Advanced Session Features

Advanced Displays, Integrations, PWA, LaunchLink, USB, ClipBoard, Keyboard Profiles, Mouse Modes, Multi-Language Config, Banners, Scripting

- Advanced Displays
- Advanced Integrations
- USB Human Interface Device Support
- Clipboard Integration
- Keyboard Profiles
- Mouse Modes
- Multi-language Configuration
- Banners and Classification Labeling
- Scripting
- USB Peripheral Support
- USB to Web Serial Redirection on Frame

Advanced Displays

Frame supports multiple monitors and 4k displays. This guide outlines the considerations, requirements, and steps to enable these features.

4k Display Support

Considerations

Administrators serving Frame sessions to end users with 4k displays can enable 4k support easily. Frame recommends considering the following before doing so:

- Customers using CPU-only instances with 1-2 vCPU(s) should test to ensure the instance is able to support 4k rendering in addition to the apps/services running on the VM.
- Administrators may need to increase the number of vCPUs for users accessing Frame with more than one 4k display.

Reference the table below to understand infrastructure requirements for supporting 4k instances:

Instance Configuration	4k Support
GPU-enabled	Fully supported
1-2 vCPU(s)	Not Recommended, Admin should test
3+ vCPUs	Supported, Admin should test

Enable 4k Displays

4k is enabled by default for GPU-enabled instances. If you wish to enable 4k for a CPU-only instance, please review the considerations mentioned above and then enable **4K Displays** in Session Settings.

Multiple Monitor Support

Multi-display configurations are useful for a variety of end user workflows. Frame's multi-monitor feature allows users to connect up to 4 displays at a time. End users may configure the display order as desired while in the session.

> Frame App supports the Frame multi-monitor feature automatically. If an end user has more than one monitor attached to their device, Frame App prompts the user if they want to use all of their monitors. Frame Administrators do not have to enable explicitly multi-monitor support for Frame App users.

Enhanced Multi-monitor Support

Users accessing Frame from Chrome or Chromium-based Edge browser can use the Enhanced Multi-monitor experience with automatic physical monitor layout detection.

> The Enhanced Multi-monitor feature is only available to users accessing Frame from Chrome or Chromium-based Edge version 100 or greater.

When using Auto Layout or Add Monitor in a browser session, the customer administrator or user must add `https://use.difr.com` as one of the **Allowed to send pop-ups and use redirects** sites, under **Settings** > **Privacy and security** > **Site Settings** > **Pop-ups and redirects** of their Chrome or Chromium-based Edge browser. This is discussed further in the Google Chrome Help article.

Refer to our Display Options documentation in our User Guide for details on how to use the Multi-monitor feature.

Advanced Integrations

Access to Frame apps and desktops can be integrated several different ways within:

- Websites
- Internal portals or services
- Operating systems
- and custom workflows

This section talks about integration tools that can be leveraged for custom integrations.

The Advanced Integrations panel can be found under a Launchpad's settings menu in the Dashboard of an account. This panel allows administrators to choose:

- Specific settings
- An application or desktop
- An identity provider
- and instance type

the administrator would like to use together. Combining these components results in copyable links that can be easily used, shared, or deployed just about anywhere that accepts hyperlinks.

image.png

image.png

PWA Links

image.png

What are PWAs?

PWAs (Progressive Web Apps) are web applications/websites that can be installed to local devices (Windows, Mac, Chrome OS, Linux, and more). Once installed, **these apps look, feel, and behave like native applications.**

PWA technology promotes a lot of best practices, offers a great experience, and is trending for a lot of other good reasons. Google has eliminated the traditional Chrome Apps in the Chrome Web Store in favor of PWAs, and Microsoft is working to integrate PWAs as first-class citizens in

the Windows Store. Combining PWA technology with Frame has some added benefits that makes life easier:

- Frame PWAs are easy to use, install, manage, and customize.
- They are cross-platform. Install them on Windows, Mac, Linux, and Chrome OS from the same PWA link.
- Portability: simply disconnect from your Frame session and seamlessly continue work on a different device.

Chromebooks/ChromeOS

Chromebooks are a great example: PWA technology with Frame allows Chromebook users to have an app like Adobe Photoshop easily accessible via the familiar Photoshop icon in their app drawer, or pinned to their shelf. It's easy to integrate these PWAs with your existing user authentication provider. For example, if you are using Google Workspace for authentication, installed PWAs can be synced between devices, making them as portable as ever. A user could have an app running on their Chromebook, disconnect their session, move to their laptop or desktop, and effortlessly launch the app from a native icon to resume their session.

Photoshop Launching Photoshop Launching on a Chromebook

Apps are easy to find PWAs are easy to find!

Photoshop enjoying some shelf life Photoshop has a longer shelf life!

Are PWAs right for my use case?

Frame PWAs are great for scenarios where users need long-term access to specific apps/desktops since they remove extra steps and clicks. However, there are some situations where Frame apps as a PWA may not be ideal. For example, short-term use cases like trials, training, and events might not make sense to have end-users installing your app if they are not going to use it again a short while later. For those situations, you may want to reach for Launch links instead.

Configuring PWA Links

Step 1. Choose an identify provider.



Step 2. Choose a pool/instance type.



> A single onboarded application or desktop can be used with multiple instance types. However, each instance type will have its own PWA link.

Step 3. Copy the PWA link for your app or desktop by clicking the **Copy PWA link** button.

That's it! You could test the links in a new tab to check for an install prompt and/or share it with a colleague or end user. Now that you have a PWA link, you've a few options for installation!

Manual PWA installation

To manually install a PWA (ideal for testing or smaller team sizes), open a PWA link on a device you would like to install it on. Once the page is loaded, look at Chrome's "omnibar" for a circular plus icon. Click it, then click **install**.

> Manual installation of PWAs is optional. Users can simply use the PWA link to launch the a session from within their browser (the option to install is still available).

Automatically install PWA Links with Managed Chrome

PWA links can be automatically deployed via Chrome to devices managed by administrative policies. There are two ways to manage Chrome:

- Cloud-managed (Google Workspace)
- Policies with on-premises tools

> **Be Careful**
> To ensure a great experience for your users, be sure your auth (identity provider and roles/permissions) and infrastructure capacity are properly configured before a large scale deployment of apps to your users.

Deployment with Google Workspace (formerly G Suite) deployment-with-google

1. Workspace Admins need to log in to their Admin console and navigate to **Device Management** > **Chrome** > **Apps & Extensions**.

2. Click on the **Add** button at the bottom right of the page, and select **Add by URL** from the context menu. Paste your pre-configured Frame PWA link in the URL field and select "**Open website in Separate Window**, then click **save**.
3. Your PWA application should appear on signed-in Chrome devices within a few moments.

.

For more information, please read Google's documentation for Adding Apps by URL.

Deployment with On-premises policies

Deploying PWA links with policies (Group Policies or otherwise) is pretty straight-forward. Please read Google's documentation on how to Automatically install web apps.

Uninstalling PWAs

PWAs are easy to uninstall. Users can simply uninstall them from the options menu at the top-right corner of the application. If you'd like to uninstall applications that were installed via Admin policies, simply remove the policy and give a small amount of time for the policy change to be reflected on connected devices – apps should be removed promptly.

Troubleshooting PWAs

When my users visit a PWA link, there's no option to install

This can happen for a few different reasons.

Troubleshooting tips:

- Make sure your browser is fully up-to-date.
- Make sure your Browser and OS support PWAs. At the the time of writing, Firefox doesn't support Desktop PWAs but does Mozilla supports them on mobile.

- Make sure your **application's icon is at least 144x144px**, though 512x512 is recommended. Modern apps are typically fine but older apps might need a fresh coat of paint. Admins can upload custom app icons in the Dashboard.
- Try refreshing the page. Sometimes certain PWA assets take a while to download and the installation prompt isn't triggered.

Will my Frame PWA applications work when offline?

No. Frame requires an internet connection to work.

PWAs on mobile devices

While PWAs are cross-platform by nature, please be aware that some users may encounter different experiences with their apps due to smaller screen size and different input methods. Please be sure to test your apps on prospective devices/operating systems/browsers before recommending them to your users.

Launch Links

What are Launch links?

Launch links are an easy way for you to provide your users with a direct link to a specific app or desktop. When a user visits a Launch link, your chosen identity provider handles authentication. Once authenticated, loading these links will immediately start a Frame session for that user.

You can easily copy a Launch Link from the Advanced Integrations panel after configuring your desired identity provider and instance type.



Add a link to your website

Launch links are easy to tie to any website using standard HTML. For example:

```
<a href="<paste your Launch link here>" target="_blank">
  A custom button
</a>
```

Okta Chiclets

Already using Okta with Frame? Great! Launch links can be added as Okta Chiclets with the help of Okta's **Bookmark App**. To get started, follow the steps below.

1. As an Okta Admin, let's add a new application.

2. Search for **Bookmark App** and select it.

3. Click on **Add**. Begin by adding the name of your Application into the *Application label* field.

4. Paste your **Launch link** (copied from Frame's Advanced Integrations dialog) into the URL field (be sure you've selected the right Okta identity provider before copying the link). Apply application visibility settings and click **Done**.

5.