Backend End Engineering

Project Report

Semester-V (Batch-2022)

Skill Barter Platform

Supervised By:

Raveesh Samkaria

Submitted By:

Yogesh Choudhary(2210990986)

Ushabh Jain(2210990929)

Group - 31

Department of Computer Science and Engineering

Chitkara University Institute of Engineering & Technologye,

Chitkara University, Punjab

Abstract

In the modern landscape of digital skill exchange, users demand intuitive, secure, and efficient platforms to connect and share knowledge. The Skill Barter Platform is meticulously designed to meet this need, offering a powerful web application built with Node.js, MongoDB, Express.js, and EJS (Embedded JavaScript Templates). This platform enables users to seamlessly trade skills, fostering a collaborative environment for mutual learning and growth . EJS is employed to create a dynamic, responsive, and engaging user interface, ensuring a smooth user experience. On the server side, Node.js provides the scalability and efficiency needed to handle multiple users, while Express.js serves as the backbone for routing and middleware, ensuring optimal performance . EJS is integrated into the frontend to render dynamic HTML content, allowing for real-time updates and an interactive user experience. MongoDB, coupled with Mongoose, serves as the database, facilitating schema-based data modeling and efficient management of user information . Security is a top priority, with Bcrypt implemented for user authentication, encrypting sensitive data like passwords to safeguard against unauthorized access. This ensures that user accounts and data are protected, providing users with a secure and trustworthy platform to exchange skills.

Table of Contents

Introduction

As the demand for skill-sharing and collaborative learning continues to grow, having an effective platform to facilitate these exchanges is essential. The Skill Barter Platform addresses this need by offering a cutting-edge web application designed to streamline and enhance the process of skill exchange. Built with Node.js and Express.js, the application delivers high performance and scalability, ensuring a seamless user experience . The platform utilizes MongoDB with Mongoose for efficient data management and schema-based modeling, allowing for easy tracking and organization of user skills and interactions. The user interface, crafted using EJS (Embedded JavaScript Templates), provides dynamic, real-time updates and an interactive experience that engages users . Security is a top priority, with Bcrypt implemented for secure password hashing and authentication, ensuring that users' personal information and communications remain protected. The application boasts a range of robust features: users can securely register and log in, create and manage detailed profiles showcasing their skills, and connect with others to exchange knowledge and services . The interface is fully responsive, offering an optimal experience across various devices, making it easy for users to access the platform from anywhere. Additionally, the application supports the integration of personalized skill offers and requests, allowing users to tailor their experience to their specific needs and interests . By combining advanced technology with a user-centered design, the Skill Barter Platform provides a powerful tool for individuals looking to share and acquire new skills. Its comprehensive approach aims to make skill exchange as efficient and rewarding as possible, fostering a community of continuous learning and collaborator .

Problem Statement

In today's fast-paced and interconnected world, individuals possess a wide range of valuable skills and knowledge but often lack access to resources or the means to acquire new skills that could enhance their personal or professional growth. Traditional skill acquisition methods, such as formal education or paid courses, can be costly and time-consuming, making them inaccessible to many . Furthermore, the global economy is increasingly shifting towards a gig-based and freelance work environment, where individuals seek flexible opportunities to exchange their skills and talents. However, there is a lack of platforms that facilitate direct, equitable exchanges of skills between individuals without the need for monetary transactions. Current online platforms primarily focus on paid services or are limited to specific regions, making it difficult for people to connect with others who have complementary skills.

The lack of a structured and accessible way to barter skills hinders the potential for mutual learning, growth, and collaboration among individuals from diverse backgrounds. This gap highlights the need for a platform that allows users to exchange skills in a fair, transparent, and secure manner, fostering a community-driven economy where knowledge and skills are the currency.

OBJECTIVES

Facilitate Skill Exchange:

Create a user-friendly platform that allows individuals to easily list their skills and browse or search for others with complementary skills .Enable users to propose and negotiate fair skill exchanges without the need for monetary transactions.

Build a Secure and Trustworthy Environment:

Implement user verification processes to ensure the authenticity and credibility of participants . Incorporate user reviews, ratings, and a feedback system to build trust within the community.

Promote Global Accessibility and Inclusivity:

Design the platform to be accessible to a diverse, global audience, with features that support multiple languages and time zones . Ensure that the platform is inclusive and caters to a wide range of skill sets, from technical skills to creative and vocational skills.

Develop a Fair Barter Mechanism:

Create a system that facilitates equitable skill exchanges, taking into account the value, time, and effort involved in the offered skills . Provide tools for users to negotiate and agree on fair exchanges before committing to a barter.

Support Community Building and Networking:

Enable users to connect with others based on shared interests, goals, and skills, fostering a sense of community and collaboration . Encourage the formation of groups or networks within the platform for users with similar skills or learning objectives.

Ensure Data Privacy and Security:

Implement robust data protection measures to safeguard user information and communications . Provide secure communication channels within the platform to protect users' privacy and maintain confidentiality during exchanges.

Offer Dispute Resolution and Support:

Establish a transparent dispute resolution process to address any conflicts or issues that arise during skill exchanges . Provide customer support and resources to assist users in navigating the platform and resolving any concerns.

Technical Details

The Skill Barter Platform is built using a modern web development stack to ensure a responsive, secure, and scalable application. Below are the technical components and tools used in the development of the platform:

Frontend

EJS (Embedded JavaScript Templates):

Used as the templating engine for rendering dynamic content on the server-side. EJS allows for seamless integration of JavaScript logic within HTML, making it easier to manage and display user data dynamically on the frontend.

CSS:

Custom CSS is used to style the platform, ensuring a clean, intuitive, and user-friendly interface. The use of custom CSS allows for flexibility in design, enabling the creation of a unique look and feel that aligns with the platform's branding.

Tailwind CSS:

Tailwind CSS is utilized as a utility-first CSS framework, enabling rapid UI development with pre-defined classes. Tailwind's responsive design features and utility-based approach significantly reduce the time required for styling, while ensuring consistency across the platform.

Backend

Express.js:

The backend is powered by Express.js, a fast, unopinionated, and minimalist web framework for Node.js. Express.js is used to handle routing, manage requests, and serve dynamic content generated by EJS. It also facilitates the integration of various middleware for enhanced functionality.

Cookie Parser Middleware:

Cookie Parser is used to parse and manage cookies, which are essential for handling session data and user authentication. It simplifies the process of reading and writing cookies in the HTTP requests, providing a streamlined way to manage user sessions.

JWT (JSON Web Token) Authentication:

JWT is implemented for secure authentication, ensuring that user data and interactions are protected. JWT tokens are issued upon successful login and are used to authenticate and authorize users in subsequent requests, providing a stateless, secure way to manage user sessions.

Mongoose:

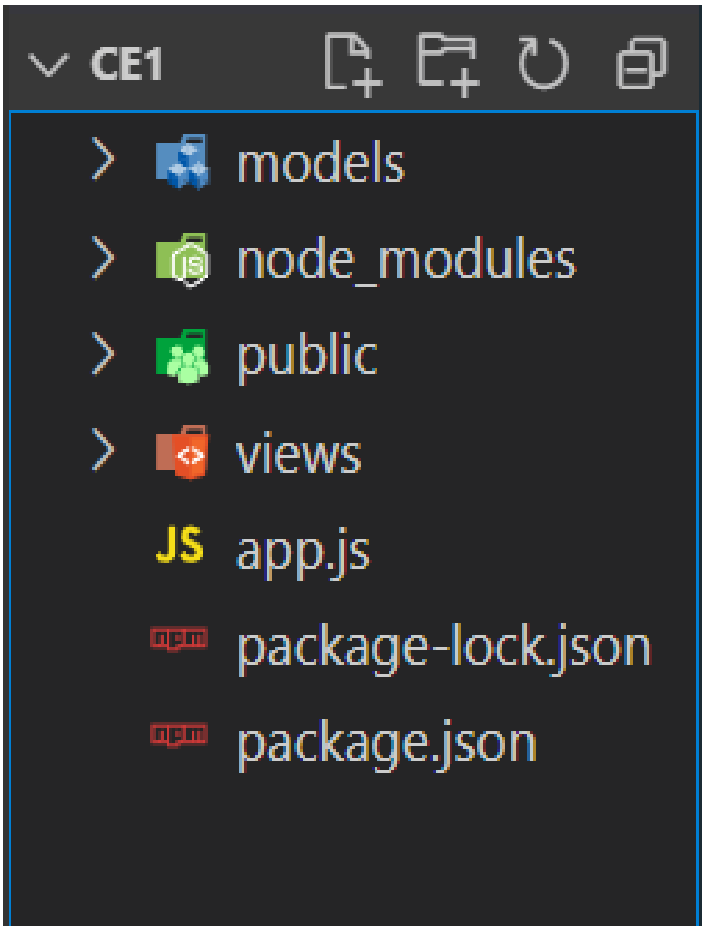
Mongoose is employed as the Object Data Modeling (ODM) library for MongoDB. It provides a schema-based solution to model the application data, making it easier to manage and enforce data structure within MongoDB. Mongoose also simplifies interactions with the database, including CRUD operations and data validation.

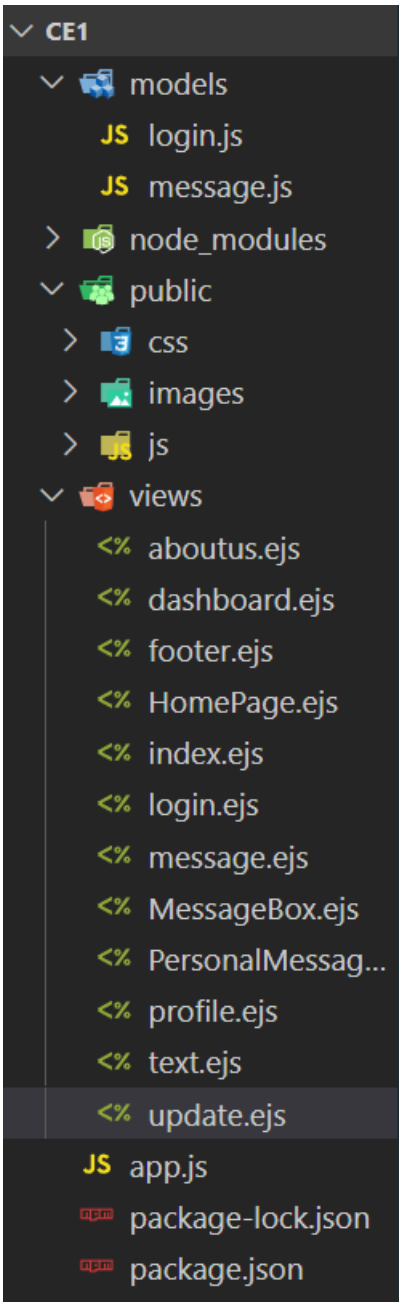
Database

MongoDB:

MongoDB is the chosen database for the Skill Barter Platform, providing a flexible, scalable, and document-oriented NoSQL database solution. MongoDB’s schema-less design allows for the storage of diverse and complex data structures, which is ideal for managing user profiles, skill listings, and exchange records.

File Structure





Code Snippets

Fronted

```
<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Skill Barter Platform</title>
    <script src="https://cdn.tailwindcss.com"></script>
    <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/6.0.0-bet
    <link rel="stylesheet" href="css/login.css">

    <style>···
    </style>
</head>

<body class="bg-gradient-to-r from-zinc-900 to-zinc-800 text-white">


    <nav class="flex items-center justify-between p-6 bg-zinc-800 shadow-md relative">···
    </nav>


    <main class="flex flex-col items-center justify-center min-h-screen text-center p-8 bg-cov
    </main>
```

```
    <section class="mt-3 p-8 bg-zinc-900 text-white">···
    </section>

    <section class="mt-16 p-8 bg-zinc-900 text-white">···
    </section>


    <footer class="bg-zinc-800 text-white py-6">···
    </footer>

</body>

</html>
```

```
 2  <html lang="en">
10  <body class="bg-gray-200 h-screen flex items-center justify-center">
23      <div class="bg-white p-8 rounded-lg shadow-md max-w-sm w-full text-center">
24          <div class="flex flex-col items-center">
26              <h2 class="text-2xl font-bold mb-2"><%= result.name %></h2>
27              <p class="text-gray-600 mb-4"><%= result.email %></p>
28              <div class="mb-4">
29                  <a href="<%= result.github %>" class="text-blue-500 hover:underline flex items
30              </div>
31              <div class="mb-6">
32                  <a href="<%= result.linkedin %>" class="text-blue-500 hover:underline flex ite
33              </div>
34          </div>
35          <div class="text-left mb-6">
36              <h3 class="text-xl font-semibold mb-2 text-center">Skills</h3>
37              <ul class="space-y-2">
38                  <li class="bg-gray-100 text-gray-700 font-bold py-2 rounded-md text-center"><%
39              </ul>
40          </div>
41          <button class="bg-red-600 hover:bg-red-700 text-white font-bold py-2 px-4 rounded focu
42              <a  href="/userdelete">Delete Account</a>
43          </button>
44      </div>
45  </body>
46  </html>
```

```
index.ejs > html > body
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Skill Barter System - Register</title>
    <script src="https://cdn.tailwindcss.com"></script>
    <style>…
    </style>
</head>
<body>
    <div class="w-full h-screen bg-gradient-to-r from-zinc-900 to-zinc-800 text-white flex ite
        <div class="p-5 w-full max-w-2xl bg-zinc-800 rounded-lg shadow-lg animate-slideIn">
            <h1 class="text-4xl my-5 font-semibold text-center">Join Skill Barter</h1>
            <p class="text-center mb-6">Create an account and start exchanging skills with oth
            <form action="/register" method="post" class="flex flex-wrap -mx-2">

                <div class="w-full md:w-1/2 px-2">
                    <input class="block  mt-3 w-full px-3 py-2 rounded-md bg-transparent borde
                </div>


                <div class="w-full md:w-1/2 px-2">
                    <input class="block  mt-3 w-full px-3 py-2 rounded-md bg-transparent borde
                </div>
```

```
<html lang="en">
<head>
    <script src="https://cdn.tailwindcss.com"></script>
    <link rel="stylesheet" href="../public/css/login.css">
    <style>…
    </style>
</head>
<body>
    <div class="w-full h-screen bg-gradient-to-r from-zinc-900 to-zinc-800 text-white flex ite
        <div class="p-10 w-full max-w-md bg-zinc-800 rounded-lg shadow-lg animate-slideIn">
            <h1 class="text-4xl my-10 font-semibold text-center">Welcome to Skill Barter</h1>
            <p class="text-center mb-6">Exchange skills, learn, and grow together. Login to ex
            <form action="/login" method="post">

                <input class="block  mt-3 w-full px-3 py-2 rounded-md bg-transparent border-2


                <input class="block  mt-3 w-full px-3 py-2 rounded-md bg-transparent border-2


                <input class="block mt-5 px-5 py-3 bg-blue-500 rounded-md hover:bg-blue-600 cu
            </form>
            <a class="mt-10 text-blue-500 block text-center hover:underline" href="/Register">
        </div>
    </div>
</body>
```

```
<!DOCTYPE html>
<html lang="en">
<head>…
</head>
<body>

    <div class="absolute top-4 left-4">…
    </div>


    <div class="absolute top-4 right-4">
        <button class="bg-green-500 hover:bg-green-600 text-white font-bold py-2 px-4 rounded
            <a href="/home">Home</a>
        </button>
    </div>

    <% if(usermessage.length>0){%>
    <div class="messages-container">
        <% usermessage.forEach(item=>{%>
        <div class="message">
            <strong class="username text"><%= item %></strong>
        </div>
        <% })%>
    </div>
    <% } %>
```

Backend

```
const express = require("express")
const app=express()

const loginModel = require("./models/login")
const messageModel = require("./models/message")

const cookieparser = require("cookie-parser");
const path = require('path');

app.use(cookieparser());
app.use(express.json())
app.use(express.urlencoded({extended:true}))
app.use(express.static(path.join(__dirname,'public')))

const bcrypt=require("bcrypt")
const jwt=require("jsonwebtoken");
// const user = require("./models/user");


app.set('view engine','ejs');

app.get("/",(req,res)=>{
    console.log(req.url);
    res.render('login')
})
```

```
app.post('/login',async(req,res)=>{
    let {username,password}=req.body;

    console.log(req.body)

    let user=await loginModel.findOne({username})
    if(!user){
        return res.status(500).send("Email or password is invalid")

    }


    bcrypt.compare(password,user.password,(err,result)=>{
        if(result){

            let token=jwt.sign({username:username,userid:user._id},'secret-key');
            res.cookie('token',token)
            res.status(200).redirect('/home');

        }
        else{
            res.send('Email or Password is Incorrect')
        }
    })
})
```

```
app.post("/register",async (req,res)=>{
    console.log(req.body)
    let {name,username,skill,email,github,linkedin,password}=req.body

    let user=await loginModel.findOne({username})
    if(user){ ···
    }

    const saltRounds=10;
    bcrypt.genSalt(saltRounds,(err,salt)=>{
        bcrypt.hash(password,salt,async (err,hash)=>{
            let user = await loginModel.create({
                name,
                username,
                skill,
                email,
                github,
                linkedin,
                password:hash
            })

            let token=jwt.sign({username:username,userid:user._id},'secret-key');
            res.cookie('token',token)
            res.render('HomePage',{name:name})
        })
    })
```

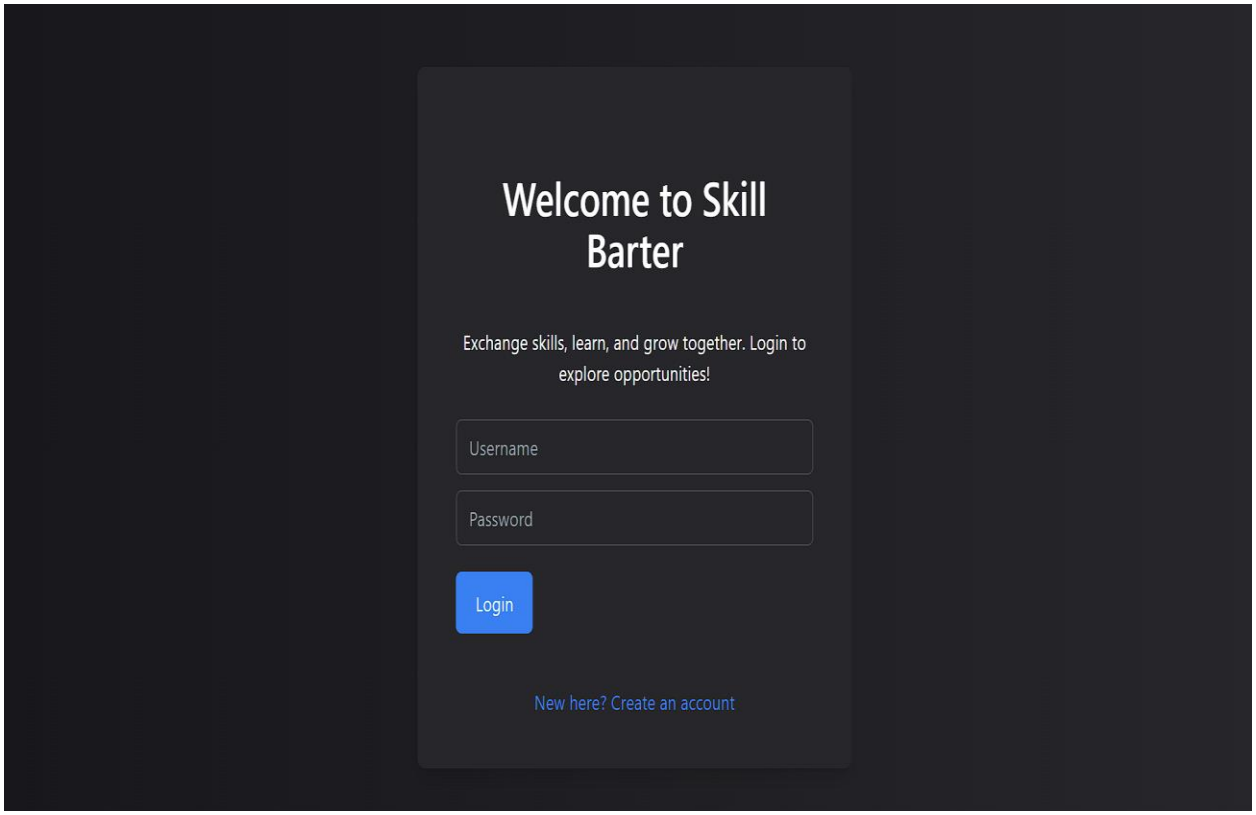
```
app.get("/",(req,res)=>{ ···
})

app.post("/register",async (req,res)=>{ ···
})

app.get("/Register",(req,res)=>{ ···
})

app.get("/footer",(req,res)=>{ ···
})

app.post("/footer",async (req,res)=>{ ···
})

app.get("/message",(req,res)=>{ ···
})

app.get("/PersonalMessages",isLoggedIn,async (req,res)=>{ ···
})

app.post('/login',async(req,res)=>{ ···
})

app.get("/aboutus",(req,res)=>{ ···
})
```

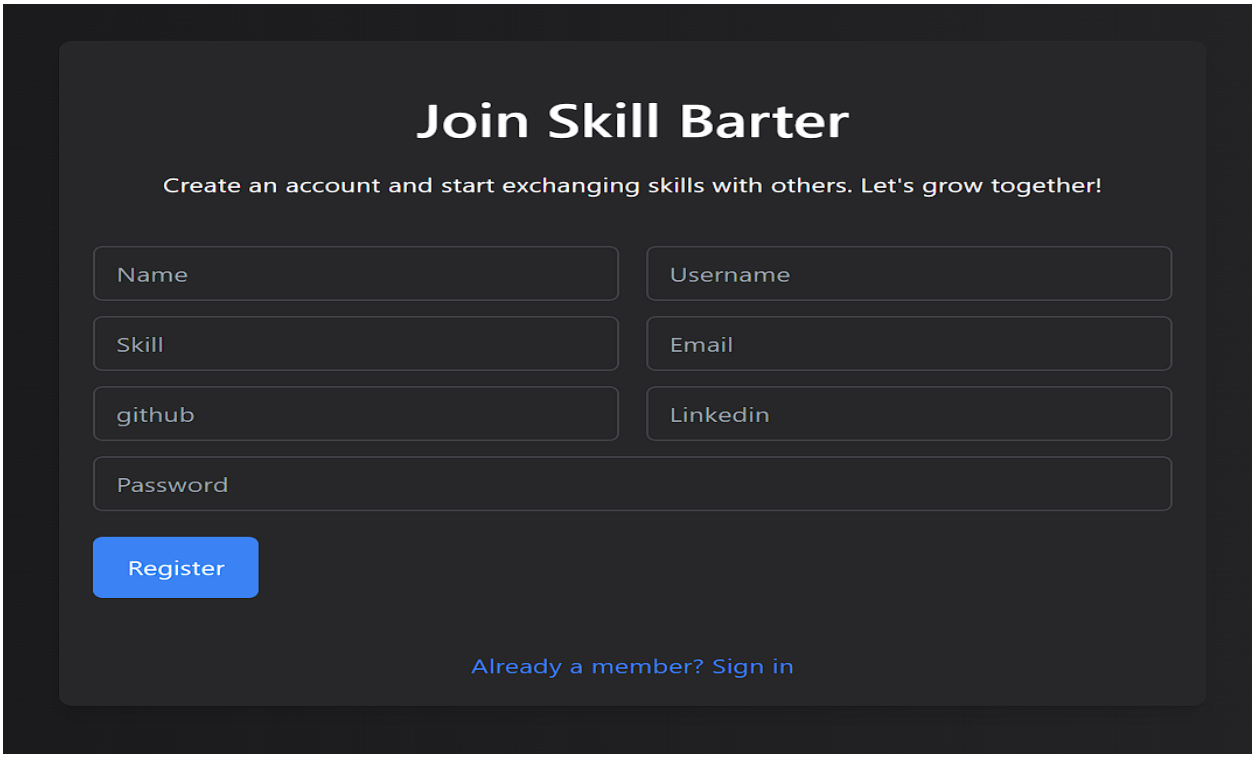
Database

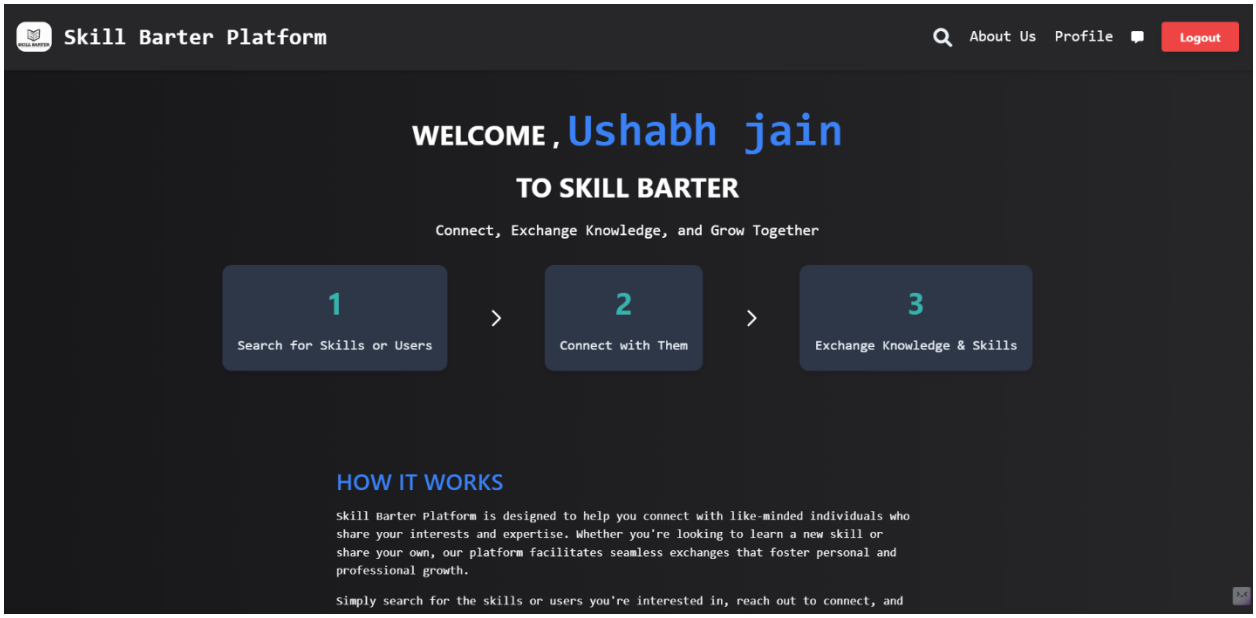
```
const mongoose = require("mongoose");

mongoose.connect('mongodb://127.0.0.1:27017/CE1')

const loginSchema = mongoose.Schema({
        name: {
            type: String,
        },
        username: {
            type: String,
        },
        skill: {
            type: String,
        },
        email: {
            type: String,
        },
        github: {
            type: String,
        },
        linkedin: {
            type: String,
        },
        password: {
            type: String,
```

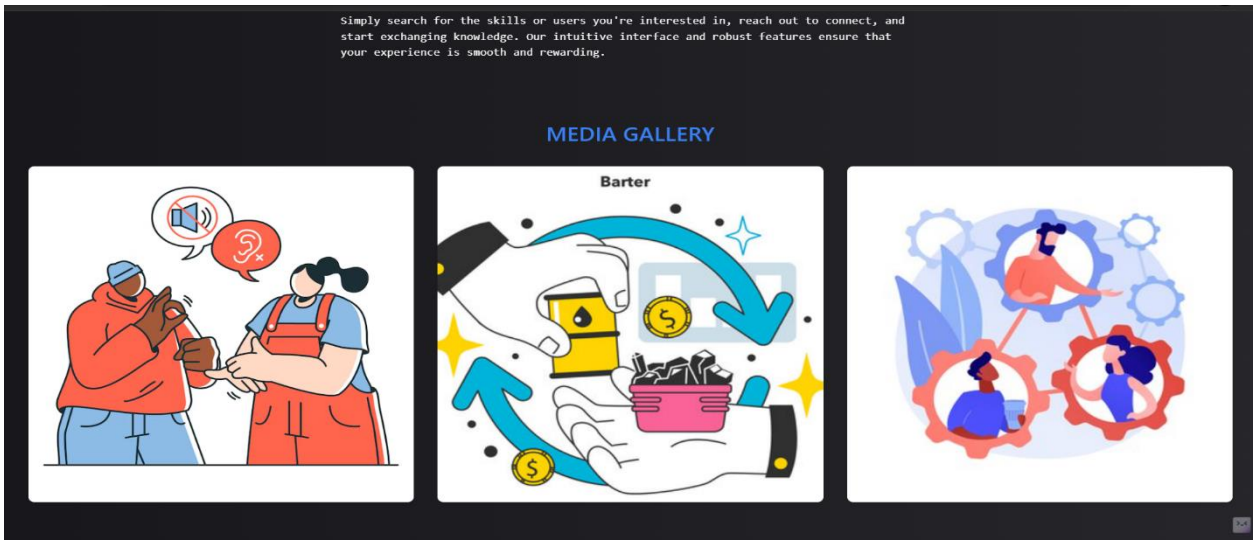
```
const mongoose = require("mongoose");

const messageSchema = mongoose.Schema({
    to: { type: String,required: true },
    from: { type: String, required: true },
    content: [{
        type:String
    }],
    lastUpdated: { type: Date, default: Date.now }
})

module.exports=mongoose.model('messages',messageSchema);
```

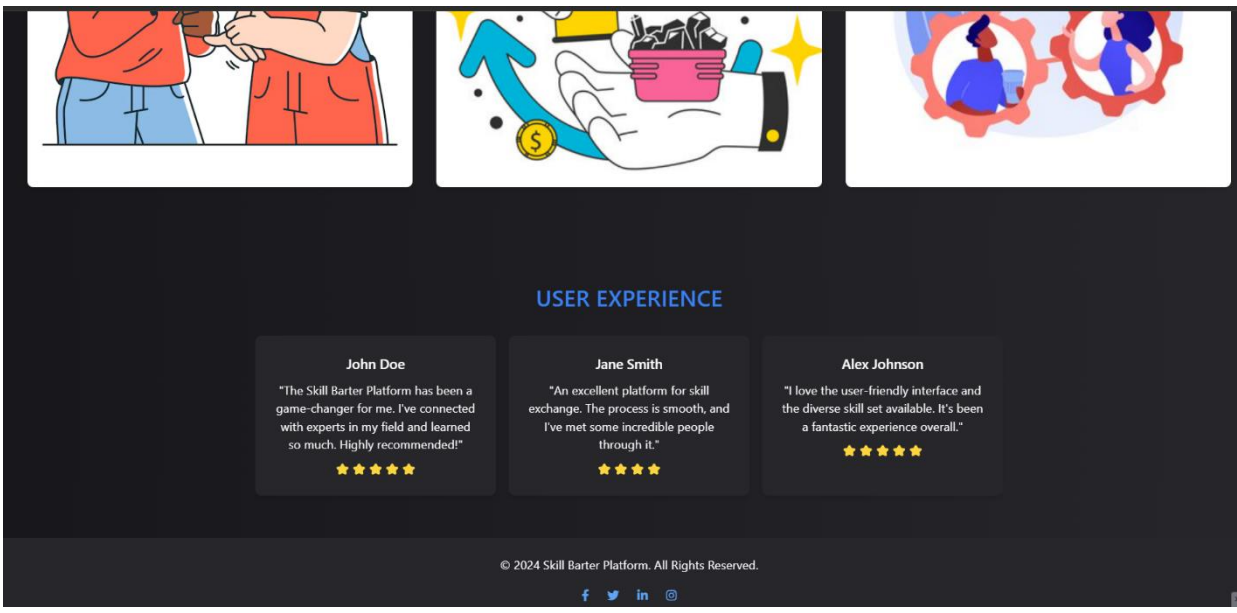
Result


Welcome to Skill Barter
Exchange skills, learn, and grow together. Login to explore opportunities!
Username
Password
Login
New here? Create an account


Join Skill Barter
Create an account and start exchanging skills with others. Let's grow together!
Name
Username
Skill
Email
github
Linkedin
Password
Register
Already a member? Sign in


Skill Barter Platform
About Us
Profile
Logout
WELCOME , Ushabh jain
TO SKILL BARTER
Connect, Exchange Knowledge, and Grow Together
1
Search for Skills or Users
2
Connect with Them
3
Exchange Knowledge & Skills
HOW IT WORKS
Skill Barter Platform is designed to help you connect with like-minded individuals who share your interests and expertise. Whether you're looking to learn a new skill or share your own, our platform facilitates seamless exchanges that foster personal and professional growth.
Simply search for the skills or users you're interested in, reach out to connect, and


Simply search for the skills or users you're interested in, reach out to connect, and start exchanging knowledge. Our intuitive interface and robust features ensure that your experience is smooth and rewarding.
MEDIA GALLERY
Barter


USER EXPERIENCE
John Doe
"The Skill Barter Platform has been a game-changer for me. I've connected with experts in my field and learned so much. Highly recommended!"
Jane Smith
"An excellent platform for skill exchange. The process is smooth, and I've met some incredible people through it."
Alex Johnson
"I love the user-friendly interface and the diverse skill set available. It's been a fantastic experience overall."
© 2024 Skill Barter Platform. All Rights Reserved.

Back

About Us

Empowering Skill Development Through Online Connections

Our Mission

We believe in the power of community and connection. Our mission is to provide a platform where people from all walks of life can meet online, share their knowledge, and learn new skills together. Whether you're a seasoned professional or just starting out, there's always something new to learn when you connect with others.

Our Values

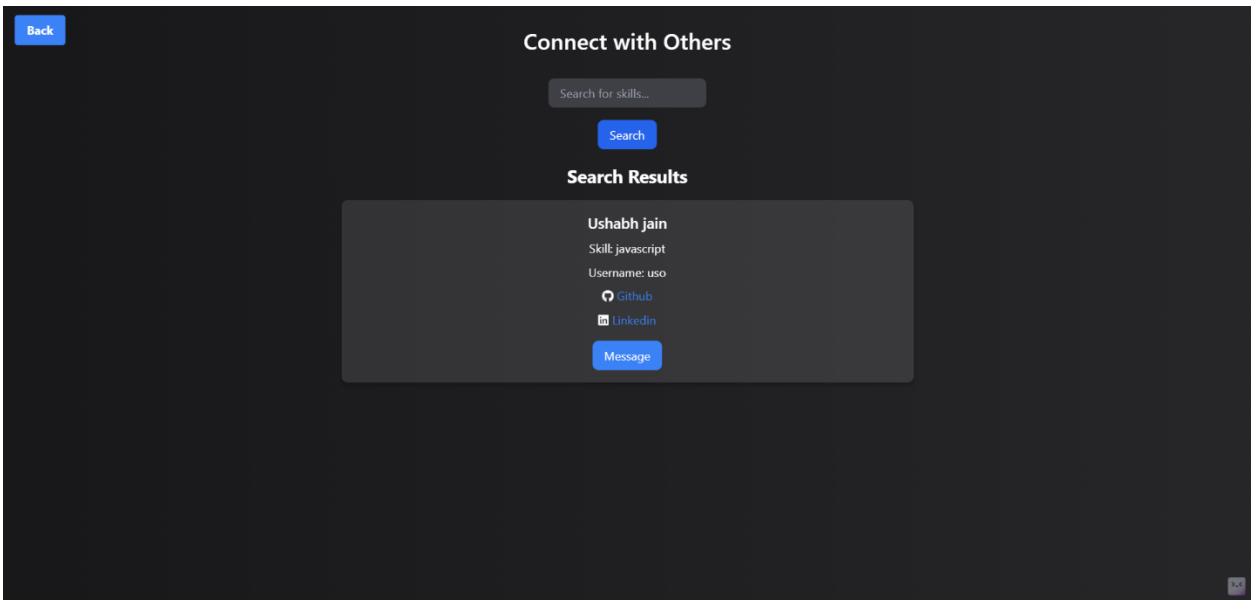
Back

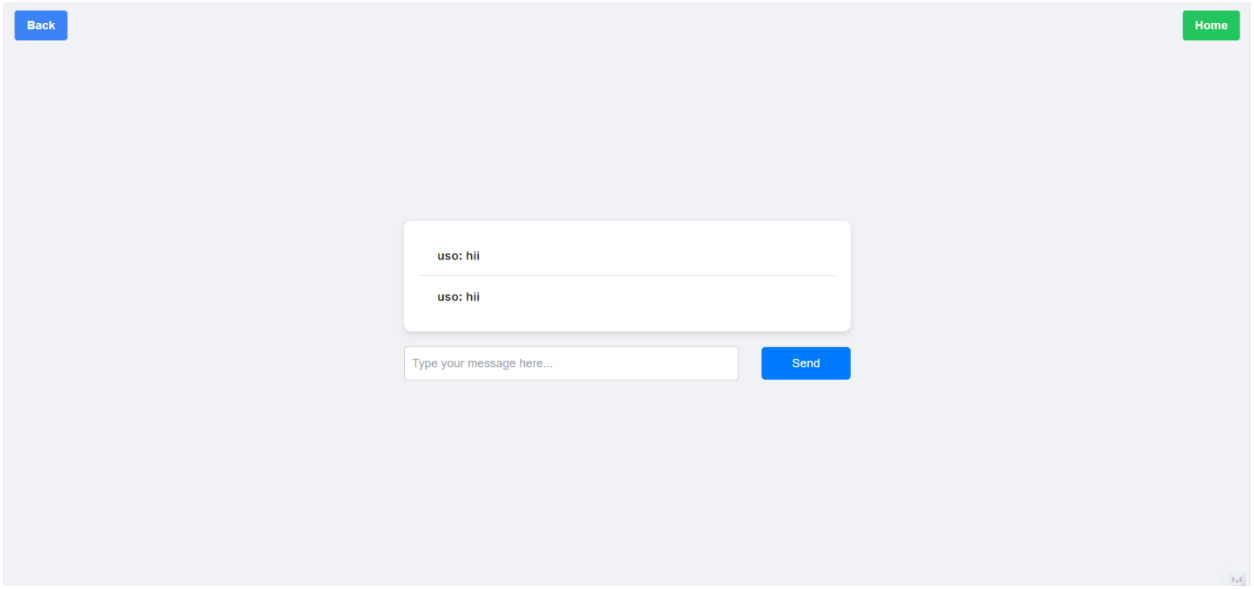
Your Messages

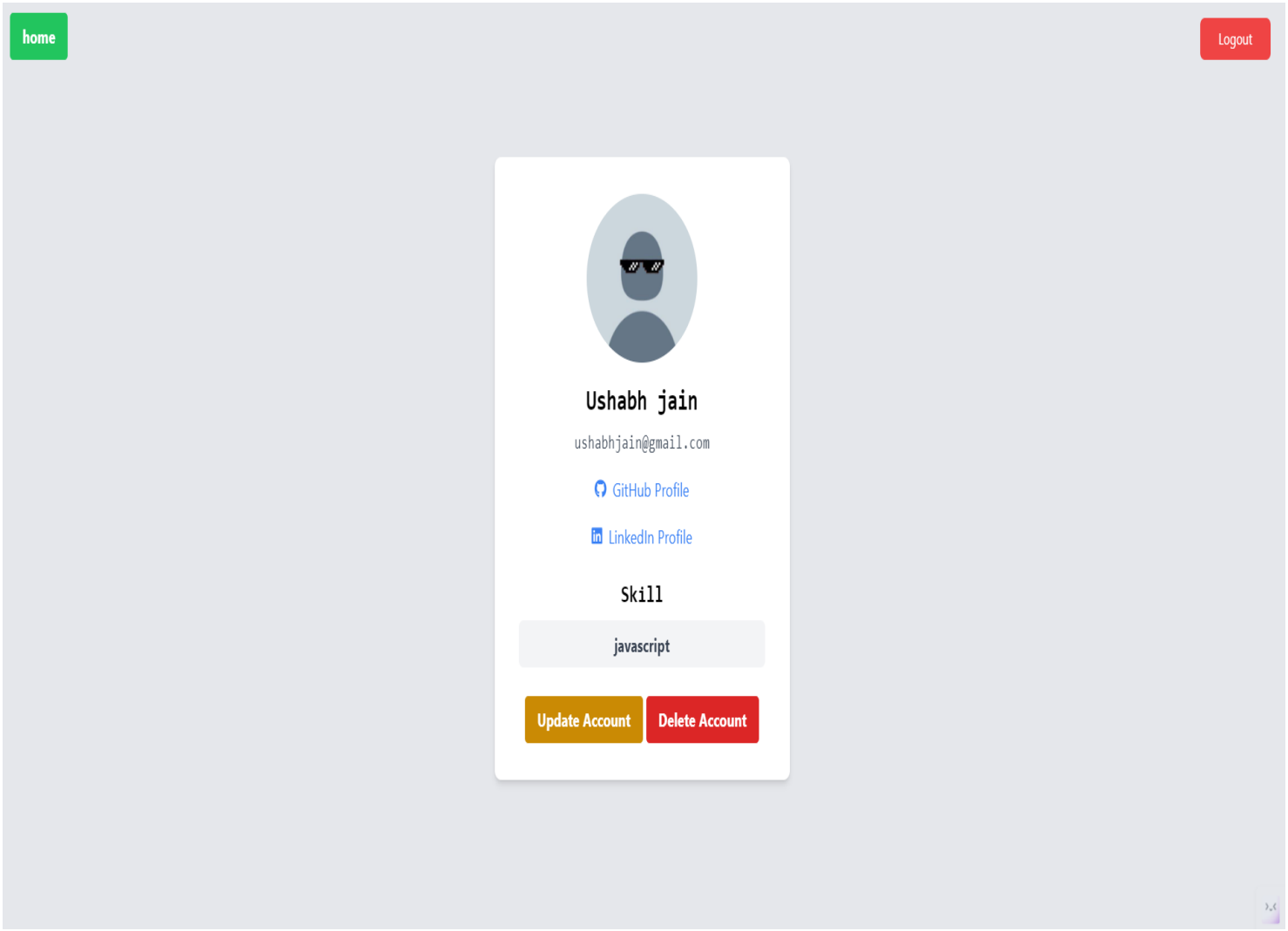
UserID: uso

Show Conversation

Delete


Back
Connect with Others
Search for skills...
Search
Search Results
Ushabh jain
Skill: javascript
Username: uso
Github
Linkedin
Message


Back
Home
uso: hii
uso: hii
Type your message here...
Send


home
Logout
Ushabh jain
ushabhjain@gmail.com
GitHub Profile
LinkedIn Profile
Skill
javascript
Update Account
Delete Account

References:

1. Tailwind :- https://tailwindcss.com/
2. MDN :- https://developer.mozilla.org/en-US/
3. Google Font :- https://fonts.google.com/
4. EJS :- https://ejs.co/
5. Express :- :- https://expressjs.com/
6. Node :- https://nodejs.org/
7. MongoDB:- https://www.mongodb.com/