

SunshineSpend / GT Fall 2025 Intern Project

1. Project Overview

The chatbot will be a web-based ChatGPT-style interface -- with a simple Q&A text box and associated responses only.

Its primary goals are to:

- Help clients locate Healthcare Provider (HCP) information, including correct name, specialty, geography, NPI, state license details, and other attributes. Input data may be incomplete, misspelled, or partially incorrect.
- Answer general questions about the CMS Open Payments program and how it works, as well as detailed questions about reports. For example, 'When I pay an HCP by check, what should I indicate as the Date of Payment?' The bot should also answer questions about the SunshineSpend Service (SaaS).
- Provide inline reports, analytics, and statistics from CMS Open Payments data (e.g., top specialties by payment volume, payment type breakdowns, or HCP-specific details).

2. How the Bot Should Behave

When a user asks 'what can you do?', the bot should reply with something like:

"I can help you identify and provide details about healthcare providers (HCPs), answer questions about the Open Payments program, explain how SunshineSpend works and summarize payment statistics. Try asking me about top specialties in Florida or a specific doctor's payment history."

The chatbot interface should:

- Include an input box for questions.
- Display responses in a chat-style format.
- Provide clear, concise answers.
- Optionally cite references (e.g., 'Data source: CMS Open Payments 2023').

3. Fall Semester Project Timeline

- 08/21) Project Kickoff:

Slack access and project documentation provided.

- 08/27) First Office Hours:

Questions answered.

- 08/29) Teams Formed:

Documentation reviewed and finalized team structure.

- By 08/31? - depends on when we receive the list of teams) Server Access:

Server credentials, working/sample code, and CMS Open Payments datasets, databases, and APIs provided to teams.

NOTE: If you require additional APIs, document them clearly and submit the request to support@sunshinespend.com -- we are generally able to respond within one business day.

- 10/08) Checkpoint #1:

Bot can assist with HCP lookups: database queries, API calls, state license lookups, and access to previous years' SunshineSpend data -- feedback provided by 10/15.

- 11/05) Checkpoint #2:

Bot can answer Open Payments program questions, CMS Open Payments report-specific questions, and SunshineSpend-related queries -- feedback provided by 11/12.

- 12/03) Checkpoint #3 / Final Deliverable -- project feedback provided by 12/10:

Bot can generate inline charts and/or text summaries related to CMS Open Payments data and previous years' client data.

4. Deliverables

- Access Instructions: Chatbot should be deployed on the team's subdomain (e.g., teamX.sunshinespend.com/bot).
- Integration: Chatbot should accept JSON API queries (/ask) so it can be tested programmatically.
- Documentation: Each team should provide:
 - * What their bot can do.
 - * How their bot works (methods, integrations, APIs, etc.).
 - * How the bot will remain up to date (data refresh mechanisms - manual or otherwise).
 - * Any other details relevant to the bot's operation.

5. Evaluation

Your chatbot will be evaluated on:

1. Accuracy:

- We will use a standard list of 20 test questions (covering HCP lookup, CMS Open Payments basics, SunshineSpend, and analytics). **No, SunshineSpend will not be providing the questions used for the evaluation. We will, however, provide a handful of sample questions.**
- The bot must answer clearly and correctly.

2. Feature Delivery:

- We will test everything your bot claims it can do.

3. Overall Functionality:

- The bot should reliably perform the three main tasks outlined in Section 1. Any additional capabilities will be considered a bonus if they perform well.

4. Clarity:

- Answers should be concise, professional, and user-friendly. When the bot doesn't know something, it should indicate uncertainty, e.g., 'Sorry, I can't help with that.'

5. Documentation:

- Each bot must be well documented, including functionality, methods, and update process.