

RouteMarker 機能仕様書

概要

RouteMarker (旧PickPoints) は、ハイキングマップのPNG画像からポイントとルートをマーキングし、座標データをJSONファイルとして出力するWebアプリケーションです。

システム構成

- **フロントエンド**: 純粋なHTML5、CSS3、JavaScript (ES6+)
- **依存関係**: なし (ブラウザネイティブAPI使用)
- **対応ファイル形式**: PNG画像 (入力)、JSON (出力・入力)

主要機能

1. 画像読み込み機能

概要

PNG形式のハイキングマップ画像を読み込み、Canvas要素に表示する機能です。

仕様

- **対応形式**: PNG画像のみ
- **読み込み方法**:
 - File System Access API (対応ブラウザ)
 - 従来のファイル入力 (フォールバック)
- **画像表示**: HTML5 Canvasに自動リサイズして表示
- **座標系管理**: 元画像座標とCanvas表示座標の相互変換

実装クラス・メソッド

- `PickPoints.handleImageSelection()`: File System Access APIを使用した画像選択
- `PickPoints.handleImageLoad()`: 従来方式でのファイル読み込み
- `PickPoints.loadImageFromFile()`: 画像ファイルの実際の読み込み処理
- `PickPoints.setupCanvas()`: Canvas要素のサイズ調整

2. ポイント編集機能

概要

地図上の重要なポイント (山頂、分岐点等) をクリックで配置・管理する機能です。

仕様

- **ポイント配置**: Canvas上でのマウスクリック
- **ポイント削除**: 既存ポイント上でのクリック

- **ID自動生成:** A-01, A-02...Z-99の形式
- **視覚表現:** 赤い円マーカー + 白抜き文字ID
- **最大ポイント数:** 2,574個 (A-01 ~ Z-99)

データ構造

```
{
  id: "A-01",           // ポイントID
  x: 123,               // Canvas座標X
  y: 456,               // Canvas座標Y
  imageX: 234,          // 元画像座標X
  imageY: 567           // 元画像座標Y
}
```

実装クラス・メソッド

- `PickPoints.handleCanvasClick()`: クリックイベント処理
- `PickPoints.addPoint()`: ポイント追加処理
- `PickPoints.removePoint()`: ポイント削除処理
- `PickPoints.generatePointId()`: ID自動生成ロジック
- `PickPoints.clearPoints()`: 全ポイント削除

3. ルート編集機能

概要

ポイント間の移動経路を中間点で定義し、ルートデータとして管理する機能です。

仕様

- **開始・終了ポイント:** 既存ポイントIDで指定
- **中間点配置:** Canvas上でのクリック
- **中間点削除:** 既存中間点上でのクリック
- **視覚表現:**
 - 中間点: 青い小さな円
 - ルートライン: 開始→中間点→終了を結ぶ線

データ構造

```
{
  routeInfo: {
    startPoint: "A-01",           // 開始ポイントID
    endPoint: "B-05",            // 終了ポイントID
    waypointCount: 3              // 中間点数
  },
  waypoints: [                  // 中間点配列
    {
```

```

        x: 123,           // Canvas座標X
        y: 456,           // Canvas座標Y
        imageX: 234,      // 元画像座標X
        imageY: 567       // 元画像座標Y
    }
  ],
  metadata: {
    imageFileName: "map01", // 元画像ファイル名
    exportedAt: "2025-08-19T..." // エクスポート日時
  }
}

```

実装クラス・メソッド

- `PickPoints.handleRouteCanvasClick()`: ルート編集時のクリック処理
- `PickPoints.addRoutePoint()`: 中間点追加
- `PickPoints.removeRoutePoint()`: 中間点削除
- `PickPoints.validateStartEndPoints()`: 開始・終了ポイント検証
- `PickPoints.clearRoute()`: ルート全削除

4. JSON出力機能

概要

作成したポイントデータやルートデータをJSON形式でファイル出力する機能です。

仕様

- **ポイントJSON**: 全ポイントの座標とメタデータ
- **ルートJSON**: ルート情報と中間点データ
- **ファイル名**:
 - 自動生成: 元画像名_points_YYYYMMDD_HHMMSS.json
 - ユーザー指定: カスタムファイル名入力可能
- **ダウンロード方式**:
 - File System Access API (推奨)
 - ブラウザダウンロード (フォールバック)

実装クラス・メソッド

- `PickPoints.exportJSON()`: ポイントデータのJSON出力
- `PickPoints.exportRouteJSON()`: ルートデータのJSON出力
- `PickPoints.downloadJSONWithUserChoice()`: ダウンロード処理統合

5. JSON読み込み機能

概要

以前に出力したJSONファイルを読み込み、ポイントやルートを復元する機能です。

仕様

- **対応形式:** RouteMarker出力形式のJSONファイル
- **復元内容:**
 - ポイント: 座標、ID
 - ルート: 開始・終了ポイント、中間点
- **座標変換:** JSON内の画像座標をCanvas座標に自動変換
- **エラーハンドリング:** 不正なJSONファイルの検出と警告

実装クラス・メソッド

- `PickPoints.handleJSONLoad()`: ポイントJSON読み込み
- `PickPoints.handleRouteJSONLoad()`: ルートJSON読み込み
- `PickPoints.loadPointsFromJSON()`: ポイントデータ復元
- `PickPoints.loadRouteFromJSON()`: ルートデータ復元

6. UI・レイアウト機能

概要

操作しやすいユーザーインターフェースを提供する機能です。

仕様

- **レイアウトモード:**
 - サイドバー (デフォルト) : 地図とコントロールを左右分割
 - オーバーレイ: コントロールを地図上に重ね表示
- **編集モード切り替え:** ポイント編集⇄ルート編集
- **リアルタイム表示:**
 - ポイント数カウンター
 - 中間点数カウンター
- **アクセシビリティ:** ARIA属性、キーボードナビゲーション対応

実装クラス・メソッド

- `PickPoints.initializeLayoutManager()`: レイアウト管理初期化
- `PickPoints.setEditingMode()`: 編集モード切り替え
- `PickPoints.updatePointCount()`: ポイント数表示更新
- `PickPoints.updateWaypointCount()`: 中間点数表示更新

7. 描画・ビジュアル機能

概要

地図上のポイント、ルート、マーカーを視覚的に表示する機能です。

仕様

- **ポイント描画:**

- 赤い円（半径8px）
- 白抜き文字でID表示
- **ルート描画:**
 - 中間点: 青い小円（半径4px）
 - ルートライン: 開始→各中間点→終了を結ぶ線
 - 開始・終了ポイント: 緑色でハイライト
- **Canvas管理:**
 - 画像とマーカーの重ね描画
 - 高DPI対応

実装クラス・メソッド

- `PickPoints.drawImage()`: 画像とマーカーの統合描画
- `PickPoints.drawPoints()`: ポイント描画
- `PickPoints.drawRoutes()`: ルート描画
- `PickPoints.drawPoint()`: 個別ポイント描画
- `PickPoints.drawRoute()`: 個別ルート描画

技術仕様

ブラウザ要件

- **必須API:**
 - HTML5 Canvas
 - FileReader API
 - JSON処理
- **推奨API:**
 - File System Access API（Chrome 86+）
- **対象ブラウザ:** Chrome, Firefox, Safari, Edge（最新版）

パフォーマンス

- **最大ポイント数:** 2,574個
- **最大中間点数:** 制限なし（実用的には数百点）
- **対応画像サイズ:** ブラウザのメモリ制限内

セキュリティ

- **ローカル処理:** すべての処理はブラウザ内で完結
- **外部通信:** なし
- **データ保存:** ローカルファイルのみ

エラーハンドリング

画像読み込みエラー

- PNG以外のファイル形式
- 破損した画像ファイル
- ファイルサイズ超過

データ整合性エラー

- 不正なJSON形式
- 存在しないポイントIDの参照
- 座標値の範囲外エラー

ブラウザ互換性エラー

- File System Access API未対応時の自動フォールバック
- Canvas描画エラーの検出と復旧

今後の拡張予定

- GPX形式データの出力対応
- 複数ルートの同時管理
- ポイント種別（山頂、小屋等）の分類機能
- 距離・標高情報の表示機能