



| Module | Portfolio | Assessment Type |
|------------------------------------|-----------|-------------------|
| Collaborative Development (5CS024) | 1 | Individual Report |

[Home work
collaboration space for
student and teachers -
back-end]

Student Id : 2329763
Student Name : Roshan Bohara
Section : L5CG8
Group : Group 8
Module Leader : Biraj Dulal
Lecturer :Prakash Shrestha
Submitted on :4/29/2024

Acknowledgement

I did like to thank the University of Wolverhampton and my college Herald College where I study me to give chance to compile this report. Also, I want to thank my mentors and my friends whose valuable contents served as a best reference for this project development. Lastly, my appreciation goes to everyone who helped me to complete this project sprint.

Contents

| | |
|--|----|
| Self-appraisal form (fill one in for each sprint) | 1 |
| Personal objectives – performance measurement | 1 |
| Evidence of good collaboration | 2 |
| Good communication and file sharing..... | 2 |
| Continuing Personal Development (CPD) | 3 |
| Appendix A..... | 3 |
| 1. Selecting the appropriate technology:..... | 3 |
| Back-end | 4 |
| React.js:..... | 5 |
| HTML..... | 5 |
| CSS | 5 |
| JavaScript..... | 6 |
| 2. Implementing the functional requirement | 7 |
| 2.1 code and their explanation..... | 21 |
| 2.2 uses of version control | 25 |
| Appendix B..... | 26 |
| 1. Evidence of Good communication and file sharing | 26 |
| 2. Evidence of Continuing Personal Development (CPD) | 27 |
| References..... | 33 |

| | |
|---|----|
| Figure 1: Code for making authentication page | 7 |
| Figure 2: Code for render urls..... | 8 |
| Figure 3: Django manage.py..... | 9 |
| Figure 4: URL patterns | 10 |
| Figure 5: Git logs | 11 |
| Figure 6: Discussion of issue | 12 |
| Figure 7: Discussion on integrating..... | 12 |
| Figure 8:problem in integration. | 13 |
| Figure 9: got innsue on my pc..... | 13 |
| Figure 10: discussion on integrating. | 13 |
| Figure 11: Learning django framework | 15 |
| Figure 12: learning Intregation..... | 15 |
| Figure 13: Learning more on Django and react..... | 16 |

Self-appraisal form (fill one in for each sprint)

| | | | |
|-----------------|--|------|---------------|
| Student number | 2329763 | Name | Roshan bohara |
| Project | Homework Collaboration Space for Students and Teachers | Date | 4/27/2024 |
| Role | Backend | Team | 5 |
| Sprint (1 or 2) | Sprint 1 | | |

Personal objectives – performance measurement

| Objectives | Evidence provided. | Evaluation <i>Student / tutor</i> | |
|--------------------------------------|---|--|--|
| Selecting the appropriate technology | Being a backend developer, it s important to have knowledge on coding and various knowledge and have to select such most suitable ones for the projects. After we had a meeting with our team members, we decided to choose a technology that most of the industries had used and were familiar and experienced with that. The language is : Django (backend),HTML, CSS, react js, and jawa script. More of this I have explain on this section Appendix A: | 9 | |
| <i>Tutor feedback:</i> | | | |

| | | | |
|---|---|-----|-----|
| Implementin gfunction requirement | As a developer, ensuring the use or the functionality of the project is the most priority which includes design, construction and testing to meet such specific requirements. I've analyzed all the things to build the project and project design needs. During that I get a wealth of ideas and plannings or features for development of it. Screenshot is attached to this section Appendix A: | 9 | |
| <i>Tutor feedback:</i> | | | |
| Good collaboration | | 14 | |
| <i>Tutor feedback:</i> | | | |
| | | /40 | /40 |

Evidence of good collaboration

Good communication and file sharing

The most important things in any project is having a good communications among team members and seamless file sharing to ensure the success of a project. Using good communication also helps to communicate among team members and also can share files amongs them. Inadequate team communication can lead to various Issues such as misunderstanding, conflict between team members and can cause serious impact on the project. Therefore, Our team are using Google chat to streamline our interactions which helps to solve the rising issue.

Continuing Personal Development (CPD)

As a backend developer, engaging in Continuous Professional Development (CPD) is a never-ending journey of exploring evolving new technologies. It is important to have knowledge on latest advancements or uprisings in technology and also emerging technologies which big companies are using. This never-ending journey entails acquiring new skills through daily life coding practice, understanding frameworks and applying newfound expertise to projects. To improve my skills, I have watched various online

Issue tracking.

Getting the role of a developer comes with the responsibility to tackle any issue related to our project like finding bugs, defects and other issues that come during the process of project development. My PM and Business analyst assist me in reviewing my work and assigning me an issue. I start working on resolving it effectively, considering the time. After I successfully address the assigned issue, there were still some issues that were in the progressing stage.

1. Selecting the appropriate technology:

In the present context we are living on 21st century and many new technologies, new inventions are happening throughout the world and in the programming sector we must consider to select the specific new technology to perform the task which helps us to make the project faster or easy to build. Selecting the specific appropriate technology is the foundation to make good applications or webdevelopment. I have selected Django, CSS, HTML and React.js to perform the task. In the present context the artistry of technology selection offers the high security, easy to use and perfect for user experience.

Among various technology available technology in the market, I have chosen some of the best technology for my project. They are listed below:

- Django
- React.js
- HTML
- CSS
- JavaScript

Back-end

The terms back end refers to the person who is behind any software or applications who works as foundation that powers the functionality and logic behind it. My role involves architecting server-side systems and managing databases and ensuring seamless communication between the frontend and the server. In my project I have chosen emerging technologies like Django, which provides a comprehensive framework for backend development which has the best security, scalability and rapid development capabilities. I have used Django's powerful features such as ORM for database management and its best authentication system which meets the requirements of my project.

React.js:

React.js is an open-source JavaScript user interface toolkit that surfaced in 2013 (w3school, n.d.) and quickly gained popularity due to its amazing capabilities. One of its most notable features is that it can create reusable user interface components, a feature that developers love for its effectiveness and adaptability. React simplifies development processes by importing these necessary components and enabling the establishment of various elements such as pet listings, user profiles, and navigation menus. React.js further claims to capture logic and behavior, resulting in a codebase that is well-organized and structured. Its usage of a virtual DOM allows for quick user interface updates in reaction to modifications in the application's state, guaranteeing a smooth user experience. React.js does a thorough examination and updates the real DOM to improve overall performance.

HTML

Hypertext Markup Language (HTML) is used by React.js, a JavaScript package mostly used for creating user interfaces, to organise the web pages on which React components are shown. A popular syntactic extension that helps define component organisation is called JSX. It looks a lot like HTML. JSX makes it easier for developers to visualise and arrange different component parts, such lists (ULs), by enabling them to easily incorporate HTML-like code directly into JavaScript scripts. React components are created from HTML elements described in JSX and easily interact with other JavaScript features. A standardised collection of user-friendly, cross-platform compatible tags and attributes are provided by HTML.

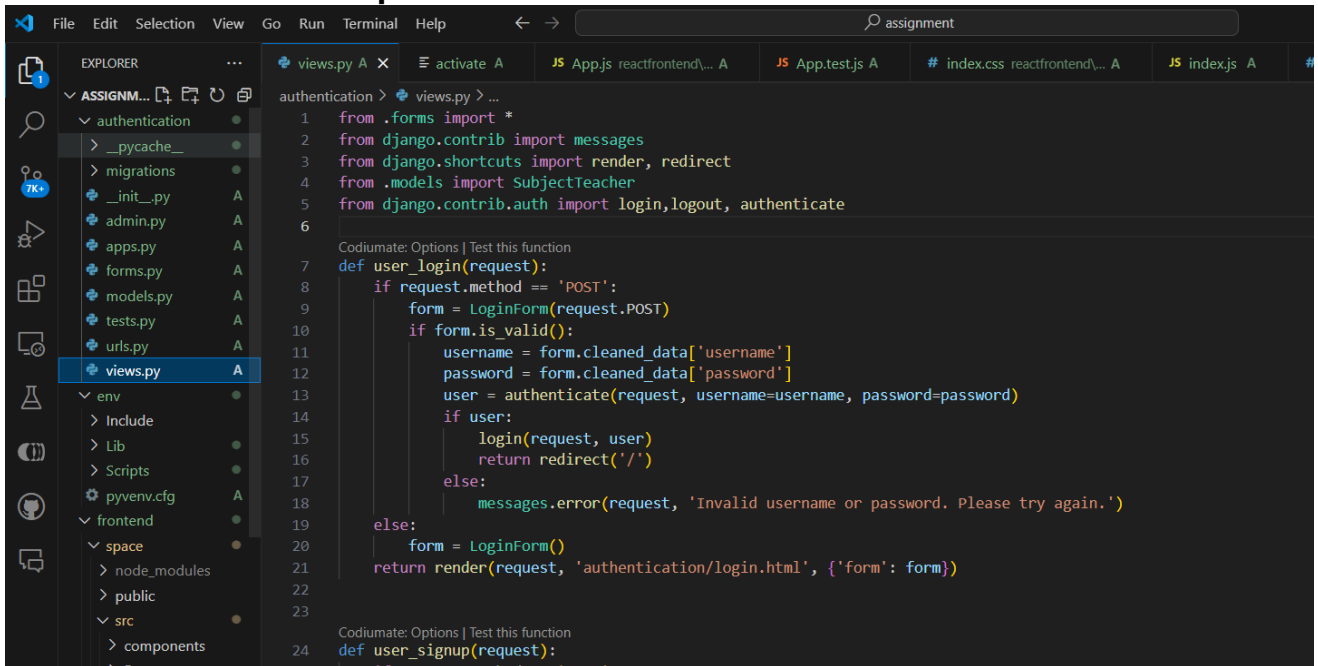
CSS

The main uses of CSS are for style elements including colour, font, borders, spacing, and overall UI component visual appearance. React.js, on the other hand, focuses primarily on the behaviour, logic, and structural makeup of components. There are several ways to add CSS styles to React.js components: external CSS files, styled-components, and inline styles. By utilising Cascading Style Sheets, external CSS files give developers and users the freedom to alter and apply styles throughout components. Widely praised for its adaptability and effectiveness, CSS ensures uniformity in design and makes it easier to turn creative ideas into aesthetically pleasing, intuitive web pages.

JavaScript

Built on top of JavaScript, React.js mainly depends on JavaScript's ability to give web pages movement and interaction in the user experience. JavaScript is a very flexible programming language that may be used for a wide range of tasks, including state management, user input validation, and server data retrieval. Nevertheless, difficulties like browser compatibility problems might occur, requiring developers to have a thorough grasp of the language as well as possible hazards. Because JavaScript is asynchronous, it can do non-blocking actions, which allows for smooth user experiences even while processing large amounts of data. Because of this feature, web pages are more engaging and responsive, which makes JavaScript a great option for creating dynamic interfaces.

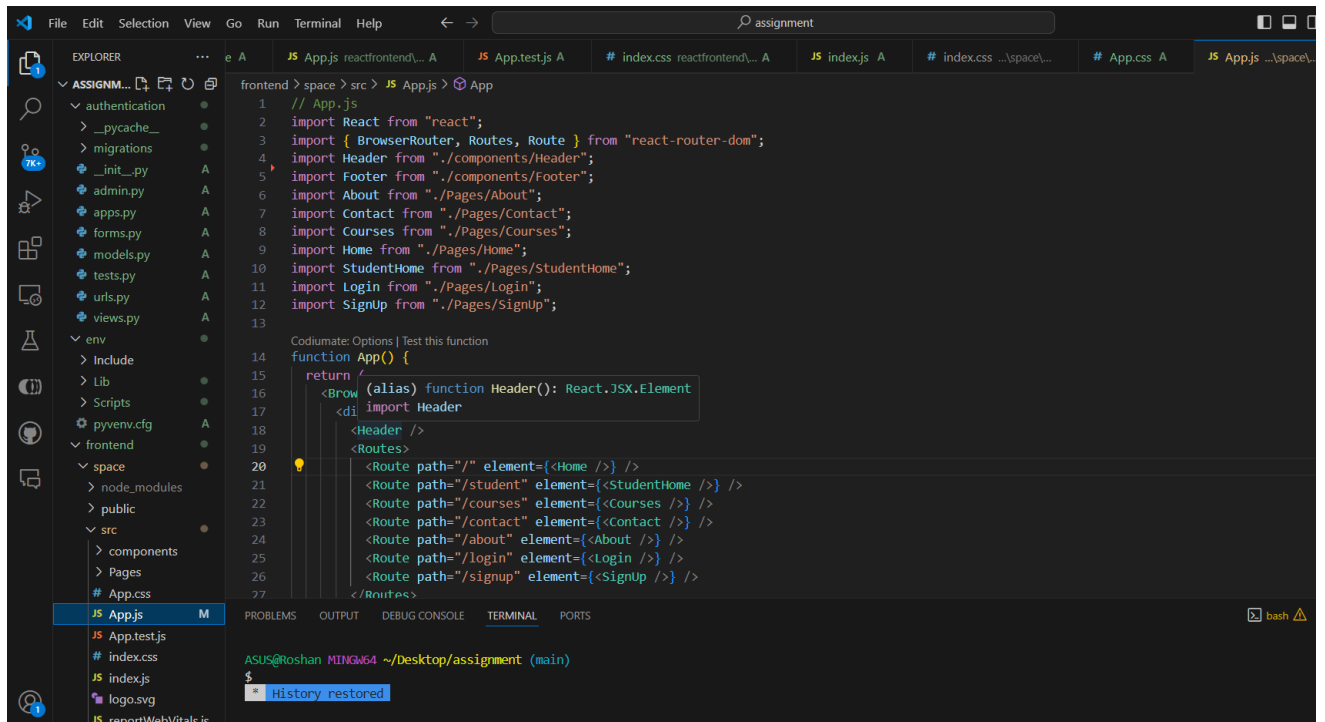
2.1 code and their explanation.

The image shows a screenshot of a code editor, likely Visual Studio Code, with a dark theme. The Explorer panel on the left shows a project structure with folders like 'authentication', 'env', 'frontend', and 'space'. The 'authentication' folder is expanded, showing files like '_pycache_', 'migrations', '_init_.py', 'admin.py', 'apps.py', 'forms.py', 'models.py', 'tests.py', 'urls.py', and 'views.py'. The 'views.py' file is selected and open in the main editor. The code in 'views.py' includes imports for Django forms, messages, shortcuts, and models. It defines two view functions: 'user_login' and 'user_signup'. The 'user_login' function checks for a POST request, validates the login form, authenticates the user, and redirects to the home page if successful. If authentication fails, it adds an error message and redirects back to the login page. The 'user_signup' function is partially visible at the bottom.

```
1 from .forms import *
2 from django.contrib import messages
3 from django.shortcuts import render, redirect
4 from .models import SubjectTeacher
5 from django.contrib.auth import login, logout, authenticate
6
7 def user_login(request):
8     if request.method == 'POST':
9         form = LoginForm(request.POST)
10        if form.is_valid():
11            username = form.cleaned_data['username']
12            password = form.cleaned_data['password']
13            user = authenticate(request, username=username, password=password)
14            if user:
15                login(request, user)
16                return redirect('/')
17            else:
18                messages.error(request, 'Invalid username or password. Please try again.')
19        else:
20            form = LoginForm()
21        return render(request, 'authentication/login.html', {'form': form})
22
23
24 def user_signup(request):
25     if request.method == 'POST':
```

Figure 1: Code for making authentication page

This Django view function handles user login. It checks if a login form was submitted (POST request) and validates the data. If valid, it retrieves username and password, authenticates the user, and logs them in with redirection (likely to the homepage) on success. If authentication fails, it adds an error message and redirects back to the login page.



```
1 // App.js
2 import React from "react";
3 import { BrowserRouter, Routes, Route } from "react-router-dom";
4 import Header from "../components/Header";
5 import Footer from "../components/Footer";
6 import About from "../Pages/About";
7 import Contact from "../Pages/Contact";
8 import Courses from "../Pages/Courses";
9 import Home from "../Pages/Home";
10 import StudentHome from "../Pages/StudentHome";
11 import Login from "../Pages/Login";
12 import SignUp from "../Pages/SignUp";
13
14 Codiumate: Options | Test this function
15 function App() {
16   return (
17     <div>
18       <Header />
19       <Routes>
20         <Route path="/" element={<Home />} />
21         <Route path="/student" element={<StudentHome />} />
22         <Route path="/courses" element={<Courses />} />
23         <Route path="/contact" element={<Contact />} />
24         <Route path="/about" element={<About />} />
25         <Route path="/login" element={<Login />} />
26         <Route path="/signup" element={<SignUp />} />
27       </Routes>
28     </div>
29   );
30 }
31
32 export default App;
```

Figure 2: Code for render urls

This React application (likely in App.js) defines components for various sections (header, footer, pages) and uses the BrowserRouter component to enable navigation between them. It sets up routes using Route components, specifying which component to render based on the URL path (e.g., About, Contact, Courses).

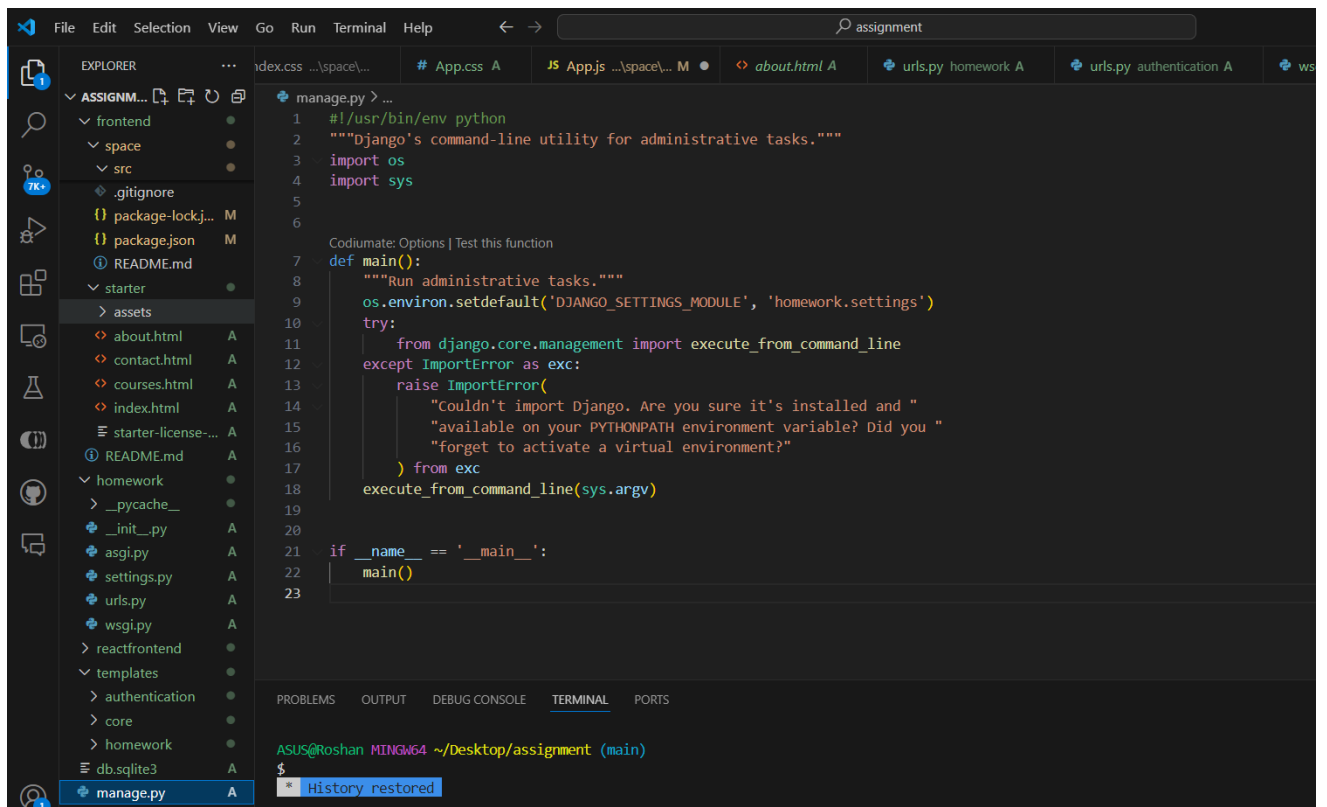
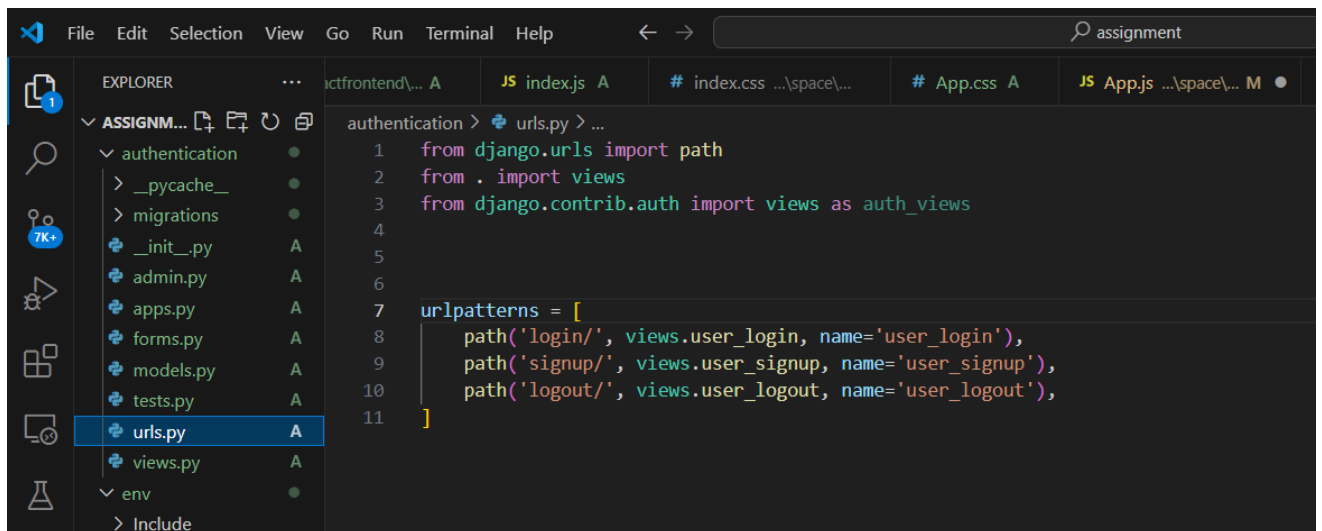


Figure 3: Django manage.py

The above code is from Django and it's the main files which controls all the other files .

The image shows a code editor interface with a dark theme. On the left is the 'EXPLORER' sidebar showing a project structure with a folder named 'authentication' containing files like '__init__.py', 'admin.py', 'apps.py', 'forms.py', 'models.py', 'tests.py', 'urls.py' (which is selected), and 'views.py'. The main editor area shows the content of 'urls.py' with the following code:

```
1 from django.urls import path
2 from . import views
3 from django.contrib.auth import views as auth_views
4
5
6
7 urlpatterns = [
8     path('login/', views.user_login, name='user_login'),
9     path('signup/', views.user_signup, name='user_signup'),
10    path('logout/', views.user_logout, name='user_logout'),
11]
```

Figure 4: URL patterns

The above provide code Django urls.py snippet from the authentication app defines URL patterns for user authentication. It maps URLs /login/, /signup/, and /logout/ to the respective view functions (user_login, user_signup, and user_logout) likely responsible for handling user login, signup, and logout functionalities within the application.

2.2 uses of version control

```
ASUS@Roshan MINGW64 ~/Desktop/assignment (main)
$ git log
commit 12555984479ce272835682812af4d7b557016fe3 (HEAD -> main, origin/main)
Author: Roshanbohara10 <119285937+Roshanbohara10@users.noreply.github.com>
Date: Sat Apr 27 16:00:29 2024 +0545

    add next line

commit 95cdd45ba0f8100f727c508fd49149cf32b8ae98
Author: Roshanbohara10 <119285937+Roshanbohara10@users.noreply.github.com>
Date: Sat Apr 27 15:59:48 2024 +0545

    Update README.md

commit 0d0e3f1ad4dcf7e9246cf75bf008bce70a278939
Author: Roshanbohara10 <119285937+Roshanbohara10@users.noreply.github.com>
Date: Sat Apr 27 15:51:03 2024 +0545

    Initial commit
```

Figure 5: Git logs

In the above code we have git logs.

Appendix B

1. Evidence of Good communication and file sharing

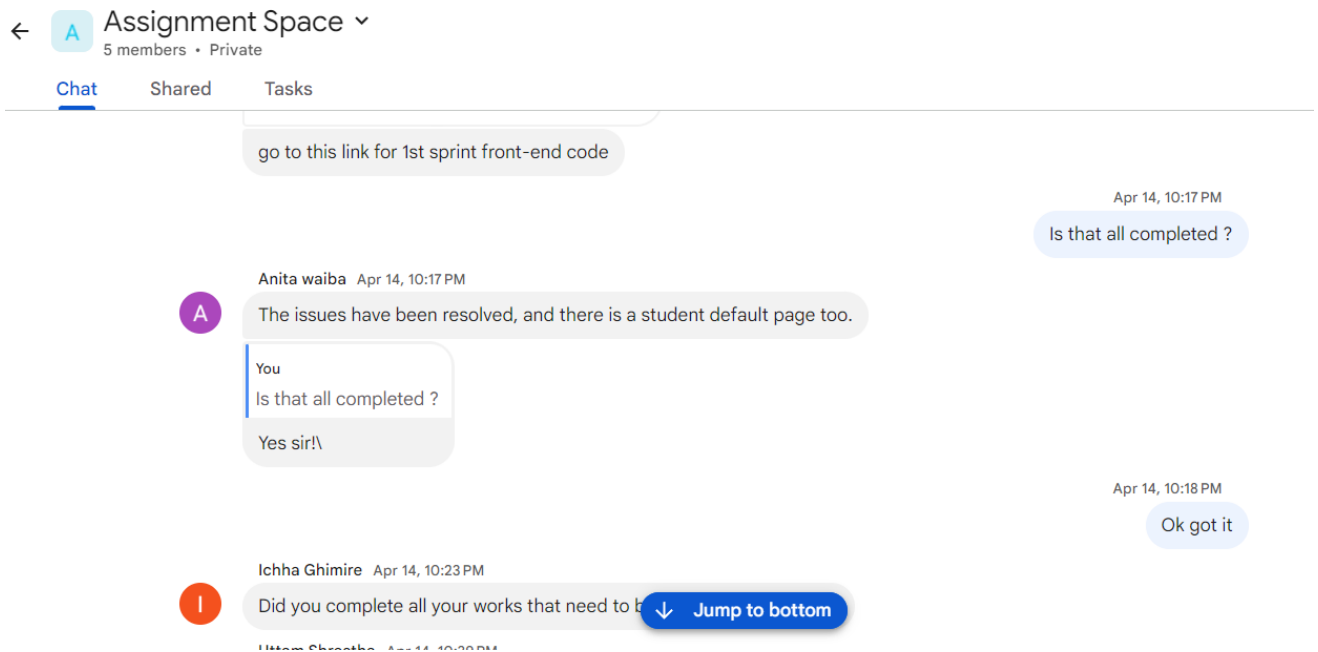


Figure 6: Discussion of issue

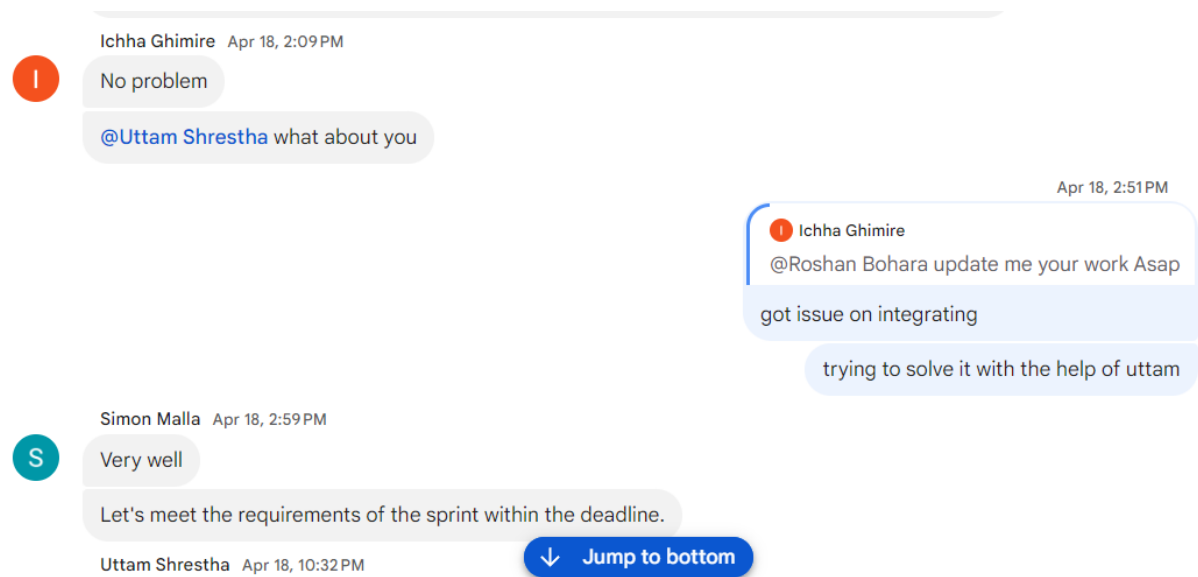


Figure 7: Discussion on integrating.

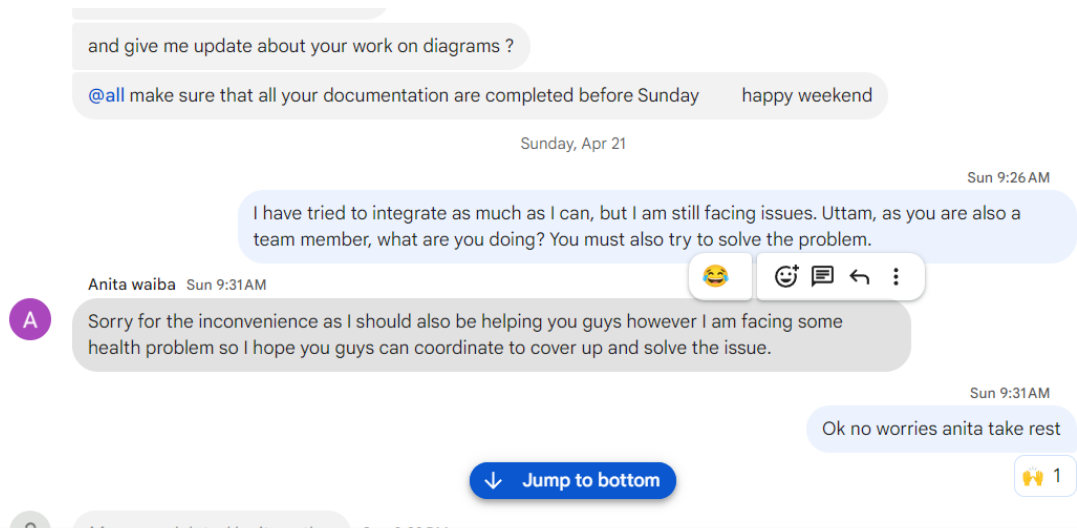


Figure 8:problem in integration.

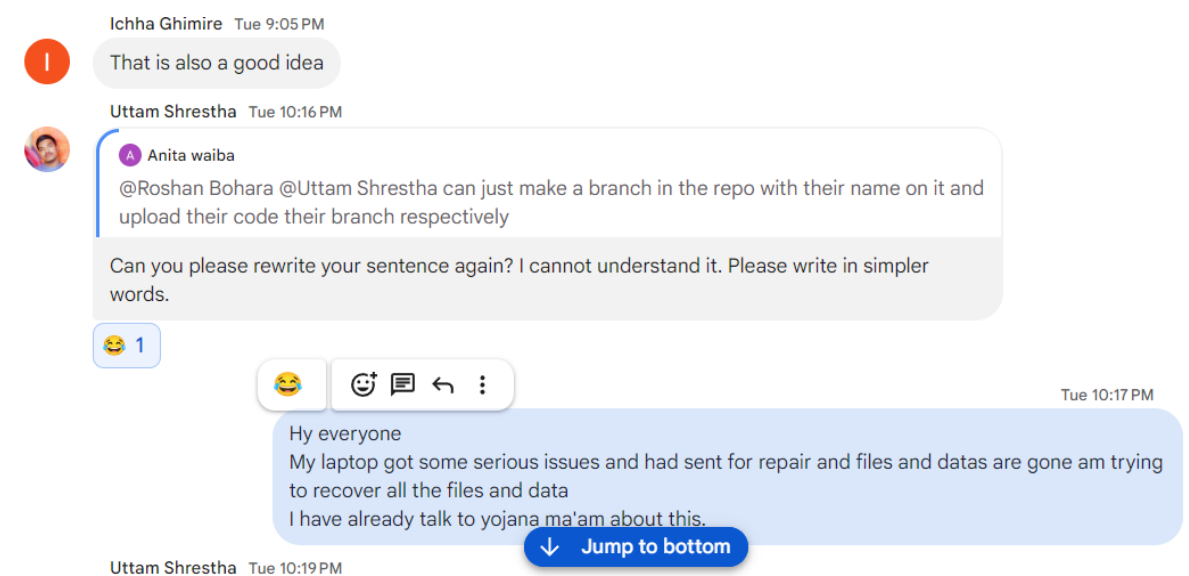


Figure 9: got innsue on my pc

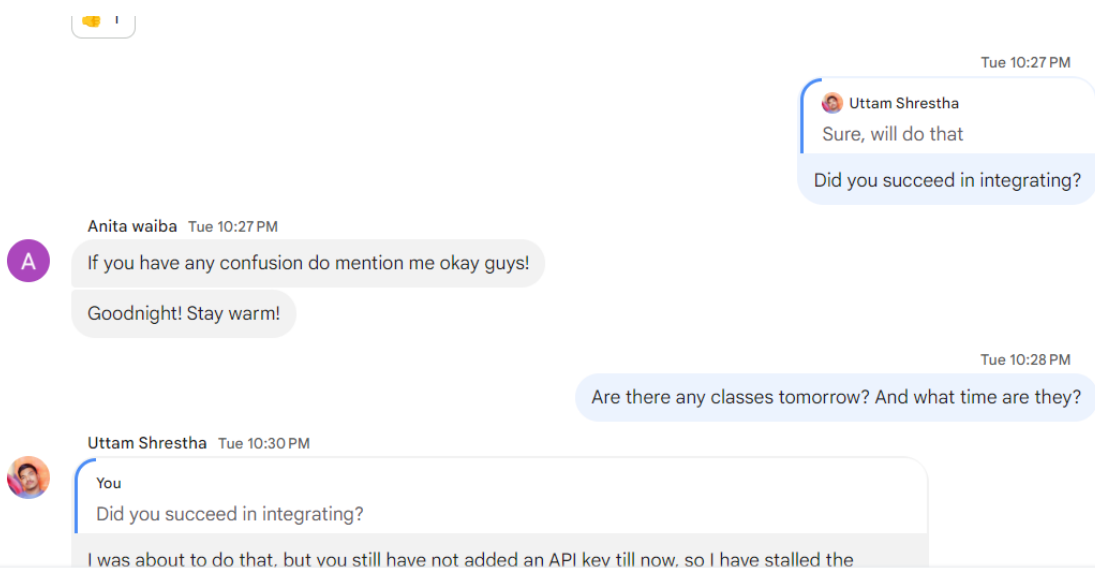


Figure 10: discussion on integrating.

2. Evidence of Continuing Personal Development (CPD)

Continuing Personal Development (CPD) refers to the process of actively seeking the new skills and new technologies in the market which are using by big companies, and which involves pursuing learning and developmental activities through enhance skills, knowledge, and personal development. It will structure self-improvement that extends beyond the normal education and training which can be found in our society or community which emphasizes growth and professional/personal effectiveness.

I have utilized online guides like YouTube, documentations (stackoverflow, n.d.), articles, and interactive coding platforms to uplift my skills through this project. Additionally, I understand deeper into subjects such as coding in Django framework within react.js, seeking insights beyond YouTube tutorials, documentations etc. Exploring or finding various websites provided me inspiration and knowledge to create visually appealing creative ideas also considering the users perspective to enhance usability. This approach positive impact on me which helps me to get positive outcomes as well for the project and expanded my understanding of database connectivity, reinforcing my learning and practical skills.

By maintaining consistency in research and problem-solving skills I gained knowledge in various areas of coding, including utilizing GitHub for version control and problem tracking. I found new creative ideas into reality through programming tools which are the best part for motivating me to keep going on the project. As a developer and student at the same time, I had to struggle with time management to make this project happens but this experience taught me to effectively manage my time table and allocate my time to meet deadlines.

Through this sprint 1 project, I encountered many challenges and problems but with my team members I have successfully completed this project with my team members. I struggled to integrate Django to js but with my team member we have solved the problems and we learned many new skills, communicate techniques etc. lastly, it enhance my ability to address challenges and equipped with valuable skills for future projects.

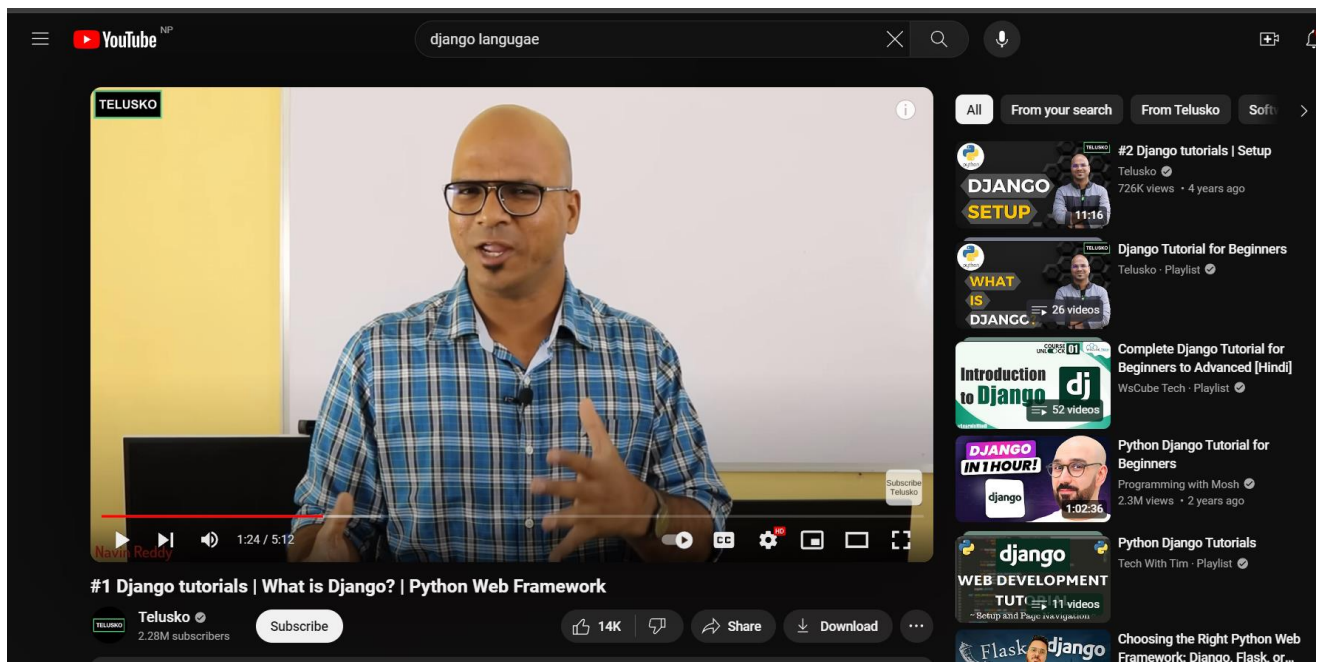


Figure 11: Learning django framework

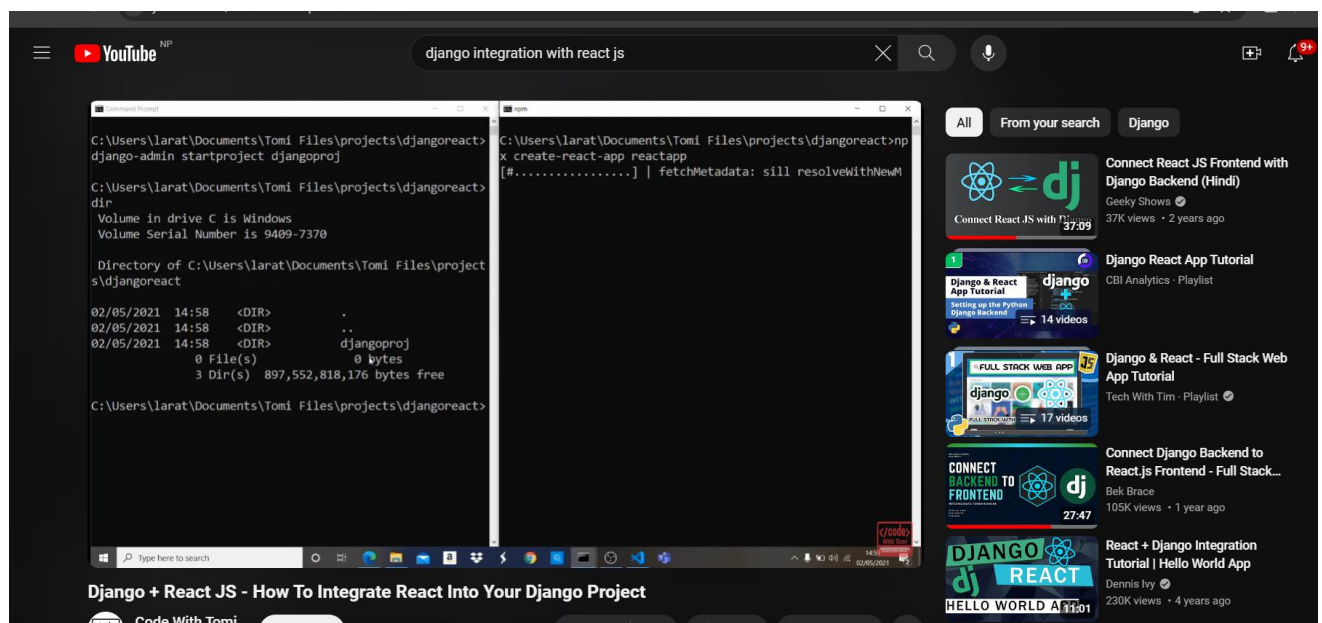


Figure 12: learning Intregation

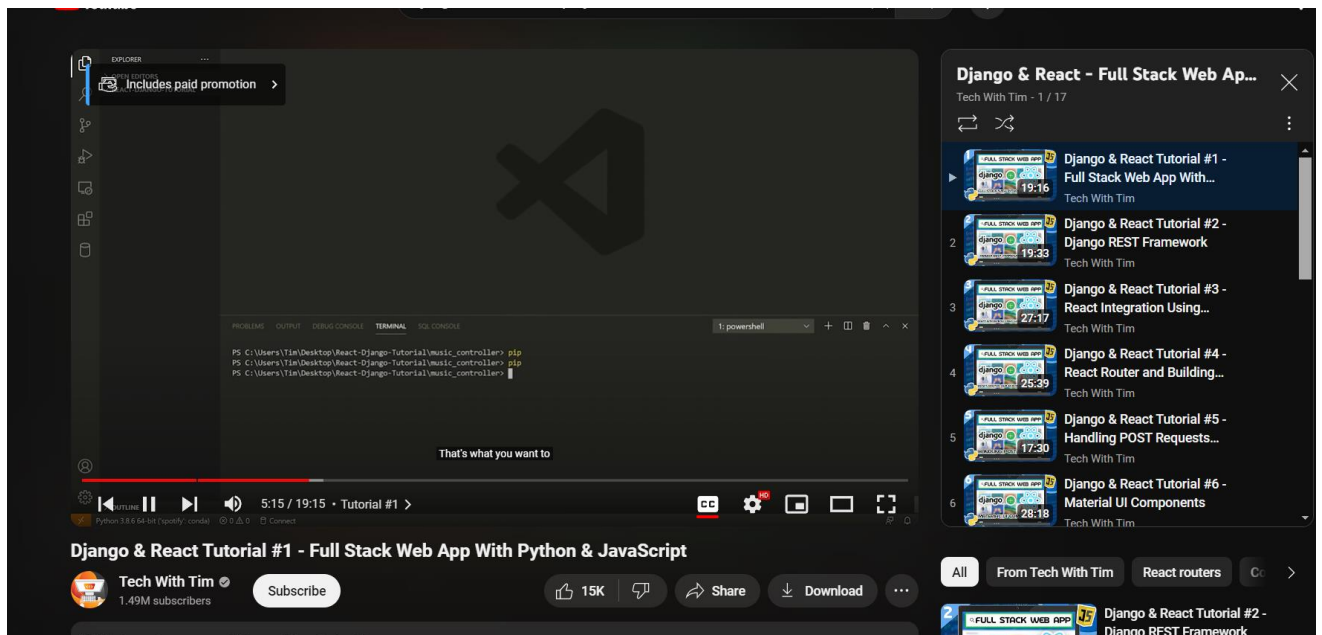


Figure 13: Learning more on Django and react.

It's all about learning the coding environment or frameworks

References

stackoverflow. (n.d.). *How to include JavaScript in Django Templates?* Retrieved from stackoverflow:
<https://stackoverflow.com/questions/30313314/how-to-include-javascript-in-django-templates>
w3school. (n.d.). *What is React?* Retrieved from w3schools:
https://www.w3schools.com/whatis/whatis_react.asp