Front-End Web Developer Nanodegree Syllabus

Contact Info

While going through the program, if you have questions about anything, you can reach us at support@udacity.com. For help from Udacity Mentors and your peers visit the Udacity Classroom.

Nanodegree Program Info

Version: 8.0.0

Length of Program: 201 Days*

** This is a self-paced program and the length is an estimation of total hours the average student may take to complete all required coursework, including lecture and project time. Actual hours may vary.*

Part 1: Web Foundations

Learn HTML, the foundational structure of all websites. Style with CSS and build your own personal portfolio website.

Project: Build a Portfolio Site

Given a pdf mockup of a website from a designer, translate it to a real website using your HTML and CSS skills.

Part 2: Web Programming with JavaScript

Learn the JavaScript programming language. Start working with common developer tools, including the Git version control system.

Project: Memory Game

Build your own version of the classic memory game 'Concentration', using JavaScript, DOM selectors and Event Listeners.

Part 3: Exploring JS - Objects, Tools and Testing

Learn about Accessibility, Object-Oriented JavaScript techniques, closures, the "this" keyword, and the new ES6 specification.

Project: Classic Arcade Game Clone

You will be provided with visual assets and a game loop engine; using these tools you must add a number of entities to the game including the player characters and enemies to recreate the classic arcade game Frogger.

Project: Feed Reader Testing

You have been given a feed reader application that another developer has worked on, and now need to write tests against it using Jasmine's testing syntax.

Part 4: Front-End Applications

Use APIs to fetch data for your applications. Begin building with Front End frameworks and build offline capable websites.

Project: Restaurant Reviews App—Stage 1

In this real world case study, given the front end code for a static Restaurant Reviews App, revise the site to be responsive and achieve accessibility standards.

Part 5: Building with React

Leverage the power of the React UI library. Convert an existing application to use React and build your own single page React app that uses the Google Maps API.

Project: MyReads: A Book Tracking App

In the MyReads project, you'll create a web application that allows you to select and categorize books you have read, are currently reading, or want to read.

Project: Neighborhood Map (React)

Build a single page map application using React and the Google Maps API. Integrate a third-party data API and make your app accessible and usable offline.

Udacity

Generated Fri Nov 1 19:28:50 PDT 2019