

# CPSC S115: Introduction to Full-Stack Web Development

## Pework & Module A

### Programming Fundamentals Ruby Basics and Object Oriented - Programming Syllabus

Unit	Learning Objectives	Reference Topics	Competency Based Assignments	Hours
<b>Pre-Work</b>				
1	<ul style="list-style-type: none"> <li>Understand the value and uses of version control with Git</li> <li>Complete the full Git project flow</li> <li>Work with the computer through the command line and BASH</li> </ul>	<ul style="list-style-type: none"> <li>Git                             <ul style="list-style-type: none"> <li>commit</li> <li>push</li> <li>pull</li> <li>remotes</li> <li>add</li> <li>fork</li> <li>merge</li> </ul> </li> <li>BASH                             <ul style="list-style-type: none"> <li>intro</li> <li>ls</li> <li>cd</li> <li>pipes</li> </ul> </li> </ul>	<ul style="list-style-type: none"> <li>Quiz: Git Basics</li> <li>Git Todo</li> <li>Git Flow</li> <li>Git Merge Conflicts</li> <li>Command Line Mystery</li> </ul>	15
2	<ul style="list-style-type: none"> <li>Create basic web sites using HTML</li> <li>Publish sites to different web hosts</li> <li>Understand the distinction between styling and content</li> </ul>	<ul style="list-style-type: none"> <li>Basic HTML                             <ul style="list-style-type: none"> <li>Lists</li> <li>Tables</li> <li>Images</li> <li>Links</li> <li>Validations</li> </ul> </li> <li>Site Publishing                             <ul style="list-style-type: none"> <li>Publishing to FTP</li> <li>Publishing to Github Pages</li> </ul> </li> <li>HTML5                             <ul style="list-style-type: none"> <li>Forms</li> <li>Media</li> </ul> </li> </ul>	<ul style="list-style-type: none"> <li>HTML Album Cover</li> <li>HTML Blog</li> <li>HTML Issue Bot 9000</li> <li>HTML Map Contact Forms Code Along</li> <li>HTML5 Video Codealong</li> <li>HTML5 Semantic Elements Codealong</li> </ul>	15
		<ul style="list-style-type: none"> <li>Video</li> <li>Semantic Elements</li> </ul>		
3	<ul style="list-style-type: none"> <li>Understand how to use CSS to style HTML elements</li> <li>Lay block objects out using the box model</li> <li>Create interfaces for mobile and web</li> </ul>	<ul style="list-style-type: none"> <li>CSS Selectors                             <ul style="list-style-type: none"> <li>Class</li> <li>Id</li> <li>Element type</li> </ul> </li> <li>Page Layout</li> </ul>	<ul style="list-style-type: none"> <li>CSS Kitten Wheelbarrow</li> <li>CSS Graffiti Override</li> <li>CSS Issue Bot 9000</li> </ul>	5

	<ul style="list-style-type: none"> <li>Use the Bootstrap framework to increase speed of development</li> </ul>	<ul style="list-style-type: none"> <li>Box model</li> <li>Overflow</li> <li>Fluid Height</li> <li>Float</li> <li>Centering</li> </ul>	<ul style="list-style-type: none"> <li>Creating Columns Codealong</li> <li>CSS Layout Quiz</li> <li>ZHW Shoes Layout</li> <li>Box Styles Codealong</li> <li>Adding Responsive Features Codealong</li> <li>Bootstrap Carousel Codealong</li> </ul>	
4	<ul style="list-style-type: none"> <li>Create basic decision trees</li> <li>Receive input from the Command Line</li> <li>Manipulate strings</li> <li>Work with Basic Looping</li> </ul>	<ul style="list-style-type: none"> <li>Conditionals <ul style="list-style-type: none"> <li>if</li> <li>switch/c ase</li> </ul> </li> <li>Looping <ul style="list-style-type: none"> <li>while</li> <li>for</li> <li>do...whil e</li> </ul> </li> <li>Command Line Input <ul style="list-style-type: none"> <li>gets</li> </ul> </li> <li>Manipulating strings <ul style="list-style-type: none"> <li>reverse</li> <li>upcase</li> <li>interpolat ion</li> <li>gsub</li> </ul> </li> </ul>	<ul style="list-style-type: none"> <li><a href="#">More Vowels</a></li> <li><a href="#">Badges and Schedules</a></li> <li><a href="#">Deli Counter</a></li> <li><a href="#">Phone Number Formatter</a></li> <li><a href="#">Cartoon Collections</a></li> <li><a href="#">Jukebox CLI</a></li> </ul>	20
5	<ul style="list-style-type: none"> <li>Why and when to use arrays</li> <li>Work effectively with adding and removing from arrays</li> </ul>	<ul style="list-style-type: none"> <li>Adding <ul style="list-style-type: none"> <li>Push <ul style="list-style-type: none"> <li>&lt;&lt;</li> </ul> </li> </ul> </li> <li>Removing <ul style="list-style-type: none"> <li>pop</li> </ul> </li> </ul>	<ul style="list-style-type: none"> <li><a href="#">My Each</a></li> <li><a href="#">Oxford Comma</a></li> <li><a href="#">Apples and</a></li> </ul>	20

	<ul style="list-style-type: none"> <li>Acting on arrays with iteration and introspection</li> <li>Retrieve, add and delete arrays nested inside other arrays</li> </ul>	<ul style="list-style-type: none"> <li>shift</li> <li>Introspection <ul style="list-style-type: none"> <li>flatten</li> <li>uniq</li> <li>include</li> <li>indexed access</li> </ul> </li> <li>Iteration <ul style="list-style-type: none"> <li>each</li> <li>map</li> <li>select</li> <li>detect</li> </ul> </li> <li>Nested Data <ul style="list-style-type: none"> <li>2D arrays</li> <li>multi-di mension arrays</li> </ul> </li> </ul>	<ul style="list-style-type: none"> <li><a href="#">Holidays</a></li> <li><a href="#">Song Sorter</a></li> <li><a href="#">Tweet Shortener</a></li> <li></li> <li><a href="#">Key for min value</a></li> <li><a href="#">Spotify API Lab</a></li> </ul>	
6	<ul style="list-style-type: none"> <li>Why and When to use Hashes</li> <li>Adding, removing and introspecting hashes</li> </ul>	<ul style="list-style-type: none"> <li>Adding <ul style="list-style-type: none"> <li>[]= method</li> </ul> </li> </ul>	<ul style="list-style-type: none"> <li><a href="#">Hashketball</a></li> <li><a href="#">NYC Pigeon Organizer</a></li> </ul>	25

	<ul style="list-style-type: none"> <li>Manipulating and normalizing data into nested hashes</li> </ul>	<ul style="list-style-type: none"> <li>Removing and introspection <ul style="list-style-type: none"> <li>select</li> <li>map</li> <li>detect</li> </ul> </li> <li>Nested Data</li> </ul>	<ul style="list-style-type: none"> <li>Apples and Holidays</li> </ul>	
<b>Module A</b>				
7	<ul style="list-style-type: none"> <li>Understand why Object Orientation was created</li> <li>Understand the benefits of Object Oriented Programming</li> <li>Differentiate between Instances and Classes</li> <li>Implement Methods and Properties in Classes</li> </ul>	<ul style="list-style-type: none"> <li>History of OO</li> <li>OO Patterns <ul style="list-style-type: none"> <li>Separation of Concerns</li> <li>Managing Internal State</li> </ul> </li> <li>Classes vs. Instances</li> <li>Classes have behaviors <ul style="list-style-type: none"> <li>Methods</li> </ul> </li> <li>Classes have attributes</li> </ul>	<ul style="list-style-type: none"> <li>Class and Instances Lab</li> <li>Instance Methods Lab</li> <li>Instance Variables Lab</li> <li>Object Attributes Lab</li> <li>OO Meowing Cat</li> </ul>	20
		<ul style="list-style-type: none"> <li>Instance Variables</li> <li>attr_accessor</li> <li>setters</li> <li>getters</li> </ul>		
8	<ul style="list-style-type: none"> <li>Understand the Object Lifecycle</li> <li>Create an initializer</li> </ul>	<ul style="list-style-type: none"> <li>Creating an init method <ul style="list-style-type: none"> <li>setting Instance Variables in init</li> <li>Other Defaults</li> </ul> </li> </ul>	<ul style="list-style-type: none"> <li>Object Initialize Lab</li> <li>OO Basics</li> </ul>	15
9	<ul style="list-style-type: none"> <li>Work with the concept of self</li> <li>Understand Class level constructs</li> </ul>	<ul style="list-style-type: none"> <li>Class Constants <ul style="list-style-type: none"> <li>Creation</li> <li>Why do we use them?</li> </ul> </li> <li>Self</li> <li>Class Variables <ul style="list-style-type: none"> <li>Why use them?</li> <li>Using as a collection of all created instances</li> </ul> </li> <li>Class Methods <ul style="list-style-type: none"> <li>understanding the all method</li> <li>Creating class methods</li> </ul> </li> <li>Private Methods <ul style="list-style-type: none"> <li>OO Programming as interface based programming</li> <li>How to define</li> </ul> </li> </ul>	<ul style="list-style-type: none"> <li>OO Basics w/ Class Constants</li> <li>OO School Domain</li> <li>OO Counting Sentences</li> <li>Class Variables and Methods Lab</li> <li>Ruby Puppy</li> <li>Advanced Class Methods Lab</li> </ul>	20
		access levels		
10	<ul style="list-style-type: none"> <li>Be able to model a domain with OO principles</li> <li>Work with larger projects and chunk them into smaller segments</li> </ul>	<ul style="list-style-type: none"> <li>Domain Modeling <ul style="list-style-type: none"> <li>Deciding on properties</li> <li>Deciding on methods</li> </ul> </li> </ul>	<ul style="list-style-type: none"> <li>OO Email Parser</li> <li>Anagram detector</li> <li>OO Cash Register</li> </ul>	10

		<ul style="list-style-type: none"> <li>• What should be public vs. private</li> </ul>		
11	<ul style="list-style-type: none"> <li>• Understand the power of relating objects</li> <li>• Use a belongs to relationship</li> <li>• Use a has many relationship</li> <li>• Collaborate between two objects</li> <li>• Use a has many through relationship</li> </ul>	<ul style="list-style-type: none"> <li>• Why Relationships are important</li> <li>• Belongs To <ul style="list-style-type: none"> <li>• Adding to all class variable</li> <li>• Domain modeling a belongs to</li> </ul> </li> <li>• Has Many <ul style="list-style-type: none"> <li>• Reciprocal relationships</li> <li>• Domain modeling a belongs to</li> <li>• Adding to class variables in both directions</li> </ul> </li> <li>• Collaborating Objects <ul style="list-style-type: none"> <li>• Relation</li> </ul> </li> </ul>	<ul style="list-style-type: none"> <li>• Belongs to Object Lab</li> <li>• Has Many Objects Lab</li> <li>• Collaborating Objects Lab</li> <li>• OO My Pets</li> <li>• OO Kickstarter</li> <li>• OO Banking</li> <li>• Has Many Objects Through Lab</li> </ul>	25
		<p>ships that aren't ownership</p> <ul style="list-style-type: none"> <li>• When to separate behavior s/attributes into new classes</li> <li>• Has Many Through <ul style="list-style-type: none"> <li>• Domain Modeling Has Many Through</li> <li>• Has Many Through just two different has many relationships</li> </ul> </li> </ul>		
12	<ul style="list-style-type: none"> <li>• Understand the value of mass assignment</li> <li>• Use simple meta programming</li> </ul>	<ul style="list-style-type: none"> <li>• Mass Assignment <ul style="list-style-type: none"> <li>• Problem with manual assignment</li> <li>• Passing in an options hash</li> </ul> </li> <li>• Meta programming <ul style="list-style-type: none"> <li>• Using the send method</li> <li>• Using send to implement mass assignment</li> </ul> </li> </ul>	<ul style="list-style-type: none"> <li>• Mass Assignment Lab</li> <li>• Custom Errors</li> <li>• OO Triangle</li> </ul>	20
13	<ul style="list-style-type: none"> <li>• Grab data from websites using Scraping</li> <li>• Integrate third party code using gems</li> <li>• Gem version control with Bundler</li> </ul>	<ul style="list-style-type: none"> <li>• Gems and Bundler <ul style="list-style-type: none"> <li>• Where to find gems</li> <li>• Open Source Licensing</li> <li>• How to choose gems</li> <li>• Installing and requiring gems</li> <li>• Using a Gemfile</li> </ul> </li> <li>• Scraping <ul style="list-style-type: none"> <li>• Nokogiri</li> </ul> </li> </ul>	<ul style="list-style-type: none"> <li>• Scraping Flatiron</li> <li>• Scraping Kickstarter</li> </ul>	5

		<ul style="list-style-type: none"> <li>Using CSS Selectors</li> </ul>		
14	<ul style="list-style-type: none"> <li>Working in large ruby projects</li> <li>Understand how to break large problems down into smaller, workable chunk</li> </ul>	<ul style="list-style-type: none"> <li>Domain Modeling</li> <li>Task Breakdown practice</li> </ul>	<ul style="list-style-type: none"> <li>Playlister CLI</li> <li>Student Scraper</li> <li>Tic Tac Toe with AI</li> <li>CLI Data Gem Assessment</li> </ul>	5

## Module B

### Web Frameworks: Web Development with Rails Syllabus

Unit	Learning Objectives	Reference Topics	Competency Based Assignments	Hours
1	<ul style="list-style-type: none"> <li>Understand the uses of SQL</li> <li>Understand the layout of basic databases</li> <li>Install and run basic commands with SQLite</li> <li>Use Aggregate functions</li> </ul>	<ul style="list-style-type: none"> <li>SQL Introduction</li> <li>Installing SQLite</li> <li>SQL Basics <ul style="list-style-type: none"> <li>SELECT</li> <li>INSERT</li> <li>UPDATE</li> <li>CREATE</li> <li>ALTER</li> </ul> </li> <li>Using .sql files</li> <li>SQLite data types</li> <li>Aggregate Functions <ul style="list-style-type: none"> <li>SUM</li> <li>COUNT</li> <li>MAX</li> <li>MIN</li> </ul> </li> </ul>	<ul style="list-style-type: none"> <li>Bear Organizer</li> <li>Aggregate Functions Lab</li> </ul>	5
2	<ul style="list-style-type: none"> <li>Understand what a JOIN is</li> <li>Understand foreign keys in has many/belongs to relationships</li> <li>Understand Complex joins and Join tables</li> </ul>	<ul style="list-style-type: none"> <li>Relational Databases <ul style="list-style-type: none"> <li>History</li> <li>Relations as related to ruby object relations</li> </ul> </li> <li>JOIN Statements <ul style="list-style-type: none"> <li>foreign keys</li> <li>Left Outer Join</li> <li>Left Inner Join</li> </ul> </li> <li>Sorting and Grouping Data <ul style="list-style-type: none"> <li>ORDER BY</li> <li>GROUP BY</li> </ul> </li> <li>Complex Joins <ul style="list-style-type: none"> <li>Many-to-Man y</li> </ul> </li> </ul>	<ul style="list-style-type: none"> <li>SQL Crowdfunding Lab</li> <li>SQL Library Lab</li> </ul>	10
		<ul style="list-style-type: none"> <li>relationships</li> <li>Join tables</li> </ul>		
	<ul style="list-style-type: none"> <li>Understand how ruby can talk to SQLite Databases</li> </ul>	<ul style="list-style-type: none"> <li>SQLite gem <ul style="list-style-type: none"> <li>Execute</li> <li>create table</li> <li>results</li> </ul> </li> </ul>	<ul style="list-style-type: none"> <li>Pokemon Scraper</li> </ul>	10

4	<ul style="list-style-type: none"> <li>• Understand the value and power of ORMs</li> <li>• build a simple ORM</li> <li>• Work with the inheritance and meta-programming required for ORMs</li> </ul>	<ul style="list-style-type: none"> <li>• Mapping Objects to tables <ul style="list-style-type: none"> <li>• Working with string interpolating INSERT</li> <li>• CREATE Table from attributes</li> </ul> </li> <li>• Mapping Tables to Objects <ul style="list-style-type: none"> <li>• Dynamically create attributes</li> <li>• all method</li> <li>• find methods</li> </ul> </li> <li>• Bi-Directional mapping <ul style="list-style-type: none"> <li>• UPDATE</li> <li>• save checking for persistence</li> </ul> </li> <li>• Dynamic ORMs <ul style="list-style-type: none"> <li>• meta programming attributes</li> <li>• Refactoring common behavior to super</li> </ul> </li> </ul>	<ul style="list-style-type: none"> <li>• Mapping to Table Lab</li> <li>• Mapping DB to Ruby Lab</li> <li>• ORM update Lab</li> <li>• ORM Bringing it all together</li> <li>• Dynamic ORM Lab</li> </ul>	10
	<ul style="list-style-type: none"> <li>• Use ActiveRecord as an ORM</li> <li>• Understand all the features ActiveRecord provides</li> <li>• Use Rake tasks to perform database migrations</li> </ul>	<ul style="list-style-type: none"> <li>• ORM To Active Record <ul style="list-style-type: none"> <li>• Translate previous custom ORM methods into ActiveRecord method</li> <li>• Use ActiveRecord ::Base</li> </ul> </li> </ul>	<ul style="list-style-type: none"> <li>• Translating ORM to AR</li> <li>• ActiveRecord Create, Read, Update, Delete</li> <li>• Rake Lab</li> <li>• Writing Migrations</li> <li>• ActiveRecord Costume Store</li> <li>• Activerecord</li> </ul>	15

		<ul style="list-style-type: none"> <li>• Rake <ul style="list-style-type: none"> <li>• Create custom rake tasks</li> <li>• Explain Rakefile</li> </ul> </li> <li>• Migrations <ul style="list-style-type: none"> <li>• Explain value of migrations</li> <li>• Write change migrations</li> <li>• Run migrations without rake</li> <li>• Run migrations with rake</li> </ul> </li> </ul>	TV-show	
--	--	---	---------	--

6	<ul style="list-style-type: none"> <li>• Understand associations in ActiveRecord</li> <li>• Map ActiveRecord association to previous associations in ruby</li> </ul>	<ul style="list-style-type: none"> <li>• ActiveRecord Macros <ul style="list-style-type: none"> <li>• has many</li> <li>• belongs to</li> <li>• has many :through</li> </ul> </li> <li>• migrations with relationships</li> </ul>	<ul style="list-style-type: none"> <li>• Activerecord- TVland</li> </ul>	5
---	--	---	--	---

7	<ul style="list-style-type: none"> <li>• Understand the history and current status of HTTP</li> <li>• Work with the HTTP Request cycle in ruby and Rack</li> <li>• Understand Rack as HTTP Middleware</li> </ul>	<ul style="list-style-type: none"> <li>• Working with the Internet <ul style="list-style-type: none"> <li>• HTTP Verbs</li> <li>• What is a protocol</li> <li>• HTTP Request headers</li> <li>• URLs</li> </ul> </li> <li>• HTTP Responses</li> </ul>	<ul style="list-style-type: none"> <li>• Rack Todo</li> <li>• Rack Lab</li> </ul>	5
---	--	---	---	---

		<ul style="list-style-type: none"> <li>Params</li> <li>accessing params</li> <li>Rack <ul style="list-style-type: none"> <li>Using the params hash</li> <li>Value of middleware</li> </ul> </li> </ul>		
8	<ul style="list-style-type: none"> <li>Implement the HTTP Request/Response Cycle with Sinatra</li> <li>Understand the value of MVC and how to</li> </ul>	<ul style="list-style-type: none"> <li>Sinatra in app.rb</li> <li>Sinatra as a gem</li> <li>Simple GET requests in</li> </ul>	<ul style="list-style-type: none"> <li>Sinatra Hello World</li> <li>Sinatra Basic Routes Lab</li> <li>Sinatra Basic</li> </ul>	10
	<p>implement it in Sinatra</p> <ul style="list-style-type: none"> <li>Render front end with ERB templating</li> </ul>	<p>sinatra</p> <ul style="list-style-type: none"> <li>Moduler Sinatra applications</li> <li>Shotgun</li> <li>Sinatra Routing <ul style="list-style-type: none"> <li>Different HTTP Verbs in Sinatra</li> </ul> </li> <li>MVC <ul style="list-style-type: none"> <li>Values of MVC</li> <li>Separation of Concerns</li> <li>How MVC works in Sinatra</li> </ul> </li> <li>Views <ul style="list-style-type: none"> <li>Templating</li> <li>in line ruby</li> </ul> </li> </ul>	<p>Views Lab</p> <ul style="list-style-type: none"> <li>Sinatra Views Lab</li> </ul>	
9	<ul style="list-style-type: none"> <li>Understand how to gather user input through forms</li> <li>Work with form data to create responses</li> <li>Create and parse nested forms to create multiple models in one submit</li> </ul>	<ul style="list-style-type: none"> <li>Dynamic Routes <ul style="list-style-type: none"> <li>Symbol routing syntax</li> <li>params hash</li> </ul> </li> <li>Basic Forms <ul style="list-style-type: none"> <li>Create HTML Forms</li> <li>Post those forms to your application</li> <li>Parse form data to create model objects</li> </ul> </li> <li>MVC with forms <ul style="list-style-type: none"> <li>full flow from form to controller, to model creation</li> </ul> </li> <li>Nested Forms <ul style="list-style-type: none"> <li>Using the name attribute of input elements</li> <li>discuss the</li> </ul> </li> </ul>	<ul style="list-style-type: none"> <li>Dynamic Routes Lab</li> <li>Basic Sinatra Forms Lab</li> <li>Sinatra MVC Lab</li> <li>Sinatra Nested Forms</li> <li>Super Sinatra Forms</li> </ul>	15
		<p>value of a nested form</p> <ul style="list-style-type: none"> <li>use the nested params hash.</li> </ul>		
10	<ul style="list-style-type: none"> <li>Be able to persist small amounts of data to the browser with sessions</li> <li>Understand the basic flow of logging in/out</li> </ul>	<ul style="list-style-type: none"> <li>Browser data <ul style="list-style-type: none"> <li>What are sessions</li> <li>What are cookies</li> <li>inspecting cookies and sessions in a browser</li> </ul> </li> <li>Sessions in Sinatra <ul style="list-style-type: none"> <li>Setting the sessions hash</li> </ul> </li> <li>Authentication <ul style="list-style-type: none"> <li>Explain registration</li> </ul> </li> </ul>	<ul style="list-style-type: none"> <li>Basic Sessions Lab</li> <li>Sinatra Logging in and logging out</li> <li>Sinatra Sessions</li> </ul>	5

		<ul style="list-style-type: none"> <li>• Login creates a session</li> <li>• logout destroys the session</li> </ul>		
11	<ul style="list-style-type: none"> <li>• How to integrate active record into sinatra</li> <li>• Using multiple controllers in sinatra</li> </ul>	<ul style="list-style-type: none"> <li>• ActiveRecord and Sinatra <ul style="list-style-type: none"> <li>• Migrations</li> <li>• rake and sinatra</li> </ul> </li> <li>• REST <ul style="list-style-type: none"> <li>• mapping routes to activerecord actions</li> <li>• choosing routes constructively</li> </ul> </li> <li>• Multiple Controllers <ul style="list-style-type: none"> <li>• mechanics of multiple controllers</li> <li>• How controller choose their view</li> </ul> </li> </ul>	<ul style="list-style-type: none"> <li>• RESTful routes lab</li> <li>• Sinatra Messages</li> <li>• Playlist Sinatra</li> <li>• NYC Sinatra</li> </ul>	15

		<ul style="list-style-type: none"> <li>• ActiveRecord associations in Sinatra <ul style="list-style-type: none"> <li>• join tables</li> <li>• mapping tables to controllers</li> </ul> </li> </ul>		
--	--	--	--	--

12	<ul style="list-style-type: none"> <li>• Translate Sinatra concepts into Rails concepts</li> <li>• Work with Rails File System to serve files using MVC</li> </ul>	<ul style="list-style-type: none"> <li>• Sinatra to Rails <ul style="list-style-type: none"> <li>• Router</li> <li>• Controllers</li> <li>• Models</li> <li>• Views</li> </ul> </li> <li>• Static Requests <ul style="list-style-type: none"> <li>• ERB rendering</li> <li>• Rails generator file structure</li> </ul> </li> </ul>	<ul style="list-style-type: none"> <li>• Rails Hello World Lab</li> </ul>	5
----	--	--	---	---

13	<ul style="list-style-type: none"> <li>• Understand Routing and REST</li> <li>• Create routes that connect to the appropriate action</li> </ul>	<ul style="list-style-type: none"> <li>• RESTful Routing <ul style="list-style-type: none"> <li>• method definitions as http verbs</li> <li>• what are the different methods</li> </ul> </li> <li>• Dynamic Routing <ul style="list-style-type: none"> <li>• Similarities to Sinatra</li> <li>• How dynamic routing fits into REST</li> </ul> </li> </ul>	<ul style="list-style-type: none"> <li>• RESTful index action lab</li> <li>• Dynamic Request Lab</li> </ul>	15
----	---	---	---	----

14	<ul style="list-style-type: none"> <li>• Create dynamic views with ActionView</li> <li>• Collect user input using form_for and rails scaffold</li> </ul>	<ul style="list-style-type: none"> <li>• Rails URL Helpers <ul style="list-style-type: none"> <li>• difference between path and route</li> <li>• Using this with link_to</li> </ul> </li> <li>• Create Action <ul style="list-style-type: none"> <li>• manual form tags</li> <li>• POST action</li> </ul> </li> <li>• Edit/Update Action <ul style="list-style-type: none"> <li>• difference between form and</li> </ul> </li> </ul>	<ul style="list-style-type: none"> <li>• Rails URL Helper Lab</li> <li>• Rails Form Tag Lab</li> <li>• Rails Create Action Lab</li> <li>• Rails Index/Show/New/Create Lab</li> <li>• Rails form_for Lab</li> <li>• Formal Affair Rails</li> <li>• Rails form_for</li> </ul>	20
----	--	--	---	----

		<ul style="list-style-type: none"> <li>• Form_for</li> </ul>	<ul style="list-style-type: none"> <li>• Rails Blog Scaffold</li> </ul>	
--	--	--	---	--



		<ul style="list-style-type: none"> <li>• using automated view helpers</li> <li>• passing in instance variables to helpers</li> <li>• assumed routes</li> <li>• Generators <ul style="list-style-type: none"> <li>• when to use scaffold</li> <li>• how to generate scaffold</li> <li>• understanding all the scaffolding creates</li> </ul> </li> </ul>		
15	<ul style="list-style-type: none"> <li>• Understand the power of validations</li> <li>• Create custom validations</li> <li>• Work with built in validations</li> <li>• Display validation errors to users</li> </ul>	<ul style="list-style-type: none"> <li>• Validation callbacks <ul style="list-style-type: none"> <li>• presence</li> <li>• uniqueness</li> <li>• numeric</li> <li>• regex matching</li> </ul> </li> <li>• Custom validations</li> <li>• Errors <ul style="list-style-type: none"> <li>• The @errors variable</li> <li>• Describing errors in form_for</li> <li>• resolving errors</li> </ul> </li> </ul>	<ul style="list-style-type: none"> <li>• Blog Associations</li> <li>• Validations</li> </ul>	10
16	<ul style="list-style-type: none"> <li>• Use ActiveRecord in Rails</li> <li>• Understand complex many to many through</li> <li>• Using AR to search and sort through data</li> </ul>	<ul style="list-style-type: none"> <li>• Associations <ul style="list-style-type: none"> <li>• through</li> <li>• Domain modeling</li> <li>• Working with JOIN tables</li> </ul> </li> <li>• Searching <ul style="list-style-type: none"> <li>• Where</li> <li>• Find</li> <li>• ordering</li> <li>• AREL</li> </ul> </li> </ul>	<ul style="list-style-type: none"> <li>• Pollywog Ranch</li> <li>• Simple Search</li> <li>• Arel Labs</li> <li>• BnB Associations</li> </ul>	25
17	<ul style="list-style-type: none"> <li>• Understand the power of refactoring to components</li> <li>• Use partials for collections, forms and other reusable views</li> <li>• Use helpers to remove logic from views</li> </ul>	<ul style="list-style-type: none"> <li>• Partials <ul style="list-style-type: none"> <li>• Collection Partials</li> <li>• Form Partials</li> <li>• Cross Controller Partials</li> </ul> </li> <li>• Helpers <ul style="list-style-type: none"> <li>• Presenter Pattern</li> <li>• Logic in Views</li> <li>• Cross controller helpers</li> </ul> </li> </ul>	<ul style="list-style-type: none"> <li>• Flatiron BNB Methods</li> <li>• Programmer Profile Partials</li> </ul>	10

## Module C

### Javascript: Creating Interactive and Performant Front Ends with Javascript Syllabus

Unit	Learning Objectives	Reference Topics	Competency Based Assignments	Hours
1	<ul style="list-style-type: none"> <li>Understand the similarities between ruby and javascript</li> <li>Know when javascript is the correct language to use</li> <li>Use Javascript to do basic programming tasks</li> </ul>	<ul style="list-style-type: none"> <li>JS Basics               <ul style="list-style-type: none"> <li>How to run JS code</li> <li>How to read JS tests</li> </ul> </li> <li>JS Data Types               <ul style="list-style-type: none"> <li>Hashes</li> <li>Variables</li> <li>Functions</li> <li>Strings</li> <li>Math</li> <li>Object Literals</li> </ul> </li> <li>JS Control Flow               <ul style="list-style-type: none"> <li>if statements</li> <li>looping</li> </ul> </li> </ul>	<ul style="list-style-type: none"> <li>JS Debugging</li> <li>JS isBetween</li> <li>FizzBuzz</li> <li>Donut Lab</li> </ul>	20
2	<ul style="list-style-type: none"> <li>Use closures to modify variable scope</li> <li>Understand prototypical inheritance</li> </ul>	<ul style="list-style-type: none"> <li>Closures               <ul style="list-style-type: none"> <li>Lexical scope</li> <li>Scoping with Var</li> <li>Anonymous Functions</li> </ul> </li> <li>Inheritance               <ul style="list-style-type: none"> <li>Prototypes</li> <li>Constructors</li> <li>Classes as Hashes</li> </ul> </li> </ul>	<ul style="list-style-type: none"> <li>Space Invaders</li> <li>JS Tweet Shortener</li> <li>Triangle.js</li> <li>Robot Simulator</li> <li>JS Tic Tac Toe</li> </ul>	10
3	<ul style="list-style-type: none"> <li>Understand when and how to use jQuery</li> <li>Work with other Libraries</li> <li>Use Selectors in JS</li> </ul>	<ul style="list-style-type: none"> <li>Basic jQuery               <ul style="list-style-type: none"> <li>JS Libraries</li> <li>History of jQuery</li> <li>the \$ selector</li> </ul> </li> <li>CSS Selectors in JS               <ul style="list-style-type: none"> <li>Selectors</li> <li>iteration</li> <li>Convenience Methods</li> </ul> </li> </ul>	<ul style="list-style-type: none"> <li>jQuery Calculator</li> <li>jQuery OO Tic Tac Toe</li> <li>Task Lister</li> <li>Flickr Image Search</li> </ul>	10
4	<ul style="list-style-type: none"> <li>Query and parse a JSON API</li> <li>Update on screen content dynamically</li> </ul>	<ul style="list-style-type: none"> <li>AJAX               <ul style="list-style-type: none"> <li>JSON parsing</li> <li>Using \$.ajax</li> <li>Async Javascript</li> <li>Displaying results</li> </ul> </li> <li>JSON</li> </ul>	<ul style="list-style-type: none"> <li>Temperature Visualizer</li> <li>Spotify with AJAX</li> <li>Inflation Calculator</li> <li>Cars Lazy Loading</li> </ul>	30

		<ul style="list-style-type: none"> <li>Understanding APIs</li> <li>Reading API Documentation</li> </ul>		
5	<ul style="list-style-type: none"> <li>Intro to how rails talks to javascript</li> <li>Optimize your javascript with Asset Pipeline</li> <li>How to build APIs for Javascript consumption</li> </ul>	<ul style="list-style-type: none"> <li>Consuming APIs from Ruby <ul style="list-style-type: none"> <li>Using Faraday</li> <li>Popular Gems</li> </ul> </li> <li>Building APIs <ul style="list-style-type: none"> <li>Returning Basic Strings</li> <li>JSON Serialization</li> <li>Using Built in to_json</li> <li>Active Model Serializer</li> </ul> </li> </ul>	<ul style="list-style-type: none"> <li>JS Manifests Lab</li> <li>CSS Manifests Lab</li> <li>Bootstrap Example Rails Lab</li> <li>Basic APIs Lab</li> <li>Refactoring APIs Lab</li> <li>DIY JSON Serializer Lab</li> <li>Using to_json Lab</li> <li>Receiving API POSTs Lab</li> <li>jQuery Tic Tac Toe</li> <li>Rails App with a jQuery Front End</li> </ul>	15

## 2Week Student Projects Syllabus

During this portion of the course students are tasked with working solo to create a project. This course provides an in-depth opportunity for student to demonstrate their learning accomplishments and get a feel for what working at a company is really like.

Students are divided randomly and begin with an initial pitch session. Students are tasked with pitching three different ideas that they are passionate about. Instructors take the pitches and then choose the final project that the students work on. Instructors choose projects based on difficulty and feasibility given the time constraints of the course. Specifically, the projects are required to have the following items to meet the minimum bar of difficulty:

- Must work directly with an API
- Must have > 3 Models and Controllers
- Must use helper methods to remove logic from views
- Must include at least two complex joins
- Must have some AJAX functionality

Unit	Learning Objectives	InClass Lectures	Lab Activities
1	<ul style="list-style-type: none"> <li>Understand developer flow with GIT</li> </ul>	<ul style="list-style-type: none"> <li>Git <ul style="list-style-type: none"> <li>branches</li> <li>Feature Branches</li> <li>Advanced Merge Conflict</li> </ul> </li> <li>GitHub <ul style="list-style-type: none"> <li>Pull Requests</li> <li>Issues</li> </ul> </li> </ul>	none
2	<ul style="list-style-type: none"> <li>Break a large project into workable chunks</li> </ul>	<ul style="list-style-type: none"> <li>Minimum Viable Products <ul style="list-style-type: none"> <li>Choosing priority features</li> <li>User Stories</li> </ul> </li> <li>Project Management <ul style="list-style-type: none"> <li>Dividing work</li> <li>Agile Workflow</li> </ul> </li> </ul>	none