

JAIN SAINT LOCATOR

B.Tech Project Report

Divya Sankhala (B22ES016) Mohanshi Jain (B22ES014) Abhinav Swami (B22AI003)

A community-driven, privacy-respecting mobile application to help devotees track publicly available movements, events, and visits (vihar) of Jain saints. The app centralizes information usually shared informally into a respectful, structured, and reliable platform while preserving dharmik decorum and saint privacy.

1. The Idea

A community-driven, privacy-respecting mobile application that helps devotees track publicly available movements, events, and visits (vihar) of Jain saints. The app centralizes information that devotees often receive informally (WhatsApp groups, pamphlets, word of mouth) into a respectful, structured, and reliable platform without compromising dharmik decorum or saint privacy.

Primary goals

- Create a trusted contributor network of verified volunteers and temple committees for saint visit updates.
- Maintain strict privacy: post only publicly shareable or consent-approved information.
- Make it easy to search for saints, follow regions, and receive notifications when saints visit subscribed areas.
- Allow contributors to manage groups (sangh) of saints and update group locations.
- Provide both map and list views of saint updates and events.

2. Implemented Features

Below we list the main features implemented (concise, user-facing descriptions) and important technical points only where essential to understand behaviour.

2.1. User Roles & Authentication

- Phone number based sign-in (Firebase Authentication).
- Three user categories: Viewers, Signed-In Viewers, Volunteers.
- Volunteers have permissions to add or update saint and event information.

2.2. Saint Profile Management

Each saint profile stores:

- Name (English and Hindi), sect, amber (Digamber/Shwetamber), gender, designation (Acharya, Muni, Sadhvvi, etc.)
- Guru, group membership, and date of Diksha.
- Linked guru and group leader information is shown automatically when available.

2.3. Events

- Support for event types: Pravachan, Diksha, Pratishttha, Vihar, Meeting, etc.
- Events include location, optional coordinates (posted only if consented), date & time, and optional poster/photo.
- Host saint is shown automatically on event cards.

2.4. Search & Filter

- Search saints by name in English and Hindi (basic two-way search implemented).
- Client-side filters combine event type and keyword search for instant responsiveness.

2.5. Map & Nearby Saints

- Map view shows saint pins when coordinates are available and centers on user location when permission is granted.
- Straight-line distances (Haversine) are used to show a "Saints Nearby" list sorted by distance.
- Tapping a saint recenters the map on that saint; directions link opens device maps.

2.6. Viewer Capabilities

- View saint profiles, events, and location updates; switch between Map and List views.
- Comment on updates. Anonymous viewers supported with limited access.

2.7. Volunteer Features (Implemented / Partial)

- Volunteers can post event information and upload posters/photos (uploads restricted to volunteers).
- Volunteers can submit vihar/location updates for saints or groups. (Verification and full dashboard features are in backlog.)

2.8. Real-time Updates

All location and event updates sync in real time across clients so users see changes instantly.

3. Planned Development (Next Sprint)

3.1. Volunteer Dashboard

- Create/manage sangh/groups: group creation, leader assignment, members list, group location, and lastUpdated timestamp.
- Enforce single-group membership for saints.
- Batch update behaviour: updating a group location updates cached location for members while preserving historical records.

3.2. Location Posting (Vihar Updates)

- Volunteers post latest known location (address, date, time, coords if public); updates are visible in real time.
- Historical vihar trails stored in `travel_history` for audit and display.

3.3. Volunteer Verification System

- Upload endorsement certificate during onboarding; admin panel to approve/deny applications.
- Badge system for trusted contributors.

4. Future Suggestions (Further Enhancements)

- A.** Enhanced Search & Multilingual Support: Advanced fuzzy search in Hindi & English, transliteration-based search, and ranking by recency and relevance.
- B.** Region-based Alerts: Users subscribe to regions and receive push notifications when a saint visits nearby (device location used only with permission; no continuous tracking of saints).
- C.** Enrich Saint Profiles: Add pratishtha details, pravachan lists, diksha history, guru lineage tree, and photos (only when available and permitted).
- D.** Event Expansion: Recurring events, multiple saints per event, automatic reminders, poster templates.
- E.** Community Feed: Timeline-style feed for recent vihar updates and events.
- F.** Admin & Moderator Tools: Flagging and review, volunteer management, approve/reject saint additions, analytics dashboards.
- G.** UI / UX Enhancements: Improved map clustering, photo gallery, cleaner saint profile

UI, quick filters ("Near Me", "Today", "This Week").

5. Technical Stack Summary

- Framework: React Native (Expo) — single TypeScript codebase for iOS / Android.
- Language: TypeScript.
- Database / Backend: Google Firebase (Cloud Firestore).
- Navigation: Expo Router (file-based: `index.tsx`, `map.tsx`, `events.tsx`).
- Mapping: React Native Maps (Google Maps provider).
- Location: Expo Location (user GPS with permission).
- UI: Core React Native components, StyleSheet, Expo Vector Icons (Feather, FontAwesome).
- Cloud Functions: Background tasks (populate guruName, sync group locations, verification workflows).

6. App Screens & Behaviour

6.1. App Foundation (`_layout.tsx`)

App shell with bottom tab bar: Home, Map, Events, Profile. Sets global styles and holds screens together.

6.2. Home Screen (`index.tsx`)

- Loads and displays full saints list from Firestore; shows a loading spinner while fetching.
- Scrollable saint cards with name, designation, and location.
- Client-side search filters downloaded list for instant results.
- View Details navigates to the saint's profile.

6.3. Map Screen (`map.tsx`)

- Requests GPS permission; fetches saints and user location.
- Calculates straight-line distance via Haversine; displays "Saints Nearby" sorted by distance.
- Interactive map with pins (coords present). Tapping a pin recenters the map; Get Directions opens device maps.

6.4. Events Screen (`events.tsx`)

- Fetches `events` collection and lists upcoming events.
- `formatDate` helper formats Firestore timestamps.
- Combined filters: select event types (All, Pravachan, Satsang) and search by keyword.

- Each event links to its details page and host saint profile.

6.5. Event Detail Screen (`[id].tsx`)

Dynamic route that fetches event details and host saint profile; shows title, description, time, and host saint section. UI placeholders for Get Directions and Add to Calendar.

6.6. Profile Screen (`Profile.tsx`)

New users start in Viewer Mode: view followed saints (coming soon), profile info, settings, and About section. Register as Volunteer opens a request form saved in Firestore for admin review.

7. Security & Rules (High-level)

Use `isVolunteer` and `volunteerMeta.approved` as guard conditions in Firestore security rules for writes to saints, events, and posters. Ensure only verified volunteers or admins can create/modify group locations and post coordinates.

Summary

The Jain Saint Locator is a community-driven platform designed to help devotees easily track the public movements, vihar routes, and event schedules of Jain saints. Instead of depending on scattered WhatsApp forwards or verbal updates, this platform organizes verified information into a structured format with privacy-preserving safeguards. Users can view current saint locations, historical routes, and upcoming events, supported by user-submitted data that can be verified before display.

Important Links

- **GitHub Repository:** [github-jain-saint-locator](https://github.com/jain-saint-locator)
 - **Firebase Console:** [firebase-jain-saint-locator](https://console.firebase.google.com/project/firebase-jain-saint-locator)
-