Hands-on Project: Deploying a Node.js Application with MySQL using systemd

Scenario

You're a DevOps engineer tasked with deploying a Node.js application that interfaces with a MySQL database on a Linux server. Your responsibility includes setting up the application to automatically start at boot time using systemd and ensuring it restarts if it crashes.

Requirements

1. Create a simple Node.js API application that:
    - Connects to a MySQL database
    - Has a `/health` endpoint that returns the database connection status
    - Has a `/users` endpoint that retrieves and returns users from the database
2. Configure a MySQL database:
    - Create a database named `practice_app`
    - Create a `users` table with columns for `id`, `name`, and `email`
    - Insert at least 3 sample users
3. Set up the Node.js application as a systemd service that:
    - Starts automatically on system boot
    - Restarts automatically if the application crashes
    - Runs under a non-root user account
    - Logs output to the systemd journal

Tasks

Part 1: Application Setup

1. Create a Node.js project directory and initialize it with npm
2. Install necessary dependencies (Express and MySQL)
3. Create a Node.js application that connects to MySQL
4. Implement the required API endpoints (health and users)

Part 2: Database Setup

1. Install MySQL if not already installed
2. Secure the MySQL installation
3. Create the required database, user, and table
4. Add sample data to the table

Part 3: systemd Configuration

1. Create a dedicated system user for running the application
2. Place your application in an appropriate directory with proper permissions
3. Create a systemd service file for your application
4. Configure appropriate service options (restart policy, dependencies, security)
5. Enable the service to start at boot time

Part 4: Testing

1. Start your service and verify it's running
2. Test that your application endpoints work correctly
3. Test that your service restarts if the application crashes
4. Reboot your system and verify the service starts automatically

---

Documentation Requirement

All students should make a proper documentation README file or PPT/docx and upload it to GitHub and submit to the Google form below:

https://forms.gle/VrEaUzfq7RRnmKgj8

Good luck with your hands-on Project!