

AI Chatbot and Portfolio using flask:

GutHub repository Link:

<https://github.com/JP7599/PortfolioSemester2>

Prologue:

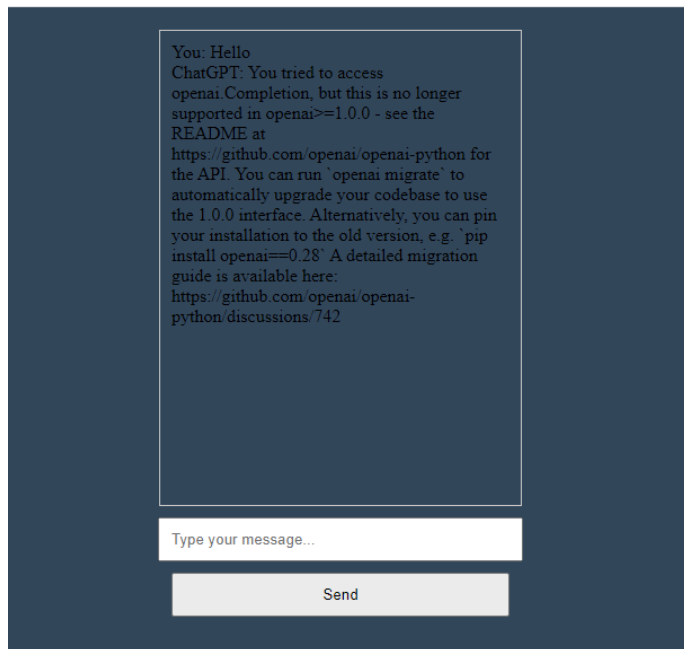
An idea i had for my portfolio was making an AI chatbot to answer questions like what my learning outcomes were and which projects i had done to achieve each one of them, basically a summary of my portfolio (an idea which became the start of many sleepless nights).

Introduction:

I decided to use the OpenAI chatGPT 3.5 API for this idea (i dont have the funds nor knowledge to train a LLM so it was easier). I had worked with this API in the past so i thought it would be a walk in the park, but the way i knew how to do it had been updated, and i took a little of time to find out the new way to do it (updating my openAI library and changing the line to call the API key,).

It's important to mention that I did not train on top of a model or anything like that, i just gave all the information the bot should know in the form of a prompt as you will see in the code at the end.

(the design of the chatbot was just a placeholder until the bot was working)



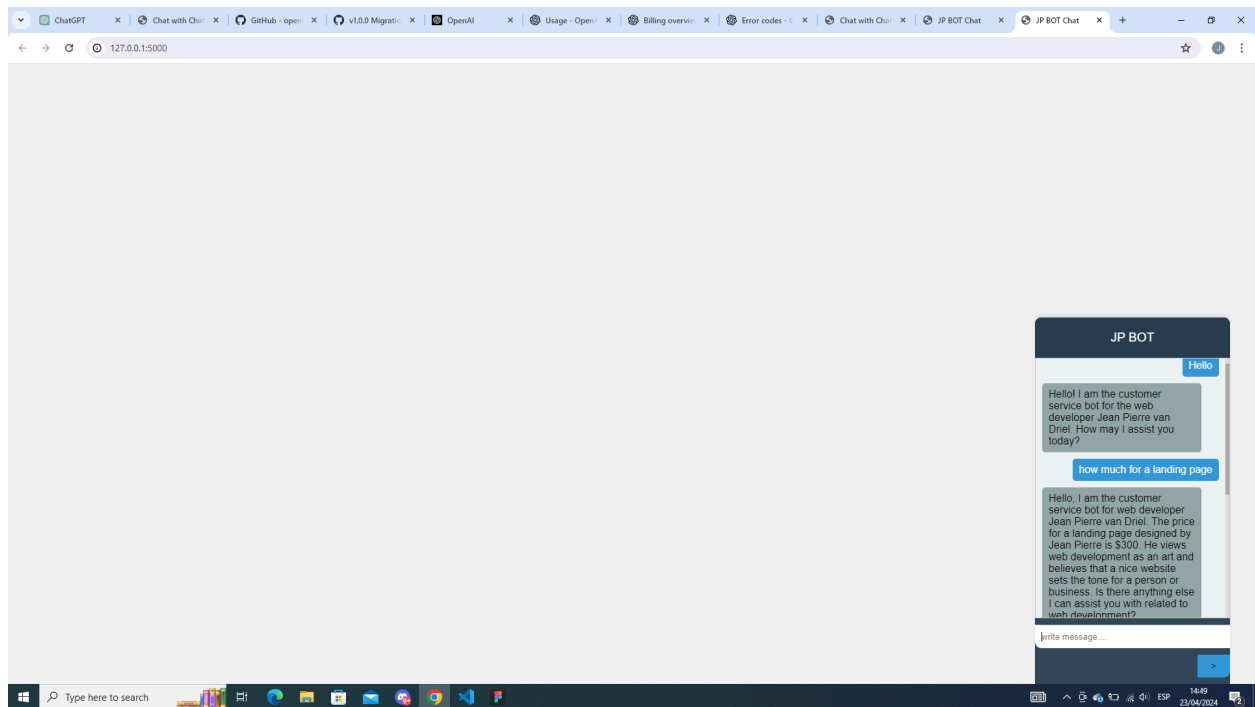
Error from the API asking to update my version of the OpenAI library.

Flaskification:

That was my first challenge in this journey, but after some reading on OpenAI's documentation i found the correct way of doing it and everything was looking bright, until i realized that with this python script my portfolio wasn't static anymore *surprised*, so to have the bot working on my portfolio, the whole portfolio should go through my flask framework's routes, this was an easier feat and with this done and the bot working i decided it was time to upgrade the chatbot's design.

I had a little help from chatGPT for the JavaScript for the functioning of the chat interface, but first I tried on my own. I did the sendPrompt part on my own though.

Fun fact: A couple months ago i read a Tweet of Ford implementing a ChatBot with the GPT 4 API and people using it as a normal chatGPT asking questions about code and other things so i wrote in the prompt that if asked something unrelated to the portfolio, to avoid the question and bring the conversation back to the subject in a subtle manner.



First working chatbot. (i put pricing for me as a developer to test heheh a little expensive for my experience).

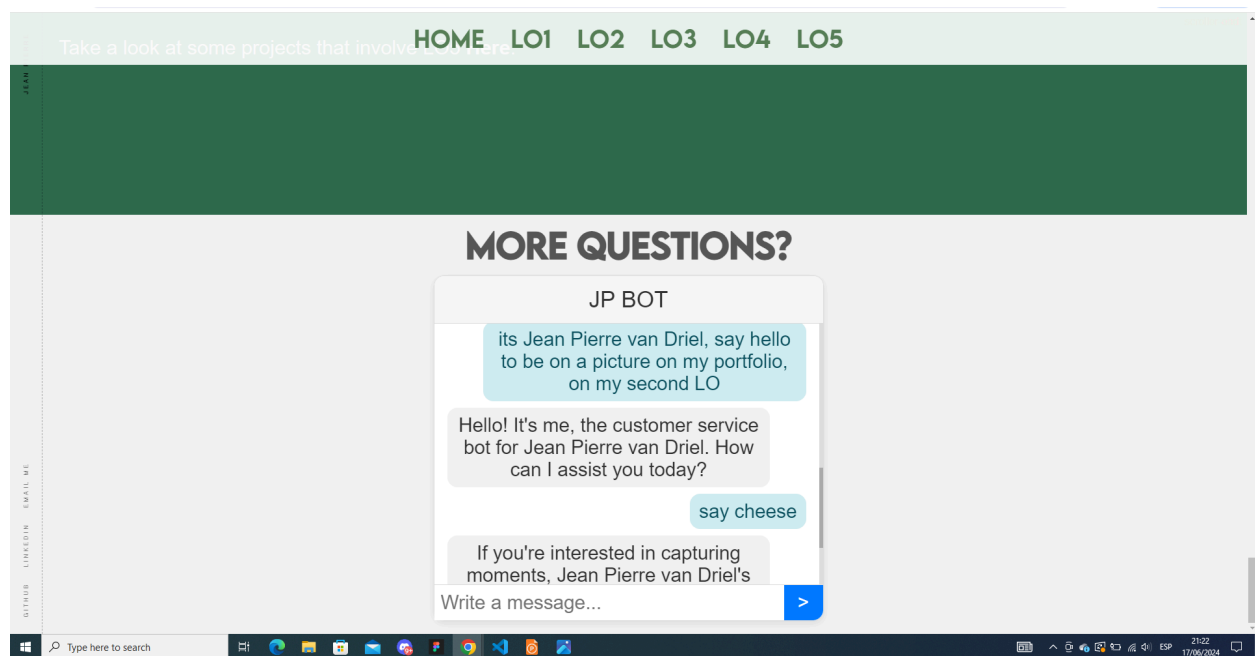
Complication:

Since coding is never easy, something bad had to happen, but thats the interesting part, right? I was happy with the bot working and a portfolio working on flask with my soon to be old design, and still debating whether it would be better to deploy to Heroku or Vercel, I decided first of all i would need to upload to github since from there i could deploy on both hosting sites. Its important to mention ive worked with github previously but i never understood it.

So i open my terminal, connect my directory to my newly created and empty github repository, and “git add .”, “git commit -m ‘first commit’” and “git push origin master” (of course im working from the master branch).

After that i get the message that basically said “*dumb man, dont post your API key in a github public repository*”. So i hid it in an .env file and hid that file in .gitignore.

With that issue solved i tried to push again but my repository was still empty, so without understanding what was happening and being very tired (i have trouble sleeping very often and this was one of this times that i had been 2 days without being able to sleep), i went to find what the issue was, did i go to Stackoverflow? Of course not, i went to chatGPT and blindly followed every step, even trying my own steps, and between furcing pushes and pulls i messed up my whole code, parts where duplicated, parts where missing and everything was a mess, so decided to start again and change my whole portfolio design.



Always professional and work focused, im so proud.

Ending:

This is getting long so ill be brief. After being able to upload my new portfolio to github, (this time backing it up before doing so), i upload it, i create a procfile and install the necessary dependencies to deploy in Heroku, from what i remember they were “Guicorn” and something else, i got this info from some blog feed i cant find now.

Fast forward of one night and one day of attempting to deploy on Heroku and reading the logs many times over to find why the connection was not being made between the code and the hosting, i accidentally saw on youtube that the procfile file in a youtube video was written with capital P, “Procfile”. This was the issue.

app.py explanation:

First i downloaded the required libraries and installed its dependencies, then loaded up my environment, created the flask app, fetch my API key from the.env file and turned it into a variable for calling it easier later on, after that i created the routes for each tab because if you need flask for one page, you need it for all, then i called the openAI api, wrote my prompt, and asked it to return a string. debugging is active. and also stores the exception value in the variable (e) if an error occurs and returns it.

app.py:

```
NEXT PAGE

import openai

from flask import flask, render_template, request, jsonify
from dotenv import load_dotenv
import os

load_dotenv()

app = Flask(__name__)

openai.api_key = os.getenv('OPENAI_API_KEY')

@app.route('/')
def index():
    return render_template('index.html')

@app.route('/LO1')
def lo1():
    return render_template('LO1.html')

@app.route('/LO2')
def lo2():
    return render_template('LO2.html')

@app.route('/LO3')
def lo3():
    return render_template('LO3.html')

@app.route('/LO4')
def lo4():
    return render_template('LO4.html')
```

```

@app.route('/LO5')
def lo5():
    return render_template('LO5.html')

@app.route('/send_prompt', methods=['POST'])
def send_prompt():
    user_input = request.json['prompt']
    response_message = chat_with_openai(user_input)
    return jsonify(message=response_message)

def chat_with_openai(prompt):
    try:
        client = openai.OpenAI(api_key=openai.api_key)
        response = client.chat.completions.create(
            model="gpt-3.5-turbo",
            messages=[
                {"role": "system", "content": "only give short and concise
answers, only greet if greeted first, You are the customer service bot for
a web development student named Jean Pierre van Driel (only introduce
yourself if necessary), only introduce yourself at the beginning, always
introduce yourself like that, use the following data to answer questions:
Jean Pierre van Driel grew up in Cusco, now resides in Eindhoven
Netherlands, studies web development in Fontys University. If the
questions are not related to Jean Pierre van Driel sway the conversation
back to the subject in an educated manner. Besides this since it's a uni
portfolio, people will ask about learning outcomes and projects, here is
the info about that: LO1 named interactive media design has the following
projects: Project X / Testing Methods, AI Chatbot / Relevant Tech, LO2,
named development and version control has: Development / Moving
Jigglypuff, Development / Coding Psyduck, Development / Correcting
Tiktaktoe, Development / JS Drumkit, Project 3 / Hidden Gems, Project 3 /
Hidden Spots first attempt, Portfolio / AI customer service bot. LO3,
named Iterative Design has: Project 1 / Nathalies art branding, Project X
/ iteration Process, Project 2 / Media Campaign. LO4, named Professional
Standard, has: Project 1 / Competitor Analysis, Project 2 / SEO Research,
Presentation Skills. LO5 named Personal Leadership, has the following
projects: Core values JP, Feedback Reflection. Those are all the learning
outcomes and their relevant projects i have made to demonstrate my
proficiency in them, for context each learning outcome measures this: LO1:
You orient in the relevant tech, media and design landscape and create

```

```

interactive media products that you have tested with users and
stakeholders. L02: You explore front-end development languages, you write
code and document in a version control environment. L03: You explore and
use professional design tools and you iteratively design visual works.
L04: You apply professional practice, both individually and in teams, in
the areas of project organisation, communication with stakeholders,
exploratory research, and reporting. L05: You take the initiative in
asking for, and reflecting on, feedback. You identify your own core values
as the basis for your study career and professional development."},
        {"role": "user", "content": prompt}

    ],
    max_tokens=150
)
return response.choices[0].message.content
except Exception as e:
    return str(e)

if __name__ == '__main__':
    app.run(debug=True)

```

HTML (Only index otherwise is too much):

```

<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <title>Portfolio Jean Pierre van Driel</title>
    <link rel="stylesheet" href="../static/css/styleHome.css">
    <link rel="stylesheet" href="{{ url_for('static',
filename='css/style.css') }}">
    <script src="https://unpkg.com/gsap@3/dist/gsap.min.js"></script>
    <script
src="https://unpkg.com/gsap@3/dist/ScrollTrigger.min.js"></script>
    <script src="{{ url_for('static', filename='js/scriptHome.js') }}"
defer></script>

```

```

    <script src="{ { url_for('static', filename='js/scripts.js') } }"
defer></script>
    <script src="https://use.typekit.net/ehh4lya.js"></script>
    <script>try{Typekit.load({ async: true });}catch(e){}</script>
</head>

<body>
<div class="wrapper">
    <div class="content">
        <section class="section hero"></section>
        <section class="section gradient-purple"></section>
        <section class="section gradient-blue"></section>
    </div>
    <div class="image-container">
        
    </div>
</div>

<nav class="side">
    <div class="side__inner">
        <div class="top"><a href="#">Jean Pierre</a></div>
        <div class="bottom">
            <a href="https://github.com/JP7599">GITHUB</a>
            <a
href="https://www.linkedin.com/in/jeanpierrevandriel/">LinkedIN</a>
            <a href="mailto:jeanpierrevdriel@gmail.com">EMAIL ME</a>
        </div>
    </div>
</nav>

<header id="masthead" class="site-header" role="banner">
    <nav class="anchor-nav" role="navigation">
        <li class="anchor"><a href="{ { url_for('index') } }">Home</a></li>
        <li class="anchor"><a href="{ { url_for('lo1') } }">L01</a></li>
        <li class="anchor"><a href="{ { url_for('lo2') } }">L02</a></li>
        <li class="anchor"><a href="{ { url_for('lo3') } }">L03</a></li>
        <li class="anchor"><a href="{ { url_for('lo4') } }">L04</a></li>
        <li class="anchor"><a href="{ { url_for('lo5') } }">L05</a></li>
    </nav>
</header>

```

```
<section class="section section--hello" id="hello">
  <div class="section__inner">
    <div class="block">
      <h2>Hello, <br>
        my name is Jean Pierre van Driel and this is my portfolio:
      </h2>
    </div>
  </div>
</section>
```

```
<section class="section section--intro" id="intro">
  <div class="section__inner">
    <div class="block">
      <a class="pre" href="#intro"><span>#1</span>Interactive media
products</a>
      <h2>Take a look
        at the projects
        I made
        for L01</h2>
      <p class="lead">
        See the projects on L01 <a href="{{ url_for('lo1')
}}">Here</a>.
      </p>
    </div>
  </div>
</section>
```

```
<section class="section section--about" id="about">
  <div class="section__inner">
    <div class="block">
      <a class="pre" href="#about"><span>#2</span>Development &
version control</a>
      <h2>Learning and exploring new programming languages is a
passion for me.</h2>
      <p class="lead">Take a look at some projects of mine from my
L02 <a href="{{ url_for('lo2') }}">Here</a>.</p>
    </div>
  </div>
</section>
```



```

<section class="section section--clients" id="clients">
  <div class="section__inner">
    <div class="block">
      <a class="pre" href="#clients"><span>#3</span>Iterative
design</a>
      <h2>I like to explore and use professional design tools to
make my work better.</h2>
      <p class="lead">You can see some of my projects from L03 <a
href="{{ url_for('lo3') }}">Here</a>.</p>
    </div>
  </div>
</section>

<section class="section section--portfolio" id="portfolio">
  <div class="section__inner">
    <div class="block">
      <a class="pre" href="#portfolio"><span>#4</span>Professional
standard</a>
      <h2>I try to keep my personal and group work organized and
easy to understand.</h2>
      <p class="lead">Take a look at some projects that involve L04
<a href="{{ url_for('lo4') }}">Here</a>.</p>
    </div>
  </div>
</section>

<section class="section section--leadership" id="leadership">
  <div class="section__inner">
    <div class="block">
      <a class="pre" href="#leadership"><span>#5</span>Personal
Leadership</a>
      <h2>I like to take initiative to make some new projects.</h2>
      <p class="lead">Take a look at some projects that involve L05
<a href="{{ url_for('lo5') }}">Here</a>.</p>
    </div>
  </div>
</section>

<main>

```

```

<div class="inquiry-section">
  <h2>MORE QUESTIONS?</h2>
  <div class="chat-widget">
    <div class="chat-header">JP BOT</div>
    <div class="chat-body" id="chatBox"></div>
    <div class="chat-input-container">
      <input type="text" class="chat-input" id="userInput"
placeholder="Write a message...">
      <button class="chat-send"
onclick="sendPrompt()">></button>
    </div>
  </div>
</div>
</main>

</body>
</html>

```

JS (Special thanks to chatGPT for the help to understand):

```

// Send message to server and receive response
function sendPrompt() {
  const inputField = document.getElementById('userInput');
  const chatBox = document.getElementById('chatBox');
  const userText = inputField.value.trim();
  inputField.value = '';

  // Add message to the chat box
  if (userText) {
    const userDiv = document.createElement('div');
    userDiv.textContent = userText;
    userDiv.className = 'chat-message user-message';
    chatBox.appendChild(userDiv);
    scrollToBottom(chatBox);

    // Send message to server and handle response
    fetch('/send_prompt', {
      method: 'POST',
      headers: {
        'Content-Type': 'application/json'

```

```

        },
        body: JSON.stringify({ prompt: userText })
    ))
    .then(response => response.json())
    .then(data => {
        const replyDiv = document.createElement('div');
        replyDiv.textContent = data.message;
        replyDiv.className = 'chat-message admin-message';
        chatBox.appendChild(replyDiv);
        scrollToBottom(chatBox);
    })
    .catch(error => console.error('Error:', error));
}

// Keep the chat scrolled to the bottom
function scrollToBottom(element) {
    element.scrollTop = element.scrollHeight;
}

// Send pressing enter
document.getElementById('userInput').addEventListener('keypress', function
(e) {
    if (e.key === 'Enter') {
        sendPrompt();
    }
});

```

CSS:

```

/* Chat widget */
.chat-widget {
    width: 100%;
    max-width: 600px;
    background-color: #ffffff;
    border-radius: 16px;
    box-shadow: 0 4px 12px rgba(0,0,0,0.1);
    margin: 0 auto;
    border: 1px solid #e0e0e0;
}

```

```
.chat-header {  
  background-color: #f7f7f7;  
  color: #333;  
  padding: 16px;  
  font-size: 1em;  
  border-radius: 16px 16px 0 0;  
  text-align: center;  
  border-bottom: 1px solid #e0e0e0;  
}
```

```
.chat-body {  
  height: 400px;  
  overflow-y: auto;  
  padding: 20px;  
  background: #ffffff;  
  display: flex;  
  flex-direction: column;  
  align-items: flex-start;  
  font-size: 0.85em;  
  width: 100%;  
  box-sizing: border-box;  
}
```

```
.chat-message {  
  background-color: #f0f0f0;  
  border-radius: 16px;  
  padding: 10px 14px;  
  margin-bottom: 10px;  
  max-width: 90%;  
  word-wrap: break-word;  
}
```

```
.chat-message.user-message {  
  align-self: flex-end;  
  background-color: #d1ecf1;  
  color: #0c5460;  
}
```

```
.chat-message.admin-message {
```

```
    align-self: flex-start;
    background-color: #f0f0f0;
    color: #333;
}

.chat-input-container {
    display: flex;
    width: 100%;
    border-top: 1px solid #e0e0e0;
    border-radius: 0 0 16px 16px;
}

.chat-input {
    flex: 1;
    padding: 10px;
    border: none;
    outline: none;
    box-sizing: border-box;
    font-size: 0.85em;
}

.chat-send {
    width: 60px;
    border: none;
    background-color: #007bff;
    color: #ffffff;
    outline: none;
    cursor: pointer;
    border-radius: 0 0 16px 0;
    font-size: 0.85em;
}

.chat-widget:after {
    content: "";
    display: table;
    clear: both;
}

.chat-body::-webkit-scrollbar {
    width: 6px;
}
```

```
}

.chat-body::-webkit-scrollbar-track {
  background: #ddd;
}

.chat-body::-webkit-scrollbar-thumb {
  background: #aaa;
}

@media (max-width: 600px) {
  .chat-widget {
    width: 100%;
    border-radius: 0;
  }
}

.inquiry-section {
  padding: 20px;
  text-align: center;
}

.inquiry-section h2 {
  margin-bottom: 20px;
  font-size: 1.5em;
}
```

Directory:

