GPX Routes Workbench

A single-screen Flet application for processing GPX routes exported from iPhone Health data.

Features

- 📱 **iPhone Health Export Instructions**: Step-by-step guide for exporting health data
- 📁 **File Management**: Import and manage GPX route files with FilePicker
- ✅ **File Selection**: Select/deselect specific route files for processing
- 🗺️ **Route Visualization**: Map routes using interactive Folium maps
- ⚡ **Speed Tags**: Automatically calculate and add speed data to trackpoints
- ✂️ **Smart Trimming**: Remove points with excessive speed (configurable threshold)
- 📤 **Hikes Integration**: Post processed routes to Hikes API
- 💾 **Persistent Data**: Save settings and preferences
- 📝 **Comprehensive Logging**: Track all operations with detailed logs

Installation

1. Clone this repository:

```
git clone https://github.com/SummittDweller/gpx-routes-workbench-usi
cd gpx-routes-workbench-using-flet
```

2. Create and activate a virtual environment:

```
# Create virtual environment
python3 -m venv .venv

# Activate virtual environment
source .venv/bin/activate  # On macOS/Linux
# or
.venv\Scripts\activate     # On Windows
```

3. Install dependencies:

```
pip install -r requirements.txt
```

Usage

Quick Start with run.sh (macOS/Linux)

The easiest way to run the application is using the provided `run.sh` script:

```
./run.sh
```

What run.sh does:

- ✓ Automatically creates a Python virtual environment (`.venv`) if it doesn't exist
- ✓ Activates the virtual environment
- ✓ Installs all required dependencies from `requirements.txt`
- ✓ Launches the GPX Routes Workbench application

First-time setup:

```
# Make the script executable (only needed once)
chmod +x run.sh

# Run the application
./run.sh
```

Note: The `run.sh` script handles everything for you. You don't need to manually create the virtual environment or install dependencies!

Manual Launch (Alternative)

If you prefer to run the application manually:

```
# Activate virtual environment first
source .venv/bin/activate  # On macOS/Linux
# or
.venv\Scripts\activate     # On Windows

# Run the application
python main.py
```

The application will open in a new window or web browser.

Exporting iPhone Health Data

1. Open the **Health** app on your iPhone
2. Tap your profile picture or initials in the top right
3. Scroll down and tap "**Export All Health Data**"
4. Wait for the export to complete (may take several minutes)

5. Save the `export.zip` file to your **Downloads** folder on your computer

✨ **AUTO-IMPORT**: If `export.zip` is in your `~/Downloads` folder when you launch the app, the workout routes will be automatically extracted to the temporary directory!

Alternatively, you can manually:

6. Extract the `export.zip` file on your computer
7. Inside the extracted folder, look for the **workout-routes** folder
8. Use the file picker in the app to import GPX files from the workout-routes folder

Processing Routes

1. **Import Files**: Routes are auto-imported from `~/Downloads/export.zip`, or click "📁 Select GPX Files" to manually import
2. **Select Routes**: Use checkboxes to select which routes to process, or use "Select All"/"Deselect All"
3. **Visualize**: Click "🗺️ Visualize Routes" to create an interactive map (saved as HTML)
4. **Add Speed Tags**: Click "⚡ Add Speed Tags" to calculate speed between points
5. **Trim by Speed**: Set max speed threshold in mph (default: 5 mph) and click "✂️ Trim by Speed" to remove slow-moving GPS drift points
6. **Post to Hikes**: Enter your Hikes API URL and click "📤 Post to Hikes"

File Structure

```
gpx-routes-workbench-using-flet/
├── main.py              # Main application code
├── requirements.txt     # Python dependencies
├── test_gpx.py         # Test suite
├── .gitignore          # Git ignore patterns
├── README.md           # This file
├── temp_gpx_routes/    # Temporary directory for GPX files (createc
├── gpx_workbench.log   # Application log file (created on first rur
└── app_data.json       # Persistent settings (created on first run)
```

Features in Detail

Persistent Data Storage

Settings and preferences are automatically saved to `app_data.json`, including:

- Temporary directory path
- Hikes API URL
- Selected files

Logging

All operations are logged to both console and `gpx_workbench.log` file for debugging and auditing.

Route Visualization

Creates interactive HTML maps using Folium that can be opened in any web browser. Multiple routes are displayed in different colors for easy comparison.

Speed Calculations

Calculates speed between consecutive points based on:

- Distance (3D or 2D)
- Time difference
- Speed stored in meters per second (m/s)

Speed-Based Trimming

Removes GPS points with unrealistic speeds. Default threshold is 5 mph, which helps filter out GPS drift and errors when the device is stationary or moving very slowly. Useful for cleaning up routes where the GPS recorded movement while you were actually standing still.

Testing

Run the test suite:

```
python test_gpx.py
```

Dependencies

- **flet** (≥0.23.0): Modern UI framework
- **gpxpy** (≥1.6.2): GPX file parsing and manipulation
- **folium** (≥0.16.0): Interactive map generation
- **requests** (≥2.31.0): HTTP requests for API integration

License

This project is open source and available under the MIT License.

Contributing

Contributions are welcome! Please feel free to submit a Pull Request.

Acknowledgments

A rewrite of the GPX Track Workbench Streamlit app using Flet and .zip exports from Apple Health.