Micro Frontend - 5 days

By Dr. Vishwanath Rao

Prerequisites

- An understanding of modern frameworks for frontend applications (Angular, Vue.js, React + Redux, or similar frameworks)
- Solid knowledge of automation (CI, CD, testing strategies, etc.)
- Experience working with cross-functional or functional teams

Objectives

- Structure your apps to scale
- Understand how to divide a monolithic app into multiple sub-apps
- Coordinate data exchanged between your microfrontends
- Apply a production-style workflow
- Deploy your microfrontends
- Isolate rules styling by applying CSS-scoping techniques
- Judge whether microfrontends are an appropriate choice for your application

Lab Setup

- Windows / Linux /MAC with minimum 8 gb RAM
- Visual Studio code
- Node JS latest
- Chrome or Edge browser

Day 1

- Overview of micro-frontends and the problems they solve Available options Example implementations Sync and async services
- How to use iframes to split frontend apps
- Incremental upgrades Simple, decoupled codebases
- ES6 Javascript for Microfrontends Universal Rendering Client Side Integration
- Micro services Architecture Modern Web Application Development

Day 2

- Monolithic Frontends Teams with Micro Frontends
- Documents-to-Applications Continuum Progressive Web Apps
- Advanced DOM for MicroFrontends Custom Elements Team Product and its development
- Page Composition

Day 3

- Page transition Prototypes Parent Child Communication DOM Modification DOM Events
- Client Parent Communication Sibling Communication Server Side Rendering or Universal Rendering Custom Elements with server side includes Data Fetching Loading the states Navigating between Pages

Day 4

- Isolated CSS Coherent User Interface Style Guides & Pattern Libraries Performance on initial load Performance while using the site Loading CSS Loading JS
- Navigating between pages
- Soft vs. hard navigation
- Universal router Server-side template composition

Day 5

- Autonomous teams
- Build-time integration Run-time integration via Web Components

- Run-time integration via JavaScript Run-time integration via iframes
- Integration Testing