

Backend APIs

A Node.js backend providing user **signup, OTP verification, login, profile retrieval, and profile editing functionalities** with JWT-based authentication and MySQL (via Sequelize ORM).

🚀 Getting Started

✓ Prerequisites

- Node.js \geq v16
- MySQL server
- .env file with required credentials

⬇ Installation

```
npm install
```

▶ Run the Server

Ensure MySQL server is running and .env variables are correctly set up:

```
node app.js
```

🌳 Project Structure

```
.
├── app.js           # Entry point
├── .env             # Environment variables
├── package.json     # Dependencies
├── jwt/
│   └── auth.js      # JWT middleware
├── routes/
│   ├── router.js    # Route definitions
│   └── controller.js # Controller logic
├── sql/
│   ├── db.js        # Sequelize DB connection
│   └── user.js       # User model definition
```

Environment Variables

Create a `.env` file in the root with the following variables:

```
PORT=3000
HASH_KEY=your_jwt_secret_key

DB_NAME=your_database_name
DB_USER=your_mysql_username
DB_PASS=your_mysql_password
DB_HOST=localhost
```

API Endpoints

All responses are in JSON format.

□ Public Endpoints

POST /signup

Register a new user with OTP generation.

Body:

```
{
  "name": "John Doe",
  "email": "john@example.com",
  "password": "password123",
  "phone_number": "9876543210",
  "gender": "male"
}
```

POST /verify

Verify email via OTP.

Body:

```
{
  "email": "john@example.com",
  "otp": "12345"
}
```

POST /login

Login using email and password.

Body:

```
{
  "email": "john@example.com",
  "password": "password123"
}
```

Returns:

```
{
  "token": "jwt_token_here"
}
```

Protected Endpoints (require JWT in header)

All these endpoints require the header:

Authorization: Bearer <jwt_token>

GET /profile

Fetch logged-in user's profile.

Response:

```
{
  "id": 1,
  "name": "John Doe",
  "email": "john@example.com",
  "phone_number": "9876543210",
  "gender": "male"
}
```

POST /edit

Update user profile.

Body:

```
{
  "name": "John Smith",
  "email": "johnsmith@example.com",
  "password": "newpassword",
  "phone_number": "9999999999",
  "gender": "male"
}
```

Tech Stack

- **Backend:** Node.js + Express
- **ORM:** Sequelize
- **Database:** MySQL
- **Auth:** JWT
- **Security:** Bcrypt for password hashing