SYSTEM DESIGN DOCUMENT

RescueRadar

Abstract

This document shows the different architectures of the RescueRadar system.

Catalyst Crew

Axolemaranjana4@gmail.com

Table of Contents

INTRODUCTION:

The South African mining industry has a long and complex history dating back to the late 19th century. It is known for its rich mineral resources, including gold, platinum, diamonds, and coal, making it a significant contributor to the country's economy. However, the industry has also faced various challenges, including safety concerns, labour disputes, and environmental issues.

One of the most pressing safety concerns in the South African mining industry is the risk faced by underground miners. These miners work in extreme conditions deep underground, where they extract valuable minerals. While advancements in technology and safety regulations have improved conditions over the years, accidents and incidents still occur leading to injuries, fatalities and in some cases, miners being trapped underground.

Incidents involving trapped or injured miners have drawn significant attention to the industry's safety issues. These incidents often result from factors such as rockfalls, equipment failures, gas leaks and human error.

Incidents that have occurred in South African mines:

“Lily Mine Tragedy (2016)”, One of the most well-known incidents in recent years involved the Lily Mine in Mpumalanga. A collapse at the mine trapped three miners underground. Despite extensive rescue efforts, the miners were not recovered, and the mine remained closed for an extended period.

“Eland Platinum Mine Incident (2014)”, An underground fire at the Eland Platinum Mine in Northwest province led to the evacuation of miners. Thankfully, no fatalities were reported, but the incident highlighted the ongoing safety risks in the industry.

“Kusasalethu Mine Incident (2017)”, A seismic event at the Kusasalethu gold mine in Gauteng led to more than 3,000 miners being trapped underground. Fortunately, all miners were safely rescued after several hours.

These incidents have raised discussions about the safety of South African miners and have led to calls for stricter regulations, improved safety measures and better training for workers in the industry. News articles and reports on these events and others like them have raised awareness about the challenges faced by underground miners in South Africa and the need for ongoing efforts to ensure their safety and well-being in this vital industry.

Purpose

The purpose of RescueRadar is to enhance the emergency response time in the mining operations by implementing a comprehensive tracking system.

Scope

Tracking Device Integration:

- The web application will integrate with tracking devices attached to miners' helmets.
- These devices will utilize RF technology to transmit data, including live location information to the nearest access points within the mine.

Real-Time Location Tracking:

- The application will display the live locations of miners on a mine map within the dashboard.
- Miners will be assigned unique nodes for identification and their positions will be continuously updated on the map using longitude and latitude coordinates.

Access Point Mapping:

- The map will also display the locations of numerous access points within the mine allowing supervisors to monitor the distribution of miners across the site.
- The app will provide information on the number of miners assigned to each access point facilitating efficient resource allocation.

User Roles and Privileges:

- The web application will have multiple user roles, including administrators, supervisors, and viewers.
- Administrators will have full control over system settings, user management, and access permissions.
- Supervisors will be able to monitor miners and access relevant information within their designated shafts.
- Viewers may have limited access or view-only privileges based on their role and permissions.

Emergency Response:

- A couple of miners will be assigned helmets that will be equipped with a panic/emergency button.
- When a miner presses this button, an alert will be sent to the Desktop interface, indicating the miner making the emergency call and the designated shaft and the nearest access point where the emergency is.
- Supervisors and administrators will receive immediate notifications, allowing for swift response and efficient rescue efforts.

Access Management:

- The application will provide functionality for granting, revoking, or blocking user privileges based on administrative decisions. This ensures that only authorized personnel can access the system and view miner location data.

Data Security and Privacy:

- The system will implement robust security measures to protect sensitive miner data and maintain privacy.
- Data encryption and access control will be implemented to prevent unauthorized access.

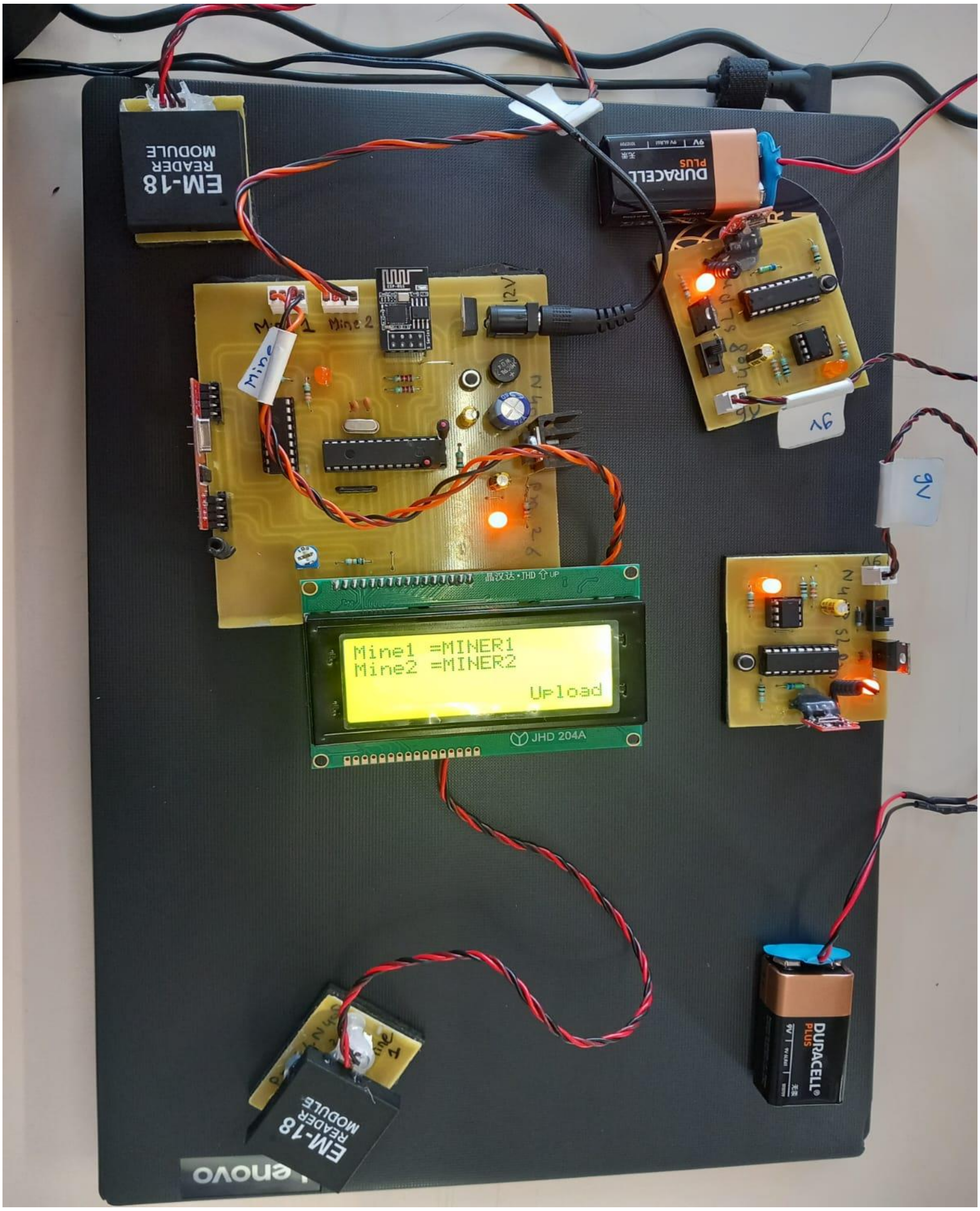
Out of scope:

Utilizing the system to track other things like machinery or trucks. Incorporating other features outside tracking i.e., having the system be able to read/check miner's temperature.

Project Executive Summary

See https://studentujac-my.sharepoint.com/:w:/g/personal/221082098_student_uj_ac_za/EVTFXkAhPxFMk8RC7d7V70YBgZ0dls6IujcHxRXSa0qivA?e=eXi8pk

System Hardware Architecture



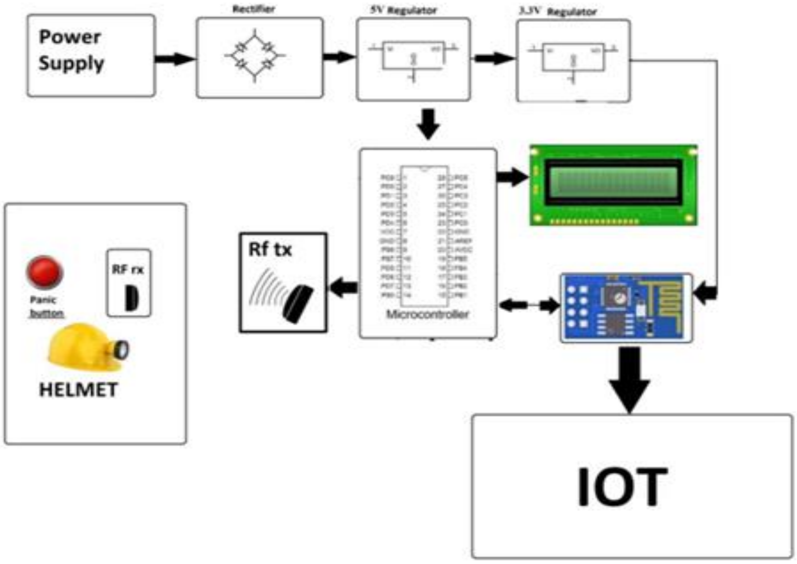
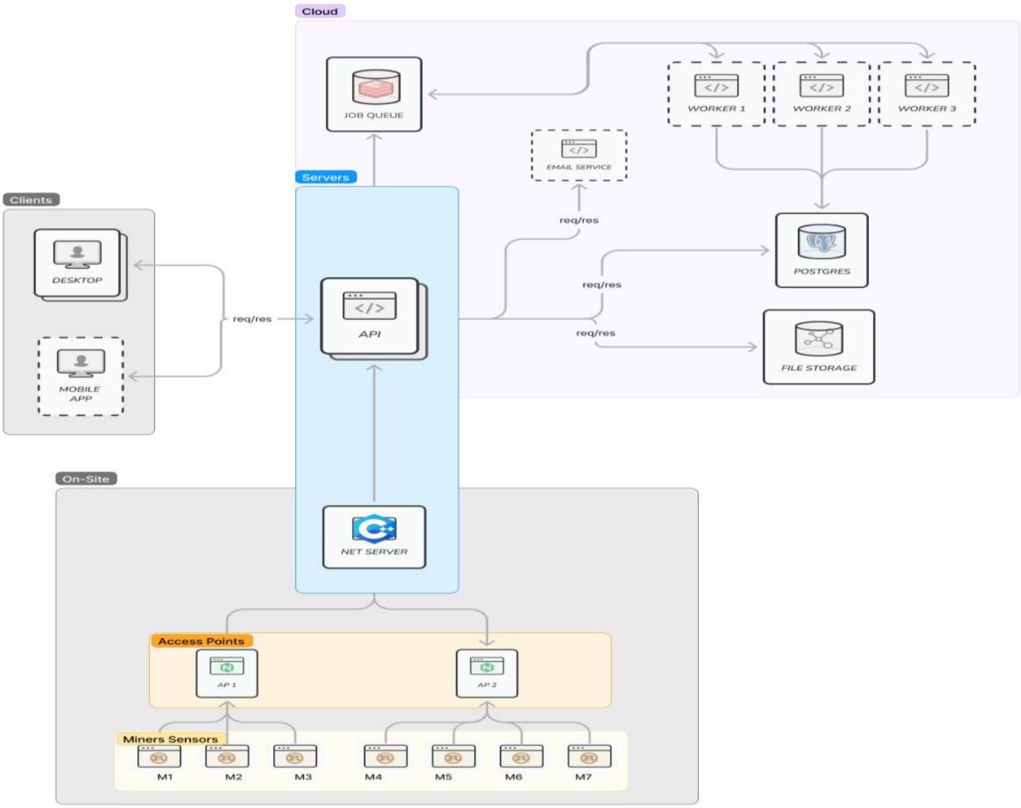
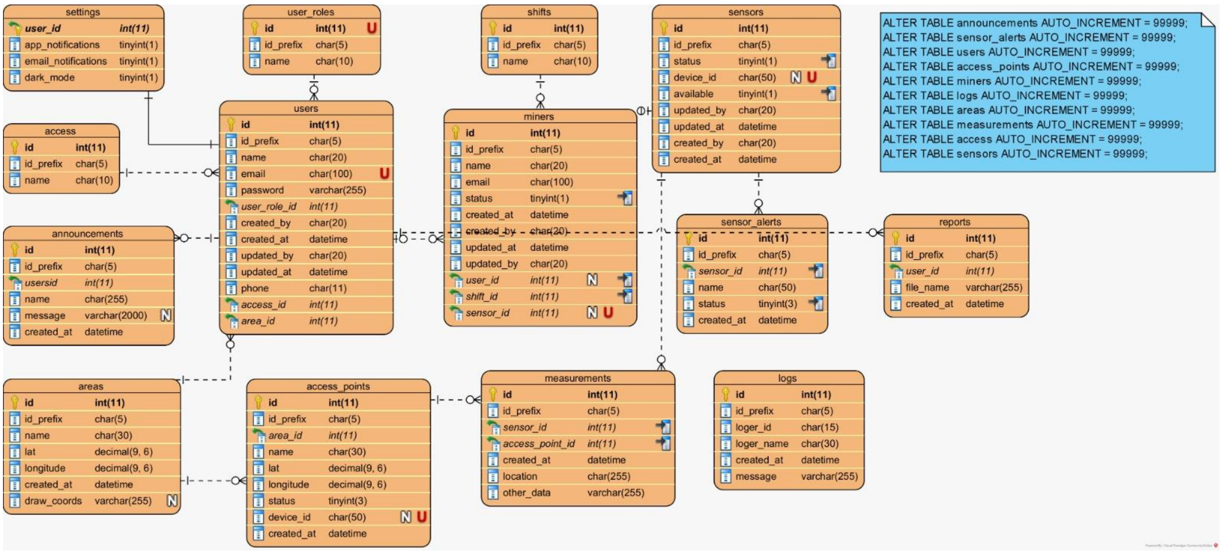


Figure 1: The figure above shows a prototype of the device which will be used that is connected to a sensor and has a Wi-Fi module, Arduino, an LCD screen, RFID readers and tags. With regards to our project when a panic button is pressed, location is tracked and will be displayed within our desktop application on the dashboard which contains a map.

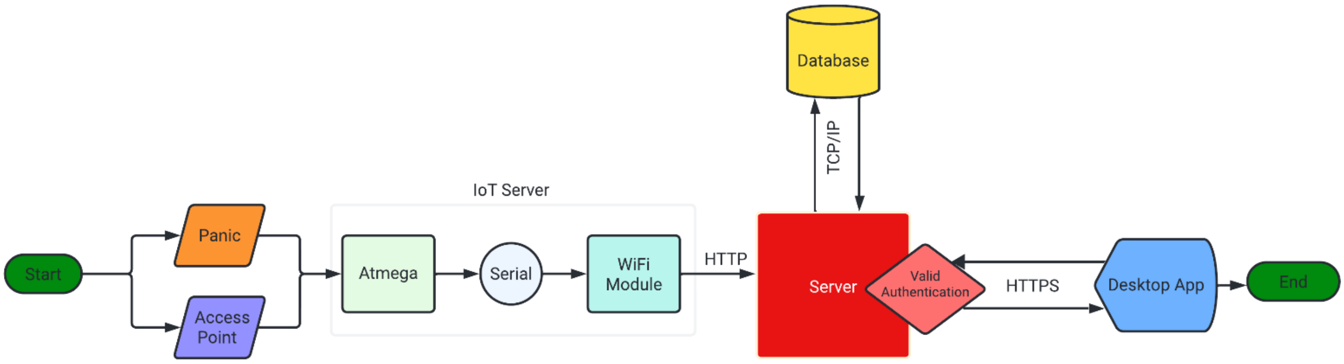
System Software Architecture



Database Architecture



Data Flow Diagram



System Integrity Tools

1. Input Controls: These controls ensure the authenticity, accuracy, completeness, and timeliness of data entered into an application. They can be implemented within application programs or the database schema. Examples include value limit controls, completeness controls, data validation controls, and field combination controls.
2. Access Controls: Registered users-admins- can, supervisors can, and viewers are allowed to view the information only.
3. Transaction Logging: Transaction logs record actions performed by a database and store the results in a separate file. They help discourage fraudulent transactions or malicious database changes and provide a recovery mechanism for erroneous transactions.
4. Complex Update Controls:
    a. Updates made to the database are in transactions.
5. Output Controls: When a user downloads reports from the reports log, the file format of the information downloaded is CSV files.

Development Tools and Guides

Design:

1. Clean and Simple Layout: The UI should have a clean and simple layout, with easy-to-use buttons, icons, and menus. Users should be able to quickly navigate to the features they need.
2. Real-Time Tracking: The UI should display real-time tracking of the miners' locations, indicating their movements and any potential hazards in the mine.
3. Customizable Dashboard: The UI should provide a customizable dashboard that allows users to configure the display of information according to their needs. This can include graphs, charts, and maps that provide a visual representation of data.
4. Color-Coded Alerts: The UI should use color-coded alerts to indicate any potential hazards or emergencies, with different colours indicating different levels of urgency.
5. Mobile Compatibility: The UI should be mobile-compatible, allowing users to access the software from their mobile devices. This can be achieved by creating a responsive design that adapts to different screen sizes.
6. Integration with Existing Systems: The UI should be designed to integrate seamlessly with existing mining equipment and software systems, providing a single interface for monitoring all aspects of the mine's operations.
7. User-Friendly: The UI should be user-friendly and intuitive, with clear and concise instructions for all functions. Users should be able to quickly learn how to use the software without requiring extensive training.

Databases:

MySQL because of: Scalability, security, Open-Source, reliability.

1. Users Table: This table would store information about the users of the system, including their name, role, and contact information.
2. Mines Table: This table would store information about the mines being monitored by the system, including the name, location, and other details about the mine.
3. Areas Table: This table would store information about the different areas within the mine, including the name, location, and other details about each area.
4. Sensors Table: This table would store information about the sensors used to track the movement of miners, including the sensor ID, location, and other details.
5. Measurements Table: This table would store the measurements collected by the sensors, including the date and time of the measurement, the location of the sensor, and the value of the measurement.
6. Alerts Table: This table would store information about any alerts generated by the system, including the date and time of the alert, the location of the sensor that triggered the alert, and the type of alert generated.
7. Reports Table: This table would store information about the reports generated by the system, including the date and time of the report, the mine or area being monitored, and other details.

Redis

1. Session Store: Redis can store session data for web applications. Since it's in-memory, it's much faster than traditional databases for this purpose. It's also easy to scale and manage.
2. Task Queue: Redis can be used as a task queue where you can push tasks into a Redis list and have worker processes or threads consume these tasks and process them asynchronously. This is useful for background processing of tasks in web applications.

System Architecture:

Distributed architecture to ensure scalability and reliability.

1. Front-end Interface: This component would be responsible for providing the user interface that mining companies would use to monitor the movement of miners in real-time. The front-end could be developed using web technologies or a desktop application.
2. Application Server: This component would be responsible for processing requests from the front-end interface and communicating with the back-end services.
3. Database: As mentioned earlier, the database would store information related to the movement of miners in real-time underground.
4. Sensor Data Ingestion Service: This service would be responsible for ingesting data from the sensors used to track the movement of miners in real-time. The data would be stored in the database.
5. Real-time Processing Service: This service would be responsible for processing the sensor data in real-time and generating alerts if any safety risks are detected.
6. Reporting Service: This service would be responsible for generating reports based on the data collected by the system. The reports could be used by mining companies to identify potential safety risks and improve the efficiency of rescue missions.
7. Integration Services: The mining tracker software could be designed to integrate with existing mining equipment and software systems to provide a more comprehensive view of mining operations.

Font-end:

1. Electron
2. Vite
3. Primereact.org for styling.

Back-end:

4. Nodejs
5. Express-JS
6. Pm2 process runner
7. Hosted on GCP (VM)

Appendices

1. https://app.getpostman.com/join-team?invite_code=760796689e9ca03e284a88acf3fdf9fa&target_code=5eb0b09456ab72d95559ad95a0364c7d