

Front End Developer 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

The goal of the Front End Web Developer Nanodegree program is to equip learners with the unique skills they need to build and develop a variety of websites and applications.

Prerequisite Skills

A well-prepared learner is able to:

- Layout a simple webpage using HTML
- Style a website element using CSS
- Write and test software with JavaScript
- Inspect websites using Developer Tools on a modern web browser (Chrome, Firefox, or Edge)
- Debug and troubleshoot errors and failures in JavaScript programs

Required Software

- GIT 2.23 or Latest
- Text Editors
- Javascript
- Node.js
- FetchApi
- HTML, CSS
- Document Object Model (DOM)
- Express, Body-Parser, CORS, Webpack Packages
- Geonames, Weatherbit, Pixabay Account

Version: 1.0.0

Length of Program: 107 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: Welcome to the Program!

Welcome to the Front-End Web Developer Nanodegree program. This is your first step on your journey to become a Front-End programmer. Learn what this program is all about as well as how to find support along your learning journey.

Part 2: CSS, Website Layout, Website Components

In this part, you'll create a multi-page blog website, using best practices for content and page styling with HTML and CSS. Practice using responsive layouts, Flexbox, and CSS Grid to create the structure and design for your own blog.

Project: Personal Blog Website

Use your new skills with HTML and CSS to build a personal blog website, complete with a main page and example blog posts!

Supporting Lessons

Lesson	Summary
HTML	Learn the basics of HTML and start using different elements to build out the basic structure of webpages.
CSS	Learn the basics of CSS and start using different properties to create appealing layouts and styling for all types of use cases.
CSS Flexbox	Build on your CSS knowledge by diving into Flexbox, a one-dimensional layout technique for helping organize items within your webpage.
CSS Grid	Extend your CSS skills even further with CSS Grid, which allows you to organize in two dimensions with rows and columns. Grid can work with Flexbox to lay out pages just like on the modern web.
Creating Responsive Layouts	Start with a basic Grid-based layout for a blog, and learn how to use media queries to build a responsive layout for device types of all sizes!

Part 3: JavaScript & The DOM

In this part, you will build a multi-section landing page. You will learn to dynamically add content to a web page and build a landing page that combines your skills with JavaScript, HTML, and CSS to update and control the page and create a dynamic user experience.

Project: Landing Page

Build a landing page that uses JavaScript to create an interactive navigational experience for the user.

Supporting Lessons

Lesson	Summary
Syntax	With this major update to the JavaScript language, a number of syntax improvements have been added. In this lesson, you'll learn to improve your JavaScript code with these syntax updates.
The Document Object Model	The Document Object Model (DOM) allows developers to interact with and control the rendered page. You'll learn how the DOM is formed, what Nodes and Elements are, and how to select items from the DOM.
Creating Content with JavaScript	Static is boring, so we'll use JavaScript and DOM methods to create new page content, update existing content, and delete content. You'll also learn how to check and set the styling of page elements.
Working with Browser Events	Hundreds of events fire every time you interact with a page. Learn what an event is, how to listen for an event and respond to it, what data is included with an event, and the phases of an event.
Performance	Learn how to measure the code you write to make sure it's performing efficiently and how reflow and repaint affect performance. Learn how to harness the JavaScript Event loop for performance gains.

Part 4: Web APIs and Asynchronous Applications

In this part, you'll apply your new skills to combine data from the OpenWeatherMap API and client side (browser) HTML forms to create a web app that records a weather journal for users. This project requires you to create an asynchronous web app that uses Web API and user data to dynamically update the UI for a Weather Journal App.

Project: Weather Journal App

Apply your new skills to combine data from the OpenWeatherMap API and client side (browser) HTML forms,

to create a web app that records a weather journal for users.

Supporting Lessons

Lesson	Summary
Course Introduction	In this lesson we will introduce the main topics of the course and preview the kind of web app that can be made with the skills developed throughout the following lessons.
Node & Express Environment	In this lesson, students will learn how to setup a Node + Express environment to develop a web application on their local machine.
HTTP Requests & Routes	In this lesson students will learn how to use Express to setup server side routes and execute client side handlers for GET and POST requests.
Asynchronous JavaScript	In this lesson students will learn how to implement asynchronous JavaScript with the Fetch API and Web APIs.

Part 5: Build Tools and Single Page Web Apps

In this part, you'll get a taste of some common production environments and tools used in a front end developer role. You'll build a web tool that allows users to run Natural Language Processing (NLP) on articles or blogs found on other websites.

Project: Evaluate a News Article with Natural Language Processing

Build a web tool that allows users to run Natural Language Processing (NLP) on articles or blogs found on other websites.

Supporting Lessons

Lesson	Summary
Intro to Build Tools	Get a brief introduction to build tools, including how they can help ease and advance your web development, and what some of the most common build tools are.
Basics of Webpack	By the end of this lesson you will understand the reasons developers use webpack, and be able to explain at a high level what webpack does.
Sass and Webpack	By the end of this lesson you will be able to explain what Sass is and its relation to CSS, and understand the steps necessary to use Sass in a project.
Final Touches	Add some final pieces to your Webpack project, get your full functionality back, and learn about Service Workers.

Part 6: Career Services

Project: Improve Your LinkedIn Profile

Find your next job or connect with industry peers on LinkedIn. Ensure your profile attracts relevant leads that will grow your professional network.

Supporting Lessons

Lesson	Summary
Industry Research	You're building your online presence. Now learn how to share your story, understand the tech landscape better, and meet industry professionals.

Project: Optimize Your GitHub Profile

Other professionals are collaborating on GitHub and growing their network. Submit your profile to ensure your profile is on par with leaders in your field.

Part 7: Capstone

You're now ready to combine all of the skills you've developed throughout the Nanodegree program to build an online travel app. You'll work with data sources from multiple APIs to create a dynamic travel weather planning application that helps people plan trips by generating weather forecasts for the places they're visiting.

Project: FEND Capstone - Travel App

Build a travel app, combining all of your Front End skills into an amazing capstone project for your portfolio. Utilize multiple APIs to present a user with all they need to know about their trip.



Udacity

Generated Thu Jun 11 00:26:44 PDT 2020