Next.js

Baseline Training

John Whaley

10/8/2022

Table of contents

[1 Introduction 2](#_Toc116131027)

[2 Goals and Benefits 2](#_Toc116131028)

[3 Prerequisites 2](#_Toc116131029)

[4 Project 2](#_Toc116131030)

[5 Course Outline 2](#_Toc116131031)

[1.1. Getting Started 2](#_Toc116131032)

[1.2. Server Side Rendering (SSR) and Static Generation (SSG) 2](#_Toc116131033)

[1.3. React Fundamentals 2](#_Toc116131034)

[1.4. Redux 2](#_Toc116131035)

[6 Member’s Responsibilities 2](#_Toc116131036)

[7 Reviewer’s Responsibilities 2](#_Toc116131037)

[8 Schedule Time Line 2](#_Toc116131038)

# Introduction

Upon completion of this course, an engineer should be able to demonstrate a basic core level of competence in the technology.

# Goals and Benefits

The goal of this course is to provide members with an understanding of the fundamentals to Next.js development and solution architecture. Through the completion of this course, members will develop a solid understanding of Next.js architecture and solution development basics. Completing the course and attaining endorsement from the reviewer will aid in CGI’s resource management and allocation, as management will have a more quantifiable way of measuring member’s skill levels.

The benefits of receiving the Next.js course endorsement from the reviewer to the member include:

* Expansion of technical knowledge
* Recognition within the organization

The benefits of receiving the Next.js course endorsement from the reviewer to CGI include:

* Fundamental architecture knowledge of Next.js across the consulting pool
* Accurate evaluation of Next.js skills across the consulting pool
* An increased culture of information sharing and peer accountability in the consulting pool

# Prerequisites

* Bootcamp
* General working knowledge of JavaScript and React.js
* React 101 Technical Training (preferred, but not required)

# Project

Develop a “NextJS” app that has two routes with the following functionality:

1. “About Me” page with the following details:
   * A Picture of yourself
   * Your Name
   * Where you went to College/University
   * An interesting fact about yourself or something you enjoy doing in your free time
   * How long you’ve worked at CGI
   * “Contact Me” form with the following fields:
     1. Name (text input)
     2. Email (text input)
     3. Message (text area)
     4. Save and Reset Buttons
   * The contact me form should send a message to a faked API containing the form data. That API should respond with a message saying “Sent” after receiving the request. The response “Sent” should show up somewhere near the form after the response has been received.
   * Clicking the “Reset” Button should reset the form back to its default values and make the “Sent” message go away
   * The form should use Validation (Bonus points for using [React Hook Form](https://react-hook-form.com/)) for at least one field in the form.
2. “API” page that utilizes a [Free Online API](https://medium.com/geekculture/10-fun-and-free-apis-to-use-for-your-next-coding-project-7d765f643f08) or your own API with a backend to display some information from a third party source
   * This page should contain at least 2 requests (example from the NASA Open API would be one GET to get a picture of a planet of the day and another GET to retrieve some information from the natural event tracker)
   * Information about what you are retrieving from that API.
   * Utilize a Grid Layout with CSS modules or Inline Styling
3. Both of these routes should utilize a common “Layout Component” that allows for navigation between the two pages.
4. The Application should utilize at least one example of Inline Styling and one example of CSS/SCSS modules.
5. The Application should use at least one common component (not including the layout)
6. The Application must have at least one Page that uses Server-Side Rendering or Static Generation (can use both but not required).

# Course Outline

In order to demonstrate competency in NextJS a member must demonstrate knowledge of the following topics:

* 1. Getting Started
* [*Official Next.js Starting Documentation*](https://nextjs.org/docs/getting-started)
* [*Typescript vs. JavaScript*](https://www.codecademy.com/resources/blog/typescript-vs-javascript/)
* [*NextJs vs ReactJS*](https://www.geeksforgeeks.org/nextjs-vs-reactjs-which-one-to-choose/#:~:text=React%20is%20a%20library%2C%20not,learn%20without%20prior%20ReactJS%20knowledge)
* [*CSS Modules*](https://nextjs.org/docs/basic-features/built-in-css-support#adding-component-level-css) */* [*Inline Styling*](https://nextjs.org/docs/basic-features/built-in-css-support#css-in-js)
  1. Server-Side Rendering (SSR) and Static Generation (SSG)
* [*Web Design Patterns - SSG, SSR, SPA*](https://medium.com/codex/web-design-patterns-ssr-ssg-and-spa-fadad7673dfe)
* [*Two Forms of Pre-Rendering in Next.js*](https://nextjs.org/docs/basic-features/pages#two-forms-of-pre-rendering)
* [*Pages in Next.js*](https://nextjs.org/docs/basic-features/pages) */* [*Data Fetching Overview*](https://nextjs.org/docs/basic-features/data-fetching/overview)
* [*getServerSideProps*](https://nextjs.org/docs/basic-features/data-fetching/get-server-side-props) */* [*getStaticProps*](https://nextjs.org/docs/basic-features/data-fetching/get-static-props)
  1. React Fundamentals
* [*Introducing JSX*](https://reactjs.org/docs/introducing-jsx.html)
* [*Rendering Elements*](https://reactjs.org/docs/rendering-elements.html)
* [*What is JSX?*](https://www.fullstackreact.com/30-days-of-react/day-2/)
* [*React JSX*](https://www.javatpoint.com/react-jsx)
* [*Components and Props*](https://reactjs.org/docs/components-and-props.html)
* [*State and Lifecycle*](https://reactjs.org/docs/state-and-lifecycle.html)
* [*Introducing Hooks*](https://reactjs.org/docs/hooks-intro.html)
* [*Hooks at a Glance*](https://reactjs.org/docs/hooks-overview.html)
* [*Using the State Hook*](https://reactjs.org/docs/hooks-state.html)
* [*Using the Effect Hook*](https://reactjs.org/docs/hooks-effect.html)
* [*React Hook Form*](https://react-hook-form.com/)
  1. Redux
* [*Getting Started with Redux*](https://redux.js.org/introduction/getting-started)
* [*Data Management with Redux*](https://www.fullstackreact.com/30-days-of-react/day-19/)
* [*Redux Actions*](https://www.fullstackreact.com/30-days-of-react/day-20/)
* [*Redux Middleware*](https://www.fullstackreact.com/30-days-of-react/day-21/)
* [*Redux in NextJS*](https://blog.logrocket.com/use-redux-next-js/) *…*

# Member’s Responsibilities

A member will be required to demonstrate knowledge of all items identified in the Course Outline section of this document above. Additionally, the member must also create a NextJS for the Project Requirements above. The intent is that the created solution does not necessarily have to contain or demonstrate all of the elements detailed in the Course Outline section. However, at demonstration time, the member must definitively demonstrate detailed knowledge of how to implement each skill.

# Reviewer’s Responsibilities

The reviewer is responsible for certifying the member as having completed this training. The reviewer must perform the following steps in the process:

* Review completed assignment with member
* Assess knowledge of all core skills identified in the Course Outline section
* Provide notification of course completion to the Member Relations Coordinator

# Schedule Time Line

The following timeline is recommended for the core competency process:

|  |  |
| --- | --- |
| Task | Timeline |
| Complete review of all items within the Course Outline section | 2 Weeks |
| Complete development of assignment | 2 Weeks |