

Team 2

Final Presentation

Kayla Quarles, Theodore Wimberly, Mariah McMichael, Telly Smith, and Oluwakanyinsola Awolowo

Our Goal

Develop an AI Chatbot to assist Morgan computer science students

Objective summary

Create an AI Chatbot specifically for Morgan State University Computer Science students that allows current students to have 24/7 access to ask about anything within the CS department. With the chatbot, students will be able to get information on various topics which will include the following: course descriptions/coursework, information about graduate programs, information about CS Faculty, and more.

Target audience

Enrolled Morgan State University Computer Science students

Our Project Approach

1. Utilize existing chatbot frameworks to **develop** our chatbot
2. Gather training data to **build** the dialogue flow between the bot and the user
3. **Research** Morgan State University branding guidelines
4. Make front-end **changes** including Morgan State University branding logos, colors, & font
5. Conduct **testing**
6. Work on final **demo** presentation

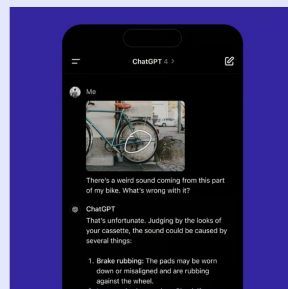
Our Research

The team evaluated the pros and cons of Microsoft Copilot and OpenAI chatbot. Then based on the research and testing we conducted, we decided to use OpenAI framework for our chatbot. The following features were evaluated:

- The implementation and capabilities
- Cost
- Ease of use

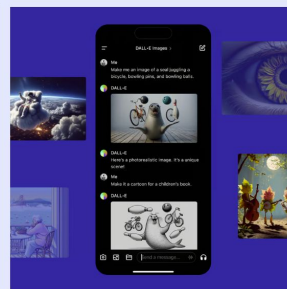
FEATURE	CHATGPT	COPILOT FOR 365
PRIMARY USE	General-purpose chatbot and text generation tool	Productivity assistant for Microsoft 365 users
PRICING	Free for basic use, \$20/month for ChatGPT 4	Requires a Microsoft 365 subscription
AVAILABILITY	Available through APIs and various platforms	Integrated into Microsoft 365 applications
STRENGTHS	Wide range of applications, including creative writing, code generation, and chat conversations	Can help with writing emails, documents, presentations, and more
WEAKNESSES	Can sometimes generate nonsensical or irrelevant text	May not be as helpful for users who are not familiar with Microsoft 365
BEST FOR	Users who need a versatile AI tool for various tasks	Microsoft 365 users who want to improve their productivity

ChatGPT can see, hear, and speak



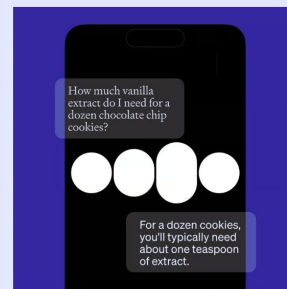
Chat with images

You can now show ChatGPT images and start a chat. Troubleshoot why your grill won't start, explore the contents of your fridge to plan a meal, or analyze a complex graph for work-related data.



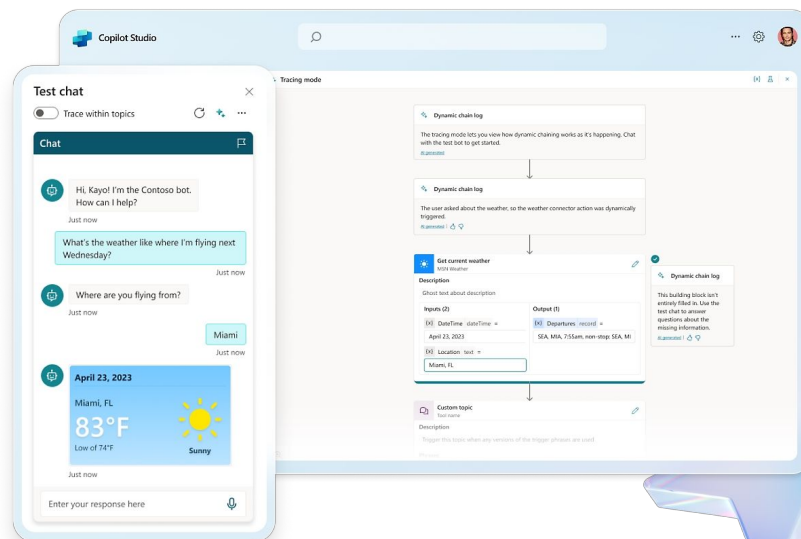
Create new images

Create images simply by describing them in ChatGPT. Invent new logos, comic strips, and photorealistic scenes right in the chat. You can bring your ideas to life with our most capable image model, DALL-E 3.



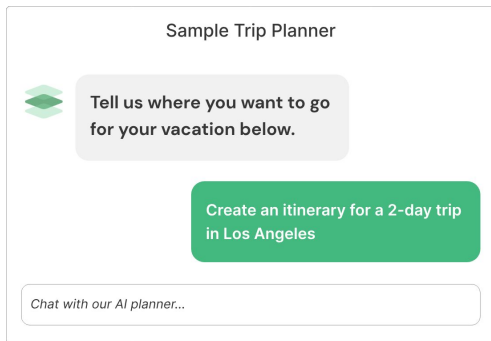
Chat with voice

You can now use voice to engage in a back-and-forth conversation with ChatGPT. Speak with it on the go, request a bedtime story for your family, or settle a dinner table debate.



OpenAI Research & Testing

The following are examples of what the team has been recently researching and testing using OpenAI:



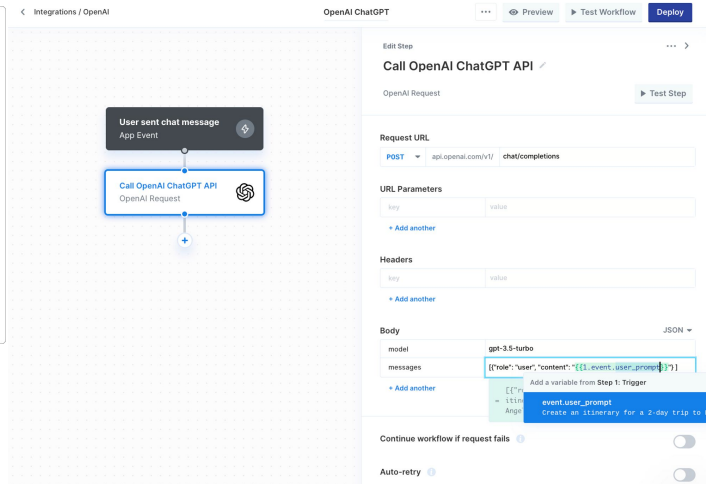
OpenAI Sample Request

Source: <https://www.useparagon.com/blog/how-to-build-a-native-openai-integration>

-This is an example of a sample request to the OpenAI API using chat API.

-We learned that OpenAI offers several different APIs and each of them offer a specific focus and use case, so you'll need to choose the API that best fits the use case that you want to provide your users.

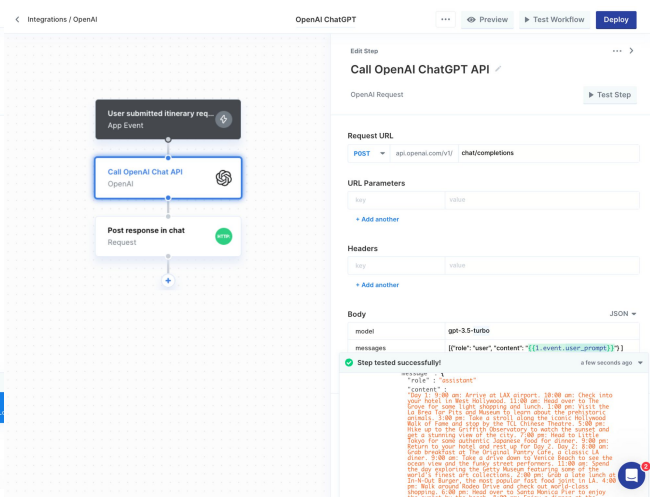
-For conversational features to an application, chat API is recommended



OpenAI Workflow Example 1

Source: <https://www.useparagon.com/blog/how-to-build-a-native-openai-integration>

Here's what the workflow behind the scenes looked like. The user's chat message is used as an app event to trigger the OpenAI integration workflow.



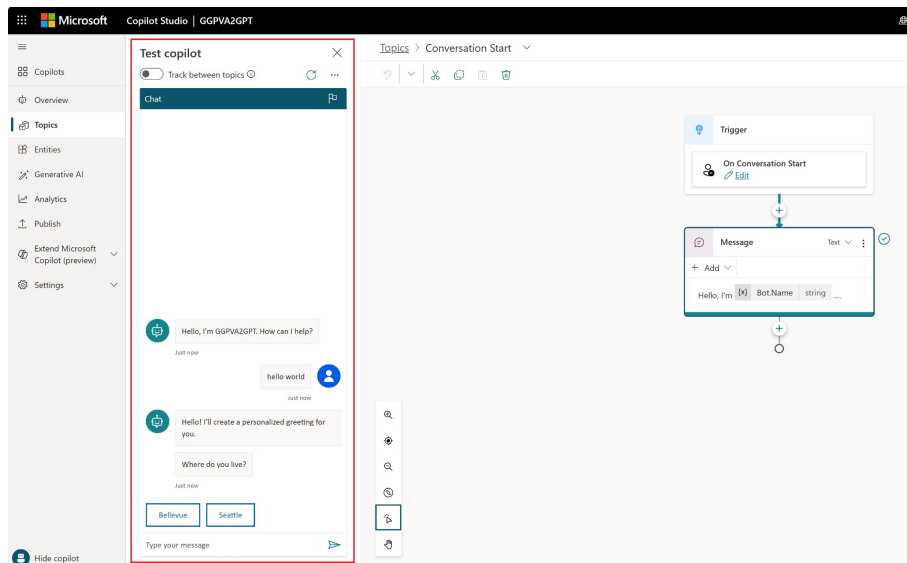
OpenAI Workflow Example 2

Source: <https://www.useparagon.com/blog/how-to-build-a-native-openai-integration>

When the test request is executed the in the workflow the response gets generated by OpenAI's chat API

Microsoft Copilot Research & Testing

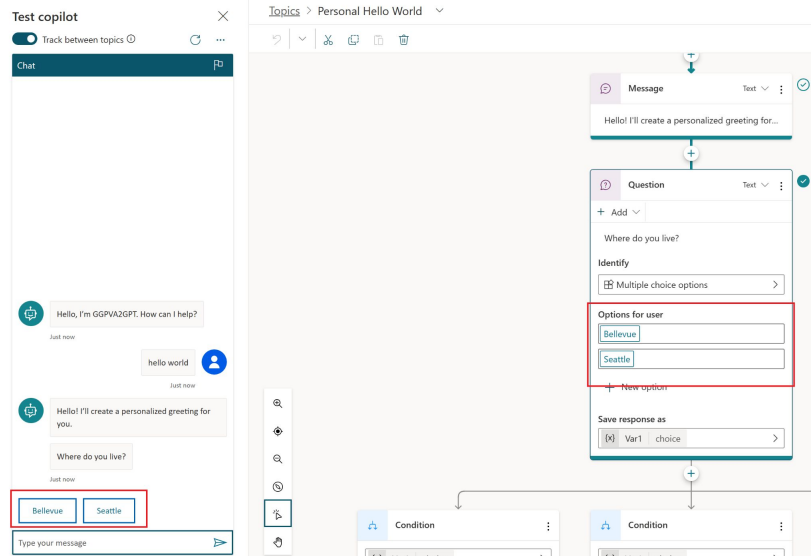
The following are examples of what the team has been recently researching and testing using Microsoft Copilot:



Copilot Studio Example 1

Source: <https://learn.microsoft.com/en-us/microsoft-copilot-studio/fundamentals-get-started?tabs=web>

Our team researched and tested content in real time using tutorials like this. In this example, content is authored into a dialog tree like the one seen here on the right. Then the conversation is tested in real time to see if it's working as expected using the copilot pane in Copilot Studio.



Copilot Studio Example 2

Source: <https://learn.microsoft.com/en-us/microsoft-copilot-studio/fundamentals-get-started?tabs=web>

-Here the user types "hello world" in the chat window, and sends the message to the copilot.

-Once the message is sent you will see on the top right portion of the dialog tree is highlighted in green that Seattle and Bellevue are presented as user options in the test copilot pane.

-Now the copilot is waiting for the user to respond while displaying suggestions on how to respond. The suggestion buttons are going to reflect what you authored within your dialog tree in the Ask a question node. Then, in the test copilot you can either select these suggestion buttons to continue, or you can enter your response into the chat window.

Gathering Training Data

We gathered training data to answer the users frequently asked questions/answers. Our focus was to provide responses for current students queries.

[Home](#)
[Announcements](#)
Modules
[People](#)
[Collaborations](#)
[Badges](#)
[Name Coach](#)
[NetTutor](#)

Collapse All

• Welcome: Begin Here

Organization Information

About this Organization

About your Organization Leader(s)

Question Board

How to use Canvas

Canvas: Getting Started

Canvas: Helpful Tips

Canvas: Additional Support

• Academic Advising

Spring 2024 Advising

SCMNS Advising Instructions

Change Your Major, Declare Your Track, or Add a Minor

Starfish

Starfish: Setting Up Your Profile

Starfish: My Success Network

Starfish: Appointments

Starfish: Dashboard and Messages

Schedule Planner

Schedule Planner

Degree Works

SCMNS Student Information Hub > Pages > About this Organization

Immersive Reader

[Home](#)
[Announcements](#)
Modules
[People](#)
[Collaborations](#)
[Badges](#)
[Name Coach](#)
[NetTutor](#)

About this Organization

About the School of Computer, Mathematical, and Natural Sciences (SCMNS)

The School of Computer, Mathematical and Natural Sciences (SCMNS) houses five academic departments: Biology, Chemistry, Computer Science, Mathematics and Physics. At SCMNS, we provide the best training to place our graduates in advanced degree programs and find employment. For this purpose, we have established formalized partnerships to admit students into professional schools, graduate schools, and place them into internship programs (Google, Facebook, Morgan Chase, Apple, Intel, etc.). Uniquely located in the Washington-Baltimore region, we have built partnerships for graduates to obtain employment in information technology, cybersecurity, health care, insurance, and chemical and pharmaceutical industries, as well as a long list of federal governmental agencies. The B.S. degree in Actuarial Science is the only such program in the state of Maryland and among the nation's HBCUs. The B.S. in Cloud Computing program is one of the first in the country and is offered in both online and face-to-face format. Both the chemistry and the medical laboratory science programs are accredited/certified by their respective societies.



Next

Training Data Source Example 2: SCMNS Student Information Hub

Source: SCMNS Student Information Hub Modules on Canvas

Training Data Source Example 1: SCMNS Student Information Hub

Source: SCMNS Student Information Hub Modules on Canvas

Gathering Training Data

- ★ Reviewed feedback and frequently asked questions from current CS students to understand the topics they have inquire about. (course registrations, homework assistant, faculty contact information, academic resources)
- ★ Data drawn from Morgan State official website, Canva SCMS Student Information Hub, and emails sent out from faculty members.

Objective:

- ★ Assisting students in navigating academic requirements, course selection and graduation pathways.
- ★ Collaborating with staff members to address student needs and promote student success.

- ★ Who can the students contact for:
 - Transferring credits
 - Academic advising
 - Internship/job opportunities
 - Specific interest, (game development,SWE AI/ML, cybersecurity)
 - Research opportunities
 - Scholarship
 - Funding for events/organizations
- ★ Can you provide more information on this course and its syllabus?
 - Listed below are
- ★ What academic support services are available to computer science students?
 - Academic advising
 - Internship opportunities
 - Organizations pages
- ★ What are the STEM oriented organizations on campus?
 - WiCS, GSDC, SWE,NSBE, SACS,
- ★ Where can I go to get help with resume building, technical interviews, career counseling?
 - Tyler Hall, Library, Counseling service, CASA, PEERS bears
- ★ What equipment and resources does the department offer to students? (devices,lab,website subscriptions)
 - Microsoft word, Adobe Photoshop, Canva,
- ★ What graduate programs are aligned with the Computer Science curriculum?
 - M.S. Advanced computing
 - M.S. Bioinformatics
 - M.S. Integrated Science

Preparing Training Data

Intent: Faculty Information

- Sample Utterances:
- "Who teaches the Artificial Intelligence course this semester?"
- "Can I find Dr. Wang's office hours?"
- "Are there any female professors in the Computer Science department?"
- Context: Queries about specific faculty members, their roles, and contact information.

Intent: Course Information

- Sample Utterances:
- "Can you provide information about the Introduction to Computer Science course?"
- "Tell me about the prerequisites for the Artificial Intelligence course."
- "What are the elective options for the Computer Science major?"
- Context: Include variations for different courses, prerequisites, and elective choices.

Intent: Academic Support

- Sample Utterances:
- "I'm struggling with my programming assignment. Can you help?"
- "How can I schedule a meeting with my academic advisor?"
- "Do you have resources for studying algorithms?"
- Context: Cover queries related to homework help, scheduling meetings, and accessing academic resources.

Intent: Enrollment and Registration

- Sample Utterances:
- "How do I register for the Game Design course?"
- "When is the deadline for adding courses this semester?"

- "Where can I find information about dropping a class?"
- Context: Inquiries related to course registration procedures, deadlines, and enrollment policies.

Intent: Career Opportunities

- Sample Utterances:
- "What internship opportunities are available for Computer Science majors?"
- "Can you recommend any networking events for students interested in software development?"
- "Where can I find resources for preparing my resume for tech internships?"
- Context: Providing information about internship opportunities, networking events, and resume preparation resources for students.

Intent: Accreditation/Certification

- Sample Utterances:
- "Are the chemistry programs at SCMNS accredited?"
- "Is the medical laboratory science program certified?"
- "Can you confirm the accreditation status of SCMNS programs?"
- Context: Confirming the accreditation and certification status of programs offered within SCMNS.

Intent: General Information

- Sample Utterances:
- "What are the office hours for the Computer Science department?"
- "How do I access the computer labs on campus?"
- "Where can I find information about scholarships for Computer Science students?"
- Context: Providing general information about departmental office hours, facilities access, and available scholarships.

Intent: Feedback Collection

- Sample Utterances:
- "Was this response helpful? Please rate it with a thumbs up or thumbs down."
- "Do you have any suggestions to improve the chatbot's performance?"
- Context: Collecting feedback from users to improve the chatbot's performance and user experience.

Our first chatbot design

MSU Computer Science Chatbot

Your Personal Assistant for Computer Science Students

Benny: Hello! How may I assist you today?:

- Course Descriptions
- Homework Help
- Graduate Programs
- Staff Members
- Student Advice
- Concerns
- Suggestions
- Other

Type your message here

Send

Our development environment for our first chatbot

```
index.html > index.html
1 <!DOCTYPE html>
2 <html lang="en">
3
4 <head>
5   <meta charset="UTF-8">
6   <meta name="viewport" content="width=device-width, initial-scale=1.0">
7   <title>MSU Computer Science Chatbot</title>
8   <link rel="stylesheet" href="style.css">
9 </head>
10
11 <body>
12   <header>
13     <div class="container">
14       <h1>MSU Computer Science Chatbot</h1>
15       <p>Your Personal Assistant for Computer Science Students</p>
16     </div>
17   </header>
18
19   <main>
20     <div class="container">
21       <div class="chat-container">
22         <div class="chat-box" id="chat-box"></div>
23         <input type="text" id="user-input" placeholder="Type your message here">
24         <button onClick="sendMessage()">Send</button>
25       </div>
26     </div>
27   </main>
28
29   <script src="script.js"></script>
30 </body>
31 </html>

script.js > onload
1 window.onload = function() {
2   var welcomeMessage = "<div><strong>Benny:</strong> Hello! How are you?</div>";
3   welcomeMessage += "<ul>";
4   welcomeMessage += "<li>Course Descriptions</li>";
5   welcomeMessage += "<li>Homework Help</li>";
6   welcomeMessage += "<li>Graduate Programs</li>";
7   welcomeMessage += "<li>Staff Members</li>";
8   welcomeMessage += "<li>Student Advice</li>";
9   welcomeMessage += "<li>Concerns</li>";
10  welcomeMessage += "<li>Suggestions</li>";
11  welcomeMessage += "<li>Other</li>";
12  welcomeMessage += "</ul>";
13
14  var chatBox = document.getElementById("chat-box");
15  chatBox.innerHTML += welcomeMessage;
16
17
18  function sendMessage() {
19    var userInput = document.getElementById("user-input").value;
20    if (userInput.trim() != "") {
21      var chatBox = document.getElementById("chat-box");
22      var userMessage = "<div><strong>You:</strong> " + userInput + "</div>";
23      chatBox.innerHTML += userMessage;
24
25      var botResponse = respondToUserInput(userInput);
26
27      var botMessage = "<div><strong>Chatbot:</strong> " + botResponse + "</div>";
28      chatBox.innerHTML += botMessage;
29
30      chatBox.scrollTop = chatBox.scrollHeight;
31
32      document.getElementById("user-input").value = "";
33
34    }
35  }

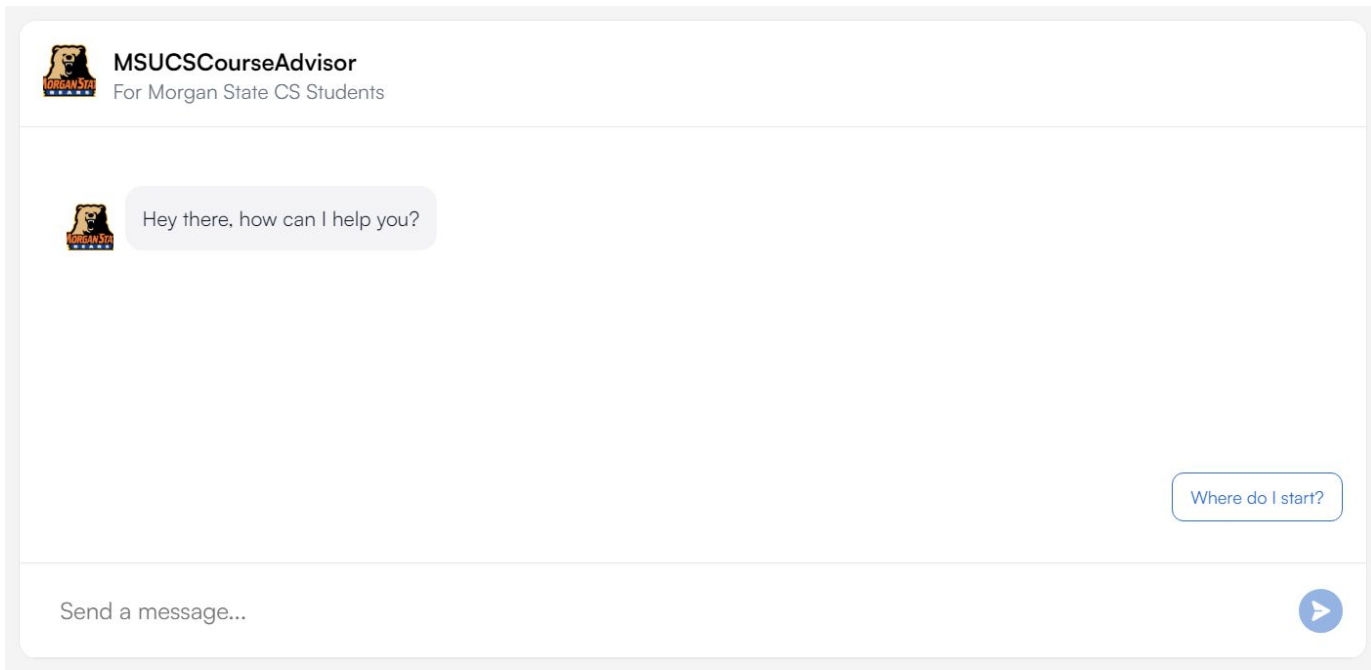
script.js > respondToUserInput
36
37
38  function respondToUserInput(userInput) {
39
40    userInput = userInput.toLowerCase();
41
42    // Predefined responses based on user input
43    var responses = {
44      "course descriptions": "You can find course descriptions on the MSU website.",
45      "homework help": "For homework help, consider reaching out to the CS Tutoring Center.",
46      "graduate programs": "Information about graduate programs can be found on the MSU Graduate School website.",
47      "staff members": "You can find information about staff members on the MSU HR website.",
48      "student advice": "For student advice, consider reaching out to the MSU Student Center.",
49      "concerns": "If you have concerns, please contact the Morgan S. College of Engineering and Computer Science.",
50      "other": "How may assist you further?",
51    };
52    // Adding more responses in this space
53    "default": "I'm sorry, I'm not sure how to respond to that. Can you rephrase?"
54  };
55
56  for (var keyword in responses) {
57    if (userInput.toLowerCase().includes(keyword)) {
58      return responses[keyword];
59    }
60  }
61
62  return responses["default"];
63 }

style.css > .chat-container
1
2  body {
3    font-family: Arial, sans-serif;
4    margin: 0;
5    padding: 0;
6  }
7
8  header {
9    background-color: #F47937;
10   color: #1B4383;
11   text-align: center;
12   padding: 10px 0;
13 }
14
15 footer {
16   background-color: #333;
17   color: black;
18   text-align: center;
19   padding: 10px 0;
20   position: static;
21   bottom: 0;
22   width: 100%;
23 }
24
25 .chatbot {
26   margin: 20px auto;
27   max-width: 600px;
28   border: 1px solid #ccc;
29   border-radius: 5px;
30   padding: 20px;
31 }
32
33 .chat-container {
34   height: 300px;
35   overflow-y: auto;
```

Development Environment for Chatbot Design 1

Source: <https://github.com/tellysmithjr5/Team-2-AI-Chatbot.git>

Our final chatbot design



Chatbot Design 2

Source: https://github.com/twscode/cosc_final/tree/main

Our development environment for our final chatbot design



Live Demo

Final Chatbot Design 2

Source: https://github.com/twscode/cosc_final/tree/main