

# 生成AIで作る… 察しのいい自販機

ご購入はこちら

中村 雅之

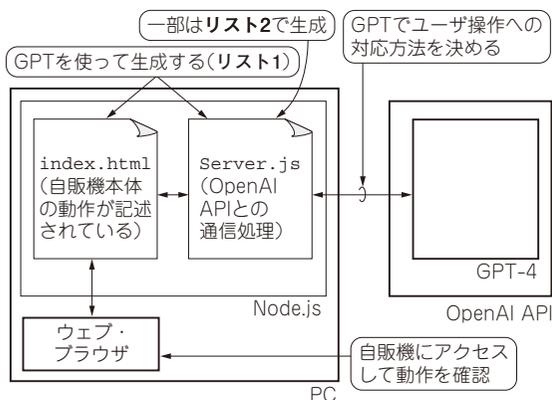


図1 生成AIを利用して作る自販機の全体像

実際に動作するシンプルで優しいUIを作ってみます。今回はウェブ・ブラウザで動作する自販機シミュレータを作ります(図1)。自販機といっても近年見られる全面LCDのものではなく、伝統的な押しボタンのタイプです。これに生成AIが組み込んだとき、どんなことが可能になるのかを探ってみます。

## ソースコードを生成する

せっかく生成AIを使うので、ソースコードも生成AIに書いてもらうことにします。しかし、これはちょっと大変でした。筆者が頑張って書いたソースコード生成プロンプトの一部をリスト1に示します。今回はウェブ・ブラウザで実行できるように、JavaScriptのソースコードを生成します。

このプロンプトを開発へ応用したい方は、次のリポジトリから取得して利用してください。

<https://github.com/petitroto/gai-vending-machine-sim>

このプロンプトから出力させたソースコードは不完全なので、最終的には筆者が修正して完成させました。完成したウェブ・アプリケーションもリポジトリに置いてあるので、手で動かしてみたい方はGitHubリポジトリから取得してください。

プロンプトを使った開発の手順について説明します。このプロンプトは有料版ChatGPTのGPT-4へ入力するか、OpenAIのAPIからGPT-4へリクエストを送ることで使用できます。本稿執筆時点の有料版ChatGPTは、入出力が長いと動作が不安定で、出力し終わる前にエラーで進まなくなることがあります。その場合はAPIを使用すると安定して出力されることがあります。

### ● ステップ1：クライアント側ソースコードの生成

リスト1(a)のプロンプトは、まずクライアント側のソースコードだけを出力させています。これは、一発目の出力は品質が十分でなく、何度か修正させる必要があるためです。ChatGPTがコードを出力し終わったら、このコードは与えた仕様を漏れなく満たすか確認してください。もし不足している仕様があれば修正してください。

と入力して、修正版を出力させてください。それを必要に応じて1回以上繰り返します。

### ● ステップ2：サーバ側ソースコードの生成

次に、サーバ側のソースコードを出力させます。プロンプトはリスト1(b)の通りです。サーバ側のソースコードについても、出力し終わったところで修正版を出力させます。クライアント側ほどではありませんが、必ず抜けや漏れがあります。

### ● ステップ3：ソースコードの配置

出力したサーバ側ソースコードを、server.jsというファイル名で保存します。また、クライアント側コードは、index.htmlとして保存し、server.jsの階層にpublicというフォルダを作って格納します。

最後に、server.jsがあるフォルダに、.envというファイルを作成します。あらかじめ、OpenAIのアカウントでAPIキーを発行しておき、次の形式でこのファイルに記載します。

OPENAI\_API\_KEY=XXXXXXXXXXXXXXXXXXXXXXXXXXXX