# MARGEN AI - Your Personal Career Advisor

**MARGEN AI is an intelligent, web-based career guidance platform designed to provide users with personalized career paths, dynamic learning roadmaps, and AI-powered tools to prepare for their professional journey.**

This application leverages the power of Google's Gemini AI to understand a user's interests and skills, suggesting tailored career options and breaking them down into actionable, step-by-step learning milestones. From skill gap analysis to mock interviews, MARGEN AI is a one-stop solution for career exploration and preparation.

## ✨ Key Features

* **🤖 AI-Powered Career Suggestions**: Get a list of 7 unique career paths based on your interests, skills, and life goals.
* **🗺️ Dynamic Roadmaps**: Select a career and instantly receive a detailed, visual roadmap with learning milestones, key skills, and high-quality resources.
* **📊 Skill Gap Analysis**: Input your current skills to see a visual analysis of how you match up with your chosen career path and what you need to learn next.
* **💬 Mock Interviews**: Engage in a simulated interview with an AI hiring manager for your chosen career, receiving relevant questions and practicing your responses.
* **🤔 Interest Assessment**: Unsure about your interests? Take a short quiz to let the AI analyze your personality and suggest relevant fields.
* **🌗 Light & Dark Mode**: A sleek, modern UI with a theme that adapts to your preference.
* **🔐 User Authentication**: Secure sign-up and sign-in functionality using email/password or phone-based OTP.

## 🛠️ Tech Stack

* **Frontend**:
  + HTML5
  + CSS3 with Tailwind CSS
  + JavaScript (Vanilla)
  + [Mermaid.js](https://www.google.com/search?q=https://mermaid-js.github.io/mermaid/%23/) for roadmap visualizations
* **Backend**:
  + Python 3
  + Flask
  + Flask-SQLAlchemy (for database)
* **AI & APIs**:
  + Google Gemini API (for all generative AI features)
  + Twilio API (for OTP authentication)
* **Database**:
  + SQLite

## 🚀 Getting Started

Follow these instructions to get a local copy up and running for development and testing purposes.

### Prerequisites

* Python 3.8 or newer
* pip (Python package installer)
* A Google Gemini API Key. Get one for free at [Google AI Studio](https://aistudio.google.com/).
* (Optional) A Twilio account with a phone number, Account SID, and Auth Token for OTP functionality.

### Installation

1. **Clone the repository:**  
   git clone [https://github.com/your-username/margen-ai-project.git](https://github.com/your-username/margen-ai-project.git)  
   cd margen-ai-project
2. **Create a virtual environment:**  
   # For macOS/Linux  
   python3 -m venv venv  
   source venv/bin/activate  
     
   # For Windows  
   python -m venv venv  
   .\venv\Scripts\activate
3. Install dependencies:  
   Create a requirements.txt file in your project root with the following content:  
   Flask  
   Flask-SQLAlchemy  
   Flask-Cors  
   python-dotenv  
   google-generativeai  
   twilio  
   werkzeug  
   gunicorn  
     
   Then, install the packages:  
   pip install -r requirements.txt
4. **Set up environment variables:**
   * Create a file named .env in the root of your project directory.
   * Add your API keys to this file. **This file should not be committed to GitHub.**

# .env file  
  
# Get your API Key from Google AI Studio  
GEMINI\_API\_KEY="YOUR\_GEMINI\_API\_KEY\_HERE"  
  
# (Optional) Get your credentials from the Twilio Console  
TWILIO\_ACCOUNT\_SID="YOUR\_TWILIO\_ACCOUNT\_SID"  
TWILIO\_AUTH\_TOKEN="YOUR\_TWILIO\_AUTH\_TOKEN"  
TWILIO\_PHONE\_NUMBER="YOUR\_TWILIO\_PHONE\_NUMBER"

1. **Run the Flask application:**  
   flask run --port 5001
2. **Open the application:**
   * The backend will be running at http://127.0.0.1:5001.
   * Open your web browser and navigate to this address to view the application.

## 📂 Project Structure

margen-ai-project/  
├── templates/  
│ └── index2.html # The main frontend file  
├── venv/ # Virtual environment folder  
├── .env # Environment variables (API keys)  
├── app.py # Flask backend server  
└── requirements.txt # Python dependencies

## 📄 API Endpoints

The backend provides the following RESTful API endpoints:

| **Method** | **Endpoint** | **Description** |
| --- | --- | --- |
| POST | /signup | Creates a new user account. |
| POST | /signin | Authenticates a user with email and password. |
| POST | /send-otp | Sends a one-time password to a user's phone. |
| POST | /verify-otp | Verifies an OTP code for login. |
| POST | /find-interests | Generates user interests based on assessment answers. |
| POST | /generate-careers | Generates career recommendations from user profile. |
| POST | /generate-roadmap | Generates a detailed learning roadmap for a career. |
| POST | /analyze-skills | Performs a skill gap analysis. |
| POST | /start-interview | Begins a mock interview with a greeting. |
| POST | /continue-interview | Continues the mock interview conversation. |
| POST | /generate-project-pitch | Generates a project idea for a specific milestone. |

## 📜 License

This project is licensed under the MIT License. See the LICENSE file for details.