GenAI Virtual Interviewer – Proof of Concept (POC)

Submitted by: Ishu Meena

Branch: Electronics & Communication Engineering

College: National Institute of Technology, Hamirpur

Project Type: Technical Documentation for Chapter Apps Inc

Problem and Our Idea

In today's time, many companies take online interviews. But it takes a lot of time and people to ask questions, check answers, and give results.

So, the idea is to make a chatbot (a smart computer program) that can do this job. It should talk like a real interviewer and ask simple or technical questions based on what the person says.

This will help:

- Save the HR team's time.

- Give the same experience to every candidate.

- Check answers and give feedback immediately.

We will use AI (Artificial Intelligence) to make the chatbot smart so it can talk and understand like a human.

Tools and Technologies (Tech Stack)

To make this project, we will use some simple tools and platforms. These tools help us build the chatbot and make it work properly.

- ChatGPT or any AI model
- This is the brain of our chatbot.
- It will understand what the person is saying and give smart replies.

1. LangChain (optional)

- Helps connect the AI with files or memory.
- Not required, but good if we want to improve the chatbot later.

2. Python or JavaScript

- These are easy languages to make a chatbot.
- We can choose one to write the code (not needed now, just for planning).

3. PDF Reader Tool (like PyMuPDF)

- If we want to take questions or data from a file (like a resume), this tool can help us read it.

4. Google Docs or Word

- Just for writing this document

Our Approach (Step-by-Step Plan)

Let me explain how we plan to make the interview chatbot. We'll break it down into simple steps so it's easy to understand.

Step 1: Getting Information from the User

- The user (student or job applicant) will give their name, resume, and job role.
- This helps the chatbot understand who it is talking to.

Step 2: Reading the Resume (Optional)

- If we want, we can let the chatbot read the resume using a PDF reader.
- This helps the bot ask questions based on the user's projects, skills, and experience.

Step 3: Starting the Interview

- The chatbot starts the interview by greeting the user.
- It asks simple questions like "Tell me about yourself" or "What do you know about this job?"

Step 4: Smart Follow-Up Questions

- Based on the user's answers, the chatbot asks more questions.
- It remembers what the user said and makes the next question fit well.

Step 5: Giving Feedback

- After 4–5 questions, the chatbot gives a simple review.
- It may say things like "Good communication" or "Try to be more clear."

Step 6: Saving the Interview (Optional)

- We can save the answers and chatbot feedback for the user to see later.

This is our plan for how the chatbot will work from start to finish in a very simple way.

Why We Chose This Approach (Our Reasoning)

We wanted to keep things simple, useful, and beginner-friendly. That's why we chose this step-by-step plan. Let me explain the reasons behind it:

1. Easy for Students and Freshers:
- Many students are nervous about interviews.
- Our chatbot gives them a chance to practice in a friendly way.
- It doesn't judge or rush them — just gives simple questions and feedback.

2. No Need for Big Setup:

- We didn't choose very heavy or complex tools.
- This chatbot idea can work even without building a full website.
- It's easy to test using simple tools like a Google form or chat page.

3. It Feels Like a Real Interview:
- The chatbot asks real questions like an HR or technical interviewer would.
- It adjusts questions based on the candidate's answers — so it feels natural, not robotic.

4. Can Be Improved Later:
- Even if we start small, this idea can be made better step by step.
- We can later add voice, video, or more question types if needed.

5. Focus is on Learning:
- The goal is not to judge someone, but to help them grow.
- The feedback tells users what they did well and where they can improve.

So, this plan is simple, helpful, and easy to build — and that's why we chose it.

It gives good value without needing a full app or website in the beginning.

Technologies We Will Use (Tech Stack)

We are keeping the technology very simple and easy to understand. These are the tools we will use:

- ➢ **Chatbot Base – ChatGPT (or any LLM)**
  - This is the main brain of our chatbot.
  - It understands what the user says and replies in a smart way.
  - We don't have to build our own AI — we can just use ChatGPT or any similar tool.

- ➢ **PDF Reader Tool (for Resume Reading)**
  - Sometimes, users upload resumes in PDF.
  - We use free tools like pdfplumber or PyMuPDF to read the text from that PDF.
  - This helps the chatbot understand more about the person.

- ➢ **LangChain (optional helper tool)**
  - This tool helps connect our AI to the resume or job description.
  - It makes sure the chatbot can ask questions related to the uploaded document.

- ➢ **Feedback System (also using AI)**
  - After the interview, the same AI gives a summary of how the candidate did.

- It checks if answers were clear, correct, and confident.

- ➢ **Basic User Interface (if needed)**
- We can use a simple webpage or form for users to start the interview.
- This is optional — not needed if we're just testing.

So, the tech stack is:

- ChatGPT or similar AI
- PDF reader tool
- LangChain (optional)
- Basic UI (optional)

It's all simple and easy to use even if you are not from a coding background.

How It Will Work (Step-by-Step Flow)

This page explains how the whole idea works, step by step — like a small story.

Step 1: Resume Upload

- The user uploads their resume as a PDF file.
- The system reads the text from this file using a simple PDF reading tool.

Step 2: Preparing Questions

- The AI reads the resume and prepares smart interview questions.

- These questions are related to the candidate's skills, education, and projects.
- If no resume is uploaded, it can still ask general questions.

Step 3: Interview Begins

- The AI (like ChatGPT) starts asking one question at a time.
- The candidate answers by typing or speaking.
- The AI listens and then asks the next question based on the previous answer.

Step 4: Judging Answers

- The AI checks every answer for clarity, correctness, and confidence.
- It gives a small score or comment after each question (optional).

Step 5: Final Feedback

- After the interview, the AI gives feedback:
- What the candidate did well
- Where they can improve
- A small summary of the overall performance

This process helps a candidate feel like they had a real interview. It also gives helpful tips to improve.

Where This Can Be Used (Real-Life Examples)

This page shows where and how the AI Interviewer can help in real life. Let's keep it super simple:

1. College Students – Mock Interviews

- Students can use this tool to practice before real placement interviews.
- It helps them get used to answering questions and feel confident.

2. Coaching Centers

- Teachers or trainers can use it to help students prepare.
- After every session, students get feedback to improve.

3. Companies – First Round of Interviews

- HR teams can use this for the first screening of candidates.
- It saves time by selecting only the best candidates for the next round.

4. Resume Checking

- The AI can read the resume and ask questions based on that.
- This helps in making the interview more personal and job-related.

5. Personal Use

- Anyone can use it at home to get better at interviews.
- It's a private and safe way to practice.

These simple examples show how useful this tool can be for students, teachers, companies, or individuals.

Tools & Technologies Used

Let's now explain the tools in the simplest way so anyone can understand:

1. ChatGPT or Similar AI

- This is the main brain of the interview tool.
- It can read, understand, and talk like a real person.
- Example: Like how ChatGPT is talking to you right now.

2. Resume Reader (PDF Tool)

- It helps the AI read resumes or files you upload.
- It finds skills, projects, marks, etc., from PDF files.

3. LangChain (Just a Helper)

- It helps the AI remember things you said earlier.
- Also helps decide what question to ask next.

4. RAG (Search and Help Method)

- AI doesn't know everything.
- This tool helps it search your uploaded files (like a resume or job info) before answering.

5. Judging Tool (Same AI)

- The same AI can also check your answers.
- It looks at your reply and gives a score (like good, okay, or needs improvement).

6. Optional: Voice Tools

- If we want to make it talk like a human, we can add voice features later (optional).

These are all the tools. Don't worry about using them right now — just explain what they do. This is enough for a POC (proof of concept).

Why I Chose This Approach

I wanted to build something that students like me could actually use. Interviews are scary, especially the first few. I thought, "What if I had a bot that talks to me like an HR person and tells me how I did?"

That's when I got the idea of using ChatGPT for this. It already talks smartly and can be trained to act like an interviewer.

This way:

- We can practice from home
- Get feedback instantly
- And feel more confident before the real interview

I kept everything simple so that anyone — even someone who doesn't know coding — can understand the plan.

What I Learned from This Project

What I Learned

- I learned how an AI can act like a human and help in interviews.
- I now know how we can give AI some info (like a resume) and make it ask related questions.
- I also learned how important it is to keep things simple when explaining a project.
- Most importantly, I learned that a small idea can solve real problems.

Final Summary

This document explains a small but useful idea — an AI chatbot that can take mock interviews.

It can:

- Ask questions
- Check answers
- Give feedback

Save time for both students and companies

I used very basic tools and kept the idea simple and easy to understand. This is just a start. In the future, it can be made better with voice, video, and more question types.

Thank you for reading!