

Course Overview

Course Overview

Foundations

Foundations is a learner-driven remote course that upskills and prepares you for our onsite bootcamp programme. It consists of two modules (core and tech) and is designed to be undertaken full time (40 hours a week) for 5 weeks. We use a number of freely available online resources, which we structure in a coherent sequence for student learning. We also provide online support and regular scheduled catch ups with a Foundations Facilitator.

Foundations consists of five sprints which cover:

Sprint 1 <ul style="list-style-type: none">- Git and GitHub basics- Command line basics- Intro to Core material- Values and Identity exploration	Sprint 2 <ul style="list-style-type: none">- Intro to HTML and CSS- Responsive design- Git and HTML best practice- Create a Learning Plan	Sprint 3 <ul style="list-style-type: none">- The DOM- Intro to JavaScript- Using the Developer Tools- Emotional Intelligence
Sprint 4 <ul style="list-style-type: none">- Combining HTML, CSS and JS- Problem solving techniques- Neuroplasticity and growth mindset- How to search and ask for help	Sprint 5 <ul style="list-style-type: none">- Managing user inputs- JavaScript event handling- Dynamic DOM updates	

Course Overview

Phase 1

The primary focus of Phase 1 is server-side development. We start by ensuring everyone is close to the same technical level and move through automated testing practices to server-side development. By the end of Phase 1 students are building database-driven web applications with dynamic page rendering. These applications are capable of adding, updating, deleting, and displaying database data in dynamic pages built with a server-side templating language. These applications are also deployed to the web and capable of solving problems for real users.

Week 1	Week 2	Week 3
<ul style="list-style-type: none">- Arrival- Ensure everyone is on the same level- Recap JavaScript fundamentals (object, arrays, functions, etc.)- Test-driven development- Internet and web fundamentals (HTTP requests/responses, DNS, etc.)- Server-side development with Node.js and Express.js	<ul style="list-style-type: none">- Server-side templating with Handlebars.js- Filesystem access with Node's `fs`- Asynchronous development using JavaScript callbacks- Unit testing synchronous and asynchronous functions- Reading & writing data to and from files on the server- Exception management and HTTP status codes- ES6 syntax (fundamentals)	<ul style="list-style-type: none">- Database development with SQLite3 and Knex.js- Database operations (Create, Read, Update, and Delete)- Database maintenance using migrations and seed data- Data access patterns with web applications- Asynchronous functions using Promises- Data modeling entity relationships- Object-oriented programming

Course Overview

Phase 2

In Phase 2 students are becoming familiar with the intensity and the culture of Dev Academy. Our technical focus transitions back to the browser and more modern web development solutions. We start by jumping straight into the React JavaScript library. After gaining some comfort with the new concepts, we begin to expose and consume web APIs that return JSON instead of rendered views. Finally, we improve the architecture of our frontend code by managing state with Redux. Phase 2 also has students presenting lightning talks and their personal projects.

Week 4	Week 5	Week 6
<ul style="list-style-type: none">- JSX syntax and why it's great- Using React to build components- More ES6 syntax- Bundling client assets with Webpack- Functional programming concepts	<ul style="list-style-type: none">- REST and API architectures- API routes and Postman- Task runners and bundlers- Superagent, and Supertest- Object-orientation and prototypal inheritance	<ul style="list-style-type: none">- Unit testing React components- Managing state with Redux- Adding and validating JavaScript type metadata- Client-side routing

Course Overview

Phase 3

By the time students get to Phase 3 they are increasingly self-directed and learning topics outside of the official curriculum. Phase 3 provides space for students to combine and begin to integrate what they've learned throughout the bootcamp. We explore a few advanced topics and give them an opportunity to explore and present new technologies: languages, libraries and frameworks. Phase 3 is also includes multiple multi-day projects and includes the fundamentals of agile development tools and practices such as scrum, Kanban, sprint planning, stand-ups, retrospectives and a prioritised backlog.

Week 7	Week 8	Week 9
<ul style="list-style-type: none"> - OAuth, authentication and authorization - Universal JavaScript: using React/Redux on the server - More time using and testing React and Redux - More functional JavaScript concepts - Custom lectures based on student interest 	<ul style="list-style-type: none"> - Multiple multi-day projects (teachers accept dev lead role) - Thunder talks (20 min presentations with demos and code) - Begin final group projects 	<ul style="list-style-type: none"> - Building and presenting final projects