

User Manual (task-oriented walkthroughs)

Video Walkthrough Link

Example 1. User Question: What are my remaining classes to graduate?

The screenshot shows the AdvisorAI chat interface. On the left is a dark blue sidebar with the AdvisorAI logo at the top. Below the logo is a 'Helpful Resources' section with links to Career & Talent Development, FIU Academic Calendar, CASE Drop-In Advising, Undergraduate Advising, and Graduate Advising. Below that is a 'Help us Improve!' section with links to Make a Suggestion, Report a Problem, and Leave a Review. At the bottom of the sidebar is a 'New Chat' button and the FIU Florida International University logo.

The main chat area has a light gray background. At the top right, there is a text input field with the question 'What are my remaining classes to graduate?' and a 'Send' button. Below the input field, a file upload icon and the text 'nicolas_PDA (1).pdf' are visible.

The AI's response is displayed in a white box with a light gray border. It begins with 'Based on your Panther Degree Audit, you are close to completing your degree requirements, but you have a few classes and credit requirements left to fulfill.' This is followed by a section titled 'You currently have courses in progress:' which lists several courses including CAI4002 Artificial Intelligence, COS3095 Technology in the Global Arena, COP4555 Prin Of Prog Lang, COT3541 Logic For Comp Sci, STA3033 Intro to Prob & Stats for CS, DIS3038 Honors Seminar V, COP4710 Database Management, BSC2011 General Biology II, BSC2011L Gen Biology Lab II, CEN4010 Software Eng I, CIS4951 Capstone II, and COP4338 Systems Programming.

Next is a section titled 'Recommended next courses:' which includes 'Credits Remaining: 6 additional credits to reach the 120 credits required for graduation.', 'Core Courses Needed: Finish either CNT4713 Net-centric Computing or COP4610 Operating Syst Princ.', and 'Elective Course Requirement: Complete any remaining elective courses not yet addressed from the Computer Science BS electives.'

Following this is 'Remaining requirements:' which states 'Honors Curriculum: 3 additional Honors credits needed to satisfy the Honors College requirements.'

Then is 'Next steps:' which lists three steps: 'Consult with your academic advisor to confirm your audit and plan for the completion of your remaining courses.', 'Consider options for fulfilling additional credit requirements, such as elective courses that fit your schedule and interests.', and 'Verify you are meeting all the Honors College obligations and graduation deadlines.'

The response concludes with 'You're on a promising path. Ensure to follow through with any in-progress courses to'.

At the bottom of the chat area is a dark blue input field with the placeholder text 'Ask a question...' and a 'Send' button.

Example 2. User Question: What sequence of classes do I need to take COP4555

The screenshot shows the AdvisorAI chat interface. On the left is a dark blue sidebar with the AdvisorAI logo at the top. Below the logo is a 'Helpful Resources' section with links to Career & Talent Development, FIU Academic Calendar, CASE Drop-In Advising, Undergraduate Advising, and Graduate Advising. Below that is a 'Help us Improve!' section with links to Make a Suggestion, Report a Problem, and Leave a Review. At the bottom of the sidebar is a 'New Chat' button and the FIU Florida International University logo.

The main chat area has a light gray background. At the top right, there is a text input field with the question 'What sequence of classes do I need to take COP4555' and a 'Send' button.

The AI's response is displayed in a white box with a light gray border. It begins with 'Credits Remaining: 6 additional credits to reach the 120 credits required for graduation.' This is followed by 'Core Courses Needed: Finish either CNT4713 Net-centric Computing or COP4610 Operating Syst Princ.' and 'Elective Course Requirement: Complete any remaining elective courses not yet addressed from the Computer Science BS electives.'

Next is a section titled 'Remaining requirements:' which states 'Honors Curriculum: 3 additional Honors credits needed to satisfy the Honors College requirements.'

Then is 'Next steps:' which lists three steps: 'Consult with your academic advisor to confirm your audit and plan for the completion of your remaining courses.', 'Consider options for fulfilling additional credit requirements, such as elective courses that fit your schedule and interests.', and 'Verify you are meeting all the Honors College obligations and graduation deadlines.'

The response concludes with 'You're on a promising path. Ensure to follow through with any in-progress courses to complete your degree requirements fully. Let me know if you have any other questions!'.

Below the response, there is a section titled 'To enroll in COP4555 - Programming Languages, you need to complete the prerequisites in the following order:' followed by a numbered list: '1. COP2210 - Programming I: You have already completed this course.', '2. COP3337 - Programming II: You have already completed this course.', and '3. COP3530 - Data Structures: You have already completed this course.'

Below the list, there is a paragraph: 'Since you have completed all the necessary prerequisites, you are eligible to take COP4555 at this time. If you have more questions or need further guidance, please let me know!'.

At the bottom of the chat area is a dark blue input field with the placeholder text 'Ask a question...' and a 'Send' button.

Example 3. User Question: What scholarships am I eligible for?

AdvisorAI

Helpful Resources

- Career & Talent Development
- FIU Academic Calendar
- CASE Drop-In Advising
- Undergraduate Advising
- Graduate Advising

Help us Improve!

- Make a Suggestion
- Report a Problem
- Leave a Review

New Chat

FIU
FLORIDA
INTERNATIONAL
UNIVERSITY

To enroll in COP4555 - Programming Languages, you need to complete the prerequisites in the following order:

- COP2210 - Programming I: You have already completed this course.
- COP3337 - Programming II: You have already completed this course.
- COP3530 - Data Structures: You have already completed this course.

Since you have completed all the necessary prerequisites, you are eligible to take COP4555 at this time. If you have more questions or need further guidance, please let me know!

Eligibility for scholarships at FIU can vary based on several factors including academic merit, departmental criteria, financial need, and specific program requirements. Here's how you can determine which scholarships are available to you:

Key Details:

- Scholarship Types:** FIU provides merit-based, departmental, first-generation, external, and prestigious scholarships, such as Bright Futures.
- Search and Application:** Use the FIU Scholarship Search platform to find scholarships relevant to your qualifications. It allows you to filter scholarships based on your specific criteria.
- Application Timing:** The primary application period is between January and March, though scholarships can be available year-round.

Resources: For detailed information and to explore scholarship opportunities, visit the [Office of Scholarships website](#). You can also contact them at scholarships@fiu.edu or 305-348-0349.

Contact the Office of Scholarships if you need personalized guidance or have further questions.

What sequence of classes do I need to take COP4555

What scholarships am I eligible for?

Ask a question...

Example 4. User Question: Can I add a math double major?

AdvisorAI

Helpful Resources

- Career & Talent Development
- FIU Academic Calendar
- CASE Drop-In Advising
- Undergraduate Advising
- Graduate Advising

Help us Improve!

- Make a Suggestion
- Report a Problem
- Leave a Review

New Chat

FIU
FLORIDA
INTERNATIONAL
UNIVERSITY

Resources: For detailed information and to explore scholarship opportunities, visit the [Office of Scholarships website](#). You can also contact them at scholarships@fiu.edu or 305-348-0349.

Contact the Office of Scholarships if you need personalized guidance or have further questions.

Adding a double major in mathematics to your existing Computer Science (CS) degree program is a viable option and can complement your studies well, especially given the strong math requirements of the CS degree. Here are some considerations to keep in mind:

Key Details:

- Math and CS Overlap:** Your CS degree already requires courses such as Calculus I, Calculus II, and Probability and Statistics which may count toward math major requirements.
- Minors and Additional Majors:** While your CS major has specific course requirements, you may use some electives to start fulfilling mathematics major requirements.
- Advising and Graduation Timeline:** Adding a second major can extend your time to graduation unless you plan carefully to combine requirements.

Next Steps:

- Contact your academic advisor to discuss the specifics of adding a mathematics double major. They can provide tailored advice based on your completed coursework and remaining degree requirements.
- Review the mathematics degree requirements to determine additional courses needed and how they fit into your current schedule.

To explore this option further and begin the process, reach out to the advising office appropriate for your college or department. They can offer more detailed guidance tailored to your situation. Let me know if you have any other questions!

Can I add a math double major?

Ask a question...

Example 5. User Question: What could hinder my graduation?



Helpful Resources

- Career & Talent Development
- FIU Academic Calendar
- CASE Drop-In Advising
- Undergraduate Advising
- Graduate Advising

Help us Improve!

- Make a Suggestion
- Report a Problem
- Leave a Review

New Chat



- Review the mathematics degree requirements to determine additional courses needed and how they fit into your current schedule.

To explore this option further and begin the process, reach out to the advising office appropriate for your college or department. They can offer more detailed guidance tailored to your situation. Let me know if you have any other questions!

Several factors could potentially delay or hinder your graduation. Here are key areas to be mindful of:

Satisfactory Academic Progress (SAP): Falling below the required academic progress standards could affect your eligibility for financial aid, which in turn might impact your ability to complete courses needed for graduation.

Graduation Deadlines: Missing critical application deadlines for graduation could postpone your graduation date. It's important to apply for graduation within the timelines specified by FIU.

Honors College Requirements: If you are part of the Honors College, failing to maintain continuous engagement, not earning the required Honors credits, or not achieving a B average in Honors courses could prevent you from meeting graduation requirements.

Key Details:

- GPA Maintenance:** A minimum cumulative GPA of 3.3 is required if graduating through the Honors College.
- Credit Completion:** Ensure you have completed all degree requirements, including total credits needed and required courses.

Resources: For more information and specific guidance, consider visiting the [Honors College Graduation Requirements webpage](#).

Stay proactive by regularly consulting with your academic advisor to ensure you are on track to meet all graduation requirements. Let me know if there's anything else you'd like to know!

What could hinder my graduation?

Ask a question...

Example 6. User Question: What can speed up my graduation?



Helpful Resources

- Career & Talent Development
- FIU Academic Calendar
- CASE Drop-In Advising
- Undergraduate Advising
- Graduate Advising

Help us Improve!

- Make a Suggestion
- Report a Problem
- Leave a Review

New Chat



Resources: For more information and specific guidance, consider visiting the [Honors College Graduation Requirements webpage](#).

Stay proactive by regularly consulting with your academic advisor to ensure you are on track to meet all graduation requirements. Let me know if there's anything else you'd like to know!

To accelerate your path to graduation, consider the following strategies:

Plan Flexible Course Scheduling: Utilize online and summer course offerings to fit more credits into your schedule without overloading during the traditional semesters. This can help you complete requirements faster.

Leverage Honors Options: Use the flexibility within the Honors College by selecting seminar sequences and other Honors options that fit your schedule and interests, and that might be offered in expedited formats like study abroad or ARCH sequences.

Monitor Credits and Requirements: Make sure you have a clear understanding of the total credits and specific course requirements needed, and keep in close contact with your academic advisor to quickly address any unexpected gaps or issues.

Key Details:

- Honors Flexibility:** Take advantage of Honors College offerings that fit different formats (e.g., seminars, online courses, study abroad) to meet requirements efficiently.
- GPA Maintenance:** Continue to maintain the necessary GPA to ensure smooth continuation through your program.

Resources: More details on the options available can be found on the [Honors College Graduation Requirements webpage](#).

Consult your academic advisor regularly to customize a plan that aligns with your graduation goals. Let me know if there's anything more specific you'd like to explore!

What can speed up my graduation?

Ask a question...

B. Admin/Operations Guide (runbook, on-call, backup/restore)

Frontend/Client

i. NodeJS

In order to install the required packages and dependencies and execute the underlying JavaScript on your local machine, you must first install **NodeJS**.

NodeJS Install Link: <https://nodejs.org/en/download>

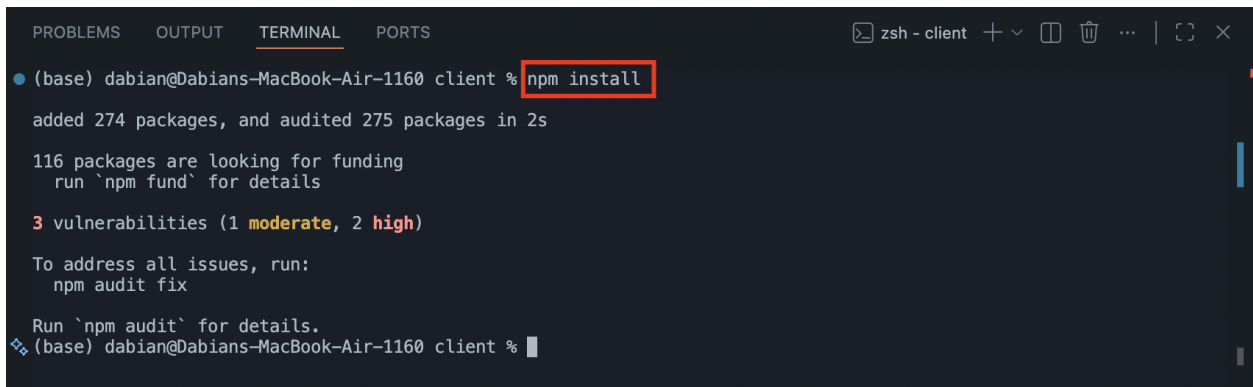
NodeJS Install Tutorial: [How to Install Node.JS and NPM on Windows](#)

ii. Node Package Manager (npm)

After installing NodeJS, it will allow you to make use of the Node Package Manager in order to install all the node packages contained within package-lock.json

npm Site: <https://www.npmjs.com/>

In order to install these packages, cd to **client** directory of the project and enter “**npm install**”

A screenshot of a terminal window with a dark background. The terminal title bar shows 'zsh - client' and standard window controls. The prompt is '(base) daban@Dabians-MacBook-Air-1160 client %'. The command 'npm install' is entered and highlighted with a red rectangle. The output shows: 'added 274 packages, and audited 275 packages in 2s', '116 packages are looking for funding', '3 vulnerabilities (1 moderate, 2 high)', and instructions to run 'npm audit fix' and 'npm audit' for details. The prompt returns to '(base) daban@Dabians-MacBook-Air-1160 client %' at the bottom.

```
PROBLEMS  OUTPUT  TERMINAL  PORTS
(base) daban@Dabians-MacBook-Air-1160 client % npm install
added 274 packages, and audited 275 packages in 2s

116 packages are looking for funding
  run `npm fund` for details

3 vulnerabilities (1 moderate, 2 high)

To address all issues, run:
  npm audit fix

Run `npm audit` for details.
(base) daban@Dabians-MacBook-Air-1160 client %
```

ii. Vite

This project frontend is built using the Vite framework. In order to run and access the frontend code to be compiled and displayed on localhost, the terminal should again be accessed to enter “**npm run dev**”


```
PROBLEMS OUTPUT TERMINAL PORTS
node - client + v [] [] ... | [] x

o (base) daban@Dabians-MacBook-Air-1160 client % npm run dev

> client@0.0.0 dev
> vite

VITE v7.3.1 ready in 777 ms

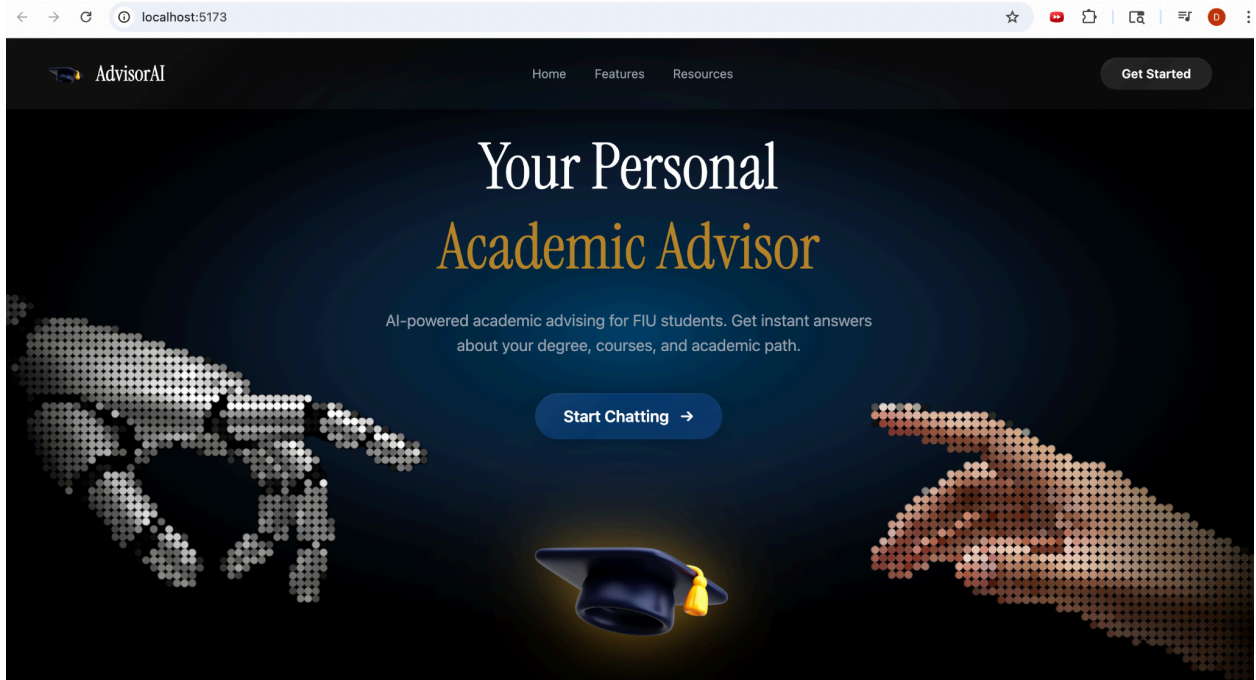
→ Local: http://localhost:5173/
→ Network: use --host to expose
→ press h + enter to show help
```

Note that this script command can be updated by editing the **package.json** file

```
package.json x
client > package.json > ...
1 {
2   "name": "client",
3   "private": true,
4   "version": "0.0.0",
5   "type": "module",
6   "scripts": {
7     "dev": "vite",
8     "build": "tsc -b && vite build",
9     "lint": "eslint .",
10    "preview": "vite preview"
11  },
12  "dependencies": {
13    "react": "^19.2.0",
14    "react-dom": "^19.2.0",
15    "react-markdown": "^10.1.0",
16    "react-router-dom": "^7.13.1"
17  },
18  "devDependencies": {
19    "@eslint/js": "^9.39.1",
20    "@tailwindcss/vite": "^4.2.1",
21    "@types/node": "^24.10.1",
22    "@types/react": "^19.2.7",
23    "@types/react-dom": "^19.2.3",
24    "@vitejs/plugin-react": "^5.1.1",
25    "eslint": "^9.39.1",
26    "eslint-plugin-react-hooks": "^7.0.1",
27    "eslint-plugin-react-refresh": "^0.4.24",
28    "globals": "^16.5.0",
29    "typescript": "^5.9.3",
30    "typescript-eslint": "^8.48.0",
31    "vite": "^7.3.1"
32  }
33 }
```

iii. Viewing Frontend

Follow the link that is provided after running the dev command (cmd + click), this is typically going to be ran on port 5173



iv. Components, Hooks and Pages Folders

The frontend is organized so day-to-day content updates can be made without touching the routing or data flow. The key files are:

client/src/main.tsx: boots the React app and loads global styles.

client/src/App.tsx: defines the top-level routes for the landing page and chat page.

client/src/pages/HomePage.tsx: composes the public landing page.

client/src/pages/ChatPage.tsx: wires the chat layout, sidebar state, and new-chat resets.

Landing Page

For landing-page content, update components under **client/src/components/HomePage/**:

Navbar.tsx: top navigation and the entry point into chat.

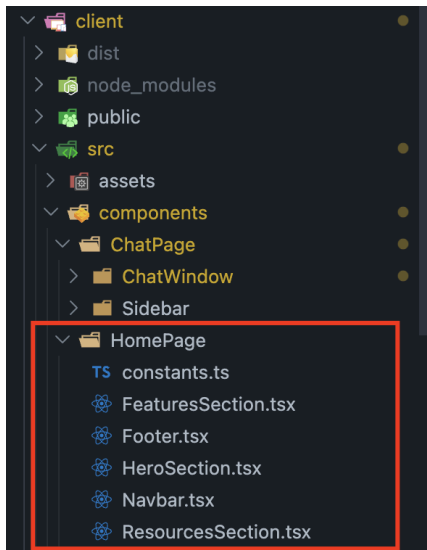
HeroSection.tsx: main banner and call to action.

FeaturesSection.ts: the three-step product explanation.

ResourcesSection.tsx: FIU resource links shown on the homepage.

Footer.tsx: footer branding.

constants.ts: shared data used by the homepage sections.



ChatPage

For chat content, update comps under **client/src/components/ChatPage/ChatWindow/**:

ChatWindow.tsx: orchestrates the pre-chat screen, message view, file upload flow, and chat composer.

PreChatScreen.tsx: the first screen users see before sending a message.

ChatMessages.tsx: renders user and AI messages, including markdown responses.

InputBox.tsx: the text composer and send/attach controls.

FileUploadPopover.tsx: file selection modal for degree audits and supporting documents.

DegreeAuditModal.tsx and **DegreeAuditGuide.tsx**: guided upload path for degree audits.

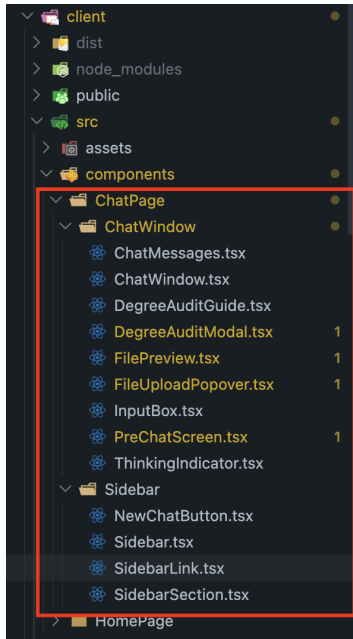
FilePreview.tsx: shows the selected file before it is sent.

ThinkingIndicator.tsx: loading state while the backend response is pending.

Sidebar.tsx: responsive sidebar shell and resource section layout.

SidebarSection.tsx and **SidebarLink.tsx**: reusable link groups for resources and feedback links.

NewChatButton.tsx: resets the active conversation.



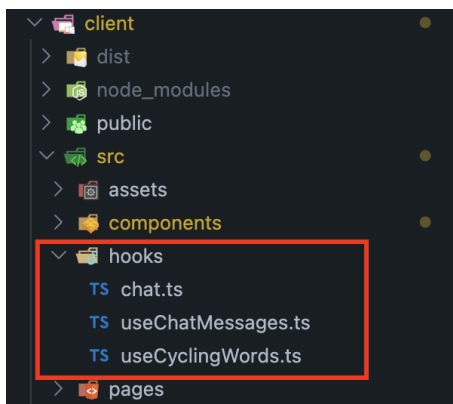
States / Request Logic

State and request logic live in `client/src/hooks/`:

useChatMessages.ts: manages the conversation state, starts new chats, and sends messages to `http://localhost:5099/api/Chat`.

useCyclingWords.ts: rotates the pre-chat prompt words on the landing chat screen.

chat.ts: shared chat constants such as the cycling words and timing interval.



Backend/Server

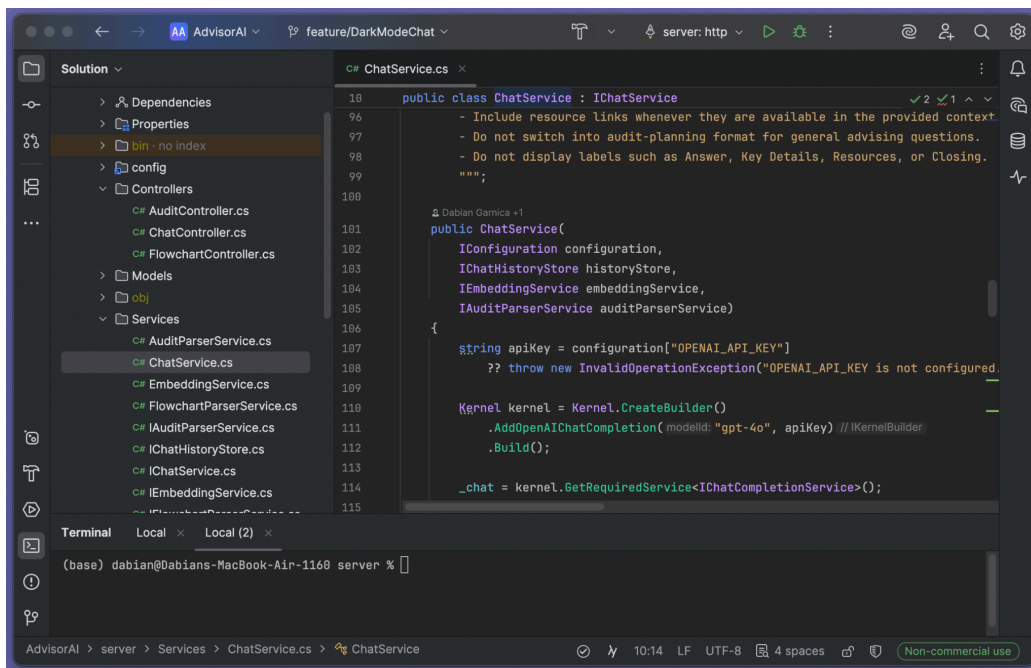
i. .NET SDK

This project backend is built with ASP.NET Core on .NET 10. To work on it locally, install the .NET SDK for your machine and use a .NET-compatible IDE such as Rider or Visual Studio.

.NET Download Link: <https://dotnet.microsoft.com/en-us/download>

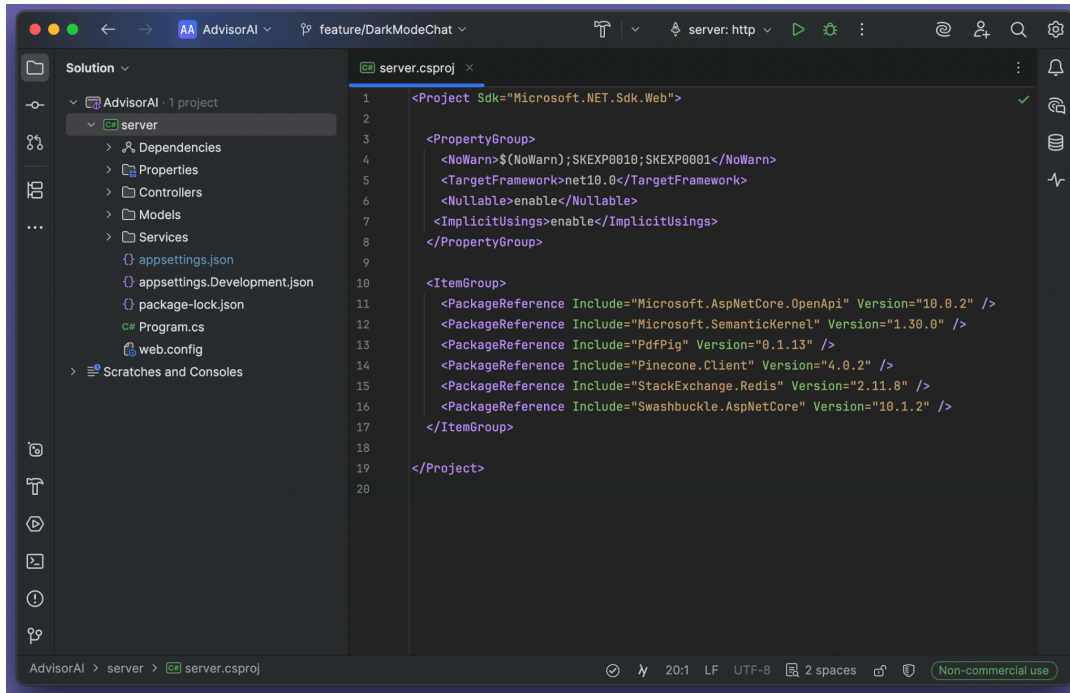
Visual Studio Download Link: <https://visualstudio.microsoft.com/downloads/>

Rider Download Link: <https://www.jetbrains.com/rider/download/>



ii. NuGet Packages

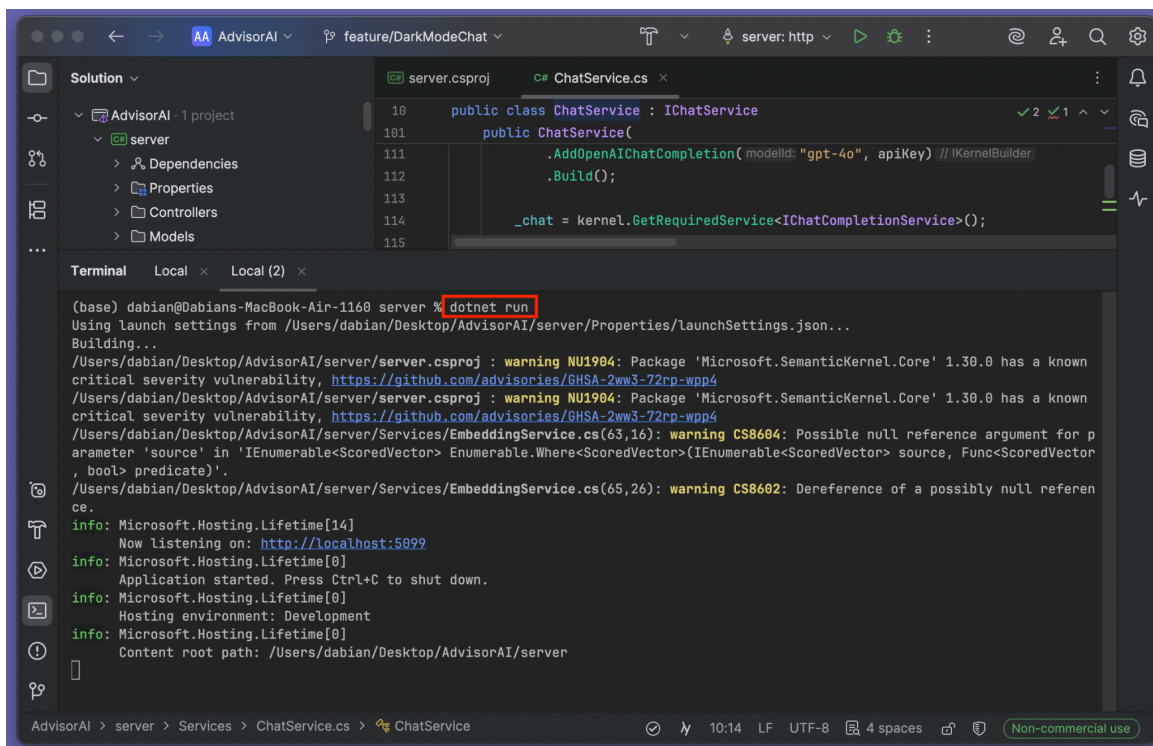
The server dependencies are managed through `server.csproj`. When you open the project in your IDE or run it from the terminal, NuGet packages such as Semantic Kernel, PdfPig, Pinecone, and StackExchange.Redis are restored automatically. If needed, run “**dotnet restore**” from the **server** directory.



iii. Running the Server

Navigate to the **server** directory and run “**dotnet run**”.

The API is configured to run locally on <http://localhost:5099>



iv. Configuration Files

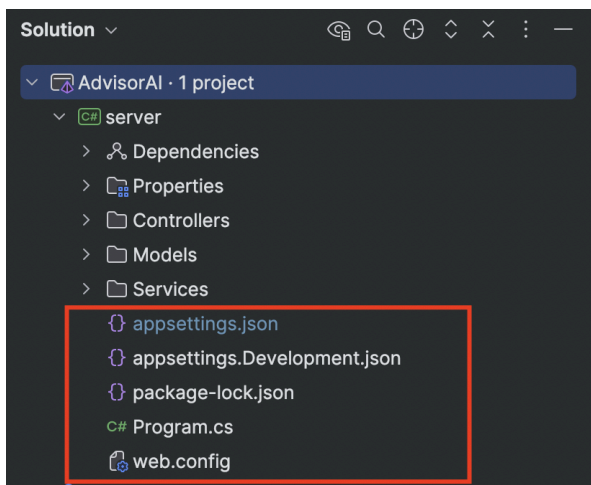
The main runtime configuration lives in **appsettings.json**, **appsettings.Development.json**, **Properties/launchSettings.json**, and **Program.cs**.

Program.cs: registers OpenAI embeddings, Pinecone, Redis-backed history, CORS, controllers, and Swagger.

appsettings.json: stores runtime settings and API keys required by the backend.

appsettings.Development.json: local development overrides.

launchSettings.json: defines the local run profiles and ports.



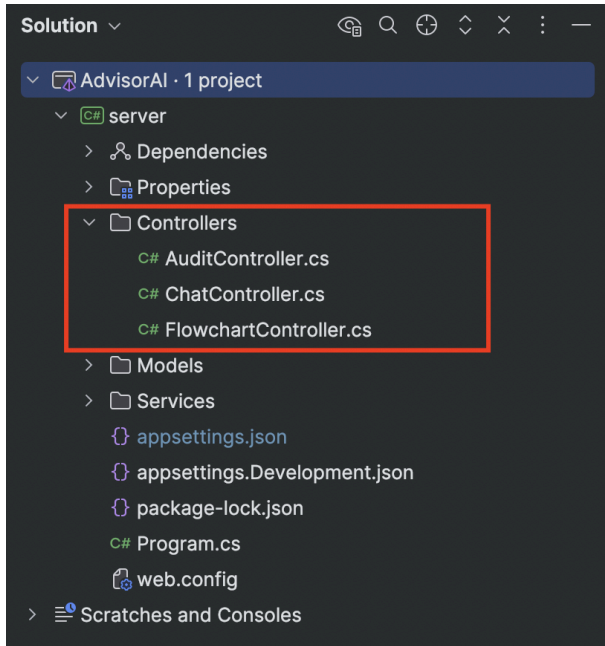
v. Controllers

The **server/Controllers/** folder contains the HTTP endpoints exposed by the backend:

ChatController.cs: handles `api/Chat` requests from the frontend chat UI and returns AI responses.

AuditController.cs: handles degree audit uploads and returns parsed audit data.

FlowchartController.cs: handles flowchart image uploads and stores parsed curriculum data.



vi. Services Folder

The **server/Services/** folder contains the application logic used by the controllers:

ChatService.cs: builds the AI prompt, merges chat context, and sends the final request to the model.

AuditParserService.cs: parses degree audit PDFs into structured audit data.

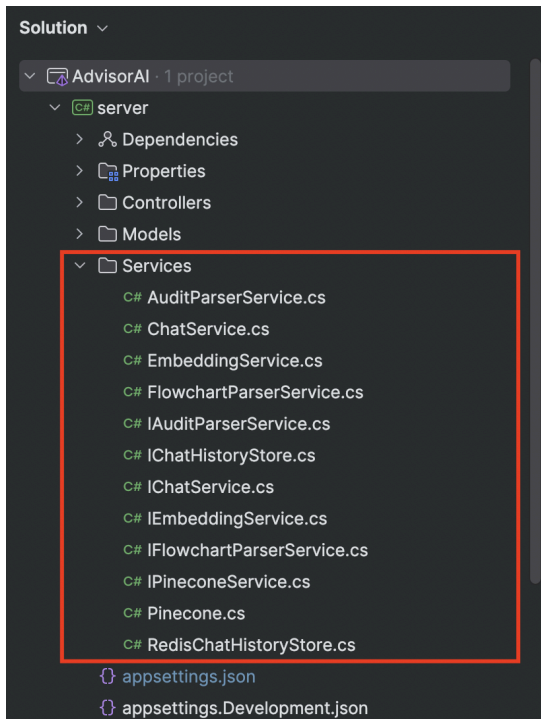
FlowchartParserService.cs: parses flowchart images into curriculum data.

EmbeddingService.cs: retrieves and stores degree requirement context used by chat responses.

RedisChatHistoryStore.cs: stores conversation history in Redis so chats remain stateful.

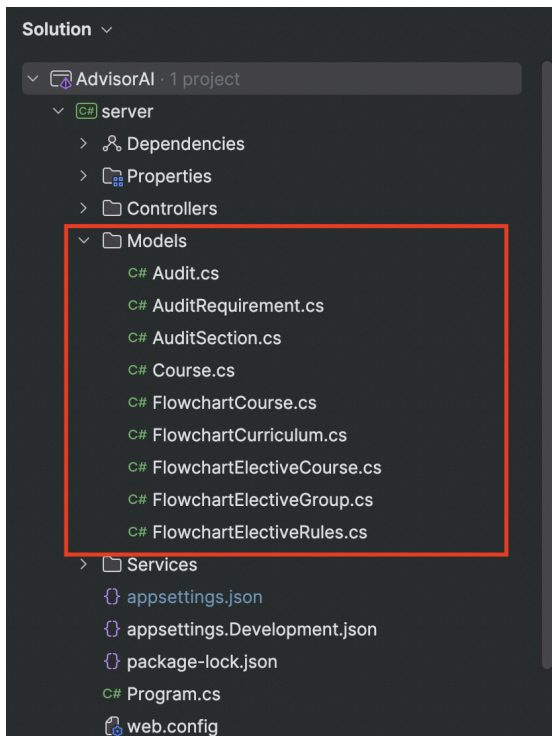
Pinecone.cs: configures the Pinecone client used for retrieval.

I*.cs interfaces: define the contracts used by the backend services.



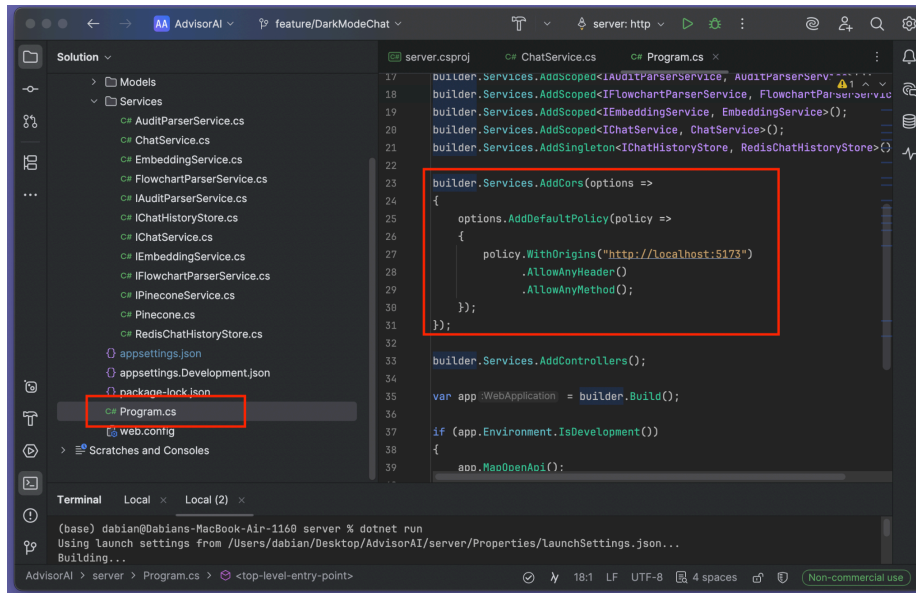
vii. Models Folder

The **server/Models/** folder contains the data structures returned and consumed by the backend endpoints, including audit, course, and flowchart-related models.



viii. Backend Notes

- The frontend chat screen expects the server to be running before it can send messages successfully.
- Keep the CORS origin in `Program.cs` aligned with the frontend development port if that port changes.



- If you update the AI behavior, edit `ChatService.cs` first, then verify the controllers still return the shape the frontend expects.

C. API Reference (OpenAPI/GraphQL schema)

The following breaks down the backend OpenAPI document into a readable reference for the purpose of documentation

Source

Runtime endpoint: <http://localhost:5099/openapi/v1.json>

Swagger UI: <http://localhost:5099/swagger>

Generated from server/Program.cs when the app runs in Development mode

Endpoints

POST /api/Audit/upload

Purpose: Upload a degree audit file and return parsed data.

Request (multipart/form-data):

Field	Type	Required
file	IFormFile (binary)	Yes

Response (200 OK):

Content Type	Return Type
application/json	Audit

POST /api/Chat for the advising chat experience

Purpose: Send message to AI Advising service

Request (multipart / form data):

Field	Type	Required
message	string	Yes
conversationId	string (UUID)	Yes
fileType	string	No
file	IFormFile (binary)	No

Response (200 OK):

Content Type	Return Type
application/json	Audit

POST /api/flowchart for parsing flowchart images and updating curriculum data

Purpose: Upload flowchart images and extract curriculum data to be used for embeddings

Request (multipart / form data):

Field	Type	Required
files	IFormFile[] (binary array)	Yes

Response (200 OK):

Content Type	Return Type
application/json	FlowchartCurriculum

Schemas

Audit

Property	Type
Sections	AuditSection[]

AuditSection

Property	Type
title	string
isSatisfied	boolean
requirements	AuditRequirement[]

AuditRequirement

Property	Type
title	string
isSatisfied	boolean
coursesTaken	Course[]

Course

Property	Type
code	string
title	string

FlowchartCurriculum

Property	Type
program	string
courses	FlowchartCourse[]
electiveRules	FlowchartElectiveRules

FlowchartCourse

Property	Type
code	string
title	string
credits	integer string
prerequisites	object
corequisites	object
category	string
standingRequirement	string
notes	string

FlowchartElectiveRules

Property	Type
totalElectivesRequired	integer string null

electiveGroups	FlowchartElectiveGroup[]
----------------	---------------------------

FlowchartElectiveGroup

Property	Type
name	string
groupMinimumRequired	integer string null
courses	FlowchartElectiveCourse[]

FlowchartElectiveCourse

Property	Type
name	string
prerequisites	object
corequisites	object

IFormFile

Property	Type
type	string (binary)