Software Development Skills Front-End

Learning Diary

By: Beenish Nazam
(002327354)

Date: 23-4-2025

I enrolled in the course by entering the enrollment key. After enrollment I read the general instructions about the course like what I'll learn during the course and what are the assignments that I must do to pass the course. I had few questions related to the course; to get the answers to those questions I booked a meeting with Mr. Majad Qureshi.

Date: 28-4-2025

Today is the meeting day. I joined the meet with Zoom at 11 AM. I asked questions that I had in my mind and got the answers. Now, everything was clear about the course.

Date: 10-06-2025

I started the course by setting up my environment. I chose VS code to write the code. I created a repository in GitHub named Frontend. After environment setup, I started watching video lectures. I watched lecture 1 and learn about HTML, history of HTML, elements used in HTML and different tags used in HTML like <p>, <b>, <img>, <a>, <h1>, <emp>. I also learned about different types of lists that can be used, and I learned about table creation. I already knew HTML, so it was like a revision for me. I had some experience with HTML but still I practice different tags in VS code. I cloned the repository and added a folder coursework and pushed the code the code into GitHub repository.

Date: 11-06-2025

Today I watched lecture no 2. The lecture was about tags <header>, <nav>, <main>, <article>, <footer> and <form> tag. <header> is a tag that we can use to create a header. The header will be displayed on all the pages of the website. I learned that <nav> tag can be used to navigate the website because we can add links of different webpages through this tag. <main> tag contains the actual content of the webpage and <footer> is used to add footer mostly at the end of the webpage. <form> tag helps us to create different types of forms. After learning about these tags, I practiced all these tags and push the code into repository.

Date: 14-06-2025

Today I watched lecture no 3 and I learned about JavaScript and its history. JavaScript is similar to C languages. We can use var, let or const to define a variable in JavaScript. Different operators available in JavaScript are arithmetic, relational etc. I learned about arrays and methods used to add and remove elements from an array. I learned about document object model which is hierarchical model. I learned that javascript used different DOM specific methods to access the DOM objects. The most common method is getElementByid(ID). After learning all these concepts I practiced these in VS code.

Date: 15-06-2025

Today I watched the lecture video 4 which was about CSS. I learned the history of CSS and about its version. I learned that CSS3 is still underdeveloped. CSS can be used to style the HTML elements. There are three ways that be used to add CSS in the HTML. Different properties like Background-color, color, font-family etc. can be used to style the elements of HTML. CSS used box model for every element. We can specify height, width, padding, margins and borders according to our need. If we want to access a specific element, we can use class name defined in the HTML to access that element. Besides classes id could also be used for same purpose. After Learning all this, I practiced and push the code in GitHub repository.

I also started my project for the course which is the blood bank system. I started creating necessary HTML webpages.

Date: 16-06-2025

During the checking about the course, I found a video where a whole website was created from scratch. So, I plan to follow the video to create my website for project. I started working on styling the header. I added a logo image and a title using a and styled them using Flexbox to make them appear side by side. I used CSS properties like background-color, padding, and font-size to make the header look better. I also made sure the logo image had a fixed height and some space between the logo and the title. I used a navigation bar with links and styled the text using font-weight, font-size, and removed the default underline with text-decoration: none.

After designing the header, I started working on the main section of the homepage, which is called the hero section. This part is important because it gives users a clear message and tells them what action to take. I added an image on one side and some text with a button on the other side. The text includes a heading that says "Donate Blood, Save Lives", a short sentence, and a button labelled "Become a Donor". I used Flexbox to place the image and the text side by side.

For styling, I made sure the image fits properly and has smooth corners. The button is styled in red with white text, so it stands out. I used utility classes like padding and text-align to arrange the items. I also added media queries to make sure the layout adjusts nicely on smaller screens. On mobile devices, the image and the text appear one below the other to fit better.

From this task, I learned how to use Flexbox to create a responsive layout and how to organize a clean and simple homepage.

After that, I created the About Us page to explain the mission, vision, and values. I copied the code of header and paste it in this page. I used clear headings, paragraph text, and a simple list to keep the content easy to read. In CSS, I styled the page with a centered layout, added spacing, adjusted font sizes, and used dark red color for headings to match the blood donation theme.

Date: 17-06-202

Today I designed a "Become a Donor" a registration form for collecting donor information like name, age, gender, and blood group. page that allows users to fill out a simple form with their personal and blood donation details and for this purpose I used <form> tag. I tried to keep the layout clean and user-friendly by using flexbox to arrange the form fields neatly. I styled the form with soft colors, rounded input fields, and a bold red submit button that changes shade on hover to make it more interactive. I also added responsive CSS so the page adjusts well on smaller screens.

After this, I made a Donor List page for my project. This page shows a list of people who are blood donors. I used HTML for the structure and CSS to make the page look nice.

Each donor is shown in a card. The card has the donor's name, blood group, city, and phone number. I used red colour because it matches the blood donation theme. I also added a small effect when someone moves the mouse over a card, it becomes a little bigger.

The cards are inside a container, and I used Flexbox to place them side by side. The layout also works well on mobile phones. I used a media query to make sure the page looks good on small screens.

Date: 18-06-2025

Today, I made a simple web page called the Blood Stock page. This page shows how much blood is available for each blood group, like A+, B-, O+, and others. I added a slider to show a message for users. In the middle of the page, I showed cards for each blood type. Each card has a blood group name and the number of units available. I used red and white colours to match the blood bank theme. When you move your mouse over a card, it becomes a little bigger.

I also made sure the page works well on small screens like phones. I used grid layout and responsive design to do that.

The project is completed. Now it's time to deploy the website on GitHub pages. I opened GitHub Desktop, selected my local project folder at:
C:\Users\HP\project. I committed and pushed the project to GitHub.
Then I went to Settings > Pages in the repository to publish it using GitHub Pages, where I select main branch and root director then saved.

Problem I faced:

The website didn't load correctly, only a blank page appeared. Then I took help from ChatGPT about why I'm facing is problem. Then I got answer that my actual HTML and CSS files were inside a subfolder named project, not in the root of the repository. GitHub Pages only serves files properly if they are in the root directory. To fix this problem, I copied all the HTML, CSS, and image files from project/ into the root of the repository folder. Then I committed the changes using GitHub Desktop and pushed them to GitHub again. After the push, I reloaded the GitHub Pages link. Now the website was live and working.

Then I checked the live website, for desktop view it was working fine. But I opened the link in mobile for bloodstock page, the cards were not displayed properly. So, I made some changes in CSS file to make it better for small screen. After these changes, I checked it again and now everything was fine and good.

This is the link of the website:

https://beenishnazam786.github.io/Frontend/

AI statement:

I acknowledge the use of ChatGPT to solve the problem the problem that I faced during the deployment phase.