1/1/2018

***** This syllabus is subject to change**

Full Stack Flex - Course Syllabus

Week 1: Learning the Fundamentals of Programming

In your first week at DigitalCrafts, you will gain an understanding of the core programming skills using the Python programming language.

In addition to learning the fundamentals, you will learn how to manage your project code by using a variety of tools, such as the Command Line, Git, and GitHub.

- Introduction to the Command Line
- Using Git and GitHub to manage your code
- Programming Fundamentals with Python

Week 2: Solving Problems Using Code

You will work through a number of coding challenges by building solutions in Python. Along the way, you will learn how and when to use data structures like dictionaries, objects, and classes.

You will tackle your first programming project by creating a 2D graphics game using the pygame library. By the end of the week, you will have plenty of experience reading, understanding, and debugging coding errors.

- Storing and working with data
    - Dictionaries
    - objects
    - classes
- Using the pygame library

Potential Project

- Building 2D graphics game

Week 3 - 4: Understanding the Foundations of Front-End

You will build your front-end web development skills starting with HTML and CSS. You will learn how to create web pages, format their contents, and finely control the layout. To do that, you will recreate several popular web page designs (such as medium.com's or google.com's look and feel).

Then, you will learn techniques to make your layouts auto-adjust to the screen size of whatever device is used to view your web pages. Finally, you'll learn to use Bootstrap, a popular and flexible styling framework.

- HTML 101
- CSS 101
- Box Model
- CSS Positioning
- Implementing a responsive design
- Flexbox
- Media Queries
- Bootstrap UI framework
- Hosting your code on surge.sh

Potential Project

- Creating a personal web page
- Making the Google search page
- Implementing a blog layout

Week 5 & 6: Handling User Input with JavaScript

Now that you can present information to your users, it is time for them to talk back! To provide this kind of interactivity on your web pages, you will begin mastering the basics of JavaScript - the language of the web. Using the same programming fundamentals you learned with Python, you will build a series of interactive games.

And, to make sure your friends and family can play these games, you will learn how to deploy your web-based games to Linux servers running on Amazon's AWS platform. This will be an intense week, but extremely rewarding!

- JavaScript 101
- Functions
- Scoping and hoisting
- Callbacks
- Forms and validation
- Deploying to Amazon Web Services

Potential Project

- Countdown timer
- Tic Tac Toe

Week 7 & 8: Building Interactive UIs with jQuery

jQuery is the most widely used JavaScript library in the world, and for good reason. You will learn how to use the power of jQuery to quickly develop highly interactive web pages. You will build more games this week, but using animations and transitions to provide visual feedback to your users.

There will be times you will need some, but not all, of jQuery's capabilities. You will learn how and when to use CSS for animation, as well as how to use Bootstrap's built-in visual feedback features.

- jQuery
- CSS transitions and animations
- Bootstrap.js
- Ajax
- Promises
- Web APIs

Potential Project

- Blackjack
- Photo gallery
- Weather widget
- Stock ticker

Week 9: SCRUM / Agile & Group Project Week

- Intro to SCRUM / Agile Development
- Google Auth

Project

- Front-End Group Project (groups mandatory)

Week 10 & 11: Introduction to Back-End Development

Up to now, you've only worked with half of the web - the half that works in the browser. To create web pages that allow for collaboration between users (multiplayer games, social media sites, etc.), you will need to write code that runs on the server. This week you will create a simple web server (yes, a web server!) using Node.js.

Just like JavaScript on the browser, there are a number of handy libraries that you can use. You will learn to install and manage those using the npm command that comes with Node.js.

One of these libraries will let you use WebSockets for building real-time applications. As an exercise, you'll build the backend and front-end of a chat application.

- Node.js
- Npm
- File IO
- async and sync APIs
- WebSockets

Potential Project

- Build a basic web server
- Build a chat application

Week 12 & 13: Databases

At the heart of any blog, social media site, or collaborative web app is its database. Often, your data will have complex relationships ("a blog post has an author and one or more comments from a registered user"). Dealing with this data is straightforward, as long as you can speak the language. This week, you will learn how to "talk" to a database using SQL (Structured Query Language).

- PostgreSQL
- Create statements
- Insert statements
- Select statements
- Where statements
- Foreign keys and joins
- Group by and aggregate statements

As your backend code gets larger and more complex, you will need ways to store data, organize your code, and develop quickly. This week you'll be using the Express.js framework, a minimalist back-end framework that will help you write backend code. You'll learn how to take advantage of Express.js' routing, templating, and session management.

- Express.js
- Templating
- Sessions
- Cookies

Potential Project

- Build your own wiki
- Intelligent flash card app

Week 14: Full Stack Group Project

Project

- Full Stack Group Project (groups optional but encouraged)

Week 15: Managing Application State with Redux

You will learn how to work with complex data in your applications using the industry-standard Redux library. Redux manages a one-way data flow and helps wrangle user interactions that cause complex changes to your data.

- Redux Intro
- The state tree
- Reducers
- Actions

Potential Project

- TBD

Week 16: Introducing React

React is a small but flexible front-end library that helps you create UIs. It is especially well-suited for UIs that need to display data that updates frequently. This week, you will learn the basics of building React applications, relying primarily on your understanding of JavaScript fundamentals.

You will start with simple functions that draw data to the page and build up your React knowledge. By the end of the week, you will be creating and (re)using highly-performant React components.

- Components
- JSX
- state and props
- Lifecycle API
- Virtual DOM
- Create-react-app
- React Dev Tools

Potential Project

- Secret message encoder
- Shopping list

- World Clock

Week 17: The React Ecosystem

React has made a lot of friends in its first few years of life. To write fluent React code that takes advantage of battle-tested libraries, you will want to be friends with React's friends. This week involves learning the basics of a lot of different tools that help you write high-quality React code.

- Create-react-app
- React Router
- Axios
- CSS-Modules

Potential Project

- Personal Dashboard app
- Live survey app

Week 18: Building an Ecommerce site with React and Redux

This week, you will put your React and Redux skills to use on a fairly complex web application. In addition, you will integrate with the Stripe payment API. (Don't worry, you won't go broke building and debugging your app - you'll only work with a testing server.)

- Stripe API
- Authentication

Potential Project

- Ecommerce site

Week 19 & 20: Project Week (group optional)

Projects

- Individual Advanced Front-end projects (groups are optional but highly encouraged)

Week 21: Review Week

Now is your chance to review over the huge amount of knowledge you've covered Together we will decide on what topics to review in class, or we could cover new technologies to prepare for the Capstone Project.

- Online Presence
- Review

Weeks 22 - 24: Capstone Project Project (group optional)

- Capstone Project (group optional)
- Demo Day