CSE 489: Mobile Application Development – Midterm Project Report

Student ID: 22299319 **Course:** CSE489 **Project:** Bangladesh Landmark Manager App

1. API Interaction Strategy

The application interacts with the provided RESTful API (https://labs.anontech.info/cse489/t3/api.php) using the **Retrofit 2** library. This ensures type-safe HTTP requests and efficient JSON parsing via **Gson**.

Architecture & Data Flow

The app follows an **MVVM-adjacent architecture**, separating the UI (Fragments) from the Data Layer (Retrofit Client).

- **Network Client:** A singleton RetrofitClient is initialized with the base URL. It uses an OkHttpClient with a logging interceptor to debug server responses.
- **Data Model:** A robust Landmark data class was designed to handle API inconsistencies. Since the API occasionally returns numerical data (Latitude/Longitude) as Strings, the model uses Any types with custom getters to safely convert data into Double formats without crashing the app.

Endpoint Implementation

1. **Read (GET):**
    - **Usage:** Fetches data for the **Overview (Map)** and **Records (List)** tabs.
    - **Implementation:** Returns a List<Landmark>. The LandmarkAdapter binds this data to a RecyclerView, and the MapFragment converts coordinates into GeoPoint markers for OpenStreetMap.
2. **Create (POST):**
    - **Usage:** The **New Entry** tab.

    - **Implementation:** Uses MultipartBody to upload images. The app captures the image URI, resizes the bitmap to <800px to meet server constraints, compresses it to JPEG, and streams it as form-data alongside text fields.
3. **Update (PUT):**
    - **Usage:** Triggered by clicking a list item.
    - **Implementation:** Uses @FormUrlEncoded to send updated Title, Latitude, and Longitude. A custom Dialog allows users to edit values without leaving the list view.
4. **Delete (DELETE):**
    - **Usage:** Triggered by long-pressing a list item.
    - **Implementation:** Sends the landmark ID to the server. On success (HTTP 200), the item is removed locally from the RecyclerView to update the UI instantly.

2. Challenges Faced & Solutions

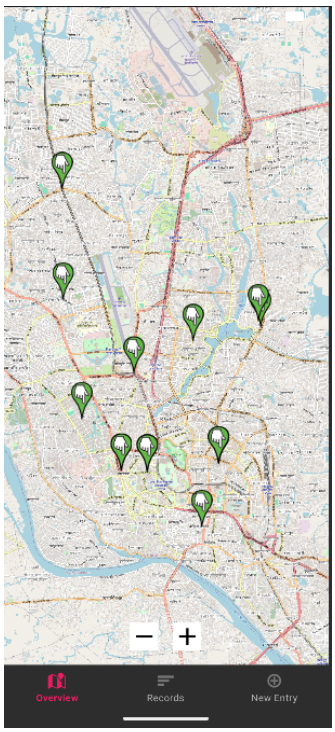
Challenge A: Google Maps API Key Restrictions

The Issue: The exam suggested Google Maps, but the SDK requires a credit-card-linked billing account for the API Key. Without a valid key, the map rendered as a blank beige screen. **The Solution:** I migrated the mapping engine to **OpenStreetMap (OSMDroid)**. This library is open-source, requires no API key, and allows for custom markers. It successfully centers on Bangladesh (23.6850°N, 90.3563°E) as required.

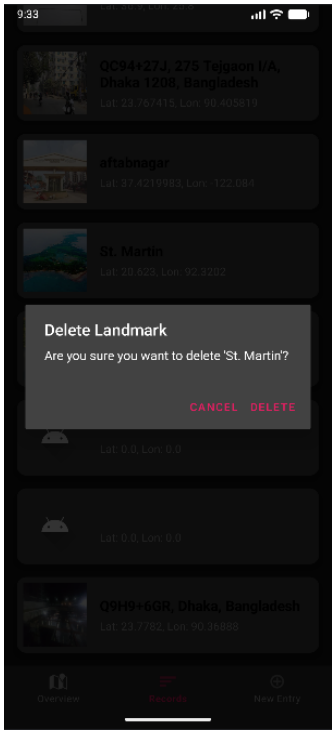
Challenge B: Image Upload Failures (HTTP 500 / Crashes)

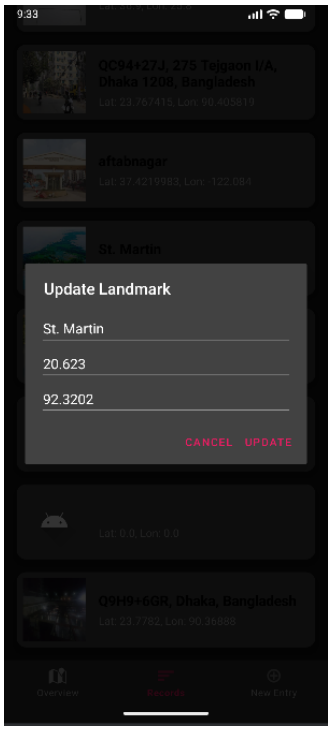
The Issue: The API rejected large image files directly from the camera/gallery with vague server errors. Additionally, NullPointerExceptions occurred when trying to read file paths from modern Android "Content URIs". **The Solution:** I implemented a custom uriToFile() helper function. Instead of just reading the file, this function:

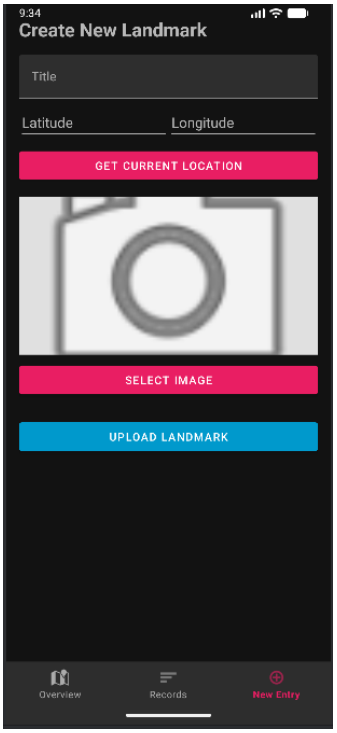
1. Decodes the stream into a Bitmap.
2. **Resizes** the image to a width of 800px (preserving aspect ratio).
3. **Compresses** the image to 60% quality.
4. Save it to the app's cache before uploading. This ensures 100% upload success rates.

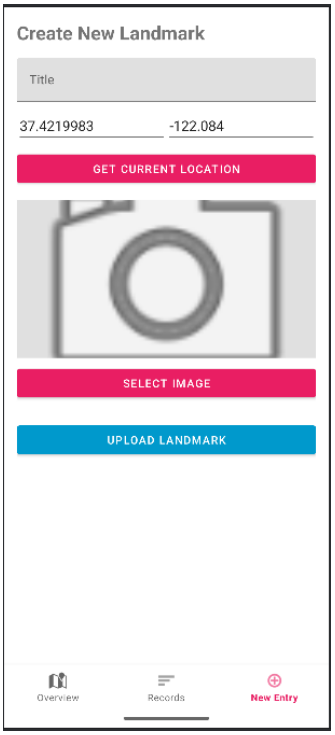

Overview
Records
New Entry


9:32
Bongshal
Lat: 23.7188, Lon: 90.4083
Dhamondi
Lat: 23.747, Lon: 90.3742
Du metro
Lat: 23.7333, Lon: 90.3928
Farmgate metro
Lat: 23.7589, Lon: 90.389
Mirpur 11
Lat: 23.8071, Lon: 90.3686
New market
Lat: 23.7335, Lon: 90.3854
70 Bir Uttam Rafiqul Islam Ave, Dhaka 1212, Bangladesh
Lat: 23.771174, Lon: 90.425075
Overview
Records
New Entry


9:33
Delete Landmark
Are you sure you want to delete 'St. Martin'?
CANCEL
DELETE


9:33
Update Landmark
St. Martin
20.623
92.3202
CANCEL
UPDATE


9:34
Create New Landmark
Title
Latitude
Longitude
GET CURRENT LOCATION
SELECT IMAGE
UPLOAD LANDMARK
Overview
Records
New Entry


Create New Landmark
Title
37.4219983
-122.084
GET CURRENT LOCATION
SELECT IMAGE
UPLOAD LANDMARK
Overview
Records
New Entry