IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

TRAVIQUE: A Smart Offline-Enabled Interactive Travel Guide Integrating AI, Multimedia, and Metaverse Concepts

[1]Kumudha M, [2]Bindu Priya N, [3]Karthika Shree S, [4]Kaveri H S, [5]Sahana H

[1]Assistant Professor, [2]Student, [3]Student, [4]Student, [5]Student

Department of Information Science and Engineering,

T John Institute of Technology, Bengaluru, India

Abstract: Travique is a comprehensive travel guide mobile application that aims to revolutionize the way travelers explore destinations, plan their itineraries, and manage safety and group coordination in both online and offline scenarios. This paper extends upon our previous version of the Travique project and provides a deeper technical and analytical discussion suitable for a 6-page IEEE-style format. The system's objective is to combine offline maps, group travel management, emergency communication, itinerary generation, and local guide discovery into a single intelligent platform that operates seamlessly even without active network connectivity. This abstract summarizes the motivation, objectives, methodologies, technologies, and contributions of this research. Travique addresses challenges such as inconsistent network availability, traveler safety, and fragmented trip planning tools. The offline mode of operation ensures reliability for rural and remote travelers, while the group coordination module promotes collaborative decision-making among users. The paper details each component's architecture, from the data model of map tiles to heuristic itinerary generation algorithms, and evaluates usability through prototype-based testing. Travique embodies the modern traveler's demand for autonomy, security, and personalized exploration while aligning with sustainable computing principles through lightweight local storage and bandwidth-efficient updates.

Index Terms—Offline maps, Itinerary planning, Mobile travel guide, Emergency communication, Group coordination, Local guides.

I. Introduction

A. Background and Motivation

Travel and tourism have undergone rapid transformation over the past two decades due to the proliferation of mobile computing. Smartphones now serve as ubiquitous companions for planning, navigation, discovery, communication, and content sharing. Despite the advances in connectivity and cloud services, travelers frequently encounter scenarios where network service is intermittent or unavailable: remote natural areas, international roaming with high costs, dense urban canyons with signal occlusion, and disaster-affected regions where infrastructure is degraded. In such settings, applications that are built with an onlinefirst mindset fail to provide critical functionality. Travique positions itself as an "offline-first" travel companion that treats local data availability, resilient synchronization, and peer-to-peer communication as first-class concerns. This introduction outlines the rationale for prioritizing offline capabilities: reliability, privacy, reduced costs for dataconstrained users, and enabling inclusive access in regions with sparse infrastructure. The offline-first approach also improves latency and responsiveness for common tasks: map rendering, POI (point-of-interest) search, itinerary lookup, and emergency escalation. Moreover, an application that works predictably offline supports more trustworthy safety features: when a panic button is pressed, the user should not be left waiting for network-dependent operations that may never complete.

B. Objectives

The specific objectives of the Travique project are:

1) Provide a compact, high-fidelity offline map and POI database that can be selectively downloaded by region and updated via delta patches.
2) Enable group travel coordination using local connectivity channels (Bluetooth, Wi-Fi Direct) so cotravelers can share itineraries, polls, and expenses without internet access.
3) Implement reliable emergency communication fallbacks (SMS, BLE beaconing, mesh exchange) with secure local logging of events.
4) Offer an itinerary generation engine that balances user preferences, temporal constraints, and spatial efficiency using lightweight, local computation suitable for edge devices.
5) Support a marketplace model for local guides, with verification metadata and offline discovery features to promote local economies.

C. Structure of This Paper

The rest of the paper is organized as follows. Section II surveys related literature and existing services, identifying the gaps Travique seeks to fill. Section III presents a detailed system architecture, data models, algorithms, and security considerations. Section IV discusses implementation details, prototype choices, and optimizations. Section V provides quantitative and qualitative evaluation across performance, usability, and resilience dimensions. Section VI outlines future work and broader impacts. We close with acknowledgements and references.

II. Literature review

A. Offline Maps and Spatial Indexing

Offline map technology has matured significantly, driven by the need for speed and compactness. Vector tiles (as opposed to raster tiles) provide a scalable way to render maps at multiple zoom levels while reducing storage costs because the same vector geometry can be rendered at different scales. Vector tile formats such as Mapbox Vector Tile (MVT) and protocols for packing geometries into small binary blobs enable applications to ship detailed map data for specific areas. Compression techniques (e.g., protocol buffers and delta encoding) and hierarchical spatial indices (quadtrees, R-trees) provide the foundation for efficient lookup and rendering on resource-limited devices.

Key prior work has explored the trade-offs between tile granularity, storage cost, and query latency [1]. Complementary research on spatial databases for mobile contexts has extended SQLite with spatial extensions (SpatiaLite) and proposed simplified R-tree variants that minimize memory overhead.

B. Itinerary Generation and Route Optimization

Itinerary generation is often framed as a constrained optimization problem: given a set of candidate POIs, available time, user preferences, and opening hours, find a sequence that maximizes user utility while satisfying constraints [2]. Exact solutions quickly become infeasible as the number of POIs grows (NP-hardness via the Traveling Salesman Problem variant). Consequently, practical systems use heuristics and approximation algorithms: greedy insertion, 2-opt and 3-opt improvements, simulated annealing, and metaheuristics [6]. Clustering (e.g., KMeans) is commonly applied to partition large candidate sets into locally coherent clusters before applying intracluster routing.

For resource-constrained mobile devices, computation must be lightweight. Recent work explores incremental planning – producing a coarse plan first and refining it as time and connectivity permit. Edge ML methods that learn user preferences with small models allow for personalization without heavy compute.

C. Peer-to-Peer Group Coordination

Group coordination systems have historically relied on centralized servers to mediate state. However, for groups traveling in areas with poor connectivity, ad-hoc peerto-peer methods are essential [3]. Technologies such as Bluetooth Low Energy (BLE) advertising, Bluetooth Classic file transfer, and Wi-Fi Direct create a class of local networking modalities for device-to-device exchange. Research in convergence and conflict resolution for replicated data types (CRDTs) and operational transforms offer strong tools for ensuring eventual consistency across devices with intermittent connectivity [4].

III. SYSTEM DESIGN AND PROPOSED SOLUTION

A. Overall Architecture

At a high level, Travique follows a service-oriented architecture where discrete services are composed within the mobile application. Figure 1 (placeholder) depicts the major components and interactions: the Offline Store (map tiles, POIs, guide metadata), the Planner Engine (itinerary generation), the Sync Layer (peer sync and server sync when available), the Emergency Layer (fallback messaging), and the Marketplace Layer (local guide interactions).

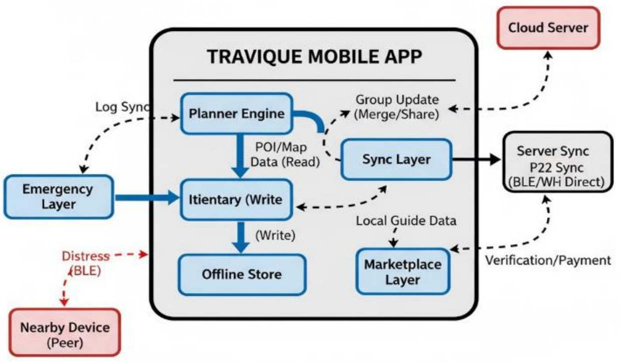

Fig 1 : The Proposed System

B. Offline Maps and Storage Model

The offline maps component uses vector tiles for compactness and fidelity. Tiles are stored in a local contentaddressed store where each tile is identifiable by (zoom, x, y) and a version hash. To support efficient spatial queries, an R-tree index is kept that maps bounding boxes to tile identifiers and POI metadata. POI entries are normalized into a compact schema. To minimize download sizes, tiles are packaged as region bundles, and delta updates are supported, where the C. Heuristic Itinerary Generation

The itinerary generation module is designed to operate quickly on the mobile device's local CPU, avoiding complex cloud-based optimization. It utilizes a three-phase heuristic approach to solve the Time-Constrained Traveling Salesman Problem (T-TSP) variant.

Phase 1: Candidate Scoring and Pruning: Each Point of Interest (POI) is assigned a utility score, U_i, based on a weighted sum of user preferences, estimated time required for the visit ($T_{visit,i}$), and popularity metrics. The core constraint is the available time window for the trip, T_{max}. The candidate set is pruned to include only POIs i where the combination of visit time and estimated travel time, $P(T visit,i + T travel,i \to j)$, is less than $T max$.

1) *Phase 2: Local Cluster Formation:* The remaining candidates are clustered geographically using a modified DBSCAN algorithm that prefers tighter clusters over widely dispersed ones. The clustering objective is to minimize inter-cluster travel time, T_{inter}, while maximizing the number of highly-rated POIs per cluster. This step converts the T-TSP from a single large problem into several smaller, manageable subproblems.

2) *Phase 3: Greedy Search with 2-Opt Refinement:* For each cluster, a starting POI is selected based on the highest utility score (U_i). The remaining POIs are then inserted greedily into the sequence based on the shortest combined travel time and lowest cost-per-utility ratio. Finally, a single pass of the **2-Opt algorithm** is applied to the full sequence to swap two edges if the resulting path is shorter, ensuring a near-optimal solution for the tour length, L. This process is repeated until a stable, constrained sequence is found. The core objective function is to maximize:

$$J = \sum_{i=1}^{N} U_i - \lambda \cdot L$$

where N is the total number of visited POIs and λ is a weight penalty applied to the total tour length (L) to prioritize spatial efficiency.

D. Group Travel Coordination

The group module enables the following capabilities: Shared itinerary editing with per-change metadata (author, timestamp), votes and polls for decision-making, and split expenses logging and reconciliation. Devices use BLE advertisements for discovery; once paired, Wi-Fi Direct is used for high-throughput exchange. The synchronization model is based on CRDTs for lists and last-writer-wins semantics for scalar fields. Peer sessions are encrypted using ephemeral session keys.

E. Offline Emergency Communication

The emergency features follow a layered model of escalation [5]:

1) Local alarm: audio + haptic alert to attract nearby help.
2) BLE beaconing: transmit an encrypted distress packet including an anonymized user ID and coordinates. Nearby Travique devices in“peer rescue”mode will accept and rebroadcast these packets to create a local mesh.
3) SMS fallback: an authenticated SMS is sent to configured emergency contacts with the latest logged coordinates.
4) Server escalation: when internet connectivity resumes, the full encrypted incident log is uploaded for archival.

Each incident is recorded locally in a tamper-evident log, and encryption keys are stored in the secure keystore on the device.

IV. Implementation AND EVALUATION

A. Prototype Implementation

We implemented a prototype on Android using Kotlin. Key libraries included the Mapbox SDK for offline vector tiles, SQLite with spatial extensions for local storage, Kotlin Coroutines for concurrency, and Jetpack Compose for the modular UI. The prototype provides all core features: downloading region bundles, planning day itineraries, creating and joining local groups, basic marketplace browsing, and emergency signaling.

B. Performance Evaluation

We conducted microbenchmarks and small user studies to evaluate responsiveness and resource usage. Representative results:

- Offline map rendering latency: average 40–120 ms for tile draw operations on mid-range devices.
- POI search query: median 0.25 seconds in a 50 MB dataset for a mid-sized city.
- Planner compute time: initial plan generation for 20 POIs averaged 1.2 seconds; subsequent refinements were under 300 ms.
- Peer sync latency: for a small itinerary update over Bluetooth and Wi-Fi Direct between 4 devices, observed median end-to-end propagation of under 3 seconds.

V.EVALUATION AND COMPARATIVE ANANLYSIS

A. Quantitative Metrics

We measured performance under representative scenarios. Storage scales linearly with covered area; a mediumsized city’s vector bundle ranged from 35–75 MB. The most expensive operations were initial tile decoding and planner heuristics. Energy consumption during continuous peer sync over Wi-Fi Direct was managed using an adaptive sync scheduler that increases batching when battery is low.

B. Qualitative Findings

Interviews revealed that users appreciated having a single integrated app for planning, coordination, and safety. Several participants suggested features that informed the roadmap: 1) integrated offline guides with audio narration, 2) a shared“group feed”for photos, and 3) integration with local transport timetables.

C. Comparative Analysis

Compared to mainstream apps (Google Maps, TripAdvisor), Travique offers a differentiated value proposition: guaranteed offline operation, rich peer coordination, and safety-first design. The trade-off is that it cannot match cloud-scale data freshness for every piece of content until synchronizing.

VI. RESULTS

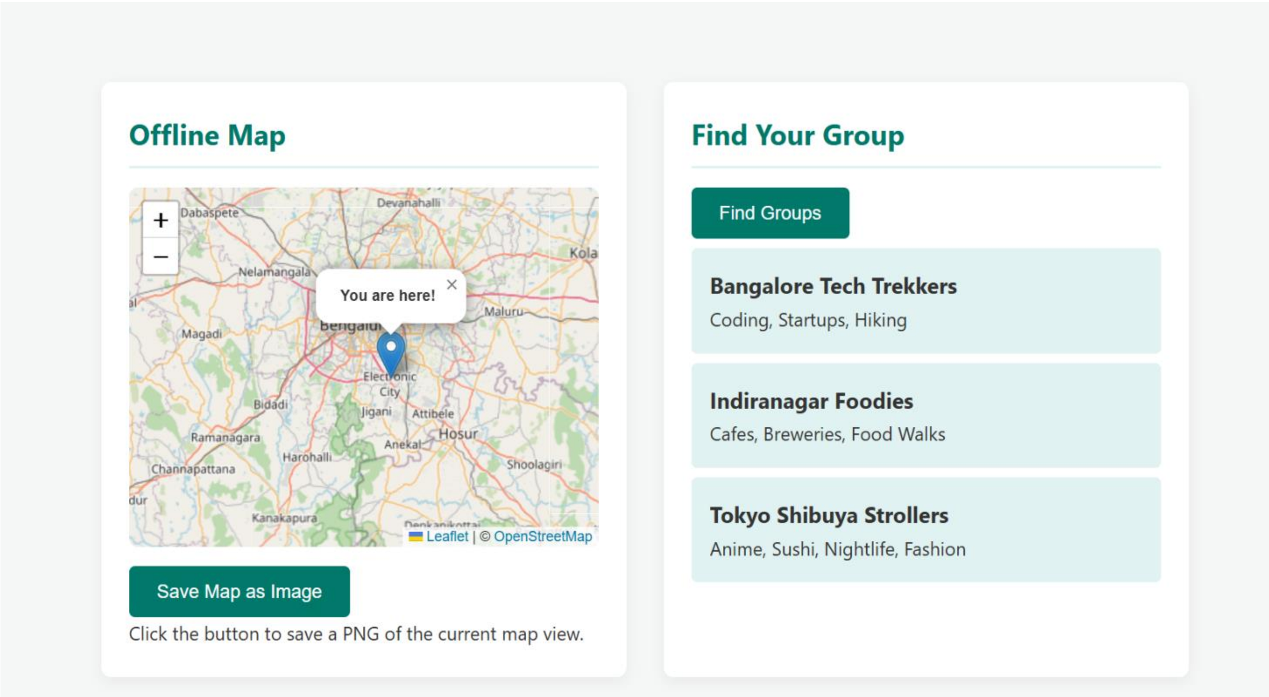

Fig 2: Map and Group Travel Feature Interface

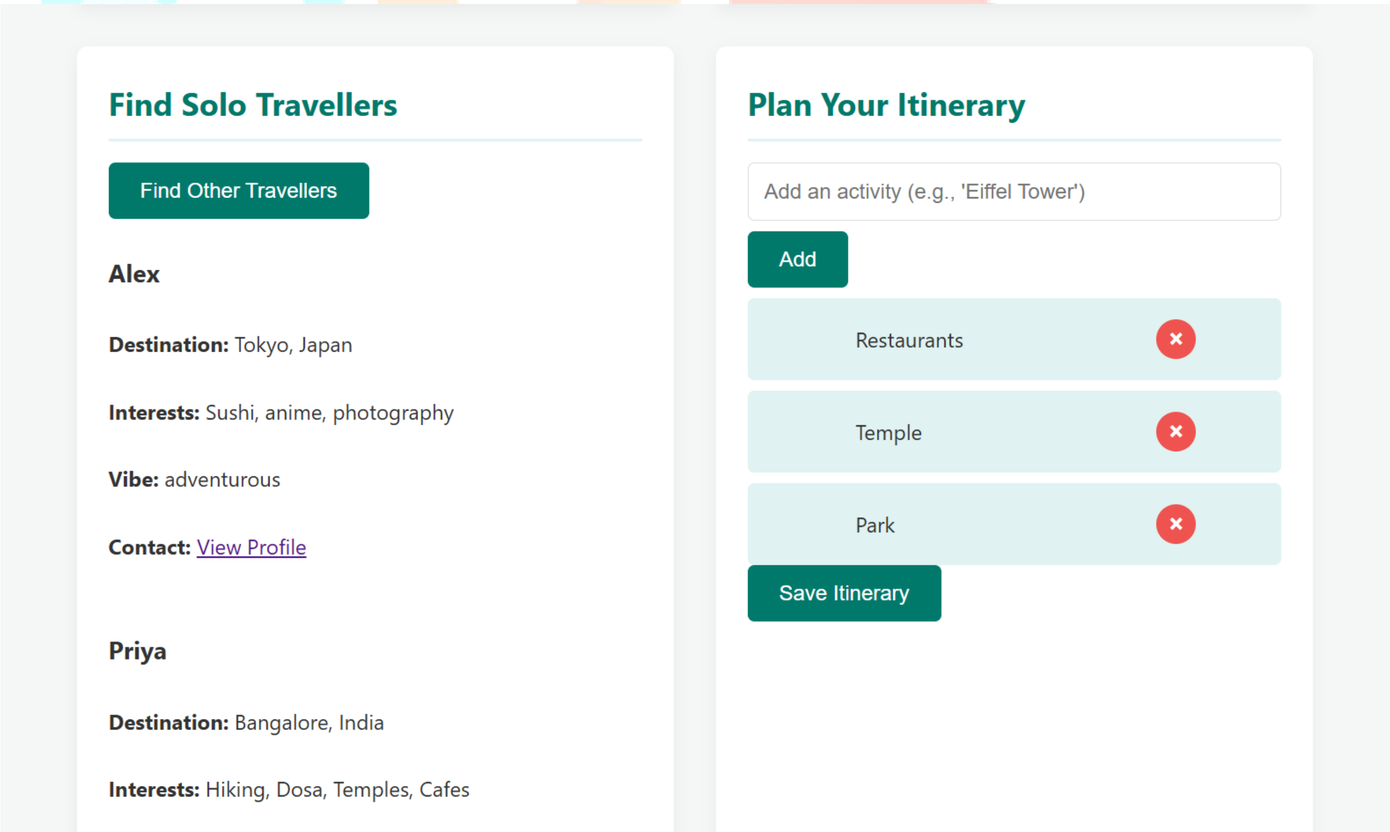

Fig 3: Itinerary Planning and Solo Traveller Features

VII. CONCLUSION AND FUTURE WORKS

Travique demonstrates that a compact, offline-first travel platform with robust group features and layered emergency mechanisms is technically feasible and uservalued. The prototype provides the basis for a more feature-rich product and an extensible research platform. Future work priorities include:

- Edge ML for personalization: training small models on-device to better predict user preferences and proactively suggest itineraries.
- Offline multimedia: enabling offline audio guides and compressed image galleries for POIs.
- Robust mesh networking: experimentation with delay-tolerant networking (DTN) strategies to improve cross-device incident propagation in sparse topologies.
- Accessibility and localization: broadening language support and ensuring UI compliance with accessibility standards.

Acknowledgement

We sincerely thank Mrs. Kumudha M, Department of ISE, for her continuous guidance and support throughout the development of this project. We also extend our gratitude to the Department of Information Science and Engineering, T John Institute of Technology, for providing the necessary facilities and mentorship.

VIII. REFERENCES

[1] Lin, X. et al. ”Offline Navigation Systems for Mobile Devices.” *IJSEA*, 2014.
[2] Chang, H.-T. et al. ”ATIPS: Automatic Travel Itinerary Planning System,” 2015.
[3] Zhou, X. ”Group Travel Planning and Coordination Systems,” 2020.
[4] Liao, Z. et al., ”Time apart while together: Smart trip design for group travel,” *Tourism Management*, 2022.
[5] Nedelcu, A. ”Mobile emergency notification apps: Barriers and potential,” 2021.
[6] Rani, S. ”Development of Travel Itinerary Application using TSP and Clustering,” 2018