TourTrack- FinalDoc

Authors: Ori Katz & Shani Halali

Supervisor: Dr. Sharon Yalov-Handzel

Institution: Department of Computer Science, Afeka

Date: June 2025

Abstract

TourTrack is a mobile application developed in collaboration with the Lower Galilee Regional Council to promote local tourism through a social and interactive hiking experience.
The motivation behind the project emerged from the need for a dedicated platform that would allow hikers to receive real-time updates, communicate with fellow travelers, and explore personalized hiking routes.

The problem addressed was the lack of an all-in-one solution for discovering nearby trails, reporting trail conditions, and connecting with users in the same area.
Existing apps provided either navigation or content sharing, but none offered a fully integrated, location-aware social network for hikers.

Our approach combined user registration, location-based interaction, and a route recommendation engine that suggests trails based on user preferences, group size, and real-time weather conditions.
The system also includes dynamic map integration, hazard and recommendation reporting, and direct communication tools such as WhatsApp and phone calls.

The result is a fully functional Android app that delivers a personalized and community-driven experience, tested successfully through various user scenarios. The app enhances travel planning, safety, and enjoyment while promoting tourism.

Our conclusion is that real-time, user-driven platforms like TourTrack can significantly improve both the individual experience of hikers and the broader goal of regional tourism development.

Table of Contents

Introduction

Motivation

Hiking is one of the most popular outdoor activities in Israel, especially during weekends and holidays.
However, travelers often encounter challenges such as lack of updated trail information, uncertainty about trail conditions, and difficulty finding others to hike with.
At the same time, local municipalities and nature authorities struggle to provide timely updates to the public and promote underutilized routes.

The Lower Galilee Regional Council identified the need for a digital platform that would serve both travelers and local authorities by offering real-time information, connecting people on the trail, and supporting local tourism development — specifically in the Tavor Trail region.
This motivated the creation of TourTrack — a social hiking application that combines maps, user-generated trail reports, live location sharing, and personalized recommendations.

Problem Statement

Current travel and navigation apps in Israel do not provide a centralized, real-time, and community-driven solution for hikers.
Most apps either focus on navigation (e.g., Google Maps) or social sharing (e.g., Travello), but fail to combine real-time trail condition updates, personalized recommendations, and direct user interaction.
Additionally, hikers have no single reliable source for verifying whether a trail is safe, crowded, or suitable for their group.

This gap results in frustrating experiences: users arrive at trails that are blocked or too difficult, miss hidden attractions, or lack social tools to connect with others nearby.
For local authorities, the lack of two-way communication limits their ability to inform and monitor what happens on their trails.

Project Goals and Objectives

TourTrack was designed to solve these problems by offering:
- An interactive trail map based on structured route data provided by regional authorities, combined with real-time user-generated reports (e.g., hazards, crowd levels, or blocked paths).

- Personalized route recommendations generated from user preferences (such as trail type, group size, interests, difficulty level), current weather conditions, and location.

- Real-time communication tools allowing users to connect with others nearby via WhatsApp or phone calls.

- A reporting system that enables users to submit trail feedback and recommendations, which are visible to other users and managed by authorities.

- Permission management that lets users control their visibility and contact preferences (e.g., block incoming calls or messages).

- A dedicated interface for local councils and nature authorities to manage route data and respond to field reports efficiently.

Overview of Approach and Main Contributions

TourTrack is a mobile Android application that combines multiple technologies and user-centered design principles:
- Modular architecture with frontend (Java/XML), backend (Node.js), and a MongoDB database
- Location-based features for displaying users and trails in real time
- Smart recommendation engine based on user input, trail metadata, and weather API
- Dynamic permissions system for managing user interactions and admin authority
- UI/UX based on real user interviews and personas, ensuring relevance for independent hikers, families, and older adults

Our main contribution is the integration of social features, live data, and tourism support into a single, localized platform that enhances the travel experience while supporting operational efficiency for local authorities.

Background and Related Work

Literature Review

One of the innovative algorithms in our project is the recommender algorithm for travelers, which aims to provide personalized recommendations to tourists.
The algorithm is based on three main methods:
Collaborative Filtering (CF), which provides recommendations based on similarity between users or items.
Content-Based Filtering (CB), which uses the characteristics of attractions and personal preferences.
Demographic Filtering (DF), which focuses on characteristics such as age and location.
The combination of these methods allows leveraging the advantages of each and solving recommendation challenges, particularly for new users without previous history in the system [1].

Additionally, the implementation of advanced big data and artificial intelligence (AI) techniques enhances the tourism industry by personalizing recommendations and improving user satisfaction.
The system is based on real-time analysis of parameters such as location, budget, interests, user feedback, and additional external factors like weather and nearby events.
The system's outputs include accurate personalization, enriching the user experience with focused recommendations and tools for quick and intelligent decision-making.
In conclusion, the combination of the algorithm with big data technologies creates more personalized and high-quality experiences, focusing on the changing needs of users and improving the quality of products and services [2].

[1] Kbaier, M. E. B. H., Masri, H., & Krichen, S. (2017, October). A personalized hybrid tourism recommender system. In 2017 IEEE/ACS 14th International Conference on Computer Systems and Applications (AICCSA) (pp. 244-250). Ieee.

[2] Hu, H., & Li, C. (2023). Smart tourism products and services design based on user experience under the background of big data. Soft Computing, 27(17), 12711-12724.

Competitor Analysis

1. GAFFL (Get a friend for life)

Is an app that serves as a social network to connect travelers planning to visit the same destinations around the world.
The app allows users to select hobbies and travel preferences, create new trips with a destination, flexible dates, and write a brief description of the trip plan, including an option to use an AI button for automatic writing.
Additionally, it enables connections with other travelers based on shared location or interests and includes a chat feature for messaging between users.
However, unlike our app, GAFFL does not display real-time user locations, and it lacks the ability to share live reports from the field, such as crowds or route changes.
Moreover, it does not have a weather forecast interface, so it cannot alert users to changing conditions or adjust the route accordingly.

2. Google Maps

Is an app by Google that provides navigation, location exploration, and transportation in a digital and convenient way.
It is considered one of the most popular platforms in the world for navigation and maps.
It offers real-time routes for cars, walking, biking, and public transportation, including alerts about traffic delays, obstacles, and roadwork.
Additionally, users can search for places such as restaurants, gas stations, and shops, view reviews, plan future routes, and share locations with others.
Our app, compared to Google Maps, offers a much more community, personalized, and tourism-focused experience in Israel.
While Google Maps provides general and global information, our app focuses on promoting local tourism with live updates, personalized adjustments, and community sharing among travelers.
Additionally, while Google Maps shows weather in certain areas, our app provides a direct weather interface that integrates with hiking routes and attractions, offering accurate recommendations based on weather conditions, significantly enhancing the travel experience.

3. Travello

Is an app where users can upload posts that include pictures, descriptions, and locations to share their travel experiences.
Users can join community groups based on the country they plan to visit – these groups function as group chats for all members who join.
Users can view the profiles of other app users who are near their location and discover the areas they have traveled to.
Another feature of the app is the option to purchase attractions.
Our app is unique in its combination of dynamic and static features, allowing users not only to share their experiences but also to receive real-time updates about routes, real-time user locations, and content tailored to the weather conditions in the area.
While Travello focuses on sharing experiences through photos and posts, our uniqueness lies in the ability to connect travelers in real time – sharing experiences, reporting on crowd levels and trail conditions, weather alerts, and the option to see and connect with other travelers nearby interactively.

4. AllTrails

Is a popular platform for hiking, running, and biking lovers, focusing on recommending nature trails worldwide.
The app displays trails with user ratings, length, difficulty, and terrain conditions, and allows users to search for trails by geographic location, download maps for offline use, and share experiences with the community.
However, most features in the app are paid, such as advanced tracking of the trail, finding users hiking in the same area, and detailed maps, while free usage is limited to trail navigation and marking favorite trails.
Our app, in contrast to AllTrails, offers a local, personalized, and tourism-focused user experience for Israel, with full access to all features at no cost.
In addition to trail recommendations, it integrates a direct interface with the weather, which impacts real-time trail conditions and recommended attractions.
The app also provides access to information on accommodations and restaurants, allowing travelers to plan their trips optimally according to terrain conditions and their needs, while sharing experiences with the local community.

System Design / Methods / Approach

Functional Requirements

TourTrack provides a set of core features designed to meet the needs of individual travelers, families, and nature authorities:

- Interactive Map Integration: Displays trails, attractions, and points of interest provided by local authorities.
- User Registration and Profile: Users register using email, phone number, or Google, and set preferences like trail type and difficulty.
- Personalized Route Recommendations: Based on user preferences, group size, activity interests, and real-time weather conditions.
- Live Trail Reports: Users can submit reports on hazards, crowding, or general recommendations during hikes.
- Nearby Users Communication: Users can view other hikers on the map and contact them via WhatsApp or phone (if permitted).
- Admin Panel: Authorities can review and manage user reports, update trail data, and delete inappropriate content.

Non-Functional Requirements

Performance: The app must load routes, user data, and recommendations within 3 seconds.

Scalability: Designed to support growth in both user base and geographical coverage.

Availability: Requires a stable internet connection to enable real-time services.

Security and Authorization: Only authorized users (e.g., admins) can manage sensitive content.

Usability: The interface is designed to accommodate users of varying ages and technological proficiency, from experienced hikers familiar with mobile apps to users with only basic digital skills.

System Architecture

The system is based on a three-tier architecture that includes the frontend (client-side), backend (server-side), and database layers. These components work together to deliver a responsive and personalized experience to the user.

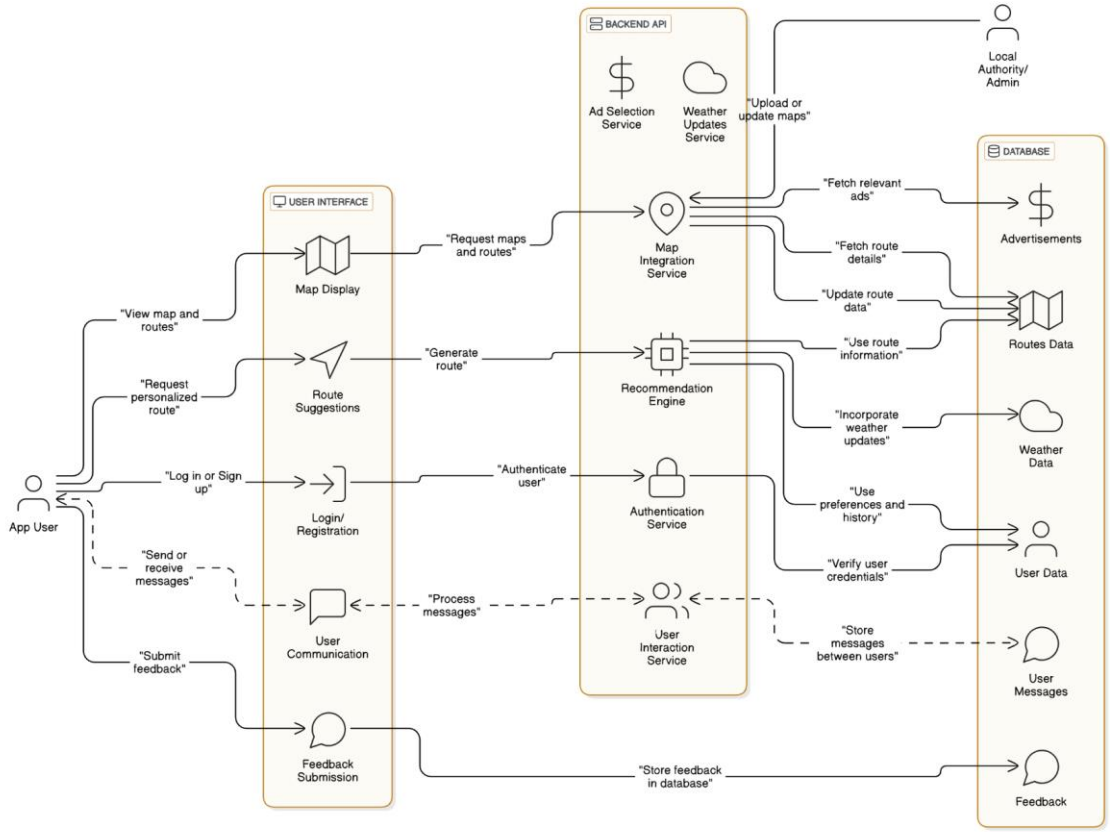
Frontend (Client-side Application)
The Android app is built using Java and XML in Android Studio.
It is responsible for presenting the user interface, capturing user input, and displaying data such as maps, recommendations, and reports.
The frontend communicates with the backend through HTTP requests (via Retrofit), sending user data, actions (e.g., registration, form submissions), and receiving structured responses.

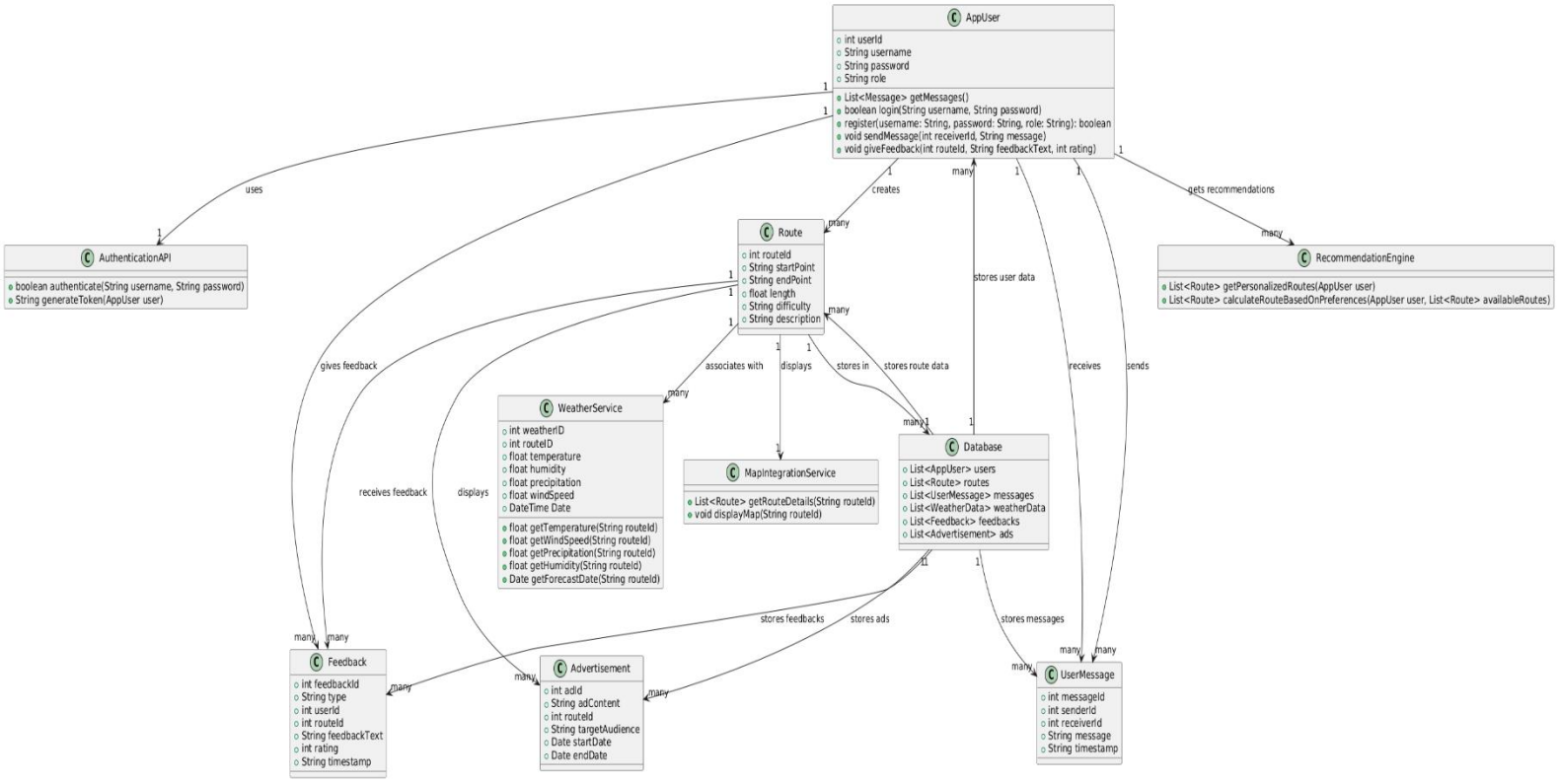
Backend (Server-side Logic)
The backend is developed in Node.js (Express framework).
It receives and processes API requests from the frontend, handles core logic such as authentication, recommendation generation, location-based user matching, and report validation.
The backend accesses the database, performs necessary operations, and returns responses to the frontend.

Database (MongoDB Atlas)
A cloud-based database stores all data, for example, users, routes, and reports.
The backend reads/writes to the database and ensures that data remains consistent and secure.
This architecture ensures modularity, maintainability, and scalability, allowing the system to evolve over time with additional features or regions.

Technology Stack

Technology	**Layer**
Android (Java/XML)	Frontend
Node.js (Express.js)	Backend
MongoDB Atlas	Database
Google Maps API, OpenWeather API, WhatsApp API	APIs
VS Code, Android Studio, MongoDB Compass	Tools


BACKEND API
Ad Selection Service
Weather Updates Service
Local Authority/ Admin
"Upload or update maps"
DATABASE
USER INTERFACE
"Fetch relevant ads"
Advertisements
Map Display
"Request maps and routes"
Map Integration Service
"Fetch route details"
"Update route data"
"View map and routes"
"Use route information"
Routes Data
"Request personalized route"
"Generate route"
Route Suggestions
Recommendation Engine
"Incorporate weather updates"
Weather Data
"Log in or Sign up"
"Authenticate user"
Login/ Registration
Authentication Service
"Use preferences and history"
App User
"Send or receive messages"
"Verify user credentials"
User Data
"Process messages"
User Communication
User Interaction Service
"Store messages between users"
User Messages
"Submit feedback"
Feedback Submission
"Store feedback in database"
Feedback


AppUser
int userId
String username
String password
String role
List<Message> getMessages()
boolean login(String username, String password)
register(username: String, password: String, role: String): boolean
void sendMessage(int receiverId, String message)
void giveFeedback(int routeId, String feedbackText, int rating)
uses
creates
gets recommendations
AuthenticationAPI
boolean authenticate(String username, String password)
String generateToken(AppUser user)
Route
int routeId
String startPoint
String endPoint
float length
String difficulty
String description
RecommendationEngine
List<Route> getPersonalizedRoutes(AppUser user)
List<Route> calculateRouteBasedOnPreferences(AppUser user, List<Route> availableRoutes)
stores user data
gives feedback
associates with
displays
stores in
stores route data
receives
sends
WeatherService
int weatherID
int routeID
float temperature
float humidity
float precipitation
float windSpeed
DateTime Date
float getTemperature(String routeId)
float getWindSpeed(String routeId)
float getPrecipitation(String routeId)
float getHumidity(String routeId)
Date getForecastDate(String routeId)
receives feedback
displays
MapIntegrationService
List<Route> getRouteDetails(String routeId)
void displayMap(String routeId)
Database
List<AppUser> users
List<Route> routes
List<UserMessage> messages
List<WeatherData> weatherData
List<Feedback> feedbacks
List<Advertisement> ads
stores feedbacks
stores ads
stores messages
Feedback
int feedbackId
String type
int userId
int routeId
String feedbackText
int rating
String timestamp
Advertisement
int adId
String adContent
int routeId
String targetAudience
Date startDate
Date endDate
UserMessage
int messageId
int senderId
int receiverId
String message
String timestamp

Methodology

Dataset(s)

TourTrack relies on three types of data sources:

1. User-Provided Data: Collected during registration and interaction with the app. Includes name, age, group size, trail preferences, reports and recommendations.

2. External Data: Google Maps API (for maps, trail overlays, and location-based queries) and OpenWeatherMap API (for real-time weather forecasts).

3. Administrative and Static Trail Data: Provided by the Lower Galilee Regional Council and other partners, including trail coordinates and attractions.

Algorithms and Models Used

TourTrack uses a custom hybrid recommendation algorithm that combines:

- Content-Based Filtering (CB): Matches trails with user-selected preferences.
- Demographic Filtering (DF): Considers factors like age and group size.
- Context-Aware Filtering: Uses weather conditions and trail availability.

The algorithm is rule-based rather than machine learning-based, allowing efficient and interpretable results on mobile devices.

User Proximity Detection

Users' locations are updated every 30 seconds.
The backend compares coordinates within a defined radius to identify nearby users.
These users are displayed on the interactive map and can be contacted if permissions allow.

Communication Control Logic

Each user controls their own communication preferences. The backend enforces these settings by preventing WhatsApp or phone calls if the recipient has disabled such options.

Report and Recommendation Management

Every report or route recommendation is linked to a user ID and timestamp.
Admin-level and authority users have extended permissions to:

- View all submitted reports
- Delete reports submitted by others, if necessary
- Detect duplicate or expired entries based on content and time

Tools and Technologies

The following tools and technologies were used:

- Android Studio (Java/XML) – Frontend development

- Node.js + Express.js – Backend logic and API server

- MongoDB Atlas – Cloud-hosted NoSQL database

- Retrofit – Android HTTP library

- Google Maps API, OpenWeatherMap API, WhatsApp – External data providers

-MongoDB Compass – Databases

- GitHub – Version control

Implementation Process

TourTrack was developed using a lightweight Agile methodology, with frequent iterations and testing at each stage.
The project began with a user-centered design phase, followed by phased backend development and integration, where each feature was connected to the frontend immediately after implementation.

<u>Development Phases:</u>

Phase 1: Mockup and UI Design
Defined user stories and user personas.
Designed the full set of screens using Wizard, including registration, preferences, main map, chat, and profile.
The entire UI structure was built first in Android Studio using Java/XML.

Phase 2: Backend Foundations
Set up the Node.js + Express backend and connected it to MongoDB Atlas.
Defined the core models: users, trails, reports, weather, and permissions.
Integrated third-party APIs: Google Maps, OpenWeatherMap, and WhatsApp.

Phase 3: Feature-by-Feature Integration
After each backend feature was developed, it was immediately connected to the relevant UI screen.

Examples: Registration + Authentication → integrated with login screen

Chat + Permissions → linked to chat screen and user settings

Phase 4: Real-Time Location and Nearby Users
Developed logic for updating user location every 30 seconds.
Backend compares coordinates to identify nearby users and return them to the map.
User permissions control who can see or contact them.

Phase 5: Testing and Refinement
Created detailed black-box tests for each flow (registration, preferences, communication, reporting).
Bug fixes and usability improvements were made continuously based on test results and user feedback.

7.5 Design Choices and Reasoning

TourTrack's design was shaped by the diverse needs of its target users, ranging from young hikers to families and older adults. The decisions made throughout the UI/UX and backend logic were driven by a desire to provide a clear, responsive, and safe travel companion.

Key Design Principles:

Map-Centered Navigation
The home screen is a live, interactive map, reflecting the app's core concept: location-based trail exploration. All primary actions (recommendations, reports, nearby users) are accessible from this central view.

Real-Time Logic Over AI Complexity
We opted for a hybrid rule-based recommendation engine rather than machine learning models. This ensures full transparency, quick responses, and better support for new users without requiring large datasets.

Inclusive Usability
The interface was designed to accommodate users with varying levels of digital proficiency.
Clear icons, large buttons, and high-contrast text were used.
The navigation flow supports quick understanding even for older or less tech-savvy users.

Color Scheme and Visual Identity

Green and beige reflect nature and calmness — echoing the hiking experience.
A clean, friendly graphic style makes the app feel trustworthy and non-intimidating.

Consistency and Feedback

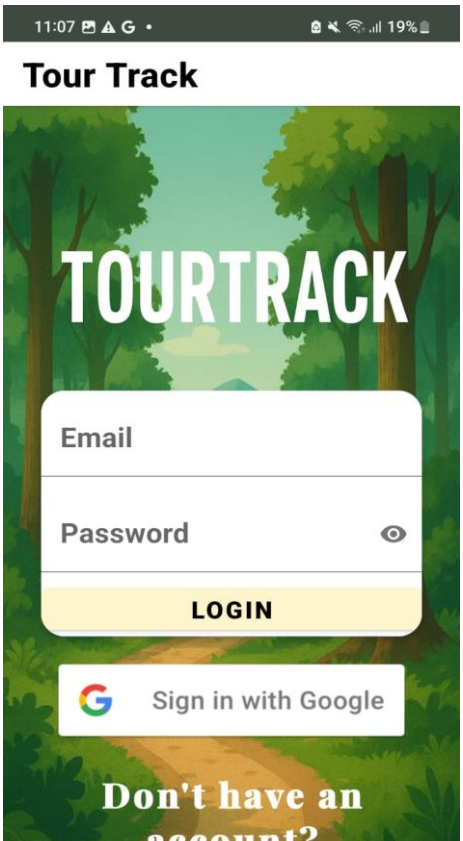
Consistent placement of elements across screens (e.g., map always central, bottom bar for navigation)
System responses (success/failure messages) are immediate and clear
All interactions (tapping on user icons, submitting reports) give the user a clear sense of control

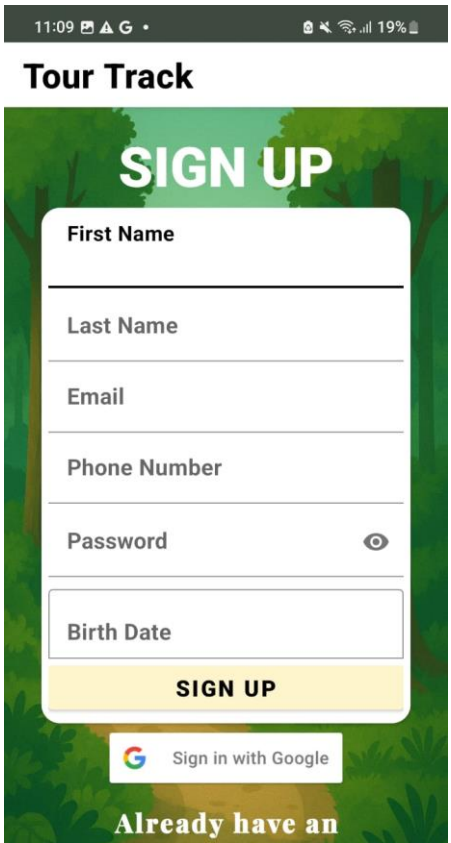
Structure of Information

Reports and route recommendations are presented in a card-based layout, making it easier to browse, tap, and act.
Filters are prominent and intuitive, allowing quick sorting by trail type, difficulty, or relevance.

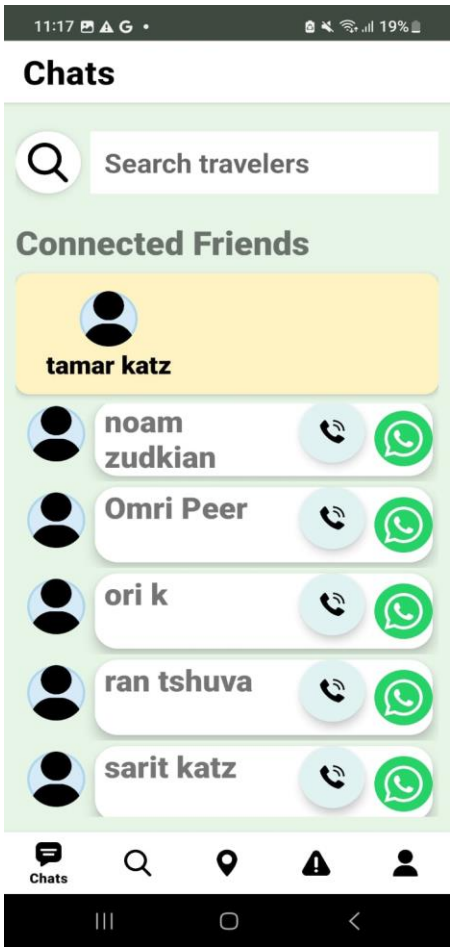
Minimizing Cognitive Load

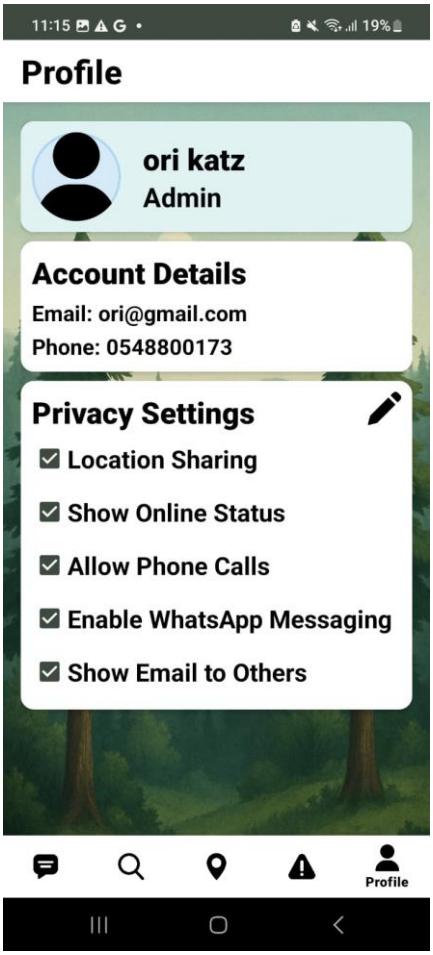
We minimized the number of screens and interactions needed to complete any action (e.g., finding a trail, submitting a report). This helps avoid overwhelming users.

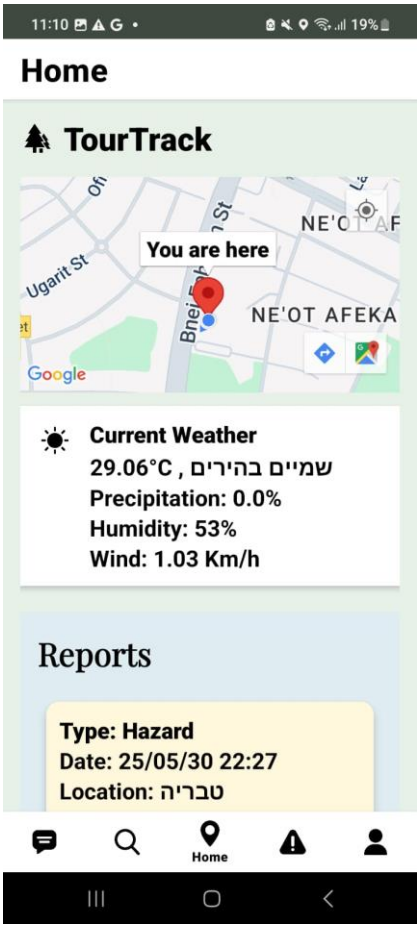


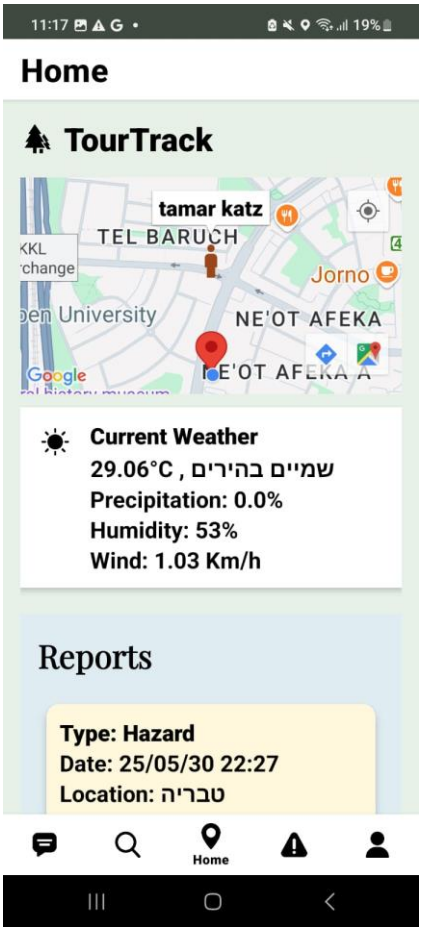


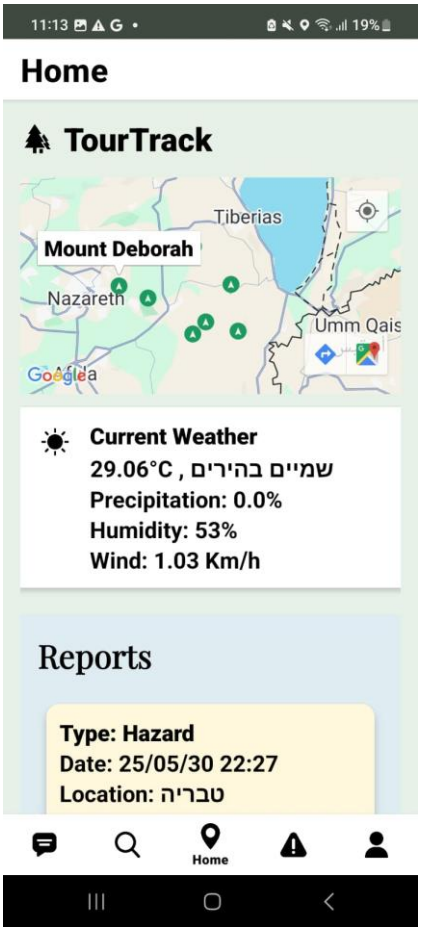

Tour Track
Select your preferred trail themes
FOREST
BEACH
RIVER
MOUNTAIN
DESERT
LAKE
CAVE
ADVENTURE
CULTURE
WATERFALL
HISTORICAL
WILDLIFE
SCENIC
AGRICULTURE
FIELD
CONTINUE


Chats
Search travelers
Connected Friends
tamar katz
noam zudkian
Omri Peer
ori k
ran tshuva
sarit katz
Chats


Profile
ori katz
Admin
Account Details
Email: ori@gmail.com
Phone: 0548800173
Privacy Settings
Location Sharing
Show Online Status
Allow Phone Calls
Enable WhatsApp Messaging
Show Email to Others
Profile


Home
TourTrack
You are here
NE'OT AFEKA
Google
Current Weather
29.06°C , שמיים בהירים
Precipitation: 0.0%
Humidity: 53%
Wind: 1.03 Km/h
Reports
Type: Hazard
Date: 25/05/30 22:27
Location: טבריה
Home


Home
TourTrack
tamar katz
TEL BARUCH
Jorno
NE'OT AFEKA
Google
Current Weather
29.06°C , שמיים בהירים
Precipitation: 0.0%
Humidity: 53%
Wind: 1.03 Km/h
Reports
Type: Hazard
Date: 25/05/30 22:27
Location: טבריה
Home


Home
TourTrack
Tiberias
Mount Deborah
Nazareth
Umm Qais
Current Weather
29.06°C , שמיים בהירים
Precipitation: 0.0%
Humidity: 53%
Wind: 1.03 Km/h
Reports
Type: Hazard
Date: 25/05/30 22:27
Location: טבריה
Home

11:17 19%

Reports

Report a Problem

Select report type:

Hazard

Location:

Add photo (optional)

What happened?

SEND REPORT

My Reports

Reports

11:14 19%

Search

Find your perfect Trail

Region in Israel:

Kfar Tavor

Trail Type Preferences

Water Trails

Difficulty Level

Easy Medium Hard

Group Size

Number of people

Age

Average age

Date Of Planning Trip

Search

11:14 19%

Search

Top Recommended Routes

Tavor River

Distance: 8 km

Difficulty: Easy

Weather: Clouds

A walk along the stream, partly shaded.

Ein Rechesh

Search

Challenges and Solutions

Challenge	Solution
Lack of real trail dataset	Collaborated with the local council to define temporary routes until the official trail data becomes available.
Internet restrictions on the college network	Used ngrok to tunnel the local backend server for external access.
Testing the nearby users feature	Manually adjusted user coordinates in MongoDB Atlas to simulate different user locations.
Testing map interactions	Simulated trail data and used location spoofing tools for movement simulation.
UX for different age groups	Tested the interface with users of different ages and adjusted layout, buttons, and labels accordingly.
Different screen sizes on our devices	Adapted layout to fit well on multiple screen resolutions with responsive design techniques.

Experiments and Results

Test Case ID	Description	Preconditions	Test Steps	Expected Result	Actual Result
TC-001	Test user registration with valid credentials	Server running, database connected	1. Open the app 2. Tap on "Sign Up" 3. Enter valid details (name, email, password, phone number, birth date) 4. Tap "Register"	User is redirected to preferences screen	The user was successfully registered and appears in the database.
TC-002	Registration with existing email or phone number	User with the same email or phone number already exists in the database	1. Open registration page 2. Enter an email or phone number already registered 3. Submit form	Error message shown: "Email or phone number already in use"	The system prevented registration and displayed an error message.
TC-003	Submit preferences	User successfully registered	1. Select trail preferences 2. Tap "Continue"	Preferences are saved and the user is redirected to the home screen	Preferences have been successfully saved to the user's data and the home screen has been loaded
TC-004	Login with incorrect password	User exists in database	1. Open login page 2. Enter correct email and wrong password 3. Submit	Error message shown: "Invalid credentials "	Error message appeared correctly.
TC-005	Display of map with user location	Location services enabled, routes in DB	1. Open the home screen 2. Grant location permission if prompted 3. Wait	Map displays user's location and the routs	View your location on the map and move around to see the

	and routes		for location to load		routes
TC-006	Real-time display of nearby users	Location permission granted, at least one nearby user exists	1.Open the home screen 2. Wait for the system to detect nearby users 3. View the map	Nearby users are shown as icons on the map. Tapping an icon reveals the user's name	Nearby users appeared on the map. User name was shown on tap
TC-007	Algorithm - Prevent form submission when fields are empty	App is running, user is logged in	1. Open the recommendations engine fragment. 2. Leave required fields (e.g., difficulty, group size, or age) empty. 3. Tap "Find Recommended Routes."	An error message appears	An error message appears indicating the missing input.
TC-008	Algorithm - Filter results by attraction	The database contains routes that are cataloged into attraction types, for example, "restaurant", "accommodation", etc. (in our case, the data only contains routes).	1. Select type attraction from the attraction dropdown. 2. Complete all fields 3. tap "Find Recommended Routes"	If you selected a trip, the first 3 routes will be displayed. If you selected anything else, a message will be displayed.	If you selected a trip, the 3 most recommended routes are displayed. If you selected anything else, a message is displayed stating that no routes exist.
TC-009	Algorithm- Show "Load More" Button	The recommendation engine	1.Complete the recommendation form and tap	A new route card appears.	A new route was loaded for us and the

		returned 3 routes and there is more data.	"Find Recommended Routes." 2. Scroll down and tap the "Load More" button.	If all routes are shown, the "Load More" button disappears and a "No routes" message appears.	load button appeared again. We pressed it until we finished the data and a message appeared: "No more routes."
TC-010	Checking the correctness of the algorithm	Existing route data in the database	1. Open the app 2. Navigate to the recommendation form 3. Fill all required fields with valid values 4. Select specific criteria (e.g., category, attraction type, difficulty level) 5. Tap "Find Recommended Routes"	The recommended routes match the selected criteria and appear sorted by relevance	The recommended routes matched the selected criteria and were displayed in relevant order.
TC-011	Chats - Display of currently connected users	User is logged in, at least one user is online	1. Open the chat screen 2. View "Connected friends" section	A list of users who are currently online (have been online in the last 10 minutes) is displayed.	Connected users appeared correctly
TC-012	Chats - Display of all registered users	User is logged in	1.Open the chat screen 2. View the "All Users" section.	A list of all app users is displayed	User list loaded successfully
TC-013	Chats -Search for a specific user	User is logged in	1. Open chat screen 2. Enter part of a name in the	Matching users are filtered and shown in real time	Search returned accurate results

			search bar		
TC-014	Chats - Call a user who allows phone calls	The target user allows calls.	1. Open the chat screen 2. Tap phone icon next to user's name	Call is initiated via phone app	Phone call initiated successfully
TC-015	Chats - Attempt to call a user who disabled phone calls	Target user has disabled phone calls	1. Open the chat screen 2. Tap phone icon next to user's name	An alert appears: "This user does not allow phone calls"	An alert message appeared as expected and the call was not placed.
TC-016	Chats - Open WhatsApp chat with a user who allows it	Target user allows WhatsApp messages	1. Open the chat screen. 2. Tap WhatsApp icon next to user's name	WhatsApp opens with that user's chat window	WhatsApp conversation opened successfully and a generic message appears
TC-017	Chats - Attempt to open WhatsApp with a user who disabled it	Target user disabled WhatsApp messages	1. Open the chat screen. 2. Tap WhatsApp icon next to user's name	Alert message: "This user does not allow WhatsApp communication"	Alert appeared as expected
TC-018	Chats - Display logged-in user's name	User is logged in	1. Open the profile screen	The user's full name is displayed at the top of the screen	User's name appeared correctly
TC-019	Chats -Update user permissions from profile screen	User is logged in	1. Open the profile screen 2. Toggle on/off permissions (e.g., allow phone calls, allow WhatsApp)	Permissions are updated in the database and reflected in the UI and other parts of the app	Permissions updated successfully

				accordingly	
TC-020	Submit a report or recommendation	User is logged in	1. Open the report screen 2. Fill in report or recommendation form Tap "Submit"3.	The entry is saved and appears under "My Reports" or "My Recommendations"	Report/recommendation submitted successfully
TC-021	View user's own reports/recommendations	User has submitted at least one item	1. Open report screen 2. Scroll to "My Reports" or "My Recommendations" section	Submitted items are listed accurately	Items loaded and displayed correctly
TC-022	Edit a report/recommendation	User is logged in and has submissions	1. Open "My Reports"/"My Recommendations" 2. Tap "Edit" on one item 3. Update and save	Changes are saved and updated immediately in the list	Edits were saved and reflected properly
TC-023	Delete a user's own report/recommendation	User is logged in and has submissions	1. Open "My Reports"/"My Recommendations" 2. Tap "Delete" on one item 3. Confirm deletion	The item is removed from the list and database	Item deleted successfully
TC-024	Display global reports/recommendations on home screen	User is logged in and location is enabled	1. Open home screen 2. Wait for location to load 3. Scroll to recommendations	Relevant reports/recommendations for the area are displayed	Items relevant to current location appeared

TC-025	Admin deletes someone else's report	Logged in as admin or authority, another user has a report	1. Go to home screen 2, Tap "Delete" on another user's report	The report is deleted from the database and removed from display	Admin successfully deleted the report

Discussion

Insights Gained

Through the implementation and testing of TourTrack, several key insights emerged:

User-centered design matters: Our early investment in personas and mockups allowed us to create an intuitive and accessible app, especially for users of different ages and technical skills.

Real-time data significantly improves user experience: Features such as nearby user display, live weather updates, and dynamic trail suggestions proved highly valuable for users planning or adjusting their hike.

Simple, rule-based recommendations were sufficient: Even without machine learning, the combination of preferences and weather filtering provided relevant and practical route suggestions.

Limitations

Despite the app's functionality and positive test results, several limitations remain:

Limited trail dataset: Since the official trail is still under development, we worked with placeholder data defined with the local council, limiting the app's initial value in production use.

No offline mode: The app requires an active internet connection for nearly all features, which may be challenging in remote hiking areas.

No real-time user authentication verification: Currently, the app does not include email or phone number verification beyond initial input, which could lead to impersonation.

Future Improvements

To address these limitations and extend TourTrack's capabilities, we suggest the following improvements:

Official trail data integration once the Tavor Trail is finalized

User verification via SMS or email to improve trust and safety

Gamification features such as badges or progress tracking to boost engagement

Conclusion and Future Work

TourTrack was developed as a smart, community-based hiking application that bridges the gap between trail discovery, live reporting, and traveler interaction. By combining personalized recommendations, real-time weather integration, and user-to-user communication, the app offers a unique experience that enhances safety, planning, and social engagement for hikers.

The project demonstrated that a rule-based approach to recommendation and a location-aware design can provide effective and practical solutions. Usability testing showed that users across different age groups were able to navigate and use the app successfully, validating the UI/UX design decisions.

In collaboration with the Lower Galilee Regional Council, the app not only supports personal exploration but also provides a foundation for improving local tourism and encouraging responsible use of natural trails.

Future Work

To further improve TourTrack and prepare it for broader use, we suggest:
Integrating finalized, official trail datasets as they become available
Expanding the recommendation logic
Developing a full admin portal for local authorities to manage data and respond to user reports
Map filters for point-of-interest categories, such as displaying only restaurants, overnight stays, or natural attractions — once the dataset becomes broader

References

[1] Kbaier, M. E. B. H., Masri, H., & Krichen, S. (2017, October). A personalized hybrid tourism recommender system. In 2017 IEEE/ACS 14th International Conference on Computer Systems and Applications (AICCSA) (pp. 244-250). Ieee.

[2] Hu, H., & Li, C. (2023). Smart tourism products and services design based on user experience under the background of big data. Soft Computing, 27(17), 12711-12724.

3. OpenWeatherMap. (n.d.). Weather API documentation.

4. MongoDB Inc. (n.d.). MongoDB Atlas documentation.

5. googleMaps API. (n.d.). googleMaps API documentation.

6, WhatsApp. (n.d.). WhatsAppAPI documentation.

7. GAFFL. (n.d.). Find a Travel Buddy to Share Costs & Experiences. https://www.gogaffl.com/

8. Travello. (n.d.). A social network for travelers. https://www.travelloapp.com/

9. AllTrails. (n.d.). Hiking & Biking Trail Maps. https://www.alltrails.com/