



## 仕様書

■ 作成者: パク・ジュンハン (朴 俊漢, PARK JUNHAN)

■ アプリ名: Amami - グルメサーチ  
あなたの周りのお店 (Anata no mawari no mise)

### ■ アプリ概要

Amami は、ユーザーの現在位置を基準に周辺のレストランを簡単に検索できるウェブアプリケーションです。ユーザーはさまざまな検索条件を設定し、目的のレストランを探し、詳細情報や地図を通じて素早く目的地に移動できます。

### ■ 重要なポイント

- **機能:** 周辺のレストランを様々な条件で検索できるだけでなく、経路案内機能も提供するように設計しました。
- **デザイン:** ユーザーエクスペリエンスを考慮し、できる限りシンプルで直感的なデザインを目指しました。

### ■ リポジトリ URL

<https://github.com/junani0v0/Amami>

### ■ 開発環境

- **開発ツール:** Visual Studio Code (バージョン 1.93.1)
- **使用言語:** JavaScript (ES6), HTML5, CSS3, JSON
- **対応 OS およびブラウザ:**
  - **OS:** Windows, macOS, Linux
  - **ブラウザ:** Chrome, Firefox, Safari など主要なブラウザ
- **開発期間:** 2 週間

### ■ 画面構成

- **検索画面:**

位置情報入力フィールド、検索範囲選択、並び替え基準選択、ジャンル選択などの検索オプションを提供するページです。  
ユーザーが住所入力または現在位置を使用してレストランを検索し、さまざまな検索条件を設定できます。
- **検索結果画面:**

検索条件に一致したレストランのリストを表示し、検索条件の再設定も可能です。  
レストランをクリックすると、モーダルウィンドウが開き、詳細情報を確認できます。
- **詳細情報画面:**

選択されたレストランの詳細情報を表示するポップアップウィンドウです。  
レストラン情報以外に、Google Maps を通じた経路案内ボタンや予約・クーポン URL 連携ボタンも含まれています。

## ■ 主な機能

- **レストラン検索**  
ユーザーの現在位置を基に、HotPepper Gourmet Web Service API を使用して周辺のレストランを検索します。
- **検索結果リスト表示**  
検索結果ページには、レストランのリストが 10 件ずつ表示され、各レストランの情報は写真、名前、所在地で構成されています。
- **レストラン情報の取得**  
検索されたレストランの詳細情報を API を通じてモーダルウィンドウに表示します。
- **道案内機能**  
Google Maps API を利用して、ユーザーの現在位置から選択したレストランまでの経路案内を提供します。
- **多言語対応**  
i18next を使用して、日本語、韓国語、英語のテキスト変換および言語変更機能を提供します。
- **現在位置に基づく検索**  
Geolocation API を使用してユーザーの現在位置を取得します。ユーザーが位置情報の許可をすると、現在位置が自動的に入力されます。
- **ページネーション**  
検索結果が多い場合、10 件ずつページネーションで表示し、ユーザーは複数ページの結果を探ることができます。
- **予約およびクーポン機能の連携**  
レストランの予約 URL やクーポン情報と連携し、予約およびクーポン利用をサポートします。
- **住所自動補完機能**  
Google Maps Places API を使用して、住所入力時に自動補完機能を提供します。
- **エラーメッセージおよび入力検証**  
緯度・経度の値が正しく入力されていない場合や、検索半径や並び替え基準が選択されていない場合に警告メッセージを表示して入力検証を行います。

## ■ 今後考慮すべき追加機能

- **ユーザーに合わせたおすすめシステム**  
ユーザーの検索履歴や好みに基づいて、おすすめのレストランを提供し、ユーザー体験を向上させる機能を追加することができます。
- **店舗レビューおよび評価機能**  
Google Places API や HotPepper API からレビューを取得してユーザーに表示したり、ユーザーが直接レビューを残せる機能を追加できます。
- **店舗画像の追加**  
HotPepper API では 1 つの画像のみ提供されるため、Google Places API から複数の画像を取得し、ユーザーに表示することができます。
- **電話連携および電話予約**  
HotPepper API では電話番号が提供されないため、Google Places API から電話番号を取得し、電話機能と連携することが可能です。
- **経路案内の改善**  
現在は緯度・経度のみで店舗の位置を提供していますが、店舗名や追加情報を一緒に提供するように改善できます。

## ■ 使用 API

### ● 外部 API およびライブラリ

- **i18next**  
多言語対応ライブラリで、テキストの翻訳および言語変更機能を提供します。
- **Google Maps JavaScript API (Places, Directions)**  
地図の表示および経路案内機能を提供します。
- **HotPepper Gourmet Web Service API**  
レストラン情報を検索および取得するための API です。
- **Tailwind CSS (v3.0)**  
ユーティリティクラスベースの CSS フレームワークで、レスポンシブデザインを効率的に構築できます。
- **Font Awesome**  
様々なアイコンを簡単に追加できるアイコンライブラリです。
- **Google Fonts (Noto Sans)**  
テキストスタイリング用のフォントです。

### ● 内蔵ブラウザ API

- **Geolocation API**  
ユーザーの現在位置情報を取得し、レストラン検索に利用します。
- **Fetch API**  
非同期ネットワークリクエストを通じて API データを取得します。
- **LocalStorage API**  
ローカルストレージにデータを保存し、取り出す機能を提供します。
- **DOM API**  
HTML 要素を動的に操作してユーザーインターフェイスを構築します。
- **Navigator API**  
ブラウザのナビゲーション情報を提供します。

## ■ アドバイスを求めたいポイント

- API Key 値を隠すために .env に値を入れて dotenv で呼び出そうとしましたが、エラーが継続して発生しました。
- i18next 多言語機能のエラー解決方法（JS ファイル内で共通ヘッダーを使用しましたが、i18next 追加後に動作せず HTML に分離）
- CORS エラーの解決方法（サーバーなしで解決可能か）
- HTML 内で独立した経路案内機能の実装方法
- Google Maps API で google.maps.Marker の警告解決方法  
（ google.maps.marker.AdvancedMarkerElement に修正しましたが、警告は消えたもののマップが読み込まれません）
- Google Maps API と HotPepper Gourmet API の距離基準の違いによる距離ソート問題の解決方法
- UI/UX の改善方法
- コードの最適化および簡素化
- セキュリティ強化方法
- エラーハンドリングおよびユーザーフィードバックの改善方法

## ■ 自己評価

- フロントエンドのみの開発は初めてだったため、API の暗号化や CORS エラーを正しく処理できなかったのが残念です。
- API キーの暗号化が解決して、アプリを公開できていれば、さらに良かったと思います。
- JS ファイルを共通で使用する中で多くの衝突が発生しましたが、次回は JavaScript を分割して使用する予定です。