FINAL ASSIGNMENT

By Kgadi Selepe and Musawenkosi Nzuza

WITS PROMATHS - December Codecamp

🌼 Final Assignment: Portfolio Website + Mini Chat Application

Congratulations on reaching the final stage of your HTML, CSS, and now JavaScript learning journey!
This final assignment will assess the practical skills you've learned. It is split into **two main sections**, each focusing on different skills.

Completing this assignment is required if you want to earn your **Certificate of Completion**.

✅ SECTION 1 : Portfolio Website (50%)

This section is COMPULSORY for everyone. You *must* submit this to get your certificate.

You will create a **professional personal portfolio website** using:

- **HTML**
- **CSS**
- JavaScript *(optional)*
- GitHub Pages **for deployment**

🚫 **Important rule: The code must be your own work.**

If your code is detected to be written by an AI/LLM, **you will forfeit this section and will NOT receive a certificate.**

⭐ What Your Portfolio Website Must Include:

Your website must clearly show **who you are**, your **goals**, and **how to contact you**.

Your website must have:

1. **Your image (a profile picture)**
2. **Your name**
3. **The degree / program you are studying**
4. **About Me section**
   - Who you are
   - What you are currently working on
   - Your career goals
5. **Skills section**

6. **Work experience** (if you have none, include activities or projects, even this coding bootcamp is something you can add)
7. **Education**
8. **Contact information**
9. **Links to your social media profiles** (LinkedIn, GitHub, etc.)

⭐ Expectations:

- The website must be **professionally styled**, clean, well-designed, and easy to navigate.
- You must use **HTML and CSS properly**, with clean structure and layout.
- The website must be deployed live using **GitHub Pages** so that anyone can visit it.

This section is worth **50% of the total assignment mark**.

✅ SECTION 2 : Mini Communication/Chat App (50%)

In this section, you will build a simple messaging application.

You will use:

- **HTML**
- **CSS**
- **JavaScript**
- GitHub Pages **for deployment**

I will provide the **backend**, so you do **not** need server-side code. You will only interact with the backend through API endpoints.

SECTION 2 is divided into two parts:

📌 Part A : User Interface (UI) (25% of total mark)

You must build the following pages:

1. Signup page

Users enter:

- Username
- Password

- Confirm Password

Your form must:

- Capture input values
- Send them to the backend signup endpoint
- Do basic validation (password length must be ≥ 8)

---

2. Login page

Users enter:

- Username
- Password

Your form must:

- Capture the values
- Send them to the backend login endpoint
- Redirect to the chat page on success

---

3. Chat Page / Messages Page

This page must contain:

- A section that **displays messages retrieved from the database**
- A text box where the user can **type a message**
- A **Send button**
- A **Logout button** that simply redirects back to the login page

The UI (HTML + CSS) for all these pages counts **25%**.

📌 Part B : JavaScript Logic (25% of total mark)

You will write the JavaScript that makes the app work using the endpoints I will give you.

You must write JS code to:

1. **Send signup data** → to backend signup endpoint
2. **Send login data** → to backend login endpoint
3. **Display messages from the database**
   - Use the "get messages" endpoint
   - Use loops to display each message in the chat area

4. **Send new messages** → to the backend
5. **Handle basic validation**
   o Check if password length is at least 8
6. **Use the DOM**
   o Read input values
   o Update the page
   o Display messages
7. **Use click listeners**
   o For the Send button
   o For Login
   o For Signup
   o For Logout

⭐ **Optional Features (5% within the JS section)**

These are optional and give you extra marks:

- **Edit your own message**
- **Delete your own message**

These use the optional endpoints I will provide.

📘 What You Need to Know to Do Section 2

You must understand the following basic JavaScript concepts:

✔ Variables

Storing values:

```
let username = "student1";
```

✔ If statements

Checking conditions:

```
if (password.length < 8) {
  alert("Password too short");
}
```

✔ Loops

Displaying messages:

```
messages.forEach(msg => {
  // display message
});
```

✔ DOM Manipulation

- Reading input values
- Creating HTML elements
- Updating the page
- Adding messages to the chat area

✔ Event Listeners

```
button.addEventListener("click", function() {
  // do something
});
```

✔ Calling APIs using `fetch()`

Sending and getting data:

```
fetch("https://example.com/signup", {
  method: "POST",
  headers: {"Content-Type": "application/json"},
  body: JSON.stringify({ username, password })
})
```

If you understand these basics, you will be able to complete the assignment.

🎥 Additional Learning Support for Your Final Assignment

To help you succeed in your Final Assignment, I will be releasing a series of **step-by-step video tutorials** and **extra learning material**. These resources are designed to guide you through all the skills you'll need to complete both parts of the project confidently.

You are beginners, and I want to make sure you understand not just *what* to do, but *how* and *why* each step works.

🔳 What the Video Lessons Will Cover

The videos will teach you exactly the skills required for the Chat App section and the Portfolio section. They will include:

✔ 1. How to interact with APIs (Backend Endpoints)

You will learn how to:

- Use fetch() to send data to a server

- Use POST requests for signup and login

- Use GET requests to load messages

- Read responses from the backend

- Handle success and error situations

This will help you with signing up, logging in, and displaying messages.

✔ 2. DOM Manipulation with JavaScript

You will learn how to:

- Select elements from your page using document.querySelector

- Read values from input fields

- Create new elements using JavaScript

- Add messages to the page using loops

- Change the content on the page dynamically

These skills are essential for showing messages in your chat window.

✔ 3. Creating, Styling, and Validating Forms

You will learn how to:

- Build signup and login forms

- Style them with CSS

- Capture form input using JavaScript

- Apply basic validation (e.g., checking password length)

This prepares you for the UI portion of Section 2.

✔ 4. Using Loops to Display Information

You will learn how to:

- Loop through an array of messages

- Create an element for each message

- Display messages in the chat area

This is required for showing messages from the database.

✔ 5. Using If Statements to Make Decisions

You will learn how to:

- Check if passwords are long enough

- Check if fields are empty

- Show alerts or messages on the page

- Handle login success or failure

This is needed for all validation and basic logic.

📘 Supplementary Learning Materials

Along with the videos, I will also post:

- Short written explanations of important concepts

- Quick reference guides (HTML tags, CSS selectors, JS basics)

- Example snippets (not for copying, but for learning)

- Practice exercises for JavaScript basics

- Troubleshooting guides (common errors and how to fix them)

These will help you learn faster and build confidence.

⭐ How These Resources Help You

These materials are meant to **guide**, not replace your own work.

They will help you understand:

- How HTML, CSS, and JavaScript work together

- How to build real features from scratch

- How to structure your code

- How to debug your work

- How APIs communicate with your website

Remember: **Your final code must be your own** , the videos show you the concepts and examples, but the final implementation should be written by you.

🎉 Final Notes

- This assignment is meant to show your **real coding ability**, not AI generated work.

- Everything should be neat, readable, and understandable.

- Ask questions if you get stuck , learning is the goal.

I will post more details next week, this document is just to show you what will be expected of you at the end. The later document will include all details and endpoints. This is to help you prep for the final assignment. Yes I used AI to compile this document.