FULL STACK APPLICATION EXAMPLE

An application that accepts username and password from the front-end and saves them in MongoDB.

1. **Create project folders**

```
mkdir login

cd login

mkdir server
```

2. **Set up backend (server)**

```
cd server
npm init -y
npm install express cors mongoose
```

3. **Create database.js, user.js and server.js files in server folder**

database.js

```
// database.js
const mongoose = require("mongoose");

mongoose.connect("mongodb://127.0.0.1:27017/testdb")
.then(() => console.log("MongoDB Connected"))
.catch(err => console.log("DB error", err));
```

server.js

```
// server.js
const express = require("express");
const cors = require("cors");
require("./database");
const User = require("./user");

const app = express();

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

app.post("/register", async (req, res) => {
  const { username, password } = req.body;

  try {
    await User.create({ username, password });
    res.json({ msg: "User saved" });
  } catch (err) {
    res.status(500).json({ msg: "error", error: err.message });
  }
});

app.listen(5000, () => console.log("Server running on port 5000"));
```

user.js

```
// user.js
const mongoose = require("mongoose");

const userSchema = new mongoose.Schema({
  username: String,
  password: String
});

module.exports = mongoose.model("User", userSchema);
```

4. Create frontend (Vite + React)

cd login

npm create vite@latest client

edit App.jsx

```
import { useState } from "react";

export default function App() {

  const [u,setU] = useState("");
  const [p,setP] = useState("");
  const [msg,setMsg] = useState("");

  const submit = async e => {
    e.preventDefault();

    const r = await fetch("http://localhost:5000/register", {
      method:"POST",
      headers:{ "Content-Type":"application/json" },
      body: JSON.stringify({ username: u, password: p })
    });

    const d = await r.json();
    setMsg(d.msg || d.message || JSON.stringify(d));
  };

  return (
    <form onSubmit={submit} style={{maxWidth:360,margin:20}}>
      <h3>Register</h3>

      <input
        placeholder="username"
        value={u}
        onChange={e=>setU(e.target.value)}
      /><br/>

      <input
        placeholder="password"
```

```
        type="password"
        value={p}
        onChange={e=>setP(e.target.value)}
      /><br/>

      <button>Save</button>
      <p>{msg}</p>
    </form>
  );
}
```

5. **Start backend**

   cd server
   node server.js

   You should see:
   MongoDB Connected
   server running on :5000

6. **Start Frontend**

   cd client

   npm run dev

7. **Verify saved data in mongosh**

   Open terminal and run

   mongosh
   use testdb
   db.users.find().pretty()

   you should see document like this

```
{
  "_id": ObjectId("..."),
  "username": "yourname",
  "password": "yourpassword",
  "__v": 0
}
```