1. Introduction

This document provides an overview of the application's deployment architecture, key features, and status. It is intended for review and outlines the technologies, infrastructure, and operational details related to the frontend, backend, and database components of the system.

2. Infrastructure Deployment

- **Terraform-Based Deployment:**
- The entire infrastructure is deployed using Terraform. This includes not only the provisioning of cloud resources but also the inclusion of custom data and scripts necessary for the deployment. All infrastructure provisioning scripts are maintained as part of the Terraform configuration, ensuring reproducibility and version control.

3. Application Architecture

3.1 Frontend

- **Platform:**

The front-end is a Node.js application deployed on the Azure App Service.

- **Features:**
    - User **registration** has been implemented and is confirmed to be working perfectly from the UI.
- **Current Status:**
    - The registration process is fully operational.

3.2 Backend API

- **Platform:**

The backend API is hosted on an Azure Virtual Machine.

- **Configuration:**
    - The API is exposed on port 8080, making it accessible for client interactions.

- **Current Status:**
    - While the API is functional, there is a noted CORS (Cross-Origin Resource Sharing) issue affecting the login process. The issue appears from the default configuration or the use of an unprioritized domain for the login endpoint.

3.3 Database

- **Platform:**

The database is hosted on the Azure MySQL service.

- **Security:**
    - Access to the database is restricted to the private network, enhancing the security of the data layer.
- **Schema Management:**
    - For any database-related schema changes, the current process involves connecting to the backend server and applying the necessary scripts directly.

4. Known Issues

- **CORS Issue with Login:**
    - **Issue:** The login functionality is facing a CORS issue due to the use of a default or previously unknown domain configuration.
- **Database Schema Changes:**
    - **Process:** Currently, schema updates are applied via the backend server connection.

5. Conclusion

The application is built on a robust infrastructure leveraging Terraform for consistent deployment, with clearly defined roles for the frontend, backend, and database layers. While the registration feature is fully operational, attention is required to resolve the CORS issue affecting login functionality. Additionally, streamlining the process for applying database schema changes would further enhance operational efficiency.