

Coastal safety and tourism suitability app

Prepared by: Shrey Ingole, Sanskruti Avhale, Ashish Mundkar, Sakshi Nagmode

Institution: BRACT's Vishwakarma Institute of Information Technology, Pune

1. Introduction

1.1 Purpose

This document outlines the detailed Software Requirements Specification (SRS) for the Beach Safety App. It is intended to be used by the developers, testers, and stakeholders to understand and agree upon the functional and non-functional requirements of the system.

1.2 Scope

The Beach Safety App is a mobile-first solution aimed at providing real-time environmental alerts and safety recommendations for beachgoers. It leverages geolocation, marine data, and weather feeds to alert users proactively. The solution will support both Android and iOS platforms, with features like offline mode, interactive maps, and role-based access control.

1.3 Definitions, Acronyms, and Abbreviations

- API: Application Programming Interface
- JWT: JSON Web Token
- LBS: Location-Based Services
- Redis: Remote Dictionary Server (In-memory key-value store)
- UI/UX: User Interface/User Experience
- CRUD: Create, Read, Update, Delete
- INCOIS: Indian National Centre for Ocean Information Services

1.4 References

- [1] Surf Life Saving Australia, BeachSafe App
- [2] FEMA App
- [3] MyShake App
- [4] PostgreSQL Documentation
- [5] Redis Official Docs
- [6] FastAPI Docs
- [7] INCOIS API
- [8] StormGlass API

1.5 Overview

This document is structured into sections that include system features, external interfaces, functional and non-functional requirements, and system attributes, each explained in detail.

2. Overall Description

2.1 Product Perspective

The Beach Safety App is a standalone mobile application that integrates with third-party weather and marine data providers via REST APIs. It uses a modular three-tier architecture comprising a mobile frontend (Flutter), a backend API server (FastAPI), and a database/cache layer (PostgreSQL, Redis).

2.2 Product Functions

- Real-time safety and weather alerts
- Geolocation-based beach hazard information
- Interactive map showing safety indicators
- Admin dashboard to override alerts
- Support for offline access
- Scheduled environmental data updates

2.3 User Classes and Characteristics

- Tourists & Public Users: Can receive alerts and view beach conditions.
- Lifeguards & Local Authorities: Can update or verify safety status.
- Admins: Can access dashboard for full control and analytics.

2.4 Operating Environment

- Platforms: Android (10+), iOS (13+)
- Backend: FastAPI on Ubuntu cloud server
- Database: PostgreSQL 14
- Cache: Redis 7
- Scheduler: APScheduler (Python)

2.5 Design and Implementation Constraints

- Reliance on external APIs for real-time data
- Battery optimization on mobile devices
- Offline data validity may be limited to last known update

2.6 User Documentation

- User Guide
- Admin Manual
- Setup Guide for Deployment

2.7 Assumptions and Dependencies

- Users have smartphones with GPS and data connection
- APIs (e.g., StormGlass, INCOIS) are available with valid data

System Architecture Diagram

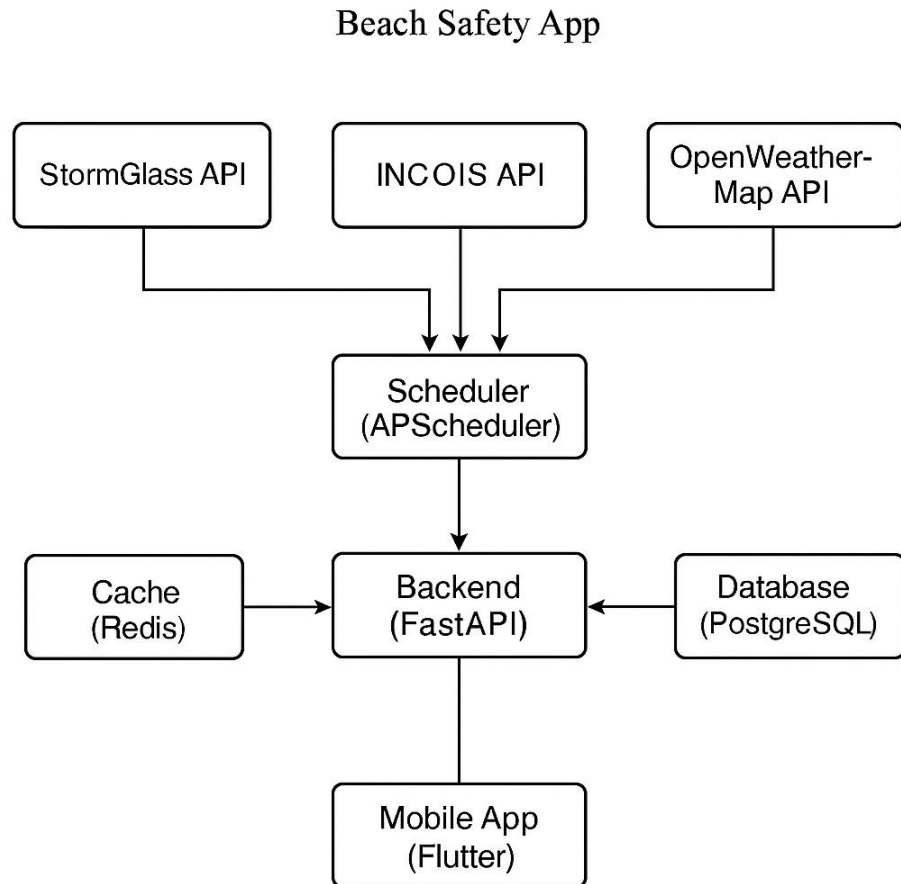


Figure 1: System Architecture