

```

## ✅ Final `README.md` for SkillSwap

```md
🎓 SkillSwap - Peer Skill Matching & Chat Platform

SkillSwap is a full-stack web application that allows users to exchange skills with one another. Users can register, list the skills they **have** and the ones they **want**, get matched with compatible peers, and initiate **WhatsApp-style real-time chat** with matched partners.

💡 Key Features

- 🔒 User authentication (JWT-based)
- 📝 Register with name, email, password, and skills
- 🔎 Get automatically matched with users based on complementary skills
- 🤝 Send skill swap requests to matched users
- 🧠 Real-time chat between users (with auto-refresh)
- 🎨 Clean UI with Tailwind CSS and smooth transitions
- 🚧 Fully responsive and mobile-ready
- 🏠 MySQL backend (runs locally)

🛡️ Technologies Used

Frontend:
- React
- React Router
- Tailwind CSS
- Axios

Backend:
- Node.js
- Express
- Sequelize ORM
- JWT (Authentication)
- CORS

Database:
- MySQL (Local instance)

📁 Project Directory Structure

```
SkillSwap/
  └── backend/
      ├── config/          # DB & JWT configuration
      ├── controllers/    # Handles route logic
      └── models/          # Sequelize models (User, Chat, Request)
```

```

```
|- routes/ # API route definitions
|- server.js # Entry point for Express server
|- package.json # Backend dependencies and scripts

-- frontend/
 |- public/
 |- src/
 |- components/ # Reusable components (Navbar, etc.)
 |- pages/ # Pages: Login, Register, Dashboard, Chat
 |- utils/
 |- App.js # Main routing logic
 |- index.css # Tailwind CSS styles
 |- main.jsx # Entry point
 |- tailwind.config.js # Tailwind setup
 |- package.json # Frontend dependencies and scripts

-- README.md # You are here ✨
```

```

🌐 How to Run Locally

> SkillSwap runs **entirely on localhost** using a local MySQL database.
No live server or deployment is required.

🔧 Prerequisites:

- Node.js (v16+)
- MySQL installed and running
- Git (optional but recommended)

📁 Step 1: Clone the Repository

```
```bash
git clone https://github.com/fighter82899/Skill-Swap.git
cd SkillSwap
```
```

> Or download as ZIP and extract.

🌈 Step 2: Set Up MySQL Database

1. Create a database in MySQL named `skillswap_db`
2. Import the schema (SQL file can be created using Sequelize or manually)

```
```sql
CREATE DATABASE skillswap_db;
```
```

> Sequelize will automatically generate tables like `Users`, `Requests`, and `Chats`.

🛡 Step 3: Configure Backend

1. Navigate to the backend folder:

```
```bash
cd backend
````
```

2. Install dependencies:

```
```bash
npm install
````
```

3. Create a ` `.env` file (or hardcode your config in ` config/db.js`) with:

```
```
DB_HOST=localhost
DB_USER=root
DB_PASSWORD=your_password
DB_NAME=skillswap_db
JWT_SECRET=your_secret_key
````
```

4. Start the backend server:

```
```bash
npm run dev # Uses nodemon for hot reload
OR
npm start
````
```

> The backend runs at ` http://localhost:5000`

🎨 Step 4: Configure Frontend

1. Navigate to frontend folder:

```
```bash
cd ../frontend
````
```

2. Install dependencies:

```
```bash
npm install
````
```

3. Start the frontend server:

```
```bash
npm start
````
```

> The app opens in `http://localhost:3000`

🚀 Application Workflow

1. **Register:** Enter name, email, password, skills you have, and skills you want
2. **Dashboard:** View your profile and matched users
3. **Send Request:** Send skill swap requests to compatible users
4. **Chat:** Once matched, chat via real-time interface (like WhatsApp)

🖼 UI Highlights

- * 🌈 Animated HomePage with gradient background
- * 💼 Dashboard with glassmorphism and match cards
- * 💬 Chat bubbles with real-time messaging
- * 📱 Fully mobile responsive

> Built with ❤️ using modern frontend design principles

🛡 Testing Endpoints (Optional - Postman)

User Auth:

- * `POST /api/users/register`
- * `POST /api/users/login`
- * `GET /api/users/profile`

Matching & Requests:

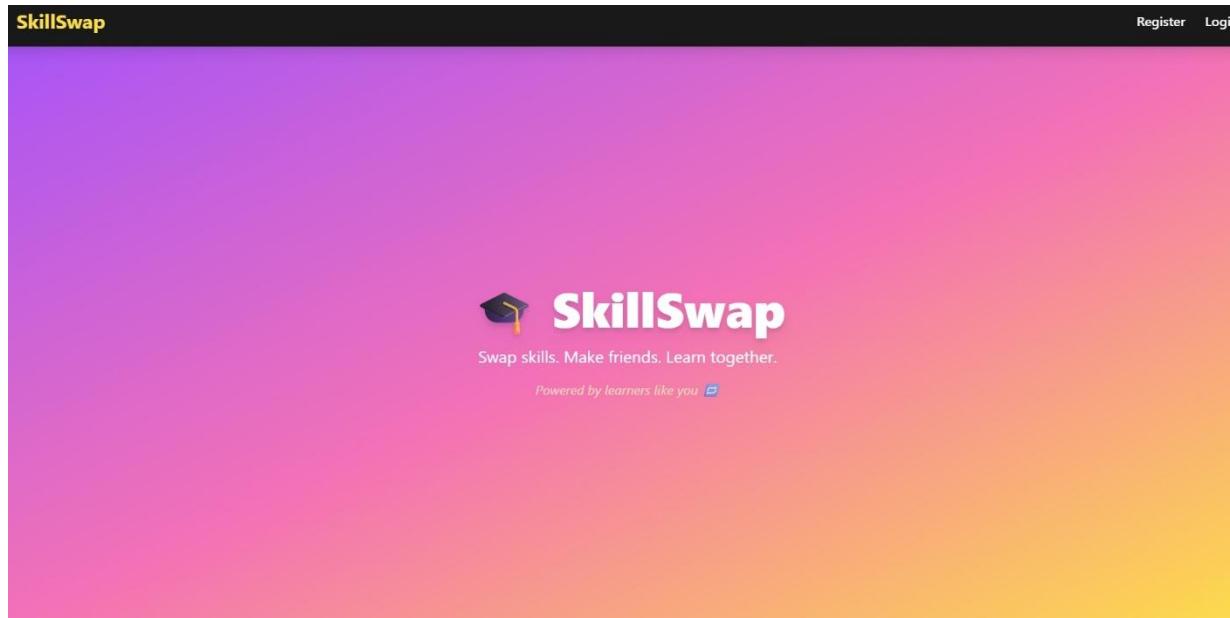
- * `GET /api/users/match`
- * `POST /api/requests/send`

Chat:

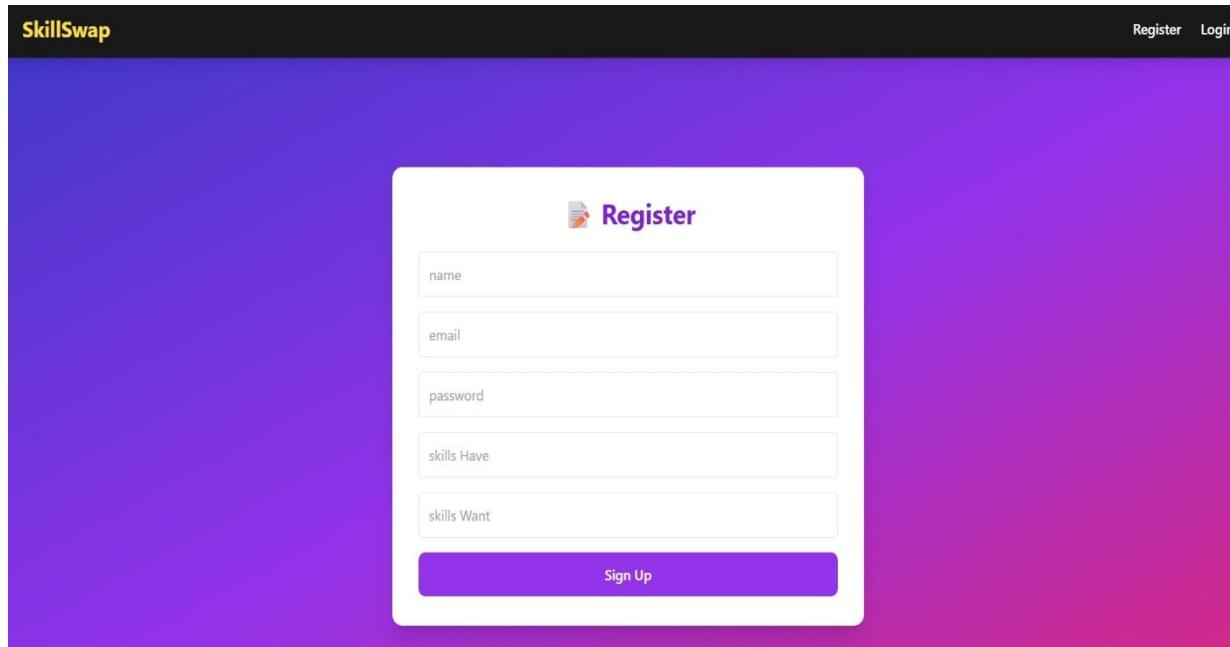
- * `POST /api/chats/send`
- * `GET /api/chats/:partnerId`

📸 Screenshots

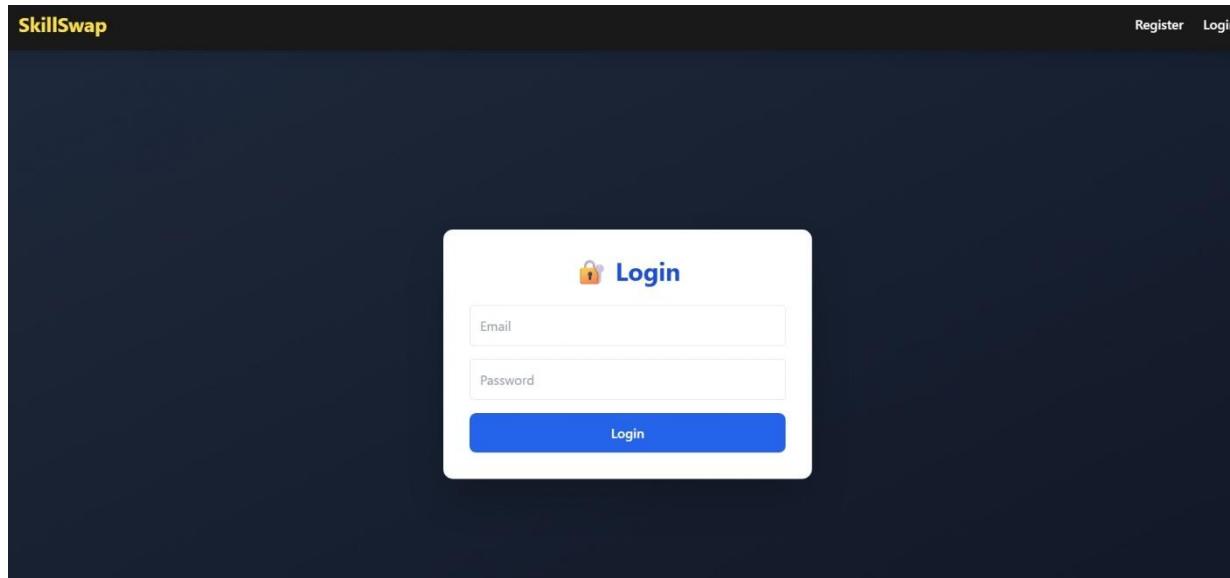
Home Page



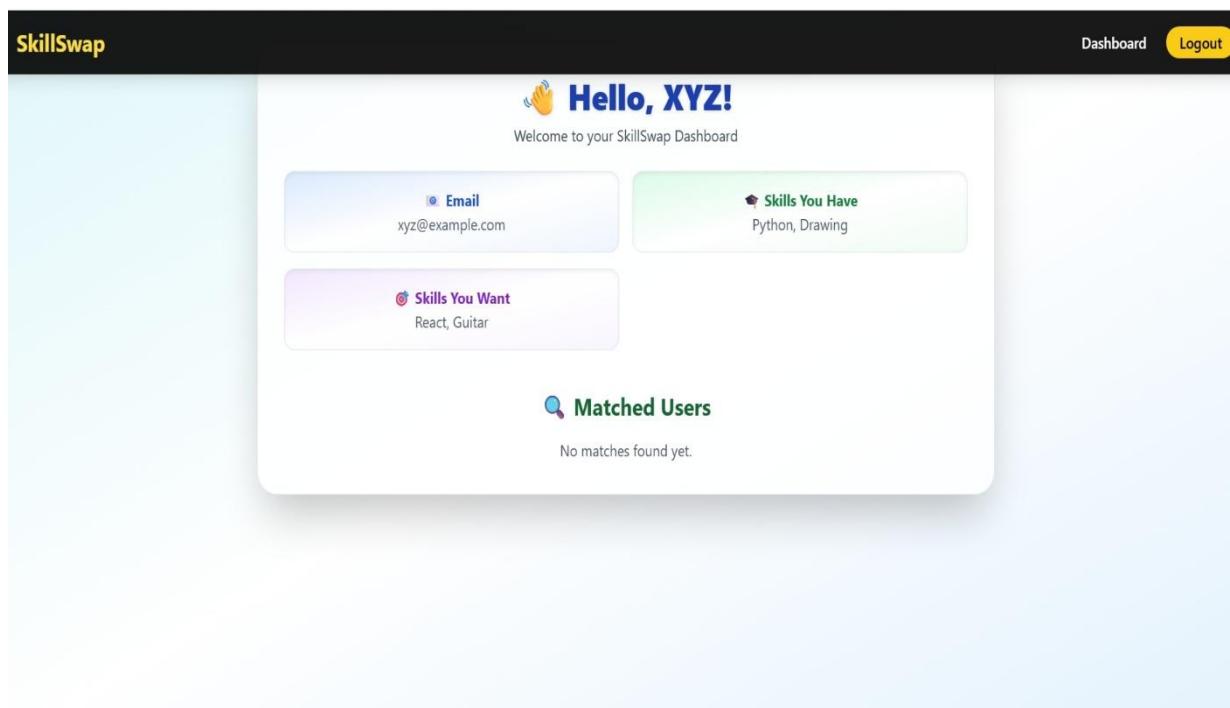
Register Page



Login Page



Dashboard Page



Matching Results

The dashboard shows a greeting "Hello, WXY!" and a welcome message "Welcome to your SkillSwap Dashboard". It displays three sections: "Email" (wxy@example.com), "Skills You Have" (java), and "Skills You Want" (python,drawing). A section titled "Matched Users" lists a user named "XYZ" with the following details: Skills Have: Python, Drawing; Skills Want: React, Guitar. A "Send Request" button is present.

Send Request Confirmation

Matched Users

XYZ

Skills Have: Python, Drawing
Skills Want: React, Guitar

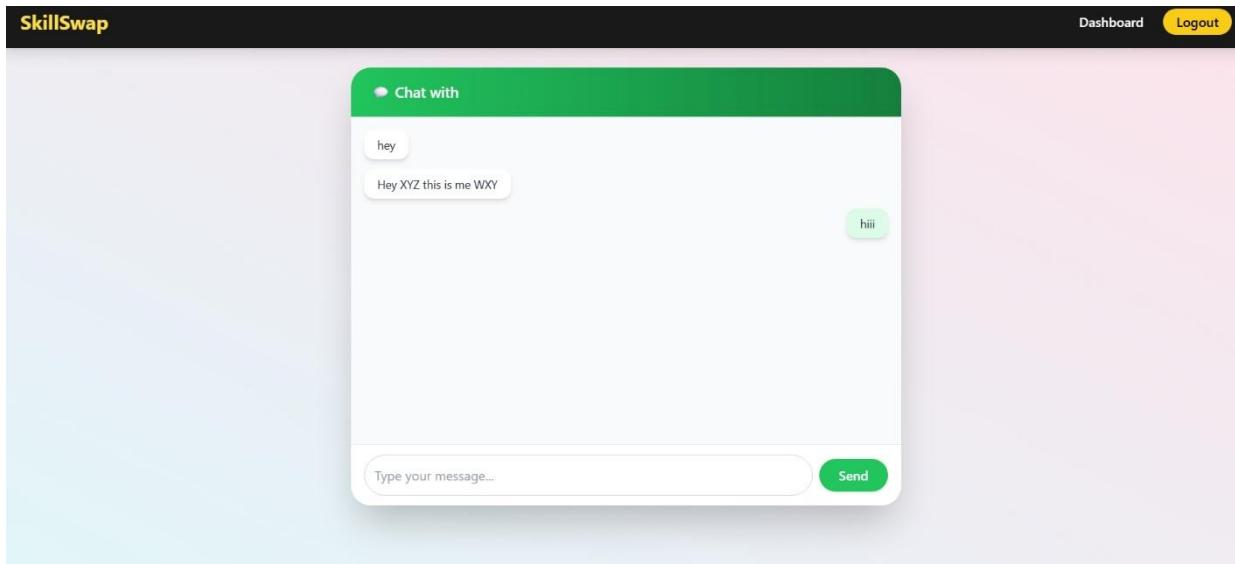
[Send Request](#)

Request Sent

[Chat](#)

Chat Page

The chat page shows a green header "Chat with XYZ". A message bubble contains the word "hi". At the bottom, there is an input field "Type your message..." and a "Send" button.



📦 Future Enhancements

- * Voice messages
 - * File sharing in chat
 - * Notifications
 - * Profile image upload
-

🤝 Contribution

This is a beginner-friendly, open-ended project.
Feel free to fork or extend it for your learning or portfolio!

📄 License

This project is open source and available for educational use.

Let me know if you want this README saved in a `*.md` file or need a version with embedded screenshots, deploy steps, or demo video instructions.

You're officially finished with the **SkillSwap full-stack app!** 🎉
🔥 Congrats on building something amazing, Buddy!!
