

A PROJECT REPORT ON "JARVIS"

TABLE OF CONTENTS

1. Introduction to Proposed System
 - a. Problem Definition
 - b. System Overview
 - c. Project Functionalities with Module Specification
 - d. Operating Environment (H/W & S/W Requirement Specification)
2. Overview of the Proposed System
 - a. Proposed System
 - b. Objectives of the System
 - c. Feasibility Study
 - d. User Requirement Specification
3. System Analysis & Design
 - a. Data Flow Diagram (Context Level Diagram)
 - b. Class Diagram / ERD
 - c. Activity Diagram
 - d. Data Dictionary with Table Specification
 - e. Use Case Diagram
4. User Manual
 - a. Operational Instructions
 - b. Input/Output Screens
 - c. Reports
5. System Limitations
6. Future Enhancements and Conclusion
7. Bibliography and Glossary

Submitted by – Manish Jangid

1. INTRODUCTION TO PROPOSED SYSTEM

1.1 Problem Definition

Job seekers often struggle with interview anxiety and a lack of real-time feedback on their performance. Traditional interview preparation methods do not provide personalized feedback or AI-driven insights. *JARVIS* aims to bridge this gap by offering an AI-powered *mock interview* trainer that evaluates users' responses based on facial expressions, speech clarity, and answer relevance.

1.2 System Overview

JARVIS(Job AI-Ready Virtual Interview System)is a web-based AI-powered *Interview Training System* that assists users in preparing for real-world interviews by analyzing their facial expressions, tone of speech, and answer quality. It provides *real-time feedback* using AI-based evaluation models. The system is developed using *Next.js for the frontend* and *Gemini-2.0-Flash AI-based backend processing*.

Key Features:

- *Speech Recognition*: Converts spoken responses into text for evaluation.
- **AI Answer Analysis*: Uses *Gemini-2.0-Flash API* to evaluate responses.
- **Interactive UI*: Built using **Next.js* for a smooth user experience.
- **Database Management*: Utilizes **Drizzle ORM with PostgreSQL* for structured data storage.

1.3 Project Functionalities with Module Specification

Modules:

1. *User Authentication Module*: Users register and log in securely.
2. *Interview Question Module*: AI selects and asks questions dynamically.
3. **Speech-to-Text Conversion Module*: Uses **DeepSpeech* or *Google Speech-to-Text API*.
4. *Answer Evaluation Module*: AI analyzes responses for correctness and professionalism.
5. *Feedback & Report Generation Module*: Provides real-time feedback and generates performance reports.

1.4 Operating Environment (H/W & S/W Requirement Specification)

Hardware Requirements

- A system with at least 8 GB RAM and *quad-core processor*.
- WebCam for *facial analysis*.
- Microphone for *voice recording and speech analysis*.

Software Requirements

- *Frontend*: Next.js (React-based framework)
- *Backend*: Flask (for AI model integration)
- *Database*: PostgreSQL (via Drizzle ORM)
- *AI Libraries*: Gemini-2.0-Flash API
- *Operating System*: Windows / Linux

2. OVERVIEW OF THE PROPOSED SYSTEM

2.1 Proposed System

JARVIS offers a *real-time AI-driven interview assessment* tool that improves job seekers' confidence by providing instant feedback on *speech clarity, facial expressions, and answer quality*.

2.2 Objectives of the System

- *Enhance interview preparation* by simulating real interview environments.
- *Provide AI-powered insights* to improve users' verbal and non-verbal communication.
- *Reduce interview anxiety* with structured feedback.

2.3 Feasibility Study

Technical Feasibility:

- Uses well-established technologies (*Next.js, Clerk, Gemini API, PostgreSQL*).
- Cloud-ready for future scalability.

Operational Feasibility:

- Simple UI ensures *ease of use* for job seekers and HR professionals.
- Can be integrated into *corporate training programs*.

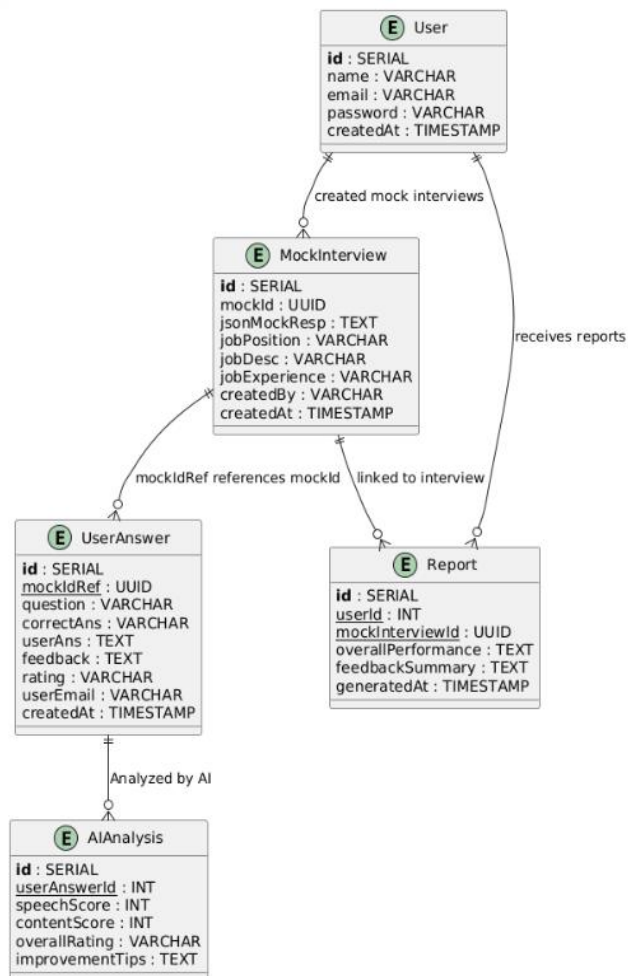
Economic Feasibility:

- Open-source technologies minimize development costs.
- Cloud hosting keeps infrastructure costs low.

3. SYSTEM ANALYSIS & DESIGN

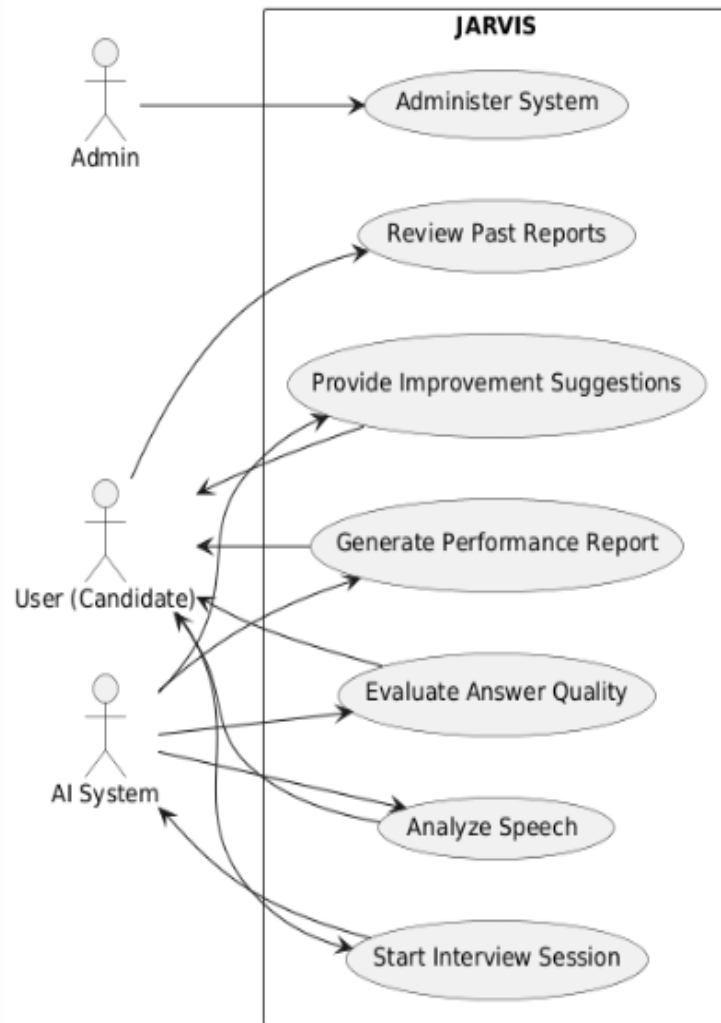
3.1 Entity Relationship Diagram (ERD)

- Users, Interview Questions, Responses, Evaluations are the key entities.



3.2 Use Case Diagram

- *Actors:* User, Admin
- *Use Cases:* Start Interview, Analyze Response, Generate Report



3.3 Database Design (Drizzle ORM)

1. MockInterview table

Field Name	Data Type	Description
id	SERIAL	Primary key, unique identifier for each mock interview session.
mockId	UUID	Randomly generated unique identifier for a mock interview.
jsonMockResp	TEXT	Stores JSON responses related to the mock interview.
jobPosition	VARCHAR	The job position for which the mock interview is conducted.
jobDesc	VARCHAR	Description of the job role.
jobExperience	VARCHAR	Required experience level for the job position.
createdBy	VARCHAR	The user/admin who created the mock interview session.
createdAt	VARCHAR	Timestamp indicating when the interview session was created.

2. UserAnswer Table

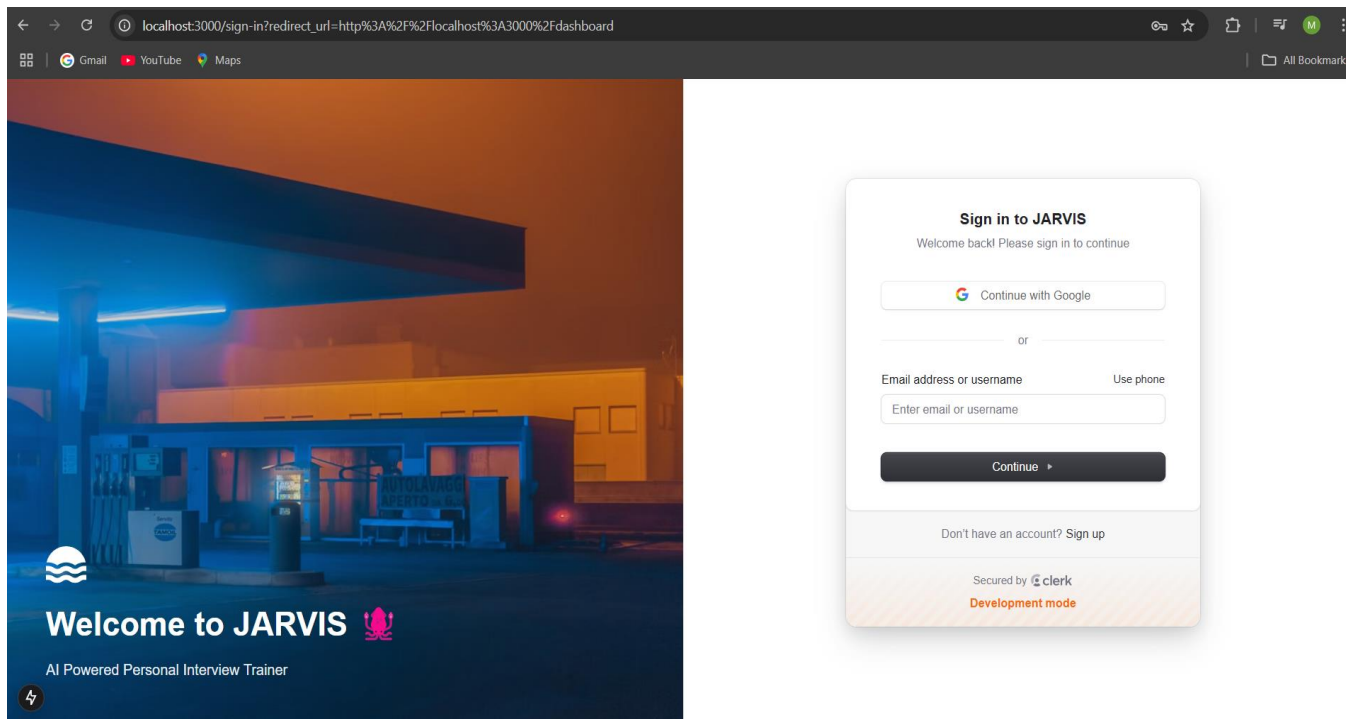
Field Name	Data Type	Description
id	SERIAL	Primary key, unique identifier for each answer record.
mockIdRef	VARCHAR	Foreign key reference to mockId in MockInterview table.
question	VARCHAR	The question asked during the mock interview.
correctAns	VARCHAR	The correct answer for the question (if applicable).
userAns	TEXT	The user's response to the question.
feedback	TEXT	AI-generated feedback on the user's answer.
rating	VARCHAR	AI-assigned rating for the answer's quality.
userEmail	VARCHAR	Email of the user who provided the answer.
createdAt	VARCHAR	Timestamp indicating when the answer was recorded.

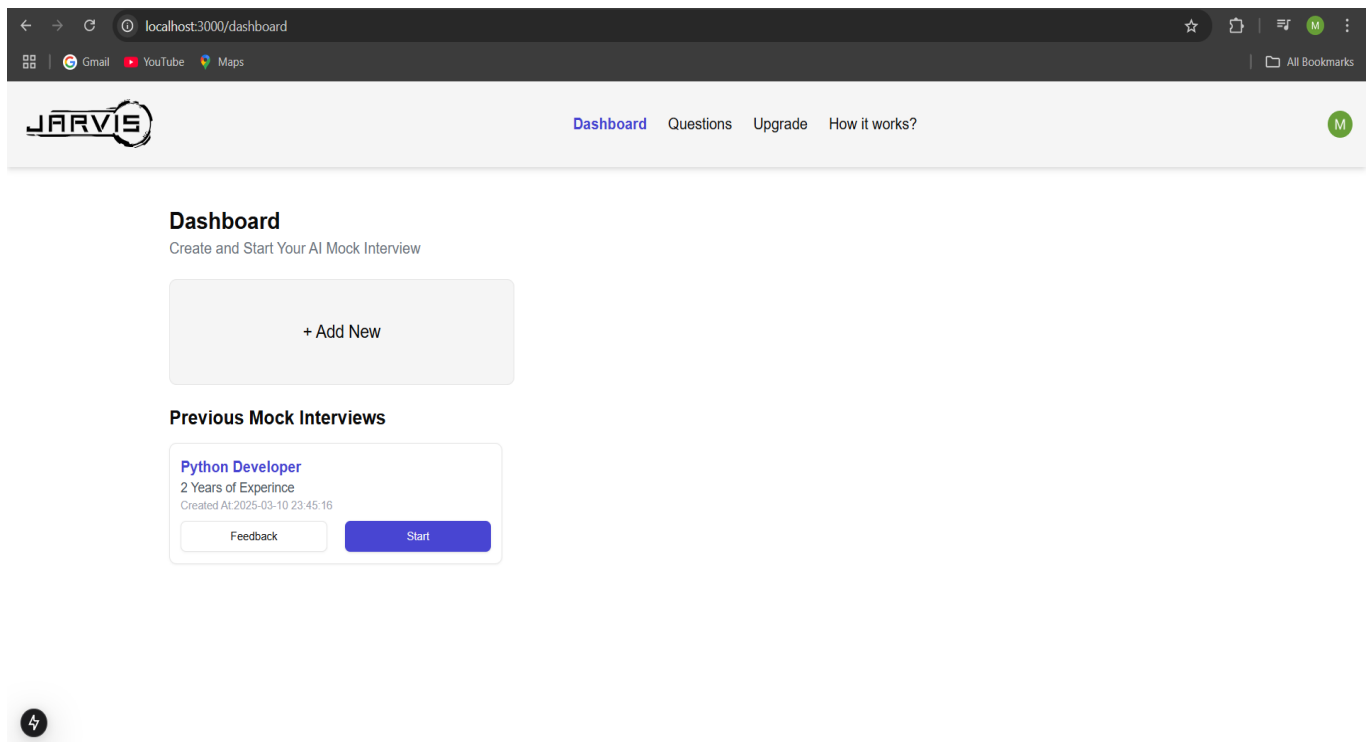
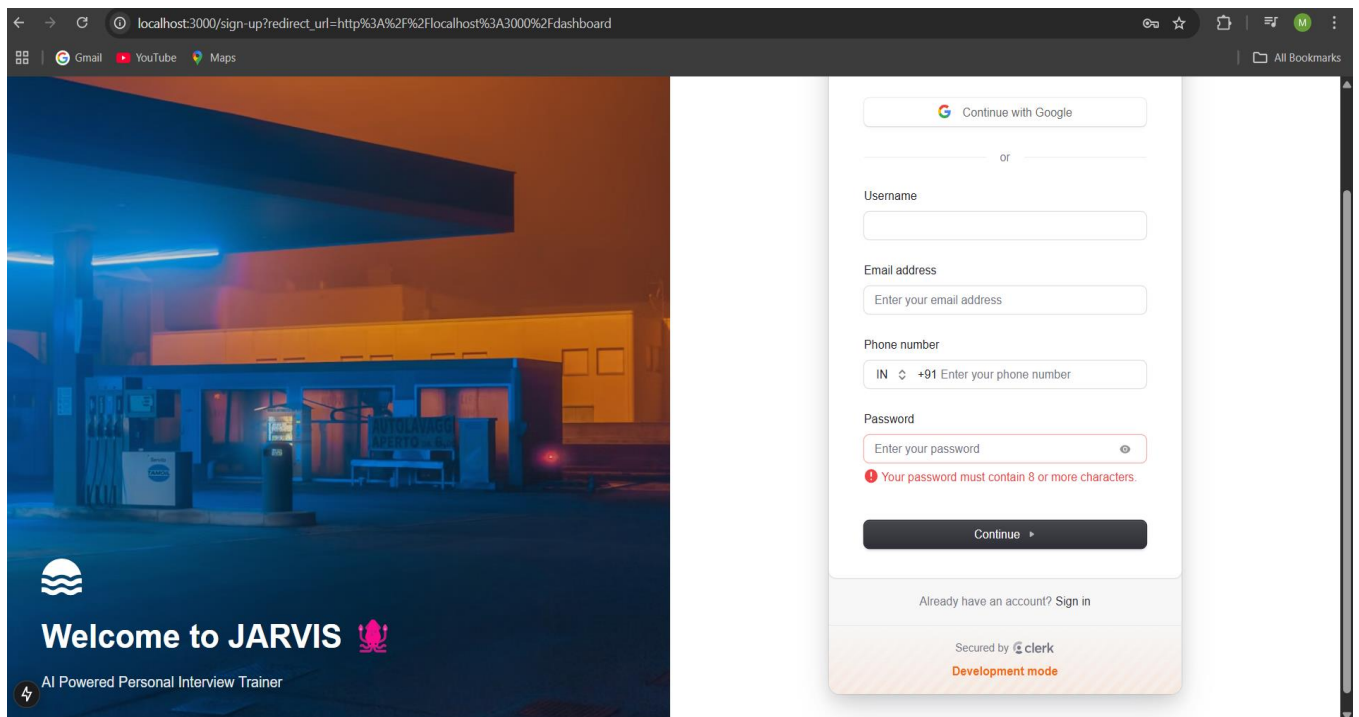
4. USER MANUAL

4.1 Operational Instructions

1. *Login* to the system.
2. Select “*Start Interview*”.
3. Answer AI-generated questions.
4. Receive *instant feedback* on speech clarity & expressions.
5. View *detailed performance reports*.

4.2 Input/Output Screens





Let's Get Started

Job Role/Job Position: Python Developer
Job Description/Tech Stack: Python, Flask, Django
Years of Experience: 2

Information

Enable Video Web Cam and Microphone to Start your AI Generated Mock Interview, It has 10 Questions which you can answer and at last you will get the report on the basis of your answers. NOTE: We never record your video, Web Cam access you can disable at any time if you want.



Enable Web Cam and Microphone

Start Interview

localhost:3000/dashboard/interview/60b2f4b4-4bff-42b5-82ec-ca98863f3ecf/start

JARVIS

Dashboard Questions Upgrade How it works?

Question #1 Question #2 Question #3 Question #4

Question #5 Question #6 Question #7 Question #8

Question #9 Question #10

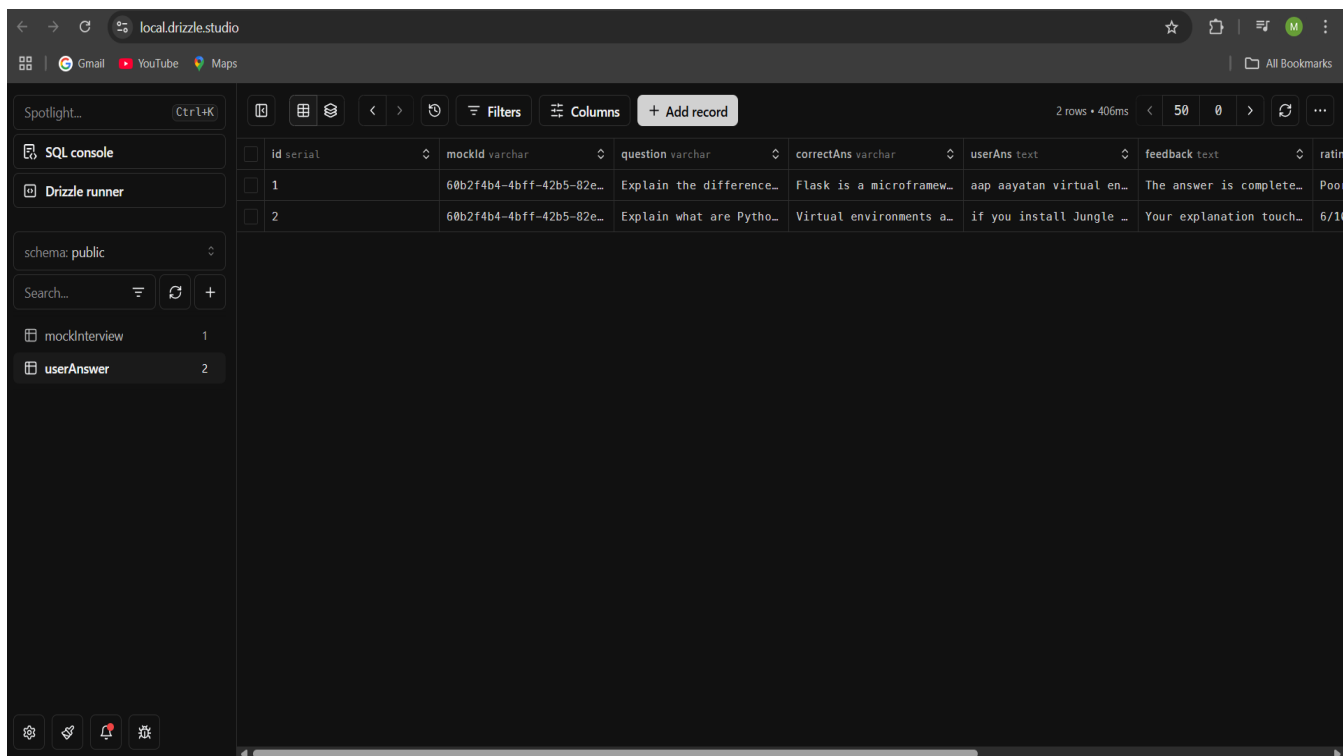
What is the purpose of the `__init__.py` file in Python packages?

Note:

Click on Record Answer when you want to answer the question. At the end of interview we will give you the feedback along with correct answer for each of question and your answer to compare it.

Record Answer

Previous Question Next Question



4.3 Reports

JARVIS provides detailed interview analysis reports that help users understand their strengths and weaknesses. These reports are generated based on AI analysis of the user's speech, facial expressions, and response quality.

❖ Parts of Report

1. Answer Quality Report - Analyzes the depth, relevance, and structure of responses.
2. Overall Performance Report - Provides a summary score and suggestions for improvement.
3. Detailed feedback and improvement suggestions
4. Actionable tips for enhancing interview performance

5. SYSTEM LIMITATIONS

Despite the robust AI-powered capabilities of JARVIS, certain limitations exist:

5.1 Limited Support for Non-English Languages

Currently, JARVIS primarily supports English-language interviews. While multilingual support is planned, accuracy for other languages is still limited.

5.2 Internet Dependency

JARVIS requires an active internet connection for real-time AI processing and report generation. Users with poor network connectivity may experience delays.

6. FUTURE ENHANCEMENTS

- *Real-time job recommendations based on performance.*
- *Integration with LinkedIn for skill endorsement.*
- *Multi-language support for non-English speakers.*

7. CONCLUSION

JARVIS provides a *next-gen AI-driven interview coaching experience*, helping users improve communication skills and interview readiness.

8. BIBLIOGRAPHY

- NextJs Documentation
- Gemini API for Answer Evaluation
- IEEE Standards for Software Development