ITMD 441 SYLLABUS

ITMD 441 Web Application Foundations

Hours: 3 credit hours / 45 contact hours

Instructor: Brian Bailey

Textbook, title, author, and year:

a. None; online material is used

Specific course information:

a. **Catalog description:** In this course students examine core web technologies that are integral in the creation of web-based applications typically delivered in a browser. The course will cover fundamental web protocols, web application architectures, markup, and scripting languages. A focus will be placed on writing modern, standards-compliant JavaScript and how it is used to interact with HTML and CSS to enable rich user interfaces and communication with other services. Current frameworks, libraries, and tools will also be explored.
b. **Prerequisites:** ITMD 361

Specific goals for the course

a. **Course Outcomes:** Students completing this course learn about the core technologies used to develop contemporary web applications. This includes the primary protocols, tools, software, markup, and scripting languages used in modern web development and how they have progressed over time. Each student successfully completing this course will demonstrate a strong foundational knowledge in the design and development of web-based internet applications. Students will use standards compliant HTML and CSS to create responsive user interfaces targeting modern browsers. Additionally, students learn about the JavaScript language, debugging techniques, and JavaScript APIs and how to effectively utilize them.
b. **Course student outcomes:**
   At the conclusion of this course, successful students will be able to:
   - Recognize HTML, CSS, and JavaScript markup and code in a web page/application
   - Describe the differences between server and client-side technology as it relates to applications delivered through internet browsers
   - Explain the history, role, and use of enabling technologies in modern web-based applications, including JavaScript, Cascading Style Sheets (CSS), and HTML including their impact
   - Outline strategies for turning interface mockups into working code prototypes
   - Describe the role of the HTTP and HTTPS protocol and the request/response cycle
   - Use valid standards compliant HTML, CSS, and JavaScript to build web pages and applications
   - Use modern browser developer tools to inspect and debug web applications
   - Describe basic responsive design principles and techniques and explain the advantages of responsive design
   - Discuss differences in ES5 and ES6+ JavaScript language and syntax
   - Develop native ES6+ JavaScript for use in the browser
   - Describe terminology and functionality afforded by advanced JavaScript programming paradigms including: Prototype, Object Oriented Programming (OOP), JavaScript Object Notation (JSON), Namespacing, Modules
   - Compare AJAX techniques for asynchronously loading content
   - Do Document Object Model (DOM) manipulation in native JavaScript
   - Describe and utilize HTML5 APIs
   - Outline the potential security threats posed to internet applications and their users and strategies for mitigating these risks
   - Use tools to manage project dependencies and source code management tools for version control in a project

Topics to be covered:

a. JavaScript, Cascading Style Sheets (CSS), HTML
b. Standard Compliant markup and code
c. HTML5 features, markup, and APIs
d. JavaScript language syntax
e. JavaScript version differences
f. History of Internet applications and web technologies
g. HTTP / HTTPS Protocol
h. Client/Server architecture
i. AJAX techniques for asynchronously loading content
j. Creating working code prototypes from interface mockups
k. Object Oriented Programming (OOP)
l. JavaScript Object Notation (JSON)
m. Namespacing
n. ES6 Modules
o. Security threats posed to internet applications
p. DOM manipulation
q. Responsive design principles
r. Introduction to command line tools and version control (Git)
s. Frameworks
t. Online API documentation