# Project Plan – Course Website

## Project Overview:

Our website functions as an online course platform. Users can sign in, browse available subjects and teachers, and enroll in courses directly through the platform. The site includes pages for registration, login, contact, and viewing course and teacher information.

## Project Goals:

The goal of this project was to build a user-friendly and functional online course platform that allows students to register, view available subjects and teachers, and enroll in courses. The platform aims to simplify access to education and improve course management.

## Team Members and Responsibilities:

### Front-End Team:

1. 1. Imran Qurbanaliyev

- Developed the homepage interface  
- Created the following HTML pages:  
 • intex.html  
 • teacher.html  
 • teachers.html  
 • dashboard.html  
- Files were added to the templates folder and pushed to the GitHub repository.

1. 2. Fatih Asgarov

- Developed the following HTML pages:  
 • about.html  
 • course.html  
 • courses.html  
- Files were placed inside the templates folder and committed to GitHub.

1. 3. Rauf Najiyev

- Designed user access pages:  
 • login.html  
 • register.html  
 • contact.html  
- Pages were added to the templates folder and uploaded to GitHub.

### Back-End Team:

1. 4. Azar Karimov

- Developed the back-end functionality of the website  
- Created two main directories: lahiye and pages  
- Ensured all server-side components were functional and pushed to GitHub.

1. 5. Orkhan Mustafayev

- Coordinated the back-end structure  
- Managed the integration between front-end pages and back-end logic  
- Ensured full project functionality and committed the final version to GitHub.

## Technologies and Languages Used:

The following programming languages and technologies were used in the development of the course website:  
  
- HTML5: For structuring the web pages  
- CSS3: For styling and layout  
- JavaScript: For client-side interactivity  
- Python: As the main back-end language  
- Django Framework: For building the web application logic  
- Git & GitHub: For version control and collaboration  
- Bootstrap: For responsive design  
- SQLite: As the default development database

## Challenges Faced:

- Ensuring compatibility between front-end pages and Django back-end logic  
- Designing responsive layouts for different screen sizes  
- Managing collaboration via GitHub among multiple team members

## Future Improvements:

- Adding a payment system for paid courses  
- Creating a mobile-friendly version  
- Implementing user dashboards for progress tracking  
- Integrating video lessons and chat support