

PRAKTIKUM SISTEM BASIS DATA

Nama	Muhammad Rifki Pratama	No. Modul	9
NPM	2206828903	Tipe	Tutam

1. Berikut adalah ss per file dari aplikasi Todo List, dimana aplikasi ini berguna sebagai note untuk membuat list dari apa yang ingin dilakukan , berikut adlaah strukturnya

Link video : https://youtu.be/aJ-R71MM_TI

Link Github : <https://github.com/MRifkiPratama/MyToDo>

Name	Date modified	Type	Size
backend	18/05/2024 18:33	File folder	
frontend	18/05/2024 19:36	File folder	

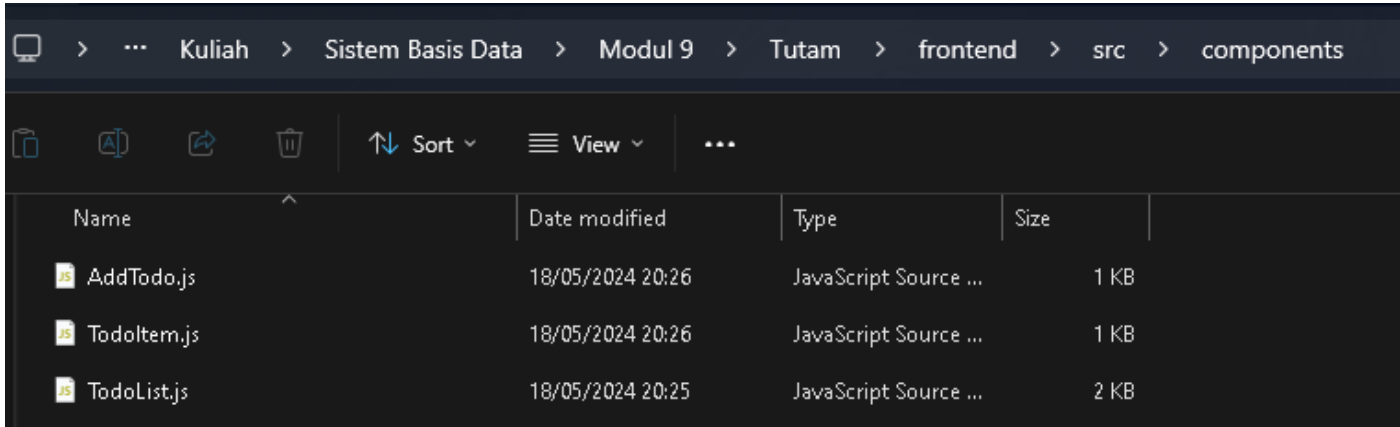
This PC > Windows (C:) > Kuliah > Sistem Basis Data > Modul 9 > Tutam > backend >				
Sort View ...				
Name	Date modified	Type	Size	
node_modules	18/05/2024 18:27	File folder		
package.json	18/05/2024 18:27	JSON Source File	1 KB	
package-lock.json	18/05/2024 18:27	JSON Source File	34 KB	
server.js	18/05/2024 18:36	JavaScript Source ...	2 KB	

This PC > Windows (C:) > Kuliah > Sistem Basis Data > Modul 9 > Tutam > frontend >

Sort View

Name	Date modified	Type	Size
.git	18/05/2024 19:14	File folder	
node_modules	18/05/2024 19:59	File folder	
public	18/05/2024 19:13	File folder	
src	18/05/2024 19:15	File folder	
.gitignore	18/05/2024 19:12	Git Ignore Source ...	1 KB
package.json	18/05/2024 19:59	JSON Source File	1 KB
package-lock.json	18/05/2024 19:59	JSON Source File	692 KB
README.md	18/05/2024 19:12	Markdown Source...	4 KB

Name	Date modified	Type	Size
components	18/05/2024 19:18	File folder	
App.css	18/05/2024 19:12	CSS Source File	1 KB
App.js	18/05/2024 20:25	JavaScript Source ...	2 KB
App.test.js	18/05/2024 19:12	JavaScript Source ...	1 KB
index.css	18/05/2024 19:12	CSS Source File	1 KB
index.js	18/05/2024 19:12	JavaScript Source ...	1 KB
logo.svg	18/05/2024 19:12	Microsoft Edge H...	3 KB
reportWebVitals.js	18/05/2024 19:12	JavaScript Source ...	1 KB
setupTests.js	18/05/2024 19:12	JavaScript Source ...	1 KB



Name	Date modified	Type	Size
AddTodo.js	18/05/2024 20:26	JavaScript Source ...	1 KB
TodoItem.js	18/05/2024 20:26	JavaScript Source ...	1 KB
TodoList.js	18/05/2024 20:25	JavaScript Source ...	2 KB

2. Berikut adalah kode dari isi filenya

- Backend
 - Server.js

```
const express = require('express');
const mongoose = require('mongoose');
const bodyParser = require('body-parser');
const cors = require('cors');

const app = express();
const port = 4000;

app.use(bodyParser.json());
app.use(cors());

mongoose.connect('mongodb+srv://muhammadrifkipratama:G4pIp6EYAPeWDETy@netlab.t
ahq0qe.mongodb.net/Modul9SBD');
const db = mongoose.connection;
db.on('error', console.error.bind(console, 'connection error: '));
db.once('open', function () {
  console.log('Connected to MongoDB Muhammad Rifki Pratama');
});

const TodoSchema = new mongoose.Schema({
  title: String,
  completed: Boolean,
});

const Todo = mongoose.model('Todo', TodoSchema);

app.post('/api/todos', async (req, res) => {
  const todo = new Todo({
    title: req.body.title,
```

```
        completed: false,
      });
      await todo.save();
      res.send(todo);
    });

app.get('/api/todos', async (req, res) => {
  const todos = await Todo.find();
  res.send(todos);
});

app.delete('/api/todos/:id', async (req, res) => {
  const result = await Todo.findByIdAndDelete(req.params.id);
  res.send(result);
});

app.listen(port, () => {
  console.log(`Server is running on http://localhost:${port}`);
});
```

- Frontend
 - AddTodo.js

```
import React, { useState } from 'react';

const AddTodo = ({ addTodo }) => {
  const [title, setTitle] = useState('');

  const onSubmit = (e) => {
    e.preventDefault();
    if (title) {
      addTodo(title);
      setTitle('');
    }
  };

  return (
    <form onSubmit={onSubmit}>
      <input
        type="text"
        value={title}
        onChange={(e) => setTitle(e.target.value)}
        placeholder="Add a new To Do List"
      />
      <button type="submit">Add</button>
    </form>
  );
};
```

```
    </form>
  );
};

export default AddTodo;
```

- TodoItem.js

```
import React from 'react';

const TodoItem = ({ todo, deleteTodo }) => {
  return (
    <div>
      <div>
        <span>{todo.title}</span>
        <button onClick={() => deleteTodo(todo._id)}>Delete</button>
      </div>
    </div>
  );
};

export default TodoItem;
```

- TodoList.js

```
import React, { useState, useEffect } from 'react';
import axios from 'axios';
import TodoItem from './TodoItem';
import AddTodo from './AddTodo';

const TodoList = () => {
  const [todos, setTodos] = useState([]);
  const [showTodos, setShowTodos] = useState(false);

  useEffect(() => {
    axios.get('http://localhost:4000/api/todos')
      .then(response => setTodos(response.data))
      .catch(error => console.error(error));
  }, []);

  const addTodo = (title) => {
    axios.post('http://localhost:4000/api/todos', { title })
```

```

        .then(response => setTodos([...todos, response.data]))
        .catch(error => console.error(error));
    };

    const deleteTodo = (id) => {
        axios.delete(`http://localhost:4000/api/todos/${id}`)
            .then(() => setTodos(todos.filter(todo => todo._id !== id)))
            .catch(error => console.error(error));
    };

    const handleShowTodos = () => {
        setShowTodos(!showTodos);
    };

    return (
        <div>
            <h1>Todo List</h1>
            <AddTodo addTodo={addTodo} />
            <button onClick={handleShowTodos}>
                {showTodos ? 'Hide list' : 'Show list'}
            </button>
            {showTodos && (
                <ul>
                    {todos.map(todo => (
                        <li key={todo._id}>
                            <TodoItem todo={todo} deleteTodo={deleteTodo} />
                        </li>
                    ))}
                </ul>
            )}
        </div>
    );
};

export default TodoList;

```

- App.js

```

import React from 'react';
import { BrowserRouter as Router, Route, Routes } from 'react-router-dom';
import { Navbar, Nav, Container } from "react-bootstrap";
import Card from "react-bootstrap/Card";
import TodoList from './components/TodoList';
const App = () => {
    return (

```

```
<div>
  {/* Navigation Bar */}
  <nav className="bg-blue-600 p-4">
    <div className="container mx-auto flex justify-between items-
center">
      <div className="text-white text-lg font-bold">My To-
Do</div>
      <ul className="flex space-x-4 text-white">
        <li><a href="#home" className="hover:text-gray-
300">Home</a></li>
        <li><a href="#profile" className="hover:text-gray-
300">Profile</a></li>
        <li><a href="#contact" className="hover:text-gray-
300">Contact</a></li>
      </ul>
    </div>
  </nav>

  {/* Router */}
  <Router>
    <Routes>
      <Route path="/" element={<ToDoList />} />
    </Routes>
  </Router>
</div>
);
};

export default App;
```

3. Berikut adalah ss dari webnya



My To-Do

- [Home](#)
- [Profile](#)
- [Contact](#)

Todo List



My To-Do

- [Home](#)
- [Profile](#)
- [Contact](#)

Todo List

- Lunch - 12:30
- Class 13:00
- Go home 14:40