RSET
RAJAGIRI SCHOOL OF
ENGINEERING & TECHNOLOGY
(AUTONOMOUS)

Project Report On

Interactive GD platform

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

Bachelor of Technology

in

Computer Science and Engineering

By

Sanghamitra Menon (U2103189)
Sanoy Boby (U2103190)
Susan Mathew (U2103204)
Vijay K V (U2103213)

Under the guidance of

Ms. Jomina John

Department of Computer Science Engineering
Rajagiri School of Engineering & Technology (Autonomous)
(Parent University: APJ Abdul Kalam Technological University)
Rajagiri Valley, Kakkanad, Kochi, 682039
APRIL 2025

CERTIFICATE

This is to certify that the project report entitled ***"Interactive GD platform"*** *is a bonafide record of the work done by* ***Sanghamitra Menon(U2103189)***, ***Sanoy Boby(U2103190)***,***Susan Mathew(U2103204)***, ***Vijay K V(U2103213)***, *submitted to the Rajagiri School of Engineering & Technology (RSET) (Autonomous) in partial fulfillment of the requirements for the award of the degree of Bachelor of Technology (B. Tech.) in Computer Science and Engineering during the academic year 2024-2025*

Ms.Jomina John
Assistant Professor
Dept. of CSE
RSET

Ms.Sangeetha Jamal
Assistant Professor
Dept. of CSE
RSET

Dr.Preetha K.G
HOD
Dept. of CSE
RSET

ACKNOWLEDGMENT

We wish to express our sincere gratitude towards **Rev.Dr.Jaison Paul Mulerikkal CMI**, Principal of RSET, and **Dr.Preetha K G**, Head of the Department of Computer Science and Engineering for providing us with the opportunity to undertake our project, "Interactive GD platform".

We are highly indebted to our project coordinator, **Ms.Sangeetha Jamal**, Assistant Professor, Dept. of CSE, for her valuable support.

It is indeed our pleasure and a moment of satisfaction for us to express our sincere gratitude to our project guide **Ms.Jomina John** for her patience and all the priceless advice and wisdom she has shared with us.

Last but not the least, We would like to express our sincere gratitude towards all other teachers and friends for their continuous support and constructive ideas.

Sanghamitra Menon

Sanoy Boby

Susan Mathew

Vijay K V

Abstract

This project presents an innovative platform for practicing group discussions. This platform will allow users to develop their communication skills by providing an environment to simulate real, group discussions. Users will practice alongside virtual participants, pre-trained language models, to have discussion experience. The platform will allow verbal discussions in a seamless manner as it will be utilizing speech-to-text (STT) and text-to-speech (TTS) processing to capture speech and provide interaction.

To have an even more engaging experience, users will be represented by an avatar-based representation that utilizes animated lip overlays of the trained language model to create a more realistic conversational experience. A unique evaluation system will provide users the opportunity to receive feedback on both relevance of content and also evaluations of facial engagement, eye contact, and general engagement over time. This evaluation system will provide feedback that users can utilize to improve their speaking and presentation skills.

Through an integrated, interactive language model, speech processing capabilities of STT and TTS, and user evaluation methods, the platform is positioned to provide an immersive context for authentic practice of group discussion skills. The interaction with the language model will allow users to practice not only presenting ideas, but responding from opposing viewpoints, and engage users in authentic speech and an opportunity for enhanced communication skills. The evaluation system methods will provide users with useful feedback on content relevance and engagement within their discussion, so they are fully aware of all aspects they should improve upon.

Contents

List of Abbreviations

- **PCA** - Principal Component Analysis
- **SVM** - Support Vector Machines
- **CNN** - Convolutional Neural Networks
- **LLM** - Large Language Models
- **NLP** - Natural Language Processing
- **RBM** - Restricted Boltzmann Machine
- **TF-IDF** - Term Frequency-Inverse Document Frequency
- **BLEU** - Bilingual Evaluation Understudy
- **WER** - Word Error Rate
- **STT** - Speech-to-Text
- **TTS** - Text-to-Speech

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Background

GDs are a vital tool in recruitment and selection of many industries in highly competitive jobs. Employers use GDs to evaluate the candidates beyond technical qualifications. It is an essential tool for getting individuals who portray good communication skills, teamwork skills, leadership and critical thinking skills. During GDs, the recruiter comes to know the interpersonal skills of the candidate and whether he can work in a group or not. Unlike written tests or individual interviews, GDs require real-time interaction with others, making it difficult to practice and simulate without others. Few candidates have a chance to train in realistic scenarios of GD discussions and, consequently, are less prepared for the subtle demands placed on them in these evaluations. In addition, traditional methods for preparing for GDs, like studying common GD topics or watching sample discussions, tend to be unsatisfactory. Without detailed feedback, candidates may find it challenging to improve on areas such as confidence, fluency, listening, and non-verbal communication, all of which are highly relevant in a GD setting.

1.2 Problem Definition

This project aims at developing an AI-driven group discussion practice platform. This platform shall provide the user with a realistic and interactive group discussion environment that simulates real-life discussions. Utilizing the platform's evaluation system, users receive feedback on key factors such as content relevance and facial evaluation, including aspects like eye closure and maintaining proper eye contact, helping them improve their engagement and effectiveness in group discussions. This new approach does not only make GD preparation accessible but also gives feedback's to track progress.

1.3 Scope and Motivation

The motivation for the project lies in the growing importance of group discussions in recruitment processes and the lack of resources candidates have for preparation. Most candidates find it hard to practice their GD skills as they do not have platforms.

Further enhancements on this platform would include high-performance AI-based realism where participants act like real virtual characters in multiple personalities and communicative styles that better reflect an authentic GD experience. The presence of emotion recognition, and virtual/augmented reality-based environments will also help provide much-needed non-verbal cue feedback.

1.4 Objectives

- Create an interactive group discussion platform where users interact with LLMs in an organized way.
- Incorporate speech processing and evaluation functions for determining content relevance, facial expressions, and engagement.
- Give feedback on user performance, assisting in enhancing discussion skills through analysis of non-verbal and verbal engagement.

1.5 Challenges

Real-time facial evaluation was a challenge, including eye contact and facial expressions under different situations; e.g., lighting, distance from the camera, and different angles. Likewise, facilitating speech-to-text and processing text while processing across diverse speaking patterns also required optimum provisions. Ensuring smooth interaction between the user and Language Models (LM) while explicitly managing turn-taking and other pauses necessitated substantial revisions.

1.6 Assumptions

1. The platform assumes that the user has a microphone and camera of sufficient quality to ensure clear audio and video input.

2. It also assumes that the user is able to maintain optimal conditions (e.g., proper lighting, camera angles) for best performance, allowing the system to accurately interpret body language, facial expressions and gestures.

1.7 Societal / Industrial Relevance

This project is of great importance to society and industry at large, especially in relation to the changing landscape of the employment marketplace, while enhancing soft skills such as teamwork, communication, and problem-solving. As part of a competitive hiring process, group discussions (GDs) are a widely used method by employers to evaluate candidates beyond a technical skill assessment. The project's AI-driven platform will enable job seekers to practice GDs in realistic scenarios and in a new technology-enhanced way of developing those transferable skills.

1.8 Organization of the Report

This report is structured to provide a comprehensive understanding of the project. Chapter 1 introduces the project by discussing the significance of group discussions, the problem statement, objectives, challenges faced, assumptions, and its societal and industrial relevance. Chapter 2 presents a literature survey, reviewing existing research on interactive group discussion platforms, including NLP-based analysis, Large Language Models (LLMs), and evaluation techniques, while also identifying gaps in current GD preparation methods. Chapter 3 outlines the system requirements, detailing the functional, software, and hardware specifications necessary for implementation, along with any cost considerations. Chapter 4 explains the system architecture, describing the working components, technologies used, and interactions within the platform. It also provides an overview of the project's workflow and timeline. Chapter 5 delves into system implementation, covering LLM integration, speech processing, evaluation mechanisms, and database management, while also discussing datasets and implementation strategies. Chapter 6 presents the results, showcasing evaluation metrics, screenshots of the platform, and a comparative analysis with existing models. Finally, Chapter 7 concludes the report by summarizing key findings, highlighting achievements, and discussing future improvements.This structured organization ensures clarity and coherence in presenting the project's development

and outcomes.

1.9 Summary

This chapter discusses the importance of group discussions in the recruitment and selection process. It improvises the candidate's skills to prepare for interviews. Traditional methods of GD preparation are usually not very effective in providing a personalized, realistic, and adaptive practice environment. This platform intends to fill that gap by offering a space for candidates to enhance their GD skills, providing them with valuable feedback. By simulating real-life scenarios, the platform with highly advanced AI capacities promises to completely change GD preparation, to become more accessible, effective, and tailored to specific needs. Its further evolution brings the possibilities for even higher level advancements in increased realism, connection to VR/AR, and increased personalization - all those should prepare the candidate for the challenges of today's recruitment.

Chapter 2

Literature Survey

2.1 Introduction

The above literature review clearly demonstrates that developing an advanced interactive group discussion platform will require several key foundational technologies. The most important component is Large Language Models, which will enable real-time and contextualized responses, thus resulting in rich and engaging conversations. The key focus will be on the NLP-based evaluation metrics that measure user inputs in terms of clarity, relevance, and engagement. Real-time feedback will drive not only the quality of conversations but the development of the participants. Also, in the case of deepfake, since each LLM has its own voice and facial identity, the system is further improved to provide a more realistic and pleasant experience for the users. The survey also entails the turn-taking mechanism in which the Round Robin algorithm is applied in the management of turn-taking to ensure that all the participants have an equal right to speak. Although this approach provides equal chances for participation, more so, there are features that help in reducing repetition and enhance the flow of the conversation. All these elements provide a basis for developing the next generation platform that integrates AI analysis, multi-modal interaction, and user-centered design

2.2 Deepfake Detection using Deep Learning

Deepfake detection has emerged as an essential task in the current trend of increasing cases of fake media. Deepfake detection is a task that is effectively solved by deep learning techniques due to their ability to detect differences or anomalies in the data. Deepfake detection entails an ensemble methods for building the detection model, which is effective because they entail running multiple models together to create improvements. The main types of ensemble strategies are:

- Bagging: For example, in Random Forests, numerous models are run utilizing different data sets, and the predictions of all the models are averaged to prevent errors and over-training of the data.

- Forms of Boosting: There is AdaBoost and Gradient Boosting which are both methods in boosting that are trained in a serial fashion with the first model being the base and each model being an attempt to build off of the first. The intent is to make sure the final combined model is as accurate as possible.

- Stacking: In this approach, multiple models generate predictions, and a different model aggregates these predictions with the aim of improving the performance of the individual models.

Based on this discussion, it can be said that ensemble methods can be viable for the purpose of improving the overall model performance (i.e., performance for deepfake detection), since mistakes that some models could have made, or overtraining that can occur in models can be averted.

2.2.1 Preprocessing

Data preprocessing is critical to improving deep learning model performance, especially in deepfake detection. Preprocessed steps include:

- Grayscale Conversion: In this way, the images are converted to the grayscale which will assist the model to learn only the lines contours of the face and texture of the skin rather than dealing with rich colour information. This means lower computational cost and simpler data learned by the model.

- Data Augmentation: Some of the techniques we employed were scaling, rotate, flipping and zooming which truly enhanced the veracity of the sets. It helps avoid the issue of overfitting and improves the performance of the models when they encounter unstandardized data sets previously. To illustrate, we first take the images and normalize them by dividing by the 1/255 factor, and then we rotate them within 20 degrees of orientation to allow variation.

- Original Dataset: When using the original datasets, it is ensured that the detailed color and texture information that some models might need for localizing features is not altered.

Models respond differently to these preprocessing strategies:

- VGG19: This architecture works best with the non-compressed, high-resolution images as it is capable of identifying fine details.
- DenseNet: Such architecture is known to perform well with grayscale as well as augmented data where it effectively uses the features extracted from various parts of the network especially when dealing with complex data.
- Custom CNNs: The custom CNN models work best with the augmented data since the large amount of data helps the models to provide more accurate results.[1]

2.2.2 Binary Classification

The main part of deepfake detection is figuring out whether images are real or fake. It all starts with a balanced set of images, both real and fake, which go through some initial cleaning or preprocessing. After that, these images are passed through different models to find key features.

- VGG19: VGG19 is known for its ability to capture detailed features. It works best with high-resolution original images.
- DenseNet: With its densely connected layers, DenseNet is perfect for grayscale and augmented data because it reuses features across multiple layers.
- Custom CNNs: These networks are designed to perform well with the specific dataset at hand, making them adaptable to different tasks.

The models produce their extracted features, which are then put into a common feature pool. In order to reduce the dimensionality of these features and improve efficiency, we apply Principal Component Analysis (PCA), thus retaining only the most significant portions of the data. Afterward, a Support Vector Machine (SVM) is used to distinguish real images from fake images. SVM finds a separation between the two classes by constructing the best possible "boundary" or hyperplane.

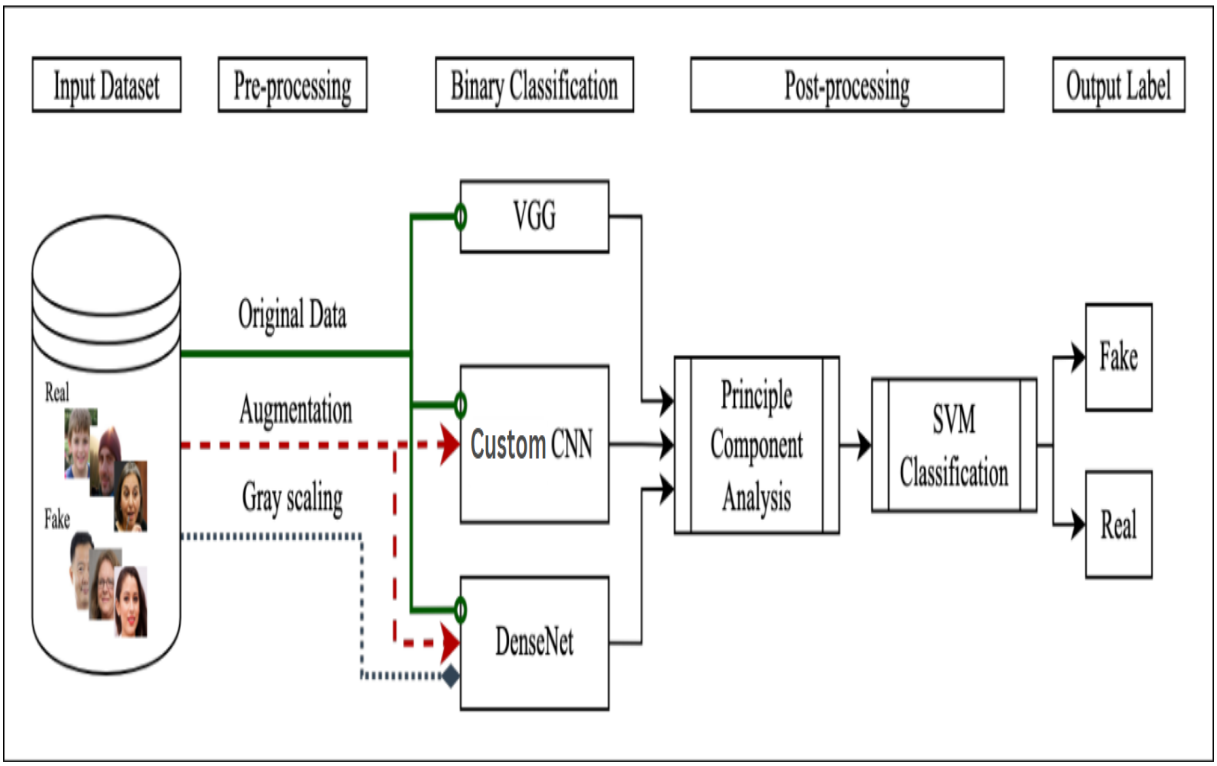


Figure 2.1: Ensemble Method Architecture Diagram

2.2.3 Postprocessing

PCA effectively minimizes the high-dimension data in the postprocessing stage and prices the features of interest such as the directions of maximum variance and minimizing irrelevant noise. This allows the SVM to concentrate on a smaller set of flipped in the PCA, and both faster computational execution and improved accuracy of classification. By training it on SVM gives the score last score. The lower flipped set creates maximum accuracy. Deepfake has turned and tied together these two methods and PCA and SVM and this is the post processing.[1]

2.2.4 Result

The performance of several models in detecting false photos was assessed using their accuracy in differentiating between actual and fake photographs. Here's how they stack up.

- VGG19:

  On its own, it received a 95 percent. With PCA-SVM, it increased to 99 percent.

Takeaway: VGG19 works best with high-resolution images, and employing PCA-SVM can help fine-tune the features and improve categorization.

- DenseNet:

  Alone, it received a 94 percent. It improved to 97 percent with the addition of PCA-SVM. Takeaway: DenseNet's performance is very dependent on how the data is processed. It performs best with original, unedited photos and gains greatly from data augmentation. However, while PCA-SVM enhances accuracy, it is less effective when images are transformed to greyscale.

- Custom CNNs:

  They achieved around 89 percent independently. Through PCA-SVM, they attained a maximum of 97 percent. The significant point is Custom CNNs benefit the most from PCA-SVM. The limited feature space relies primarily on the important features of the data, which is exactly the scenarios where these models are perfectly suited for deepfake detection. Overall, VGG19 and customized CNNs perform the best when utilized with the original data, while DenseNet's output is mainly influenced by the preprocessing, augmentation, and training process of the input.

2.3 Evaluating Real-Time Idea Flexibility for collaborative Creativity using NLP

Collaborative creativity refers to the capacity to work collaboratively to create and revise ideas to facilitate creation, problem solving, and brainstorming activities. To do a task well in Group Collaboration participants need to operate in a way that can pivot and adapt a process were new information is generated and delivered to them - this constant experimentation of measuring creativity and flexibliity in the group context has many major challenges such as complexity of the group composition and volume of participants interaction. NLP approaches are inexpensive in nature and provide capacity to scale and higher performance in measuring the both the idea expansion process and what and what participants do on the provided input. Conversation analysis can gather insights from reproducing NLP-based techniques such as sentiment analysis, topic modeling, text summarization, and interpret those insights that also provide quantitative metrics in

thinking and process of creativity that are descriptive and complex.This paper discusses these techniques and their potential use in brainstorming, innovative and problem-solving processes.

2.3.1 Understanding Collaborative Creativity

This process involves generating, modifying and evolving ideas with respect to group interaction. The most important aspect of a productive collaboration will involve being open to new ideas, challenging held beliefs and considering multiple points of view. In this context of collaboration, which requires a great deal of flexibility for everyone involved, adapting to changes and being open to contributing to the growth of ideas are key.Because things will be complex when groups act, traditional approaches may miss some of these nuances. NLP offers a scalable framework for monitoring and evaluating conversations generally, which can reflect idea flexibility, adaptability from participants and group coherence; therefore, these also give us insights in relation to the collaborative process for creativity which involve both demonstrably above average areas of engagement and also in terms of possible areas of improvement.[2]

2.3.2 The Role of Natural Language Processing in Collaborative Creativity

The NLP subfields enable the evaluation of the questioning during a group session and then the questioning related data from a group session to be used to assess an individual's creativity. They are as follows.

- Sentiment Analysis: The analysis of the emotional tone of the participant is useful to detect any shifts in the group's interactions particularly their level of engagement or frustration during the inquiry. This might also indicate degrees of agreement, conflict, or disengagement.

- Topic modeling: It also consists of attending to the main ideas and concerns of the talk to help keep participants focused on some shared issues.

- Summary of text texts: This helps in condensing where a particular question concerning a particular discussion can be answered.

- Entity Recognition: This identifies relevant key terms and concepts and their role as primary building blocks of what is built or in what it may deviate from.

All of these tools assist in enhancing the understanding of a person regarding multi task questioning during a group activity and with the help of these tools, a person can raise or switch perspectives.

2.3.3 Natural Language Processing and Its Impact on Collaborative Creativity

Real-time idea evaluation will let the team check the process of what goes through in their planning sessions uninterrupted. This can be aided by thousands of applications through NLP and many more to come:

- Idea Change Observation: NLP allows for exact measurement on the introduction, change or deletion of an idea. This clearly illustrates the interactive creativity of the group.
- Re-Conceptualization Measurement: NLP observes the group's capacity to innovate and adapt by measuring the degree its members re-frame or build upon ideas in response to new information.[2]
- Relation to The Theme: Contributions are rated appropriately based on the level of effectiveness established in the discussed theme, and as such, the engagement remains on the drifts of conversations.

In numbers, NLP can account for the changing of ideas and the intertwinement and development of ideas over the brainstorming process. This information is essential because it creates an understanding of the responsiveness of the team and the overall level of creativity generated.

2.3.4 Result

Collaborative creativity implemented in concert with NLP does offer some understanding of the evolution of ideas and provides a measure of flexibility, as indicated by many examples. NLP is a reference to the moments when the conceptual categories of ideas shift, or ideas react to some specified criterion in which we provide numerical estimations of the degree of that reaction.To maintain the applications of text to voice in collaborative meetings, strategies such as text summarization and topic modelling are instrumental

in maintaining the directed and focus of the group, during an exchange of conversation.Sentiment analysis can be useful in showing the level of dynamics of the emotions of parties engaged in discussion, for example it shows how active participants take turns speaking or engaged in discussions.Therefore it shows that NLP not only serves to determine the degree of flexibility of ideas, but it meets the needs of improving group communication and cooperation, which is pertinent in any creative context.

2.4 Large Language Models and Rule-Based Approaches in Domain-Specific Communication

AI has risen to be in charge of almost all fields such as finances, healthcare and customer service. Among the AI branches, AI language processing attempts to assist computers in learning how humans use and experience language. Some of them include virtual assistants, customer service representatives and chatbots. There exist two types of language processing; these include LLM and RBM. The most famous LLMs are GPT-3 and BERT, which depend on methods of figuring out the con- relation among words and phrases in a sentence. This equips them with skills such as content generation to respond to customer questions just in time. RBMs, on the other hand, are rule-based. They operate using logical pre-established assumptions. It is not quite flexible like an LLM; however, this can be harnessed for certain types of tasks for a consistent output. Therefore, though the LLM is open-ended and creative in thought, so also cannot it be said about the RBM that is very successful for just straightforward tasks provided that rules are stable. Alone or combined, instrumental to the successful execution of the AI on their day-to-day tasks rest these models. The main goal of this seminar is the difference between LLMs and RBMs with respect to their use in domain-specific scenarios. The merits, demerits, and features of LLMs and RBMs are considered for use in applications such as customer interaction and medical diagnostics. Both methodologies will be explored in the seminar, with an exposition on how one can choose an adequate model to solve a number of NLP tasks, hence enabling developers and organizations to enhance efficiency and productivity.

2.4.1 Strategies

Model training is the process of teaching a model to recognize patterns and make predictions. LLMs and RBMs have different training schemes as each corresponds to the architecture of it's model.

- Training Process for LLMs

  The LLM training process adopts a transfer learning methodology. LLMs utilize a pre-trained model following training on a massive dataset of general language patterns. Subsequently, these models go through additional fine-tuning with some degree of domain-specific data for narrowly scoped tasks. Significant parameters for training LLMs are: -Epochs-the number of passes through the dataset. Each LLM should train for multiple epochs in an effort to enhance the learn-ing model-Learning rate-the adjustment rate of the model parameters. The adjusted learning rate will avoid overfitting and underfitting. LLMs are tested after fine-tuning via measures such as a loss and accuracy in effort to optimize in the task for which the model is designed.

- Training Process for RBMs

  Rule-Based Models employ an alternative strategy, founded on pre-established rules that govern the manner in which input data are handled. They are trained on fewer data, usually including less complex NLP processes like:Tokenization-breaking texts into words or phrases. Stemming lowers words to their base form. Stop word removal assists in the elimination of insignificant words for the purpose of keyword finding.

- Retrieval-Based Techniques in RBMs

  In RBMs, methods like TF-IDF (Term Frequency-Inverse Document Frequency) can be applied in information retrieval activities for assessing term relevance, or Cosine Similarity when used to quantify the similarity between terms and the document content[3]

2.4.2 Model Evaluation and Analysis

Model evaluation is an important phase in performance analysis in-depth to enhance the performance. The performance evaluation metrics are:

- Direct Selection: The model reply chosen which is the most relevant and accurate one is usually selected by users.
- Modification: The user adjusts the model's reply so that it becomes more relevant.
- Custom Responses: user-typed customized responses when pre-defined answers are not sufficient.

Evaluation is also conducted through metrics automated, for instance, BLEU, Word Error Rate (WER), Precision, and Recall to measure the model accuracy with regard to relevance in actual settings.[3]

2.4.3 Result

Training data to contrast LLMs and RBMs revealed another set of benefits to them. LLMs appear to handle better needs for contextual awareness and subtle language generation, utilizing their application in changing contexts such as customer interaction and content creation. Conversely, RBMs can tackle activities of a well-defined structure much more effectively, providing a clearer and quicker performance in set situations, for example, answering frequently asked questions by customers. Metrics like BLEU scores and WER showed LLMs were having trouble encoding with more complex word sequences, whereas RBMs still dominate on more straightforward rule-based tasks. On human judgment, a naturally occurring LLM, naturally, was adaptable and malleable, whereas an RBM was stable and more direct across structured tasks.

Metrics like BLEU scores and WER indicated LLMs were struggling to encode with complex word sequences, while RBMs continue to score high in simpler rule-based tasks. On human evaluation, an LLM, of course, was versatile and adaptive, while an RBM was more stable and straightforward over structured tasks.

2.5 A Survey of Variants of Round Robin CPU Scheduling Algorithms

Round Robin Allocation is a commonly used scheduling technique to regulate turn-taking in various systems: human-computer interactions and multiparty conversations. The primary goal of it is to provide each participant (human or artificial intelligence) with an equal and round-robin chance to speak. The process is best in scenarios where equal

involvement is required because turn-taking is rotated. Simple to adopt, the Round Robin guarantees that the order of interaction has structure and balance, thus making it suitable for group discussions, interactive systems, and settings where fairness is at stake.

2.5.1 Fundamental Concept

The Round Robin Allocation model uses a very simple turn-taking scheme. Ev- ryone takes a turn to talk in a set order. Each speaker is given a specific amount of time to talk. When one participant's time is up, the software proceeds to the next participant. This goes on until all are summoned to speak. Some key features of round robin allocation include the following: All participants-human or AI are noted down, to emulate the nature of the conversational demeanor. Rotation step by step will enable such that the participants take turns. Each participant has a set duration for a turn. The discussion flows uninterrupted from the previous turn to the initial speaker around the turn, thus providing a smooth flow. It is an effective method for maintaining the discussion organized and for allowing everyone to be heard.[4]

2.5.2 Strategy

The Round Robin Allocation process has some essential steps to function properly:

- Initialization

Participants List/Queue: First, initialize and prepare a list of participants (say, N).

Pointer/Index: Employ a pointer or index to monitor who's talking. Initialize the index at 0 (the first individual).

Allocation of Time: Determine how long each individual gets to talk and give everyone the same amount of time.

- Turn Allocation

Awarding Turns: The system grants the existing participant a chance to talk. The time duration is enforced, and their response is noted down.

Tracking Turn Duration: The system observes how long each individual talks and waits until either the time expires or they are finished speaking.

Shifting to the Next Participant: After an individual is finished speaking, the system increments the index to shift to the next individual. When it reaches the last individual (N-1), it loops back to the first individual.

- End of Discussion

Condition to End the Loop: The loop runs until it is stopped, such as everything being discussed, a time-out, or conditions set by users.

Termination: When the loop terminates, the system stops, and discussion ends.[4]

Aspect	**Details**
Description	Allocates turns in a rotating sequence for balanced participation.
Time Complexity	$O(n \cdot b/q)$
Space Complexity	$O(n)$
Implementation	Maintains a queue and rotates turns, using timestamps to ensure fairness.
Benefits	Simple and fair, ensures equal participation.

Table 2.1: Round Robin Allocation

2.5.3 Result

The Round Robin Allocation approach in a dialogue or conversation system has some important advantages:

- Balanced Participation: All participants, whether human or machine, have an equal opportunity to talk in a well-organized and fair sequence. No one is excluded for an extended period, ensuring everyone has a balanced chance to contribute to the conversation.

- Simplicity and Efficiency: It is simple to implement and does not involve complex procedures. The system follows an uncomplicated and easy-to-understand rotation of turns, which facilitates smooth flow of discussion in group settings.

- Prevents Dominance: Since the turns are rotated, no single individual can dominate the discussion or monopolize all the talk time. This helps make every person's voice heard equally.

- Smooth Turn Transitions: The system does a good job tracking turns and handling transitions smoothly, keeping the conversation flowing with few interruptions or overlaps.

However, although it performs well in some contexts, Round Robin might be challenged in more interactive or dynamic discussions. Here, it may not be able to manage interruptions or impromptu exchanges very well. For instance, in a real-time customer service chat, quicker alternatives that can reply quickly to emergency questions or favor some participants would be more helpful.

2.6 Summary and Gaps Identified

2.6.1 Summary

The various approaches outlined here indicate how conversation systems and team work can be improved. NLP methods provide analysis of the real-time development of what occurs in group talk, providing a sense of the evolution of ideas over time. NLP methods utilize such activities as sentiment analysis, topic modeling, and summarization of text to offer insight. Large Language Models (LLMs) excel at tasks which require a high amount of context and flexible text generation, but they need a lot of computer power. Rule-Based Models (RBMs) are good for clear, well-structured tasks, but they don't cope well with more complicated or nuanced interactions. Round Robin Allocation ensures reasonable participation in group environments, though it doesn't cope with interruptions or shifting priorities. Deepfake technology combines realistic visuals and identities onto AI, making conversations more realistic. Every technique has its own strengths and weaknesses, and there are still problems to be solved, such as making systems work on a larger scale, dealing with real-time changes, and making efficient use of resources.The different methods discussed show ways to make conversation systems and group work

better. NLP techniques help analyze what's happening in group discussions in real-time, giving insights into how ideas change over time. These techniques use things like sentiment analysis, topic modeling, and text summarization to provide useful information. Large Language Models (LLMs) are great for tasks that need lots of context and adaptable text generation, but they require a lot of computer power. Rule-Based Models (RBMs) are good for clear, structured tasks, but they don't handle more complex or subtle interactions well. Round Robin Allocation helps ensure fair participation in group settings, though it doesn't do well with interruptions or changing priorities. Deepfake technology adds realistic visuals and identities to AI, making conversations more engaging. Each method has its own strengths and weaknesses, and there are still challenges to solve, like making systems work on a larger scale, handling real-time changes, and using resources efficiently

2.6.2 Gaps Identified

1. Real-Time Adaptation in Dynamic Environments:Round Robin has difficulty with interruptions and priority management. NLP finds it difficult to adjust to dynamic conversations and changing topics because of the lack of real-time context capture.

2. High Computational Demands and Scalability: LLMs and deepfakes demand extensive computational resources. Round Robin presupposes a basic queue but isn't suitable for situations with a high number of participants.

3. Lack of Feedback Mechanisms: Modern systems lack instantaneous feedback such as visual or audio feedback. Round Robin doesn't provide mechanisms to gauge participant engagement during turn-taking.

4. Ethical and Realism Concerns: Deepfakes pose ethical concerns and lacks emotional sensitivity.

   item Bias and Empathy Limitations: NLP and deepfake algorithms may miss biases and miss capturing empathy, resulting in unjustified conclusions.

   item Hybrid Approach Exploration: The book is promising in the possibility of hybrid models but doesn't specify how these could be incorporated with responsiveness (e.g., mixing Round Robin with priority-based scheduling).
   item Handling Uneven Turn Durations Round Robin is based on fixed turn lengths

but does not account for situations when participants need more or less time depending on input complexity or priority.

5. Limited Application in AI-Driven Customer Support: Round Robin's applicability in situations that involve prioritization (e.g., critical customer inquiries) is noted as an issue, with no definite solution.

6. Integration with Advanced AI Features: Opportunities to add Round Robin with technologies such as emotion recognition, topic modeling, or reinforcement learning are not yet established.

7. Insufficient Support for Multi-Modal Inputs: As of now, it does not support multi-modal inputs such as voice, gestures, or visual inputs.

8. Difficulty in Dealing with Overlaps and Interruptions: Natural dialogues usually contain interruptions and overlap speech, which mechanisms such as Round Robin cannot accommodate.

9. Limited Real-World Validation: Most approaches are not fully tested under diverse real-world scenarios and are primarily validated within controlled environments, which constrains their practical reliability

2.6.3 Comparison Table

Title	**Deepfake Detection using Deep Learning**
Advantages	- Detailed model evaluation. - High accuracy in detecting deepfakes.
Disadvantages	- High computational resources required. - Needs large labeled datasets for training.
Title	**Evaluating Real-Time Idea Flexibility for collaborative Creativity using NLP**
Advantages	- Real-time analysis of idea evolution. - Provides insights into group dynamics.
Disadvantages	- Complex interpretation of group behavior. - May struggle with unstructured data.
Title	**Large Language Models and Rule-Based Approaches in Domain-Specific Communication**
Advantages	- Clear distinction between LLMs for dynamic tasks and RBMs for structured ones. - Insightful for choosing the right model for specific tasks.
Disadvantages	- LLMs are resource-intensive. - RBMs lack flexibility.
Title	**A Survey of Variants of Round Robin CPU Scheduling Algorithms**
Advantages	- Fair and efficient turn-taking. - Simple implementation.
Disadvantages	- Struggles with dynamic or interruptive conversations. - Limited flexibility for complex interactions.

Table 2.2: Summary of approaches in Interactive GD Platform

Chapter 3

Requirements

3.1 Hardware and Software Requirements

1. **Webcam or Built-in Camera:** A camera with minimum 720p resolution is required for capturing real-time video of the user during the group discussion. The camera's resolution and quality play a significant role in the accuracy of facial expression and posture analysis.

2. **Microphone:** A functioning microphone is essential for speech recognition and real-time conversation processing.

3. **Processor:** Minimum Intel i3 or equivalent AMD processor (2.4 GHz dual-core) to handle real-time speech processing and language model interactions.

4. **RAM:** Minimum 8GB RAM for smooth operation of the Flask backend and language model inference.

5. **Operating System:** Windows 10/11 or macOS 10.15+ (64-bit) with latest browser support (Chrome/Firefox recommended).

6. **Python (3.8+):** Required for backend development using Flask and integration with language models.

7. **Flask Framework:** Lightweight Python web framework (version 2.0+) for backend API development.

8. **MongoDB:** NoSQL database (version 4.4+) for storing user profiles, discussion logs, and evaluation results.

9. **WebKit Speech Recognition API:** Browser-based API for converting user speech to text (Chrome/Firefox supported).

10. **Ollama (for LLaMA):** Local deployment of LLaMA model for offline response generation (if applicable).

11. **Google Gemini :** Cloud-based access to Gemini language model for participant responses.

12. **Qwen**: Integrated evaluation model for assessing discussion quality and providing comprehensive feedback metrics.

3.2 Functional Requirements

1. **User Authentication:** Secure email-based login system to authenticate users and track discussion history.

2. **Topic Selection:** Users must be able to select or change discussion topics before starting a session.

3. **Real-Time Speech Processing:** The system shall convert user speech to text in real-time using WebKit API.

4. **Automated Participant Responses:** Integration with LLaMA (via Ollama) and Gemini to generate context-aware responses during discussions.

5. **Turn Management:** The system shall enforce turn-taking rules, with AI participants responding if the user remains inactive for 10 seconds.

6. **Posture Analysis:** Periodic screenshots (every 2 minutes) to evaluate user posture during discussions.

7. **Speech Evaluation:** Analysis of speech fluency, repetition, and topic relevance post-session.

8. **Feedback Dashboard:** Detailed performance report after each discussion, including:

    - Speech metrics (speed, filler words)
    - Posture assessment
    - Content relevance score

9. **Discussion Logging:** All sessions must be stored in MongoDB with timestamps, transcripts, and evaluation data.

10. **Cross-Browser Compatibility:** The platform shall function on Chrome and Firefox with consistent performance.

Chapter 4

System Architecture

This chapter explains the system architecture, components, algorithms, data flow digram, tools, technologies. Identified for training detection models, as well as critical deliverables. The chapter recapitulates module breakdowns, job splits, and project timelines.

4.1 System Architecture of Interactive GD Platform

The architecture of the system is as shown below:

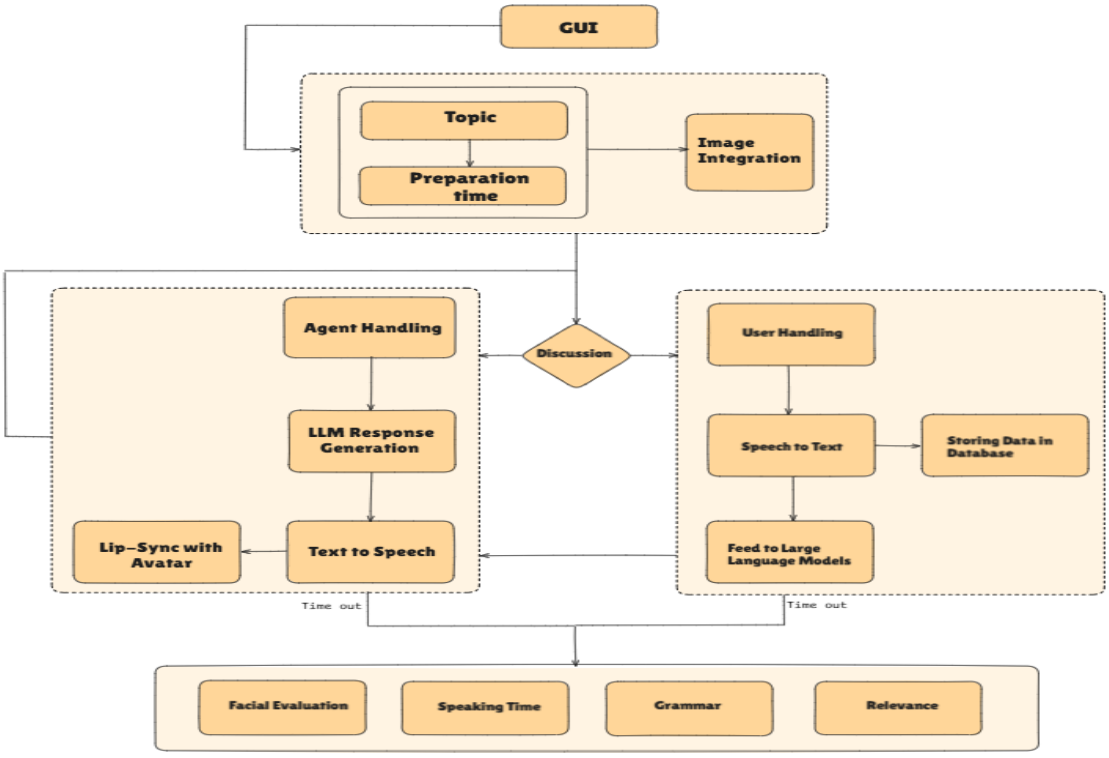


Figure 4.1: System Architecture

- Creating an environment

  The system creates an environment for group discussions by choosing a topic and simply giving it to the user. It shows avatars for the LLMs, using prefabricated images that are in sync with mouth movements to mimic discussion.

- User-Agent Interaction

  The discussion will start with either the user or an agent beginning the exchange first. User speech is first converted using Speech-to-Text (STT), and the model is fed the stored information to produce a reply. The response will be reviewed and converted to speech using Text-to-Speech (TTS) so that when the user hears the speech, it looks to be the face of the avatar that is validating.

- Evaluation

  The system evaluates user performance based on the co-relevance of the words to the topic maintain attention and facial stance engagement. The spoken content is examined to affirm relevance or discussion topic conformance, and periodic screenshots are will assess engagement practices like face to face, posture, and orientation of face. The system will provide feedback for the user to improve discussion practices.

4.2 Component Design

There are four broad components within the scope of this project and they are as follows.

4.2.1 Environment and Initialization

- Selection of Topics:

  After the successful login of the user, a dynamically generated topic of discussion appears to grace the interface. The topic is extremely well crafted in such a manner that it is in harmony with the objective of the discussion, so that there is an active exchange of views by all the participants.

- Preparation Time:

  The user receives a brief, but sufficiently good preparation time, during which he/she collect his thoughts, do a bit of research, and craft an informed reply to the subject.

- Entering the Environment:

  The actual discussion is carried out in a virtual setting, into which the user goes in order to initiate the actual discussion after having done some preparation. It is an extremely interactive virtual meeting area that supports extensive interaction of participants, where the LLM model adds most of the contribution.

- Interacting with the LLM Model:

  In this context, the model plays a critical function in the discourse. It responds to the user's query in advanced language handling methods of the LLM, thereby creating meaningful and context-appropriate responses.

4.2.2 Model Interaction

- The LLM Models:In this project, the LLMs that are utilized are LLAMA, and Gemini. Each model is represented by an avatar, with an animated lip-sync overlay to simulate the appearance of speaking, in order to take-on different roles in the group discussion. The avatars are representations of the models with visual features that are representative of their respective personalities and speaking roles.

- To facilitate natural and dynamic discussions, each LLM model automatically responds with both positive and negative viewpoints, as you might see a real discussion participant do in an actual conversation. This allows for a more relaxed flow through a demonstration of the exchange of ideas instead of strictly pre-assigned roles.

4.2.3 Agent Functions and Features

- Agents Implementation:

  The LLM models act as discussion participants, engaging in conversations with the user to simulate a realistic group discussion environment. Each model responds naturally, providing diverse perspectives rather than strictly adhering to predefined roles.

- Conversation Handling:

  The system processes and stores user inputs to facilitate evaluation. User speech is converted to text using WebKit STT, and the model-generated responses are converted back to speech for a seamless interaction experience.

- Real-Time Engagement:

  The discussion flow is maintained through a structured turn-taking mechanism. If no one initiates the conversation, OLAMA starts speaking, followed by Gemini. The user can raise their hand to take a turn, ensuring active participation. The system also prevents overlapping responses.

4.2.4 Feedback System

- Facial Evaluation: Analyzes eye contact and facial expressions during conversation. Aids in the evaluation of non-verbal communication and engagement skills. The user's photographs are captured at varied intervals in the course of the discussion process to verify their facial affects and engagement. These images are to be tested, and a library suitable for face detection and value estimation is utilized to accomplish it i.e. MediaPipe. The kernel examines eye movement features and the face orientation.

- Speaking Time: Measures how long the user speaks compared to the agent. Ensures balanced participation and identifies dominant or passive behavior.

- Grammar: Checks spoken sentence structure and correctness. Provides feedback on fluency and common grammatical errors.

- Relevance: Evaluates if the user's responses stay on topic. Encourages clear, focused, and meaningful contributions.

- Feedback to the User: According to the score generated, the user receives instant feedback: a close-up view of their participation, alignment and keyword usage so that they can do better on their contributions.

4.3 Use Case Diagram

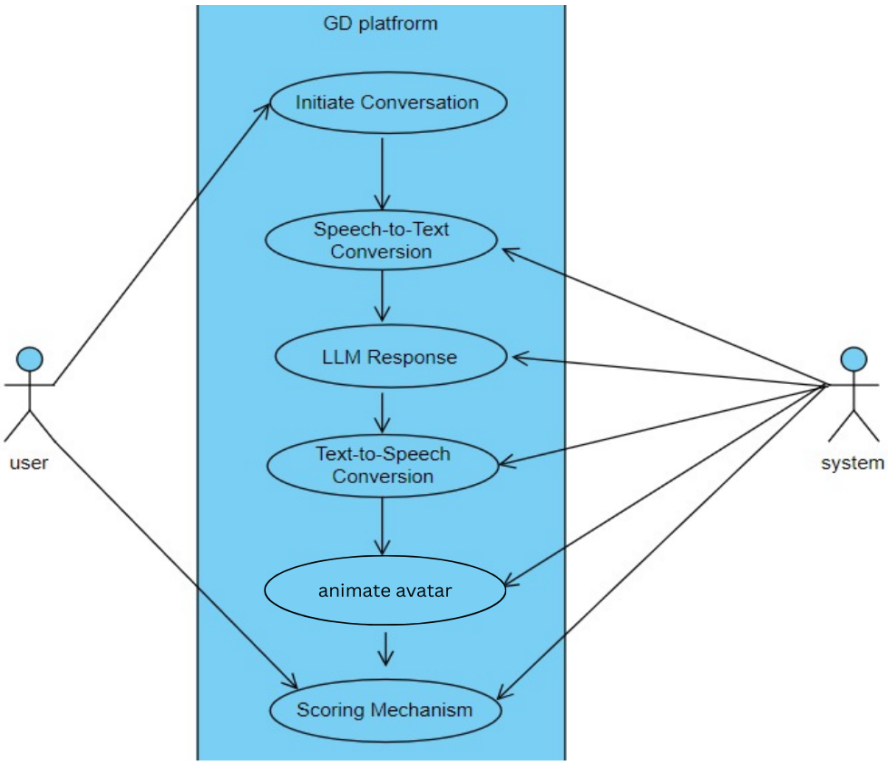


Figure 4.2: Use Case Diagram

The use case diagram represents the sequential flow of interactions between the user and the GD platform. Initially, the user logs into the platform using their Gmail account and then initiates the group discussion. Once the discussion begins, the user presses the "Talk" button to speak, and their audio is captured by the browser. This input is processed using the WebKit Speech Recognition API, which converts the spoken words into text. The transcribed text is then passed to the integrated large language models—Gemini and OLAMA—which analyze the input and generate an appropriate response. This text response is converted back into speech using the Web Speech Synthesis API to create a natural conversational experience. Each LLM is represented by a animated avatar, where lip-syncing is achieved by overlaying animated lips on the avatar images to match the generated speech. Meanwhile, the user's spoken inputs and screenshots taken every two minutes are stored in a MongoDB database, structured to keep track of the user's speech logs, topic discussed, and captured images. At the end of the session, a scoring mechanism evaluates the user's performance based on their speaking time, fluency, and relevance of

content. This feedback is then displayed on a post-discussion dashboard to help users assess and improve their group discussion skills.

4.4 Module Divisions

- Model Handling:

  The model handling module is the brain of the system and relies on large language models to formulate intelligent and relevant multi-turn dialogues. In this implementation, we use Gemini and LLaMA to generate context-aware replies from different perspectives. Gemini outputs fact-based, informative responses, while LLaMA produces argumentative and diverse responses, which works well in discussion-style interactions. These models are loaded on-the-fly and are provided with loading efficiencies, which are built to support low-latency. Memory management systems, such as buffer storage, are invoked to enable completion of interactions, utilizing recent replies to maintain the context and avoid repetition across a breadth of interactions. All of this continues to allow fluid and relevant interactions across multiple stages. The module also supports logic to context appropriately switch between models and filter output for safety and relevancy.

- Animation:

  The animation system enables realistic, real-time avatar interactions using Wav2Lip, a cutting-edge lip-syncing tool. It animates static images by syncing lip movements with audio, creating deepfake-like visuals that appear convincingly live. Each video frame is synthetically generated to match the speech, preserving facial identity and background. This creates the illusion of a talking avatar in real time. To enhance realism, additional expressions like head movement, blinking, and eye motion can be added using tools like MediaPipe or Avatarify. The final animated output is live-streamed, ensuring smooth and responsive visual feedback in sync with the audio.

- Speech-text module:

  The speech-text module enables fluid, voice-based interaction between the user and the system by converting spoken language to text and vice versa. Speech recognition

is handled using WebKit's built-in APIs, which transcribe user speech in real time and send it to the language model for processing. The generated response is then synthesized using text-to-speech (TTS) technology, allowing the avatar to speak back with natural-sounding voices. Different voices can be assigned to different AI personas, enhancing the immersive experience. This module also supports turn-taking management, ensuring a smooth conversational flow and avoiding overlaps or interruptions. Custom vocabulary support allows the system to adapt to specific discussion topics, making interactions more accurate and engaging. Overall, this module ensures seamless, intuitive voice communication with the AI.

- Evaluation:

  The evaluation module plays a key role in assessing user interaction quality through four main parameters: grammar, topic relevance, speaking time, and facial expressions. It combines both verbal and non-verbal analysis for comprehensive feedback. Tools like Qwen and LanguageTool are employed to examine grammatical correctness and linguistic fluency, while TextStat analyzes sentence readability and structure. Topic relevance is assessed by comparing user input with the ongoing discussion context. Speaking time is measured to ensure the user maintains a balanced interaction with the AI agent. For non-verbal analysis, MediaPipe evaluates facial behavior such as eye contact, engagement level, and attentiveness. All these insights are combined into a final performance score, helping users identify areas of strength and improvement in both speech and presentation.

4.5 Work Breakdown

4.6 Work Schedule - Gantt Chart

The system architecture, the different modules, the technologies used in their creation, and the expected system results are all covered in this chapter.

Sanghamitra Menon *STT and TTS*	**Sanoy Boby** *Animation*
Susan Mathew *Model handling*	**Vijay KV** *Evaluation*

Table 4.1: Work Breakdown and Responsibilities

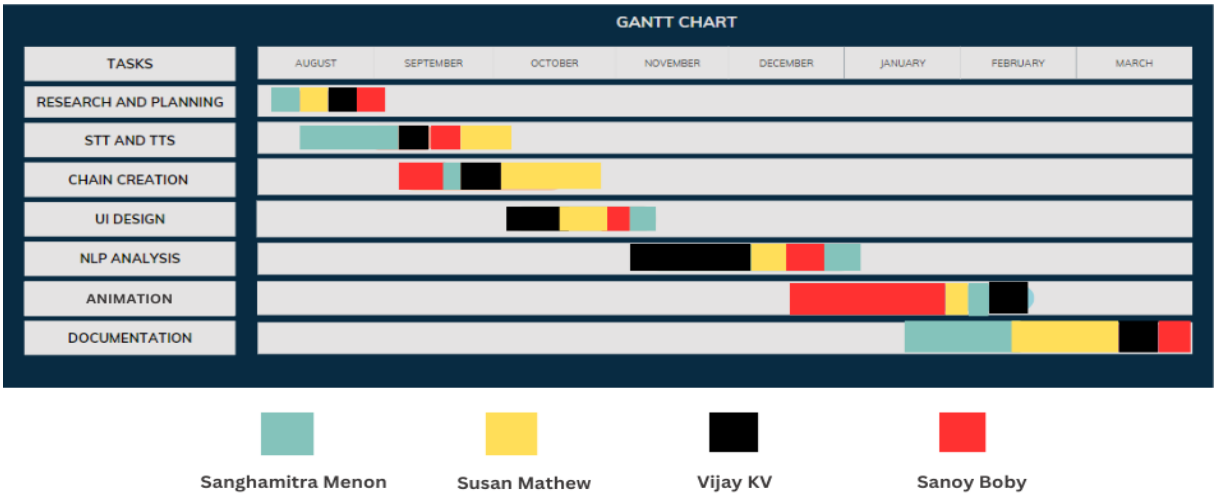


Figure 4.3: Gantt Chart

Chapter 5

System Implementation

This chapter discusses the system implementation of the web-based application developed to aid users in practicing group discussions with AI language multimodal models (LLMs) – the Group Discussion Practicing Platform. The platform allows users to engage in simulated group discussions with the Gemini and OLAMA AI models which serve as other participants in the discussion. This platform offers a speech recognition and feedback training system combined with natural language comprehension that addresses the needs of users who wish to refine their group discussion capabilities. In this platform, the architecture, user interaction flow, speech processing, integration of the LLM model, evaluation system, and database design are described in detail. The system is designed to facilitate users to improve their performance in group discussions by providing rich and interactive practice environments alongside useful feedback.

5.1 Overview of the System Architecture

In order to enhance and captivate users, the Group Discussion Practicing Platform uses several systems that function synergistically. The system is made of a client-side component which allows the user to interact with the system, a backend server that handles and response to the user and LLM models that take the place of the participants of the discussion via speech and action, and a DB that stores users' comments and their achievements. With simple and appealing design, the frontend is user-centric and has a live video feed of the user along with animated avatars of the LLM models Gemini and OLAMA. The backend powered by Flask guarantees the effective performance of the platform. User verification, speech recognition, and interactivity with the LLMs all occur simultaneously and in real-time to give the user the best experience. User information, discussion records, and marks are all saved in the database which is powered by MongoDB. The combination

of these components allows for the use of new technologies and useful feedback from the user, giving them an easy practice and great experience in refining their group discussion skills.

Key Components of the System Architecture

1. **Frontend (User Interface)** The frontend is designed with HTML, CSS, and JavaScript, making the platform responsive and usable on a variety of devices. The user interface has a live video stream of the user, as well as animated avatars of the LLM model, providing an immersive interface for a group discussion.

2. **Backend (Flask)** The backend runs on Flask, a light web framework that facilitates user requests, per-session data storage, as well as the processing of speech-to-text, and text-to-speech for the capabilities. It acts as the coordinator between all of the system components to maintain work flow.

3. **Database (MongoDB)** MongoDB is used to retain relevant user data such as speech logs, session observations, and feedback forms through the evaluation function. This architecture provides a scalable storage solution to accommodate multiple users in group discussion and fast query capabilities.

4. **LLM Integration** The integration of both the Gemini and OLAMA models is essential for modeling the participants of the group discussion. The models generate intelligible re-Responses for an exciting experience through their ability to actively flow throughout the conversation, drawing context from previous conversations, as needed.

5. **Speech Processing** WebKit Speech Recognition is used for converting user speech to text and generating text-to-speech responses for the LLM models. This enables real-time communication between the user and the AI models with minimal latency.

6. **Evaluation System** The evaluation system provides feedback on the user's performance in three key areas: speech fluency, posture, and content relevance. This comprehensive feedback mechanism is crucial for helping users improve their discussion skills over time through measurable metrics.

5.2 User Interaction Flow

Users engage in the Group Discussion Practicing Platform navigated by a predetermined intuitive flow. The flow ultimately helps prepare, practice, and gather feedback on the different stages of a group discussion. The flow mimics a group discussion setting, where the user is a primary speaker, and the group includes LLM models (Gemini and OLAMA) as discussion group members. The user initiates their interaction within the platform by signing into the platform, at which point they are taken to a dashboard and begin the discussion stage. Once logged into the platform, the user is given a discussion topic and rules for the discussion. The user will be provided with a one minute preparation time, during which the user can review the topic or the user can decide to change the topic. The system accommodates users if they are not feeling prepared and possibly gives them the option to review or change a topic.

Once the preparation time ends, the user enters the discussion phase, where the user's live video is displayed alongside the animated avatars of Gemini and OLAMA. If the user does not speak in the first 10 seconds of the discussion process, OLAMA will speak first, followed by Gemini, and the two LLMs will alternate. However, the user may interject at any time by raising their hand and having the LLMs wait for the user to finish the comment. The user can easily engage this feature by using a simple button located on the interface. The discussion will last 30 minutes and the user will engage with the LLMs providing thoughts and feedback to the LLMs prompts. The system will capture the user's speech and take screenshots every two minutes to assess the user's posture and facial expressions. At the end of the discussion, the user will receive a feedback dashboard in portrait-style view on content about the evaluation of the user's fluency of speech, posture, and relevance of the content they discussed with the LLMs. The feedback helps the user identify which obviously unavoidable item(s) may need improvement to help guide them with thoughts on future group discussions.

5.3 System Design and Backend Architecture

The Group Discussion Practicing Platform has a backend system powered by Flask, a lightweight and powerful web framework, that runs on Python. Flask is the main functionality of the system, structured to handle the user's request, manage the session, and

connect the user to the LLM models. User processing is done in Flask routes in which the user is authenticated, the speech is processed, and session data is stored in MongoDB. Flask also communicates with the WebKit Speech Recognition engine that is responsible for the speech-to-text and text-to-speech features.

Once a user logs into the application, Flask retrieves the user's profile information from the MongoDB database and initiates a session user will use during the group discussion to track user interactions. The Flask server facilitates the group discussion session performance the communication of the user with the LLM thick language models. The backend Flask system performs an important functionality for storing the user' speech logs, a log of periodic screen shots to track user performance, and the evaluation results in all content in a User Speech collection in MongoDB. This allows the system to track overall user development across multiple group session discussions and observe their performance overtime. The flexibility and integration of Flask and MongoDB streamlines the storage of user content.

5.4 Speech Processing and LLM Integration

The Group Discussion Practice Platform utilizes WebKit Speech Recognition to provide real-time speech-to-text and text-to-speech functions. This technology converts the user's speech into text for the LLM models to process and respond to the user input. The WebKit Speech Recognition engine is integrated into the frontend and backend of the application, establishing a seamless communication pathway between the user and the LLM models. Specifically, the user's speech is recorded through the device microphone, converted to text, and later used as input for the LLM models, Gemini and OLAMA.

Gemini and OLAMA are the AI models that simulate group discussion participants and generate intelligent responses based on the conversation flow. They are integrated into the application via RESTful APIs and allow the frontend to send user input and receive responses. Both Gemini and OLAMA are designed to generate varied responses and context to respond to users input naturally engaged in discussion. The application designers want to give the user the experience of talking to an avatar by adding images of human avatars representing the LLM models and animate the avatar's lips using an overlay to simulate talking interaction realism.

5.5 Evaluation System

The platform features a strong evaluation system that gives the user feedback on their performance in the context of the group discussion in real-time. This performance evaluation system is particularly focused on analyzing the user's Facial Engagement and Content Relevance, which are both foundational areas of effective communication in a group discussion, and the evaluation system provides unique feedback to support user improvement in both areas.

1. **Facial Analysis**: The platform has the ability to perform facial analysis utilizing periodic screenshots taken during the discussion session. The performance evaluation system specifically takes account of eye contact (did the user maintain their focus on the camera), and the user's facial orientation to verify they were appropriately positioned with their head. This evaluation helps support improvement in virtual communication habits by providing feedback to the user about their attentiveness and engagement level within the conversations. The user feedback is included in their post-discussion report.

2. **Content Relevance**: Content relevance is an important part of group discussion, because if the user is off topic, they may not be contributing to the discussion, which is then a loss for all users. The platform evaluation system reviews how well users stay on the discussion topic or how they veer off topic or of relevance. In cases of not staying on topic, the system will provide feedback to the user as part of the corrective feedback system, oriented to supporting the user in staying focused on key discussion points of the discussion.

The feedback generated by the evaluation system is stored in the MongoDB database, allowing users to review their progress over time and track improvements in their group discussion skills.

5.6 Database Design

The Group Discussion Practicing Platform uses MongoDB to store user data and session information. MongoDB's flexible, document-based structure is ideal for managing the

large volume of data generated during each group discussion, including user profiles, speech logs, and evaluation results. The platform's database is organized into two primary collections: Users and User Speech.

Database Structure

1. **Users Collection:** This collection stores basic information about the user, such as:

   - `_id`: Unique ObjectId (e.g., `ObjectId('67ebf5cee662598cc9c1c5cb')`)
   - `user_id`: String identifier (e.g., `"108995649414446690449"`)
   - `email`: User's email address (e.g., `"U2103213@rajagiri.edu.in"`)
   - `name`: Full name (e.g., `"VIJAY KV Rajagiri"`)
   - `photo_url`: Profile photo URL (e.g., `"https://lh3.googleusercontent.com/..."`)
   - `created_at`: ISO timestamp (e.g., `2025-04-01T14:18:54.399+00:00`)

2. **User Speech Collection:** This collection stores discussion session data:

   - `_id`: Unique ObjectId (e.g., `ObjectId('67ec0f7430f6c7re8a9c3796')`)
   - `user_id`: Reference to Users collection (e.g., `"112899430461594501 62"`)
   - `topic`: Discussion subject (e.g., `"Climate Change and Its Impact on Global Economics"`)
   - `speech_entries`: Array of transcript entries (initially empty)
   - `screenshots`: Array of periodic screenshots (with objects containing analysis data)

This database structure is designed to facilitate efficient tracking of user progress, as it stores historical data for multiple sessions. By organizing data into separate collections for users and speech logs, the platform ensures that user information is well-structured and easily accessible for analysis and feedback generation. MongoDB's NoSQL architecture allows for scalability, making it easy to expand as more users participate in group discussions.

5.7 Security and Privacy

Security and privacy are fundamental considerations in the design of the Group Discussion Practicing Platform. The platform ensures that user data is stored securely and that sensitive information, such as personal profiles and speech logs, is protected. To maintain user privacy, all data is encrypted both during transmission (using HTTPS) and at rest (using encryption mechanisms in MongoDB). Users are authenticated using secure login methods, and their personal data is only accessible to authorized system components. Furthermore, periodic audits and vulnerability assessments are conducted to identify and mitigate potential security risks.

5.8 Summary

The Group Discussion Practicing Platform is a comprehensive solution for individuals looking to enhance their group discussion and communication skills. By integrating advanced technologies such as speech recognition, LLM models, and AI-driven evaluation systems, the platform provides a highly interactive and insightful learning experience. The user-friendly interface, combined with real-time feedback on speech fluency, posture, and content relevance, allows users to continuously improve their performance. The backend architecture, using Flask and MongoDB, ensures smooth data handling and scalability, while security measures protect user privacy. As a result, this platform not only supports effective skill development but also contributes to fostering better communication habits, preparing users for real-world group discussions and debates.

Chapter 6

Results and Discussions

In this section, we share the findings from our Interactive GD Platform, built for realistic group discussion experiences, focused around a virtual avatar interface. The system consists of four main modules: model handling, animation, evaluation, and a speech-to-text interface that all come together smoothly and intelligently for ease of interaction throughout the discussion. The model handling module provides the conversational backbone, leveraging advanced language models like Gemini and LLaMA, working to produce coherent, multi-turn responses based on the topic of discussion. Memory buffers have also been embedded to help maintain the flow of conversation and avoid repeating prior statements. The avatar animation module allows us to provide a verbal experience by creating the appearance of a talking avatar using Wav2Lip to synchronize lip movements in real-time to the audio stream. We added additional expressiveness to the avatar while talking using MediaPipe and Avatarify that provide capabilities for blinking and head movement as well. The evaluation module provides feedback on your performance based on grammar, topic relevance, speaking time, and facial expressions during the discussion. We utilized Qwen, LanguageTool, TextStat, and MediaPipe in the evaluation module to provide feedback across multiple domains. Finally, the speech-to-text module provides an interactive voice experience as users speak virtually with the avatar, through WebKit APIs transcribing the speech to text for the user and converting the response back to speech using real-time voice rendering and custom vocabulary for fluidity. Turn-taking controls and response options naturally supported back-and-forth conversations. Testing and user feedback confirm the platform's reliability, responsiveness, and potential to enhance virtual learning, interview training, and collaborative discussions.

6.1 Overview

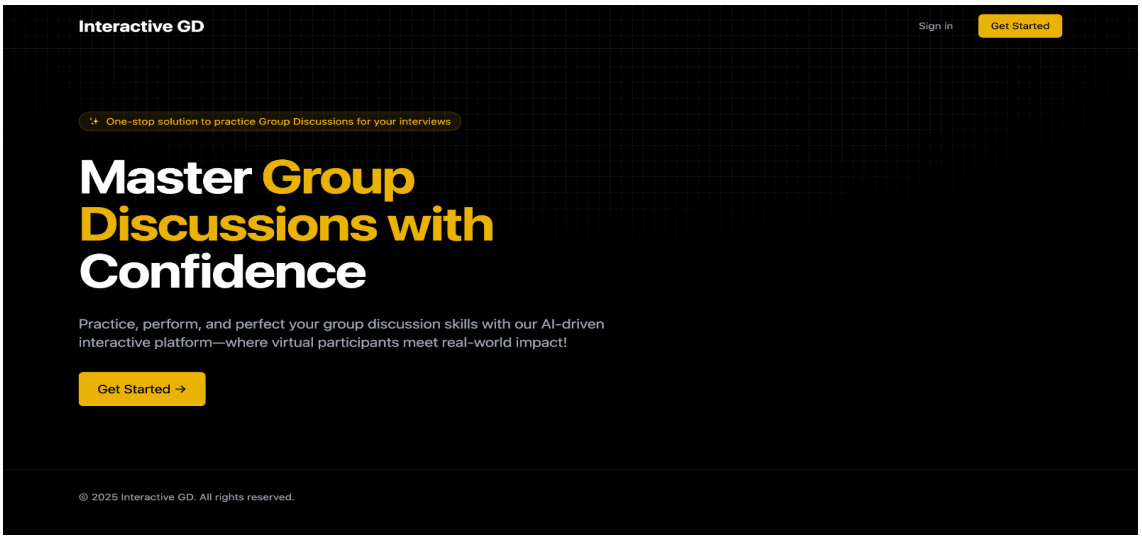


Figure 6.1: Landing page

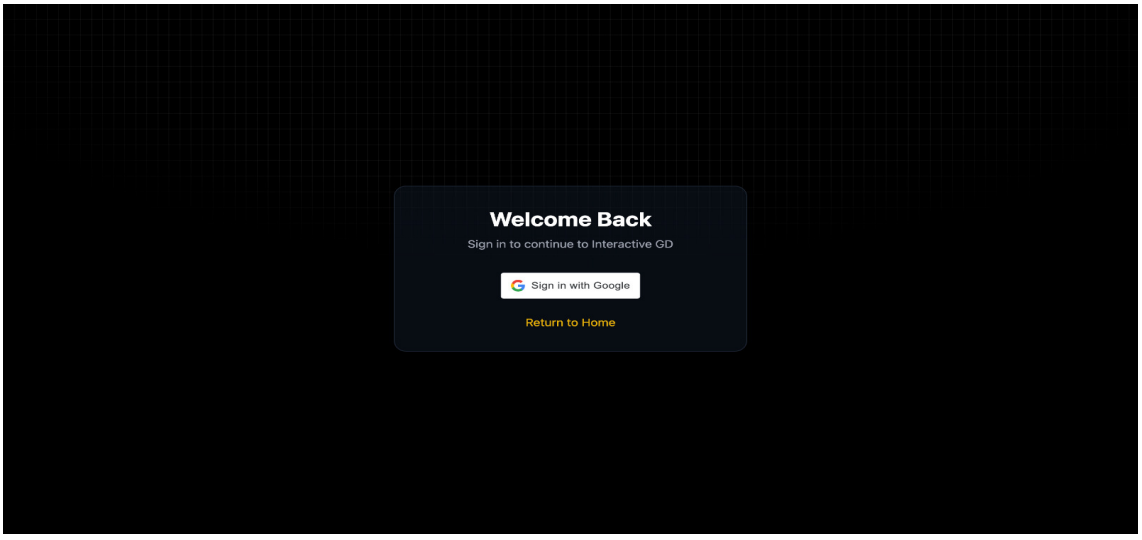


Figure 6.2: Sign-In page

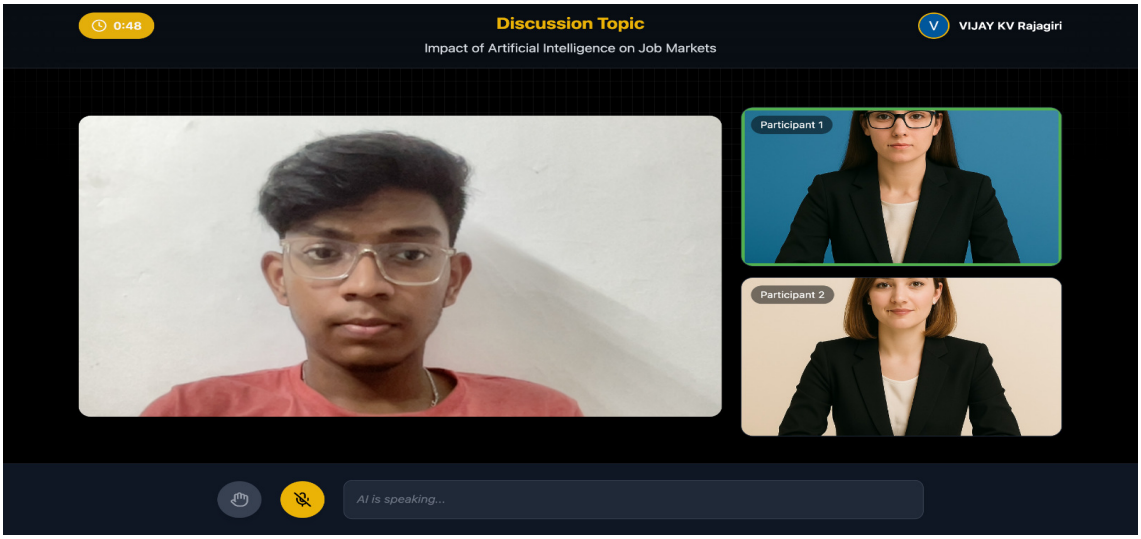


Figure 6.3: Virtual GD Environment

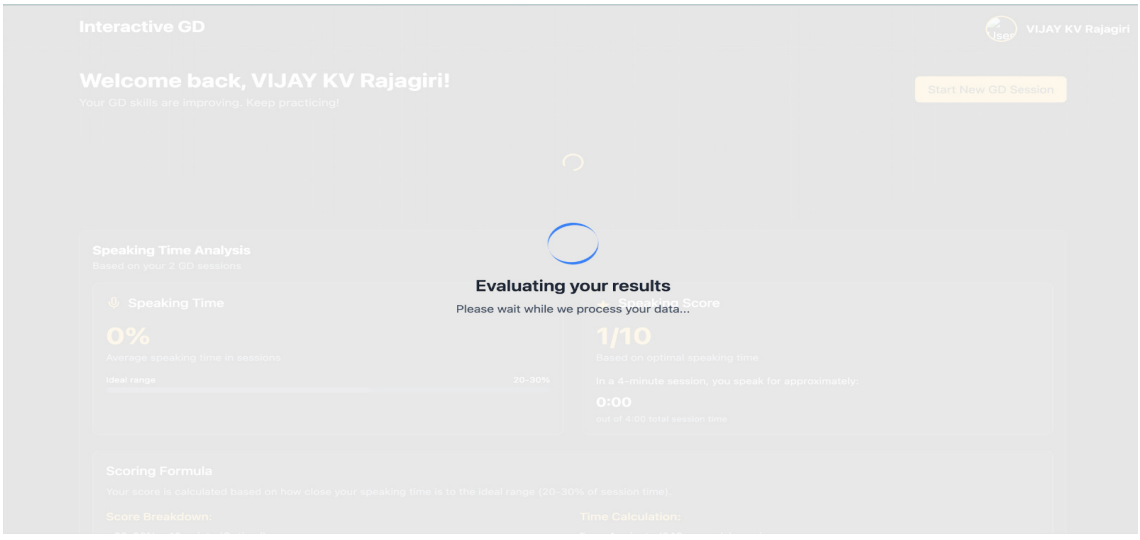


Figure 6.4: Result Evaluation

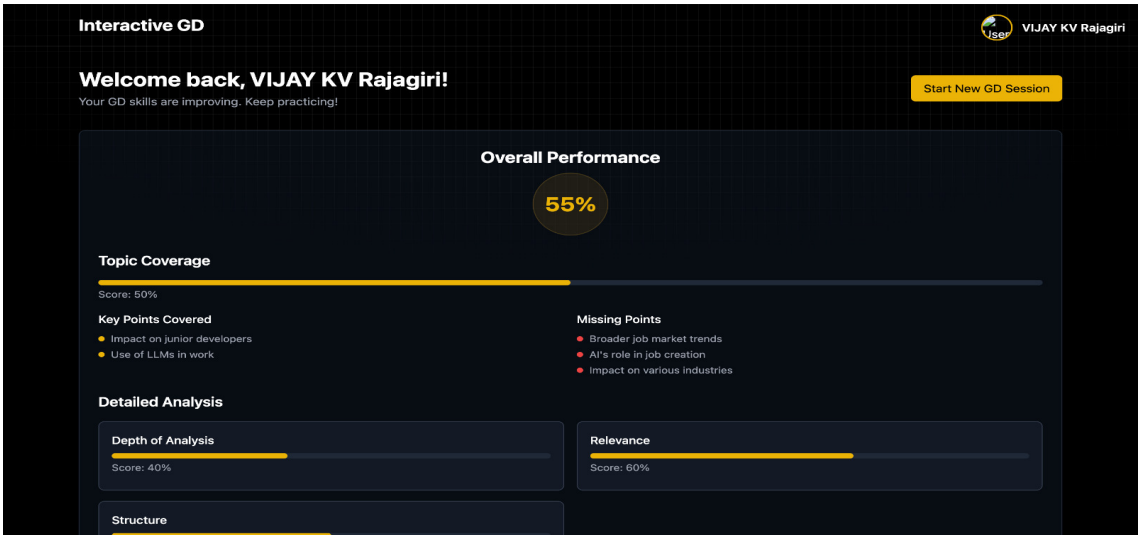


Figure 6.5: Overall Results

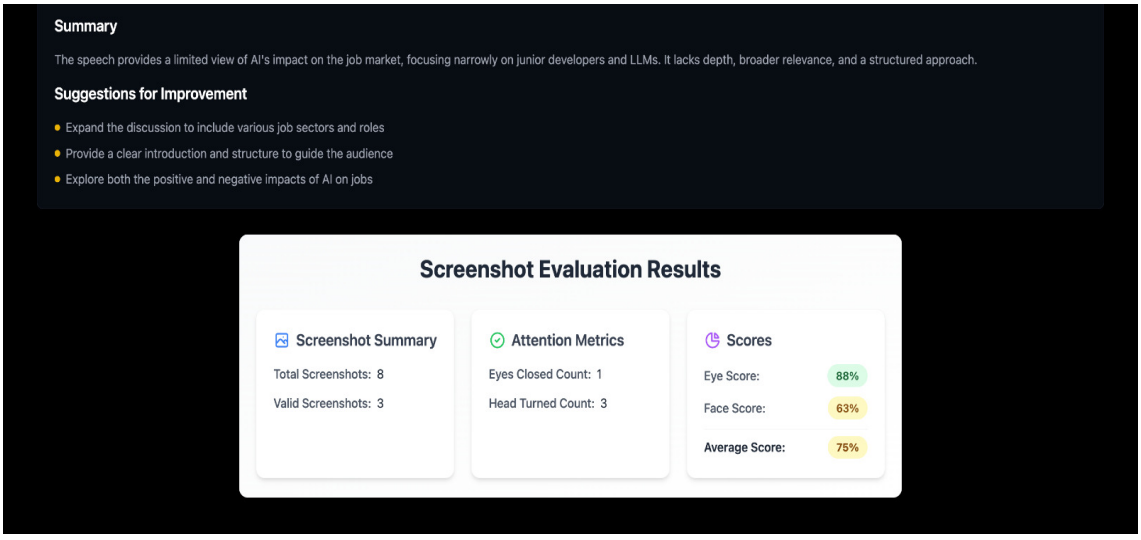


Figure 6.6: Performance Summary

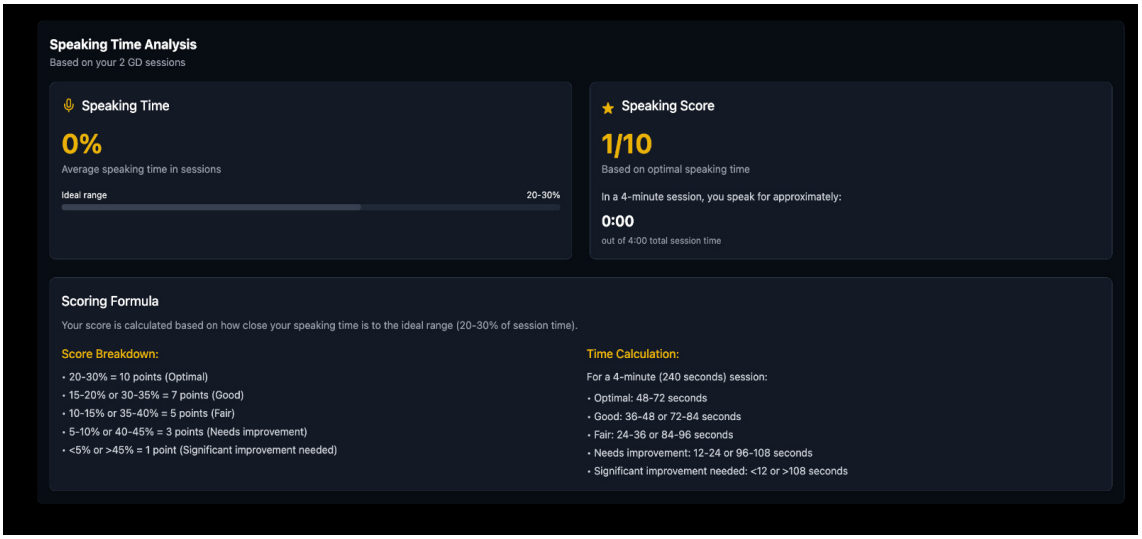


Figure 6.7: Analysis

Figure 6.8: Database Schema for Users

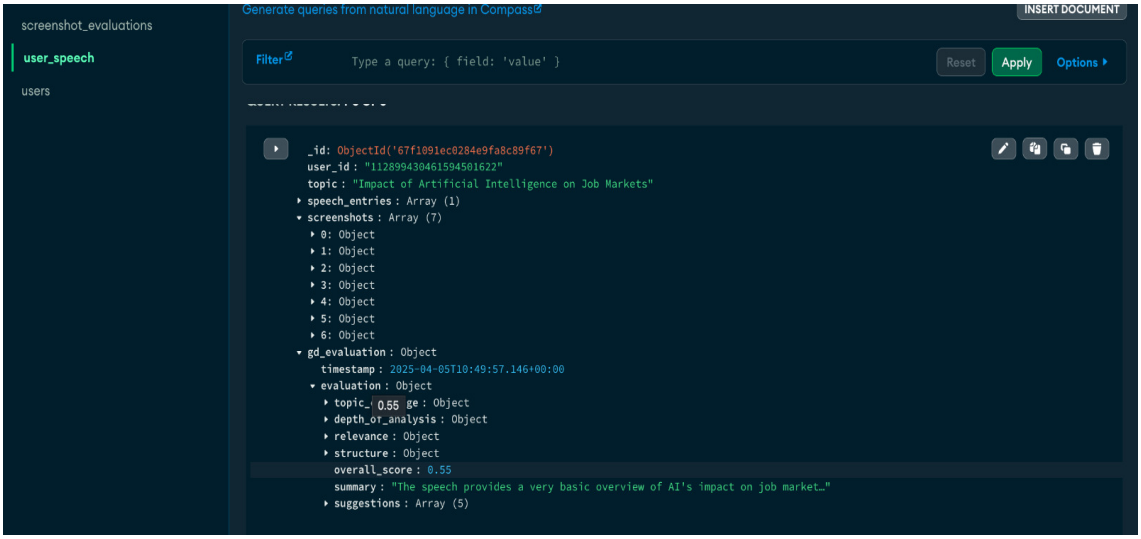


Figure 6.9: Database schema for user's speech

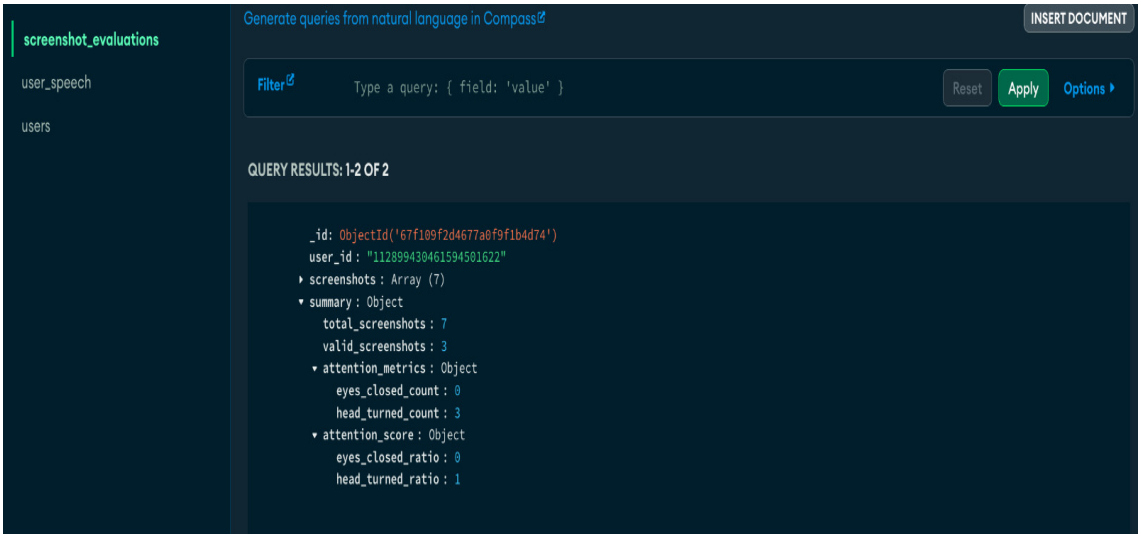


Figure 6.10: Database schema for screenshot evaluation

Chapter 7

Conclusions & Future Scope

The end goal of the project was to create a LLM integrated interactive group engagement where several models like Gemini and LLAMA could engage in stimulating conversations. But, the effective deployment of such a platform provided a structured platform encompassing turn-based dialogues and model integration alongside performance evaluation features which are administered in a competitive manner. This practice aimed to fulfill the requirements of collaborative thinking and teaching by mitigating biases and allowing fluidity of input on varying issues. Leveraging more LLMs added the value of in-depth conversations that accompanied greater insights thus being beneficial not only in schools but also in the corporate world and in other social scenarios.

Future injections would focus on enhanced multilingual features of the solution and consequently ensure language barrier free global communications. The personalization and customization of the platform could be improved through sentiment interpretation facilities and dynamic user profiling integrated inside the system. Furthermore, easy plus secure access to the platform can be achieved through provision of mobile applications, cloud-based deployment, enhanced encryption and other privacy techniques.

References

[1] M. Taeb and H. Chi, "Comparison of deepfake detection techniques through deep learning," *Journal of Cybersecurity and Privacy*, vol. 2, pp. 89–106, 03 2022.

[2] I. Ul Haq, M. Pifarre, and E. Fraca, "Natural language processing approach to evaluate real-time flexibility of ideas to support collaborative creative process," *International Journal of Emerging Technologies in Learning (iJET)*, vol. 19, pp. 93–107, 06 2024.

[3] D. Halvoník and J. Kapusta, "Large language models and rule-based approaches in domain-specific communication," *IEEE Access*, vol. PP, pp. 1–1, 01 2024.

[4] O. Samuel, O. Oluwatosin, and O. Segun, "A survey of variants of round robin cpu scheduling algorithms," *FUDMA JOURNAL OF SCIENCES*, vol. 4, no. 4, pp. 526 – 546, Jun. 2021. [Online]. Available: https://fjs.fudutsinma.edu.ng/index.php/fjs/article/view/513

Appendix A: Presentation

Interactive GD Platform

Team members

Sanghamitra Menon
Sanoy Boby
Susan Mathew
Vijay K V

Guide

Ms. Jomina John

Contents

Problem definition

Effective communication and collaboration skills are vital in academic and professional settings, yet many lack opportunities for structured practice. There is a clear need for a dedicated platform that allows users to practice group discussions, receive constructive feedback, and build confidence, ultimately enhancing their communication skills for collaborative challenges.

Purpose & need

The purpose of this project is to create a platform that enhances group discussions through multiple conversational models, allowing individuals to practice their speaking skills in a real-life simulation. This tool fosters communication and critical thinking without interruptions, making it valuable for interview preparation and helping candidates excel in group discussions.

Project objective

The platform aims to provide individuals with a structured environment to practice and improve their group discussion skills. It will facilitate engaging conversations, offer real-time feedback, and connect users with diverse perspectives, fostering effective communication and collaboration.

Literature survey

PAPER	ADVANTAGES	DISADVANTAGES
Ul Haq et al. [1] Natural Language Processing Approach to Evaluate Real-Time Flexibility of Ideas to Support Collaborative Creative Process (2024).	• NLP can analyze conversations in real-time and providing instant feedback	• NLP struggles to fully understand and emotional tone or cultural context
Halvoník et al. [2] Large Language Models and Rule-Based Approaches in Domain-Specific Communication (2024).	• LLMs can be fine-tuned for specific tasks and domains, improving their performance in particular areas • LLMs can generate varied viewpoints, enriching discussions by presenting multiple angles on a topic.	• LLMs depend on high-quality and diverse training data for effective performance. • For real-time updates, LLMs require continuous internet access to deliver accurate information.

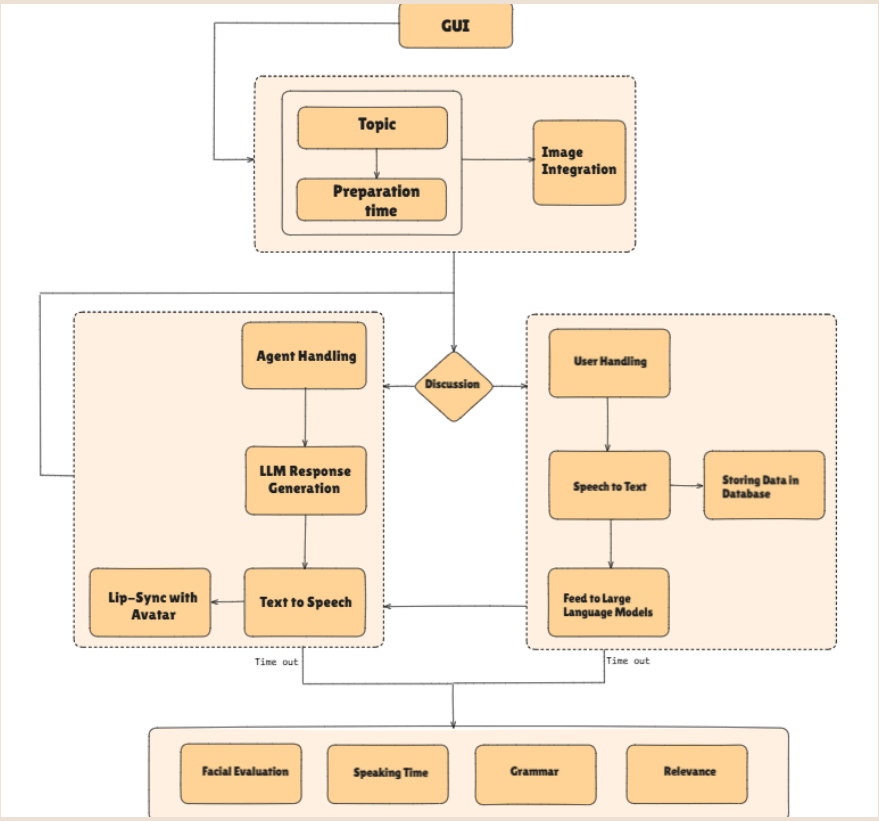
PAPER	ADVANTAGES	DISADVANTAGES
Olofintuyi et al. [1] A Survey of Variants of Round Robin CPU Scheduling Algorithms (2020).	• Equal turn-taking ensures all LLMs contribute.	• Turn-based delays can slow down the interaction.
Topsakal et al. [1] Creating Large Language Model Applications Utilizing LangChain: A Primer on Developing LLM Apps Fast (2023).	• Optimised memory compononets to keep track of conversations	• Limited size of buffers

Methodology

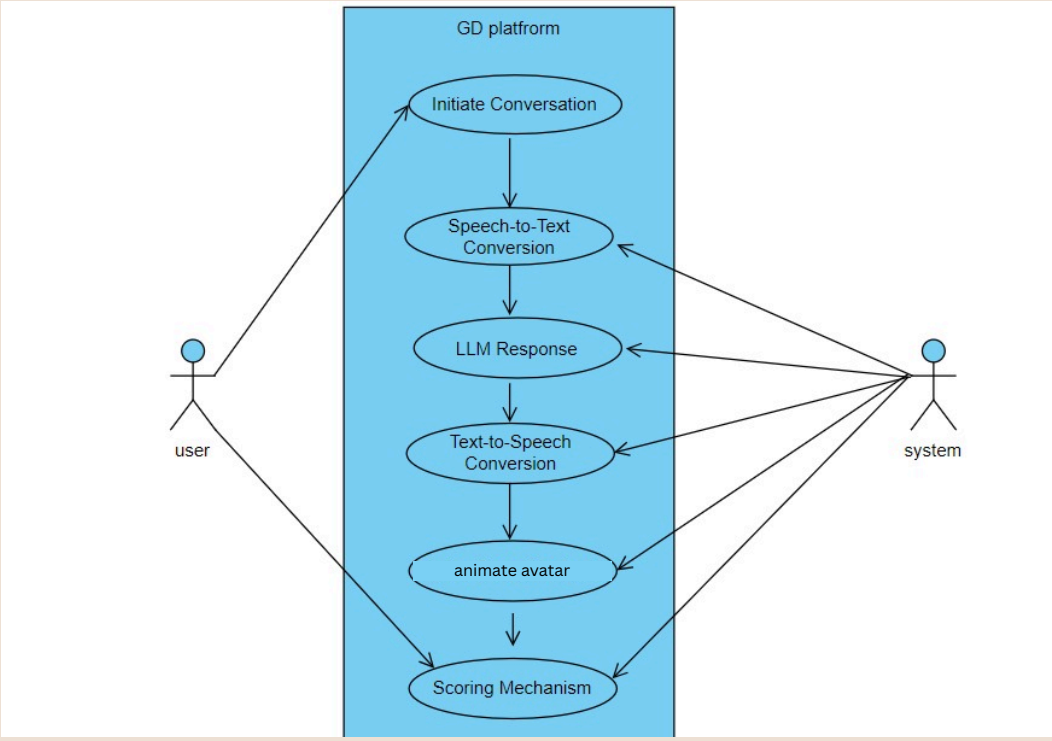
- The platform simulates group discussions by integrating human users with two AI participants powered by Gemini and LLaMA models.
- Animated avatars with real-time lip synchronization enhance participant realism
- Real-time speech processing utilizes WebKit API for bidirectional speech-text conversion

- Periodic facial engagement analysis evaluates eye contact and orientation through OpenCV-processed screenshots.
- The Qwen model performs content evaluation, scoring topic adherence and argument coherence.
- MongoDB's architecture efficiently stores user profiles and session-specific interaction data.
- Post-session feedback combines facial engagement metrics and content evaluation in visual dashboards.

Architecture diagram



Use case diagram

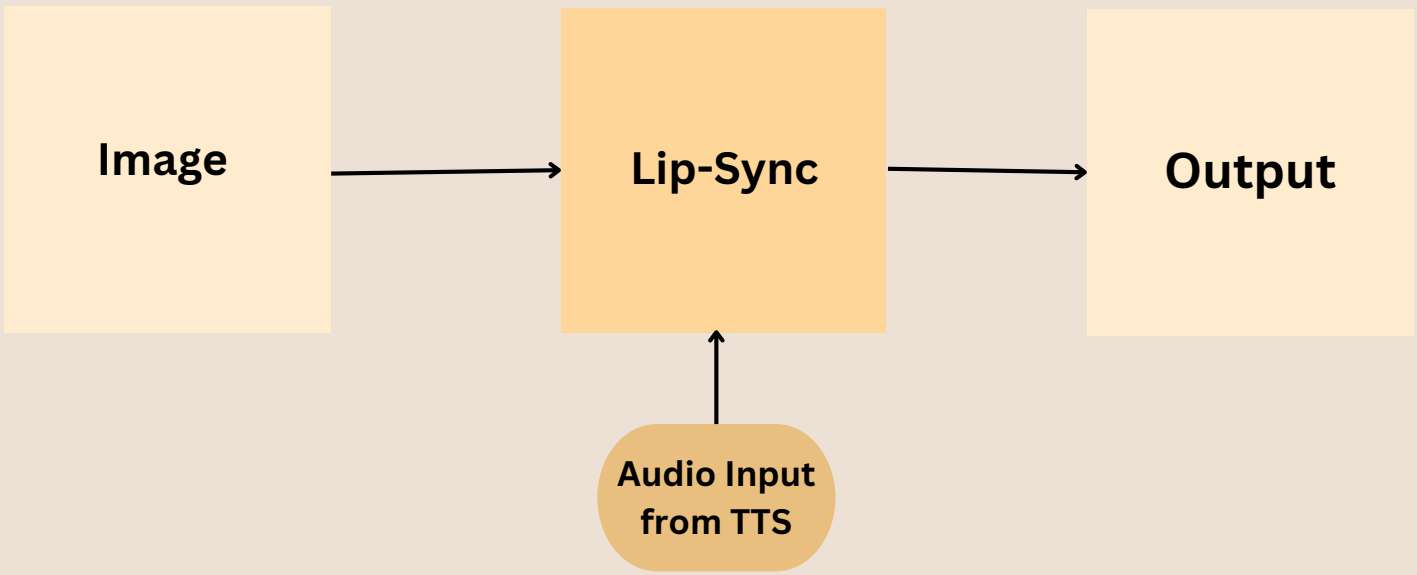


Modules

Model handling	**Evaluation**
Animation integration	**Speech-text module**

Animation Integration

- Wav2lip is an open-source project that uses AI to animate images
- It utilizes deep learning techniques like GANs to synchronize a person's lip movements with an image, enabling realistic lip-sync animations.

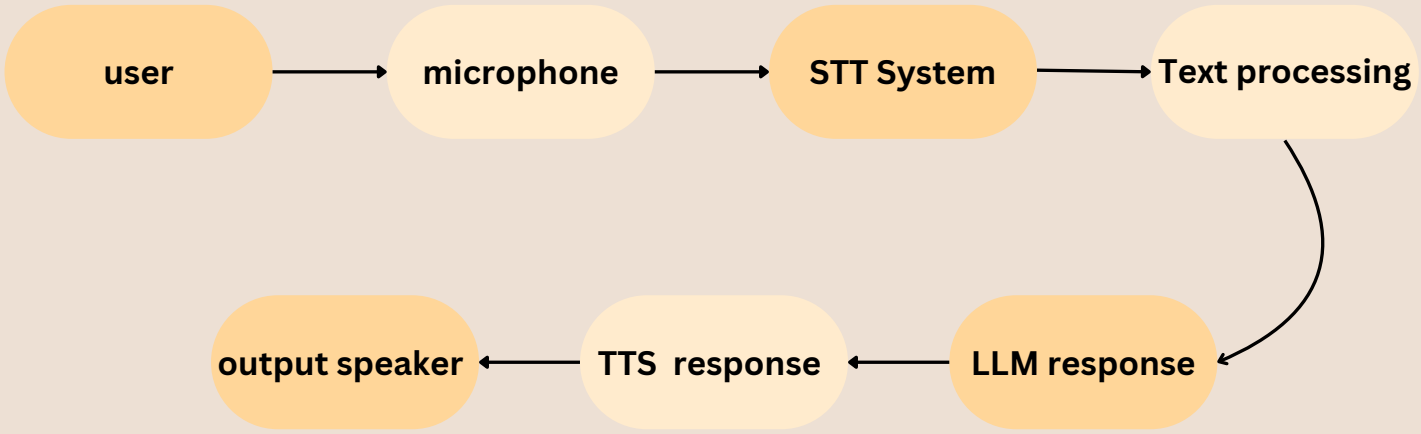


Animation Integration process

- It recieves audio and static image as input
- Wav2lip modifies the face of the static image (keeping the background and other facial features constant) to lip-sync the movements with the speech audio.
- After Avatarify generates video frames based on the lip-synced audio, stream these frames back to the frontend in real time.

Speech-to-Text (SST) and Text-to-Speech (TTS)

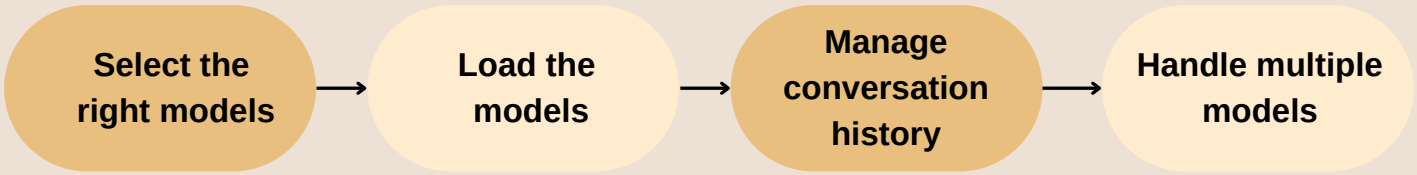
- User Speaks: User presses a button to talk.
- STT Conversion: Transcribes speech to text.
- LLM Response: LLM generates a reply.
- TTS Conversion: Converts text to speech.
- Synthesized speech: Played back to the user.



SST and TTS process

- Speech Recognition Toolkit: WebKit Speech Recognition API is used to transcribe user speech into text for processing by the LLM.
- Speech-to-Text Processing: When the user presses "Talk," the Speech Recognition Toolkit captures and transcribes the speech into text, which is then processed by the LLM to generate a response.
- Text-to-Speech Output: The LLM-generated response is converted into speech using Text-to-Speech, played back to the user with distinct voices assigned to each LLM for realistic interaction.

Model handling process



Select the right models

Decide which models will be used in the platform for generating conversation responses.

Gemini

- Gemini's advanced reasoning capabilities enable meaningful and context-aware contributions to group discussions.
- It’s multimodal abilities allow seamless integration of text, images, and audio, enhancing interactive dialogue experiences.

LLaMA

- LLaMA (Large Language Model Meta AI) is an open-source, large-scale language model developed by Meta, designed to understand and generate human-like text across multiple applications.
- It is useful for a wide range of tasks, such as natural language processing, conversation generation, content creation, and enhancing AI-driven systems with scalable and efficient performance.

Load the models

- Ensure the models are properly loaded and ready to generate responses.
- Gemini is loaded to generate detailed, context-aware responses, effectively balancing counterarguments and supporting discussions.
- LLaMA, when loaded, delivers flexible and scalable responses, ensuring seamless integration into conversations while providing both supportive and challenging points.
- The models are initialized to seamlessly integrate and respond based on user input and conversation history.

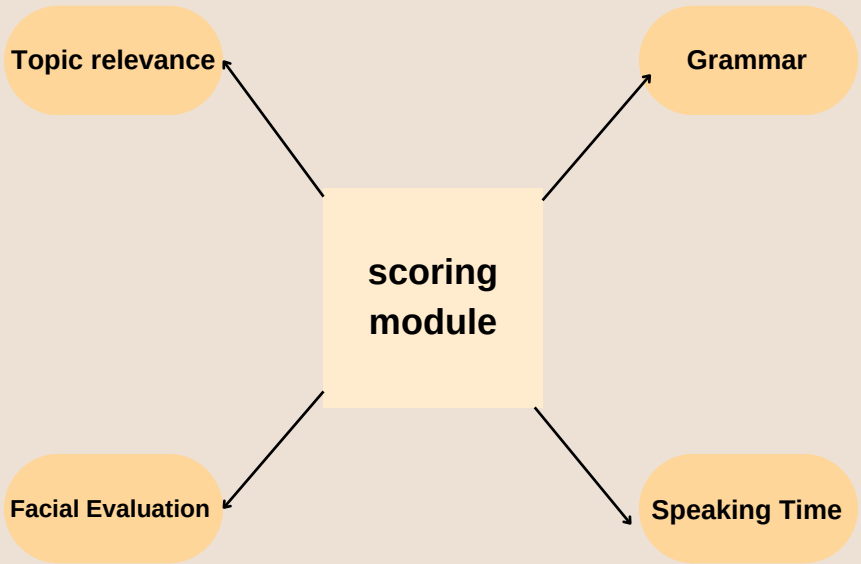
Managing Conversation History

An array is used to store LLM responses to prevent repetition and maintain context-aware replies throughout the discussion.

Allow User Contributions

Users can contribute thoughts only after a model completes its point, maintaining a smooth flow of conversation.

Evaluation



Topic Relevance

- Speech evaluation implemented using the Qwen language model.
- Qwen provides insights in three areas:
  - Analysis, Key Points Covered, and Missing Points.
- Scoring based on:
  - Depth of Analysis (0.8), Relevance (0.85), Structure (0.7).
- Generates an Overall Score: 7.8/10.
- Delivers a summary and improvement suggestions to enhance speech quality.

Grammar

- Evaluation is performed on transcribed text only (no audio analysis).
- Text analysis is done using:

1. TextStat – for readability, complexity, and structural scoring.
2. LanguageTool (Python) – for grammar, spelling, and style checks.

Speaking Time

- User speaking time is extracted using state management during the Group Discussion (GD).
- Tracks when a user starts and stops speaking within the total GD duration.
- Final speaking score is calculated using the formula:
  (User's Speaking Time / Total GD Time) × 100

Work breakdown & responsibilities

Sangamitra Menon

STT and TTS

Speech-to-Text and Text-to-Speech modules enable seamless voice interaction and real-time AI-driven dialogue simulation.

Sanoy Boby

Animation

Avatars are animated with moving lips that match the speech, making the discussion feel more real.

Susan Mathew

Model handling

The system picks and loads the best AI models to keep the conversation smooth and on-topic.

Vijay KV

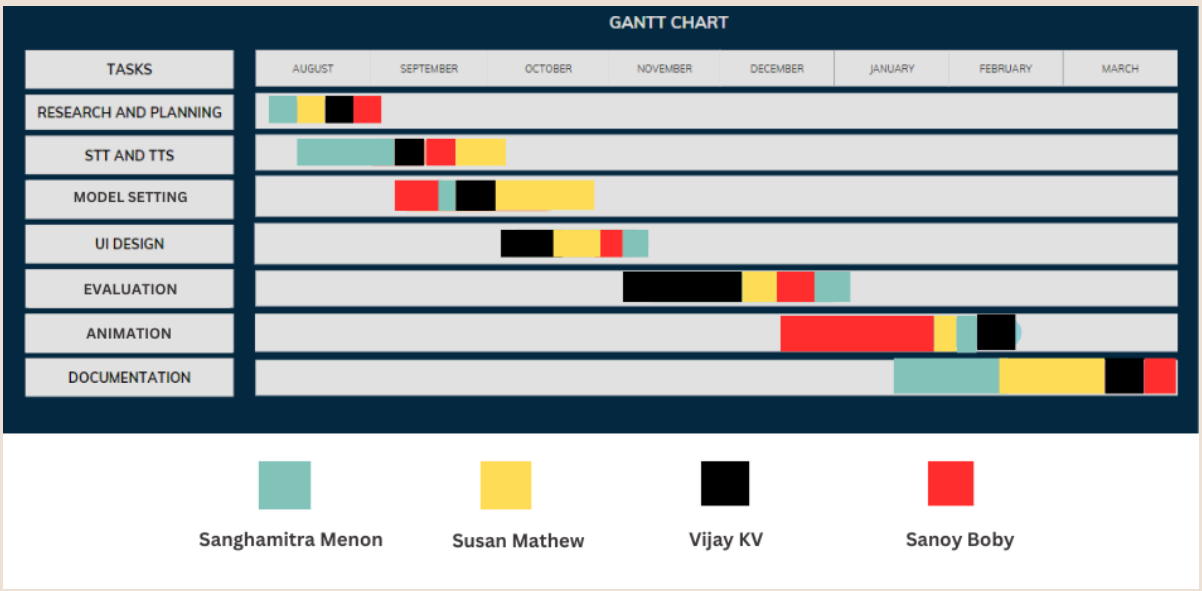
Evaluation

The evaluation system assesses topic relevance, grammar, facial engagement, and speaking time to generate performance feedback.

Assumptions

- User Consent is given for facial and voice data analysis.
- System Requirements like GPU and internet are met.
- User Readiness to engage with AI-driven feedback.
- Model Availability and stability of LLM APIs is ensured.

Gantt chart



Requirements

Hardware

- Intel Core i5, 8GB RAM
- 4GB GPU
- NVIDIA GeForce GTX
- 1650 - graphics card
- OS: Windows 10 64-bit,
- Ubuntu

Software

- Development environment: Visual Studio code
- Libraries: React,tailwind,openpose
- Frameworks: Flask
- Database: MongoDB

Budget

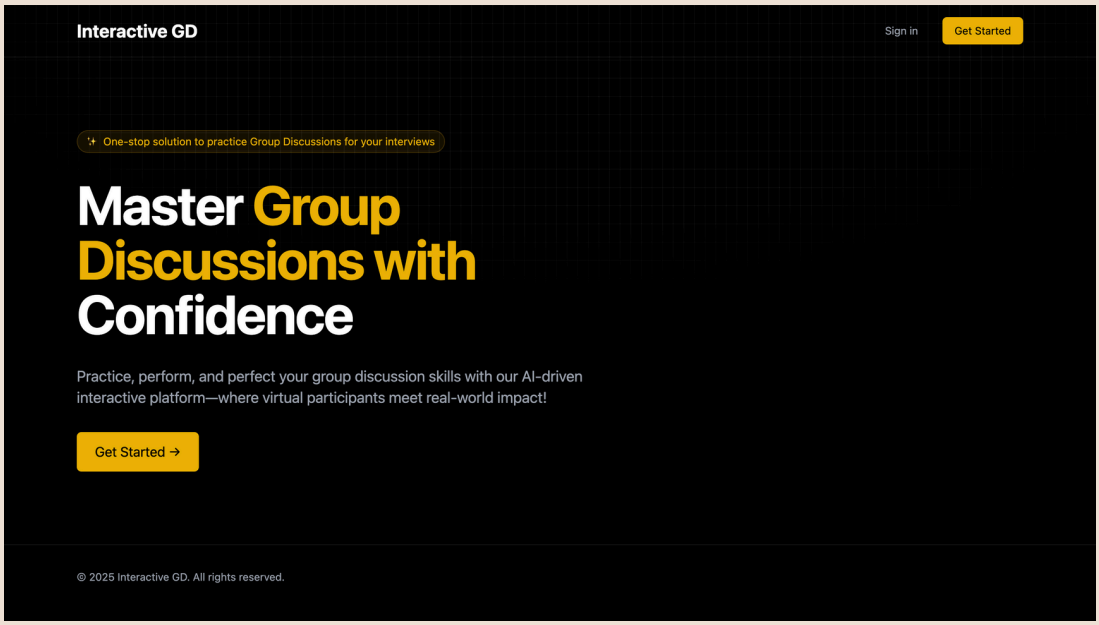
SL NO	ITEMS	AMOUNT
1	Hosting Service	2000
2	Cloud Database	1000
3	Avatarify (Premium)	1000
4	Publishing	2000
	TOTAL:	6000

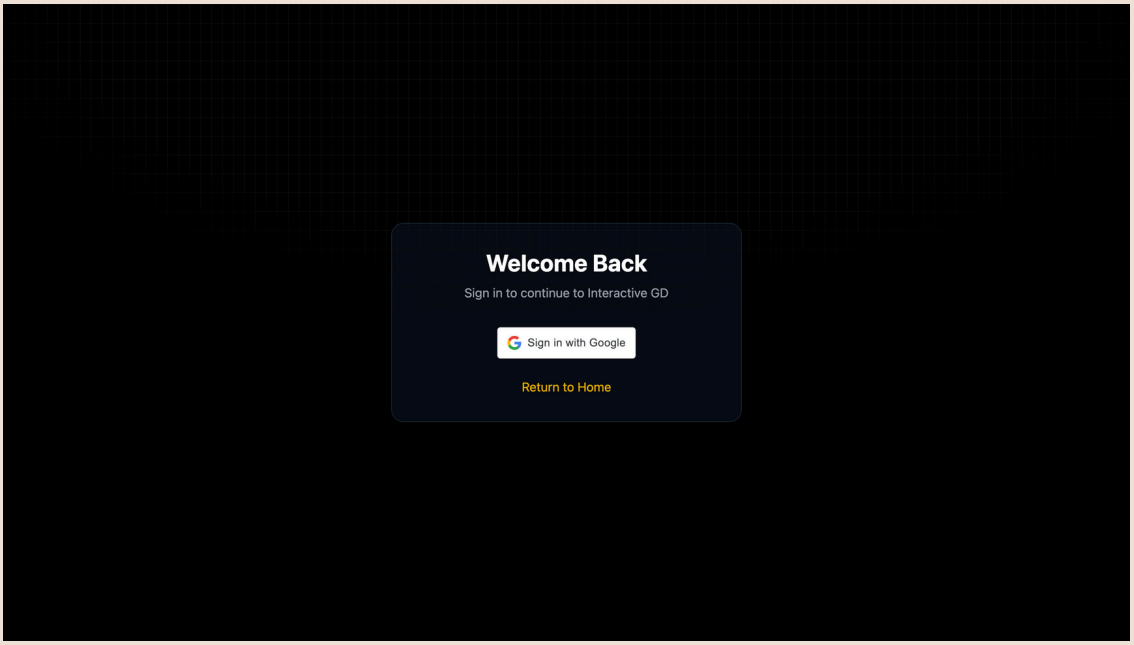
Risk & Challenges

- Performance and Scalability: Processing large amounts of data efficiently and quickly to generate video elements could pose performance challenges, especially when dealing with diverse video lengths.

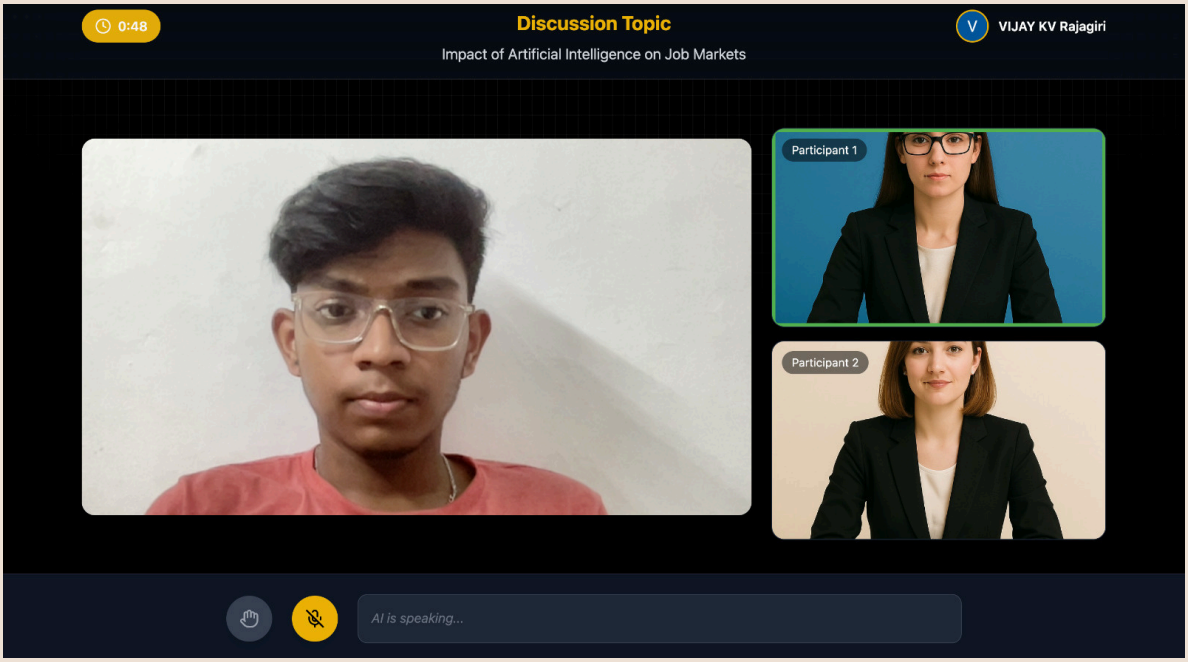
- Requires beta-testing to uncover edge cases and bugs.

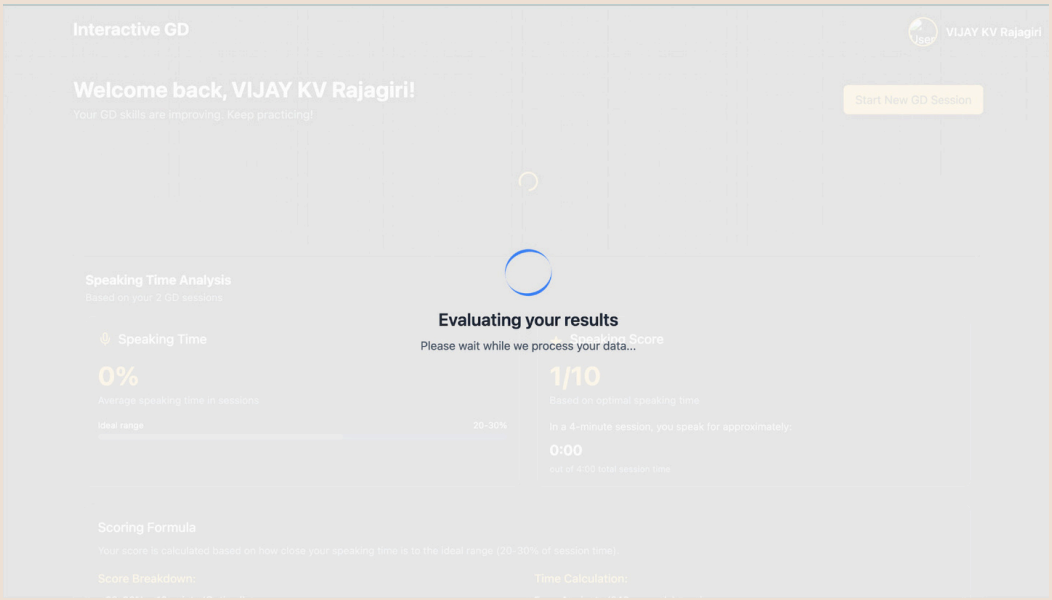
Outputs

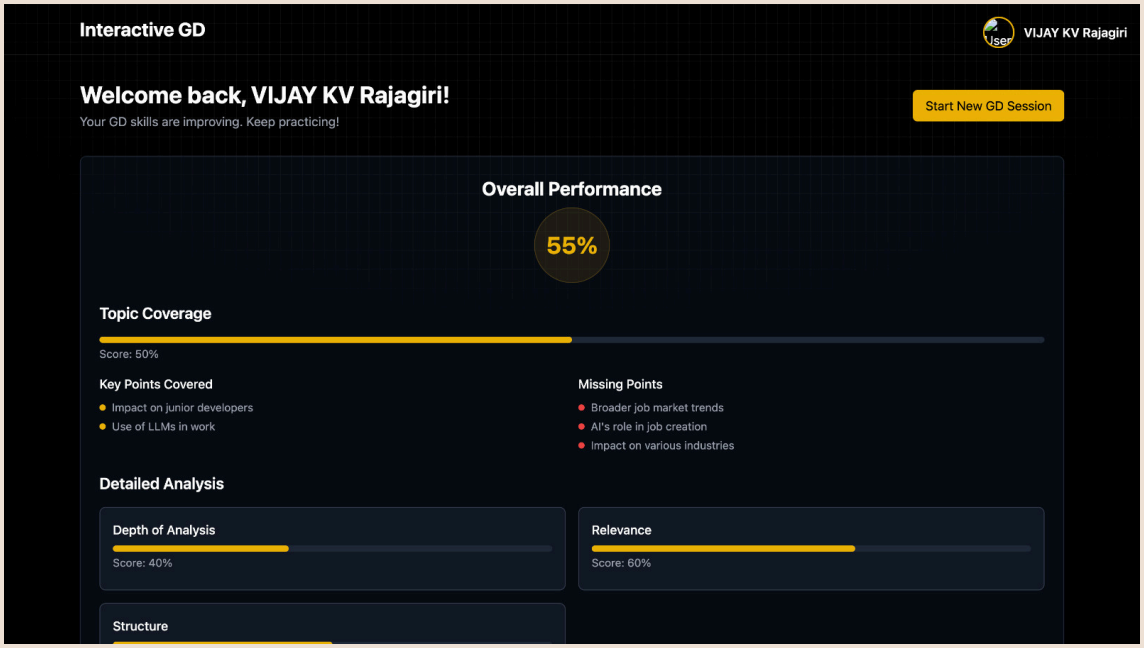


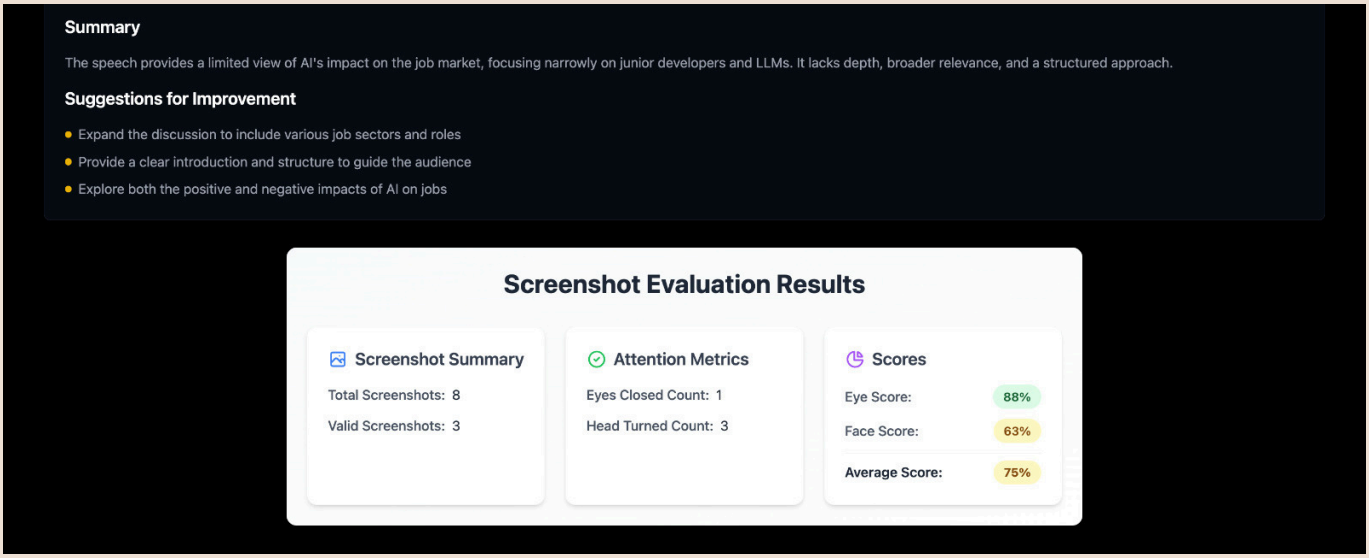

Welcome Back
Sign in to continue to Interactive GD
Sign in with Google
Return to Home

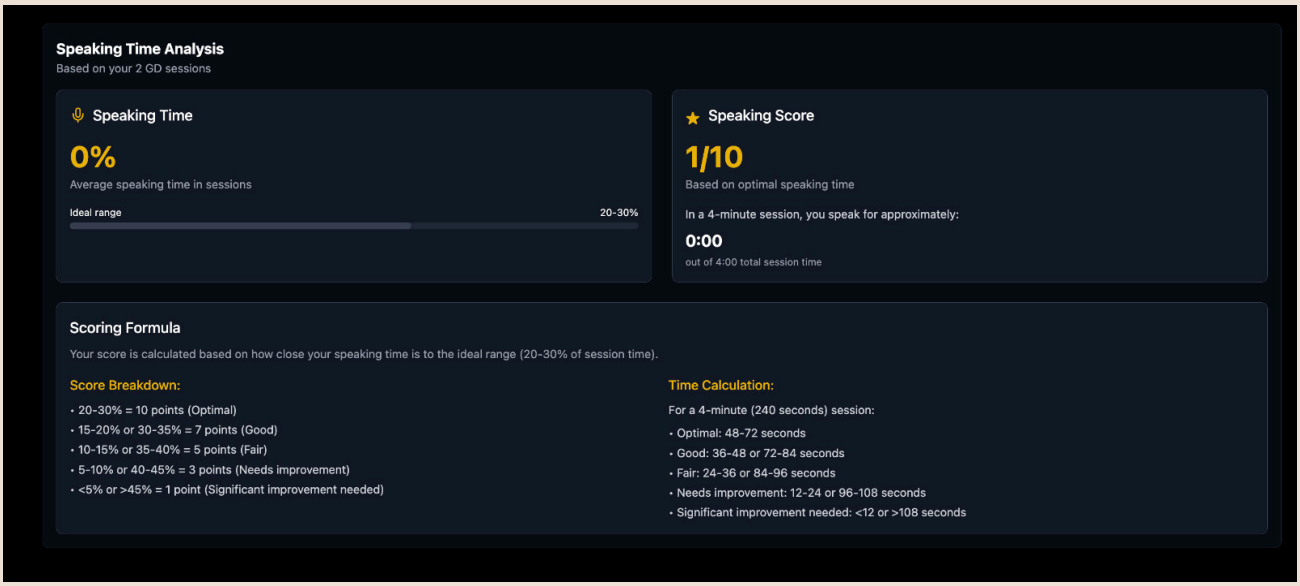

Interactive GD
Prep Time: 26s
GD Topic
Impact of Artificial Intelligence on Job Markets
Guidelines for the Group Discussion:
1. Listen actively and respect others' opinions
2. Speak clearly and concisely
3. Support your arguments with relevant examples
4. Encourage quieter participants to contribute
5. Stay on topic and avoid personal attacks
6. Summarize key points towards the end


0:48
Discussion Topic
Impact of Artificial Intelligence on Job Markets
VIJAY KV Rajagiri
Participant 1
Participant 2
AI is speaking...


Interactive GD
VIJAY KV Rajagiri
Welcome back, VIJAY KV Rajagiri!
Your GD skills are improving. Keep practicing!
Start New GD Session
Speaking Time Analysis
Based on your 2 GD sessions
Evaluating your results
Please wait while we process your data...
Speaking Time
0%
Average speaking time in sessions
Ideal range
20-30%
Score
1/10
Based on optimal speaking time
In a 4-minute session, you speak for approximately:
0:00
out of 4:00 total session time
Scoring Formula
Your score is calculated based on how close your speaking time is to the ideal range (20-30% of session time).
Score Breakdown:
Time Calculation:


Interactive GD
VIJAY KV Rajagiri
Welcome back, VIJAY KV Rajagiri!
Your GD skills are improving. Keep practicing!
Start New GD Session
Overall Performance
55%
Topic Coverage
Score: 50%
Key Points Covered
Impact on junior developers
Use of LLMs in work
Missing Points
Broader job market trends
AI's role in job creation
Impact on various industries
Detailed Analysis
Depth of Analysis
Score: 40%
Relevance
Score: 60%
Structure


Summary
The speech provides a limited view of AI's impact on the job market, focusing narrowly on junior developers and LLMs. It lacks depth, broader relevance, and a structured approach.
Suggestions for Improvement
Expand the discussion to include various job sectors and roles
Provide a clear introduction and structure to guide the audience
Explore both the positive and negative impacts of AI on jobs
Screenshot Evaluation Results
Screenshot Summary
Total Screenshots: 8
Valid Screenshots: 3
Attention Metrics
Eyes Closed Count: 1
Head Turned Count: 3
Scores
Eye Score: 88%
Face Score: 63%
Average Score: 75%



Conclusion

The Group Discussion Practicing Platform provides a realistic, AI-powered environment to improve communication skills through interactive sessions with LLM avatars. With integrated speech processing, facial evaluation and real-time feedback, the system delivers personalized insights that enhance performance and readiness for real-world GD scenarios.

Future references

- VR/AR Integration for immersive, real-life GD simulations.
- Multilingual Support to cater to diverse user backgrounds.
- Emotion Detection to analyze tone, expression, and mood.
- Academic & Corporate Use as a soft-skill training tool.

References

1. Ul Haq, I., Pifarré, M., & Fraca, E. (2024). Natural Language Processing Approach to Evaluate Real-Time Flexibility of Ideas to Support Collaborative Creative Process. International Journal of Emerging Technologies in Learning (iJET), 19(05), pp. 93–107. https://doi.org/10.3991/ijet.v19i05.47465
2. D. Halvoník and J. Kapusta, "Large Language Models and Rule-Based Approaches in Domain-Specific Communication," in IEEE Access, vol. 12, pp. 107046-107058, 2024, doi: 10.1109/ACCESS.2024.3436902.
3. Olofintuyi, S.S., Omotehinwa, T.O. and Owotogbe, J.S., 2020. A survey of variants of round robin CPU scheduling algorithms. FUDMA JOURNAL OF SCIENCES, 4(4), pp.526-546.
4. Topsakal, Oguzhan, and Tahir Cetin Akinci. "Creating large language model applications utilizing langchain: A primer on developing llm apps fast." International Conference on Applied Engineering and Natural Sciences. Vol. 1. No. 1. 2023.

THANK YOU

Appendix B: Vision, Mission, Programme Outcomes and Course Outcomes

Vision, Mission, Programme Outcomes and Course Outcomes

Institute Vision

To evolve into a premier technological institution, moulding eminent professionals with creative minds, innovative ideas and sound practical skill, and to shape a future where technology works for the enrichment of mankind.

Institute Mission

To impart state-of-the-art knowledge to individuals in various technological disciplines and to inculcate in them a high degree of social consciousness and human values, thereby enabling them to face the challenges of life with courage and conviction.

Department Vision

To become a centre of excellence in Computer Science and Engineering, moulding professionals catering to the research and professional needs of national and international organizations.

Department Mission

To inspire and nurture students, with up-to-date knowledge in Computer Science and Engineering, ethics, team spirit, leadership abilities, innovation and creativity to come out with solutions meeting societal needs.

Programme Outcomes (PO)

Engineering Graduates will be able to:

1. Engineering Knowledge: Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.

2. Problem analysis: Identify, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.

3. Design/development of solutions: Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.

4. Conduct investigations of complex problems: Use research-based knowledge including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.

5. Modern Tool Usage: Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations.

6. The engineer and society: Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal, and cultural issues and the consequent responsibilities relevant to the professional engineering practice.

7. Environment and sustainability: Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.

8. Ethics: Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.

9. Individual and Team work: Function effectively as an individual, and as a member or leader in teams, and in multidisciplinary settings.

10. Communication: Communicate effectively with the engineering community and with society at large. Be able to comprehend and write effective reports documentation. Make effective presentations, and give and receive clear instructions.

11. Project management and finance: Demonstrate knowledge and understanding of engineering and management principles and apply these to one's own work, as a member and leader in a team. Manage projects in multidisciplinary environments.

12. Life-long learning: Recognize the need for, and have the preparation and ability to engage in independent and lifelong learning in the broadest context of technological change.

Programme Specific Outcomes (PSO)

A graduate of the Computer Science and Engineering Program will demonstrate:

PSO1: Computer Science Specific Skills
The ability to identify, analyze and design solutions for complex engineering problems in multidisciplinary areas by understanding the core principles and concepts of computer science and thereby engage in national grand challenges.

PSO2: Programming and Software Development Skills
The ability to acquire programming efficiency by designing algorithms and applying standard practices in software project development to deliver quality software products meeting the demands of the industry.

PSO3: Professional Skills
The ability to apply the fundamentals of computer science in competitive research and to develop innovative products to meet the societal needs thereby evolving as an eminent researcher and entrepreneur.

Course Outcomes (CO)

After the completion of the course the student will be able to:

Course Outcome 1: Model and solve real world problems by applying knowledge across domains (Cognitive knowledge level: Apply).

Course Outcome 2: Develop products, processes or technologies for sustainable and socially relevant applications (Cognitive knowledge level: Apply).

Course Outcome 3: Function effectively as an individual and as a leader in diverse teams and to comprehend and execute designated tasks (Cognitive knowledge level: Apply).

Course Outcome 4: Plan and execute tasks utilizing available resources within timelines, following ethical and professional norms (Cognitive knowledge level: Apply).

Course Outcome 5: Identify technology/research gaps and propose innovative/creative solutions (Cognitive knowledge level: Analyze).

Course Outcome 6: Organize and communicate technical and scientific findings effectively in written and oral forms (Cognitive knowledge level: Apply).

Appendix C: CO-PO-PSO Mapping

COURSE OUTCOMES:

After completion of the course, the student will be able to:

SL.NO	DESCRIPTION	Bloom's Taxonomy Level
CO1	Model and solve real-world problems by applying knowledge across domains (Cognitive knowledge level:Apply).	Level3: Apply
CO2	Develop products, processes, or technologies for sustainable and socially relevant applications. (Cognitive knowledge level:Apply).	Level 3: Apply
CO3	Function effectively as an individual and as a leader in diverse teams and comprehend and execute designated tasks. (Cognitive knowledge level:Apply).	Level 3: Apply
CO4	Plan and execute tasks utilizing available resources within timelines, following ethical and professional norms (Cognitive knowledge level:Apply).	Level 3: Apply
CO5	Identify technology/research gaps and propose innovative/creative solutions (Cognitive knowledge level:Analyze).	Level 4: Analyze
CO6	Organize and communicate technical and scientific findings effectively in written and oral forms (Cognitive knowledge level:Apply).	Level 3: Apply

CO-PO AND CO-PSO MAPPING

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	3	3	2	3	2	2	1					3		
CO2	3	3	3		2	3	2	2		1			2	2	
CO3									3	3	2	1			3
CO4								2	3	2	2	2			3
CO5	2	3	3	2	3							2	3		
CO6					2			2	2	3	2	1			3

3/2/1: high/medium/low

JUSTIFICATIONS FOR CO-PO MAPPING

Mapping	Level	Justification
101003/CS722U.1-PO1	M	Knowledge in the area of technology for project development using various tools results in better modeling.
101003/CS722U.1-PO2	M	Knowledge acquired in the selected area of project development can be used to identify, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions.
101003/CS722U.1-PO3	M	Can use the acquired knowledge in designing solutions to complex problems.
101003/CS722U.1-PO4	M	Can use the acquired knowledge in designing solutions to complex problems.
101003/CS722U.1-PO5	H	Students are able to interpret, improve, and redefine technical aspects for design of experiments, analysis, and interpretation of data, and synthesis of the information to provide valid conclusions.
101003/CS722U.1-PO6	M	Students are able to interpret, improve, and redefine technical aspects by applying contextual knowledge to assess societal, health, and consequential responsibilities relevant to professional engineering practices.
101003/CS722U.1-PO7	M	Project development based on societal and environmental context solution identification is the need for sustainable development.
101003/CS722U.1-PO8	L	Project development should be based on professional ethics and responsibilities.
101003/CS722U.1-PO9	L	Project development using a systematic approach based on well-defined principles will result in teamwork.
101003/CS822U.1-PO10	M	Project brings technological changes in society.
101003/CS822U.1-PO11	H	Acquiring knowledge for project development gathers skills in design, analysis, development, and implementation of algorithms.

101003/CS822U.1-PO12	H	Knowledge for project development contributes engineering skills in computing and information gatherings.
101003/CS822U.2-PO1	H	Knowledge acquired for project development will also include systematic planning, developing, testing, and implementation in computer science solutions in various domains.
101003/CS822U.2-PO2	H	Project design and development using a systematic approach brings knowledge in mathematics and engineering fundamentals.
101003/CS822U.2-PO3	H	Identifying, formulating, and analyzing the project results in a systematic approach.
101003/CS822U.2-PO5	H	Systematic approach is the tip for solving complex problems in various domains.
101003/CS822U.2-PO6	H	Systematic approach in the technical and design aspects provides valid conclusions.
101003/CS822U.2-PO7	H	Systematic approach in the technical and design aspects demonstrates the knowledge of sustainable development.
101003/CS822U.2-PO8	M	Identification and justification of technical aspects of project development demonstrates the need for sustainable development.
101003/CS822U.2-PO9	H	Apply professional ethics and responsibilities in engineering practice of development.
101003/CS822U.2-PO11	H	Systematic approach also includes effective reporting and documentation, which gives clear instructions.
101003/CS822U.2-PO12	M	Project development using a systematic approach based on well-defined principles will result in better teamwork.
101003/CS822U.3-PO9	H	Project development as a team brings the ability to engage in independent and lifelong learning.

101003/CS822U.3-PO10	H	Identification, formulation, and justification in technical aspects will be based on acquiring skills in design and development of algorithms.
101003/CS822U.3-PO11	H	Identification, formulation, and justification in technical aspects provides the betterment of life in various domains.
101003/CS822U.3-PO12	H	Students are able to interpret, improve, and redefine technical aspects with mathematics, science, and engineering fundamentals for the solutions of complex problems.
101003/CS822U.4-PO5	H	Students are able to interpret, improve, and redefine technical aspects with identification, formulation, and analysis of complex problems.
101003/CS822U.4-PO8	H	Students are able to interpret, improve, and redefine technical aspects to meet the specified needs with appropriate consideration for public health and safety, and the cultural, societal, and environmental considerations.
101003/CS822U.4-PO9	H	Students are able to interpret, improve, and redefine technical aspects for design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.
101003/CS822U.4-PO10	H	Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools for better products.
101003/CS822U.4-PO11	M	Students are able to interpret, improve, and redefine technical aspects by applying contextual knowledge to assess societal, health, and consequential responsibilities relevant to professional engineering practices.
101003/CS822U.4-PO12	H	Students are able to interpret, improve, and redefine technical aspects for demonstrating the knowledge of, and need for sustainable development.

101003/CS822U.5-PO1	H	Students are able to interpret, improve, and redefine technical aspects, apply ethical principles, and commit to professional ethics and responsibilities and norms of the engineering practice.
101003/CS822U.5-PO2	M	Students are able to interpret, improve, and redefine technical aspects, communicate effectively on complex engineering activities with the engineering community and society at large, such as being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.
101003/CS822U.5-PO3	H	Students are able to interpret, improve, and redefine technical aspects to demonstrate knowledge and understanding of the engineering and management principle in multidisciplinary environments.
101003/CS822U.5-PO4	H	Students are able to interpret, improve, and redefine technical aspects, recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.
101003/CS822U.5-PO5	M	Students are able to interpret, improve, and redefine technical aspects in acquiring skills to design, analyze, and develop algorithms and implement those using high-level programming languages.
101003/CS822U.5-PO12	M	Students are able to interpret, improve, and redefine technical aspects and contribute their engineering skills in computing and information engineering domains like network design and administration, database design, and knowledge engineering.
101003/CS822U.6-PO5	M	Students are able to interpret, improve, and redefine technical aspects and develop strong skills in systematic planning, developing, testing, implementing, and providing IT solutions for different domains, which helps in the betterment of life.

101003/CS822U.6-PO8	H	Students will be able to associate with a team as an effective team player for the development of technical projects by applying the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.
101003/CS822U.6-PO9	H	Students will be able to associate with a team as an effective team player to identify, formulate, review research literature, and analyze complex engineering problems.
101003/CS822U.6-PO10	M	Students will be able to associate with a team as an effective team player for designing solutions to complex engineering problems and design system components.
101003/CS822U.6-PO11	M	Students will be able to associate with a team as an effective team player, use research-based knowledge and research methods including design of experiments, analysis, and interpretation of data.
101003/CS822U.6-PO12	H	Students will be able to associate with a team as an effective team player, applying ethical principles and committing to professional ethics and responsibilities and norms of the engineering practice.
101003/CS822U.1-PSO1	H	Students are able to develop Computer Science Specific Skills by modeling and solving problems.
101003/CS822U.2-PSO2	M	Developing products, processes or technologies for sustainable and socially relevant applications can promote Programming and Software Development Skills.
101003/CS822U.3-PSO3	H	Working in a team can result in the effective development of Professional Skills.
101003/CS822U.4-PSO3	H	Planning and scheduling can result in the effective development of Professional Skills.
101003/CS822U.5-PSO1	H	Students are able to develop Computer Science Specific Skills by creating innovative solutions to problems.

101003/CS822U.6-PSO3	H	Organizing and communicating technical and scientific findings can help in the effective development of Professional Skills..