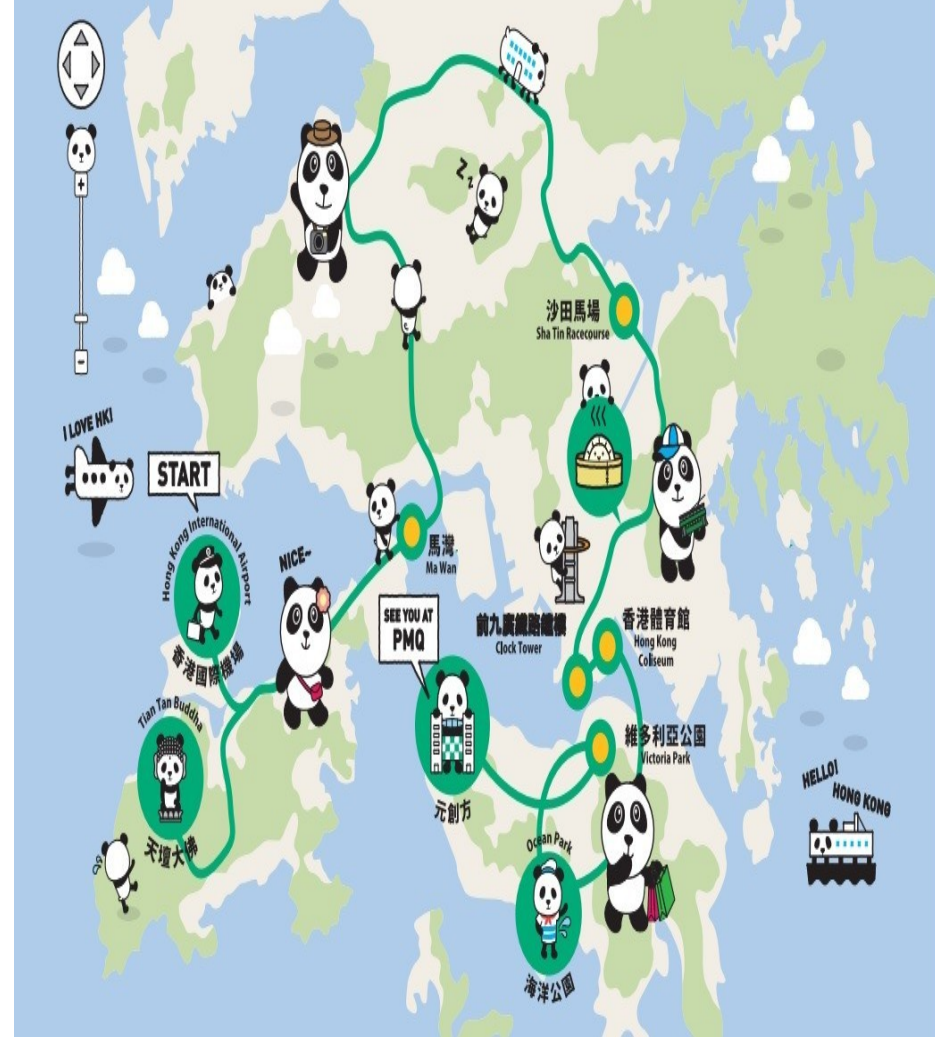


Open Source GIS

Concept Presentation

Team: LocAround
Winter semester
2024



Idea behind the Project

- Imagine you're in a new city, and you are willing to spend your day by exploring local events and activities. You open your phone or laptop, ready to plan, but there are lots of problems in your planning.



Idea behind the Project(problem

- 1: Difficulty to find local events
- 2: Lack of personalized recommendations
- 3: Inefficient route planning
- 4: Massive tools and lack of integration
- 5: Missed opportunities to discover new activities



Description of the Project (How to solve the Problem)

- **Interactive Event Map:** Solves Problem 1 (Difficulty Finding Local Events) by displaying activities around the user's location.
- **Personalized Recommendations:** Resolves Problem 2 (Lack of Personalized Recommendations) through a preference- based filtering system.
- **Optimized Route Planning:** Tackles Problem 3 (Inefficient Route Planning) by offering walking, biking, and driving routes with accurate travel times.



Description of the Project (How to solve the Problem)

- **Integrated Solution:** Addresses Problem 4 (Overwhelming Tools and Lack of Integration) by combining event discovery and routing into a single, user-friendly platform.



Architecture

Frontend Components:

- **UI Design:** Landing page for user preferences (event type, activity, location).
- **Interactive Map:** Displays event markers based on user preferences (Leaflet.js or OpenLayers).
- **Route Display:** View walking, biking, or driving routes on the map (calculated route time).

Backend Components:

- **REST API:** Handles event filtering and route calculation based on user input.
- **Routing Server:** Uses OpenStreetMap data to calculate optimized routes (walking, biking, driving).
- **Database:** Stores event data (PostgreSQL with PostGIS or MongoDB) or linked directly to API

Technologies:

- **Frontend:** React.js, Leaflet.js, Axios.
- **Backend:** Node.js/Django, PostgreSQL (PostGIS), OSRM/Graph Hopper for routing.

Data sources

Base Map and Routing:

- OpenStreetMap (OSM) for geographical and transportation data.
 - Use of **Maplibra** or **Leaflet** to integrate map tiles (base map)
 - Points of interest (restaurant, parks, paths, bike lanes, ...)
 - Road calculation for different transportation (**Graph Hopper + Opentripplanner-GTFS.file**)

Event Data:

- Open APIs
 - **Eventbrite API** : search for events based on criteria such as location, date, and event type
 - **Meetup API** : search for groups and events based on location and user interests

User Input:

- Preferences and location provided by users.
 - Geolocation services to obtain the user's current position : **HTML5 Geolocation API**
 - Address Input: Allow users to manually enter their address or current location.

Components

Frontend:

- For event and route display :
 - GraphHopper:
 - Features: Event Markers, Route Display, Interactive Elements
- Filters for user preferences :
 - User Interface: Form Elements, Sliders and Dropdowns, Real-Time Filtering
 - Filtering Logic: Client-Side Filtering, Server-Side Filtering

Backend:

- Database for event storage and routing profiles.
 - Database Choice: Relational Database, NoSQL Database
 - Data Storage: Event Data, Routing Profiles, API layer for data exchange.
 - RESTful API: Endpoints, Authentication, Documentation



Components

Routing Server:

- Preloaded with OSM data: Data Import, Data Updates
- Multi profiles for walking, biking, and driving.

Event Management:

- System for adding and updating event data :
 - Eventbrite API : Integration, Synchronization, Event Creation

User Location Service:

- Detects and uses user location for nearby event filtering :
 - HTML5 Geolocation API: Location Detection, Permission Handling, Nearby Event Filtering



Project Plan (Milestones)

Week 1 (Dec 2 – Dec 8): Set up project structure, repository, and define roles/tasks.

Week 2 (Dec 9 – Dec 15): Configure routing engine, work on backend APIs and event the filtering.

Week 3 (Dec 16 – Dec 22): Build frontend map interface and integrate event filtering functionality.

Dec 20 : Milestone presentation

Week 4 (Dec 23 – Dec 29): Christmas Break

Week 5 (Dec 30 – Jan 5): Continue with the frontend map interface and integrate event filtering functionality.

Week 6 (Jan 6 – Jan 12): Continue with routing engine, work on backend APIs and event the filtering.

Week 7 (Jan 13 – Jan 19): Test routing functionality and conduct usability testing.

Week 8 (Jan 20 – Jan 24): Write documentation, finalize UI and prepare the presentation.

Jan 24: Final Presentation



Thank you
LocAround Team