**Step 1: Set Up Node.js Environment on the Server**

1. **Update and Upgrade Packages**:

sudo apt update

sudo apt upgrade -y

1. **Install Node.js and npm**:

curl -sL https://deb.nodesource.com/setup\_16.x | sudo -E bash -

sudo apt install -y nodejs

1. **Verify Installation**:

node -v

npm -v

**Step 2: Install and Configure a NestJS Application**

1. **Install NestJS CLI**:

npm install -g @nestjs/cli

1. **Create a New NestJS Application**:

nest new my-nest-app

cd my-nest-app

1. **Install PostgreSQL Client Library**:

npm install @nestjs/typeorm typeorm pg

1. **Create the** .env **File** in the root directory:

DB\_HOST=localhost

DB\_PORT=5432

DB\_USERNAME=db\_user

DB\_PASSWORD=password

DB\_NAME=test\_db

1. **Install** dotenv:

npm install dotenv

1. **Configure TypeORM in** app.module.ts:

import { Module } from '@nestjs/common';

import { TypeOrmModule } from '@nestjs/typeorm';

import { AppController } from './app.controller';

import { AppService } from './app.service';

import \* as dotenv from 'dotenv';

dotenv.config(); // Load environment variables

@Module({

imports: [

TypeOrmModule.forRoot({

type: 'postgres',

host: process.env.DB\_HOST,

port: parseInt(process.env.DB\_PORT, 10),

username: process.env.DB\_USERNAME,

password: process.env.DB\_PASSWORD,

database: process.env.DB\_NAME,

entities: [\_\_dirname + '/\*\*/\*.entity{.ts,.js}'],

synchronize: true,

}),

],

controllers: [AppController],

providers: [AppService],

})

export class AppModule {}