Kiosk

- Self-Service Kiosk Overview
- FileWave Kiosk for macOS and Windows overview (15.3+)
- FileWave Kiosk for iOS/iPadOS overview (15.3+)
- App Portal / Kiosk v2 transition (15.3+)
- FileWave App Portal for iOS (IPA Install)
- Automatic updating of iOS/iPadOS Kiosk (15.3+)
- FileWave Kiosk displays Company Information
- Kiosk Package Updates Category (macOS PKG)
- Kiosk should show only Kiosk associated Apps (15.3+)
- Setting the Primary Color, Name and Logo in Kiosk/App Portal (15.3+)
- App Portal on iOS/iPadOS devices that don't support the application (15.3+)
- Applications Preventing Reboot (macOS/Windows)
- Troubleshooting

    - Checking what version of iOS/iPadOS App Portal is being pushed out (15.3+)
    - Removing pre-15.3 Kiosk Customizations (macOS/Windows)
    - Kiosk with macOS in a VM: Enabling Metal support
    - Resolving SSL and Manifest Validation Errors with FileWave Kiosk Installation (15.3+)

Self-Service Kiosk Overview

FileWave supports two methods of distributing content. The first is direct interaction from the FileWave Admin(s) where applications and other content are associated with devices or Groups of devices as part of a centrally managed deployment scheme. The second method is by using the self-service Kiosk and allowing the end user to choose the items to be installed on their device. Because the FileWave processes run at root level, the end user does not need to be a local administrator in order to install applications and content through the Kiosk.

The Kiosk is activated on computers by installing the FileWave client and having at least one Fileset configured as a Kiosk item associated with that Client. The Kiosk is activated on a mobile device when that device enrolls with the FileWave MDM.

Filesets can be configured as Kiosk items in FileWave Admin and can be added to unique categories, such as a specific department or class, or just by application type. You can even create a Kiosk Fileset of an iOS application from the App Store. The user gets the link to the store and the application or book would be downloaded from Apple when they request it. Kiosk items can be managed using Apple's VPP Managed Distribution model so that assigned applications can be installed by a user; but returned to the FileWave Admin for re-use at a later date.

Mobile Kiosk versus Desktop Kiosk

- Mobile: FileWave Kiosk for iOS/iPadOS overview (15.3+)
- Desktop: FileWave Kiosk for macOS and Windows overview (15.3+)

Related Content

- Setting the Primary Color, Name and Logo in App Portal (15.3+)
- Applications Preventing Reboot (macOS/Windows)

# FileWave Kiosk for macOS and Windows overview (15.3+)

This article will explain the details of the refreshed Kiosk for macOS and Windows that is included in FileWave 15.3+.

Browse Payloads/Filesets

The "Browse Payloads/Filesets" screen is a central feature within the app that enables users to explore and discover a diverse array of payloads/filesets available for access or download. Much like the experience of browsing through the Android Play Store or the Apple App Store, this screen offers users an organized and visually appealing interface that showcases different payloads/filesets, each accompanied by relevant details and information.

Key components and elements commonly found on the "Browse Payloads/Filesets" screen include:

- Company Logo: The visual representation of the company's logo, reinforcing brand identity and providing a familiar visual reference.
- Device Info: A button leading to the "Device Information" page provides users with quick access to essential insights about specific devices, such as company details, enrollment information, verification status, and app version.
- Payload/Fileset icon: Each payload/fileset is represented by an icon that provides a visual cue of its content or purpose.
- Payload/Fileset Name: The name of the payload/fileset is displayed alongside the corresponding thumbnail. This gives users a quick understanding of the payload's/fileset's context.
- Install/Uninstall Button: This view includes an "Install"/"Uninstall" button that users can click to initiate the installation/uninstallation process of the selected payload/fileset, streamlining the user's interaction and providing a clear call to action.
- Reinstall Button: This button is visible in two cases (for non-VPP filesets):
    - When the user tried to install fileset, and installation failed for some reason:
    - When the fileset is already installed on the device.
- Button Behavior: The installation process is now accompanied by clear button states, reflecting the various stages:
    - Install: Initiates the installation process.
    - Requested: Indicates that an installation request is pending.
    - Uninstall: Allows users to uninstall the selected payload/fileset.
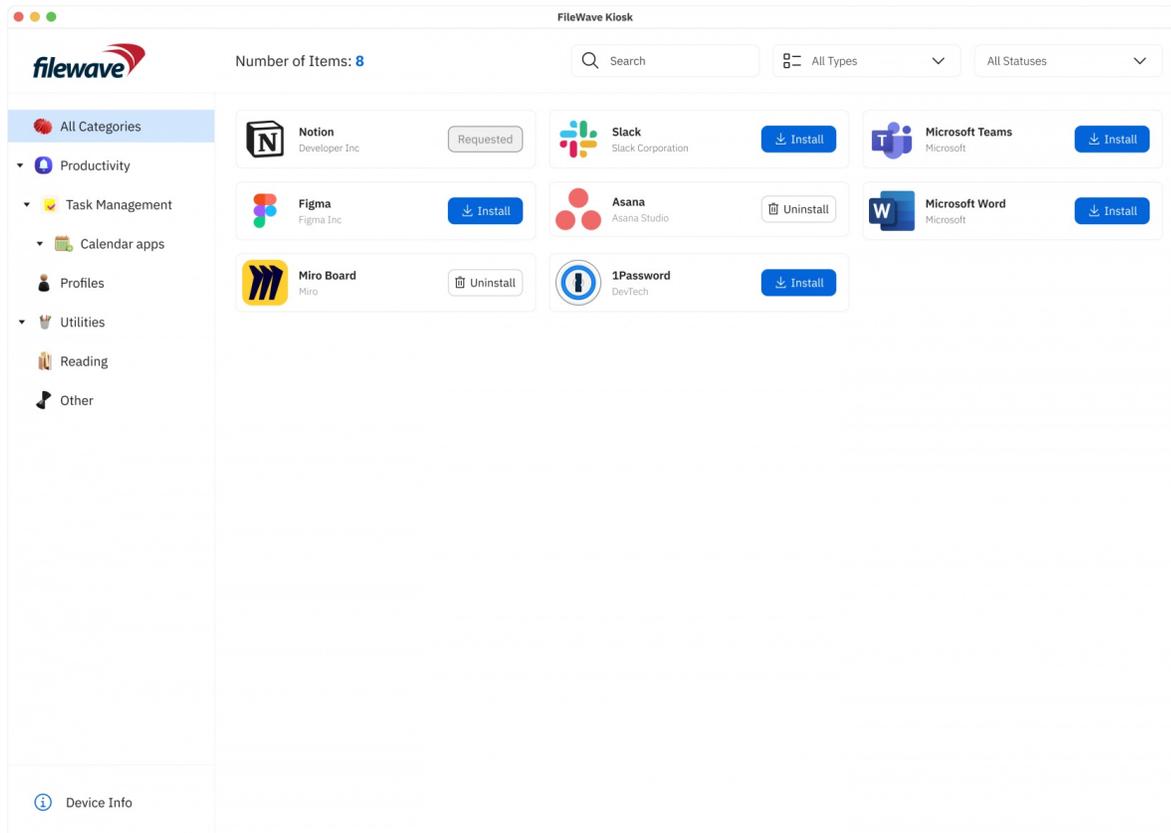
- Number of items: The number of items shows how many items are visible on the screen.
- Search Box: A search box is conveniently integrated, empowering users to quickly locate specific payloads/filesets based on keywords. This feature enhances precision and accelerates the search process.
- Scalable Left Sidebar: Responding to user preferences, the left sidebar is now scalable, allowing users to customize its width, expanding or minimizing it based on their preference. This flexibility enables users to optimize their workspace, with the left sidebar scalable up to 50% of the screen size.
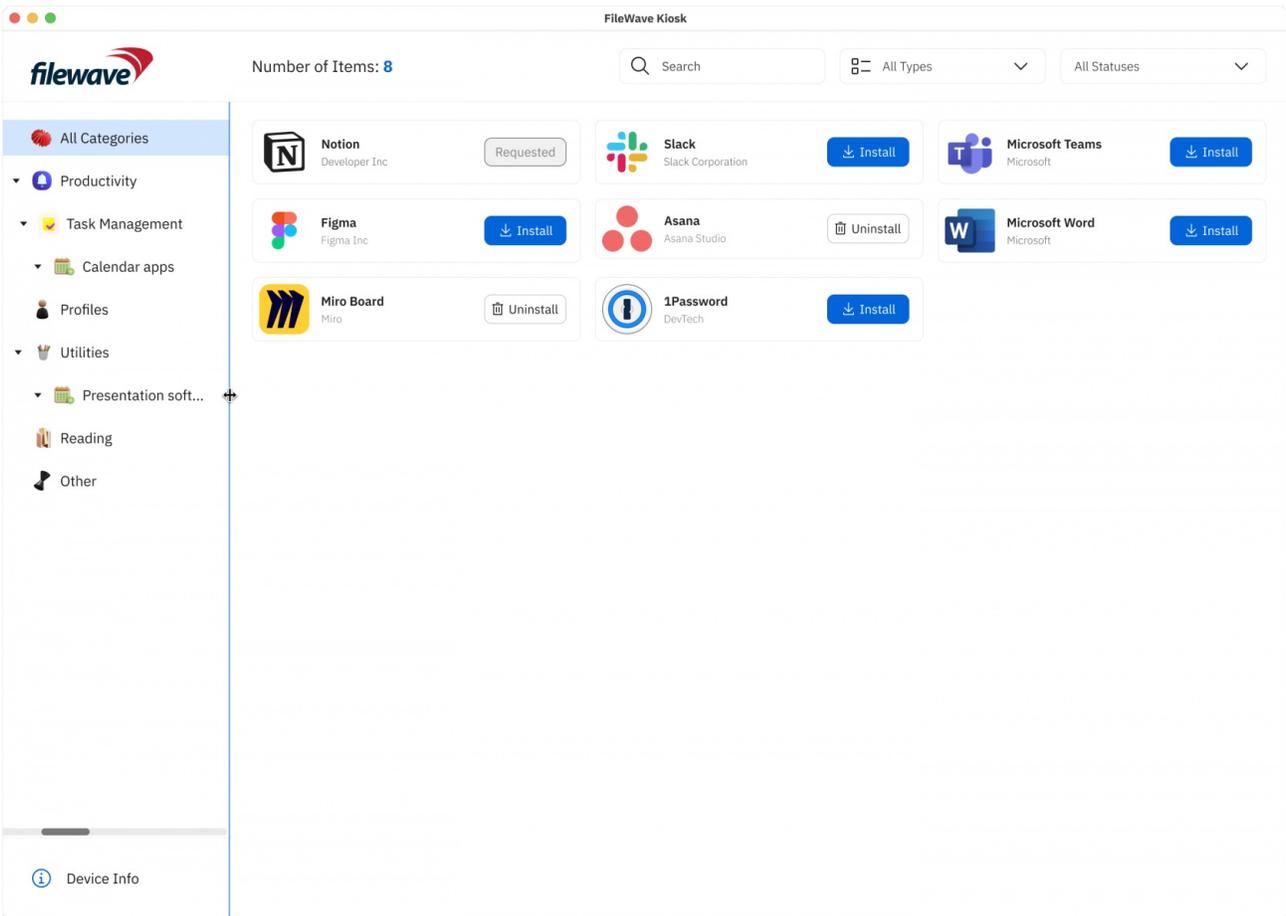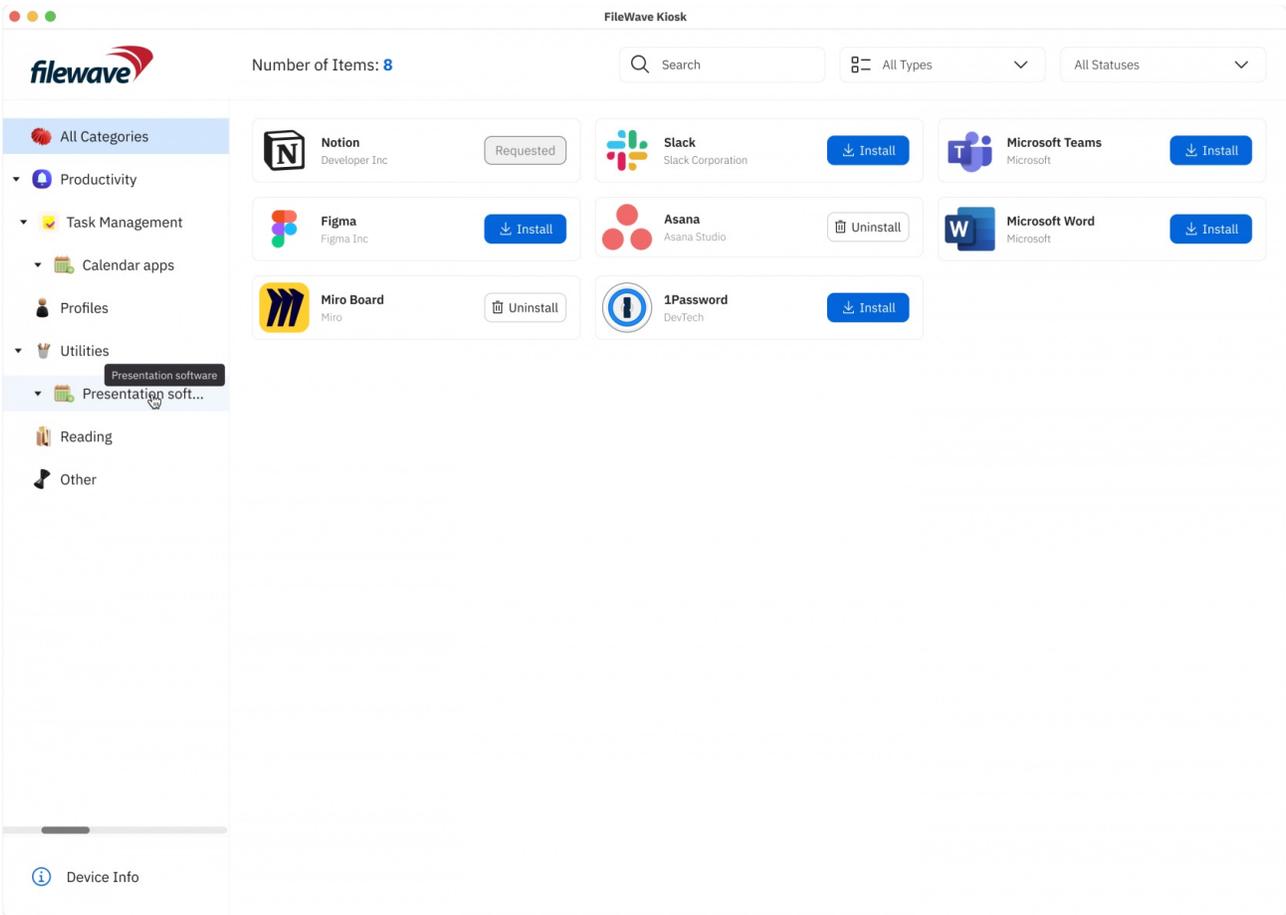
- Left Side Menu: Categories are prominently featured in the left side menu, allowing users to effortlessly navigate and filter payloads/filesets based on their specific interests. This user-friendly approach enhances efficiency and ease of use. Double click, or clicking on the icon on the left side of the category will expand the category and subcategories will be shown.

- **Statuses Filter:** Filter payloads/filesets based on their installation status (Installed, Not Installed).
- **Types Filter:** Tailor the view by filtering payloads/filesets according to types such as "Applications," "Books," "Documents," and more.

These enhancements not only simplify the user experience but also provide a comprehensive set of tools for effective payload/fileset management. By incorporating these features, we aim to empower users with a more personalized and efficient approach to navigating and managing their payloads/filesets within FileWave.

- **Payload/Fileset Cards:** Each payload/fileset is usually presented as a card containing its icon and name. Tapping on a card leads to a more detailed page for the payload/fileset.
- **Pagination:** If there are numerous payloads/filesets available, an infinite scrolling mechanism helps users navigate through multiple pages of listings.
- **Scrolling Animations:** Fluid scrolling animations and transitions contribute to a smooth browsing experience, enhancing users' engagement while exploring different payloads/filesets.
- **Loading Animation:** During data retrieval, a loading animation provides users with visual feedback, indicating that the app is actively fetching and populating the content.
- **Error Handling:** Robust error handling mechanisms ensure that clear and user-friendly error messages guide users in case of connectivity issues or technical glitches.
- **Flags for Enhanced Communication:** In the pursuit of transparent communication, we've introduced flags that offer clear insights into specific aspects of payloads/filesets. These flags include:
  - **Restart Needed:** A flag signaling that the installation or update of a payload/fileset requires a system restart for the changes to take effect. This ensures users are promptly informed about necessary actions for seamless functionality.
  - **Software Update:** This flag indicates that the payload/fileset represents a software update. Users will receive relevant messages, guiding them through the update process and ensuring they stay informed about crucial software enhancements.
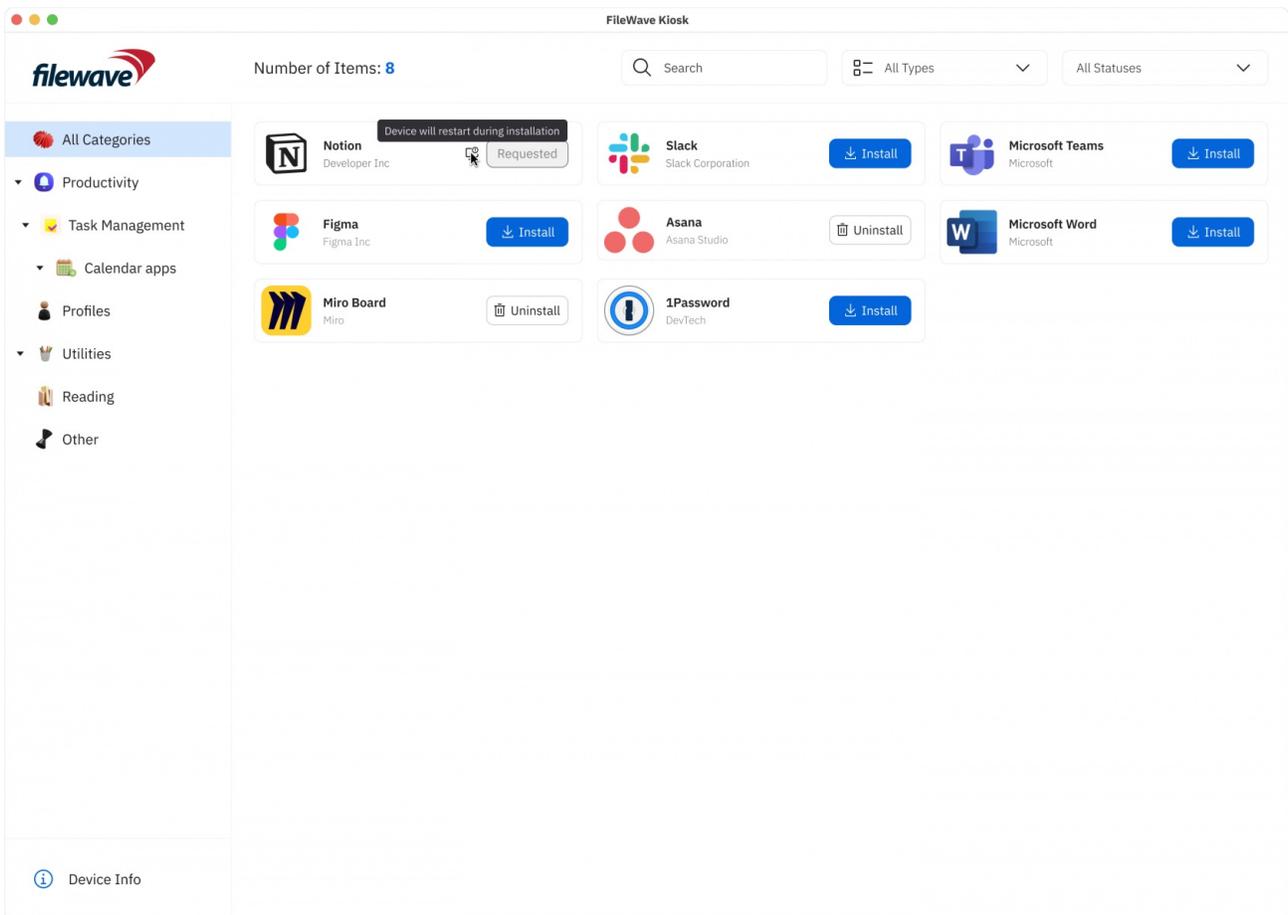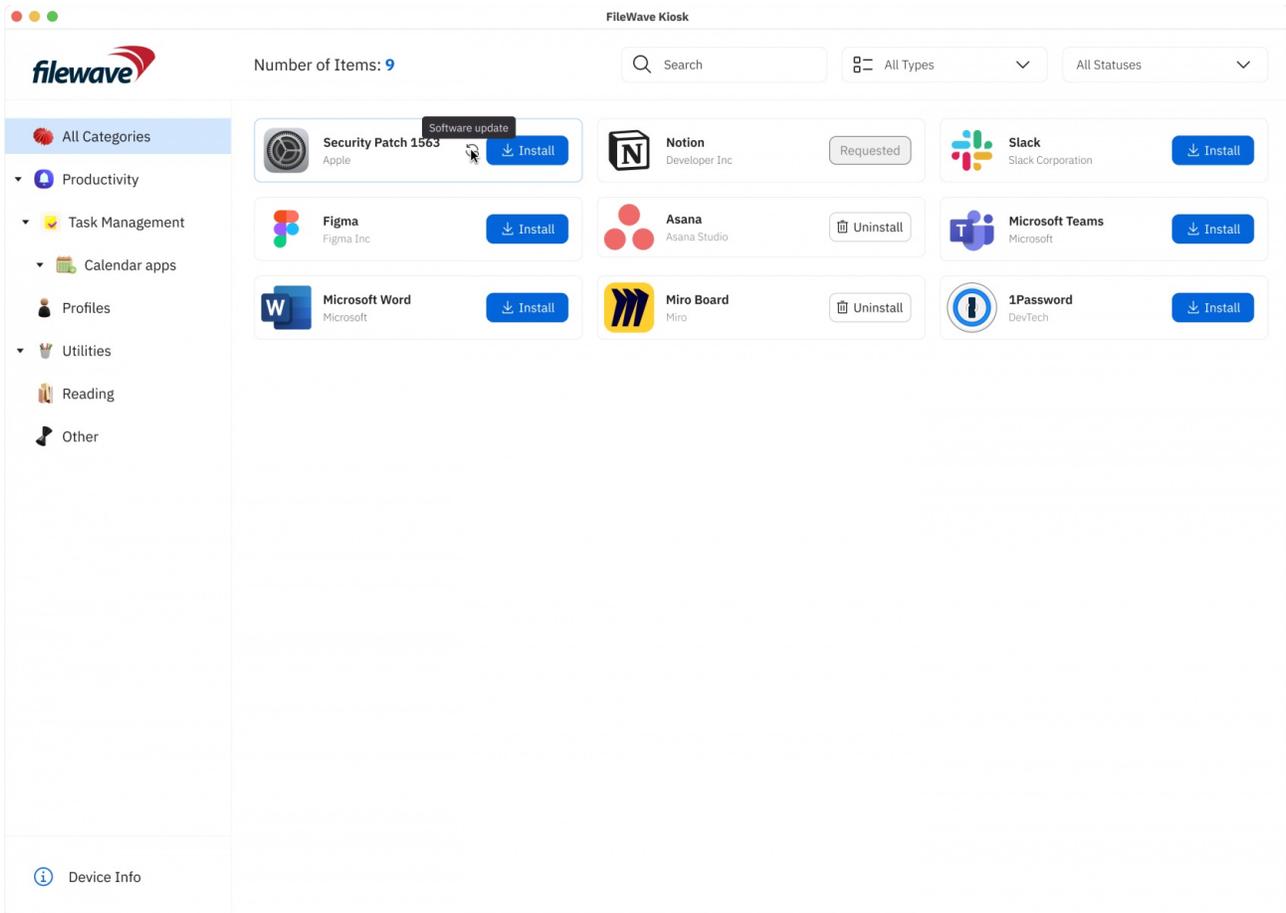
- VPP support (macOS):
  - JOIN Modal for Organization Selection:
    - macOS users accessing the JOIN section will encounter a modal displaying a comprehensive list of organizations available for joining. This intuitive interface simplifies organization selection and facilitates seamless integration with VPP.
  - Streamlined Registration Process:
    - Upon selecting the "Join" button, macOS users will seamlessly transition to the App Store app to complete the registration process. This streamlined workflow minimizes user effort and enhances efficiency in finalizing organization registration.
  - Dynamic Organization Visibility:
    - Upon successful registration with an organization, it will be automatically removed from the list of available organizations. This dynamic update ensures that macOS users are presented with accurate and relevant options, optimizing their selection process.
  - User-Friendly Interface:
    - If there are no more organizations available for joining, the list at the bottom right will intelligently hide, providing a cleaner interface for macOS users. This decluttered view enhances user experience and facilitates focused decision-making.
  - Compatibility with VPP Versions:
    - Registration functionality is fully compatible with both VPP v1 and VPP v2, ensuring seamless integration and operation across different VPP versions for macOS users.
  - Error handling (in case installation failed):
    - Icons with error text tooltips are shown in the Apps List page, providing users with quick access to error information.
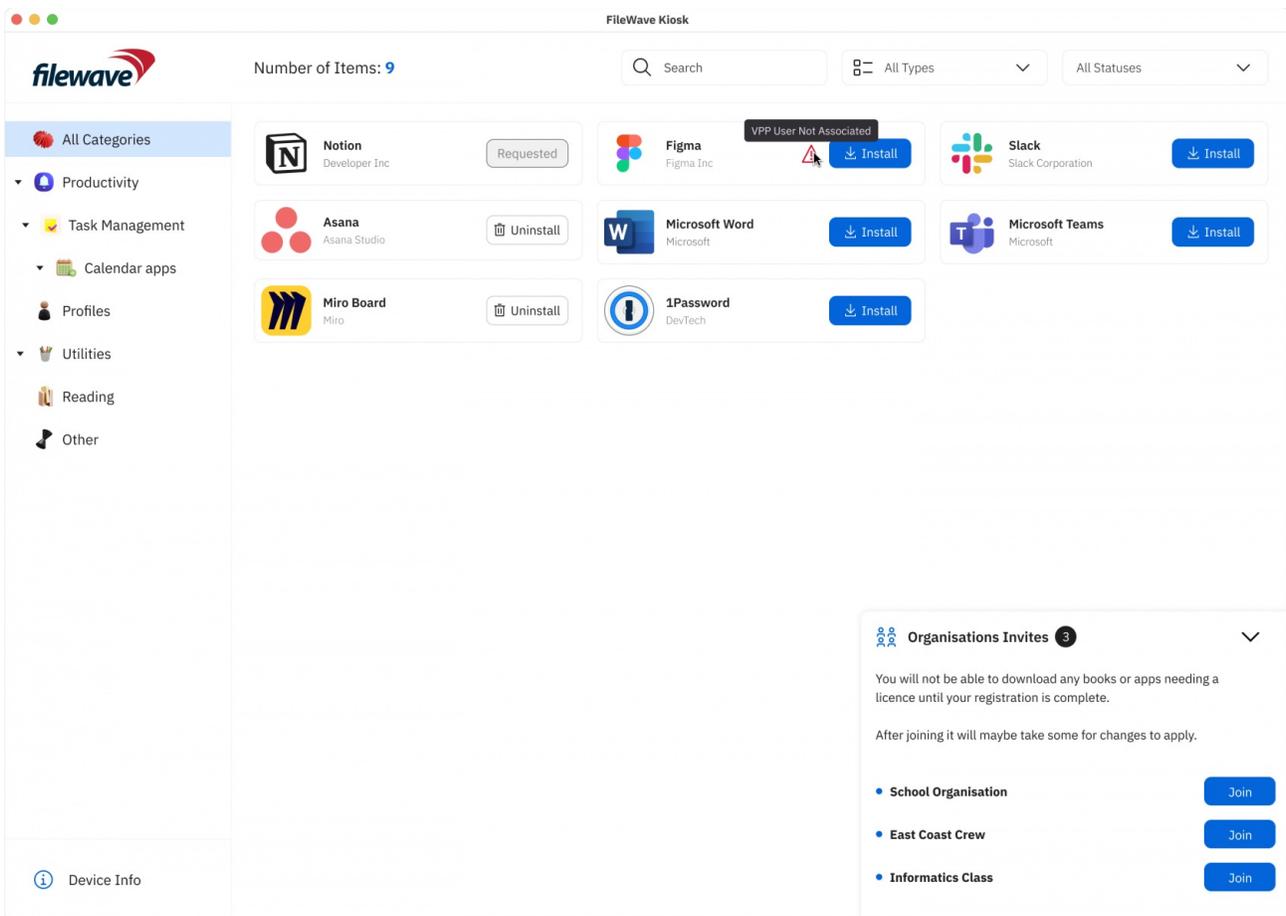
Payload/Fileset details view

We are excited to introduce significant enhancements to the Desktop Kiosk versions, mirroring the user-centric improvements made for iOS/iPadOS. With these updates, users can expect a more intuitive and informative experience when interacting with filesets and payloads on their desktop devices.

Key Features:

- Details View Redesign:
  - The "Details View" on Desktop Kiosk now mirrors the comprehensive and informative interface seen on iOS/iPadOS. Users encounter a screen that offers in-depth information about selected filesets/payloads, empowering them to make informed decisions.
- Key Elements and Components:
  - Fileset/Payload Icon: Visual identifiers help users quickly recognize the selected fileset/payload.
  - Fileset/Payload Name: Clear titles provide clarity about the content or purpose.
  - Creator Information: Details about the developer or source establish credibility and context.
  - Description: Comprehensive descriptions outline functionality, features, and benefits.
  - Information: Users can access details like the last update date, version, and size of the fileset.
  - User Interaction Buttons: Interactive buttons allow users to initiate actions such as installation, with progress updates visible until completion.
  - Return to Browse: A back button facilitates seamless navigation back to the previous browsing screen.
  - Scrolling Animations: Fluid animations enhance the browsing experience, providing engaging transitions between content.
  - Loading Animation: Visual feedback informs users of data retrieval processes, ensuring they are aware of ongoing operations.
  - Error Handling: Robust error messages guide users through connectivity issues or technical glitches, maintaining a smooth user experience.

< **Back**

🗑 Uninstall

Notion - The All in one Workspace
Developer Inc.
acquired

Status ✓ Installed

Description

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

 Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum. Vuptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident,

Information

Updated	Version	Size
12 Aug 2021	1.5.0	17.3 MB

These enhancements aim to streamline the desktop browsing experience, aligning it closely with the intuitive interface seen on iOS/iPadOS. By providing users with comprehensive information and intuitive navigation, the Desktop Kiosk versions empower users to make informed decisions and maximize productivity.

Device Information view

We're thrilled to unveil enhancements to the Device Information view on Desktop Kiosk, paralleling the user-centric improvements made for iOS/iPadOS. This revamped interface provides users with a comprehensive overview of vital device details, fostering transparency and effective management within the app's ecosystem.

Key Features:

- Header Navigation:
  - Users can easily navigate back to the Browse Apps view using the back button located in the header, ensuring seamless transitions between different sections.
  - The last verified date provides users with the latest verification status of the device, facilitating real-time monitoring.
  - A prominent "Verify Device" button in the header enables users to manually trigger a device verification process, ensuring up-to-date validation.
- Device Information Section:
  - The "Device Information" page now offers an exhaustive display of essential device details, empowering users with crucial insights. This section includes:
    - Name: Unique identifier for the device.
    - Address: Network address associated with the device.
    - Version: Operating system version installed on the device.
    - Platform: Platform type (e.g., Windows, macOS).
    - Model Version: Device model information.
    - Tickle Interval: Frequency of communication between the device and the server.
    - File Check Interval: Interval for checking file updates or changes.
- Server Information Section:
  - Additionally, users can access pertinent server-related information for seamless integration. This section comprises:
    - FileWave Server (IP Address): Server address for file management.
    - SSL: Indication of Secure Socket Layer (SSL) protocol status.
    - Status: Current server connection status.
    - Server Connection: Details about the server connectivity.
- Booster Section:
  - Users can now conveniently view and manage boosters directly from the Desktop Kiosk interface. This section presents a comprehensive list of boosters along with their corresponding IP addresses, streamlining booster management tasks.

These enhancements bolster transparency and facilitate efficient device management within the Desktop Kiosk environment. By

providing users with a holistic overview of device and server information, coupled with booster management capabilities, the Desktop Kiosk version empowers users to make informed decisions and optimize their workflows with ease.

FileWave Kiosk

< Back

Verify Device

Device Information

Name	**Victor's Workstation**
Address	**10.2.4.199**
Version	**12.2.0. (Rev. 4dcf548b)**
Platform	**macOS 10.12 Sierra (10.12.5)**
Model Version:	**4**
Tickle Interval:	**12 sec**
File Check Interval:	**1440 min**

Server Information

FileWave Server:	**127.0.0.1:20015**
SSL	**True**
Status:	**Running**
Server Connection:	**Connected (No Updates)**
Last Successfull Connection:	**10.05.17 01:23**

Boosters

1. fwbooster1.okcps.org(0-299) (10.3.112.80)
2. fwbooster1.okcps.org(0-299) (10.3.112.80)
3. fwbooster1.okcps.org(0-299) (10.3.112.80)
4. fwbooster1.okcps.org(0-299) (10.3.112.80)
5. fwbooster1.okcps.org(0-299) (10.3.112.80)

This application has the potential to report information to the enrollment server. These settings are controlled by your local administrator.

FileWave Privacy Policy

App Version: 15.3.0   Build Number: 100c1428

FileWave International Holding AG ©2023

Logo and primary color selection

The Adding Logo and Primary Color Selection feature enhances customization within the app by enabling users to personalize the visual representation of their profiles or entities. This feature allows users to define their brand identity and establish a cohesive look and feel throughout the app. Here's how the feature works:

- Logo Upload: Users can upload their company or entity logo, which serves as a distinctive visual identifier. The logo could be in various formats, such as JPEG, PNG,... Once uploaded, the logo is displayed in relevant sections of the app where the user's profile or entity is showcased.
- Primary Color Selection: Users can choose a primary color that resonates with their brand's identity. This color becomes the dominant hue used for interface elements such as buttons, headers, and accents throughout the app. Users can select the color from a color palette or enter a custom color code.
- Visual Consistency: The uploaded logo and selected primary color are applied consistently across various sections of the app. This ensures a coherent and visually pleasing experience for users and anyone interacting with their profiles or entities within the app.
- Personalization: By allowing users to upload their logos and select a primary color, the app offers a personalized touch, making users' presence and contributions stand out within the app's environment.
- Branding Impact: The combination of a unique logo and primary color strengthens brand recognition and recall, creating a strong visual association with users' profiles or entities.
- User-Friendly Interface: The process of adding a logo and selecting a primary color is designed to be intuitive, with clear instructions and user-friendly controls that guide users through the customization process.
- Flexibility: Users can update or modify their logo and primary color selection at any time to reflect changes in their branding or preferences.

The "Add Logo and Primary Color Selection" feature empowers users to infuse their identity into the app's interface, enhancing recognition, and contributing to a personalized and engaging experience for both users and their audiences.

Related Content

- App Portal / Kiosk v2 transition (15.3+)
- Setting the Primary Color and Logo in Kiosk/App Portal (15.3+)

FileWave Kiosk for iOS/iPadOS overview (15.3+)

What

The Self-Service Kiosk is how you can offer an easy way for your users to install approved software without the need for IT to specifically assign it to a iPad or iPhone. Instead, you can assign a collection of approved applications to your devices, and then your users can pick from that collection whatever apps they need. The FileWave App Portal is getting a major overhaul starting with FileWave Version 15.3. You'll automatically see this new Kiosk pushed out to your iPhones and iPads when you upgrade to any version of FileWave from 15.3 or higher as discussed in: Automatic updating of iOS/iPadOS Kiosk (15.3+)

When/Why

The transition from Technical Preview version 15.1 to the Production version 15.3 of our app signifies a major leap forward in terms of user experience and visual appeal, much like the transformation seen when comparing the Android Play Store and the App Store. In this upgrade, the primary focus has been on completely revamping the user interface (UI) to provide a sleek, modern, and highly intuitive platform that caters to our users' evolving expectations.

With version 15.3.1, the app undergoes a stunning visual makeover, drawing inspiration from the aesthetic excellence of both the Android Play Store and the App Store:

1. Unified Design Paradigm: Our new UI design unifies elements from both the Android Play Store and the App Store, creating a harmonious blend of familiarity and innovation. This design consistency ensures that users across various platforms feel comfortable while navigating the app.
2. Intuitive Navigation: Introduces a reimagined navigation system. Users can effortlessly explore and discover content thanks to intuitive menus, recognizable icons, and seamless navigation flows, akin to the ease of navigation in both app marketplaces.
3. Visual Delight: The new UI is not just about functionality; it's a visual treat. Engaging animations, subtle transitions, and tastefully curated visuals combine to make every interaction with the app a delightful experience, echoing the immersive nature of the Android Play Store and the App Store.
4. Enhanced Customization: Introduces enhanced customization options, allowing users to personalize their interface just as they would within the Android Play Store and the App Store. Users can arrange and prioritize content according to their preferences, granting them a sense of ownership over their app experience.
5. Responsive Design: Much like the Android Play Store and the App Store's responsiveness across devices, our app now adapts seamlessly to various screen sizes and orientations. This ensures a consistent and optimized experience whether users are browsing on smartphones, tablets, or desktops.

By transitioning to the 15.3 release of the Kiosk, with a UI overhaul reminiscent of the Android Play Store and the App Store, our app solidifies its commitment to delivering an unparalleled user experience. This transformation isn't just about aesthetics; it's a testament to our dedication to meeting the evolving needs of our users in an engaging, intuitive, and visually striking manner. As we bridge the gap between the old and the new, version 15.3 sets the stage for a future where our app continues to evolve in tandem with user expectations and industry trends.

How

Browse Payloads/Filesets

The "Browse Payloads/Filesets" screen is a central feature within the app that enables users to explore and discover a diverse array of Payloads/Filesets available for access or download. Much like the experience of browsing through the Android Play Store or the Apple App Store, this screen offers users an organized and visually appealing interface that showcases different Payloads/Filesets, each accompanied by relevant details and information.

Key components and elements commonly found on the "Browse Payloads/Filesets" screen include:

- Company Logo: The visual representation of the company's logo, reinforcing brand identity and providing a familiar visual reference.
- Device Information: A button leading to the "Device Information" page provides users with quick access to essential insights about specific devices, such as company details, enrollment information, verification status, and app version.
- Payload/Fileset icon: Each Payload/Fileset is represented by icon that provides a visual cue of its content or purpose.
- Payload/Fileset Name: The name of the Payload/Fileset is displayed alongside the corresponding thumbnail. This gives users a quick understanding of the payload's/Fileset's context.
- Install Button: This view includes an "Install" button that users can click to initiate the installation process of the selected Payload/Fileset, streamlining the user's interaction and providing a clear call to action. If the Payload/Filesets is already installed, button will not be visible.
- Category Filters: Users can typically filter Payloads/Filesets by Types such as "Applications," "Books," "Documents," and more. Filtering is possible by Categories (these should be configured in FileWave Central) and also by Status (Installed, Not Installed). This categorization helps users narrow down their search based on their specific interests.

iPhone	iPad
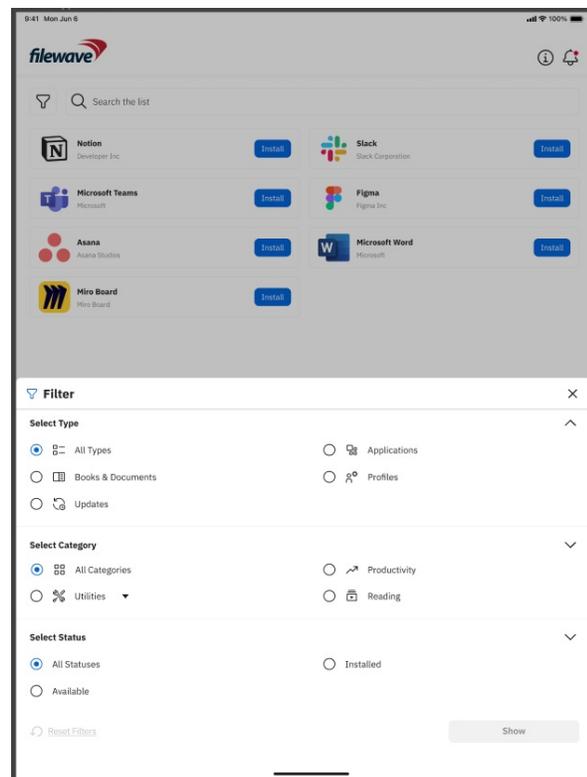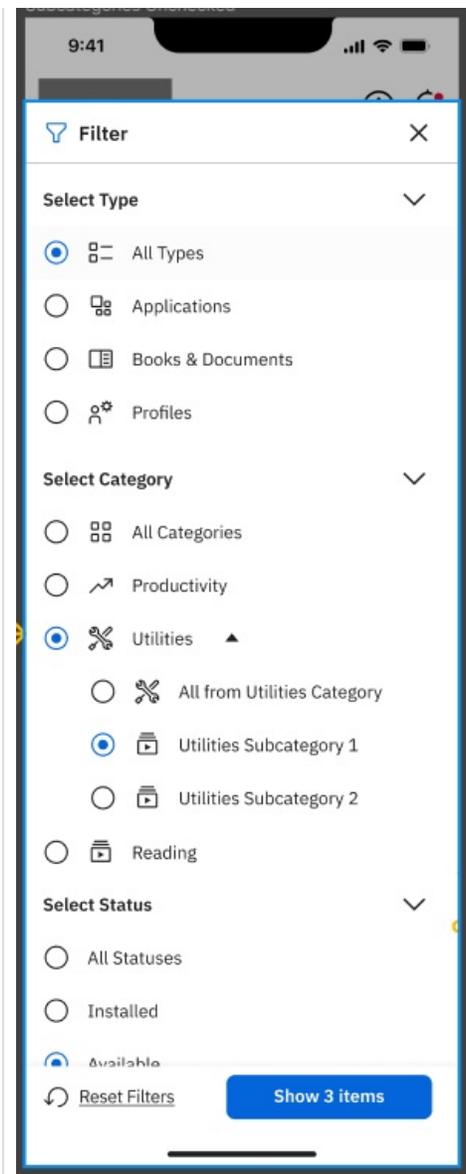
- Search Bar: A search bar allows users to directly search for specific Payloads/Filesets by their name.
- Payload/Fileset Cards: Each Payload/Fileset is usually presented as a card containing its icon and name. Tapping on a card leads to a more detailed page for the Payload/Fileset.
- Pagination: If there are numerous Payloads/Filesets available, an infinite scrolling mechanism helps users navigate through multiple pages of listings.
- Access/Download Button: A clear button to access or download the Payload/Fileset is generally included on each card, allowing users to initiate the relevant action seamlessly.
- Scrolling Animations: Fluid scrolling animations and transitions contribute to a smooth browsing experience, enhancing users' engagement while exploring different Payloads/Filesets.
- Loading Animation: During data retrieval, a loading animation provides users with visual feedback, indicating that the app is actively fetching and populating the content.
- Error Handling: Robust error handling mechanisms ensure clear and user-friendly error messages guide users in case of connectivity issues or technical glitches.
- Landscape and Portrait Views: The screen seamlessly adapts to both landscape and portrait orientations for iPad, while for iPhone only portrait orientation is available, ensuring a consistent and optimal browsing experience across different device orientations.

iPhone	iPad
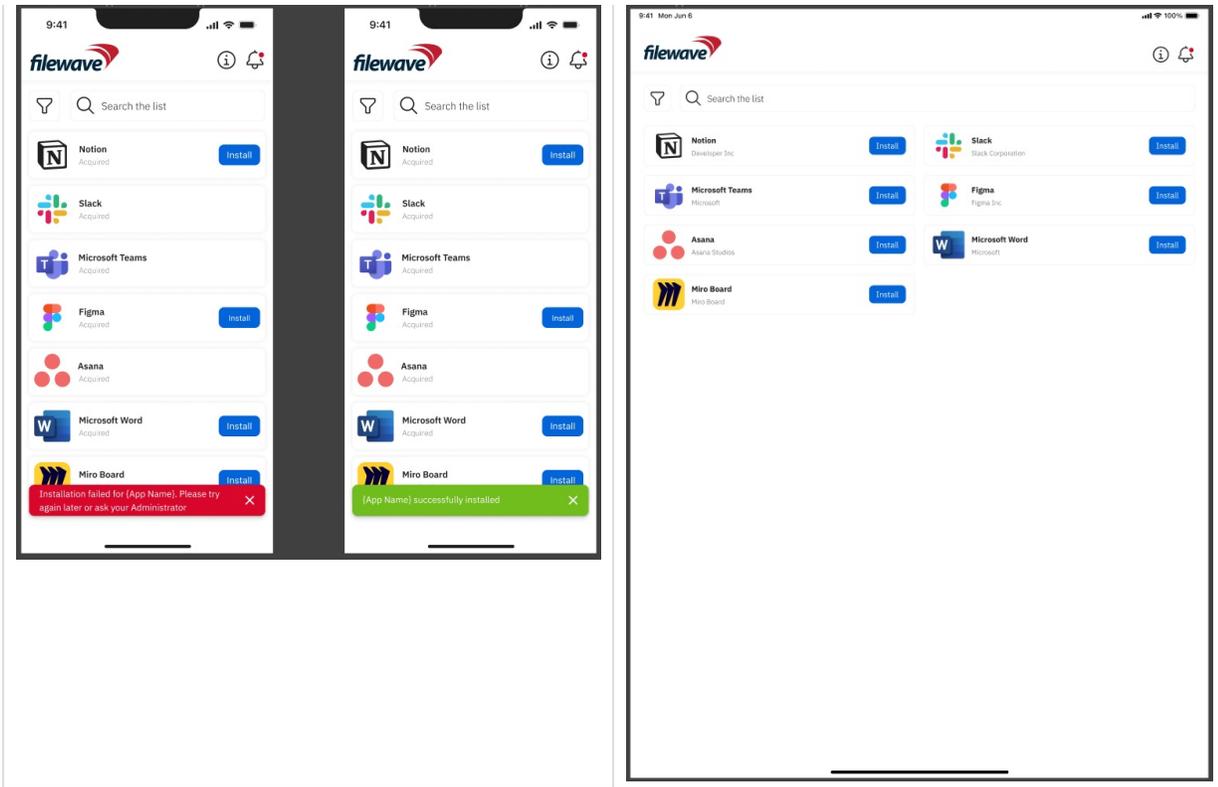
The "Browse Payloads/Filesets" screen serves as a gateway for users to immerse themselves in the app's content ecosystem, allowing them to discover and access various Payloads/Filesets that align with their needs and interests. Its design, layout, and functionality play a crucial role in delivering a user-friendly and enjoyable browsing experience, reminiscent of the fluid exploration on the Android Play Store and the Apple App Store.

Payload/Fileset details view

The "Details View" is the screen that users encounter after clicking on a specific Payload/Fileset they wish to install or access. This screen provides comprehensive information about the selected Fileset, offering a deeper understanding of its features, functionality, and relevance. Much like the detailed view of an app in the Android Play Store or the Apple App Store, this interface is designed to empower users to make informed decisions about their installation or usage.

Key elements and components often present on the "Details View" include:

- Fileset/Payload Icon: The icon representing the selected Payload/Fileset, serves as a visual identifier for users.
- Fileset/Payload Name: The name or title of the Fileset, providing clarity about the content or purpose.
- Creator Information: Details about the developer, creator, or source of the Fileset, establishing credibility and context.
- Description: A comprehensive description of the Fileset's functionality, features, and benefits, outlining what users can expect.
- Information: Provides information regarding when was Fileset last updated, and what is the version and size of the Fileset.
- Screenshots or Previews: Visual representations of the Fileset's interface, content, or use cases through screenshots or short videos.
- User Interaction Buttons: Buttons to perform actions - "Install" enabling users to initiate the desired action. Once the Install is triggered, the button will change state to show the progress of the installation. Once the installation is complete - the button will disappear, but a new field "Status" will be shown which will give final information about the status of the Fileset.
- Return to Browse: A back button to return to the previous "Browse Payloads/Filesets" screen for further exploration.
- Scrolling Animations: Fluid scrolling animations and transitions contribute to a smooth browsing experience, enhancing users' engagement while exploring different Payloads/Filesets.
- Loading Animation: During data retrieval, a loading animation provides users with visual feedback, indicating that the app is actively fetching and populating the content.
- Error Handling: Robust error handling mechanisms ensure that clear and user-friendly error messages guide users in case of connectivity issues or technical glitches.

| iPhone	iPad
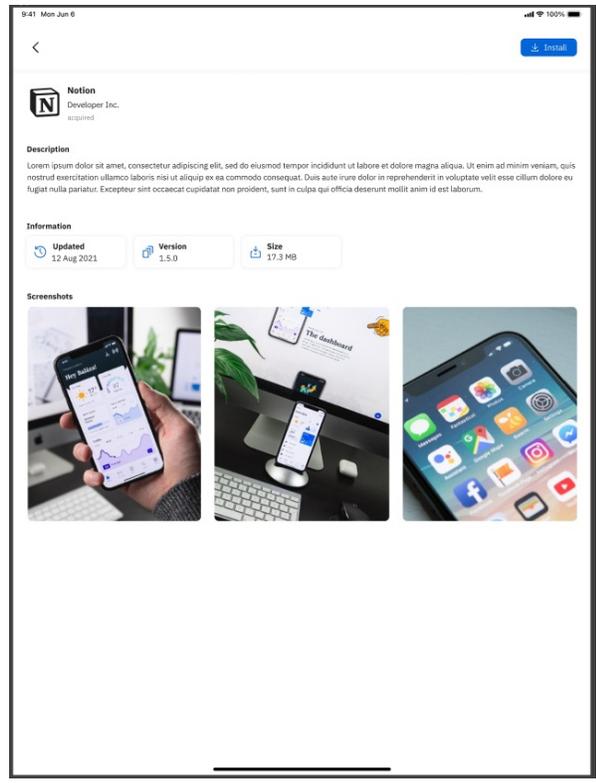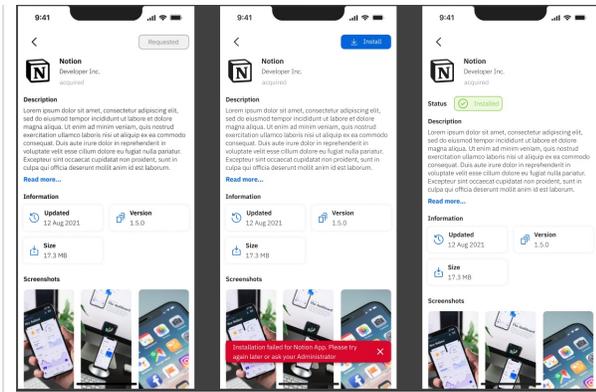
The "Details View" enriches the user's decision-making process by providing an in-depth look into the selected Fileset's attributes. By offering clear explanations, visuals, and relevant information, this interface ensures that users can confidently proceed with the installation or access of the chosen Fileset, enhancing their overall app experience.

Device Information view

The "Device Information" page provides a comprehensive overview of essential details about a specific device, enhancing transparency and management within the app's ecosystem. This page showcases crucial information related to the device and its integration, allowing users to make informed decisions and take necessary actions. The following information is displayed:

- Company Logo: The visual representation of the company's logo, reinforcing brand identity and providing a familiar visual reference.
- Company Name: The name of the company associated with the device, establishes a clear connection between the device and its ownership.
- Address: The address of the company, providing users with a means to locate or contact the company physically.
- Phone Number: The contact phone number of the company, offering users a direct channel for communication.
- Email: The email address of the company, providing an electronic means of communication and support.
- Enrollment Information: Details about where the device was enrolled, indicating the source or method through which the device became part of the app's ecosystem.
- Last Verification Date: The date on which the device was last verified or confirmed as operational, enabling users to monitor the device's status.
- Verify Device Button: A prominently displayed button that allows users to manually trigger a device verification process, ensuring real-time validation.
- App Version and Build Number: Information about the version and build number of the app installed on the device, indicating the current state of the application.

iPhone	iPad
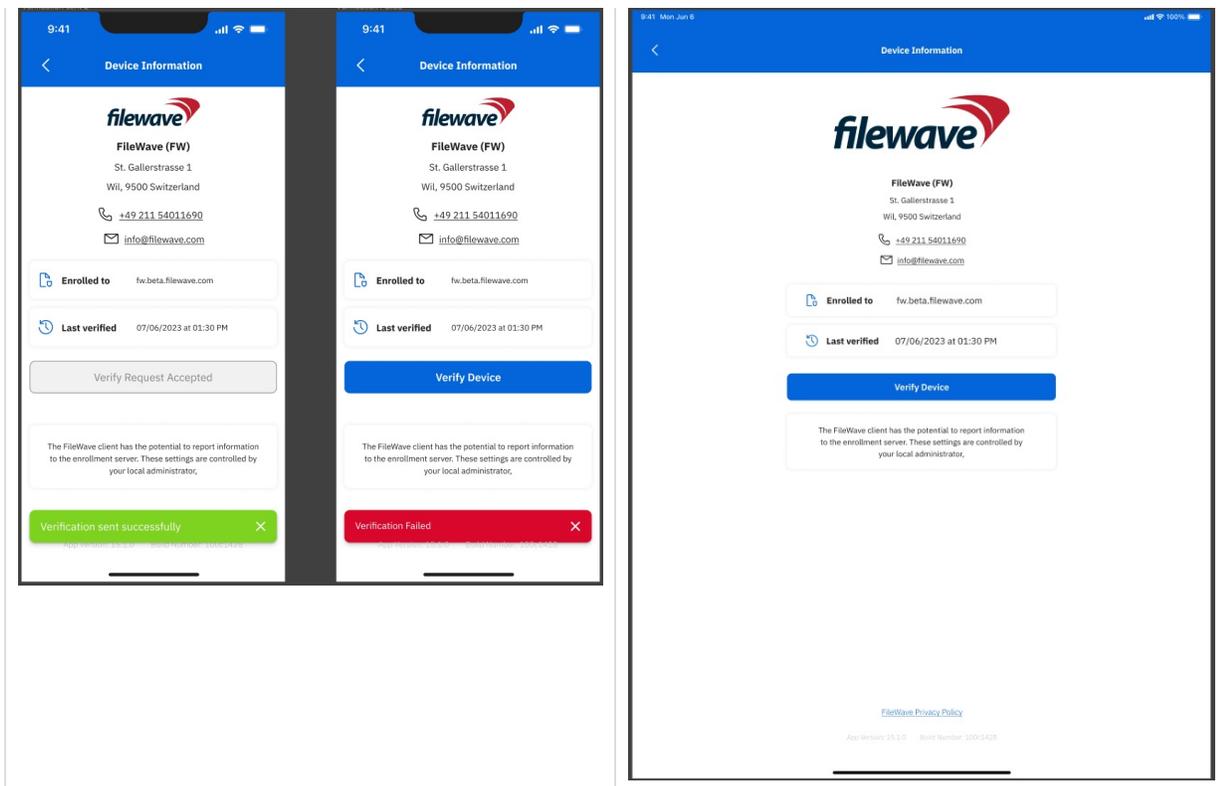
The "Device Information" page serves as a valuable tool for users to gain insights into the device's ownership, enrollment history, status, and app-related details. By presenting this information in a user-friendly manner, the page enhances user management and facilitates seamless interactions within the app's ecosystem.

Logo and primary color selection

The Adding Logo and Primary Color Selection feature enhances customization within the app by enabling users to personalize the visual representation of their profiles or entities. This feature allows users to define their brand identity and establish a cohesive look and feel throughout the app. Here's how the feature works:

- Logo Upload: Users can upload their company or entity logo, which serves as a distinctive visual identifier. The logo could be in various formats, such as JPEG, PNG,... Once uploaded, the logo is displayed in relevant sections of the app where the user's profile or entity is showcased.
- Primary Color Selection: Users can choose a primary color that resonates with their brand's identity. This color becomes the dominant hue used for interface elements such as buttons, headers, and accents throughout the app. Users can select the color from a color palette or enter a custom color code.
- Visual Consistency: The uploaded logo and selected primary color are applied consistently across various sections of the app. This ensures a coherent and visually pleasing experience for users and anyone interacting with their profiles or entities within the app.
- Personalization: By allowing users to upload their logos and select a primary color, the app offers a personalized touch, making users' presence and contributions stand out within the app's environment.
- Branding Impact: The combination of a unique logo and primary color strengthens brand recognition and recall, creating a strong visual association with users' profiles or entities.
- User-Friendly Interface: The process of adding a logo and selecting a primary color is designed to be intuitive, with clear instructions and user-friendly controls that guide users through the customization process.
- Flexibility: Users can update or modify their logo and primary color selection at any time to reflect changes in their branding or preferences.

The "Add Logo and Primary Color Selection" feature empowers users to infuse their identity into the app's interface, enhancing recognition, and contributing to a personalized and engaging experience for both users and their audiences.

How do I change the logo and primary color? It's easy if you follow this guide: Setting the Primary Color, Name and Logo in Kiosk/App Portal (15.3+) but if you are a Hosted customer, you will need Customer Technical Support to help you because you'll need to change files on the server itself.

Related Content

- Kiosk
- FileWave Kiosk for macOS and Windows overview (15.3+)

App Portal / Kiosk v2 transition (15.3+)

Evolution of App Portal for iOS/iPadOS: Transition from Technical Preview to Official Release

In our commitment to continuous improvement and user-centric development, we are excited to announce a significant milestone in the journey of App Portal for iOS/iPadOS. Starting from version 15.3.0, App Portal has officially graduated from its technical preview phase, now standing as a fully realized and polished application.

Key Points:

1. From Technical Preview to Real App:
   In previous releases, App Portal for iOS/iPadOS was introduced as a technical preview, allowing users to explore its functionalities and provide valuable feedback. With version 15.3.0, we are thrilled to elevate App Portal to the status of a fully-fledged, official application.
2. Enhancements and Stability:
   The transition signifies not only the culmination of user feedback but also substantial enhancements and optimizations to ensure the stability, reliability, and performance of the App Portal for iOS/iPadOS.
3. A Seamless User Experience:
   Users can now enjoy a seamless and refined experience with App Portal, harnessing its full potential for efficient and intuitive mobile application management.5.3.0 and beyond.

Desktop Kiosk for macOS and Windows

The transition from App Portal version 15.0.0 to version 15.3.0 represents a significant evolution, akin to the transformative experience observed when comparing the Android Play Store and the App Store. This upgrade places a primary emphasis on completely overhauling the user interface (UI) to provide a sleek, contemporary, and exceptionally intuitive platform, aligns with our users' evolving expectations.

While version 15.0.0 laid the groundwork for functionality, version 15.3.0 responds to user feedback and industry trends, recognizing the demand for a more sophisticated and streamlined UI. Here's how the new version draws inspiration from the design excellence of both the Android Play Store and the App Store:

1. Unified Design Paradigm:
   The UI design seamlessly unifies elements from both the Android Play Store and the App Store, creating a harmonious blend of familiarity and innovation.
2. Intuitive Navigation:
   Building on the success of version 15.0.0, version 15.3.0 introduces a reimagined navigation system with intuitive menus, recognizable icons, and seamless flows, akin to the ease of navigation in both app marketplaces.
3. Visual Delight:
   The new UI isn't just functional; it's a visual delight with engaging animations, subtle transitions, and tastefully curated visuals, providing an immersive experience similar to the Android Play Store and the App Store.
4. Enhanced Customization:
   Version 15.3.0 brings enhanced customization options, allowing users to personalize their interface similar to the Android Play Store and the App Store, fostering a sense of ownership over their app experience.
5. Responsive Design:
   Like the responsiveness of the Android Play Store and the App Store across devices, our app now adapts seamlessly to various screen sizes, ensuring a consistent and optimized experience.

Additionally, with version 15.3.0, we've introduced some enhancements. Users will now receive clear indications when installing certain filesets that a reboot is required for optimal performance. This improvement ensures transparency and empowers users with the necessary information for a seamless experience. By incorporating these enhancements, App Portal solidifies its commitment to delivering a cutting-edge, user-friendly experience that aligns with evolving user expectations and industry trends. Version 15.3.0 sets the stage for a future where App Portal evolves dynamically, staying at the forefront of user-centric innovation.

> ⚠ Please note that there are some pre-15.3 customizations that no longer have any effect as well as one customization that can result in seeing both the old and new Kiosk on a macOS or Windows system. Please review the related content below.

Related Content

- Removing pre-15.3 Kiosk Customizations (macOS/Windows)
- Setting the Primary Color, Name and Logo in Kiosk/App Portal (15.3+)
- Applications Preventing Reboot (macOS/Windows)

FileWave App Portal for iOS (IPA Install)

> ❗ For FileWave 15.3.0+ please note that the IPA is automatically deployed. Once 15.3.0 is released please see: Automatic updating of iOS/iPadOS Kiosk (15.3+)

Once an iOS/iPadOS device is enrolled, the FileWave App Portal will be automatically installed. The version installed after enrollment is a Web Based Application, allowing users to self-install associated Applications, Profiles, and Books or trigger a Verify.

FileWave offers an Enterprise version of the App Portal, which is available from the FileWave Download pages (there should be a version to match the server release). The Enterprise version additionally offers:

- Location Tracking (Limitations apply)
- TeamViewer Notifications

> ⚠ Location Tracking relies upon the user accepting both Location Tracking and allowing the FileWave App Portal access to the location data. If users do not accept these, no Location Data will be received. These are limitations imposed by Apple.

> ❗ The IPA is signed for a year on creation. The signature is renewed by FileWave for versions that are still supported. It is therefore necessary to re-download the Enterprise version and redistribute it, prior to or at the time of expiry, for the App to continue to function. The Web App version has no such consideration.

Installing the Enterprise FileWave App Portal

Requirements

- iOS IPA available from FileWave Downloads

Creating the Enterprise FileWave App Portal Fileset

From the FileWave Admin application, navigate to:

- Filesets > New Mobile Fileset > Enterprise

Two options are available:

- Import a local .ipa file
- Use a remote .ipa file
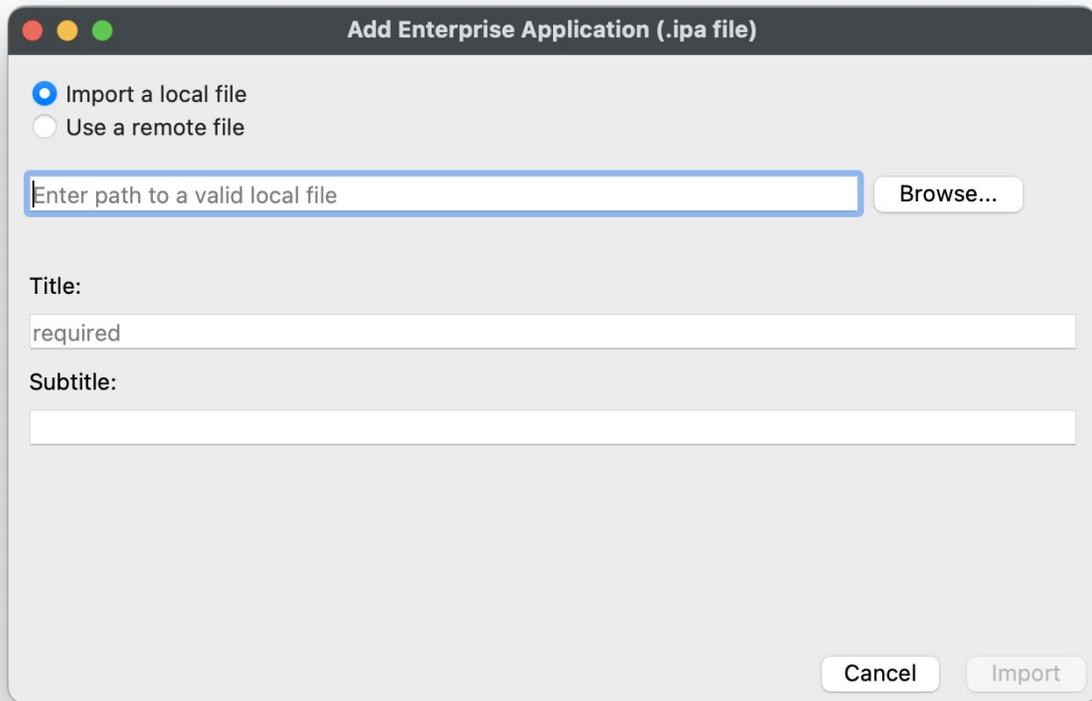
Local File

The IPA may be downloaded from the FileWave downloads page. Select the IPA version to match the FileWave Server version.

From the Fileset creation window, browse to the location of the downloaded file, choose 'Import' and then 'Done'

Remote File

The file may be placed in a remote location. This could be either a chosen server or directly from the FileWave Download site.
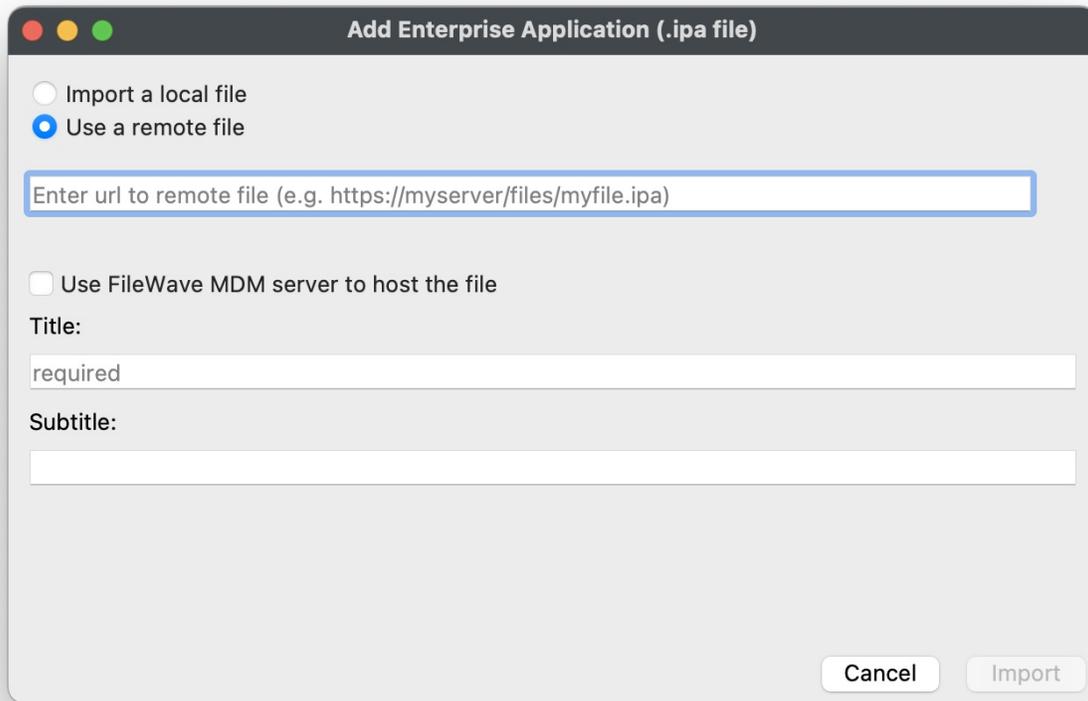
Remote Server

If using a chosen server, download the IPA as above, import it into the chosen server, and add the URL path to the Fileset.

FileWave Download Site

If using the FileWave Download site, use the provided static path from the Downloads page as the URL

> ⚠️ It is also possible to tick the box to 'Use FileWave MDM server to host the file'. This is essentially the same as the local process. FileWave will then pull the IPA from the download to create a local file for distribution from the FileWave Server.

Remote vs Local Considerations

Local

The FileWave server is used to host the IPA, the IPA is directly stored on the FileWave Server and delivered from the FileWave Server to devices.

This option has the following considerations:

- iOS devices quantity is high (>1,000) and external traffic should be kept to a minimum
- No downstream internet connection usage, potentially upstream usage if the FileWave MDM server is exposed to the internet, and devices are outside the internal network.

Remote

The FileWave Server does not store the IPA, but instead sends the device details of where to pull the IPA.

Considerations in this instance are:

- iOS devices quantity is high (>1,000) and to avoid saturating the FileWave MDM Server with requests for the IPA
- It is not a concern that each device will make its own connection to the download location
- Light on the FileWave MDM Server, since load and speed of distribution relies on the downstream internet connection to devices.

Deploying the Enterprise FileWave App Portal Fileset

Deployment of the IPA is as simple as creating an association, either directly or via a group. On receipt of the IPA Fileset, the device will automatically remove the Web App Portal and it will be replaced with the IPA version. Similarly, if providing an updated version,

the new version may be associated and the old association removed.

User Experience

It should be expected the user will receive the following prompts, due to TeamViewer integration:
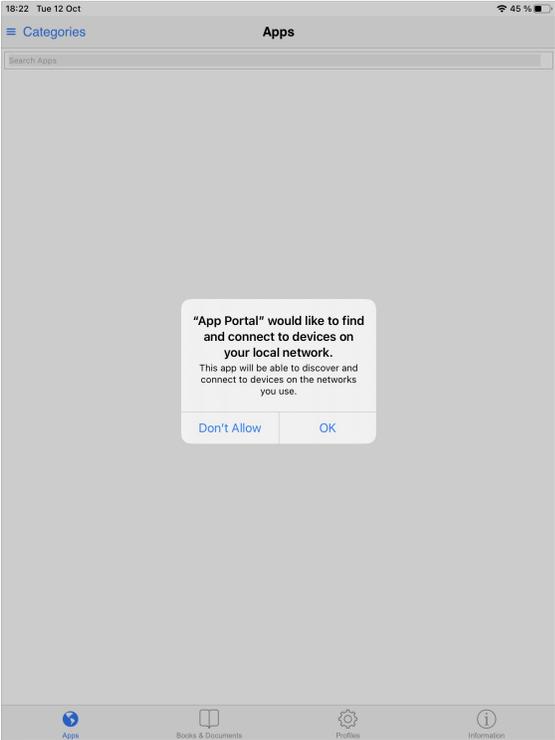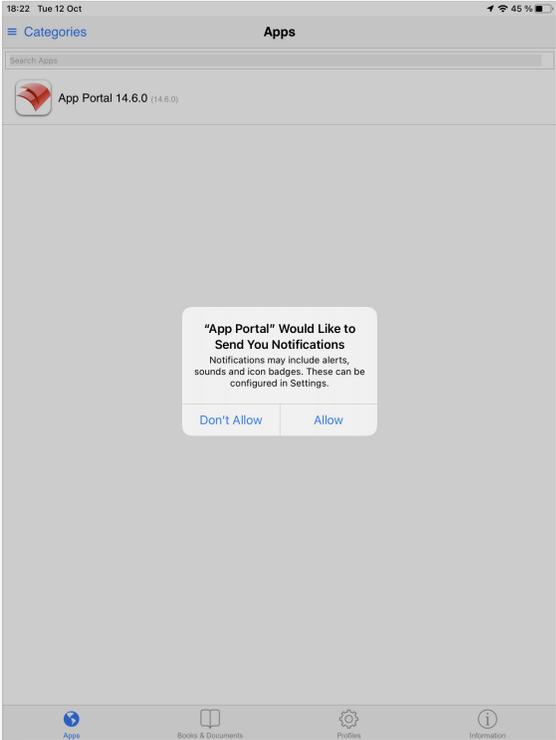
They should also expect to receive the following due to Location Tracking:

Search Apps

App Portal 14.6.0 (14.6.0)

Allow "App Portal" to use your location?

The FileWave client has the potential to report information (including location) to the enrollment server. These settings are controlled by your local administrator.

⌖ Precise: On

Mairie de Hoenheim

AV. DU RIED

RUE WOLFF

RUE NEU

D468

AV. DU GÉNÉRAL DE
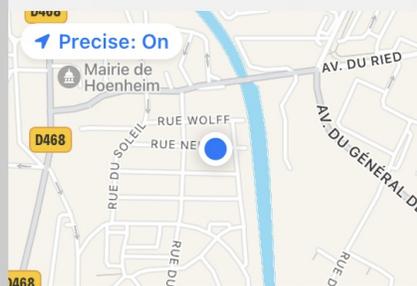
Allow Once

Allow While Using App

Don't Allow

Apps   Books & Documents   Profiles   Information

Additionally, they should be prompted regarding the ability to continue Location Tracking when the App is not in use:
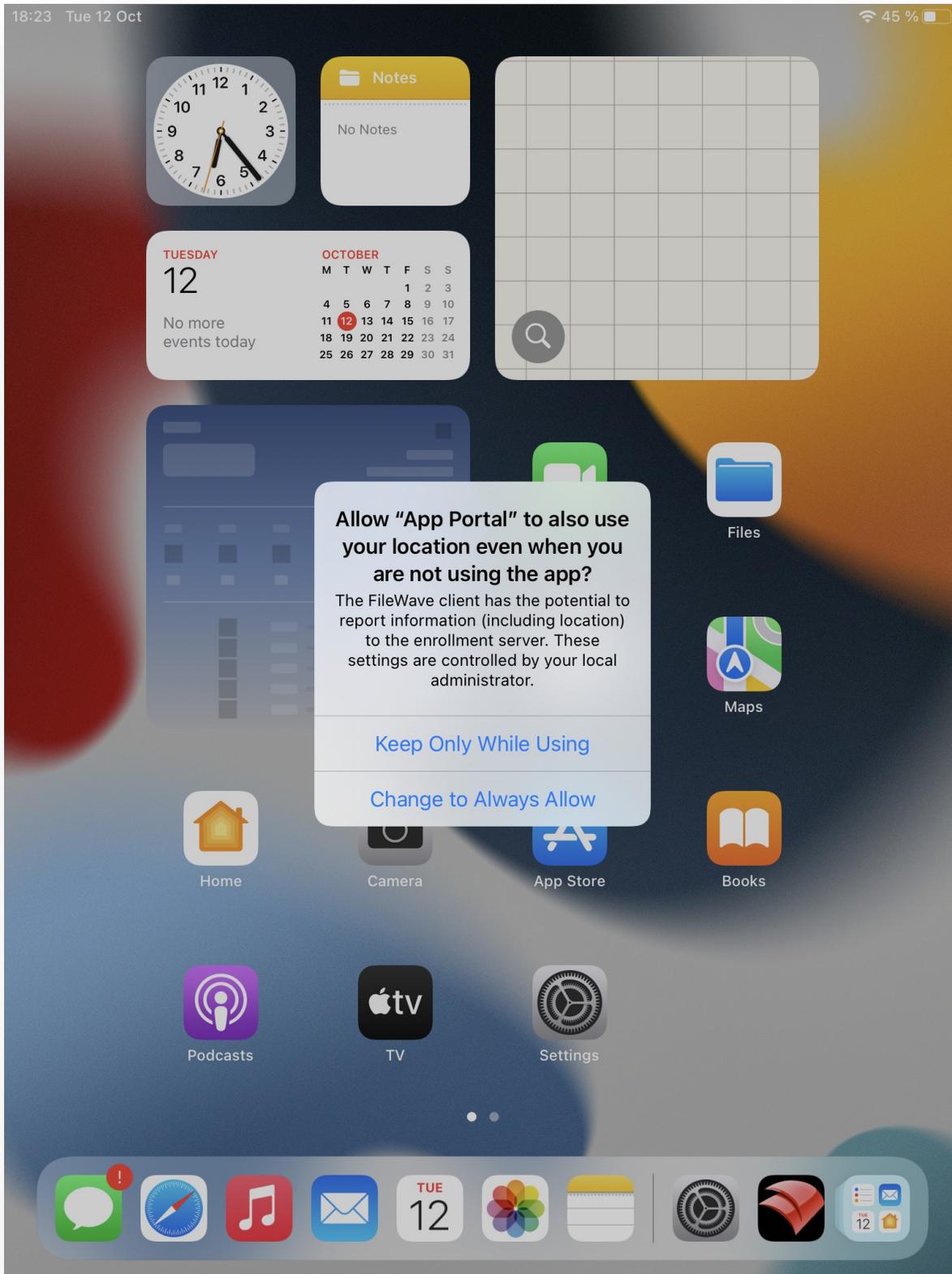
⚠ Location Tracking also relies upon the device being configured as such, as highlighted in the KB: Location Tracking

App Portal IPA Removal

If it is chosen that the IPA is no longer a requirement, removal of the association for the IPA will not only remove the IPA but the Web App version will once again be automatically deployed to devices.

IPA Expiry

If the installed App Portal IPA is allowed to expire, the App will fail to launch and the users will be prompted:
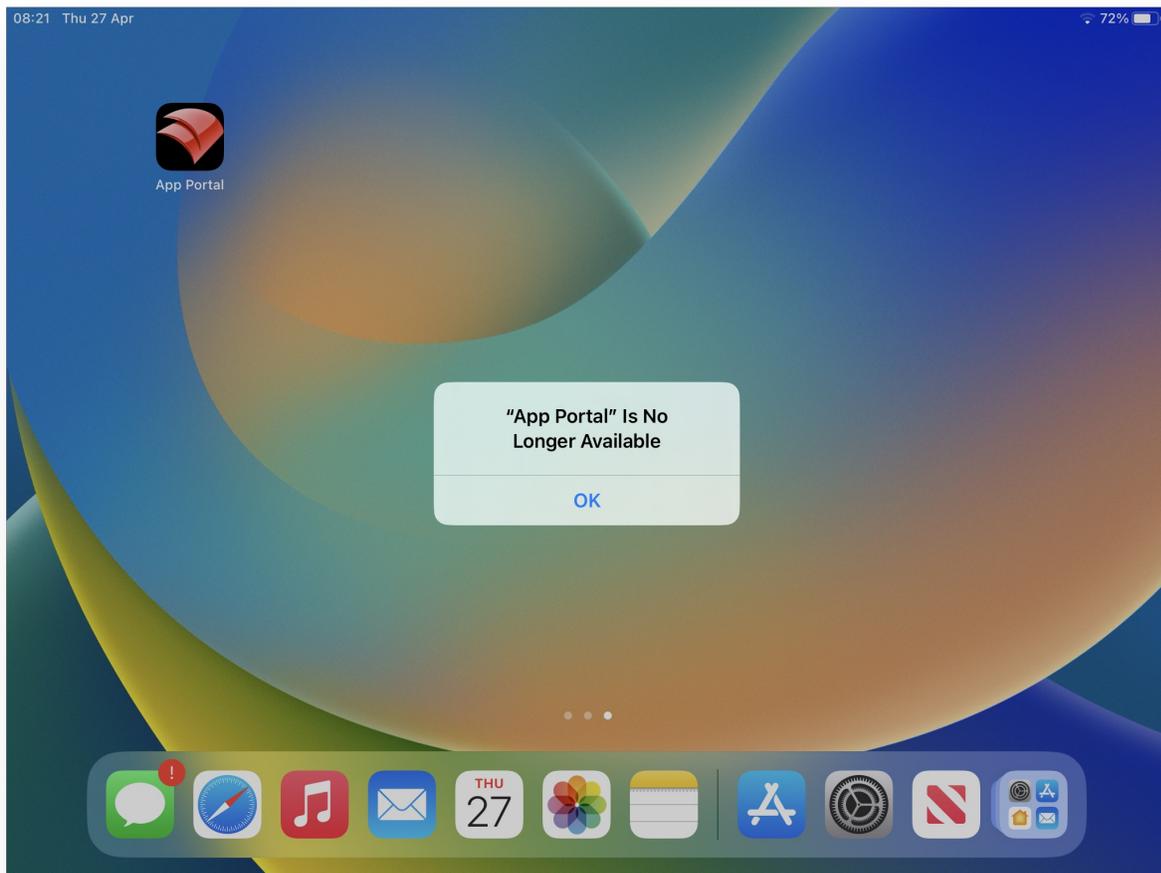
Deploying a newly downloaded IPA to devices should address this.

Automatic updating of iOS/iPadOS Kiosk (15.3+)

What

The New App Portal in FileWave 15.3.0 represents a significant upgrade designed to enhance the management and distribution of apps across iOS/iPadOS devices. This feature aims to simplify app deployment, allowing for a more streamlined and efficient process for administrators and end-users alike. By automatically installing the New App Portal on managed devices, FileWave ensures that users have access to the latest and most relevant applications required for their tasks, without the need for manual updates or installations.

When/Why

In order to simplify and ensure that the Kiosk is always current we have implemented a feature that will automatically handle the updates to the Kiosk so that you no longer need to worry about the .IPA on iOS or iPadOS.

How

- FileWave 15.3.0 will automatically install the new App Portal to all iOS/iPadOS devices managed by FileWave.
- The Webclip to the old App Portal will be removed from the devices.
- Filesets of the previous native App Portal will not be impacted by this change, they will still be installed on associated devices. Administrators can remove these associations to keep only the new App Portal on the devices.
- Every upcoming release of the new App Portal will be automatically deployed to all iOS/iPadOS devices.

Related Content

- [Setting the Primary Color, Name and Logo in Kiosk/App Portal (15.3+)](#)
- [Resolving SSL and Manifest Validation Errors with FileWave Kiosk Installation (15.3+)](#)
- [Default TCP and UDP Port Usage](#)

FileWave Kiosk displays Company Information

Description

As an enhancement to the FileWave Kiosk / App Portal on macOS, Windows, and iPadOS/iOS, version 15.4 includes a view for Organisation Information.

Information

Details for the organisation may be filled out in the Preferences tab of FileWave Central for Organisation Info. These details were already available for users to observe in the iOS Kiosk App. Now, they are available also in the the computer Kiosk for a uniform experience. To edit what is shown simply go to Preferences -> Organization Info in FileWave Central and you can update the information.
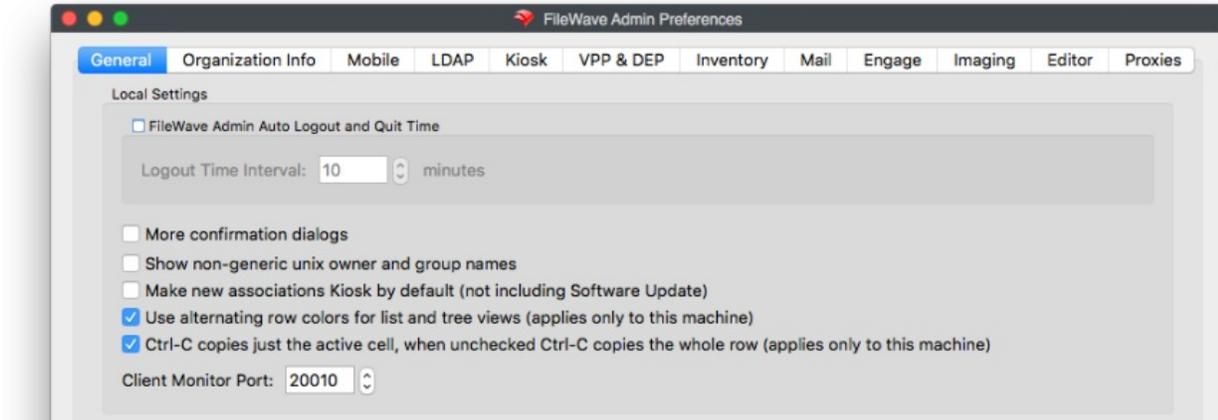
macOS and Windows
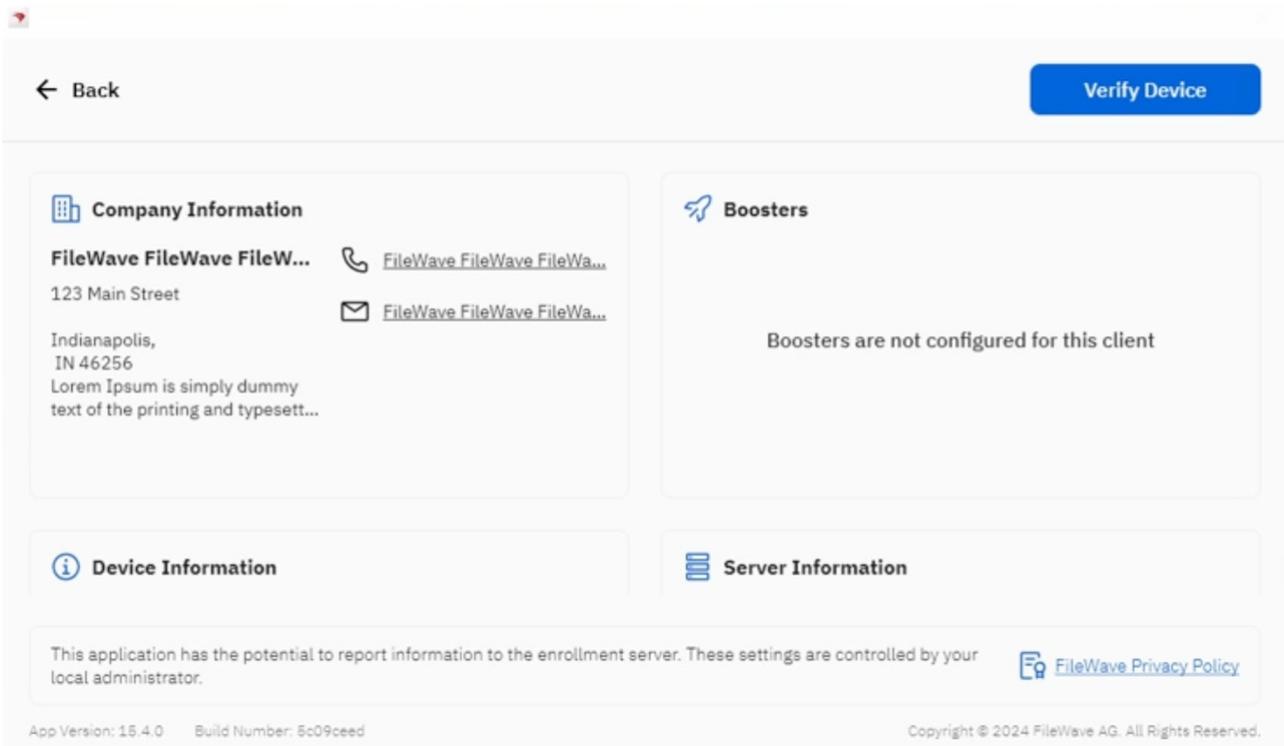
Mobile like iOS and iPadOS

FileWave
FW
123 Sesame Street
New York, NY 10028

📞 **555-555-1212**

✉️ **help@me.com**

🗎 **Enrolled to**   support2.filewave.net

🕘 **Last Verified**   Jun 11, 2024 17:00:17

Verify Device

This application has the potential to report information to the enrollment server. These settings are controlled by your local administrator.

FileWave Privacy Policy

Kiosk Package Updates Category (macOS PKG)

Description

Kiosk may show 'Package Updates' for associated PKG Filesets where older PKGs are already installed on the device.
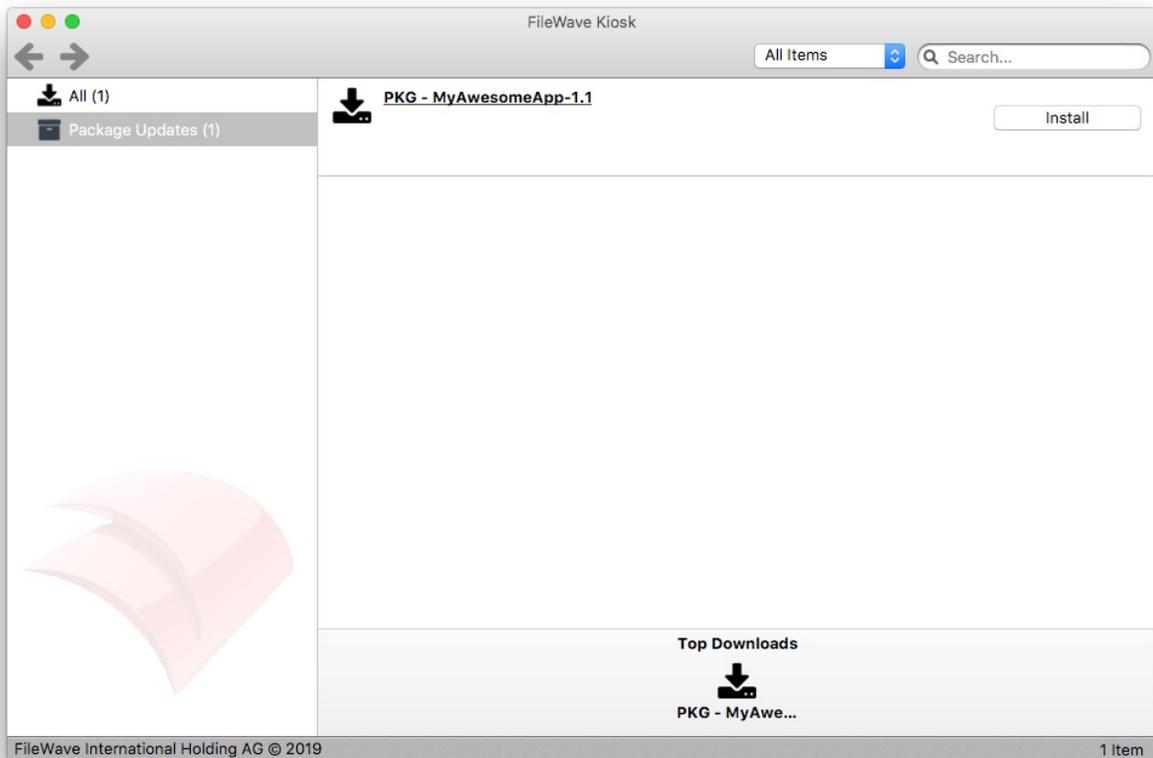
Information

When a PKG Fileset is associated, as a Kiosk item, to a device, the FileWave Client will check the PKG receipts on the device. If a matching receipt is found and the version of this installation is older, the Kiosk will show this as a 'Package Update'. This is an automated category and therefore will not be visible in either:

- Admin > Preferences > Kiosk > Categories
- Admin > Fileset > Properties > Kiosk > Category

For example, a device that already has version 1.0 of My Awesome App, installed as a PKG.
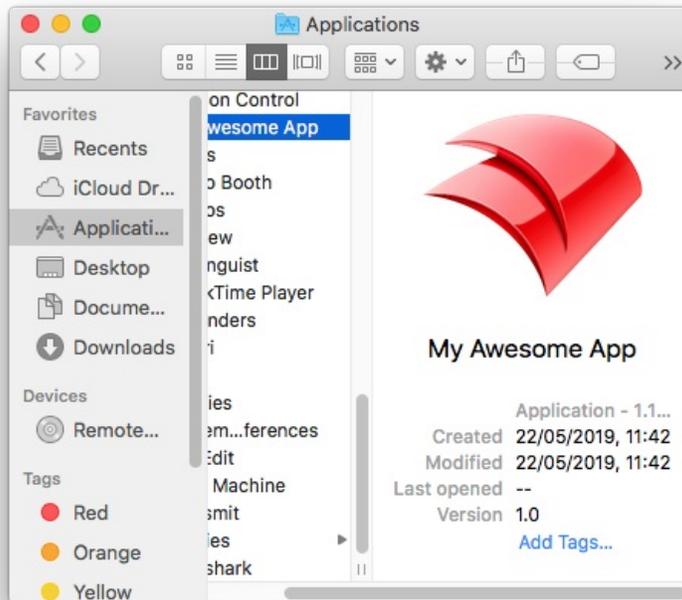
Since installation was by PKG, there should be a receipt. If you know the identifier of the PKG, you may use Terminal to confirm this:

```
# pkgutil --pkg-info com.myawesomeapp.pkg.app
package-id: com.myawesomeapp.pkg.app
version: 1.0
```

Upload and create a Kiosk association of a newer PKG version of the App
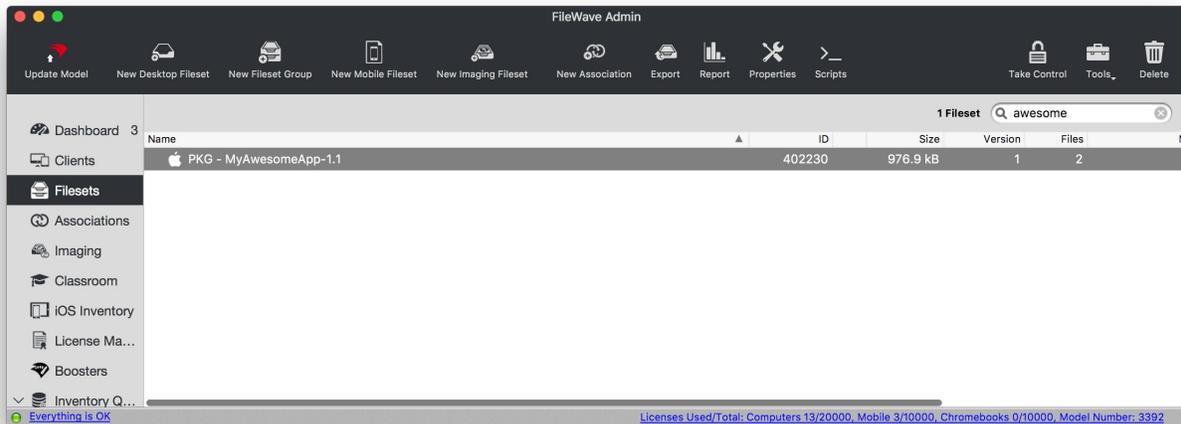
When the device receives this new association, the Kiosk drop down menu will add an additional option 'Available Updates':
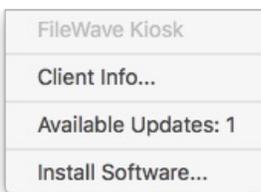
Clicking on this will open the Kiosk Installer window, showing the added 'Package Updates' category as shown above in the description.

It does not matter how the older PKG was initially installed, e.g. through FileWave or locally. This is particularly useful for

warning users, who have the option to install their own PKG and can see that an update has been offered through FileWave.

PKG version is set within the PKG by the developer of the device. If the developer does not increase the number from prior versions, then the Fileset cannot be shown as a 'Package Update'.

PKG installers have their own version and this need not match the App version as shown in Finder.

Kiosk should show only Kiosk associated Apps (15.3+)

What

Before FileWave 15.3, apps that are directly deployed to the device and have the "Allow removal" option checked are shown in Kiosk. If the user decides to remove them from their device, that app is shown in Kiosk—in case the user changes their mind and wants to have them on their device again.

When/Why

This was done before when there wasn't Verify. Now, with Verify, this process is automated (the app will be installed again after Verify), so there is no sense in showing those apps in the Kiosk anymore.

How

In FileWave 15.3+, the Kiosk shows an item in the app list only if it is Kiosk-associated with the device. Non-kiosk-associated items do not appear on the list. This is more straightforward behavior, but it may be confusing for someone who was used to seeing apps in the Kiosk before that are now not showing.

Related Content

- Kiosk

Setting the Primary Color, Name and Logo in Kiosk/App Portal (15.3+)

What

This article provides step-by-step instructions for configuring the new Kiosk2 and App Portal introduced in FileWave 15.3.0, including how to set a primary color, name, and logo for the App Portal / Kiosk on FileWave v15.3.0+.

When/Why

For brand consistency, options exist (depending on device type) for configuring:

- Colour
- Name
- Logo that appears in macOS/Windows menu bar/system tray
- Logo that appears in macOS/Windows Kiosk application

> ℹ️ When referring to a custom logo, the expected image should be: 112 x 96, 8-bit/color RGBA, non-interlaced
> Some changes are server side. If hosted, please contact Technical Support for assistance.

How

▼ iOS & iPadOS

Devices of this nature use the App Portal. As of FileWave 15.3, the App Portal IPA is automatically delivered to devices and will automatically update.

Customisation Options:

- Logo (Default to FileWave Logo if no custom logo set)
- Colour

Logo Customisation

> ℹ️ If you are a Hosted Customer then you will need Technical Support to assist with these changes for now.

Direct editing of the IPA is not feasible. As such, the logo needs to be hosted from the FileWave Server. The chosen image needs to be placed in the following location:

`/usr/local/filewave/django/static/kiosk2/images/`

For the IPA to utilise this image, edit the the following file:

`/usr/local/filewave/django/filewave/settings_custom.py`

The line may be added to the bottom of the custom settings file and needs to include the FQDN of the FileWave Server, along with the named image in the format:

KIOSK_LOGO_URL = 'https://[fully_qualified_domain_name]/filewave/static/kiosk2/images/[file name]'

For example, imagine those details are:

- image.svg
- demo.filewave.ch

The line added will then appear as:

`KIOSK_LOGO_URL = 'https://demo.filewave.ch/filewave/static/kiosk2/images/image.svg'`

Supported file formats are:

```
- SVG
- JPEG
- PNG
```

```
  – GIF
  – Animated GIF
  – WebP
  – Animated WebP
  – BMP
  – WBMP
```

Customise Colour

Colour may be customised by editing the same custom settings file. The colour should be presented in hex of either 6 or 8 characters. The line to add (again this may be added to the bottom of the custom settings file), should be of the format:

KIOSK_PRIMARY_COLOR = '[hex colour]'

 For example:

```
KIOSK_PRIMARY_COLOR = '#e580ff'
```

or

```
KIOSK_PRIMARY_COLOR = '#FF0000CC'
```

> ℹ️ The FileWave Server Service should be restarted after modification of the custom settings python file.

▼ macOS

macOS access the Kiosk by way of the Menu Bar

macOS Kiosk App Logo

Customisation Options:

- Logo (Default to FileWave Logo if no custom logo set) - Note that this logo is the one seen in the app. To customize the one in the menu bar you must look at Local macOS Logo below.
- Colour
- Kiosk App Name

All 3 items can be specified through a single file:

macOS:

```
/usr/local/sbin/FileWave.app/Contents/MacOS/FileWave\ Kiosk.app/Contents/custom/settings.json
```

The contents should appear similar to:

```
{
    "logo_url": "https://my.company/logo.svg",
    "primary_color": "#0465DA",
    "kiosk_app_name": "My Kiosk"
}
```

Defined as:

Key Name	Key Value	Default
logo_url	URL path to hosted custom image	FileWave Logo
primary_color	Colour in hex format, e.g. #RRGGBB	#0465DA
kiosk_app_name	Text Name of opened Kiosk Window	FileWave Kiosk

> ℹ️ It may be necessary to reboot the computer before changes take effect; particularly if customisation of the older Kiosk was removed at the same time that the new Kiosk and customisation is added.

> ✅ If the custom icon is added as per the instructions for i(Pad)OS devices, the same URL may be used in the above settings.

macOS Menu Bar Logo

The above method requires a logo to be hosted. However, it may be desirable to provide a local logo instead. The following methods provide a pre-configured Fileset to achieve this. The logo in this case is the icon in the menu bar only.

FileWave Kiosk2 Menu Bar Icon.fileset.zip

The Fileset needs to be edited prior to deployment. Out of the two files shown, the chosen custom image should be named to match the first png: 'tray-icon-macos.png'. Leave the other file in place.
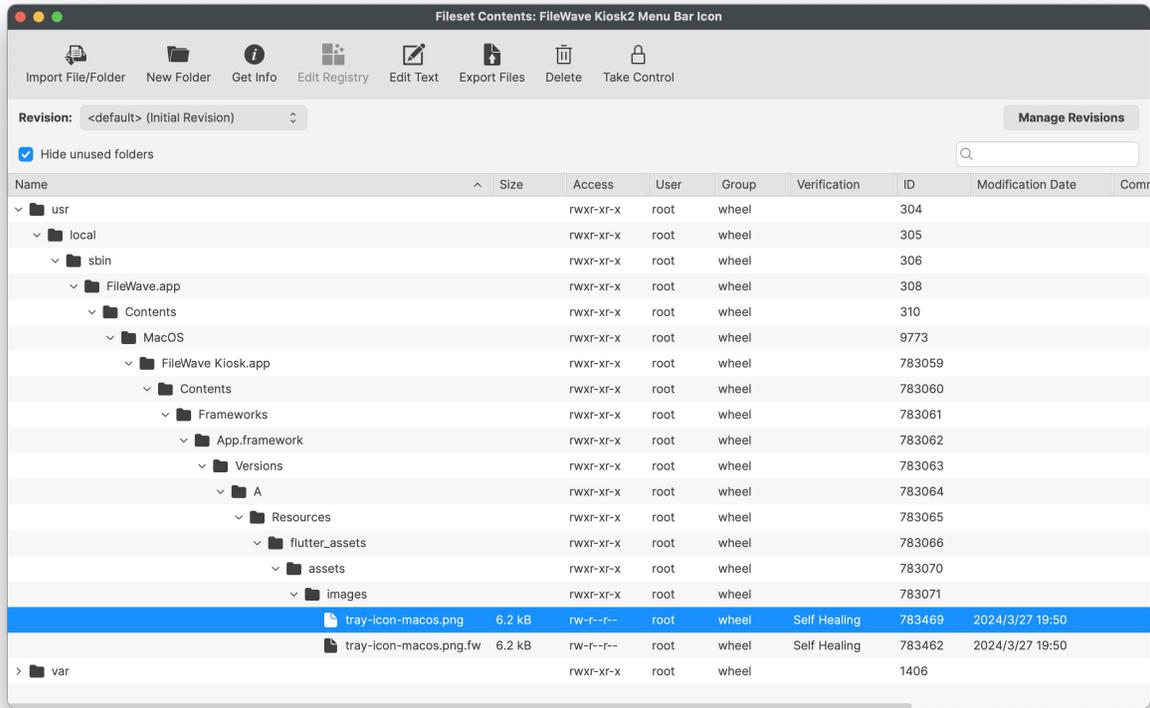
 Dragging the custom icon over the shown file should prompt a message as below. Select Replace.
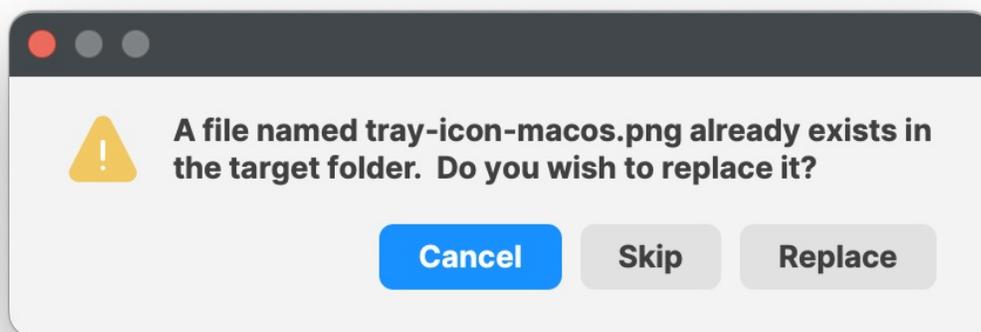
Associate with a test device, and once happy with the look, you can broaden the association.

The Fileset is designed to:

- Restart the Kiosk after activation, automatically ensuring the user is immediately presented with the new logo
- On de-activation, the '`tray-icon-macos.png.fw`' file will be copied to replace the custom logo (resetting the default) and again restarting the Kiosk for immediate change.

ℹ️ The new image will not show unit the FileWave Client is restarted.

▼ Windows

Windows access the Kiosk by way of a System Tray item

Windows Kiosk App Logo

Customisation Options:

- Logo (Default to FileWave Logo if no custom logo set)
- Colour
- Kiosk App Name

All 3 items can be specified through a single file:

Windows:

`C:\Program Files (x86)\FileWave\kiosk\custom\settings.json`

The contents should appear similar to:

```
{
    "logo_url": "https://my.company/logo.svg",
    "primary_color": "#0465DA",
    "kiosk_app_name": "My Kiosk"
}
```

Defined as:

Key Name	Key Value	Default
logo_url	URL path to hosted custom image	FileWave Logo
primary_color	Colour in hex format, e.g. #RRGGBB	#0465DA
kiosk_app_name	Text Name of opened Kiosk Window	FileWave Kiosk

ℹ️ It may be necessary to reboot the computer before changes take effect; particularly if customisation of the older Kiosk was removed at the same time that the new Kiosk and customisation is added.

✅ If the custom icon is added as per the instructions for i(Pad)OS devices, the same URL may be used in the above settings.

Windows System Tray Logo

The above method requires a logo to be hosted. However, it may be desirable to provide a local logo instead. The following methods provide a pre-configured Fileset to achieve this. The logo in this case is the icon in the System Tray only.

FileWave Kiosk2 Sytem Tray Icon.fileset.zip

The Fileset needs to be edited prior to deployment. Out of the four files shown, the chosen custom icon should be named to match the two icon files: 'tray-icon-dark.ico' and 'tray-icon-light.ico'. Leave the other two files in place.

ℹ️ Unlike the other images, the Windows System Tray is using '.ico' files. These need to be of equal size, e.g. 256x256, 128x128, etc.

✅ Consider exporting the current 'ico' files, editing them with the chosen logo and then upload, if unsure how to create '.ico' files.

Fileset Contents: FileWave Kiosk2 Sytem Tray Icon

Name	Size	Access	User	Group	Verification	ID	Modification Date	Comment	Creation D
Program Files (x86)		rwx------	root	staff		134			
FileWave		rwxr-xr-x	root	wheel		6156			
kiosk		rwxrwxr-x	root	admin		783471			
data		rwxrwxr-x	root	admin		783472			
flutter_assets		rwxrwxr-x	root	admin		783473			
assets		rwxrwxr-x	root	admin		783474			
images		rwxrwxr-x	root	admin		783475			
tray-icon-dark.ico	105 kB	rwxrwxr-x	root	admin	Self Healing	783489	18/10/2023 16:01		18/10/2023
tray-icon-dark.ico.fw	105 kB	rwxrwxr-x	root	admin	Self Healing	783487	18/10/2023 16:01		18/10/2023
tray-icon-light.ico	104.4 kB	rwxrwxr-x	root	admin	Self Healing	783490	18/10/2023 16:01		18/10/2023
tray-icon-light.ico.fw	104.4 kB	rwxrwxr-x	root	admin	Self Healing	783488	18/10/2023 16:01		18/10/2023

Dragging the custom icon over the shown files should prompt a message as below. Select Replace.

Replace

A file named tray-icon-dark.ico already exists in the target folder. Do you wish to replace it?

Replace   Skip   Cancel
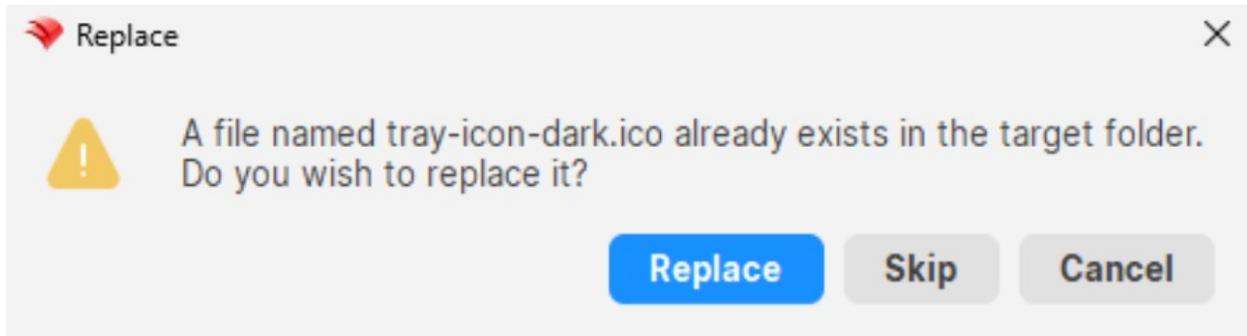
Associate with a test device and once happy broaden the association.

ⓘ The new image will not show unit the FileWave Client is restarted.

Related Links

- [FileWave Kiosk for macOS and Windows overview (15.3+)](#)
- [Removing pre-15.3 Kiosk Customizations (macOS/Windows)](#)

App Portal on iOS/iPadOS devices that don't support the application (15.3+)

What

If you have an iOS/iPadOS device that is older than what is officially supported by FileWave and you are working ok but you can't get the Kiosk (App Portal) to install.

When/Why

You can revert for now to the old WebClip using the documented process here. This is a temporary workaround.

How

It is possible to activate Web Clip of an old App Portal on devices which don't support new App Portal.

1. Create new Apple Profile → Web Clip Profile (can be done in both FileWave Central and Anywhere)

2. Fill in URL field with: `https://fqdn:20443/ios/recommended_apps/%device_id%#apps` replacing "fqdn" with your server's DNS name.
3. Deploy profile to desired device(s).

Related Content

- [Automatic updating of iOS Kiosk (15.3+)](#)

# Applications Preventing Reboot (macOS/Windows)

Description

Before the FileWave client reboots a computer, where the Fileset properties are set to reboot, a warning can be expected to 'Quit All Running Processes' with a list for the user to address.

Some processes do not respond to FileWave Clients requests to quit when "Quit all" or "Quit Application" is being used, because they are being immediately restarted by LaunchDaemon after quitting, or do not respond properly to the signals sent to them by FileWave. This can result in users being blocked from installing FileSets with the "Reboot" Flag set.

Information

Although this warning is usual, there are instances where certain apps or processes can appear in the queue that are unexpected, for example:
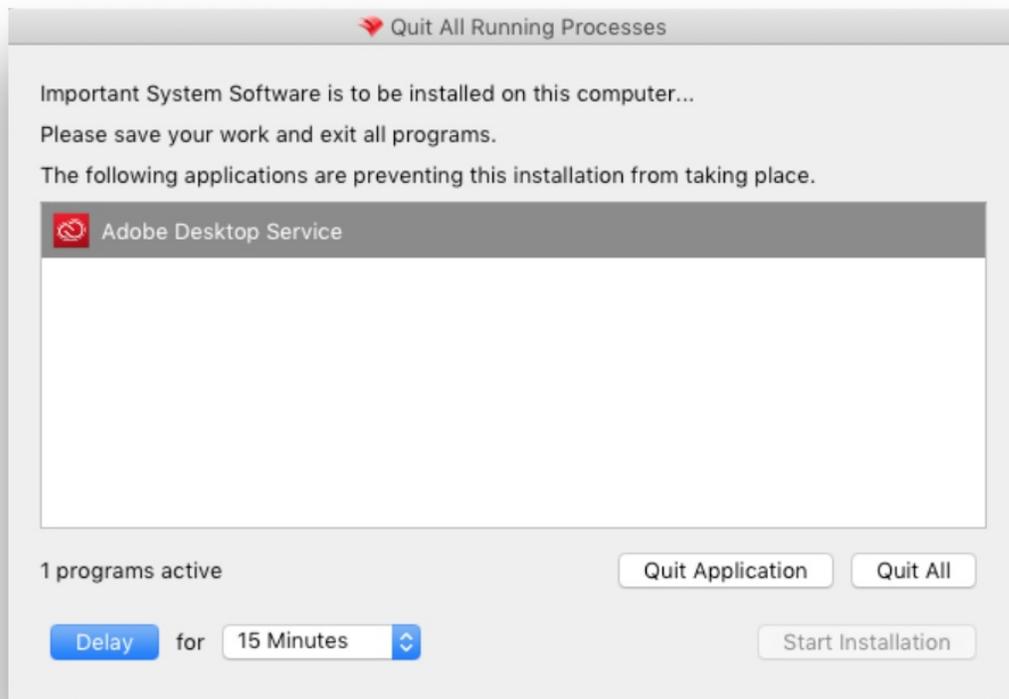
As a service, this can show even though the user is not using Adobe Apps at the time.

Solution

Requirements to prevent this experience are:

- Customised Kiosk Fileset
- macOS client - Bundle ID of the App or Service to exclude
- Windows client - Name of running process

A settings file may be configured to allow for certain apps. Create a fileset containing the settings.ini and then associate it to the desired clients. The directory where the file should be placed depends on the operating system:

macOS

```
/usr/local/sbin/FileWave.app/Contents/Resources/fwGUI.app/Contents/custom
```

Windows

```
C:\Program Files (x86)\FileWave\custom
```

The changes will only be visible in the desktop Kiosk after the user logs out and logs in again.

To exclude an unexpected process from being listed and preventing the reboot, first obtain the Bundle ID or process name of the listed App or Service.

Obtain Bundle ID or Process Name

Windows

- Send any Fileset that has its Fileset Properties set to 'Requires Reboot' to the test machine
- Hover over the desired process to exclude
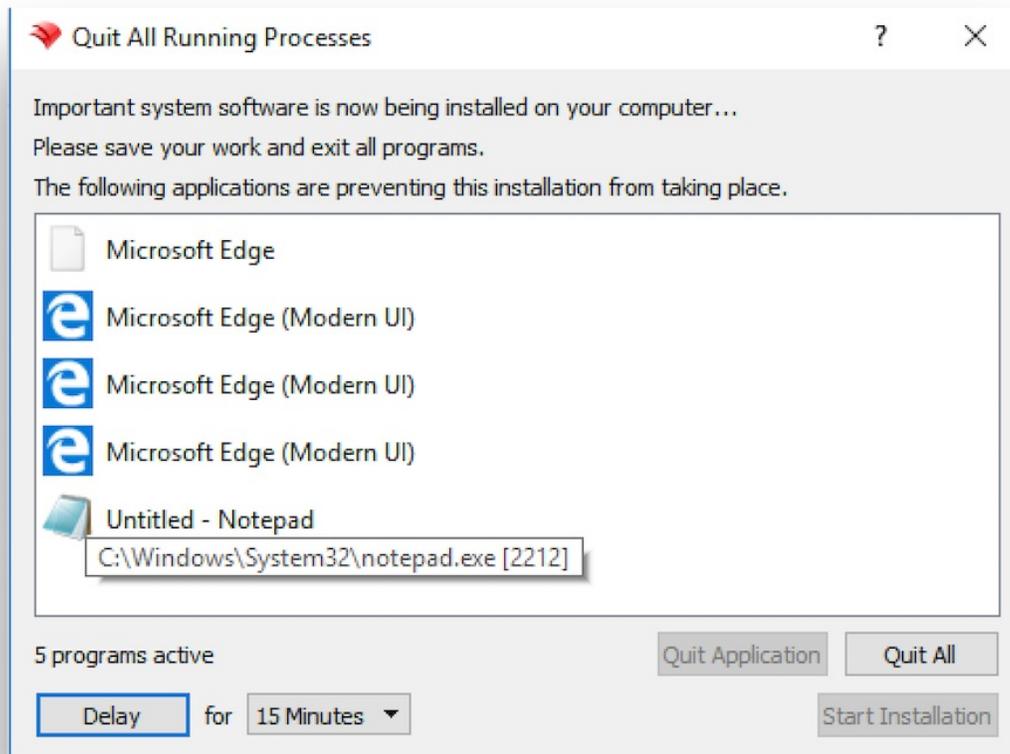
In this example, to exclude notepad you require: 'notepad.exe'

macOS

- Change a test device's Debug Level to 99 through the Client Monitor
- Send any Fileset that has its Fileset Properties set to 'Requires Reboot' to the test machine
- Once device prompts to quit applications, open the the FWGUI.log found in the users Application Support directory.

```
~/Library/Application\ Support/FileWave/FWGUI.log
```

There should be lines similar to the following, for each process that has prevented the reboot:

```
|DEBUG|INSTALLER|[addProcessInfo] new process added /Applications/Utilities/Terminal.app pid 67345 bundleID
com.apple.Terminal
|DEBUG|INSTALLER|[addProcessInfo] new process added /Applications/Preview.app pid 73400 bundleID com.apple.Preview
```

```
|DEBUG|INSTALLER|[addProcessInfo] new process added /Applications/TextEdit.app pid 73718 bundleID
com.apple.TextEdit
```

Configure 'settings.ini'

The 'settings.ini' file should be configured with the following format (Bundle IDs or processes being a comma separated list):

```
[General]
ignoredApplications=com.bar, com.foo
```

For the example of Windows Notepad, it should read:

```
[General]
ignoredApplications=notepad.exe
```

> ✅ To avoid the newer default processes Microsoft now include, consider ignoring the following:

```
[General]
ignoredApplications=StartMenuExperienceHost.exe,SearchApp.exe,TextInputHost.exe
```

For the example of macOS TextEdit, it should read:

```
[General]
ignoredApplications=com.apple.TextEdit
```

Combining this with current generic settings, it may look something like:

```
show_top_downloads=false
window_title="My Customized Kiosk"
hide_system_tray=false

[General]
ignoredApplications=com.adobe.acc.AdobeDesktopService,com.apple.TextEdit
```

Deployment

Deploy the 'settings.ini' file to the following locations:

macOS

```
/usr/local/sbin/FileWave.app/Contents/Resources/fwGUI.app/Contents/custom
```

Windows

```
C:\Program Files (x86)\FileWave\custom
```

Troubleshooting

# Checking what version of iOS/iPadOS App Portal is being pushed out (15.3+)

What

How do I know what version of App Portal (Kiosk) is being pushed out to iOS/iPadOS devices?

When/Why

After a FileWave upgrade, there likely will be a new version of the iOS/iPadsOS IPA pushed out through the automatic upgrade process. Because this process is a bit invisible, verifying what the server believes is the version it is pushing out is sometimes necessary.

How

In FileWave Central, go to Assistants -> Manage Administrators and then pick your user account, though depending on permissions, you may want to use fwadmin. Grab the base64 token seen below in the image, including the = at the end of it.
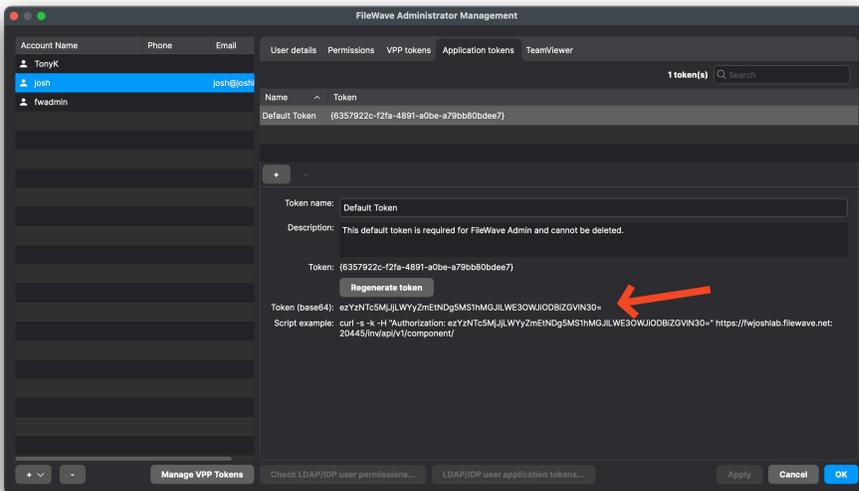
Now you can ask the server. Replace `ezk5ODxxxmE0OyyyyyyyOX0=` below with your token, and replace `support2.filewave.net` with your server's DNS name.

Here is a command you can run in Terminal.app on a macOS system or any system that has curl installed:

```
curl -H "Authorization: ezk5ODxxxmE0OyyyyyyyOX0=" https://support2.filewave.net/filewave/api/kiosk/internal/kiosk-status
```

Here is a PowerShell example of the same command:

```
# Define the URL to make the request
$url = "https://support2.filewave.net/filewave/api/kiosk/internal/kiosk-status"

# Define the headers, including the Authorization token
$headers = @{
    "Authorization" = "ezk5ODxxxmE0OyyyyyyyOX0="
}

# Use Invoke-WebRequest to make the GET request with headers
$response = Invoke-WebRequest -Uri $url -Headers $headers -Method Get

# Output the content of the response
$response.Content
```

The output of the command will be something like below.

```
{"installed":{"ipa_url":"https://fw-kiosk-v2-
ipas.filewave.cloud/15.3.1/App_Portal_15.3.1.ipa","ipa_md5":"81628b83dad72e274ef93ea031db1484","build_date":"2024-
03-27T20:08:49.192563Z","bundle_version":"15.3.1"},"latest":{"ipa_url":"https://fw-kiosk-v2-
ipas.filewave.cloud/15.3.1/App_Portal_15.3.1.ipa","ipa_md5":"81628b83dad72e274ef93ea031db1484","build_date":"2024-
03-27T20:08:49.192563Z","bundle_version":"15.3.1"},"available_versions":["15.3.1"]}%
```

Related Content

- [Self-Service Kiosk Overview](#)

Removing pre-15.3 Kiosk Customizations (macOS/Windows)

Prior to FileWave 15.3 you had a method of customizing the Kiosk that is not used in 15.3 and beyond. Some of the customizations will be ignored with 15.3, but there are things you'll want to potentially remove and you may want to stop pushing out customizations that are no longer enforced. This article is a paired down version of the old article and simply contains information on what customizations you might have had in the past so you can look to potentially remove the files/edits.

The look of the FileWave desktop Kiosk was customized using Qt Style Sheets. You would have had to create a file named fwGUI.qss and deploy it to clients in the right location. Several icons were also customizable by placing your custom icon in the right place with the right file name. All of this can be deployed via a Fileset.

Check out Setting the Primary Color, Name and Logo in Kiosk/App Portal (15.3+) and Applications Preventing Reboot (macOS/Windows) for the things that can be customized in FileWave 15.3 and beyond.

Deploy

fwGUI.qss -- You would have created a Fileset containing the `fwGUI.qss` and the icons, then associate it to the desired clients. The directory where these files was placed depends on the operating system. In 15.3 and beyond you don't want to push the qss file or icons out as they will have no effect on the Kiosk.

macOS

```
/usr/local/sbin/FileWave.app/Contents/Resources/fwGUI.app/Contents/custom
```

Windows

```
C:\Program Files\FileWave\custom (might be "Program Files (x86)", depending on the platform)
```

The changes will only be visible in the desktop Kiosk after the user logs out and logs in again.

Menu Bar/System Tray Settings

settings.ini -- contained generic kiosk ui settings. Especially important is that if `hide_system_tray = false` that you will end up with 2 icons for the Kiosk where one is the old Kiosk if you are on FileWave 15.3.

Example

```
window_title = "Your New Window Title"
show_top_downloads = true
hide_system_tray = false
```

- The "window_title" attribute was displayed in the menu bar menu item - as well as in the kiosk window title area.
- The "show_top_downloads" attribute was used to hide the top downloads UI on a per client basis.
- "hide_system_tray" determined whether or not there will be an icon displayed in the system tray. Setting 'false' will force the Kiosk to show even with no current association.

> Example Filesets that you might have used to push out these settings:
> Windows (64) - Kiosk_Customizer_for_Windows64.fileset.zip
> Windows (32) - Kiosk_Customizer_for_Windows32.fileset.zip
> macOS - Kiosk_Customizer_for_macOS.fileset.zip
> macOS/Windows - Applications Preventing Reboot

Customizable icons

You might have included icon files that you used to replace them in Kiosk before 15.3. These icons will not appear in 15.3 or newer.

File name	Description	Example	Pixel Size
action_back.png	Icon for the "Back" action in the Software Installation dialog		48x48

		←	
action_forward.png	Icon for the "Forward" action in the Software Installation dialog	→	48x48
background_icon.png	Icon used for the watermark	(FileWave icon)	512x512
rating_star_half.png	Icon for half a rating star	⯪	40x40
rating_star_off.png	Icon for a rating star that is off	☆	40x40
rating_star_on.png	Icon for a rating star that is on	★	40x40
tray_icon.png	System tray icon. If missing, the background_icon.png is used.	(FileWave icon)	16x16
warning.png	Warning icon, displayed when a fileset has problems (e.g. missing VPP user)	❗	64x64

Related Content

- [Setting the Primary Color, Name and Logo in Kiosk/App Portal (15.3+)](#)

Kiosk with macOS in a VM: Enabling Metal support

Description

The new Kiosk requires 'Metal' support. VMware does not support metal by default. It is possible to add support, however the details provided are Beta and not only may the VM become unstable, but the host may also Kernel Panic.

> ⛔ Use cautiously

Without the necessary configuration, the new Kiosk will not show and the user logs will show something like:

```
2024-05-13 10:40:15.324 FileWave Kiosk[1241:5674] Could not acquire Metal device.
2024-05-13 10:40:25.936 FileWave Kiosk[1241:5674] Unable to create FlutterView; no MTLDevice or MTLCommandQueue
available.
```

Configuration

The following configuration requires the VM to be shutdown (not paused) prior to editing.

Locate the containing folder of the VM. Within this will be a .vmx file. For example, if the containing VM Folder were called 'macOS 12.vmware, the file to edit should be called 'macOS 12.vmx':

```
/Users/sholden/Documents/Virtual\ Machines/macOS\ 12.vmwarevm/macOS\ 12.vmx
```

Add the following two lines:

```
appleGPU0.present = "TRUE"
svga.present = "FALSE"
```

Once added, the VM may then be booted. To remove this configuration, again shutdown the VM, edit the file, this time removing these two entries.

It may also be necessary to define the aspect ratio in the same file with the following two lines. 1920 x 1080 used as an example:

```
appleGPU0.screenWidth="1920"
appleGPU0.screenHeight="1080"
```

> ⛔ Although this may work, in practice the VM crashed too often or became seemingly unusable. Since the host may also Kernel Panic, recommendation would be to avoid configuring this on VMware test devices, unless absolutely necessary. Even then, consider undoing this once the test has been completed.

Resolving SSL and Manifest Validation Errors with FileWave Kiosk Installation (15.3+)

What

This article addresses an issue encountered during the installation of the new FileWave Kiosk v.15.3.1 on macOS and Windows devices. Users may encounter SSL errors and manifest validation failures that prevent the Kiosk from installing correctly.

When/Why

These installation errors typically occur when upgrading to FileWave v.15.3.1 and are primarily due to conflicts with certain content filters or proxy settings that block or misclassify necessary URLs. This is particularly relevant for organizations using content filtering solutions like Lightspeed, which may categorize essential URLs as unknown, thus blocking them.

```
"InstallApplication    command error         2024-04-22T07:36:38    2024-04-22T07:36:38
Could not validate manifest..An SSL error has occurred and a secure connection to the server cannot be made.
com.filewave.ios.app.kiosk2    "
```

How

To resolve these installation issues, follow the steps below:

1. Check Proxy and Firewall Settings: Ensure that your organization's proxy or firewall settings are not blocking access to FileWave URLs.
2. Whitelist Necessary URLs: Add the following URLs to the whitelist in your content filter or proxy settings:
   - `https://fw-kiosk-v2-ipas.filewave.cloud/`
   - `*.filewave.cloud`
   This change allows devices to communicate securely with FileWave servers and access the necessary resources for installing the Kiosk.
3. Reattempt Installation: After updating your settings, reattempt the installation of the FileWave Kiosk on the affected devices.

```
# Example command to verify if the URL is accessible from your network
curl -Iv https://fw-kiosk-v2-ipas.filewave.cloud/
```

If the issue persists, check your SSL certificate settings and network configuration for any discrepancies that might be interfering with secure connections.

Related Links

- Default TCP and UDP Port Usage

Digging Deeper

The SSL error typically indicates an underlying issue with the secure connection setup between the client device and FileWave servers. This can be caused by SSL certificate verification failures, misconfigured proxies, or stringent network security policies that incorrectly classify or block legitimate URLs required for FileWave operations. Adjusting content filtering policies or proxy settings often resolves these issues, but further investigation into SSL trust settings may be required for complex network environments.