

Hackathon Project Phases Template

Project Title:

AI Personalized Email Generator

Team Name:

Team LXII

Team Members:

- Venkat Reddy Bitla
 - Shiva Teja Malkapuram
 - Umesh Chandra Papishetty
 - Uday Chaithanya Varaganty
-

Phase-1: Brainstorming & Ideation

Objective:

To develop a Django-based AI-powered email generator that reads recipient details from a CSV file, generates personalized emails using Hugging Face models, and sends both bulk and individual AI-generated emails via SMTP.

Problem Statement:

- Writing and personalizing emails is time-consuming.
- Ensuring consistency while addressing individual preferences is challenging.
- Manual email formatting can lead to errors.

Proposed Solution:

- A Django web application that reads recipient details from a CSV file or user input.
- Uses Hugging Face AI models to generate email content based on the provided tone and template.
- Sends both general-purpose emails and bulk emails via SMTP (Gmail/Outlook).

Supported Email Templates:

1. Professional email applying for a Software Engineer position.
2. Formal email proposing a business collaboration.
3. Portfolio follow-up email to a client about a previous discussion.
4. Product promotion email encouraging the recipient to buy.
5. Webinar invitation email urging people to register.
6. Friendly email asking users to sign up for newsletters.
7. Persuasive email urging people to register for an upcoming event.
8. Free trial offer email urging the recipient to sign up.
9. Professional and empathetic rejection email to a job candidate, appreciating their time and effort.

Target Users:

- Individuals looking to send personalized job applications.
- Organizations for marketing and outreach campaigns.
- HR teams managing mass communication.
- Businesses promoting their services and events.

Expected Outcome:

- A functional web-based application that generates and sends both personal and bulk AI-generated emails automatically.

Phase-2: Requirement Analysis

Technical Requirements:

- **Backend:** Django (Python-based web framework).
- **AI Model:** Hugging Face NLP models (Mistral-7B).
- **Frontend:** Django HTML templates.
- **Database:** SQLite (Django model handling).
- **Email Sending:** SMTP (configured in Django settings).

Functional Requirements:

- Read **recipient details, email tone, and template** from a **CSV file** or user input.
- Generate **customized email content** using **Hugging Face models**.
- Send **both personal and bulk emails** using **SMTP**.

Challenges:

- **Handling API rate limits** for Hugging Face models.
- **Ensuring emails don't go to spam** (SPF/DKIM settings).
- **Proper email formatting** for different email clients.

Phase-3: Project Design

System Architecture:

1. User **uploads a CSV file** or enters details manually.
2. Django **processes the input data** and extracts details.
3. The **Hugging Face AI model** generates **personalized email content**.
4. Emails are sent via an **SMTP server**.

User Flow:

- 1. **Step 1:** Upload CSV file (containing recipient email, template, and tone) or manually enter email details.
 - 2. **Step 2:** AI generates personalized emails.
 - 3. **Step 3:** Emails are sent to recipients.
-

Phase-4: Project Planning (Agile Methodologies)

Sprint Planning with Priorities

Sprint Task	Priority	Duration	Deadline	Assigned To	Dependencies
Django project setup	● High	3 hours	Day 1	Shiva	Django installation
CSV file & user input handling in Django	● High	3 hours	Day 1	Umesh	Django setup
AI model integration (Hugging Face)	● High	4 hours	Day 1	Venkat	CSV processing ready
Email sending setup (SMTP)	● High	3 hours	Day 1	Venkat	SMTP credentials
Testing & Debugging	● High	2 hours	Day 2	Uday	Backend ready
Basic UI for file upload & form	● Low	2 hours	Day 2	Shiva	Backend ready

Phase-5: Project Development

Technology Stack Used:

- **Backend:** Django
- **AI Model:** Hugging Face Transformers API (Mistral-7B)
- **Frontend:** Django HTML Templates
- **Email Sending:** SMTP via Gmail

Development Process:

- **CSV File Handling:** Read input CSV and extract user details.
- **AI Model Integration:** Hugging Face API generates personalized emails.
- **Bulk Email Sending:** Django sends emails via SMTP.
- **Logging & Error Handling:** Ensure emails are tracked and errors logged.

Challenges & Fixes:

- **Challenge:** Hugging Face API rate limits.
 - **Fix:** Optimize requests and cache responses.
 - **Challenge:** Emails going to spam.
 - **Fix:** Use a verified sender email with SPF/DKIM settings.
-

Phase-6: Functional & Performance Testing

Objective:

Ensure that the AI Email Generator functions correctly, meets requirements, and performs efficiently.

Testing Plan:

Test Case	Expected Outcome	Test Type	Tester	Status	Fix Required
CSV file upload	Successful read*	Functional	Uday	Pass	No
AI-generated email quality check	Proper content	Functional	Umesh	Pass	No
Bulk email sending via SMTP	Emails sent	Performance	Shiva	Pass	No
Spam detection & filtering	Avoid spam	Security	Venkat	Pending	Yes
UI responsiveness	Smooth flow	UI/UX	Entire Team	Pass	No
Error handling during email sending	Graceful fail	Functional	Venkat	Pass	No
