Asset Tracker

Project Overview

Asset Tracker is a comprehensive asset management solution designed to help organizations efficiently track, register, and manage their assets. The application features a superuser dashboard, user dashboard with asset management, asset history, reports, and profile options. It ensures secure access control, allowing only the superuser to edit and delete assets while enabling all users to view and download reports.

Team

- **Developers**: Unathi Nkone, Jason Cameron, Sawongwa Thambo, Ontiretse Mosupye
- **Backend**: MongoDB, Node.js
- **Frontend**: React.js
- **Collaboration**: GitHub (repository created by Junior)

Features

1. **Superuser Dashboard**:
    - Dedicated dashboard for superusers with additional administrative capabilities.
    - Create and manage users and assets.
    - Access Control List (ACL) setup to restrict edit and delete permissions to the superuser.
    - Generate and download reports.
2. **User Dashboard**:
    - View and manage assigned assets.
    - Access asset history and generate reports.
    - Profile management options.
3. **Authentication**:
    - User registration and login functionality.
    - Superuser can create new users.
4. **Reporting**:
    - Generate detailed reports of assets and their history.
    - Download reports for offline access and record-keeping.

Development Plan

Backend Development (Day 1-5)

1. **Day 1**: Setup project repository, database schema design, and basic server setup.
2. **Day 2**: Implement asset registration and user authentication functionality.
3. **Day 3**: Develop asset inventory management and user assignment features.
4. **Day 4**: Build reporting and download functionalities.

5. **Day 5**: Testing and debugging backend functionalities.

Frontend Development (Day 6-10)

1. **Day 6**: Setup frontend framework and design initial UI components.
2. **Day 7**: Develop UI for asset registration and user authentication.
3. **Day 8**: Implement user dashboard for asset management and history viewing.
4. **Day 9**: Create superuser dashboard and integrate reporting functionalities.
5. **Day 10**: Integrate frontend with backend, perform end-to-end testing, and final debugging.

Tools & Technologies

- **Backend**: Node.js, Express, MongoDB
- **Frontend**: React.js
- **Version Control**: Git (GitHub)
- **Project Management**: Trello or Jira

Deliverables

- A fully functional asset management solution with specified features.
- A well-documented codebase.
- A user guide for the asset management system.