

**Work Plan for Introduction to Web Development**

**Objective**

To provide students with a comprehensive understanding of web development, including both front-end and back-end technologies, tools, and processes. By the end of the course, students will be able to create, deploy, and maintain a functional website.

**Course Duration**

10 weeks, with two 2-hour sessions per week.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Week** | **Session** | **Topic** | **Content** | **Activity** |
| **Week 1** | Session 1 | Course Overview and Importance of Web Development | Overview of the course, importance of web development | Create a personal web development goals list |
|  | Session 2 | Basic Concepts and Tools | Introduction to front-end and back-end development, setting up a development environment | Install Visual Studio Code and other necessary tools |
| **Week 2** | Session 1 | Introduction to HTML | HTML structure, elements, and attributes | Build a simple HTML page with headings, paragraphs, and lists |
|  | Session 2 | Advanced HTML | Forms, tables, and multimedia elements | Create a registration form with various input types |
| **Week 3** | Session 1 | Introduction to CSS | CSS syntax, selectors, and properties | Style the HTML page created in Week 2 |
|  | Session 2 | Advanced CSS | Layout techniques (flexbox, grid), responsive design | Create a responsive layout for the HTML page |
| **Week 4** | Session 1 | Introduction to JavaScript | JavaScript syntax, variables, and data types | Add interactivity to the HTML page (e.g., form validation) |
|  | Session 2 | Advanced JavaScript | DOM manipulation, event handling | Create a simple interactive feature (e.g., a to-do list) |
| **Week 5** | Session 1 | Server-Side Programming | Overview of server-side languages (PHP, Python, Node.js), setting up a local server | Create a simple server-side script |
|  | Session 2 | Databases | Introduction to databases (MySQL, MongoDB), connecting to a database and performing CRUD operations | Create a database and perform basic operations |
| **Week 6** | Session 1 | Integrating Front-End and Back-End | Overview of full-stack development | Start a project to integrate front-end and back-end components |
|  | Session 2 | Working with APIs | Introduction to APIs and RESTful services, consuming APIs using JavaScript | Create a front-end interface that interacts with a public API |
| **Week 7** | Session 1 | Version Control with Git | Introduction to Git and GitHub, basic Git commands and workflows | Create a repository and practice version control |
|  | Session 2 | Frameworks and Libraries | Overview of popular frameworks (React, Angular, Vue.js), setting up a project with a front-end framework | Create a simple application using a front-end framework |
| **Week 8** | Session 1 | Deployment Basics | Introduction to web hosting and deployment, deploying a website using GitHub Pages | Deploy the project created in previous sessions |
|  | Session 2 | Advanced Deployment | Continuous Integration and Continuous Deployment (CI/CD), setting up a CI/CD pipeline | Implement a CI/CD pipeline for the project |
| **Week 9** | Session 1 | Performance Optimization | Techniques for optimizing web performance, tools for measuring and improving performance | Optimize the project for performance |
|  | Session 2 | SEO Basics | Introduction to SEO techniques, on-page SEO best practices | Implement SEO improvements for the project |
| **Week 10** | Session 1 | Final Project Work | Work on the final project, integrating all learned concepts | Individual assistance and feedback |
|  | Session 2 | Project Presentations and Review | Present final projects to the class, review and feedback on projects, course summary and future learning resources | Present final projects |

**Evaluation**

* **Assignments**: Weekly assignments based on lecture topics (40%)
* **Final Project**: Comprehensive project integrating front-end and back-end development (40%)
* **Participation**: Class participation and discussions (20%)

**Notes for Students**

* **Preparation**: Ensure your development environment is set up before each class.
* **Resources**: Utilize provided learning resources and recommended books.
* **Practice**: Consistent practice is key to mastering web development concepts.
* **Questions**: Do not hesitate to ask questions during or after class for clarification.