

Creating Your Own Podcast with the Help of AI.

(Thomas) Sebastian Jensen





(Thomas) Sebastian Jensen

SENIOR SOFTWARE ENGINEER



[@tsjdevapps](https://twitter.com/tsjdevapps)



[thomassebastianjensen](https://www.linkedin.com/in/thomassebastianjensen)



jensen@medialesson.de



medium.com/@tsjdevapps | tsjdev-apps.de



Introduction



Azure OpenAI Service

**GPT-4o Mini, TTS,
DALL-E-3**

**Multimodal
input and output**

**Fast response
times**

**Safe by
design**



Azure OpenAI Service

- Your prompts (inputs) and completions (outputs), your embeddings, and your training data:
 - are NOT available to other customers.
 - are NOT available to OpenAI.
 - are NOT used to improve OpenAI models.
 - are NOT used to train, retrain, or improve Azure OpenAI Service foundation models.
 - are NOT used to improve any Microsoft or 3rd party products or services without your permission.
 - Your fine-tuned Azure OpenAI models are available exclusively for your use.
- The Azure OpenAI Service is operated by Microsoft as an Azure service; Microsoft hosts the OpenAI models in Microsoft's Azure environment and the Service does NOT interact with any services operated by OpenAI (e.g. ChatGPT, or the OpenAI API).



OpenAI Models

	GPT-4o Mini	GPT-4o	GPT-4 Turbo	GPT-4	GPT-3.5 Turbo
Input Context Window	128k tokens	128K tokens	128k tokens	8k tokens	4k tokens
Maximum Output Tokens	16.4k tokens	16.4k tokens	4k tokens	8k tokens	4k tokens
Release Date	18.07.2024	13.05.2024	06.11.2023	14.03.2023	28.11.2022
Knowledge Cutoff	October 2023	October 2023	December 2021	September 2021	September 2021
Input Pricing	\$0.15 per million tokens	\$5.00 per million tokens	\$10.00 per million tokens	\$30.00 per million tokens	\$0.50 per million tokens
Output Pricing	\$0.60 per million tokens	\$15.00 per million tokens	\$30.00 per million tokens	\$60.00 per million tokens	\$1.50 per million tokens
MMMU Benchmark	59.4	69.1	-	34.9	-

MMMU - Massive Multi-discipline Multimodal Understanding

<https://context.ai/compare/gpt-4o/o1-preview-2024-09-12>



Podcastr – Console



Podcast

A **podcast** is a a **digital audio or video program** available for streaming or download. It covers a **wide range of topics** including news, storytelling, interviews, educational content, and entertainment. A podcast is **accessible anytime and anywhere**, making it convenient for on-the-go listening or viewing. It is **available on various platforms** such as Apple Podcasts, Spotify, Google Podcasts, and specialized podcast apps.



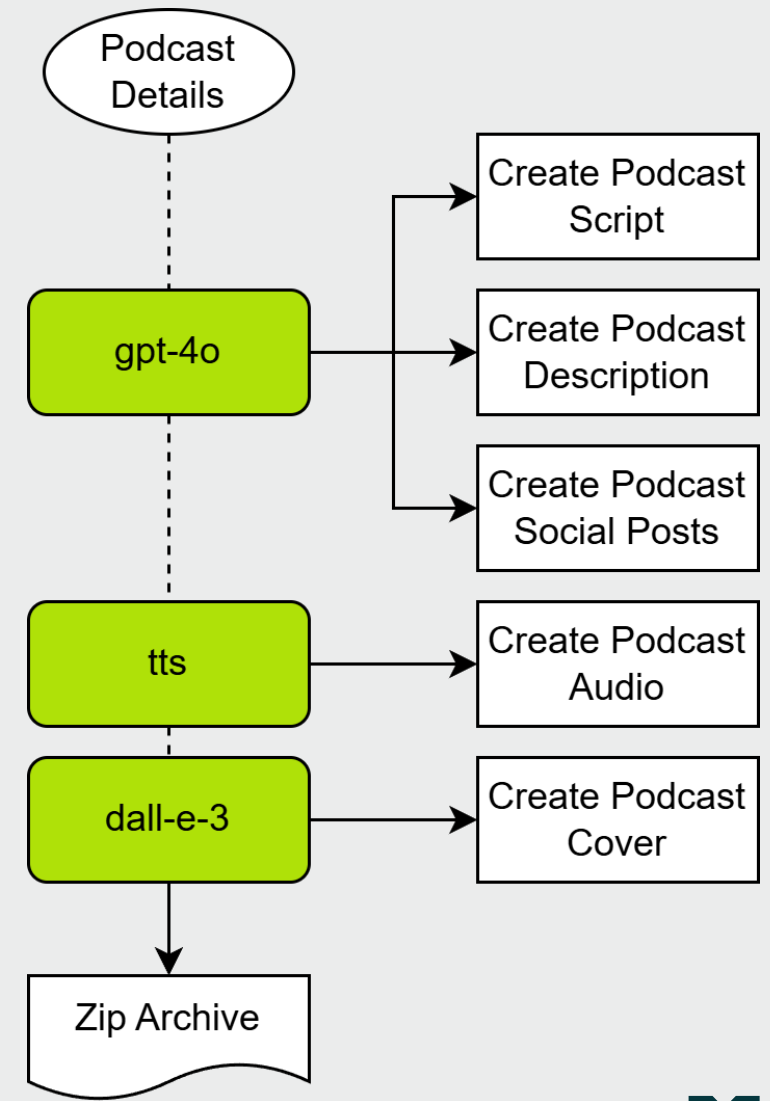
General Idea

- Get some details about a podcast episode from the user
- Create a script for a podcast episode
- Create a description of the podcast episode
- Create some social media posts about the podcast episode
- Create the audio file for the podcast episode
- Create a cover for the podcast episode
- Save everything in one zip archive



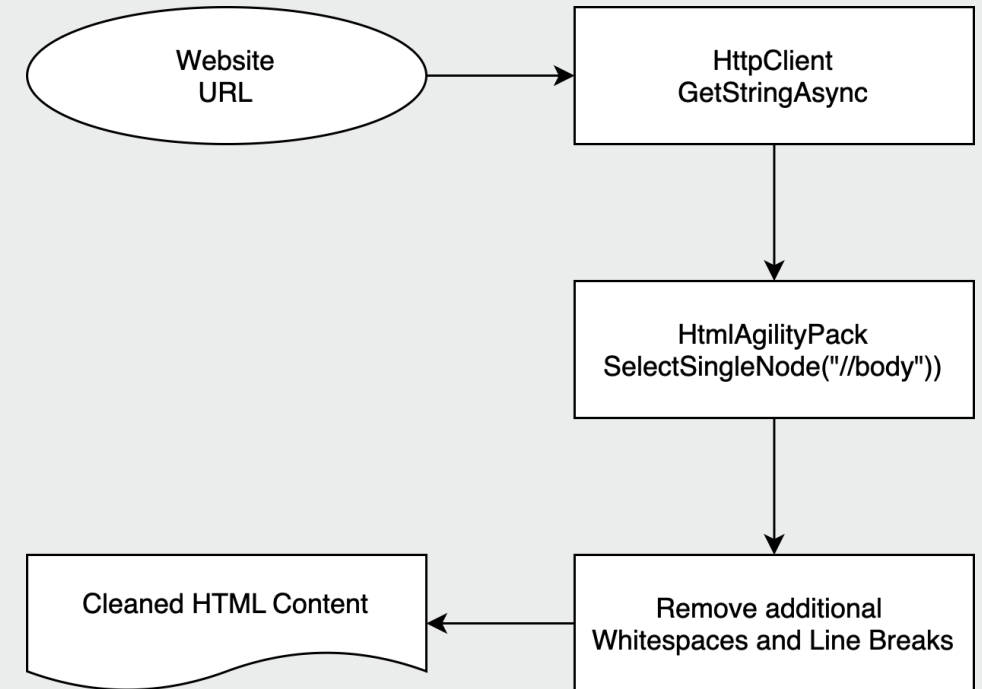
Multi Model Orchestration

- Integration of multiple AI models and C# logic
- GPT-4o (Mini) for Content Generation
- TTS for Audio Generation
- Dall-E-3 for Image Generation



How to Access Website Data?

- Idea: Get the content from a Medium blog post
- AI models cannot scrape websites directly
- Use a simple `HttpClient` to retrieve website content
- Utilize `HtmlAgilityPack` to extract the body of the website
- Clean up the HTML body to reduce the number of input tokens

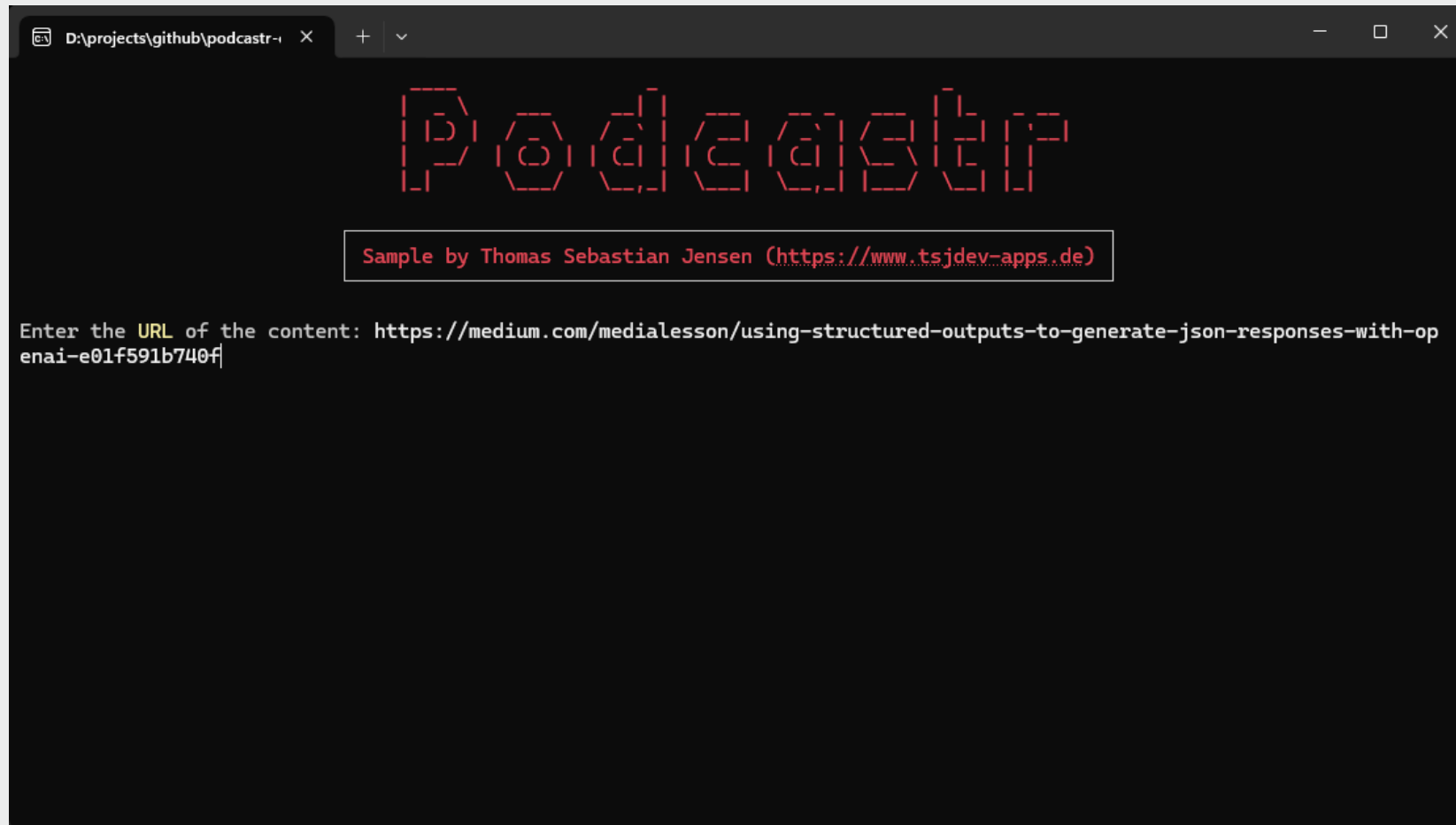


Multi Language and Voice Support

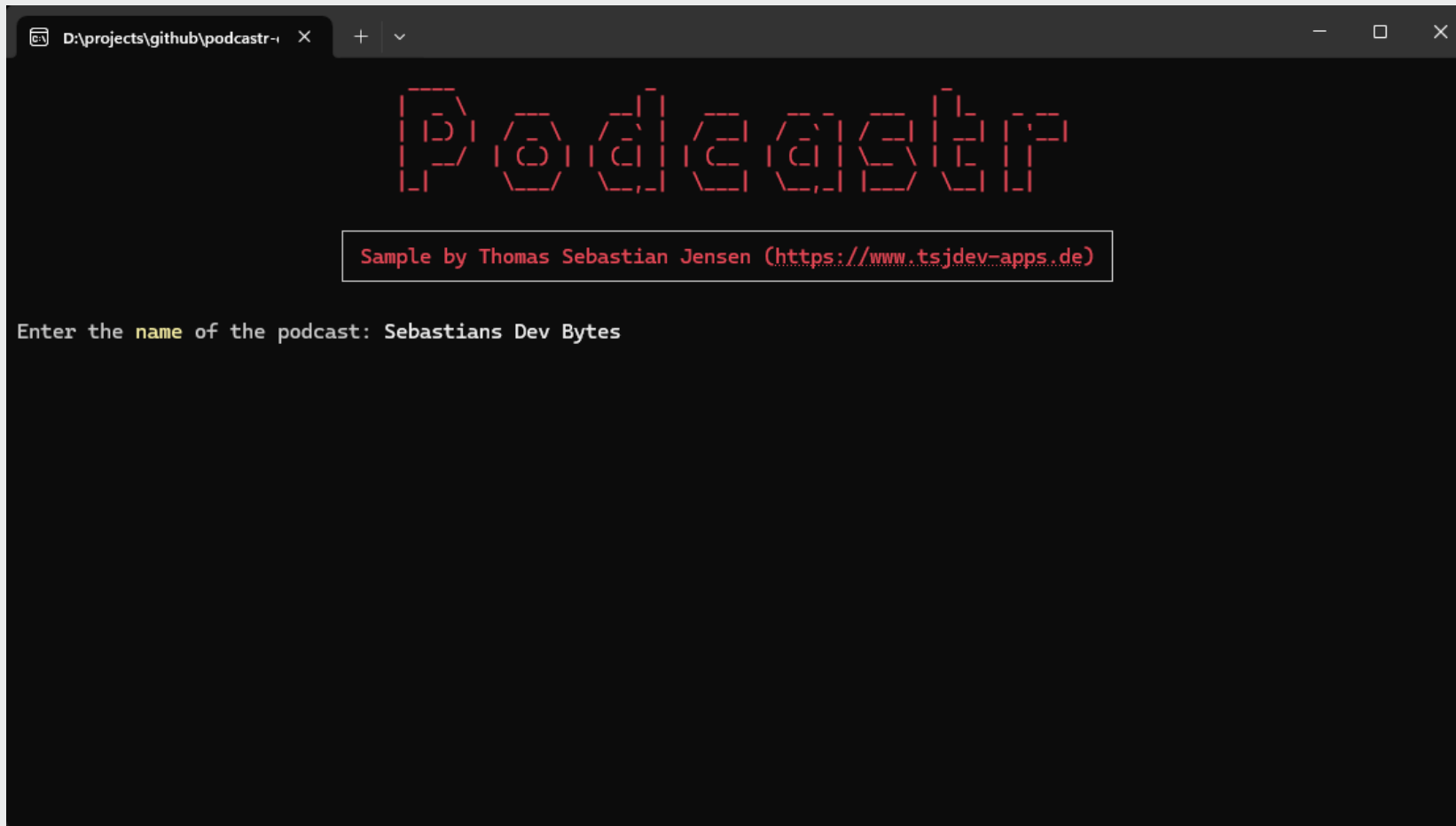
- Language of the podcast is not depending on the language of the content
 - Supports over 80 languages covering 97% of humanity
 - Maintains high translation speed and quality
 - Enhances global accessibility
-
- Six different voices are available
 - Voices are optimized for English, but able to speak all languages



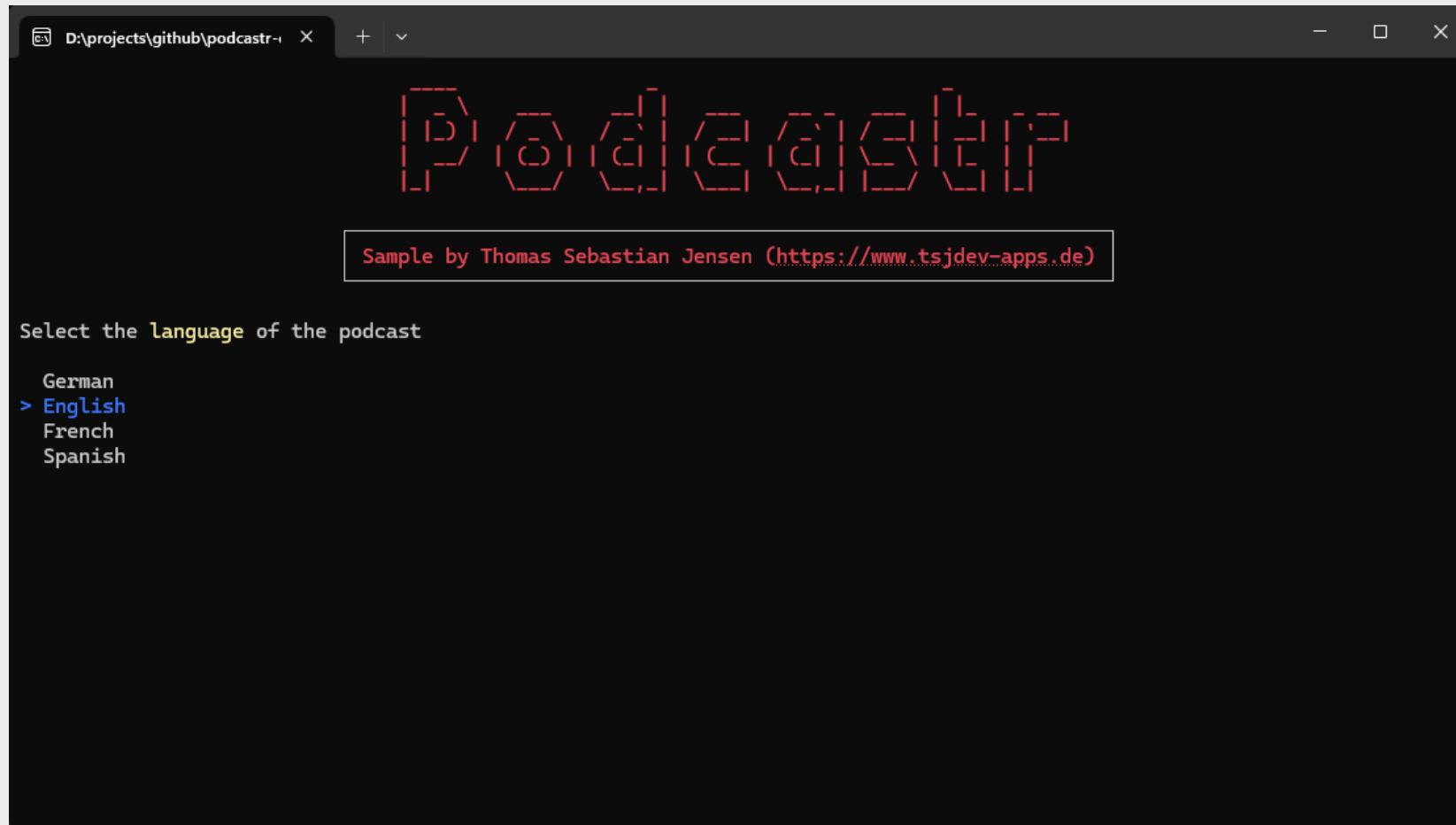
Content URL



Podcast Name



Podcast Language



The screenshot shows a terminal window with a dark background. At the top, the title bar indicates the file path 'D:\projects\github\podcastr-'. The main content area features the word 'Podcastr' in a large, red, dashed outline font. Below it, a red text box contains the attribution 'Sample by Thomas Sebastian Jensen (https://www.tsjdev-apps.de)'. The prompt 'Select the language of the podcast' is displayed in a light green font. A list of languages is shown: 'German', '> English' (highlighted in blue), 'French', and 'Spanish'.

```
D:\projects\github\podcastr- x + v

Podcastr

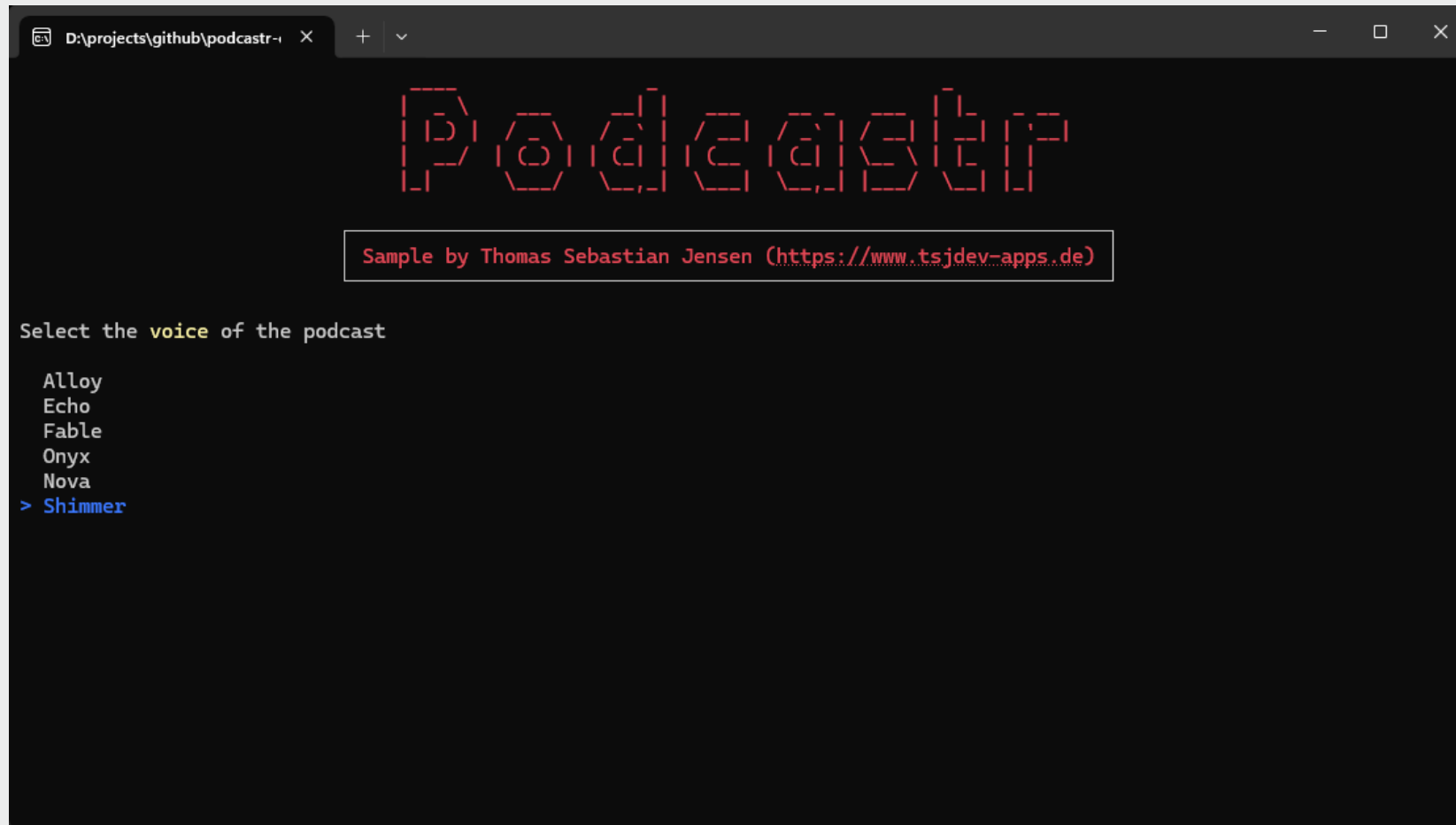
Sample by Thomas Sebastian Jensen (https://www.tsjdev-apps.de)

Select the language of the podcast

German
> English
French
Spanish
```



Podcast Voice



The screenshot shows a terminal window with a dark background. At the top, the window title bar indicates the file path 'D:\projects\github\podcastr-'. Below the title bar, the word 'Podcastr' is displayed in a large, red, dashed, monospace font. Underneath this, a red text box contains the attribution 'Sample by Thomas Sebastian Jensen (<https://www.tsjdev-apps.de>)'. The main part of the terminal shows a prompt 'Select the voice of the podcast' followed by a list of voice options: Alloy, Echo, Fable, Onyx, Nova, and Shimmer. The 'Shimmer' option is highlighted with a blue prompt character '>'.

```
D:\projects\github\podcastr- X + v

Podcastr

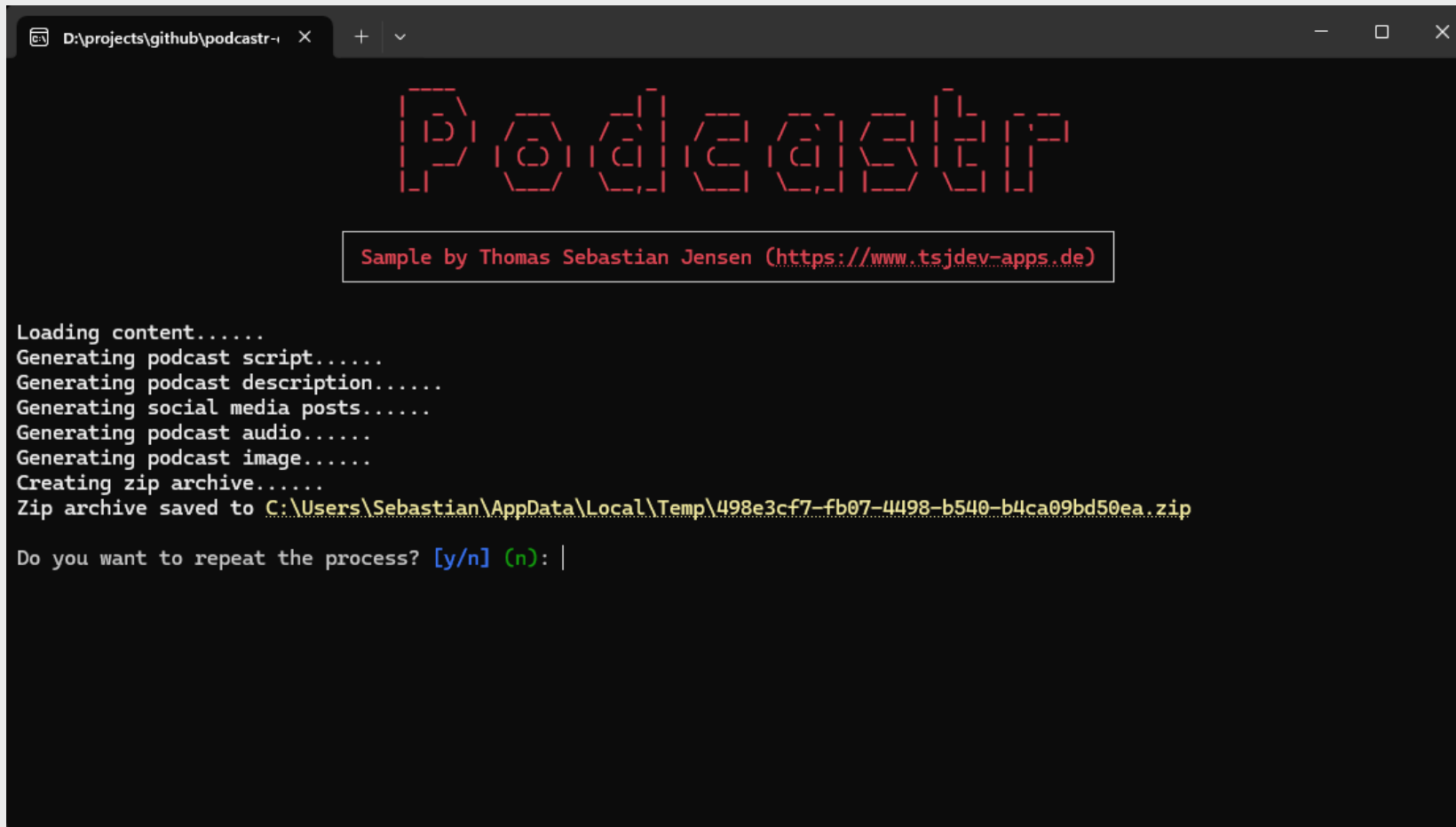
Sample by Thomas Sebastian Jensen (https://www.tsjdev-apps.de)

Select the voice of the podcast

Alloy
Echo
Fable
Onyx
Nova
> Shimmer
```



Podcast Generation



```
D:\projects\github\podcastr- x + v

Podcastr

Sample by Thomas Sebastian Jensen (https://www.tsjdev-apps.de)

Loading content.....
Generating podcast script.....
Generating podcast description.....
Generating social media posts.....
Generating podcast audio.....
Generating podcast image.....
Creating zip archive.....
Zip archive saved to C:\Users\Sebastian\AppData\Local\Temp\498e3cf7-fb07-4498-b540-b4ca09bd50ea.zip

Do you want to repeat the process? [y/n] (n): |
```



Results



Sebastian's Dev Bytes ✓
@sebastiansdevbytes

✨ Ready to change the way you code with AI? 🤖🚀

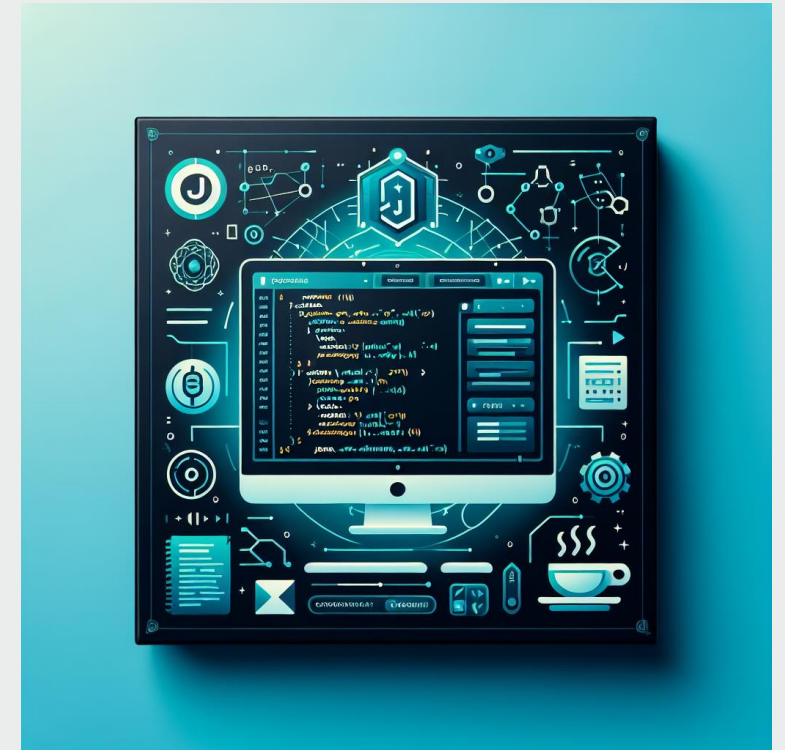
In the latest episode of "Sebastian's Dev Bytes," we dive into Structured Outputs with OpenAI!

Learn how to get structured responses in your .NET projects—no more messy data!

Tune in now! 🎧👉

#AI #OpenAI #DevBytes

2:30 PM · 12/03/24 Sebastian's Dev Bytes



Live Demo – Console Application



Future Prospects

- Currently the application is just a Proof of Concept
- Use **Function Calling** to let the AI decide if a website need to be crawled
- Use **Structured Outputs** to get a JSON structure from the AI containing the Script, the Description and the Social Media Posts
- Validate the audio file by transcribing it again using the `whisper-1` model and compare it to the original podcast script
- Upload the new podcast episode to the podcast hoster using an API
- Publish social media posts after the podcast episode has been uploaded and published



Podcastr – Blazor



Blazor WebAssembly

- Client-Side Execution: Runs entirely in the browser, reducing server load.
- Modern Architecture: Single-page application (SPA) framework.
- Flexible Hosting: Can be hosted on a CDN or static web servers.
- No Server Dependency: Fully autonomous execution after initial load.
- Initial Load Time: Larger download size due to WebAssembly payload.
- Debugging Challenges: Debugging WebAssembly in the browser can be more complex.
- Security Considerations: All app logic is exposed in the client.



Blazor Server Side Rendering

- Fast Load Time: Minimal initial payload; UI rendered on the server.
- Centralized Processing: Heavy computations are handled server-side.
- Easier Debugging: Traditional server-side debugging applies.
- Small Client Footprint: Lightweight client requirements.
- Network Dependency: Requires constant server connection via SignalR.
- Latency Issues: UI interactions depend on round trips to the server.
- Hosting Requirements: Requires a .NET-capable server



First Look

Podcastr

Podcastr

Content URL

Podcast Content


Podcast Name

Podcast Language

German

Podcast Voice

Alloy

 Generate Podcast

© 2024 by Thomas Sebastian Jensen



Live Demo – Blazor Application



Closing Remarks



■ CLOSING REMARKS

Conclusion

- Combine different AI models to maximize their potential
- Use the Azure.AI.OpenAI NuGet package to integrate OpenAI or Azure OpenAI.
- Make sure to use preview versions of the Azure.AI.OpenAI NuGet package
- Invest effort in crafting prompts to achieve optimal results from the AI models.
- Always review the output before publishing, as the AI may occasionally struggle with dates or other details.



Source Code of the Console Application

You will find the complete source code of the Podcastr Console application on GitHub.



github.com/tsjdev-apps/podcastr-console

tsjdev-apps / podcastr-console

<> Code

Issues

Pull requests

Actions

Projects

Wiki

Simple .NET 9 console application to create a Podcast using Azure OpenAI.

MIT license

0 stars

0 forks

1 watching

1 Branch

0 Tags

Activity

Private repository

main

Go to file

+

<> Code

tsjdev-apps

Merge pull request #3 from tsjdev-apps/featu...

4d4f700 · 6 hours ago

.github/workflows	Create dotnet.yml	7 hours ago
docs	Add generated podcast audio.	6 hours ago
src/Podcastr	Update Program.cs	7 hours ago
.gitignore	Initial commit	7 hours ago
LICENSE	Initial commit	7 hours ago
Podcastr.sln	Add basic solution.	7 hours ago
README.md	Fix typo.	6 hours ago

README

MIT license



Follow our adventures and learn more...

Our blog with free articles about AI, cloud and software engineering



medium.com/medialesson

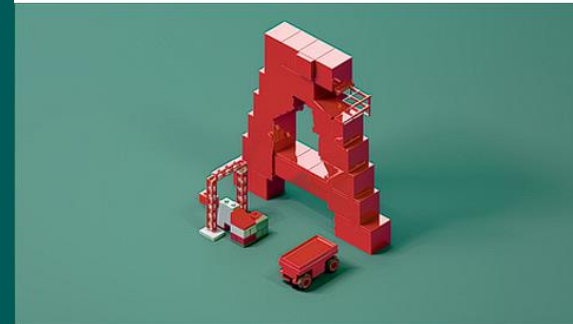


Using the Microsoft Information Protection SDK in a containerized .NET app

Reading and writing sensitivity labels becomes essential when handling confidential enterprise documents. The Microsoft Information...



Philipp Bauknecht
Oct 14 · 3 min read



Kick-start: Angular project with Nx

Creating a new Angular workspace is extremely easy thanks to its CLI tool. However, things can get a little bit more complicated when...



Diego Langerica Fuentes
Oct 31 · 8 min read



Using Structured Outputs to Generate JSON responses with OpenAI

Explore the Structured Outputs functionality within the Azure.AI.OpenAI NuGet package to generate JSON responses in your .NET AI project.



Sebastian Jensen
Oct 1 · 7 min read

Latest





MEDIALESSON

Digital Excellence
Delivered.