

ITMD 465 SYLLABUS

ITMD 465 Rich Internet Applications

Hours: 3 credit hours / 45 contact hours

Instructor: Brian Bailey

Textbook, title, author, and year: None; online material is used

Specific course information

- a. **Catalog description:** Students learn to create interactive rich internet applications using web development frameworks, applications, and techniques that primarily operate on the client-side. These applications often exhibit the same characteristics as desktop applications and are typically delivered through a standards-based web browser via a browser plug-in or independently via sandboxes or virtual machines. Current software frameworks used to download, update, verify, and execute these applications are addressed as well as writing applications for deployment in these frameworks.
- b. **Prerequisites:** ITMD 361

Specific goals for the course

- a. **Course Outcomes:** Students completing this course will be proficient in JavaScript and be able to use it to implement internet applications that utilize rich features. Each student successfully completing this course will demonstrate a strong knowledge in the design and development of JavaScript powered internet applications. Students will be able to use the suite of HTML5 JavaScript APIs, with JavaScript programming and debugging techniques; to create user interfaces for responsive applications in modern browsers. Students will be able to style their interfaces with modern applications of standards compliant CSS. Additionally, students will be able to effectively utilize current JavaScript libraries and APIs which may include ones such as ES6, jQuery, canvas, geolocation, audio and video, browser storage, Babel, Electron, web components, ReactJS, and Node.js to enable rapid JavaScript application development.
- b. **Course student outcomes:**
At the conclusion of this course, each successful student will be able to:
 - Describe the differences between server and client-side technology as it relates to applications delivered through internet browsers
 - Discuss the history of Internet applications and the web technologies that have enabled that history

- Describe the relationship between browser technology and traditional applications
- Explain the history, role, and use of enabling technologies in modern web-based applications, including JavaScript, Cascading Style Sheets (CSS3), and HTML 5
- Outline traditional and assisted AJAX techniques for asynchronously loading content
- Outline strategies for turning interface mockups into working code prototypes
- Describe terminology and Develop applications using the functionality afforded by advanced JavaScript programming paradigms including: Prototype, Object Oriented Programming (OOP), JavaScript Object Notation (JSON), Namespacing, Modules
- Outline the potential security threats posed to internet applications and their users and strategies for mitigating these risks.
- Discuss differences in ES5 and ES6+; use Babel to enable ES6 in ES5 environments
- Develop native ES6 JavaScript for use in the browser, Node.js, and ReactJS.
- Compare and contrast the use of jQuery and native JavaScript or CSS features.
- Manipulate the Document Object Model in plain JavaScript.
- Describe responsive design principles and techniques and explain the advantages of responsive design as it relates to desktop and mobile web-based applications.
- Recall ways to detect the support of HTML5 features
- Describe and utilize HTML5 APIs including Canvas, Geolocation, & Local Storage
- Describe the use of the HTML5 Audio and Video APIs and compare and contrast the HTML5 APIs to legacy audio and video solutions
- Manage project dependencies / toolchains with Node.js, NPM, and package.json files.
- Describe how to use online API documentation to learn a new framework or library.
- Use git / github as source code management tools for version control in a project.
- Create web components and SPAs using the ReactJS framework.
- Explain the benefits of using JSX in ReactJS.
- Use modern browser developer tools to inspect and debug web applications.

ITMD 465 Topics to be covered:

- a. Server and client-side technology in applications delivery
- b. History of Internet applications and enabling web technologies
- c. Enabling technologies in modern web-based applications: JavaScript, Cascading Style Sheets (CSS), HTML
- d. AJAX techniques for asynchronously loading content
- e. Creating working code prototypes from interface mockups
- f. JavaScript programming paradigms including: Prototype, Object Oriented Programming (OOP), JavaScript Object Notation (JSON), Namespacing, Modules
- g. Security threats posed to internet applications
- h. ES5 and ES6+ and Babel to enable ES6 in ES5 environments
- i. Native ES6 JavaScript for use in the browser, Node.js, and ReactJS.
- j. jQuery
- k. DOM manipulation in JavaScript
- l. Responsive design principles and techniques
- m. HTML5 features, HTML 5 markup and the DOCTYPE
- n. HTML5 APIs: Canva, Geolocation, Local Storage, and Audio and Video
- o. Manage project dependencies and toolchains with Node.js, NPM, and package.json files
- p. Online API documentation
- q. Using git / github as a source code management system for version control in a project
- r. ReactJS framework
- s. JSX in ReactJS.
- t. Inspecting and debugging web applications