Published by :
http://www.ijert.org

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 12 Issue 04, April-2023

Salesforce Experience Cloud Portals (Community Cloud) Templates Options

Aura vs LWR

Sagar Sopan Wani
Sr Technical Architect, Accenture LLP

Ganesh Nemade
Technical Architect, SAP
.

***Abstract*—The objective of this paper is to provide a comprehensive study of Salesforce Experience Cloud Portals Templates. The paper aims to shed light on the which templates to use for portals development. In this paper, we will define LWR (Lighting Web Runtime), its characteristics, and the deployment model of LWR.**

***Keywords*—Cloud Computing, Architect, Salesforce CRM, Communities portals.**

I. INTRODUCTION

Learn how to build sub-second performant sites on our new Lightning Web Runtime (LWR). As we look to the future to deliver consumer-grade experiences, whether it's a commerce storefront, a marketing journey, a microsite, or even these industry bolt solutions, Lightning Web Runtime will provide even more performance and scalable experiences. Learn when to use Aura vs LWR which leads to uncover business opportunities.

Salesforce LWR stands for Lightweight Runtime (LWR) for Salesforce. LWR is a framework that allows developers to build custom web applications that can be hosted on the Salesforce platform. LWR provides a set of tools and APIs that enable developers to build fast and responsive web applications with a modern JavaScript framework of their choice, such as React, Vue, or Angular. LWR applications can be developed and tested locally, then deployed to the Salesforce platform for production use.

One of the key benefits of LWR is its ability to deliver a fast, responsive user experience. LWR applications are designed to run on the client-side, which means that they can be cached and loaded quickly, even on slow connections. Additionally, LWR applications are built to be modular and scalable, which makes them easy to maintain and extend over time.

LWR is part of the broader Salesforce Platform, which provides a range of tools and services for building custom applications, integrating with external systems, and managing data and workflows. LWR is particularly useful for developers who want to build custom web applications that leverage the power and scalability of the Salesforce platform.

II. PORTALS AURA SITES OR LWR SITES

When deciding between Aura and LWR for building custom web applications on the Salesforce platform, there are a few key factors to consider:

A. Performance: If your application requires high performance, fast page load times, and a responsive user interface, LWR may be the better choice. LWR applications are designed to run on the client-side and are optimized for speed and responsiveness. Aura applications, on the other hand, rely on server-side rendering and can be slower to load.

B. Complexity: If your application is complex and requires a lot of server-side processing, Aura may be the better choice. Aura has a powerful server-side controller framework that allows developers to build complex applications with ease. LWR, on the other hand, is designed to be lightweight and is best suited for simpler applications.

C. Customizability: If you require a high level of customizability, LWR may be the better choice. LWR allows developers to use modern JavaScript frameworks like React, Vue, or Angular, which provide a lot of flexibility and customization options. Aura, on the other hand, has its own proprietary framework that can be more restrictive.

D. Compatibility: If your application needs to be compatible with older browsers or devices, Aura may be the better choice. Aura is supported by all major browsers and is designed to work well on a wide range of devices. LWR, on the other hand, is designed to leverage modern browser features and may not work as well on older devices.

When deciding between Aura and LWR, it's important to consider factors like performance, complexity, customizability, and compatibility to determine which framework will be the best fit for your specific needs.

Accelerate Channel Sales
Deliver Online Self-Service
Integrate eCommerce
Drive Brand Engagement
Measure Partner Performance
Boost Employee Productivity

III SET UP THE LWR TEMPLATE FOR BUILDING LIGHTNING WEB RUNTIME (LWR) APPLICATIONS

1. Install the Salesforce CLI: The Salesforce CLI is a command-line interface tool that allows you to interact with the Salesforce platform from your local machine. Install the CLI by following the instructions on the Salesforce CLI installation page (Visual Studio code provides plugin for this)

2. Create a new LWR project: Once you have installed the Salesforce CLI, open a terminal or command prompt and navigate to the directory where you want to create your new LWR project. Then, run the following command to create a new LWR project using the LWR template command:

```
sfdx force:project:create --projectname MyLWRProject --template lwr
```

This will create a new project directory named "MyLWRProject" and populate it with the necessary files and configurations for an LWR application.

Create a new LWR app: Once you have created your LWR project, you can create a new LWR app within it by running the following command:

```
sfdx force:lightning:lwr:create --appname MyLWRApp
```

This will create a new LWR app directory within your project directory, named "MyLWRApp", and populate it with the necessary files and configurations for a basic LWR application.

Deploy your LWR app: Once you have created your LWR app, you can deploy it to your Salesforce org by running the following command:

```
sfdx force:source:deploy --sourcepath MyLWRApp
```

This will deploy your LWR app to your Salesforce org and make it available for use.

Test your LWR app: Once your LWR app is deployed, you can test it by navigating to the URL of the app in your Salesforce org. The URL will be in the format of:

https://[mydomain].lightning.force.com/lwr/MyLWRApp

Replace "[mydomain]" with the domain name of your Salesforce org, and "MyLWRApp" with the name of your LWR app.

When building Lightning Web Runtime (LWR) experiences on the Salesforce platform, you can use Lightning Web Components (LWC) to create reusable UI components that can be used across your application.

To use LWC components in your LWR application,

follow these steps:

Create a new LWC component: To create a new LWC component, use the Salesforce CLI to run the following command:

```
sfdx force:lightning:component:create --type lwc --componentname MyLWCComponent
```

This will create a new LWC component directory named "MyLWCComponent" within your project directory, and populate it with the necessary files and configurations for a basic LWC component. Add your LWC component to your LWR app: To add your LWC component to your LWR app, open the "main.js" file located in your LWR app directory and import your component at the top of the file, like this:

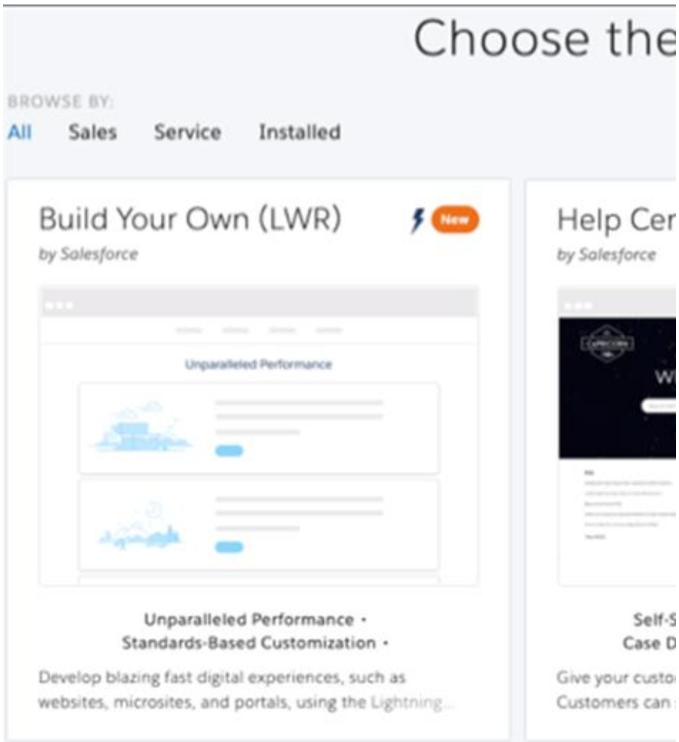
```
import MyLWCComponent from 'c/myLWCComponent';
```

Then, add your component to the "components" property of your app configuration object, like this:

```
const app = {

        components: {MyLWCComponent},

// Other app configurations

        };
```

This will make your LWC component available for use within your LWR app. Use your LWC component in your LWR app: To use your LWC component in your LWR app, add it to your app template file using its HTML tag, like this:

```
<my-lwc-component></my-lwc-component>
```



Unparalleled Performance • Standards-Based Customization •

Develop blazing fast digital experiences, such as websites, microsites, and portals, using the Lightning...

Self-S
Case D

Give your custo
Customers can

This will render your LWC component within your LWR app. By creating and using LWC components in your LWR application, you can create a modular and reusable UI that is easy to maintain and extend over time. Additionally, since LWC components are built on web standards like HTML, CSS, and JavaScript, they are easy to learn and can be used across a wide range of web development platforms and frameworks.

IV WHAT WILL BE THE FUTURE?

Salesforce is continually investing in Lightning Web Runtime (LWR) to enhance its capabilities and provide developers with new tools and features to build powerful and engaging web experiences on the Salesforce platform. Here are some of the features and updates that are coming to LWR in the near future.

Declarative App Builder: Salesforce is working on a declarative app builder for LWR that will enable developers to build LWR applications without having to write any code. This will allow for more efficient development and faster time-to-market for LWR applications.

Lightning Locker: Lightning Locker is a security feature that provides enhanced isolation between LWR components and the Salesforce platform. This feature is currently in beta and is expected to be released soon.

Performance improvements: Salesforce is always working to improve the performance of LWR applications, and is investing in new technologies like WebAssembly and Service Workers to help optimize the runtime performance of LWR.

Integration with other Salesforce products: Salesforce is also working to enhance the integration between LWR and other Salesforce products, such as Einstein and Tableau, to enable developers to build more powerful and sophisticated web experiences on the platform.

Enhanced developer tools: Finally, Salesforce is investing in new developer tools and resources to help developers build and deploy LWR applications more efficiently. This includes enhancements to the Salesforce CLI, improved documentation and tutorials, and more.

REFERENCES

[1] Alecia Teo on Experience cloud expert https://www.learnexperiencecloud.com/.

[2] https://trailhead.salesforce.com for Experience cloud sites.