

Instalasi Project Express JS

by Jihadul Akbar

Buat Database

Buat database

```
CREATE DATABASE fjs-backend;;
```

Buat Table Baru User

```
CREATE TABLE users (  
  user_id INT PRIMARY KEY AUTO_INCREMENT,  
  name VARCHAR(50) NOT NULL,  
  email VARCHAR(100) NOT NULL UNIQUE,  
  password VARCHAR(255) NOT NULL,  
  is_active BOOLEAN DEFAULT TRUE,  
  level ENUM('Admin', 'Petugas', 'Pelanggan'),  
  expire_time DATETIME,  
  created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP,  
  updated_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP  
);
```

Insert Tabel

```
INSERT INTO users  
  (user_id, name, email, password,  
   is_active, level, expire_time,  
   created_at, updated_at)
```

```
VALUES  
  ('1234', 'Jihadul Akbar',  
   'jihadulakbar@gmail.com',  
   '12345678', '1', 'Admin',  
   '2024-11-11 07:55:48.000000',  
   current_timestamp(),  
   current_timestamp());
```

Instalasi Project Express JS

Inisialisasi Proyek dengan Express.js

Buat File package.json

```
npm init -y
```

Inisialisasi package

```
npm install express mysql2 dotenv body-parser
```


Buat File index.js

```
const express = require('express')
const app = express()
const port = 3000

app.get('/', (req, res) => {
  res.send('Hello World!')
})

app.listen(port, () => {
  console.log(`Example app listening on port ${port}`)
})
```

Jalankan File `index.js`

```
node index.js
```

Nodemon

***Nodemon** adalah utilitas yang akan memantau perubahan apa pun pada sumber Anda dan secara otomatis memulai ulang server Anda. Sempurna untuk pengembangan aplikasi Node.js*

Instalasi Nodemon dengan NPM

```
npm install -g nodemon
```

Jalankan Nodemon

```
nodemon index.js
```

Struktur Project

```
api-backend
├── .env
├── index.js
├── routes
│   └── userRoutes.js
├── controllers
│   └── userController.js
├── utils
│   └── db.js
```

Configurasi dotenv

Buat File .env

```
DB_HOST=127.0.0.1  
DB_USER=root  
DB_PASSWORD=  
DB_NAME=fjs-backend
```

Buat File index.js

```
require("dotenv").config();
const express = require("express");
const bodyParser = require("body-parser");

const app = express();
app.use(bodyParser.json());

const db = require("./utils/db"); // Menghubungkan ke database

app.use("/api/users", require("./routes/userRoutes"));

app.listen(3000, () => {
  console.log("Server running on port 3000");
});
```


Buat File routes/userRoutes.js

```
const express = require("express");
const router = express.Router();
const userController = require("../controllers/userController");

router.get("/", userController.getUsers);
router.get("/:id", userController.getUserById);
router.post("/", userController.createUser);
router.put("/:id", userController.updateUser);
router.delete("/:id", userController.deleteUser);

module.exports = router;
```

Buat File controllers/userControllers.js

```
const db = require("../utils/db");

// Mendapatkan semua user
exports.getUsers = (req, res) => {
  db.query("SELECT * FROM users", (err, results) => {
    if (err) return res.status(500).send(err);
    res.json(results);
  });
};

// Mendapatkan user berdasarkan ID
exports.getUserById = (req, res) => {
  const { id } = req.params;
  db.query("SELECT * FROM users WHERE user_id = ?", [id], (err,
    if (err) return res.status(500).send(err);
    res.json(results[0]);
  );
};
```

Buat File utils/db.js

```
// db.js
const mysql = require("mysql2");

const db = mysql.createConnection({
  host: process.env.DB_HOST,
  user: process.env.DB_USER,
  password: process.env.DB_PASSWORD,
  database: process.env.DB_NAME,
  connectTimeout: 10000,
});

db.connect((err) => {
  if (err) {
    console.error("Database connection failed:", err);
    return;
  }
});
```

Postman

Download Postman

Postman adalah platform API untuk membangun dan menggunakan API. Postman menyederhanakan setiap langkah siklus hidup API dan merampingkan kolaborasi sehingga Anda dapat membuat API yang lebih baik—lebih cepat.

Test API

endpoint : <http://localhost:3000/api/users>

GET: Mendapatkan Daftar Pengguna

1. Buka Postman dan pilih metode GET.
2. Masukkan URL endpoint API Anda, misalnya:
<http://localhost:3000/api/users>.
3. Klik tombol Send untuk mengirim permintaan.
4. Postman akan menampilkan respons yang dikirim server, seperti daftar pengguna dalam format JSON.

GET: Mendapatkan Pengguna Berdasarkan ID

1. Pilih metode GET.
2. Masukkan URL endpoint API untuk mengambil data pengguna berdasarkan ID, misalnya:
<http://localhost:3000/api/users/1> (di mana 1 adalah user_id).
3. Klik Send.
4. Anda akan menerima respons data pengguna yang sesuai.

POST: Menambahkan Pengguna Baru

1. Pilih metode POST.
2. Masukkan URL endpoint API untuk menambahkan pengguna, misalnya:
<http://localhost:3000/api/users>
3. Pilih tab Body -> raw -> JSON.
4. Masukkan data pengguna baru dalam format JSON, misalnya:

```
{  
  "name": "John Doe",  
  "email": "johndoe@example.com",  
  "password": "password123",  
  "is_active": true,  
  "level": "user",  
  "expire_time": "2024-12-31 23:59:59"  
}
```

5. Klik Send.
 6. Jika berhasil, Anda akan melihat respons dari server dengan data pengguna yang baru ditambahkan.
-

PUT: Memperbarui Data Pengguna

1. Pilih metode PUT.
2. Masukkan URL endpoint untuk memperbarui data pengguna berdasarkan ID, misalnya:
<http://localhost:3000/api/users/1>.
3. Pilih Body -> raw -> JSON.
4. Masukkan data yang ingin diperbarui, misalnya:

```
{  "name": "Jane Doe",
  "email": "janedoe@example.com",
  "password": "newpassword",
  "is_active": true,
  "level": "admin",
  "expire_time": "2025-12-31 23:59:59"
}
```

5. Klik Send.
6. Anda akan menerima respons yang menunjukkan bahwa data pengguna berhasil diperbarui.

The first of these is the fact that the system is not a simple one. It is a complex system, and as such, it is not possible to understand it by looking at its parts in isolation. The system is a whole, and its behavior is determined by the interactions between its parts. This is a fundamental principle of systems thinking, and it is one that is often overlooked in traditional approaches to problem-solving.

The second of these is the fact that the system is dynamic. It is not a static system, and its behavior changes over time. This is another fundamental principle of systems thinking, and it is one that is often overlooked in traditional approaches to problem-solving.

The third of these is the fact that the system is open. It is not a closed system, and it interacts with its environment. This is another fundamental principle of systems thinking, and it is one that is often overlooked in traditional approaches to problem-solving.

The fourth of these is the fact that the system is self-organizing. It is not a system that is controlled from the outside, and it is not a system that is designed from the top down. It is a system that organizes itself, and its behavior emerges from the interactions between its parts. This is another fundamental principle of systems thinking, and it is one that is often overlooked in traditional approaches to problem-solving.

The fifth of these is the fact that the system is resilient. It is not a system that is fragile, and it is not a system that is easily disrupted. It is a system that is able to withstand change, and it is able to adapt to new circumstances. This is another fundamental principle of systems thinking, and it is one that is often overlooked in traditional approaches to problem-solving.

The sixth of these is the fact that the system is sustainable. It is not a system that is unsustainable, and it is not a system that is doomed to failure. It is a system that is able to continue to exist, and it is able to thrive in the face of change. This is another fundamental principle of systems thinking, and it is one that is often overlooked in traditional approaches to problem-solving.

The seventh of these is the fact that the system is equitable. It is not a system that is unfair, and it is not a system that is biased. It is a system that is able to provide for the needs of all its members, and it is able to ensure that everyone has a fair chance of success. This is another fundamental principle of systems thinking, and it is one that is often overlooked in traditional approaches to problem-solving.

The eighth of these is the fact that the system is transparent. It is not a system that is opaque, and it is not a system that is hidden. It is a system that is able to be understood, and it is able to be managed. This is another fundamental principle of systems thinking, and it is one that is often overlooked in traditional approaches to problem-solving.

The ninth of these is the fact that the system is accountable. It is not a system that is irresponsible, and it is not a system that is unaccountable. It is a system that is able to be held responsible for its actions, and it is able to be held accountable for its results. This is another fundamental principle of systems thinking, and it is one that is often overlooked in traditional approaches to problem-solving.

The tenth of these is the fact that the system is inclusive. It is not a system that is exclusive, and it is not a system that is discriminatory. It is a system that is able to include everyone, and it is able to ensure that everyone has a voice. This is another fundamental principle of systems thinking, and it is one that is often overlooked in traditional approaches to problem-solving.

The eleventh of these is the fact that the system is innovative. It is not a system that is stagnant, and it is not a system that is uncreative. It is a system that is able to generate new ideas, and it is able to implement them. This is another fundamental principle of systems thinking, and it is one that is often overlooked in traditional approaches to problem-solving.

The twelfth of these is the fact that the system is adaptable. It is not a system that is inflexible, and it is not a system that is rigid. It is a system that is able to change, and it is able to adapt to new circumstances. This is another fundamental principle of systems thinking, and it is one that is often overlooked in traditional approaches to problem-solving.

The thirteenth of these is the fact that the system is resilient. It is not a system that is fragile, and it is not a system that is easily disrupted. It is a system that is able to withstand change, and it is able to adapt to new circumstances. This is another fundamental principle of systems thinking, and it is one that is often overlooked in traditional approaches to problem-solving.

The fourteenth of these is the fact that the system is sustainable. It is not a system that is unsustainable, and it is not a system that is doomed to failure. It is a system that is able to continue to exist, and it is able to thrive in the face of change. This is another fundamental principle of systems thinking, and it is one that is often overlooked in traditional approaches to problem-solving.

The fifteenth of these is the fact that the system is equitable. It is not a system that is unfair, and it is not a system that is biased. It is a system that is able to provide for the needs of all its members, and it is able to ensure that everyone has a fair chance of success. This is another fundamental principle of systems thinking, and it is one that is often overlooked in traditional approaches to problem-solving.

The sixteenth of these is the fact that the system is transparent. It is not a system that is opaque, and it is not a system that is hidden. It is a system that is able to be understood, and it is able to be managed. This is another fundamental principle of systems thinking, and it is one that is often overlooked in traditional approaches to problem-solving.

The seventeenth of these is the fact that the system is accountable. It is not a system that is irresponsible, and it is not a system that is unaccountable. It is a system that is able to be held responsible for its actions, and it is able to be held accountable for its results. This is another fundamental principle of systems thinking, and it is one that is often overlooked in traditional approaches to problem-solving.

The eighteenth of these is the fact that the system is inclusive. It is not a system that is exclusive, and it is not a system that is discriminatory. It is a system that is able to include everyone, and it is able to ensure that everyone has a voice. This is another fundamental principle of systems thinking, and it is one that is often overlooked in traditional approaches to problem-solving.

The nineteenth of these is the fact that the system is innovative. It is not a system that is stagnant, and it is not a system that is uncreative. It is a system that is able to generate new ideas, and it is able to implement them. This is another fundamental principle of systems thinking, and it is one that is often overlooked in traditional approaches to problem-solving.

The twentieth of these is the fact that the system is adaptable. It is not a system that is inflexible, and it is not a system that is rigid. It is a system that is able to change, and it is able to adapt to new circumstances. This is another fundamental principle of systems thinking, and it is one that is often overlooked in traditional approaches to problem-solving.

DELETE: Menghapus Pengguna Berdasarkan ID

1. Pilih metode DELETE.
2. Masukkan URL endpoint untuk menghapus pengguna berdasarkan ID, misalnya:
<http://localhost:3000/api/users/1>.
3. Klik Send.
4. Anda akan mendapatkan respons yang menyatakan bahwa pengguna berhasil dihapus.

Memeriksa Respons dan Status Code

Setiap kali Anda mengirim permintaan, periksa status code di bagian atas respons Postman:

- 200 OK untuk permintaan yang berhasil.
- 201 Created jika berhasil menambahkan data.
- 404 Not Found jika endpoint atau data tidak ditemukan.
- 500 Internal Server Error jika terjadi kesalahan pada server.

—

Q & A

Refrensi

- [expressjs](#)
- [postman](#)
- [nodemon](#)
- [mysql2](#)
- [dotenv](#)
- [express-validator](#)
- [express-fileupload](#)
- [nodejs](#)
- [npm](#)
- [github](#)