

User Authentication System – Project Structure

Readable, structured overview of folders, files, and key components

Stack: Node.js · Express · MongoDB · Mongoose · JWT · Swagger/OpenAPI

OVERVIEW

1. Executive Summary

This document summarizes the structure of the **User Authentication System** project in a clean, easy-to-read format. It highlights the main folders, important files, and the technologies used to implement a secure authentication API.

Goal

Provide user registration, login, and protected endpoints using JSON Web Tokens (JWT) and MongoDB.

Core Technologies

Node.js, Express, MongoDB, Mongoose, bcrypt, JWT, Swagger/OpenAPI.

ROOT

2. Project Root Files

The root directory contains configuration, scripts, and metadata that define the project.

- **.env** – Environment variables (database URI, JWT secret, port, etc.).
- **mongoDbOnlineScriptConnection.txt** – Notes on connection string details for MongoDB.
- **package.json** / **package-lock.json** – Project metadata, npm scripts, and locked dependency versions.
- **testMongo.js** – Script to test MongoDB connectivity.
- **User-Authentication-System-structure.txt** – Original text description of the folder structure.

CONFIG

3. Configuration & Migrations

3.1 Configuration (config/)

- **config/db.js** – Central database configuration and connection logic for MongoDB, typically using Mongoose.

3.2 Migrations (migrations/)

- **migrations/runMigrations.js** – Script to run database migrations or initialization tasks (for example, seeding or index creation).

SOURCE

4. Application Source (src/)

The **src** directory contains the main application code, grouped by responsibility.

4.1 Core Files

- **src/server.js** – Entry point for the Express server; sets up middleware, routes, and server startup.
- **src/swagger.js** – Swagger/OpenAPI configuration for generating and serving API documentation.

4.2 Controllers

- **src/controllers/auth.controller.js** – Handles authentication logic such as registration, login, and token handling.

4.3 Middleware

- **src/middleware/auth.middleware.js** – Authentication middleware (for example, JWT verification and access control).

4.4 Models

- **src/models/Users.js** – Mongoose model defining the User schema and related behavior.

4.5 Routes

- **src/routes/auth.routes.js** – Express routes for authentication endpoints (such as `/api/auth/register` and `/api/auth/login`).

4.6 Utilities

- **src/utlis/tokenBlacklist.js** – Utility for managing blacklisted tokens to support logout and token revocation.

DEPENDENCIES

5. Key Dependencies (node_modules/)

The **node_modules** directory contains all npm packages used by the project. The table below lists the most important ones and their roles.

Package	Purpose
express	HTTP server framework providing routing and middleware support.
mongoose	ODM library for MongoDB, used to define schemas and interact with the database.
mongodb	Official MongoDB driver, used directly or via Mongoose.
bcrypt	Secure password hashing for storing and verifying user credentials.
jsonwebtoken	Creation and verification of JSON Web Tokens (JWT) for stateless authentication.
dotenv	Loads environment variables from the .env file into <i>process.env</i> .
swagger-jsdoc	Generates an OpenAPI/Swagger specification from JSDoc comments.
swagger-ui-express	Serves the Swagger UI interface for interactive API documentation.
nodemon	Development tool that restarts the server automatically when files change.

Many additional sub-dependencies (such as *body-parser*, *cookie*, *validator*, and others) support HTTP parsing, validation, and other low-level functionality.

APPENDIX

6. Directory Tree (Reference)

The following appendix shows the project's folder path listing in a console-style format. Deeply nested dependency details are shortened for readability.

```

Folder PATH listing
Volume serial number is 4460-1D1F
C:.\
|   .env
|   mongoDbOnlineScriptConnection.txt
|   package-lock.json
|   package.json
|   testMongo.js
|   User-Authentication-System-structure.txt
|
+---config
|   db.js
|
+---migrations
|   runMigrations.js
|
+---node_modules
|   .package-lock.json
|
|   +---.bin
|   |   js-yaml
|   |   nodemon
|   |   semver
|   |   swagger-jsdoc
|   |   z-schema
|   |   ...
|   +---@apidevtools
|   |   json-schema-ref-parser
|   |   openapi-schemas
|   |   swagger-parser
|   |   ...
|   +---mongodb
|   |   mongodb.d.ts
|   |   README.md
|   |   ...
|   +---mongoose
|   |   index.js
|   |   README.md
|   |   ...
|   +---express
|   |   index.js
|   |   ...
|   +---swagger-jsdoc
|   |   index.js
|   |   ...
|   +---swagger-ui-express
|   |   index.js
|   |   ...
|   ... (additional dependency folders and files omitted for brevity)
|
\---src
|   server.js
|   swagger.js
|
+---controllers
|   auth.controller.js
+---middleware
|   auth.middleware.js
+---models
|   Users.js
+---routes
|   auth.routes.js
\---utils
    tokenBlacklist.js

```

