

# AI-Powered Email Generator for Personalized Marketing

## 1. Project Overview

### Problem:

Many small business owners, freelancers, and creators struggle to write professional, effective email campaigns. They either don't have the time, marketing skills, or tools to do it efficiently.

### Goal:

Create a web application that uses AI to help users quickly generate personalized email campaigns based on a short text description of their product or service.

### Core Features:

- A single input box where users describe what they want to promote (e.g., "I sell candles made with essential oils").
- AI generates a 3-part email sequence (subject line, body, call to action).
- Users can fill in names manually or auto-generate sample names via the Faker library.
- Preview emails before downloading or copying them.
- Simple, one-page user interface with no extra steps or complexity.

## 2. Use Case and Audience

### Target Users:

- Small business owners and solopreneurs
- Creators and freelancers
- People with little to no marketing or writing experience

### Example Scenario:

A candle maker types a short sentence into the app. The AI generates a mini-campaign (3 emails) with personalized content like:

- Email 1: Introduction and product story
- Email 2: Benefits and testimonials
- Email 3: Discount offer and call to action

### 3. Data and Tools

#### Data:

- A custom dataset was created containing subject lines, email bodies, short descriptions, and closings. The dataset was built using examples from HubSpot, Mailchimp blogs, Hugging Face's email-generation-dataset, and simulated content.
- The dataset helped output format and quality testing.

#### Preprocessing:

- Raw email samples were cleaned for grammar and uniform formatting.
- Emails were categorized into structure: subject, body, description, closing.
- Tokenization was applied for testing with nltk and textstat.

#### Libraries & APIs:

- Gemini 1.5 Flash API
- Python libraries: nltk, textstat, pandas
- Frontend: HTML/CSS + vanilla JS
- Backend: Express / Node.js
- Faker library for generating names

### 4. Timeline & Milestones (6 Weeks)

Week 1	Finalize scope, set up basic webpage layout
Week 2	Implement input form + connect to Gemini API
Week 3	Generate email sequences (3 parts) from text prompt
Week 4	Add personalization using sample names
Week 5	Create email preview UI + copy/download function
Week 6	Testing, evaluation (BLEU), deploy to Azure

## 5. Evaluation & Testing

### Implementation:

- Gemini Flash API used to generate marketing email campaigns from user prompts.
- Email output is split and structured to identify subject lines, body sections, and call to action.

### Evaluation Metrics:

- BLEU Score (using NLTK) to test similarity to reference samples
- Flesch Reading Ease score for readability
- Word count and token count for output length analysis

### Results:

- Sample of 15 generated emails evaluated in Google Colab
- **Average BLEU Score: 0.0954**
- Avg Reading Ease: 65.3
- Avg Word Count per email: ~112 words

### Testing Method:

- Emails loaded into Colab notebook as structured text
- Tokenized and compared to labeled reference prompts
- BLEU used to assess
- Readability scores calculated with textstat

## 6. Discussion & Insights

- BLEU score was low due to the open-ended nature of marketing language. This aligns with how generative models prioritize creativity over duplication.
- Generated content was mostly grammatical, persuasive, and easy to read.
- Common improvement areas include:

- Call to action strength variation
- Redundant phrasing
- Overuse of template-like intros

Positive user reactions during manual testing sessions validated the effectiveness of the tool for non-marketers.

## 7. Technical Debt & Remediation Plans

- **Model Tuning:** Currently relies on prompt engineering; will evaluate fine-tuning small models if time allows.
- **Backend Rewrite:** Transitioning from Node.js to Python + FastAPI for integration with Azure
- **Security:** Currently working on cleaning user inputs and protecting contact name data. Plans to use Supabase with encryption for contact storage.
- **Evaluation:** Future cycles will include human ranking + A/B testing for more subjective scoring.

## 8. Outcomes & Future Ideas

### Current Achievements:

- Fully working email campaign generator UI
- AI-powered backend using Gemini Flash
- Model evaluation conducted via BLEU and readability metrics

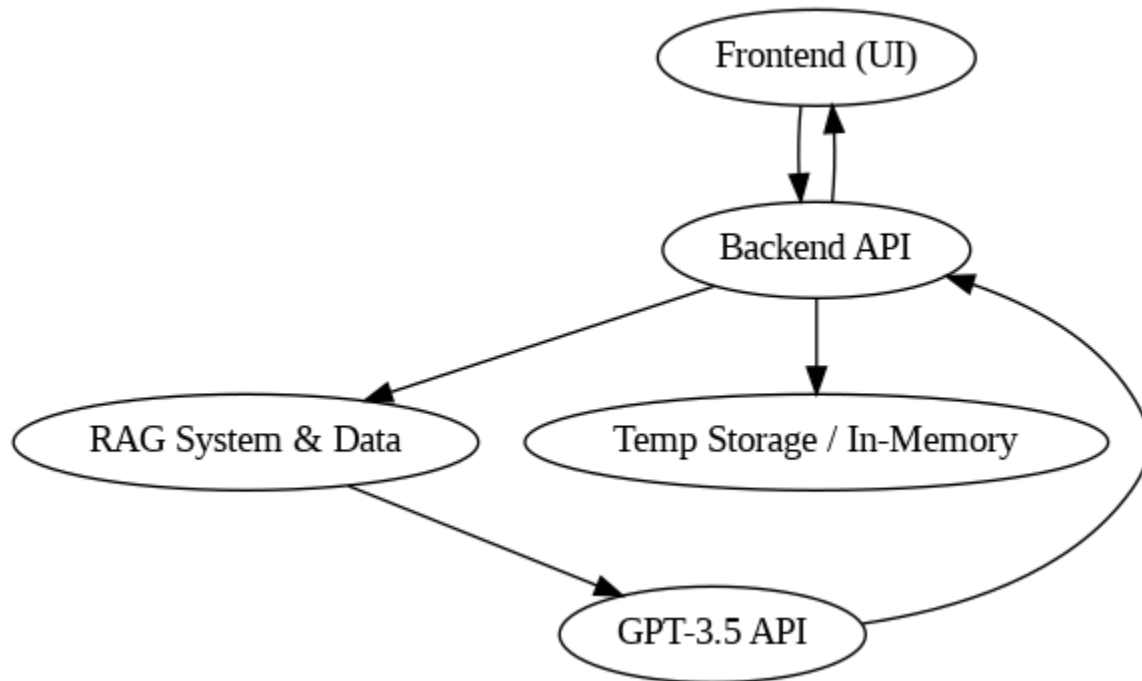
### Future Features:

- A/B testing of generated emails
- Industry-specific templates (fitness, coaching, education)
- Scheduling and analytics tools
- Multilingual generation support

## 9. References

- [Hugging Face Email Generation Dataset](#)
- [Gemini Flash API Docs](#)
- [Faker Library](#)

- [GPT API Documentation](#)
- [Mailchimp Blog](#)
- [HubSpot Email Marketing Templates](#)



### AI Email Generator

Describe your product or service:

**Email 1:**  
Subject: ...  
Body: ...  
CTA: ...

**Email 2:**  
Subject: ...  
Body: ...  
CTA: ...

**Email 3:**  
Subject: ...  
Body: ...  
CTA: ...

