JavaScript Frameworks

Introduction

In this section, you will explore popular JavaScript frameworks, what they do, and the advantages and disadvantages of the most popular ones. You will go over in detail: Ember.js, Vue.js, and Angular.

Duration: **15** mins

Objectives

After completing this reading, you will be able to:

1. Identify the types of JavaScript frameworks
2. List and describe popular JavaScript frameworks
3. Differentiate between different frameworks

JavaScript Frameworks - Overview

JavaScript frameworks provide developers with pre-written code that handles common programming patterns and makes it easier to create web applications within that context.

Both front-end and back-end JavaScript frameworks exist to help add interactivity to web applications. Each framework has its own unique purpose. When determining which framework to use, you should identify the key features you need in your applications and then find a framework that meets as many of these requirements as possible.

This reading will introduce you to some common frameworks used extensively at both enterprise and personal levels. There are many JavaScript frameworks available, so it is recommended to do some additional research about frameworks not mentioned in this reading.

Ember.js

Ember.js is an open source front-end JavaScript framework focused on large-scale web applications (rather than small ones). It is a highly opinionated framework, and while it does have some flaws, it also has significant advantages that make it the framework of choice for many applications today.

Ember.js follows the “convention over configuration” principle, also known as “Coding by convention”. This software design paradigm aims to recreate the number of decisions a developer must make while still maintaining flexibility and reducing repetition throughout the application.

Features

Ember.js has a variety of unique features that make it the framework of choice in many applications. This section will not provide an exhaustive list but will describe the main features and advantages of Ember.js.

- All URLs are assigned to **route objects**, which are used to manage the URL and control what is displayed to a user
- Every route is associated with a **model** which contains data about the current state of the web app

- Templates are used to automatically update the model if content in the application is changed. They help create the HTML code of your web app
- Services are objects which live for the duration of the application and are used for holding long-lived data. They can be made available in different parts of your application
- It contains many custom HTML tags, called components, which help in creating reusable and maintainable applications
- Ember.js has its own command line interface (Ember CLI) and debugging tool (Ember Inspector), which are ideal for debugging and integrating directly with the framework

Advantages and Disadvantages

Advantage	Disadvantage
Consistency due to strict rules and architecture	Steep learning curve
Detailed documentation	Small community, hard to find answers
Large selection of add-ons	Not suitable for smaller projects

Vue.js

Vue.js is a JavaScript framework used for building user interfaces by building on top of standard HTML, CSS, and JavaScript. Its component-based programming model helps develop both simple and complex user interfaces.

Features

Vue.js has two core features:

1. Declarative Rendering - Data can be rendered to the DOM using a straightforward template syntax. This enables developers to declaratively describe HTML output based on a JavaScript state.
2. Reactivity - Vue.js automatically tracks state changes in JavaScript and updates the DOM accordingly.

Other prominent features of Vue.js include:

- Vue.js has its own command line interface (Vue CLI) used to start, build, and compile projects
- Built-in directives can be used to perform actions on the front-end. A directive is a special token in the markup that can manipulate a DOM element when a specified condition occurs
- Watchers handle any data changes without requiring a developer to add additional events
- Navigation between pages is performed using the Vue Router

Advantages and Disadvantages

Advantage	Disadvantage
Fast and lightweight	Difficult to maintain consistency due to its overflexibility
Extensive documentation	Relatively fewer components and plugins
Reactive	Lack of support for large-scale projects

Angular

Angular is a component-based framework used for creating single page applications. It is most commonly used for client-side web applications and is embedded into HTML documents to enable user interactions.

Features

- Angular has its own command line interface (Angular CLI) used to build projects, add components, and deploy the application
- The component router loads apps quickly
- Template syntax makes it easy to create various user interface views
- Code generation enables metadata to be created to get code output for components that's optimized for JavaScript virtual machines
- API allows complex animations to be integrated with very little code

Advantages and Disadvantages

Advantage	Disadvantage
Secure applications	Steep learning curve
Simple testing	Performance tradeoff
Reusability of components	Heavy framework

Congratulations! You have successfully completed this reading

Author(s)

Michelle Saltoun

Changelog

Date	Version	Changed by	Change Description
2022-10-21	1.0	Michelle Saltoun	Initial version created