**Geolocation Gallery Application Documentation**

**Table of Contents**

1. [Introduction](#introduction)

2. [System Architecture](#system-architecture)

3. [Installation and Setup](#installation-and-setup)

4. [Core Components](#core-components)

5. [Database Schema](#database-schema)

6. [Geolocation Services](#geolocation-services)

7. [User Interface](#user-interface)

8. [Permissions Handling](#permissions-handling)

9. [Error Handling](#error-handling)

10. [Performance Optimization](#performance-optimization)

11. [Testing Strategy](#testing-strategy)

12. [Security Measures](#security-measures)

1. **Introduction**

**Purpose**

The Geolocation Gallery Application is a mobile application developed using React Native and Expo, designed to provide users with a comprehensive image management system that integrates geolocation tracking and local storage.

**Key Objectives**

- Capture and store images with location metadata

- Provide an intuitive gallery browsing experience

- Enable geospatial image visualization

- Implement robust local data management

2. **System Architecture**

**Technology Stack**

- Frontend: React Native

- Runtime: Expo

- Database: SQLite

- Geolocation: Expo Location

- Navigation React Navigation

**Component Layers**

1. **Presentation Layer**

- UI Components

- Screen Layouts

- Interactive Elements

2. **Service Layer**

- Database Operations

- Geolocation Services

- Image Processing

3. **Data Layer**

- SQLite Database

- Local Storage

- Geolocation Data Management

3. **Installation and Setup**

**Prerequisites**

- Node.js (v16+)

- npm or Yarn

- Expo CLI

- Android Studio / Xcode

**Development Environment Setup**

```bash

# Clone the repository

git clone https://github.com/yourusername/geolocation-gallery-app.git

# **Navigate to project directory**

cd geolocation-gallery-app

# I**nstall dependencies**

npm install

# **Install Expo-specific packages**

expo install expo-location expo-sqlite expo-image-picker

```

**Configuration Files**

- `app.json`: Expo application configuration

- `babel.config.js`: Babel transpilation settings

4. **Core Components**

Image Gallery Component

- Grid/List view of images

- Full-screen image viewer

- Carousel navigation

- Pinch-to-zoom functionality

Map Integration Component

- Displays image locations

- Interactive markers

- Zoom and pan capabilities

**Database Management Component**

- CRUD operations

- Image metadata handling

- Performance-optimized queries

5. **Database Schema**

**Images Table**

```sql

CREATE TABLE images (

id INTEGER PRIMARY KEY AUTOINCREMENT,

file\_path TEXT NOT NULL,

timestamp DATETIME DEFAULT CURRENT\_TIMESTAMP,

latitude REAL,

longitude REAL,

location\_name TEXT,

tags TEXT,

description TEXT

);

```

**Indexing**

- Index on `timestamp`

- Index on `latitude` and `longitude`

6. **Geolocation Services**

**Location Tracking**

- Continuous background tracking

- Accuracy configurations

- Battery optimization modes

**Geocoding**

- Convert coordinates to readable addresses

- Reverse geocoding support

**Location Caching**

- Implement intelligent caching mechanism

- Reduce unnecessary location queries

7. **User Interface**

**Design Principles**

- Responsive layout

- Adaptive to different screen sizes

- Intuitive navigation

- Consistent color scheme and typography

**Key Screens**

1. Gallery Dashboard

2. Image Details View

3. Map Visualization

4. Search and Filter Screen

5. Settings and Permissions

8. **Permissions Handling**

**Required Permissions**

- Camera access

- Location services

- Storage read/write

- Background location (optional)

**Permission Workflow**

1. Graceful permission requests

2. Clear explanation of permission usage

3. Fallback mechanisms for denied permissions

4. Settings deep link for manual permission adjustment

9. **Error Handling**

**Error Types**

- Network errors

- Location service unavailable

- Database connection issues

- Image processing failures

---

\*\*Version\*\*: 1.0.0

\*\*Last Updated\*\*: [2024/12/03]

\*\*Maintained By\*\*: [Thapelo Somo/Mlab]