**Moneta Web-React Primer**

Moneta Web establishes a set of patterns and build techniques to help teams get started quickly on new applications using React/Angular, and Moneta Bootstrap. This primer focuses on React and Webpack.

**What you’ll build**

In this starter, you will build a new application using the Moneta Web technology stack and understand the code structure and build workflow. You will also learn about Kickstart, which will be used to generate a base project.

**What you’ll need**

* About 60 minutes
* Favorite Java IDE
* Dev Shell
* Gaia Dev account

If you don’t already have Dev Shell installed go to the Getting Started page and follow the installations. Execute the following in Dev Shell to install the required tools:

ds install git+nodejs10+maven3.5+jdk8+cf

**Project Structure**

If you examine the project structure, you will find 4 Maven projects embedded in a root Maven project:

* **Root project** (pom.xml): Maven Aggregator project that is aggregating the 4 child modules so they can be built as a group.
* **REST project** (react-demo-rest/pom.xml): Spring Boot application using Moneta Boot that will host the REST services backing the UI. This project produces a JAR file that is included in the final Spring Boot executable JAR file.
* **Web project** (react-demo-web/pom.xml): Hosts the HTML, JavaScript, styles, and front-end dependencies. This project is built with Webpack and integrates a complete front-end build workflow. The src folder contains the JSX files that define various React components. This project produces a JAR file containing static resources resources that is included in the Sprint Boot executable JAR file.
* **Parent project** (react-demo-parent/pom.xml): Parent POM project for REST, Web, and Application modules that manages the dependency versions and common plugins.
* **Application project** (react-demo-app/pom.xml): Maven JAR project that generates the final deployable JAR file. This project depends on the REST and Web projects.

**Technology stack overview:**

The following technologies are used as a base for building web applications using Moneta Web.

**Web**

* React – An open-source JavaScript library created by Facebook to build user interfaces. React uses JSX (an extension of JavaScript), which allows component-based JavaScript and HTML markup to be managed as a single unit.
* Moneta Bootstrap – A customized JPMC theme for Bootstrap, which is the most popular CSS framework for developing responsive web applications.

**Server**

* Spring Boot – A framework designed to simplify the bootstrapping and development of Spring applications.
* Moneta Boot – A JPMC framework that provides extensions for Spring Boot to allow for easier adoption at JPMC.

**Build**

* Maven – A project management tool used to fetch Java dependencies and build the overall project into an artifact that is published to Firmwide Repository Service (FRS). If you are not familiar with Maven, refer to Building Java Projects with Maven.
* NPM – A popular package manager for NodeJS applications. Most of the modern front-end tooling is built using NodeJS and distributed as NPM packages.
* Webpack – Webpack is a module bundler. Primarily for JavaScript, webpack can, if the corresponding loaders are included, take all of your application’s web assets and bundle them into static files. These static files are then served to the web browser in order to run your application.

**Testing**

* Jest – A JavaScript testing framework with a focus on simplicity. Jest can work out-of-the-box, config free, on some JavaScript projects.

**Web technology overview**

NPM