

WalkLogger 機能仕様書

バージョン: 202601 最終更新日: 2026年1月24日

1. 概要

1.1 アプリケーション名

WalkLogger - GPS位置記録

1.2 目的

地理院地図上に現在地を表示し、GPS位置と写真を記録するPWA（Progressive Web App）。ウォーキングやハイキングなどのアウトドア活動で位置情報と写真を同時に記録・管理できる。

1.3 動作環境

- プラットフォーム: Webブラウザ（PWA対応）
- 対応ブラウザ: Chrome、Safari、Firefox（Geolocation API、Camera API対応ブラウザ）
- 必須環境: HTTPS環境またはlocalhost（Geolocation API、Camera APIの要件）

1.4 技術スタック

項目	技術
地図	Leaflet.js 1.9.4 + 国土地理院タイル
フロントエンド	Vanilla JavaScript (ES6)
スタイル	カスタムCSS（レスポンシブ対応）
ローカルストレージ	IndexedDB
クラウドストレージ	Firebase (Firestore + Storage)
認証	Firebase Anonymous Authentication
PWA	Service Worker + Web App Manifest

2. 画面構成

2.1 メイン画面

- 地図表示領域: 国土地理院地図を全画面表示
- ステータス表示: 画面上部に現在のステータスと座標情報を表示
- コントロールパネル: 画面下部に操作ボタンを配置
- データ管理パネル: Dataボタンで切り替え表示

2.2 ダイアログ

ダイアログ名	用途
写真一覧ダイアログ	保存済み写真のグリッド表示
写真拡大ダイアログ	選択写真の拡大表示と詳細情報
カメラダイアログ	写真撮影UI（撮影＋方向選択）
記録統計ダイアログ	データサイズ・記録統計の表示
ドキュメント選択ダイアログ	Firebase保存データの読み込み選択
ドキュメント名入力ダイアログ	Firebase保存時の名前入力
データ初期化確認ダイアログ	Start時の既存データ確認

3. 機能詳細

3.1 GPS追跡機能

3.1.1 GPS追跡開始（Startボタン）

- **状態:** 初期状態で有効、追跡中は無効
- **動作:**
 1. 既存データの確認ダイアログを表示
 - データがある場合: 「データ初期化」または「データ追記」を選択
 - データがない場合: 「新規記録を開始」または「キャンセル」を選択
 2. Wake Lock APIで画面スリープを防止
 3. iOS 13以降の場合、DeviceOrientation許可を要求
 4. Geolocation API（watchPosition）でGPS追跡を開始
 5. GPS位置を以下の条件で記録:
 - 初回は必ず記録
 - 60秒以上経過、または20m以上移動した場合に記録（GPSの精度を超えて移動した場合のみ）
- **GPS位置オプション:**
 - `enableHighAccuracy: true`
 - `timeout: 10000ms`
 - `maximumAge: 0`

3.1.2 GPS追跡停止（Stopボタン）

- **状態:** 追跡開始後に有効
- **動作:**
 1. GPS監視（watchPosition）を停止
 2. Wake Lockを解放
 3. トラッキングデータをIndexedDBに最終保存
 4. 最後の記録地点を保存

3.1.3 位置データ形式

```
{
  lat: number,      // 緯度（小数点以下5桁）
  lng: number,      // 経度（小数点以下5桁）
  timestamp: string, // ISO 8601形式
  accuracy: number  // 精度（小数点以下1桁、メートル）
}
```

3.2 写真撮影機能

3.2.1 撮影フロー（Photoボタン）

- **状態:** GPS追跡中のみ有効
- **動作:**
 1. カメラダイアログを表示
 2. 背面カメラでプレビュー表示（720x1280pxを希望）
 3. シャッターボタンで撮影
 4. プレビュー確認画面を表示
 - "Retake"ボタン: 再撮影（カメラ画面に戻る）
 - "Text"ボタン: テキストメモを入力（プロンプト表示）
 - 方向ボタン（左・上・右）: 方向を選択して保存
 5. 方向を選択すると:
 - 選択した方向の矢印アイコンを画像下部にスタンプ
 - 現在のGPS位置情報を紐付け
 - 入力されたテキストメモがある場合は紐付け
 - IndexedDBに保存
 - 地図上に写真マーカー（オレンジ色の丸）を追加

3.2.2 写真データ形式

```
{
  data: string,      // Base64形式の画像データ（JPEG、品質0.6）
  timestamp: string, // ISO 8601形式
  direction: string, // "left" | "up" | "right"
  location: {
    lat: number,      // 緯度（小数点以下5桁）
    lng: number       // 経度（小数点以下5桁）
  },
  text: string       // [任意] 写真へのメモテキスト
}
```

3.3 データ管理機能

3.3.1 写真一覧（Listボタン）

- 保存済み写真をグリッド表示
- サムネイルクリックで拡大表示

- タイトル「Photo Gallery」と写真数の間に改行を入れて表示
- 拡大表示時に撮影日時、位置情報、方向、およびテキストメモ（ある場合）を表示

3.3.2 データサイズ表示（Sizeボタン）

- 表示項目:
 - トラック件数と位置記録点数
 - GPSデータサイズ（KB/MB）
 - 写真枚数
 - 写真データサイズ（KB/MB）
 - 写真解像度
- GPS追跡中はリアルタイム更新

3.3.3 Firebase保存（Saveボタン）

- 前提条件: GPS追跡を開始した後のみ有効
- 動作:
 1. ドキュメント名入力ダイアログを表示（デフォルト: 追跡開始日時）
 2. 同名ドキュメントが存在する場合は自動で連番付与（例: name_2, name_3）
 3. 写真をFirebase Storageにアップロード（認証ありの場合のみ）
 4. プロジェクトデータをFirestoreに保存

3.3.4 Firebaseからの読み込み（Reloadボタン）

- 動作:
 1. Firestoreからプロジェクト一覧を取得（作成日時の降順）
 2. ドキュメント選択ダイアログを表示
 3. 選択したドキュメントのデータをIndexedDBに復元
 4. 地図上にトラックと写真マーカーを表示
- 表示項目: ドキュメント名、作成日時、ユーザーID（一部）、トラック件数、写真枚数

3.4 地図表示機能

3.4.1 地図設定

項目	値
タイルソース	国土地理院標準地図
最小ズーム	5
最大ズーム	18
初期位置	最後の記録位置 or デフォルト（箕面大滝: 34.853667, 135.472041）
初期ズーム	13（デフォルト） / 15（位置取得時）

3.4.2 マーカー表示

マーカー種別	外観	用途
--------	----	----

マーカー種別	外観	用途
現在位置マーカー	緑の三角形（矢印型）	現在地と進行方向を表示
写真マーカー	オレンジ色の丸（12px）	写真撮影位置を表示

3.4.3 軌跡表示

- 色: #4CAF50（緑）
- 線幅: 4px
- 透明度: 0.7

3.5 デバイス方向取得機能

3.5.1 DeviceOrientation API

- iOS Safari: `webkitCompassHeading`を使用
- Android Chrome等: `alpha`値から方角を計算（ $360 - \alpha$ ）
- 現在位置マーカーの向きをリアルタイム更新

3.5.2 GPSヘディング

- `position.coords.heading`が利用可能な場合はそちらを優先

3.6 画面スリープ防止機能

3.6.1 Wake Lock API

- GPS追跡開始時に画面スリープを防止
- 追跡停止時に解放
- ページが再表示された時に自動再取得
- 非対応ブラウザでは警告をコンソール出力

4. データ永続化

4.1 IndexedDB構成

4.1.1 データベース情報

項目	値
データベース名	WalkLoggerDB
バージョン	2

4.1.2 オブジェクトストア

ストア名	キー	インデックス	用途
------	----	--------	----

ストア名	キー	インデックス	用途
tracks	id (autoIncrement)	timestamp	トラッキングデータ
photos	id (autoIncrement)	timestamp	写真データ
settings	key	-	設定（最終位置など）

4.1.3 tracksデータ構造

```
{
  id: number,           // 自動採番
  timestamp: string,    // セッション開始時刻 (ISO 8601)
  points: [{            // 位置データ配列
    lat: number,
    lng: number,
    timestamp: string,
    accuracy: number
  }],
  totalPoints: number // 記録点数
}
```

4.1.4 photosデータ構造

```
{
  id: number,           // 自動採番
  data: string,          // Base64画像データ
  timestamp: string,    // 撮影日時 (ISO 8601)
  direction: string,    // 方向 ("left" | "up" | "right")
  location: {
    lat: number,
    lng: number
  },
  text: string           // [任意] 写真へのメモテキスト
}
```

4.2 Firebase構成

4.2.1 Firestore構成

```
projects/
├── {projectName}/      // ドキュメント名 (日時または任意名)
│   ├── userId: string
│   ├── startTime: string
│   ├── createdAt: timestamp
│   ├── lastPosition: { lat, lng, zoom, timestamp }
│   └── tracks: []      // トラック配列
```

```
├─ photos: []      // 写真メタデータ配列
├─ tracksCount: number
└─ photosCount: number
```

4.2.2 Firebase Storage構造

```
projects/
├─ {projectName}/
│   └─ photos/
│       └─ {timestamp}.jpg  // 写真ファイル
```

4.2.3 写真メタデータ（Firestore内）

```
{
  url: string,          // ダウンロードURL
  storagePath: string,  // Storageパス
  timestamp: string,
  direction: string,
  location: { lat, lng },
  text: string          // [任意] 写真へのメモテキスト
}
```

5. PWA機能

5.1 Service Worker

5.1.1 キャッシュ対象

- ./index.html
- ./styles.css
- ./manifest.json
- ./js/firebase-config.js
- ./js/app-main.js（およびESモジュール群）
- Leaflet CSS/JS（CDN）

5.1.2 キャッシュ戦略

リソース種別	戦略
国土地理院タイル	ネットワーク優先（オフライン時は空レスポンス）
その他リソース	キャッシュ優先（キャッシュミス時はネットワーク取得後キャッシュ）

5.1.3 キャッシュバージョン

- 現在: walklogger-v4
- アップデート時は古いキャッシュを自動削除

5.2 Web App Manifest

項目	値
name	WalkLogger - GPS位置記録
short_name	WalkLogger
display	standalone
orientation	portrait-primary
theme_color	#4CAF50
background_color	ffffff
lang	ja

5.3 アイコン

サイズ	ファイル
180x180	icons/icon-180.png
192x192	icons/icon-192.png
512x512	icons/icon-512.png

6. UIコンポーネント

6.1 メインコントロールパネル

ボタン	色	機能
Start	緑 (#4CAF50)	GPS追跡開始
Stop	赤 (#f44336)	GPS追跡停止
Photo	青 (#2196F3)	写真撮影
Data	紫 (#9C27B0)	データ管理パネル表示

6.2 データ管理パネル

ボタン	色	機能
List	オレンジ (#FF9800)	写真一覧表示
Size	シアン (#00BCD4)	データサイズ表示
Save	緑 (#4CAF50)	Firebase保存

ボタン	色	機能
Reload	インディゴ (#3F51B5)	Firebase読み込み

6.3 カメラUI

状態	表示要素
撮影前	カメラプレビュー、シャッターボタン（白丸）、閉じるボタン
撮影中	カメラプレビュー、シャッターボタン（白丸）、閉じるボタン
撮影後	撮影画像、方向ボタン×3（左・上・右）、Textボタン、Retakeボタン、閉じるボタン

7. エラーハンドリング

7.1 GPS関連エラー

エラーコード	メッセージ
PERMISSION_DENIED	位置情報の使用が許可されていません
POSITION_UNAVAILABLE	位置情報が利用できません
TIMEOUT	位置情報の取得がタイムアウトしました

7.2 カメラ関連エラー

エラー名	メッセージ
NotAllowedError	カメラの使用が許可されていません
NotFoundError	カメラが見つかりません

7.3 Firebase関連エラー

- 認証エラー時は詳細なコンソールログを出力
- 写真アップロード失敗時は個別にスキップして続行
- 認証なしでもGPS記録は継続可能

7.4 IndexedDB関連エラー

- 初期化失敗時はアラートを表示してページ再読み込みを促す
- 削除がブロックされた場合はリトライ（最大3回）

8. 座標系・単位

8.1 座標系

- **測地系:** WGS84（世界測地系）
- **形式:** 10進数度（Decimal Degrees）

- **精度:** 緯度・経度ともに小数点以下5桁

8.2 データサイズ表示

- 10MB以下: KB単位 (4桁精度)
- 10MB超: MB単位 (4桁精度)

8.3 距離計算

- Haversine公式を使用
- 地球半径: 6,371,000m

9. 制限事項

9.1 ブラウザ制限

- Geolocation API: HTTPS必須 (localhost除く)
- Camera API: HTTPS必須
- Wake Lock API: 一部ブラウザ非対応
- DeviceOrientation API: iOS 13以降は許可要求必須

9.2 Firebase制限

- 写真アップロード: Firebase認証必須
- プロジェクト名連番: 最大100まで

9.3 記録条件

- GPS位置は60秒以上経過または20m以上移動で記録
- 写真撮影はGPS追跡中のみ可能

10. ファイル構成

```
WalkLogger/
├─ index.html          # メインHTML
├─ styles.css          # スタイルシート
├─ manifest.json       # PWAマニフェスト
├─ service-worker.js   # Service Worker
├─ js/                 # JavaScriptモジュール (ES6)
│   └─ app-main.js     # メイン初期化・イベント設定
│   └─ config.js       # 定数・設定値
│   └─ state.js        # グローバル状態管理
│   └─ utils.js        # ユーティリティ関数
│   └─ db.js           # IndexedDB操作
│   └─ map.js          # 地図表示・マーカー管理
│   └─ tracking.js     # GPS追跡・位置更新
│   └─ camera.js       # カメラ・写真撮影
│   └─ camera.js       # カメラ・写真撮影
│   └─ firebase-ops.js # Firebase操作
```

```
|   |─ ui.js           # UIモジュール統合 (re-export)
|   |─ ui-common.js  # 共通UI関数
|   |─ ui-photo.js   # 写真関連UI
|   |─ ui-dialog.js  # ダイアログ関連UI
|   |─ firebase-config.js # Firebase設定
|   |─ firebase-config.template.js # Firebase設定テンプレート
|─ icons/
|   |─ icon-180.png
|   |─ icon-192.png
|   |─ icon-512.png
|─ docs/
|   |─ funcspec-202601.md # 機能仕様書 (本書)
|   |─ UsersGuide-202601.md # 利用者の手引
|   |─ FIREBASE_SETUP.md # Firebase設定ガイド
```

10.1 モジュール構成

| config.js | 定数・設定 | DB_NAME, GPS_RECORD_, PHOTO_ || state.js | 状態管理 | map, isTracking, trackingData 等 || utils.js | 汎用関数 | formatDate, calculateDistance 等 || db.js | DB操作 | initIndexedDB, saveTrack, getAllPhotos 等 || map.js | 地図機能 | initMap, updateCurrentMarker 等 || tracking.js | GPS追跡 | startTracking, stopTracking 等 || camera.js | カメラ | takePhoto, capturePhoto 等 || firebase-ops.js | Firebase | saveToFirestore, reloadFromFirestore 等 || ui.js | UI統合 | (ui-common, ui-photo, ui-dialogの統合) || ui-common.js | 共通UI | updateStatus, toggleVisibility 等 || ui-photo.js | 写真UI | showPhotoList, showPhotoViewer 等 || ui-dialog.js | ダイアログ | showDataSize, showDocumentListDialog 等 || app-main.js | 初期化 | initApp, setupEventListeners 等 |

11. 変更履歴

日付	バージョン	内容
2026-01-22	202601	初版作成 (現行コードからの機能仕様書化)
2026-01-24	202601	GPS記録条件にGPS精度チェックを追加、写真解像度を720x1280pxに固定、写真品質を0.6に変更、コードをES6モジュール構成にリファクタリング (10モジュール)、ファイル構成を更新
2026-01-29	202601	写真撮影時にテキストメモを追加する機能を実装 (Textボタン)、Photo Galleryのタイトル表示を調整、保存データ構造にtextフィールドを追加