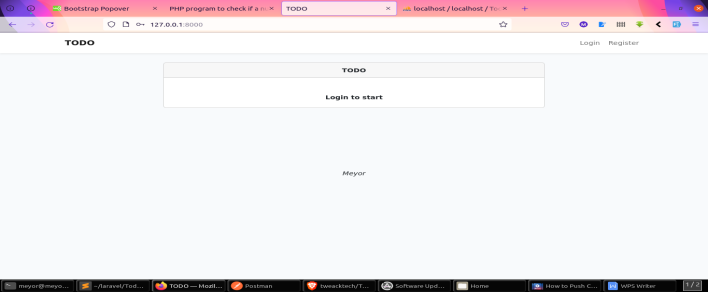
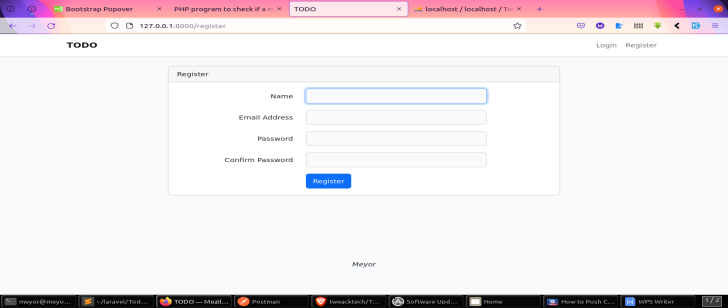
**Documentation for Todo laravel app**

**User documentation:**

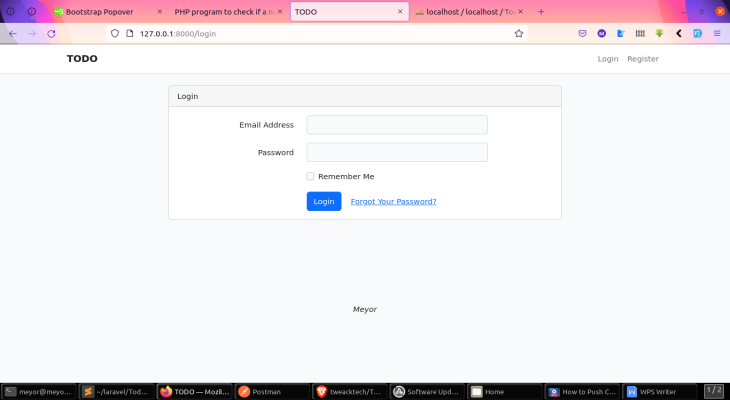


Landing page



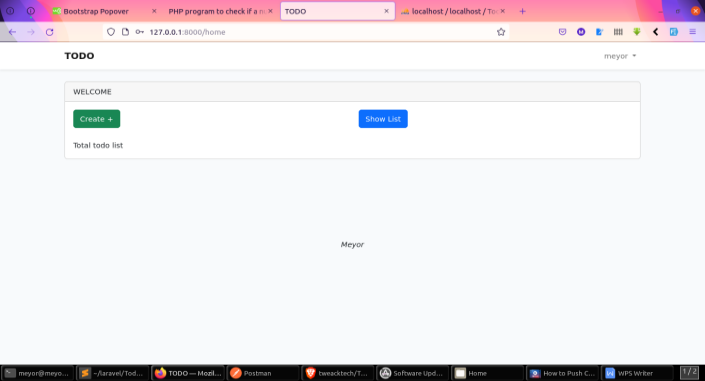
Registration page

User register using: Name, email and password(8 character)



Login page

User login with Email address and password

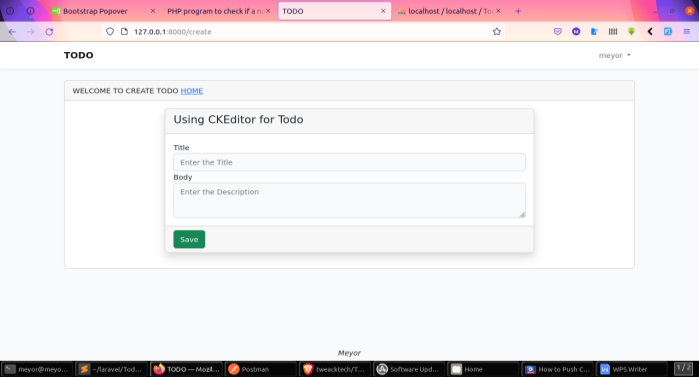


Home page

Home page consist of two buttons which is the Create and Show list.

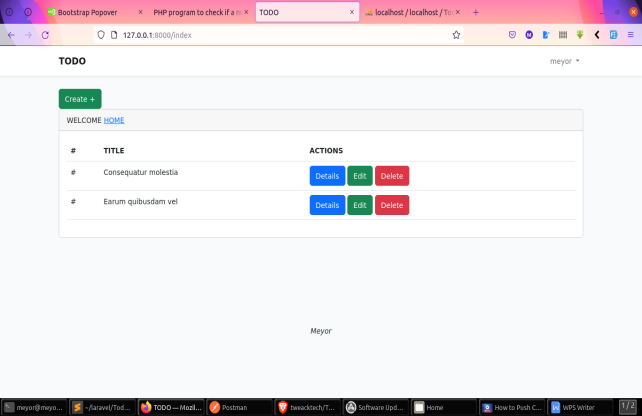
Create button for creating new todo

Show list button to show existing todo list



Create page

For creating new todo with title and descriptions then save

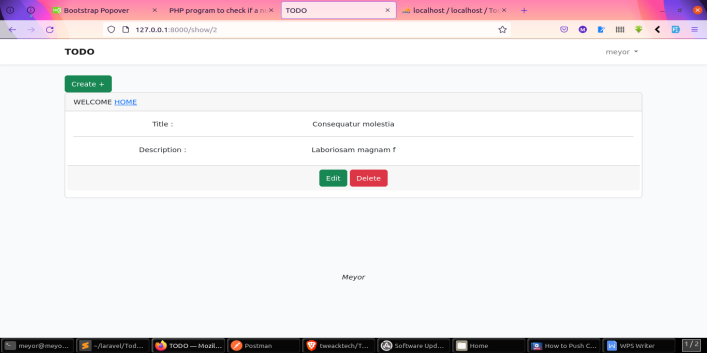


Show list page

This page list all existing to do, each with a detail,edit and delete button

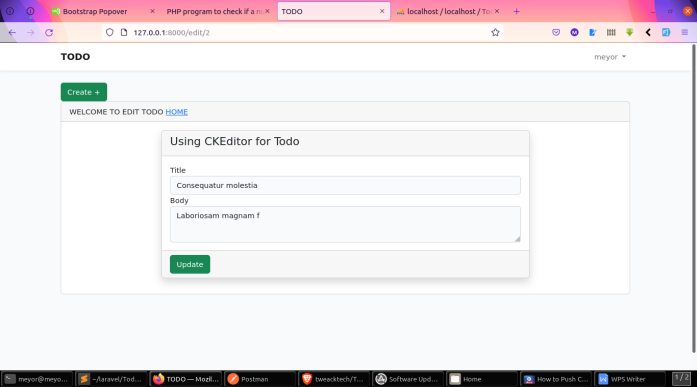
Clicking detail shows the title and description.

Clicking edit takes u to new page to edit selected item



Detail page

Show the detail of a selected item



Edit page

User able to edit item title or description

**Technical Documentation:**

Database Configuration :From env file,

DB\_DATABASE=Todo

DB\_USERNAME=root

DB\_PASSWORD=””

Routes: consist of web and api which I used

Web hold the routes for user interface

Api holds the routes for the api

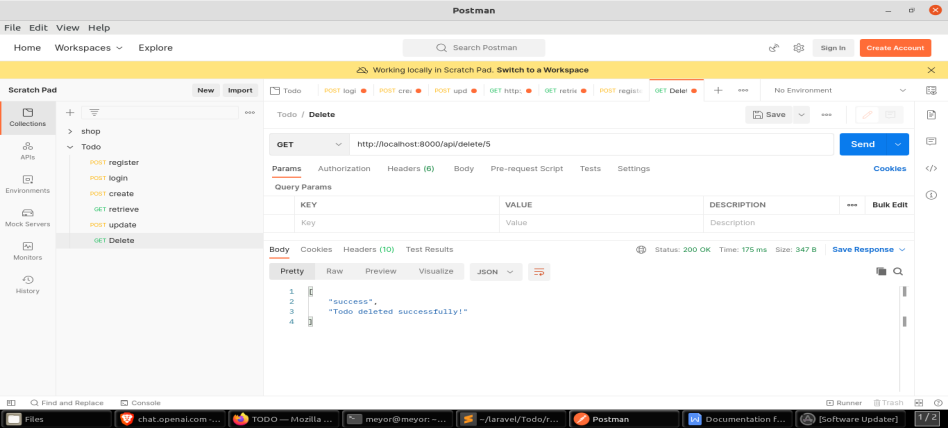
Resource: fold holds all the blade views for the user interface

App/Http/Controller: holds the controllers for both web and the api.

**API DOCUMENTATION:**

I actually used postman for testing .

|  |  |  |  |
| --- | --- | --- | --- |
| # | Page | Url | Method |
| 1 | User register | http://127.0.0.1:8000/api/register?email=mmoso@admin.com&password=12345678&name=pop | Post |
| 2 | Login | http://127.0.0.1:8000/api/register?email=admin@admin.com&password=12345678 | Post |
| 3 | Create todo | http://127.0.0.1:8000/api/store?user\_id=1&title=meyor&description=dari dammmmmm | Post |
| 4 | Retrieve todo | http://localhost:8000/api/show/1 | Get |
| 5 | Update todo | http://127.0.0.1:8000/api/update/1?title=meyor&description=dari dammmmm | Post |
| 5 | Delete Todo | <http://localhost:8000/api/delete/1> | Get |
| 6 | Retrieve all | <http://localhost:8000/api/index> | Get |



Postman api testing