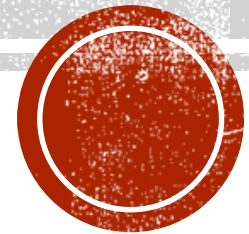


Module-1 Section 1.5

Web User Interface

Demo



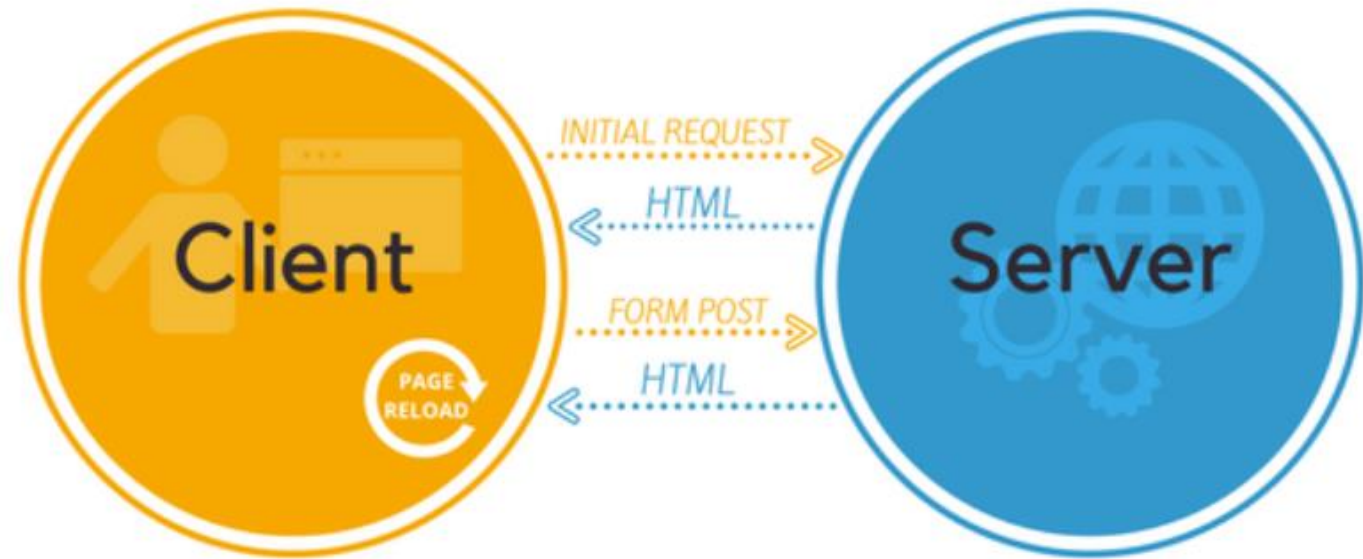
Contents

- Traditional Web Apps vs Modern Web Apps
 - Traditional Page Lifecycle
 - MPA (Multi-Page Application)
 - Java Spring
 - Python Django
 - ASP.NET MVC/ Razor
 - PHP Laravel
 - Modern approach: (Microservices)
 - SPA (Single Page Application)
 - Angular, React, Vue
- Web UI Development
 - Sketch (Interface into reality)
 - Workflow
 - Confirm sketch (structure and workflow)
 - Choose style
 - Follow best practice and Validate
 - Preview
 - Mock
 - Prototype



Traditional Web Apps

- Pros
 - Perfect approach where menu navigation is an essential part
 - Good Choice and easy for SEO management
 - Better chance to Rank
 - No JavaScript required
- Cons
 - Front end and Backend development are tightly coupled
 - No option to use same backend with mobile applications
 - Development becomes quite complex
 - Longer time of application development
 - Needs to user frameworks for client and server side



Forbes



SPA Web Apps

- Pros

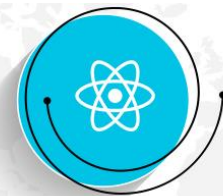
- Fast and smooth UX
- Headers, sidebars and footers remain the same to save time loading the page
- Local Caching and offline functionality
- Perfect Mobile Adaptability
- API reuse
- Streamlined development
- Decoupling front-end/back-end
- Easy debugging

- Cons

- SEO issues
- Memory leaks
- Security issues
- Navigation issues (no FWD/BACK button)



Angular.js



React.js



Backbone.js



Ember.js



Parameter	SPA	MPA
Speed	Initial load takes longer than in case of MPAs, but further usage of the app is seamless and faster	Loads slower and reloads each time when the user clicks on any button
Security	Prone to hackers' attacks, however, security can be improved with the right approach	Require a thorough approach to protect each separate page of the website
Development process	Reusable back-end code and completely separated back-end and front-end development	Numerous dependencies between front-end and back-end
JavaScript dependencies	Are present so depending on the type and version of a browser, problems may occur with an initial load	No JS dependencies
Link sharing	One single link to the entire app, however, deep linking allows linking to a specific piece of content	A separate link for every page of the MPA
User experience	Responsive, uninterrupted and mobile-friendly	Better information architecture
Examples	Gmail , Google Maps , Trello	Amazon , Forbes , CNN



PWA

- Progressive Web Apps
- Not created with a single technology
- <https://www.smashingmagazine.com/2016/08/a-beginners-guide-to-progressive-web-apps/>



DEMO & QUIZ

