**Full Stack Technical Assessment**

Report file for findings and outputs for scrawlr full stack assessment.

Backend: Lumen PHP 🡪Directory: ../src/api/

Frontend: Vue.js 🡪Directory: ../src/ui

**Backend (api):**

**Useful Commands:**

php artisan migrate

php -S localhost:8080 -t public/

updated .env for database connection properties

php artisan migrate

php artisan make:model Todonotes --migration

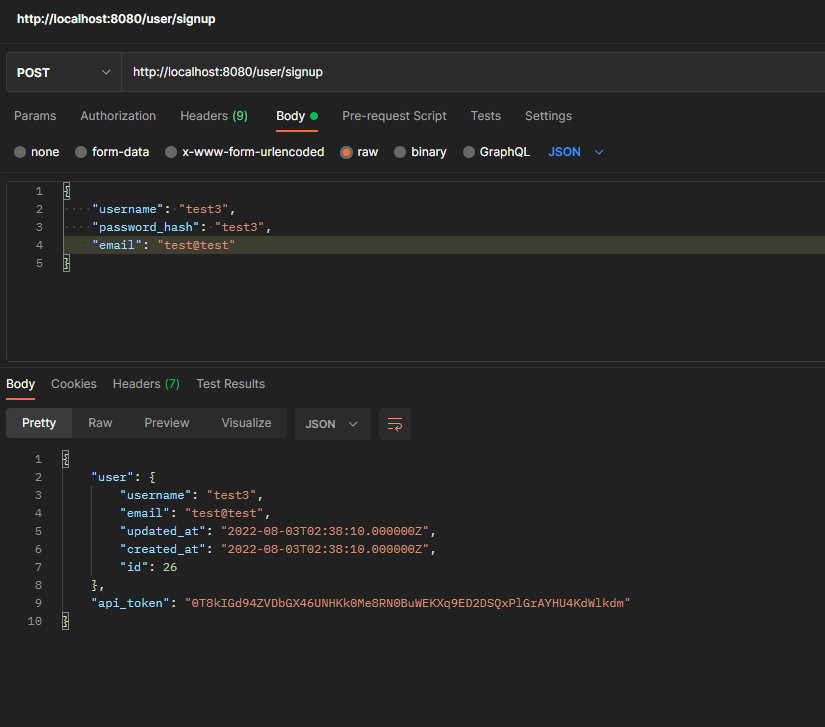
php artisan migrate:fresh

Notes:

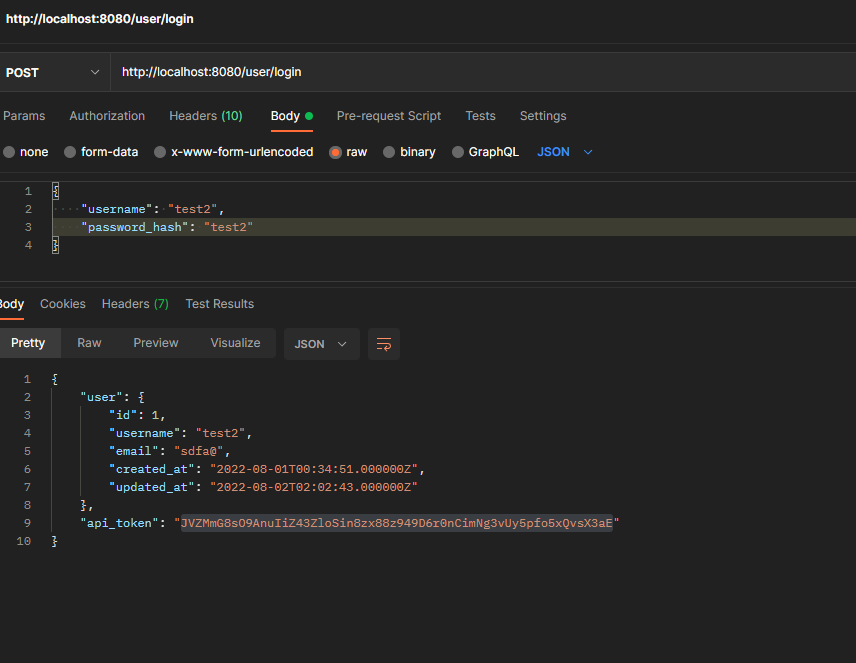
1. Small changes made to Users, TodoNote Migration files to provided migrations
2. Developed model files: User.php and Todonotes.php (Directory: /api/app/models)
3. Developed controllers: TodoNotesController.php and UsersController.php (Directory: /api/app/controllers/..)
4. Developed endpoints in web.php file according to provided comments in that file
5. Made required configuration changes in app.php file and configured Auth middleware for authentication
6. Validations to data, token authentication
7. Postgres database and its required configurations

**Outputs:**

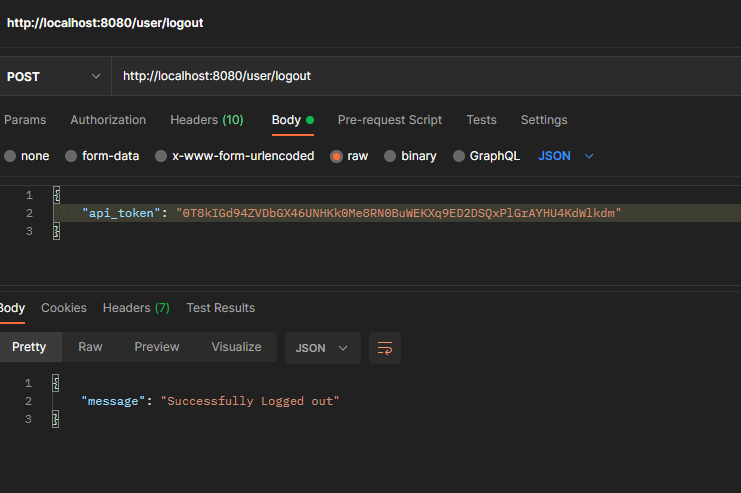
1. <http://localhost:8080/user/signup>



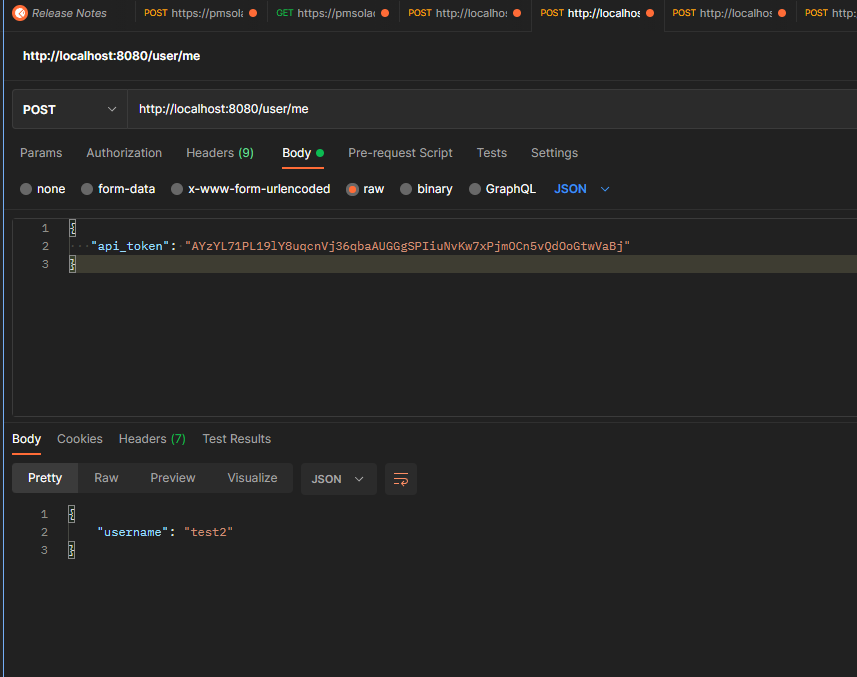
1. <http://localhost:8080/user/login>

. 

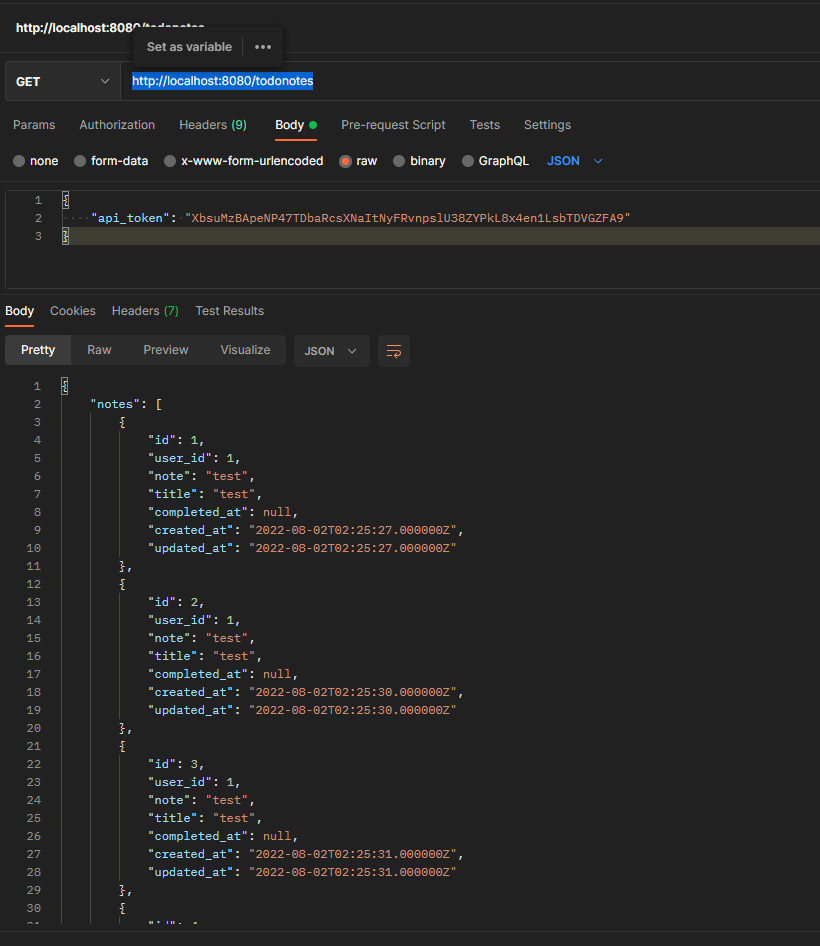
1. <http://localhost:8080/user/logout>



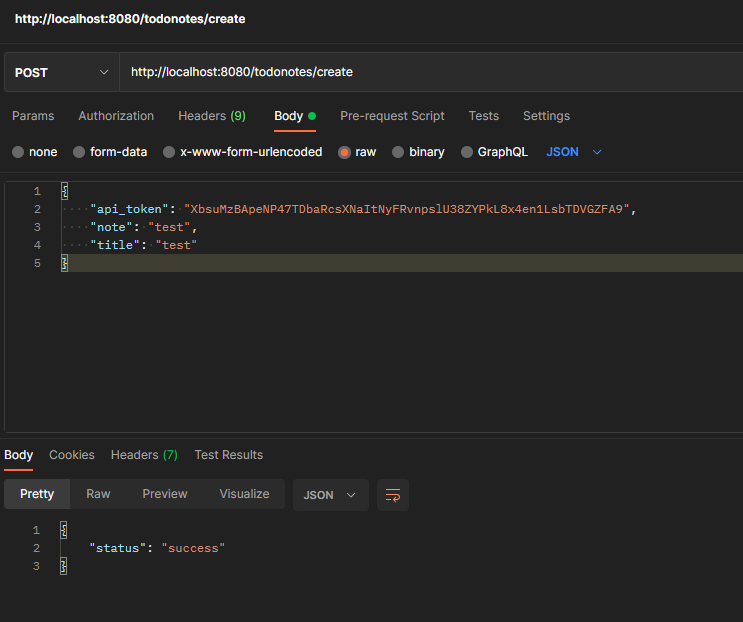
4.. <http://localhost:8080/user/me>



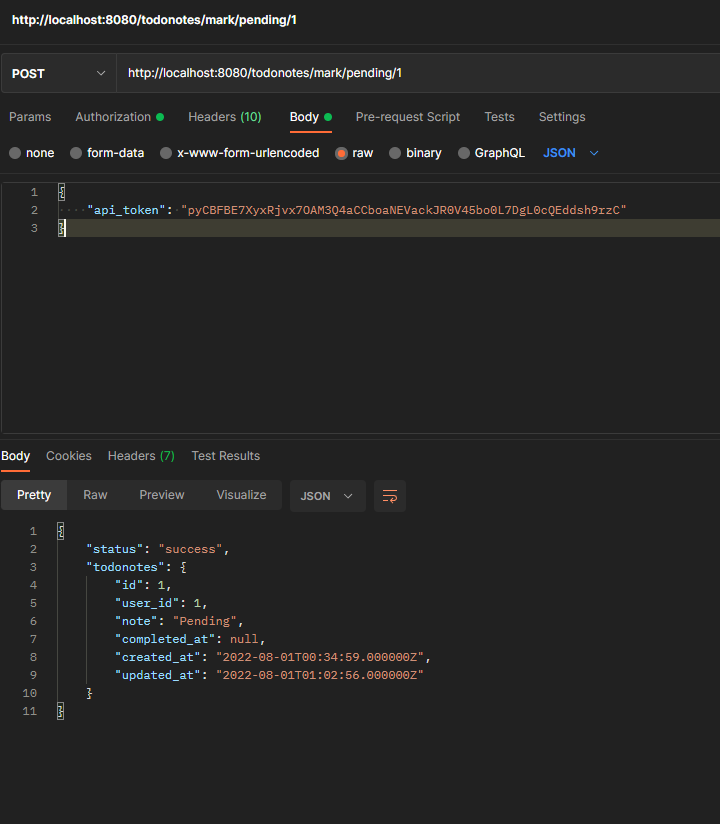
1. <http://localhost:8080/todonotes>



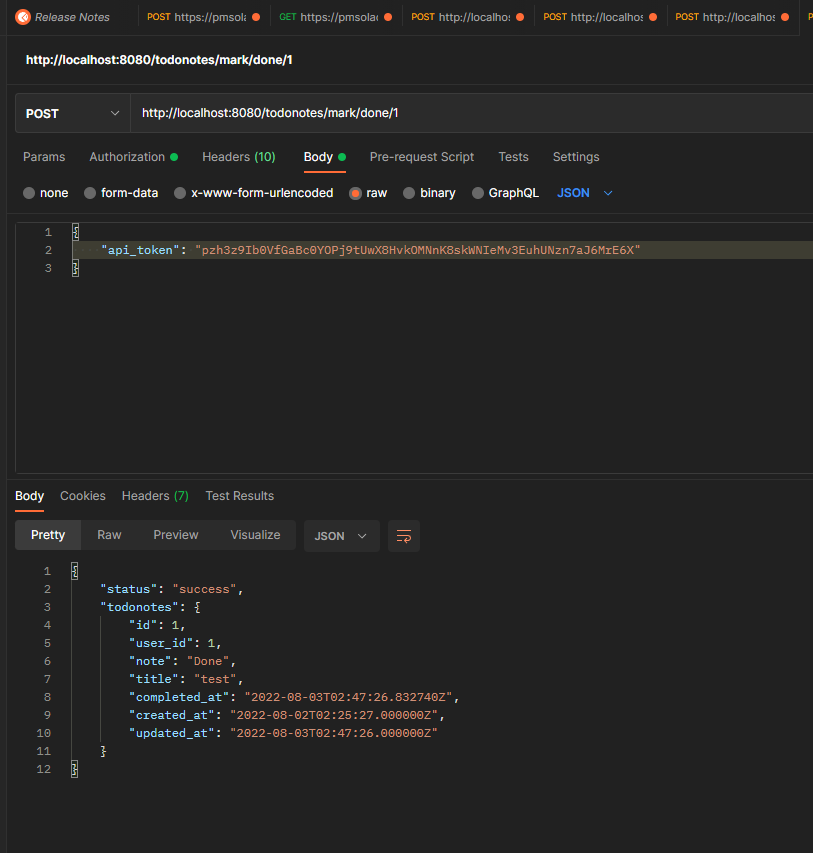
1. <http://localhost:8080/todonotes/create>



7 <http://localhost:8080/todonotes/mark/pending/1>



8 <http://localhost:8080/todonotes/mark/done/1>



**Frontend (api):**

**Useful Commands:**

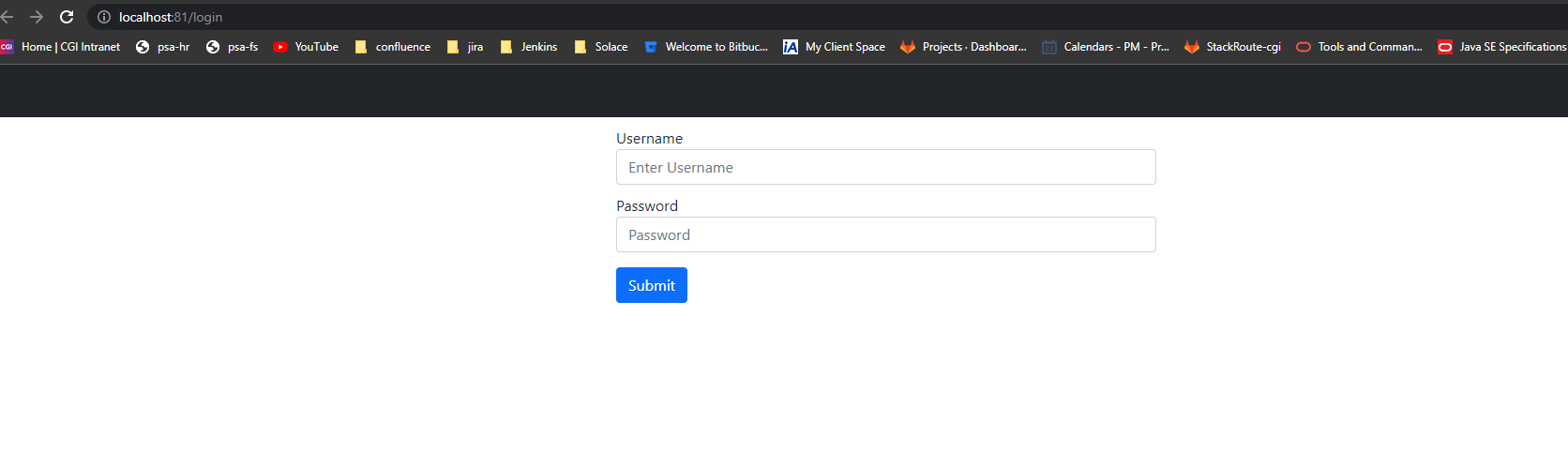
1. npm install
2. npm run dev -> start server

Note:

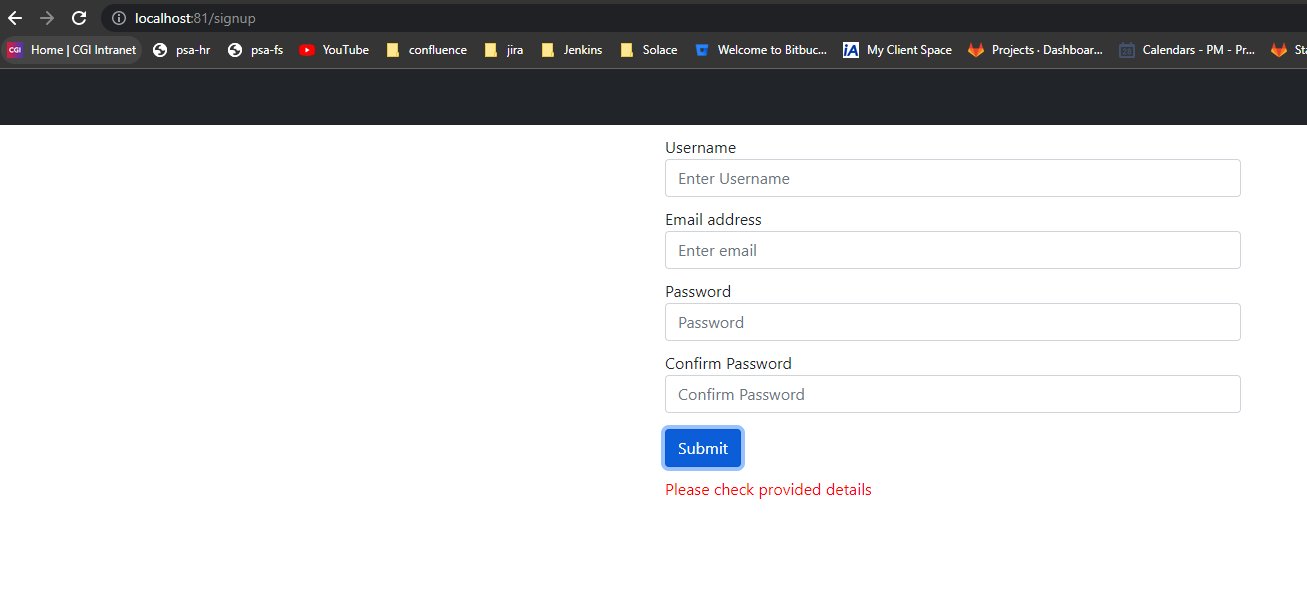
1. Develped UI pages: signup, login, todonotes
2. Signup: Credentials, validation to check parameters, creating token and storing in session storage
3. Login: Credentials, if user details not existing and not valid validation message. Redirecting to todonotes page when logged in. Storing username and toke in session storage
4. Todonotes page: Creating a notes, showing notes created by logged user, create a notes and updating notes
5. Routers: Headers shows logout, home page, todonote urls to navigate through pages
6. Validation as required
7. Data fetching: Used axios for fetching and updating data
8. Offline Mode: Cashing routers/pages with vue pwa plugin, register service workers

UI pages:

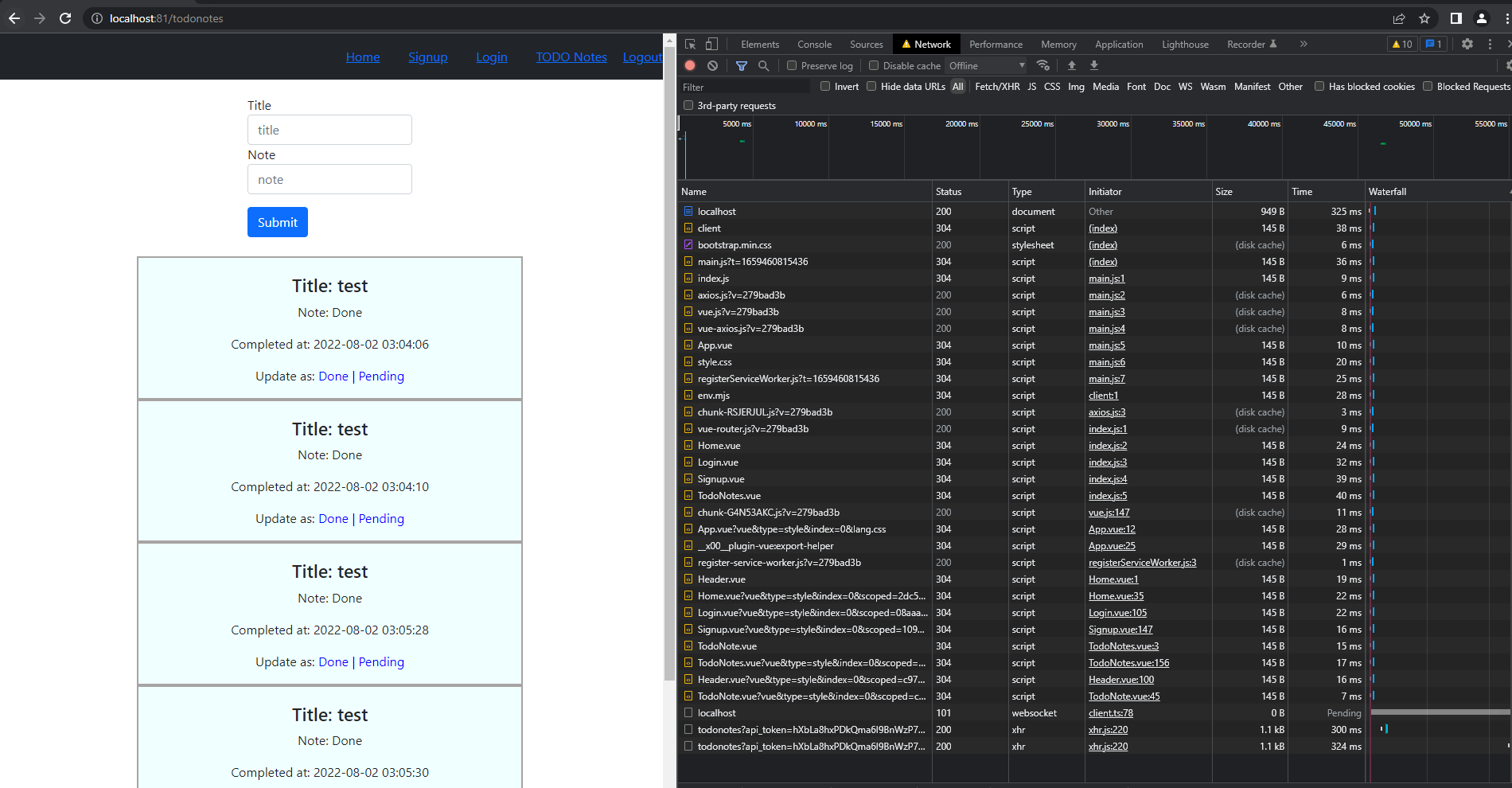
1.Login



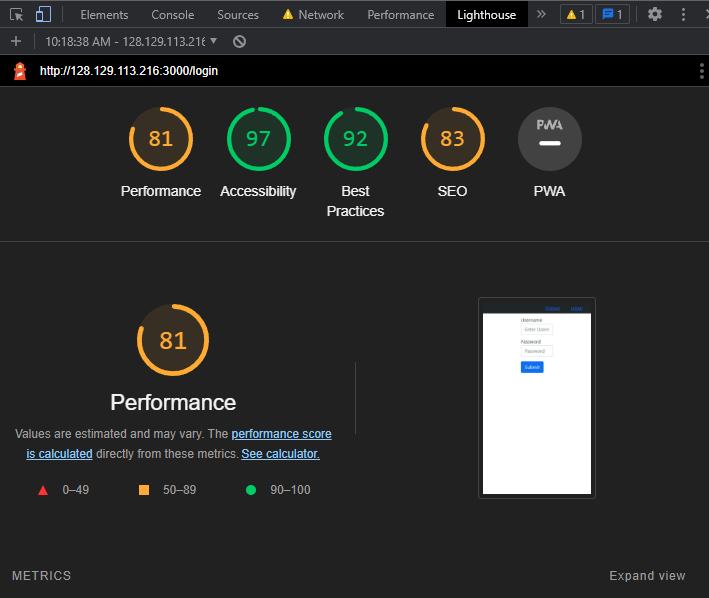
1. Signup:



1. Todonotes



Report of page:



Offiline Mode: Caching

