

Stanford Summer Institutes: Web Technologies

Course Information:

- First class day: June 19th
- Last class day: July 8th
- Instructor: Chandranil Chakrabortii, cchakrab@ucsc.edu
- Teaching Assistant : TBD

Course Description:

The SPCS Summer Institutes High School Program Web Technologies course will introduce the students to the fundamentals of Web technologies and help them gain experience in creating Web applications. The course will provide a general overview of markup languages, scripting languages, network protocols, event-driven programming, and databases, and see how they all work together to deliver exciting applications. The World Wide Web continues to provide a foundation for the development of a broad range of increasingly influential and strategic technologies, supporting a large variety of applications and services, both in the private and public sectors. There is a growing need for management and decision makers to gain a clearer understanding of the application development process, from planning through to deployment and maintenance. This course will provide an insight into architectures, protocols, standards, languages, tools and techniques; an understanding of approaches to more dynamic and mobile content; and demonstrate how to analyse requirements, plan, design, implement and test a range of web applications. Students will work in teams to develop a project which they would present in class via a live demo. The course is divided into three modules:

Module 1 : Foundations of Web Technology

The first module covers the basic technologies on which the Web is founded. Aspects covered include: historic development of the Web; 'architecture' and basic client server architecture; protocols such as HTTP; content markup (HTML, XHTML, CSS, XML) and issues of accessibility and usability; standards and standardisation organisations (W3C, Internet working group); and security (firewalls, HTTPS, certificates).

Module 2 : Web Architectures

This module develops 'architecture' by exploring different approaches and their properties as well as considering components (databases, registries etc.) which are used

in different approaches and examining the nature of different server and client side languages.

These aspects include; tiered, service orientated, and network architectures; the role of the database, web services, registries, scalability, reliability, approaches to security (cookies, certificates etc), and server and client side implementation languages.

This module includes JavaScript programming activities using Node.js and Express. All the code required to produce a simple web application is provided and explained.

Module 3 : Developing applications

The final block explores how applications are planned, designed and developed by IT professionals, examining project planning, application design, development environments and tools as well as application deployment and maintenance.

Course Text and References:

- Node.js in Action - Mike Cantelon, Marc Harter et al.
- Web Development with Express and Node - Ethan Brown
- The Node Beginner Book - M. Kiessling
- Node.js Tutorial for Beginners by thenewboston
- Lynda.com - Web Technology Fundamentals
- USC - Web Technologies - Prof. Ellis Horowitz and Prof. Marco Papa(Course ID : CSCI571)
- UTSA - Web Technologies - Prof. Kay Robbins (Course ID: CS 4413)
- SJSU - Web-Based Application Systems and Tools - Prof. Jerry Gao (Course ID: CMPE296U)
- The Open University - Web Technologies (Course ID: TT284)

Course Goals:

- Learn new concepts, ask questions and clear up your confusion.
- Use your imagination to think of new ideas and implement them.
- Work in teams to build a collaborative team environment.
- Comment your code thoroughly to smooth the debugging process.
- Build a dynamic website and present in class via a live demo.
- Have fun!

Course Schedule:

- Day 1: Getting Started!
 - Morning
 - Introduction
 - Icebreaker: Name and Nickname

- Course Introduction : A brief outline of the topics the course will cover
 - About Stanford University's contribution to Computer Science
 - Modern Computer Science trends like Facebook, Google, Android etc.
 - Course Themes: Computational Thinking, Debugging, and Teamwork
- Introduction to Web Technology
 - Overview of Web Technologies
 - Introduction to HTML
 - Afternoon
 - Build your personal website using HTML
 - Write a survey on three popular Web Protocols used today
- Day 2: HTML forms and Introduction to Javascript
 - Morning
 - Overview of HTML forms
 - Introduction to Style Sheets
 - Form validation using Javascript
 - Recap.
 - Afternoon
 - Create a simple Email Registration form using HTML, CSS and Javascript
- Day 3: Introduction to Relational Databases
 - Morning
 - Introduce basic concepts of relational databases
 - Design simple tables appropriate for web applications
 - Database design principles
 - Transactions and synchronization
 - Afternoon
 - Review the Basic MySQL Tutorial from w3schools.com
 - Complete the Databases Assignment on basic SQL operations

- Day 4: Introduction to Git and Databases continued
 - Morning
 - Databases Recap
 - Keys and Joins
 - Introduction to Eclipse IDE
 - Introduction to Git
 - Set up Eclipse to use GIT
 - Afternoon
 - Complete the Databases Assignment on SQL join operations
 - Identify teammates for Final Project
- Day 5 : NoSQL databases and MongoDB
 - Morning
 - Introduction to NoSQL databases
 - Introduction to MongoDB
 - MongoDB comparison with MySQL
 - Basic CRUD operations
 - Aggregation Framework.
 - Afternoon
 - Complete the Databases Assignment on MongoDB
 - Final Project Proposal
- Day 6: Introduction to Express
 - Morning
 - The JavaScript Revolution
 - Brief history of Express- A Node.js Application framework
 - Node - A new kind of Web Server
 - The Node Ecosystem
 - Project Ideas
 - Afternoon
 - Design the Project Outline Document

- Day 7: Getting started with Node
 - Morning
 - Installing Node
 - How to use the Terminal
 - Using Node.js with Express
 - Using NPM packages and Project Metadata
 - Debugging and Testing code
 - Afternoon
 - Create a simple personal website using Node and Express
- Day 8: Persistence
 - Morning
 - Filesystem Persistence
 - Database Persistence
 - Database Connections with Mongoose
 - CRUD Operations
 - Recap
 - Afternoon
 - Start working on Final Project
- Day 9: The Request and Response Objects
 - Morning
 - The Parts of a URL
 - HTTP Request Methods
 - Request/Response Headers
 - The Request/Response Objects
 - Debugging and Testing code
 - Afternoon
 - Continue working on Final Project
- Day 10: Form Handling
 - Morning
 - Sending Client Data to the Server
 - Different Approaches to Form Handling

- Form Handling using Express
- File Uploads
- Recap

- Afternoon

- Continue working on Final Project
- Create a Project status update report

- Day 11: Cookies and Sessions

- Morning

- Cookies in Express
- What to Use Sessions For
- Using Sessions
- Using Sessions to Implement Flash Messages
- Recap

- Afternoon

- Continue working on Final Project
- Implement Cookies and Sessions in the Final Project

- Day 12: Implementing MVC in Express

- Morning

- Overview of the MVC Architecture
- Models
- View Models
- Controllers
- Conclusion

- Afternoon

- Continue working on Final Project
- Implement MVC Structure in the Final Project

- Day 13: REST APIs and JSON
 - Morning
 - JSON and XML
 - Our API
 - Cross-Origin Resource Sharing (CORS)
 - Using Express to Provide an API
 - Using a REST Plugin
 - Afternoon
 - Continue working on Final Project
- Day 14: Live Demo and Wrapping Up
 - Morning
 - Wrapping Up
 - Live Demo of Student Projects
 - Review of the Projects by Students
 - Afternoon
 - WrapUp: Programming after SPCS
 - Make changes and publish the Final Project
 - Last Minute Debugging
 - Course Surveys