Course Title: JavaScript Client-Side Scripting Language Syllabus

Course Description: This course is designed to provide students with a comprehensive understanding of JavaScript programming language for web development. Students will learn the fundamentals of JavaScript syntax, DOM manipulation, event handling, asynchronous programming, and more. The course will cover both basic and advanced topics to equip students with the skills needed to build interactive and dynamic web applications.

Prerequisites: Basic knowledge of HTML and CSS. Prior programming experience, especially with languages like Python or Java, is helpful but not required.

Course Objectives:

1. Understand the fundamental concepts of JavaScript programming language.
2. Learn how to manipulate the Document Object Model (DOM) using JavaScript.
3. Develop skills in handling events and creating interactive web pages.
4. Gain proficiency in asynchronous programming with JavaScript, including promises and async/await.
5. Explore advanced topics such as ES6 features, closures, and error handling in JavaScript.

Course Outline:

1. **Introduction to JavaScript**
   - History and evolution of JavaScript
   - Setting up JavaScript development environment
   - Writing and running JavaScript scripts
2. **JavaScript Basics**
   - Variables and data types
   - Operators and expressions
   - Control structures (if-else, switch, loops)
3. **Functions and Scope**
   - Declaring and invoking functions
   - Function parameters and return values
   - Function scope and closures
4. **Arrays and Objects**
   - Working with arrays and array methods
   - Creating and manipulating objects
   - Object-oriented programming concepts in JavaScript
5. **DOM Manipulation**
   - Introduction to the Document Object Model (DOM)
   - Accessing and manipulating DOM elements

- Modifying HTML and CSS properties

6. **Event Handling**
   - Handling user events (click, hover, submit, etc.)
   - Event listeners and event propagation
   - Event delegation for efficient event handling
7. **Asynchronous JavaScript**
   - Introduction to asynchronous programming
   - Working with setTimeout and setInterval
   - Promises and asynchronous functions
8. **AJAX and Fetch API**
   - Making asynchronous requests with XMLHttpRequest
   - Fetch API for modern AJAX requests
   - Handling responses and error handling
9. **ES6 Features**
   - Arrow functions and template literals
   - Destructuring and spread/rest operators
   - Classes and inheritance
10. **Error Handling**
    - Handling errors with try-catch blocks
    - Throwing and catching custom errors
    - Error propagation and handling asynchronous errors
11. **Browser Storage**
    - Working with browser cookies
    - Using localStorage and sessionStorage
    - Storing and retrieving data in the browser
12. **Introduction to Modern JavaScript Libraries and Frameworks (Optional)**
    - Overview of popular JavaScript libraries and frameworks (React, Angular, Vue.js, etc.)
    - Advantages of using libraries and frameworks for web development
    - Basic concepts and features of a chosen library or framework

Assessment:

- Weekly assignments to reinforce learning concepts.
- Midterm exam covering topics covered in the first half of the course.
- Final project requiring students to develop an interactive web application using JavaScript.

Textbook: "Eloquent JavaScript" by Marijn Haverbeke

Additional Resources:

- Online tutorials and documentation (MDN Web Docs, JavaScript.info etc.).

- Supplemental readings and materials provided by the instructor.

Grading:

- Assignments: 30%
- Midterm Exam: 20%
- Final Project: 40%
- Participation and Attendance: 10%

Attendance Policy: Regular attendance is expected. Students are allowed a maximum of three unexcused absences. Excessive absences may result in a reduction of the final grade.

Office Hours: Instructor office hours will be held twice a week for additional help and clarification.