**Software Requirements Specification Document For PumpTrack**

**Festus Helao Shatipamba**

**218207431**

Table of Contents

[**Introduction** 3](#_Toc174431538)

[**Overall Description** 3](#_Toc174431539)

[1. User Authentication 3](#_Toc174431540)

[2. Home Page 4](#_Toc174431541)

[3. Pump List Menu 4](#_Toc174431542)

[4. Location History 4](#_Toc174431543)

[5. Geofencing and Alerts 4](#_Toc174431544)

[6. User Profile 4](#_Toc174431545)

[7. Notifications 5](#_Toc174431546)

[8. Additional Functionalities 5](#_Toc174431547)

[**Specific Requirements** 5](#_Toc174431548)

[1. Functional Requirements 5](#_Toc174431549)

[1.1 User Authentication 5](#_Toc174431550)

[1.2 Home Page and Map View 6](#_Toc174431551)

[1.3 Pump Management 6](#_Toc174431552)

[1.4 Location History 6](#_Toc174431553)

[1.5 Geofencing and Alerts 6](#_Toc174431554)

[1.6 Notifications 6](#_Toc174431555)

[1.7 User Profile Management 7](#_Toc174431556)

[1.8 Reporting and Analytics 7](#_Toc174431557)

[1.9 Search Functionality 7](#_Toc174431558)

[1.10 Offline Functionality 7](#_Toc174431559)

[2. Non-Functional Requirements 7](#_Toc174431560)

[2.1 Performance Requirements 7](#_Toc174431561)

[2.2 Design Constraints 7](#_Toc174431562)

[2.3 Software System Attributes 8](#_Toc174431563)

[2.3.1 Reliability 8](#_Toc174431564)

[2.3.3 Security 8](#_Toc174431565)

[2.3.4 Maintainability 8](#_Toc174431566)

[2.3.5 Portability 9](#_Toc174431567)

[2.4.1 GPS Tracking Device Interface 9](#_Toc174431568)

[2.4.2 Map API Interface 9](#_Toc174431569)

[2.4.3 Firebase Interface 9](#_Toc174431570)

# **Introduction**

This Requirements Specification Document outlines the functional and non-functional requirements for the development of a mobile application intended to track borehole pumps in real-time using GPS technology. The mobile application will be developed using Flutter and Dart, ensuring a cross-platform solution that is both efficient and user-friendly. The backend infrastructure will be supported by Firebase, providing real-time database capabilities and seamless cloud integration.

Namibia, as one of the driest countries in Africa, faces significant challenges in water management, with groundwater being a critical resource. Borehole pumps play a vital role in supplying water to isolated communities, particularly during periods of drought when surface water sources are unreliable. However, the increasing theft of these pumps has posed a severe threat to water security, leaving communities without access to water and imposing financial burdens due to the high costs of replacing stolen equipment.

This mobile application aims to address this issue by enabling the real-time tracking of borehole pumps, thereby aiding in their timely recovery and reducing theft-related disruptions. This document will provide a detailed overview of the requirements for the mobile application, including user requirements, system architecture, and the specific functionalities that the application will need to fulfill to meet the objectives outlined in the research study. The aim is to ensure that all stakeholders have a clear understanding of what the application will achieve and how it will be developed to address the problem of borehole pump theft in Namibia.

# **Overall Description**

The mobile application for tracking borehole pumps is designed to mitigate the challenges posed by pump theft in Namibia by utilizing real-time GPS tracking technology. The application will be developed using Flutter and Dart, ensuring cross-platform compatibility, and Firebase will be used for real-time data management and cloud integration. The application will offer a range of functionalities that are crucial for monitoring and securing borehole pumps. The following sections outline the key features and functionalities of the application.

## 1. User Authentication

Registration Page: New users can create an account by providing necessary details such as email, password, and other relevant information.

Login Page: Existing users can log in using their credentials (email and password).

Password Recovery: Users who have forgotten their passwords can reset them via an email recovery process.

## 2. Home Page

Interactive Map View: After logging in, users are directed to the home page, where they can view a dynamic map displaying all the borehole pumps being monitored. The pumps are represented by markers on the map, with each marker displaying the pump’s current status (e.g., secure, in motion, alert).

Real-Time GPS Tracking: The application receives and displays real-time GPS data from tracking devices installed on the borehole pumps. Users can see the live location of each pump on the map.

## 3. Pump List Menu

View All Monitored Pumps: Users can access a list of all borehole pumps currently being monitored. Each entry in the list will display essential details about the pump, such as its name, current status, last known location, and installation date.

Pump Details: By selecting a specific pump from the list, users can view detailed information about that pump, including its current and historical locations, status, and other relevant data.

Remove Pump: Users have the option to remove a pump from the monitoring system if it is no longer in use or needs to be deregistered.

## 4. Location History

View Historical Data: Users can view the location history of a selected pump, showing its past movements over a specified period.

## 5. Geofencing and Alerts

Geofence Setup: A geofence can be established around each pump’s location. This is a virtual boundary that triggers alerts if the pump is moved outside the defined area.

Alerts Page: Users can access a dedicated alerts page to view a list of all notifications, including geofence breaches, unauthorized movements, and other significant events.

Real-Time Alert Notifications: When a pump crosses the geofence, the application will immediately generate an alert and notify the user via in-app notifications and push notifications.

Real-Time Tracking on Alert: If a geofence breach occurs, the application automatically switches to a real-time tracking mode on the map view, allowing users to monitor the pump’s movement live.

## 6. User Profile

View and Edit Profile Information: Users can access and update their profile information, including their name, email, and other personal details.

Change Password: Users have the option to change their password from the profile section.

Logout: Users can securely log out of the application from the profile section.

## 7. Notifications

In-App Notifications: Important updates, alerts, and other notifications are displayed within the app.

Push Notifications: Critical alerts, such as geofence breaches, are also sent as push notifications to the user’s device to ensure immediate attention.

## 8. Additional Functionalities

Pump Status Indicators: Visual indicators (such as colored markers) on the map show the status of each pump, including whether it is secure, in motion, or has triggered an alert.

Search Functionality: A search bar in the Pump List Menu allows users to quickly locate specific pumps by name or ID.

Reporting and Analytics: Users can generate reports on pump activity, including movement patterns, alert history, and other analytics that can help in decision-making and security planning.

# **Specific Requirements**

This section outlines the detailed requirements for the borehole pump tracking mobile application, covering functional and non-functional aspects, performance requirements, design constraints, software system attributes, and external interfaces.

## 1. Functional Requirements

### 1.1 User Authentication

Registration: The application shall allow new users to register by providing their full name, phone number, email address, and a password. The system shall validate the email format, ensure the phone number is in the correct format, and enforce password security requirements (e.g., minimum length, complexity). Users also have the option to sign up using third-party authentication providers such as Google.

Login: The application shall allow registered users to log in using their email address and password. Users also have the option to sign in using third-party authentication providers such as Google.

Password Recovery: The system shall provide a password recovery option that sends a reset link to the user's registered email address.

### 1.2 Home Page and Map View

Interactive Map: The application shall display a map on the home page with markers representing the real-time location of each borehole pump being tracked.

Real-Time GPS Tracking: The application shall receive and update the GPS location of each pump in real-time on the map.

### 1.3 Pump Management

View Pump List: The application shall allow users to view a list of all monitored borehole pumps. Each entry shall include the pump’s name, current status, last known location, and other relevant details.

View Pump Details: The application shall allow users to select a pump from the list to view detailed information, including its location history and current status.

Remove Pump: The application shall allow users to remove a pump from the monitoring system. This action shall require user confirmation to prevent accidental removal.

### 1.4 Location History

View Historical Data: The application shall allow users to view the historical location data of a selected pump over a specified period.

### 1.5 Geofencing and Alerts

Geofence Configuration: The application shall allow users to set up a geofence around each pump’s location. Users shall be able to define the radius of the geofence.

Alert Generation: The application shall generate an alert if a pump moves outside its configured geofence. Alerts shall be displayed within the app and sent as push notifications.

Real-Time Tracking on Alert: Upon triggering a geofence breach, the application shall enable real-time tracking mode for the affected pump on the map.

### 1.6 Notifications

In-App Notifications: The application shall display notifications for events such as geofence breaches, new pump additions, or system updates.

Push Notifications: The application shall send critical alerts, like geofence breaches, as push notifications to the user’s device.

### 1.7 User Profile Management

View/Edit Profile: The application shall allow users to view and edit their profile information, including their name, email, and password.

Change Password: The application shall provide an option for users to change their password.

Logout: The application shall allow users to securely log out of their account.

### 1.8 Reporting and Analytics

Generate Reports: The application shall allow users to generate reports on pump activity, including location history, alerts, and usage patterns

1.9 Search Functionality

Search Pumps: The application shall include a search bar that allows users to search for specific pumps by name, ID, or other identifiers.

## 2. Non-Functional Requirements

### 2.1 Performance Requirements

Response Time: The application shall update the real-time GPS location of pumps on the map within 3 seconds of receiving new data.

### 2.2 Design Constraints

Platform: The application shall be developed using Flutter and Dart, targeting both iOS and Android platforms.

Backend: The application shall use Firebase for real-time database services, authentication, and cloud storage.

GPS Tracking Device Compatibility: The application shall be compatible with the GPS tracking devices used on the borehole pumps, requiring that they conform to a specified protocol for sending location data.

Map Integration: The application shall use Google Maps API for displaying maps and tracking pump locations.

### 2.3 Software System Attributes

### 2.3.1 Reliability

System Uptime: The application shall be available 99.9% of the time, excluding scheduled maintenance.

Error Handling: The application shall gracefully handle errors, providing informative messages to the user and logging errors for further analysis.

2.3.2 Availability

Availability: The application shall be available for use 24 hours a day, 7 days a week.

### 2.3.3 Security

Data Encryption: All data transmitted between the client and the server shall be encrypted using TLS.

Authentication: The application shall enforce strong password policies and support multi-factor authentication (MFA).

Authorization: The application shall restrict access to certain features (e.g., pump removal) based on user roles.

### 2.3.4 Maintainability

Modular Design: The application shall be designed in a modular fashion to allow for easy updates and maintenance.

Documentation: The application shall include comprehensive documentation for both users and developers.

### 2.3.5 Portability

Cross-Platform Support: The application shall be compatible with both Android and iOS devices.

Data Portability: Users shall be able to export pump data for use in other systems or applications.

2.4 External Interfaces

### 2.4.1 GPS Tracking Device Interface

Data Protocol: The application shall interface with GPS tracking devices using a predefined data protocol that sends location updates (latitude, longitude, timestamp) to the Firebase backend.

### 2.4.2 Map API Interface

Google Maps API: The application shall integrate with Google Maps API for displaying pump locations and geofencing boundaries on the map.

### 2.4.3 Firebase Interface

Authentication: The application shall use Firebase Authentication for managing user login, registration, and password recovery.

Database: The application shall use Firebase Real-time Database for storing and retrieving pump data, including location history and user profiles.

Cloud Messaging: The application shall use Firebase Cloud Messaging (FCM) to send push notifications for alerts and other important updates.