**Production Deployment Guide – Entitlement Review**

*By Divyansh Singh Parihar*

**Prerequisites**

* Ubuntu 22.04 LTS server
* Visual Studio Code
* MongoDB Compass
* Node JS (18 Compatible)
* Mongosh
* MongoDB Community Server

**Install Ubuntu Server 22.04.5 GUI**

1. sudo apt update && sudo apt upgrade -y
2. sudo apt install ubuntu-desktop
3. sudo reboot

**Install Node.js 18**

**Windows**

* Go to:
  + <https://nodejs.org/>
* Download LTS version (e.g., 20.x.x LTS).
* Run the installer:
* Click “Next” through all steps.
* Keep default options selected.
* After installation, open Command Prompt:
  + node -v
  + npm -v

Confirms installation checking the version og the node and npm.

**Linux**

**NodeSource Repository (Stable)**

1. Remove previous installations
   1. sudo apt remove nodejs npm
2. Install curl if not present
   1. sudo apt update
   2. sudo apt install curl -y
3. Fetch setup script for latest LTS (e.g., 20.x)
   1. curl -fsSL https://deb.nodesource.com/setup\_20.x | sudo -E bash -
4. Install Node.js
   1. sudo apt install -y nodejs
5. Verify
   1. node -v
   2. npm -v

Download and install nvm:

1. curl -o- https://raw.githubusercontent.com/nvm-sh/nvm/v0.40.3/install.sh | bash
2. in lieu of restarting the shell
   1. \. "$HOME/.nvm/nvm.sh"
3. Download and install Node.js:
   1. nvm install 22
4. Verify the Node.js version:
   1. node -v # Should print "v22.17.0".
   2. nvm current # Should print "v22.17.0".
5. Verify npm version:
   1. npm -v # Should print "10.9.2".

**Install Visual Studio Code**

**Windows**

**Steps:**

1. Go to: <https://code.visualstudio.com/>
   1. Download User Installer for Windows (64-bit).
2. Run installer:
   1. Accept license.
   2. Keep default location.
   3. Enable options:
   4. "Add to PATH"
   5. "Register Code as an editor"
   6. "Add ‘Open with Code’ to context menu"
   7. Finish setup.
3. Open Command Prompt:
   1. code
4. Confirms successful installation.

**Linux**

1. sudo apt install ./<file>.deb
2. echo "code code/add-microsoft-repo boolean true" | sudo debconf-set-selections
3. sudo apt-get install wget gpg
4. wget -qO- https://packages.microsoft.com/keys/microsoft.asc | gpg --dearmor > packages.microsoft.gpg

sudo install -D -o root -g root -m 644 packages.microsoft.gpg /etc/apt/keyrings/packages.microsoft.gpg

echo "deb [arch=amd64,arm64,armhf signed-by=/etc/apt/keyrings/packages.microsoft.gpg] https://packages.microsoft.com/repos/code stable main" |sudo tee /etc/apt/sources.list.d/vscode.list > /dev/null

rm -f packages.microsoft.gpg

1. sudo apt install apt-transport-https
2. sudo apt update
3. sudo apt install code # or code-insiders

**Install MongoDB Compass**

**Windows**

1. Go to: <https://www.mongodb.com/try/download/compass>
2. Select Windows 🡪 Download .exe (MSI Installer).
3. Run the installer:
4. Click through defaults.
5. Finish installation.
6. Launch MongoDB Compass.
7. Verify version from the GUI.

**Linux**

# Download latest .deb package

1. wget https://downloads.mongodb.com/compass/mongodb-compass\_1.45.0\_amd64.deb

# Install

1. sudo apt install ./mongodb-compass\_1.45.0\_amd64.deb

# Launch

1. mongodb-compass &

**Install Mongosh**

**Windows**

1. **Go to:** [**https://www.mongodb.com/try/download/shell**](https://www.mongodb.com/try/download/shell)
2. **Select:**
   1. **OS: Windows**
   2. **Package: MSI**
3. **Download and install .msi file.**
4. **During installation, allow "Add to PATH".**
5. **Open Command Prompt and run:**
   1. **Mongosh**
6. **Confirms shell is working.**

**Linux**

**Install MongoDB Community Server**

**Windows**

1. Go to: <https://www.mongodb.com/try/download/community>
2. Select:
   1. OS: Windows
   2. Package: MSI
   3. Version: 6.0.x (or latest stable)
3. Download .msi installer.
   1. Run installer:
   2. Choose Complete setup.
   3. Enable Install MongoDB as a Service.
   4. Uncheck MongoDB Compass if already installed.
   5. Finish installation.
4. Open Command Prompt and run:
   1. mongod –version
   2. Confirms server is installed.
5. Verify Service Running:
   1. services.msc

Find MongoDB in the list 🡪 Should be running automatically.

**Git Clone The Project**

**Step 1: Install Prerequisites (if not already installed)**

**Windows**

Install Git: <https://git-scm.com/download/win>

**Step 2: Clone the Repository**

git clone <https://github.com/SomethingForWork/Final_Product.git>

**Step 3: Install Dependencies for Server**

cd Final\_Product/server

npm install

**Step 4: Install Dependencies for frontend**

cd Final\_Product/frontend

npm install

**Step 5: Start The Project**

cd Final\_Product/server

npm start

cd Final\_Product/frontend

npm start

**Or Don’t Start As We Are Deploying But Open It In Visual Studio Code.**

**MongoDB Authentication Must Be Enabled**

By default, MongoDB on Linux may not have authentication enabled.

To enable it:

1. Open the MongoDB config file (usually /etc/mongod.conf):
   1. sudo nano /etc/mongod.conf
2. Under the security section, add:
   1. security:

authorization: enabled // only 2 spaces before authorization

* 1. Make sure to add 2 spaces before authorization otherwise it will through error

1. Restart MongoDB:
   1. sudo systemctl restart mongod

**Create Users with Roles**

You must create users with the correct roles in the database you want to use.

For your app, you already have:

use restrict\_app db.createUser({ user: "restrict\_user", pwd: "random", roles: [{ role: "readWrite", db: "restrict\_app" }] })

If you want more roles (e.g., admin, read-only), create them similarly:

db.createUser({

user: "admin\_user",

pwd: "adminpassword",

roles: [{ role: "dbAdmin", db: "restrict\_app" }]

})

Each user can have different roles.

**Connection String**

Your connection string should specify the user, password, and database:

mongodb://restrict\_user:random@localhost:27017/restrict\_app

If you want to connect as a different role, change the user and password accordingly.

**IF The Error Comes OF ECONNREFUSED Comes When Starting MongoDB Connection**

**Check status:**

sudo systemctl status mongod

**Start if inactive:**

sudo systemctl start mongod

**Enable on boot:**

sudo systemctl enable mongod

**Wrong hostname or port**

Default MongoDB port is `27017`.

Make sure Compass is connecting to:

mongodb://localhost:27017

Change The IP In REACT\_APP\_API\_URL = To System IP in the `server/index.js` file

Also Change It In CORS Setup [Lines 65 Usually]

const app = express()

app.use(cors({

origin: 'http://10.0.2.15:3000', // Change The IP Here

credentials: true

}));

First We Need To Create A User Of Admin Privileges To Login To The App With The Admin Privileges

Paste This In Your VS Code Terminal

curl -X POST http://172.25.137.59:3002/create-admin -H "Content-Type: application/json" -d "{\"name\": \"Admin Name\", \"email\": \"admin@example.com\", \"password\": \"yourpassword\"}"

Change http: 172.25.137.59:3002 To Your Server IP

**TOWARDS THE PRODUCTION**

**Install pm2**

**sudo npm install -g pm2**

**pm2 start index.js --name your-backend**

**pm2 startup**

**pm2 save**