COURSE ABSTRACT

Merative SPM Design System for Developers

9D80G

Course Description

This course provides learners with a technical understanding of the Merative Cúram Social Program Management (SPM) Design System and how it can be used to develop responsive web applications.

The SPM Design System is built specifically to help create government web applications to the highest usability and accessibility standards. The Design System includes best practices, design principles, imagery, and brand style. Developers need a practical understanding of the SPM Design System so that they can develop high-quality web applications for citizens and other users. During the course, learners review the development environment, develop pages using Design System components, integrate with the SPM server, and test their pages. At the beginning of the course, learners will review the main third-party technologies that are used during the course. Approximately 70% of the time is spent completing exercises.

At the end of the course learners will implement a prototype solution for the user experience they sketched in the pre-requisite course: *Merative SPM Design System for Designers* (9D79G).

General Information

Delivery Method: Classroom (ILT) and self-paced with lab.

Audience: This course is intended primarily for developers and technical architects who will work on SPM implementation projects that use the SPM Design System. This course is also useful for testers and support staff who will work with the SPM Design System.

Topics: The course covers the following topics:

- Review of the tools and technologies for front-end development

- Developing pages using SPM Design System components
- Integrating using REST APIs
- Unit testing

Learning Objectives: After completing the course, learners should be able to:

- Describe the purpose of the web app development technologies and tools that are used on the course.
- Outline the technical foundation of the SPM Design System
- Install the Design System and configure the development tools.
- Implement pages using Design System components.
- Configure the theme and style of pages.
- Use a JSON server to simulate server interactions via a REST API
- Access a REST API on a n SPM Server
- Write simple unit tests for a component.
- Troubleshoot page errors using the IDE debugger and browser tools.
- List considerations for adding custom and third-party components.
- List useful resources for front-end development

Prerequisites: Learners must have completed the following course before taking this course:

- *Merative SPM Design System for Designers* (9D79G)

In addition, learners must have some experience of using the following technologies:

- HTML, CSS, JavaScript (ES6)
- JSX and React

Duration: 16 Hours (2 days ILT)

Skill Level: Intermediate

Version: This course was last updated for an earlier version of the SPM Design System. However, the course covers core topics that do not change much. An Addendum lists the recent changes.

Notes

The course duration gives learners an estimate of how much time they need to allocate to the course. The course duration does not specify the actual time required to complete the course, which varies by learner. The course agenda shows the schedule for a classroom (ILT) delivery. Learners taking this course in a self-paced environment should allow more time for exercises.

Course Agenda

Unit 1 - Tools and Technologies for Front-End Development

Lesson 1 – Course Introduction
Duration: 10 minutes

Learning objectives: After completing this lesson, learners should be able to:

- List the course objectives.
- Outline the course format and resources.
- List sources of help.

Lesson 2 – Review of Basic Technologies and Tools
Duration: 30 minutes

Learning objectives: After completing this lesson, learners should be able to:

- List the types of tools and technologies used for front-end development.
- Outline the purpose of the following tools and technologies:
  - CSS
  - Modern JavaScript
  - NPM
  - VS Code
  - Browser development tools
  - GitHub
- Set up a basic development environment.

Exercise 1 – Review the Development Environment
Duration: 35 minutes

Lesson 3 – Review of React and Associated Tools
Duration: 30 minutes

Learning objectives: After completing this lesson, learners should be able to:

- Outline the purpose and features of React.
- List tips for developing applications using React.
- Briefly describe the use of Redux for state management
- State the purpose of JSX, Babel, and Webpack
- Outline the browser tools for troubleshooting and debugging.

Exercise 2 – Review React
Duration: 30 minutes

Exercise 3 – Use Tools for Debugging
Duration: 20 minutes

Unit 2 - Developing Pages Using Design System Components

Lesson 1 – Setting up the Design System
Duration: 15 minutes

Learning objectives: After completing this lesson, learners should be able to:

- Describe the packages required to create standalone web apps.
- List the pre-requisites for using the SPM Design System
- Install the SPM Design System

Exercise 4 – Review SPM Design System Packages
Duration: 30 minutes

Lesson 2 – Using Design System Components
Duration: 20 minutes

Learning objectives: After completing this lesson, learners should be able to:

- List typical developer tasks
- Access developer documentation
- Create sample pages.

Exercise 5 – Create a Simple Page
Duration: 30 minutes

Exercise 6 – Create a Page with a Header and Footer
Duration: 50 minutes

Lesson 3 – Theming and Styling
Duration: 20 minutes

Learning objectives: After completing this lesson, learners should be able to:

- Describe how theming and styling are used.
- Change the colors using Sass properties.
- Describe how fonts are selected.
- Apply CSS styles.

Exercise 7 – Change the Theme and Style
Duration: 40 minutes

Unit 3 - Integration and Testing

Lesson 1 – Accessing REST APIs
Duration: 35 minutes

Learning objectives: After completing this lesson, learners should be able to:

- Provide an overview of REST APIs and their use in SPM.
- List the steps for implementing REST APIs on the server-side.
- Fetch data using a mocked API.
- View out-of-the-box REST APIs using Swagger.
- Fetch data from the SPM server

Exercise 8 – Access a REST API (Mocked Server)
Duration: 40 minutes

Exercise 9 – Access a REST API (SPM Server)
Duration: 35 minutes

Lesson 2 – Unit Testing
Duration: 20 minutes

Learning objectives: After completing this lesson, learners should be able to:

- Describe the purpose of Jest for testing React apps.
- Define simple Shallow tests.
- Define simple Snapshot tests.
- List useful test resources

Exercise 10 – Write Simple Tests
Duration: 40 minutes

Lesson 3 – Additional Development Considerations
Duration: 10 minutes

Learning objectives: After completing this lesson, learners should be able to:

- List considerations when developing custom components and using third-party components.
- Add an external component from a page.
- Describe the process for taking on new releases of the Design System

Exercise 11 – Add an External Component
Duration: 25 minutes

Lesson 3 – Course Conclusion
Duration: 5 minutes

Learning objectives: After completing this lesson, learners should be able to:

- Restate the course objectives.
- List further training
- Implement prototype pages from wireframes.

Exercise 12 – Case Study
Duration: 180 minutes

Addendum

Learning objectives: The addendum provides an overview of the significant updates to the SPM Design System since the course was initially released. This includes the following topics:

- New packages
- New components, templates, and patterns
- Additional changes