VIT
Vellore Institute of Technology
(Deemed to be University under section 3 of UGC Act, 1956)

AI Based Chatbot to Answer FAQs

Team Name:J04

Team Mentor: Dr. Shynu P G

Date: 20/05/2023
Time: 16:00

Pratyaksh Saxena	22MCA0168
Shivam Singh	22MCA0257
Oeyshik Das	22MCA0263
Priyanshi Singhal	22MCA0248
Bhavyanshi Jain	22MCA0130
Palak Chawla	22MCA0141

Problem Statement

Domain: Artificial Intelligence and Android

Problem Statement

Absence of official Android application for ChatGPT. However, There is recently an ios application introduced.

My Idea

- AI-based chatbots can overcome this limitation by using natural language processing (NLP) to understand and answer questions.
- NLP is a field of computer science that deals with the interaction between computers and human language.
- We interacted with the Open AI's extensive language model via the official Open AI API.

Benefits

- Natural Language Processing (NLP): AI chatbots utilize NLP algorithms to understand and interpret user inputs. This enables them to process and respond to natural language queries, making interactions with users more conversational and intuitive.
- Machine Learning: AI chatbots can employ machine learning techniques to continuously improve their performance. They can learn from user interactions, feedback, and data, allowing them to adapt and provide more accurate and relevant responses over time.
- Automated Task Execution: Chatbots can automate routine tasks and processes, such as answering frequently asked questions, providing product information, or routing inquiries to the appropriate department. This automation saves time and effort for both users and businesses.
- Integration with Systems and APIs: AI chatbots can integrate with various systems and APIs, such as customer relationship management (CRM) platforms, knowledge bases, or e-commerce platforms. This integration allows chatbots to access relevant data and provide real-time information to users.

- Contextual Understanding: AI chatbots can maintain context during conversations, remembering previous interactions and user preferences. This contextual understanding enables them to provide personalized responses and recommendations based on the user's history or specific needs.
- Sentiment Analysis: AI chatbots can employ sentiment analysis techniques to understand the emotional tone and intent behind user messages. This allows them to gauge customer satisfaction, detect potential issues, and respond accordingly, providing a more empathetic and tailored experience.
- Chatbot Analytics: AI chatbots often come with built-in analytics capabilities. They can track and analyze user interactions, including user behavior, frequently asked questions, and conversion rates. These insights help businesses identify trends, measure performance, and make data-driven improvements.
- Channel Consistency: AI chatbots can be deployed across multiple channels, such as websites, mobile apps, messaging platforms, or voice assistants. This ensures consistent and unified customer experiences across different touchpoints, improving brand perception and engagement.
- Security and Compliance: AI chatbots can be designed with security measures in mind, ensuring the protection of sensitive user data. They can also be programmed to adhere to specific industry regulations and compliance standards, reducing the risk of data breaches or legal issues.

Algorithm

Step 1: Project Setup and Configuration

1. Set up an Android development environment with Java.
2. Create a new Android project in your preferred IDE.
3. Configure the necessary dependencies and libraries for HTTP requests and JSON parsing.

Step 2: User Interface Design

1. Design and implement the ChatBot user interface (UI) with appropriate components such as EditText, RecyclerView, and a send button.
2. Implement the necessary UI logic to display messages in the chat interface.

Step 3: OpenAI API Integration

1. Sign up for the OpenAI API and obtain the necessary API key or authentication credentials.
2. Configure the HTTP client in your Android project to communicate with the OpenAI API endpoints.
3. Implement the necessary functions to make HTTP requests to the OpenAI API for text generation.

Step 4: User Input Processing

1. Retrieve user input from the chat interface.
2. Preprocess the user input if necessary (e.g., removing special characters or formatting).
3. Prepare the input data in a format suitable for the OpenAI API (e.g., JSON payload).

Step 5: Communicating with the OpenAI API

1. Send the user input data as a request to the OpenAI API endpoint.
2. Handle the API response, extracting the generated text or message from the API response payload.

Step 6: Displaying Chatbot Responses

1. Update the chat interface with the chatbot response received from the OpenAI API.
2. Implement the necessary UI logic to display the chatbot's messages in the chat interface.

Step 7: Iterative Conversations

1. Implement a mechanism to allow for iterative conversations with the chatbot.
2. Store previous user inputs and chatbot responses to maintain context and continuity in the conversation.

Step 8: Error Handling and Edge Cases

1. Implement appropriate error handling for API communication failures or invalid responses.
2. Consider edge cases such as empty user input or unexpected API responses and handle them gracefully.

Step 9: Testing and Debugging

1. Test the chatbot's functionality on various devices and Android versions.
2. Debug and address any issues or inconsistencies.

Step 10: Deployment and Continuous Improvement

1. Prepare the app for deployment by generating a signed APK.
2. Publish the app to the Google Play Store or distribute it through other channels.
3. Gather user feedback and monitor the app's performance to make iterative improvements

Prerequisite

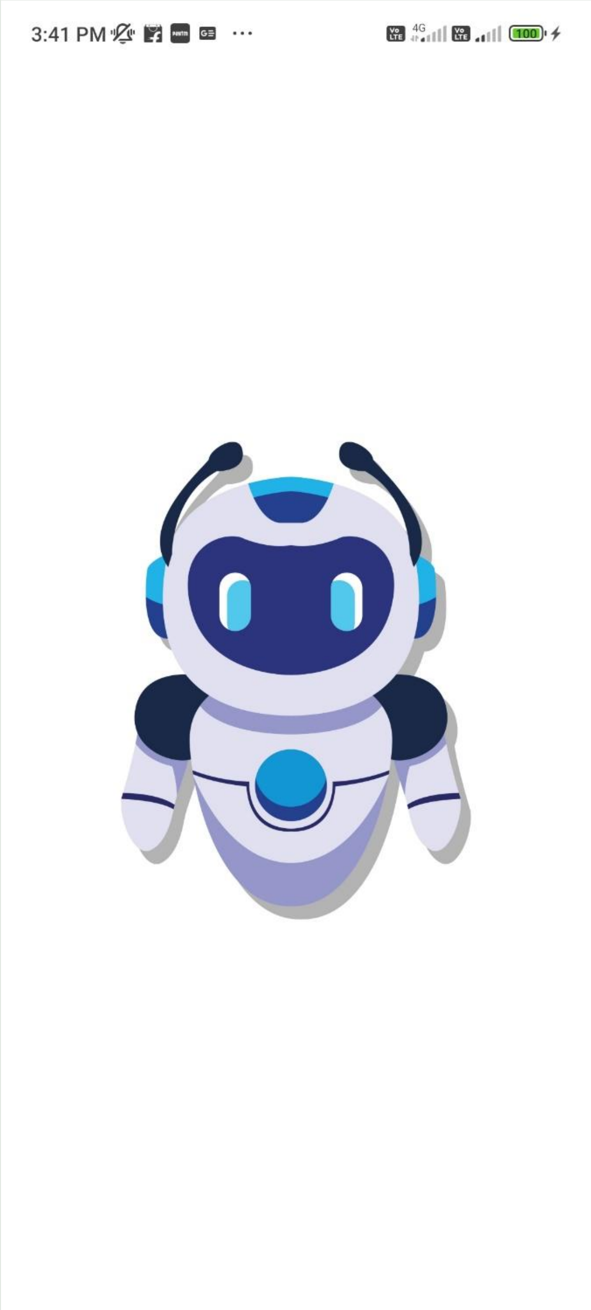
To build an AI Chatbot application on Android platform using Java, the following skills are typically required:

- ✓ **Android Development:** Proficiency in Android app development using Java or Kotlin, including knowledge of UI design, activities, fragments, RecyclerView, and handling user input.
- ✓ **Java Programming:** Strong understanding of Java programming language, including object-oriented programming concepts, data structures, and algorithms.
- ✓ **HTTP and API Integration:** Familiarity with making HTTP requests and handling API integrations in Android using libraries like Retrofit, OkHttp, or Volley.
- ✓ **JSON Parsing:** Knowledge of parsing and manipulating JSON data in Android applications using libraries like Gson or JSONObject.
- ✓ **OpenAI API Knowledge:** Understanding of the OpenAI API and its features, including authentication, request formats, and response handling.

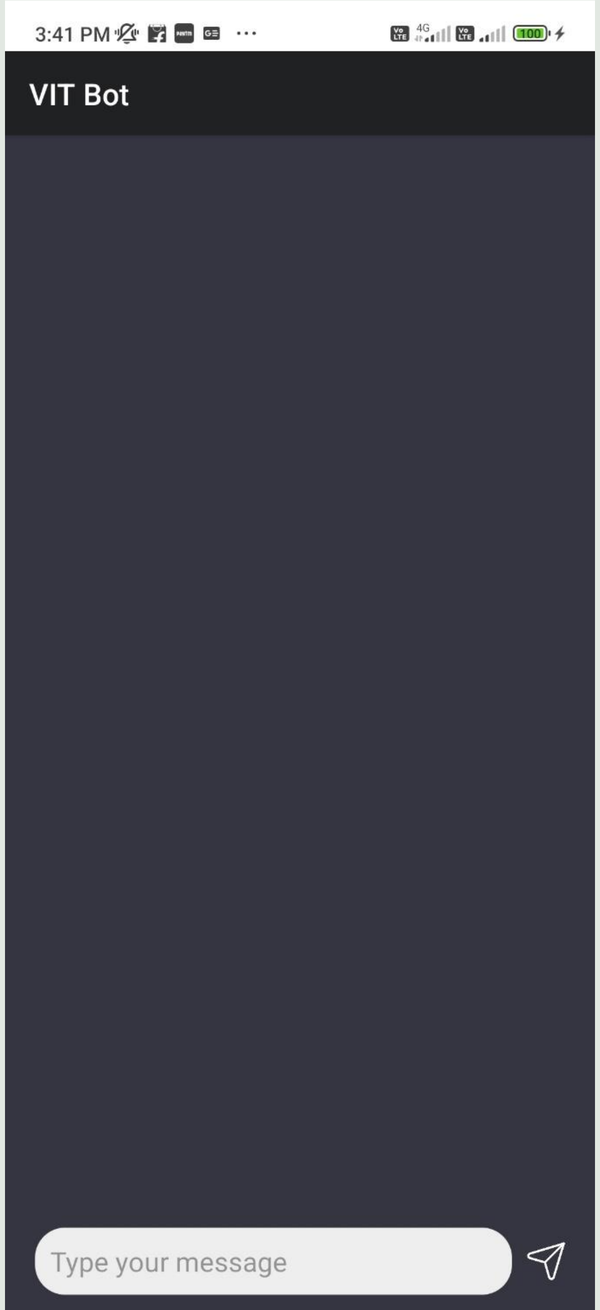
- ✓ **User Interface Design:** Ability to design and implement user interfaces using XML layouts, UI components, and appropriate UX principles for a chat-based application.
- ✓ **Data Processing and Manipulation:** Skills in processing and manipulating user input and API responses, including text preprocessing, data formatting, and context handling.
- ✓ **Error Handling and Debugging:** Proficiency in identifying and resolving errors, handling exceptions, and implementing proper error handling mechanisms for network requests and API responses.
- ✓ **Testing and Debugging:** Knowledge of testing methodologies, unit testing, and debugging techniques in Android development using tools like Android Studio and adb.
- ✓ **Deployment and Publishing:** Understanding of the process of generating signed APKs, publishing apps on the Google Play Store, and managing app releases.

Demo

Splash Screen

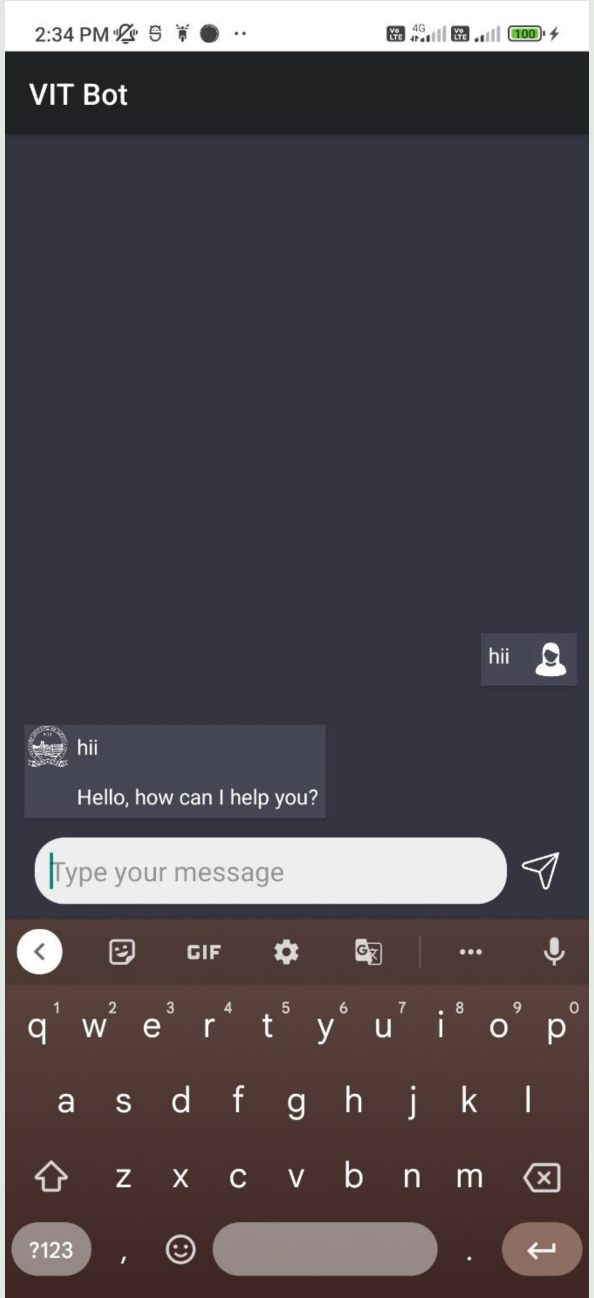


Home Screen



Sample Screens

1

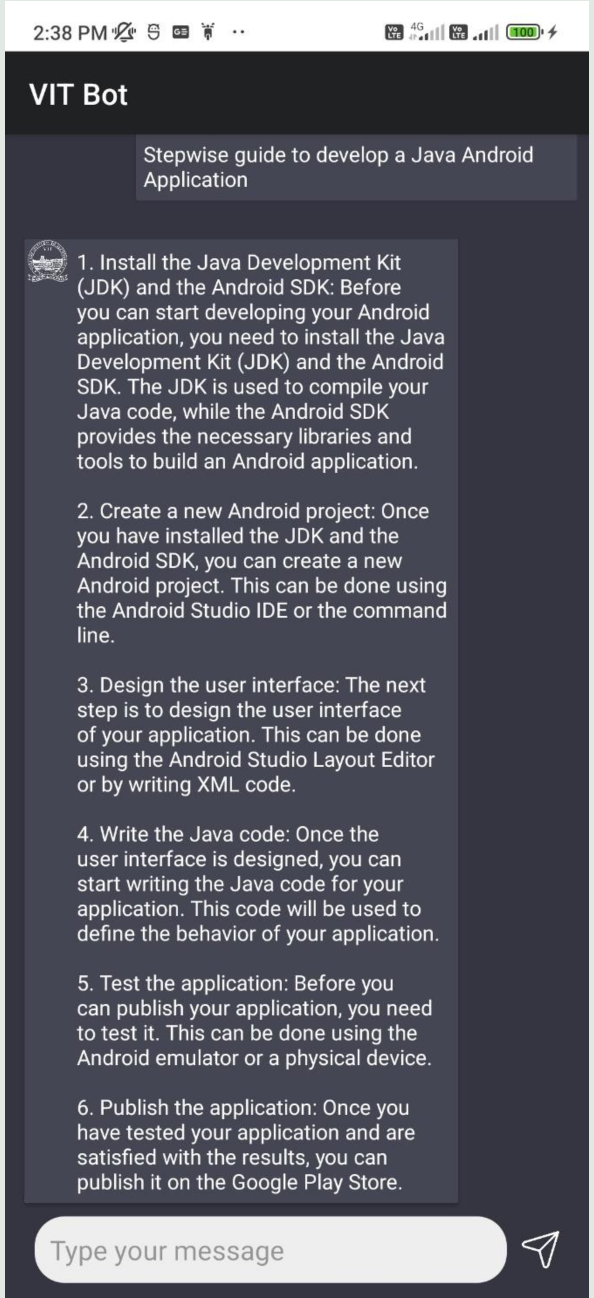


hii

hii

Hello, how can I help you?

2



Stepwise guide to develop a Java Android Application

1. Install the Java Development Kit (JDK) and the Android SDK: Before you can start developing your Android application, you need to install the Java Development Kit (JDK) and the Android SDK. The JDK is used to compile your Java code, while the Android SDK provides the necessary libraries and tools to build an Android application.

2. Create a new Android project: Once you have installed the JDK and the Android SDK, you can create a new Android project. This can be done using the Android Studio IDE or the command line.

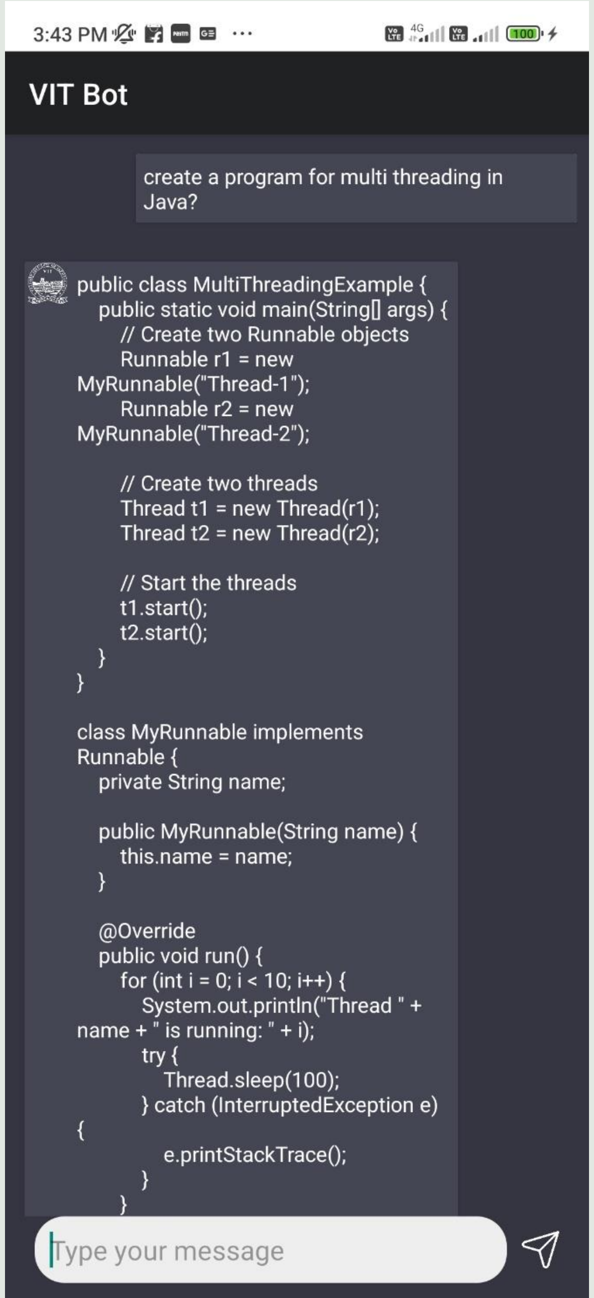
3. Design the user interface: The next step is to design the user interface of your application. This can be done using the Android Studio Layout Editor or by writing XML code.

4. Write the Java code: Once the user interface is designed, you can start writing the Java code for your application. This code will be used to define the behavior of your application.

5. Test the application: Before you can publish your application, you need to test it. This can be done using the Android emulator or a physical device.

6. Publish the application: Once you have tested your application and are satisfied with the results, you can publish it on the Google Play Store.

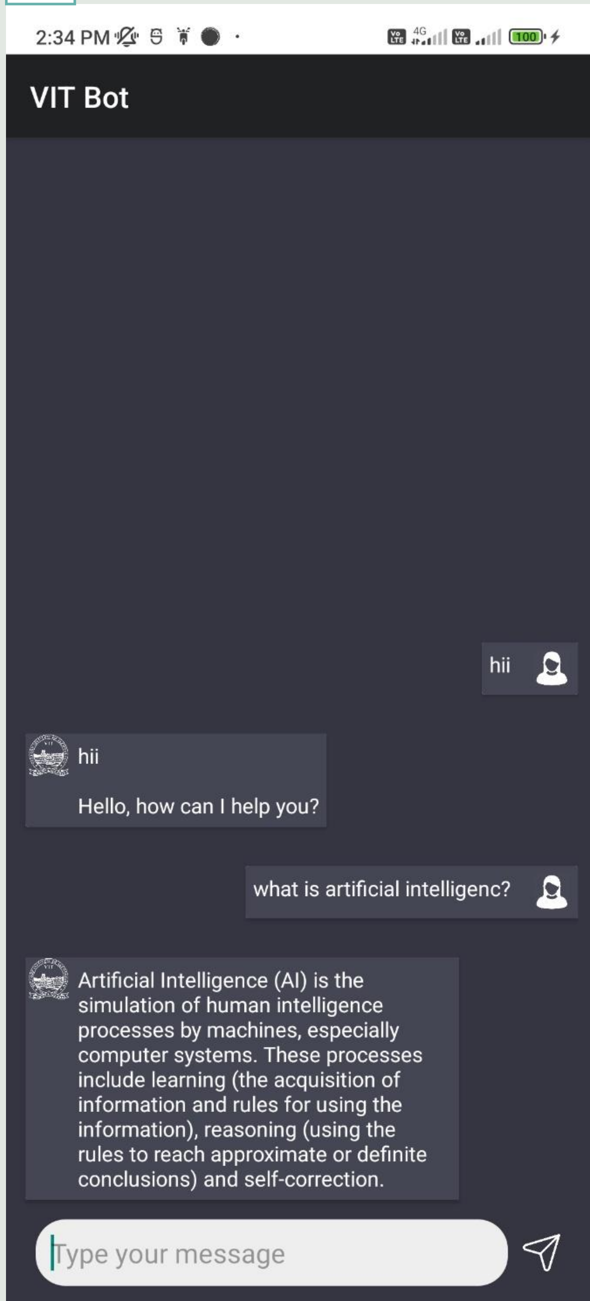
3



create a program for multi threading in Java?

```
public class MultiThreadingExample {
   public static void main(String[] args) {
      // Create two Runnable objects
      Runnable r1 = new
MyRunnable("Thread-1");
      Runnable r2 = new
MyRunnable("Thread-2");

      // Create two threads
      Thread t1 = new Thread(r1);
      Thread t2 = new Thread(r2);

      // Start the threads
      t1.start();
      t2.start();
   }
}

class MyRunnable implements
Runnable {
   private String name;

   public MyRunnable(String name) {
      this.name = name;
   }

   @Override
   public void run() {
      for (int i = 0; i < 10; i++) {
         System.out.println("Thread " +
name + " is running: " + i);
         try {
            Thread.sleep(100);
         } catch (InterruptedException e)
{
            e.printStackTrace();
         }
      }
```

4



hii

hii

Hello, how can I help you?

what is artificial intelligenc?

Artificial Intelligence (AI) is the simulation of human intelligence processes by machines, especially computer systems. These processes include learning (the acquisition of information and rules for using the information), reasoning (using the rules to reach approximate or definite conclusions) and self-correction.

THANK YOU