

**Shri Vaishnav Vidyapeeth
Vishwavidyalaya, Indore (M.P.)
Department of Computer Science Engineering**

Assignment 1



Semester: VIII

Subject Code: BTCS801

Roll No.: 21100BTCSE10020

Section: CS-D

Submitted By:

Submitted to:

Utkarsh Pandey

Prof. Shuchi Mandhanya

Shri Vaishnav Vidyapeeth Vishwavidyalaya, Indore
Shri Vaishnav Institute of Information Technology
Computer Science and Engineering
Internship/Project (BTCS801) - Fourth Year - Eight Semester

ASSIGNMENT- 2

Questions -

1. Explain your project title.

Project Title: Gmail Agent Agent (Relevance AI)

The Gmail Agent is an AI-powered virtual assistant built with Relevance AI, designed to automate and streamline Gmail inbox management. It intelligently reads, summarizes, and categorizes incoming emails, detects intent (e.g., meeting requests, support issues, sales inquiries), and suggests or sends auto-replies using predefined workflows.

2. Explain the different technologies you explored during the initial phase for your project.

1. Relevance AI

- **Why explored:** To leverage no-code/low-code workflows, AI models, and vector search capabilities.
- **Use Case:** Classifying and clustering emails, detecting intent, and generating smart replies using prompt workflows.

2. Gmail API (Google Workspace)

- **Why explored:** To securely access, read, send, and manage emails.
- **Use Case:** Authenticating users, fetching real-time email data, and sending automated replies.

3. OpenAI / GPT Models

- **Why explored:** To enhance natural language understanding and generate humanlike responses.
- **Use Case:** Summarizing emails, identifying tone, and generating personalized replies.

4. Zapier & Make (Integromat)

- **Why explored:** For workflow automation and connecting Gmail to Relevance AI without heavy backend coding.

- **Use Case:** Automating the flow from Gmail to Relevance AI, triggering workflows based on email events.

3. Explain the basics of your finalized technology.

1. Relevance AI (Core Platform)

- **Purpose:** Acts as the brain of the agent.
- **What it does:**
 - Uses **no-code workflows** to automate email classification and response.
 - Applies **pre-trained AI models** to detect intent, sentiment, and urgency.
 - Can **generate replies** or perform actions based on email content.

2. Gmail API

- **Purpose:** Connects the agent to the user's Gmail.
- **What it does:**
 - Fetches incoming emails.
 - Allows reading subject, body, sender info.
 - Enables sending smart replies generated by the AI.
 - Uses OAuth 2.0 for secure access.

3. Automation Platform (e.g., Zapier / Make) •

- Purpose:** Bridges Gmail with Relevance AI.
- **What it does:**
 - Watches for new emails in Gmail. ○ Sends email data to Relevance AI for processing. ○ Triggers reply actions or workflows automatically.
 - **No coding needed**, easy to set up and scale.

4. GPT (OpenAI or Relevance LLMs)

- **Purpose:** Enhances replies with natural language generation.
- **What it does:**
 - Summarizes long emails.
 - Writes professional and human-like responses. ○ Customizes tone based on user preferences

4. Summarize your one – month work on your project.

Over the past month, I worked on designing, building, and testing **four intelligent AI agents** using **Relevance AI's no-code platform**. Each agent was built for a specific use

case to automate tasks, improve efficiency, and demonstrate practical applications of generative AI and automation.

1.Sales Note Taker Agent

- **Goal:** Automatically summarize sales call transcripts.
 - **Tools:** Relevance AI workflows, transcript upload trigger, LLM summarization.
 - **Progress:** Built a flow that identifies key moments, pain points, and next steps from calls, and saves them into a CRM-friendly format.
-

2.Customer Support Agent

- **Goal:** Auto-categorize and respond to common support emails or tickets.
 - **Tools:** Relevance AI + Gmail API + Prebuilt intents.
 - **Progress:** Trained the agent to detect issue types, sentiment, urgency, and generate helpful replies with minimal human input.
-

3.Tabby – File Converter Agent

- **Goal:** Convert files (PDFs, Word, etc.) to desired formats and summarize content.
 - **Tools:** Relevance AI + Upload trigger + file parser + LLM summarizer.
 - **Progress:** Created an agent that accepts uploaded documents, converts and summarizes them, and returns the output via email or downloadable link.
-

4.LinkedIn Agent

- **Goal:** Generate personalized LinkedIn messages for leads or connections.
 - **Tools:** Relevance AI + CSV uploader + prompt templates.
 - **Progress:** Developed a campaign-ready workflow that generates context-aware messages for outreach based on user data (name, industry, role).
-

Overall Outcome

- Explored no-code AI workflow design.
- Gained hands-on experience in using **Relevance AI's automation tools, LLMs, and API integrations.**

- Built practical solutions for **sales, support, document processing, and lead generation.**
- Improved speed, personalization, and productivity in each use case.

5. Discuss what problem you faced during the last month when you worked on.

1.API Limitations & Restrictions

Faced rate limits and authentication issues with Amazon Wandor API.

2.Airtable Sync Issues

Delay in data updates and occasional syncing failures.

3.Webhook Delays

Some webhooks took time to trigger due to server response issues.

4.Gmail Invoice Automation

Parsing invoice data accurately from different email formats was challenging.

5.Zapier Task Limits

Free-tier task limits restricted automation execution.

6.Social Media Posting Errors

Some platforms required additional API permissions for auto-posting.

7.Make.com Database Complexity

Managing large data sets efficiently required optimization.

6. Add one question about the delivering or future use

1. Enhanced AI Integration – Using AI (ChatGPT, Google Gemini) for intelligent decisionmaking in emails, invoices, and campaign management.

2. Scalable Infrastructure – Migrating to cloud-based databases (e.g., Airtable, Firebase) for handling large-scale data efficiently.

3. Advanced API Handling – Implementing better error handling, caching, and optimizing API calls to avoid rate limits.