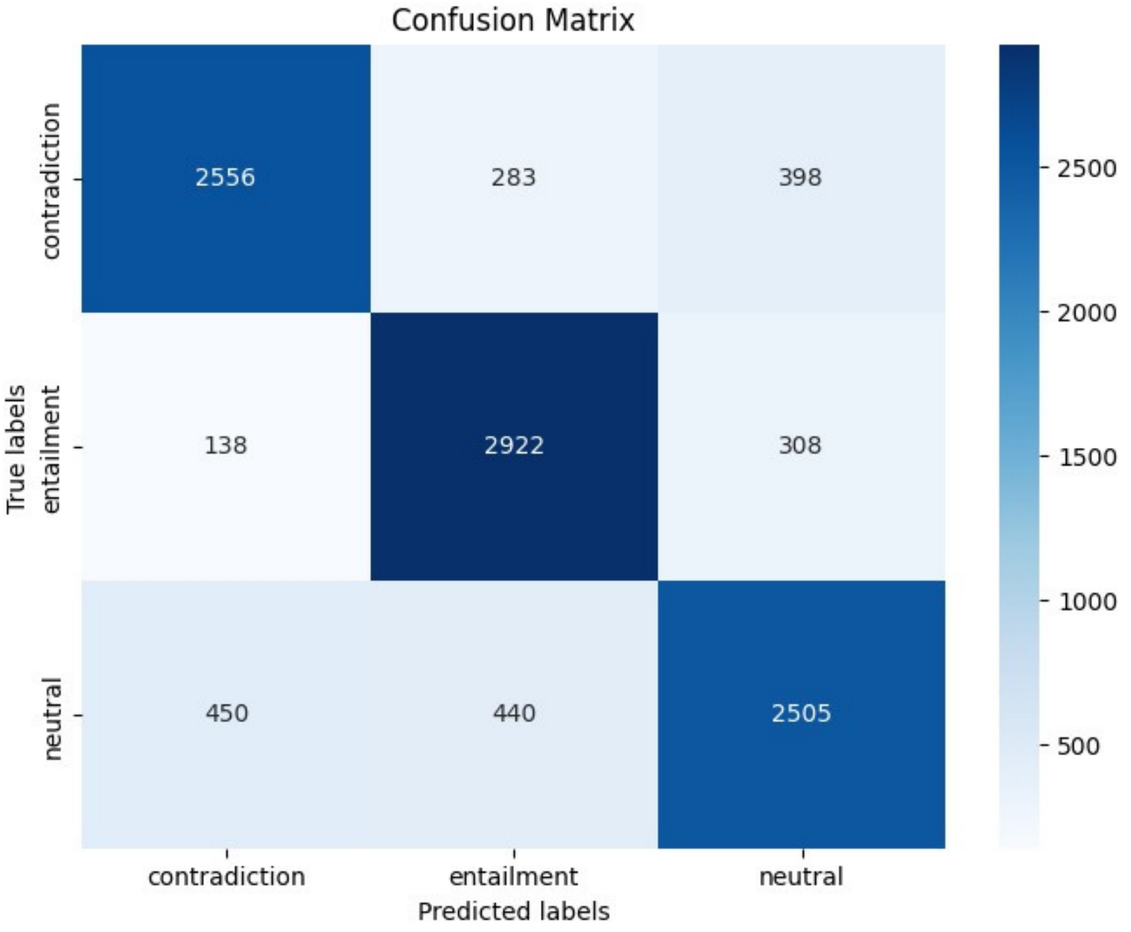
RESULT ANALYSIS

Performence metrics of the model after fine-tunning

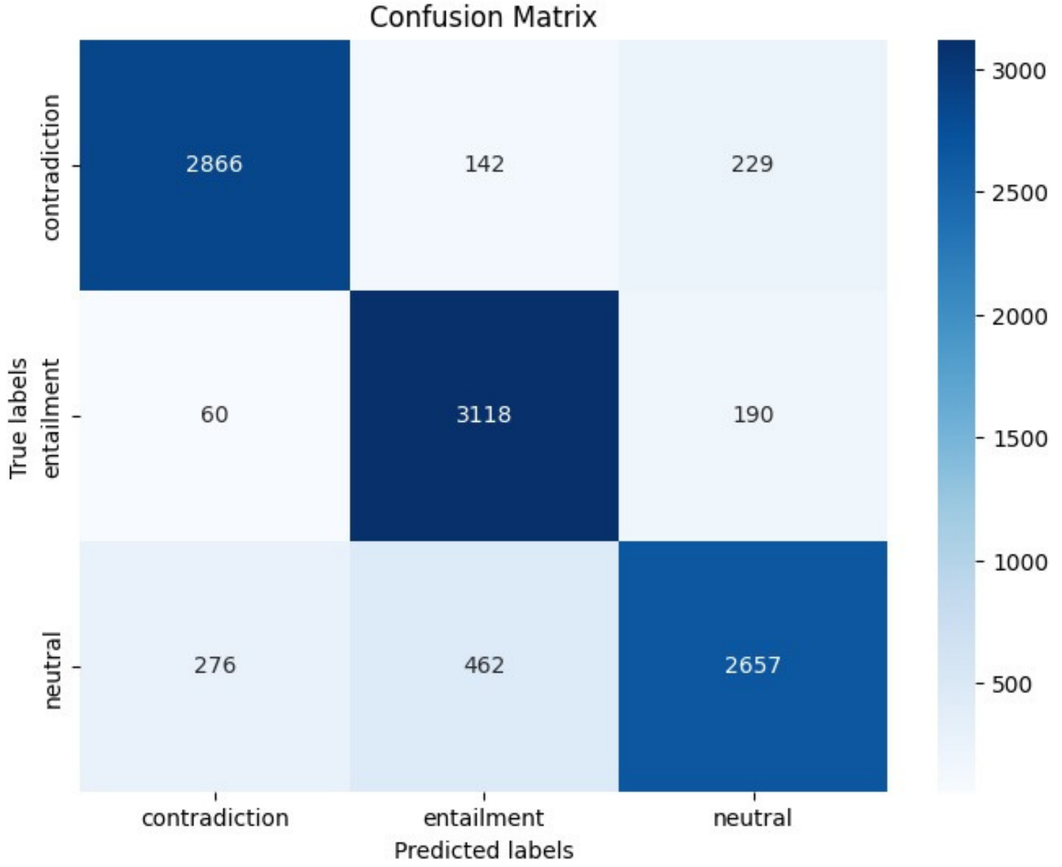
- **Accuracy Metrics: 0.87**
- **Model Loss: 0.46**



Confusion Metrics

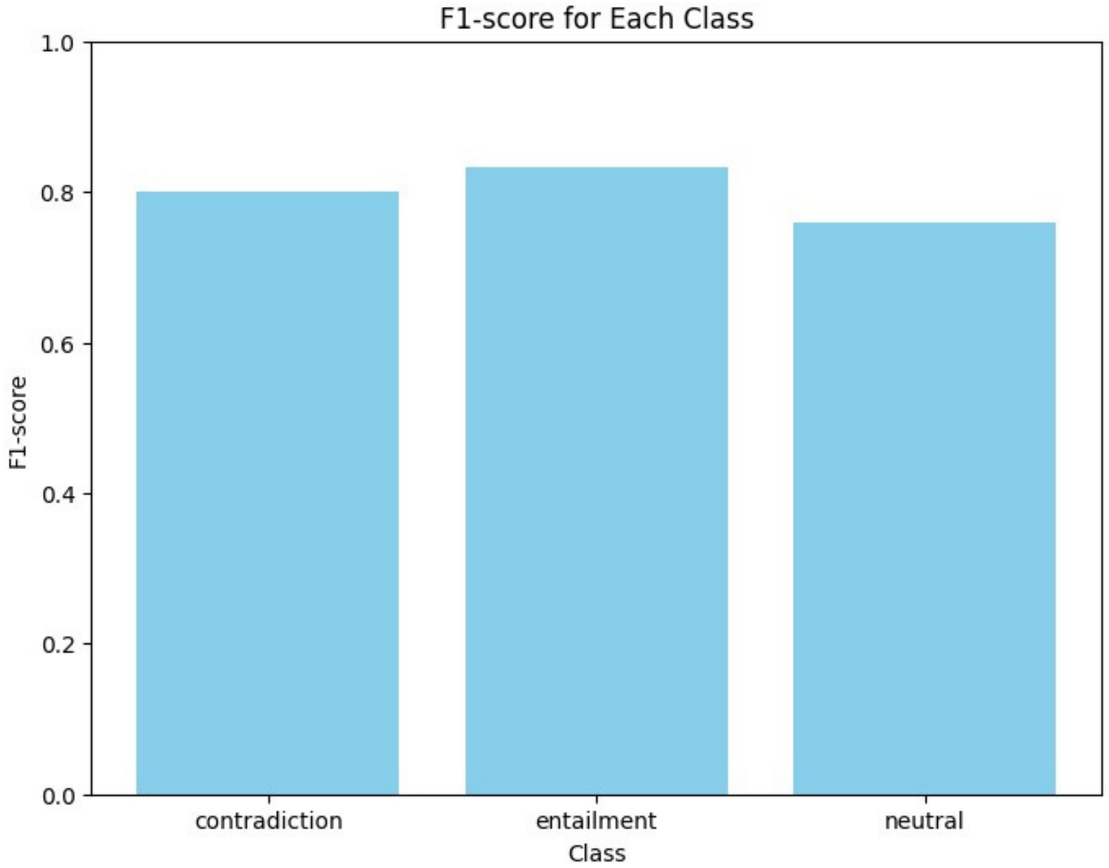


Before fine tunning

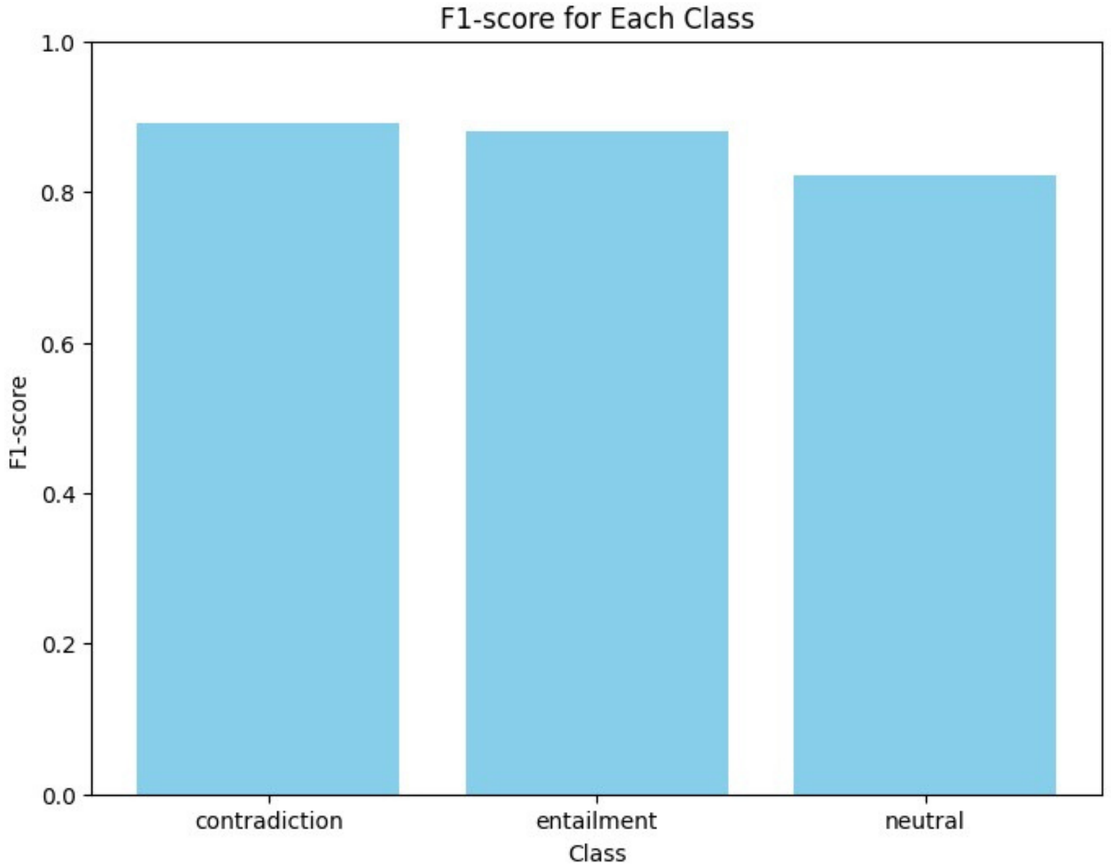


After fine tunning

F1-score



Before fine tunning



After fine tunning

CONCLUSION

- The majority of the exam types is online multiple-choice questions, are available to grade them. On the contrary, end semester exams are generally subjective, so there is a great demand for a system that can automatically score these brief answers without spending too much time.

- Thus, our system provides a platform for all educational institutes and assigns accurate grades to the student's subjective answers. It aims to score student responses using the parameter like similarity index and words matching.

- The model solution provided by the teacher will be compared to the solution submitted by the student, and suitable grades will be assigned based on the above parameter. Such methods can be useful in many online evaluation platforms and college portals since they save time.

REFERENCES

[1] J. Wang and Y. Dong, ‘‘Measurement of text similarity: A survey,’’ Information, vol. 11, no. 9, p 421, Aug. 2020.

[2] M. Han, X. Zhang, X. Yuan, J. Jiang, W. Yun, and C. Gao, ‘‘A survey on the techniques, applications, and performance of short text semantic similarity,’’ Concurrency Comput., Pract. Exper., vol. 33, no. 5, Mar. 2021.

[3] M. S. M. Patil and M. S. Patil, ‘‘Evaluating Student descriptive answers using natural language processing,’’ Int. J. Eng. Res. Technol., vol. 3, no. 3, pp. 1716–1718, 2014.

[4] P. Patil, S. Patil, V. Miniyar, and A. Bandal, ‘‘Subjective answer evaluation using machine learning,’’ Int. J. Pure Appl. Math., vol. 118, no. 24, pp. 1–13, 2018.

[5] J. Muangprathub, S. Kajornkasirat, and A. Wanichsombat, ‘‘Document plagiarism detection using a new concept similarity in formal concept analysis,’’ J. Appl. Math., vol. 2021, pp. 1–10, Mar. 2021.

[6] "Subjective Answers Evaluation Using Machine Learning and Natural Language Processing 2021 " by Muhammad Farrukh Bashir, Hamza Arshad, Abdul Rehman Javed Natalia Kryvinska And Shahab S. Band.

[7] “Tool for Evaluating Subjective Answers using AI 2021 (TESA)” by Shreya Singh, Prof. Uday Rote, Omkar Manchekar, Prof. Sheetal Jagtap, Ambar Patwardha, Dr. Hariram Chavan.

THANK YOU

Presentation by Group 17