

# Streamie



# Overview

Streamie is an app designed for Apple platforms (iOS, tvOS, macOS), made for video streaming, recording and sharing, with a design focus on correctness, reliability and scalability.

Streamie is used by schools, businesses and home users for security system video walls, remote video monitoring, secure video recording and many other use cases. Streamie is also used as a component in other products and services that require a reliable mechanism for IP camera streaming and recording, with particular emphasis on multi-camera time synchronization and recording correctness where no video frames are dropped.

Streamie integrates with 3rd party systems that extend its functionality in terms of recording storage systems, proprietary camera system access, realtime AI image processing and on-screen display augmentation, remote control APIs, live streaming, enterprise access controls, webhooks, granular permissions and more.

Streamie's core functionality is summarized in the following categories.



# Architecture

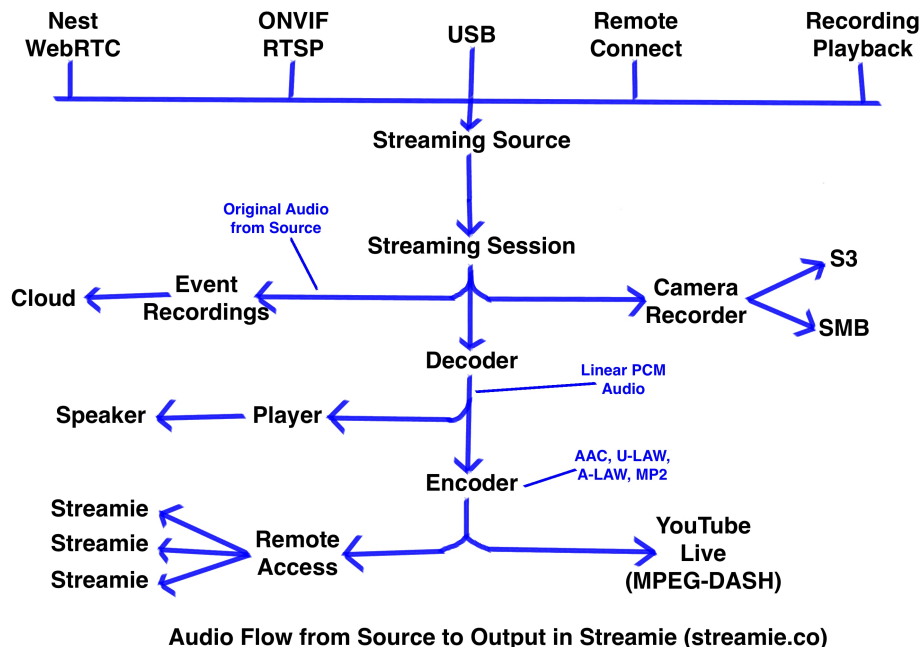
Streamie’s highest level streaming architecture is seen as three distinct layers: streaming, routing and displaying. Streamie features an architectural design that accommodates inputs from various sources, outputs to various destinations, and transcoding as required, all seamless to the inputs and outputs.

**Streaming** involves interaction with the network or another wire protocol (such as USB). The protocol-specific characteristics of the particular input source are stripped, and a stream is reduced to an internal representation of audio and video frames, along with corresponding metadata that can include information necessary to initialize or adjust audio and video decoders or event information such as motion detection.

**Routing** receives input from a single input source, along with the source characteristics. By default, nothing else happens. As output destinations are added to the router, audio and video input might be decoded or transcoded as appropriate. This work is performed optimally and on demand. It is performed seamlessly to the output destination, which doesn’t need to be concerned about the input characteristics. Additionally, multiple output destinations that each require identical output will not cause multiple transcode operations to occur, for instance.

**Displaying** is considered one of the three layers of the architecture, but functional changes muddied these waters a bit. In addition to displaying, output destinations include on-disk recording, relaying and off-device AI processing. In the case of audio, “displaying” might mean playing to attached speakers. Displaying is a critical feature for many users, and is also a key point of scalability concerns. Optimizing the display of video frames and reacting intelligently to display errors is essential.

The following diagram is specific to audio, but is generally applicable to the video pipeline as well.



# Protocols & Codecs

Supported streaming protocols include:

- RTSP (UDP, TCP)
- RTSPS (RTSP over SSL)
- WebRTC (Google Nest)
- HomeKit (Apple)
- MPEG-DASH (YouTube)
- RAW H264 (Rhombus)
- HLS (pre-alpha)

Input streaming codecs include:

## VIDEO

- H.264 (AVC1)
- H.265 (HEVC)

## AUDIO

- AAC
- AC3
- AMR
- G.711a
- G.711mu
- G.726
- MP3
- L16
- PCM

Industry standard cameras and NVRs:

Dahua, Lorex, Hikvision, Cisco Meraki, Axis, Rhombus, Ubiquiti, Google, Amcrest, Reolink, Dell, Aqara, AV Costar, Ctronics, D-Link, Eufy, Ezviz, Foscam, Sony, TP-Link, UniView, Vikylin, Wansview, Wyze, and more.



# Product Markets

## Security

The most common Streamie use case category, many potential customers in this market will discover Streamie in the App Store or via a web search. The use cases in this category range from monitoring a single crib cam, to school districts or nation-wide retail chains relying on alarm integration, 24/7 recording, police remote access — and everything in-between.

## Core Product

Another use case involves the development of a product that, at its core, requires integration with IP camera streaming. For one company, that involved using Streamie to precisely time-synchronize the recording of several cameras in each customer location so that an AI backend could post-process the recordings to perform an analysis, training an ML model.

With AI Classify integration support built right into Streamie, real-time processing is also possible for situations where that is appropriate. In one use case that was explored with a potential customer, Streamie would be deployed in large stadiums. Video walls, powered by Streamie, would show a set of cameras each augmented with overlaid data from real time AI processing of each video feed.

## Tech Licensing

Not all use cases are fit for general consumption. Sometimes the Functions feature is adequate to extend Streamie's functionality without having to change Streamie's core code. In some situations, that might not be possible.

Combined with carefully crafted contracts, Streamie's core tech can be licensed and re-deployed (including new backend API servers, domains, tunnels, cloud storage, etc.) for a customer, incorporating whatever changes the customer may need, without regard to how that might impact Streamie's core user base. The tech licensee would then (if needed) have access to the source code to continue supporting those changes.

## White Label

Streamie, re-branded. For certain customers we can help promote their brand by re-branding Streamie for them. A new app would be deployed to the App Store, complete with their app icon, their app name and minor app customizations.

# Customer Stories

## School

Campus security is essential at any school. A common limitation of a typical NVR-based camera system is the limited (or non-existent) ability to deploy multiple video walls. Additionally, one Streamie customer complained of “live viewing” latency of more than 15 seconds with their new, high end camera system. With Streamie deployed side-by-side with their existing camera system, they saw that latency decrease to a fraction of a second. Streamie also enables the deployment of as many video walls as a school requires.

The Streamie Modules feature allows text and image content to be displayed on screens, along with video streams, allowing school announcements and other information to be automatically displayed to students or visitors, including alerts or reminders of upcoming events.

Coordinating with local security and police departments, campus video streams and recordings can be remotely accessed in Realtime Crime Centers and by remote, authorized devices.

## Church

Church facilities often sit empty for significant portions of each week, making intelligent and effective video security paramount. Streamie can be deployed on the devices of those in charge of church security so they can receive immediate notifications related to intrusions or other events.

Churches often rely on live streaming. Streamie has built-in support for live streaming to YouTube that can be integrated with a church sound system and a PTZ camera. Local live streaming allows painless setup of overflow rooms. The Streamie Modules features allows the live stream to present announcements and other information.

## Retail (Nation-Wide)

Your IT staff are responsible for the reliable operation of a half dozen video walls in each of your 80+ locations across the country. With Streamie, you have one account set up per location. You can seamlessly switch between accounts to monitor, verify and remotely manage each device at each location. You don't require assistance from local employees. You have drastically reduced the number of on-site visits required of regional IT staff. Your costs are down. Your reliability is up. And the on-site employees could not be happier.

## Professional Office

Lawyers. Doctors. Real estate professionals. You had a pricey video security system installed and now you have live monitoring of your office from the convenience of a utility closet and remote access from the convenience of an app with a visual design stolen from the 1980s. Congratulations!

Your contract IT guy tells you he can get the office set up with additional video walls in a couple offices and at the front desk...today. That's Streamie.

Additionally, your office's productivity is tied to a specific workflow and your IT guy tells you that he can put those key details on each screen alongside the security cameras.

## Car Dealership

You have a security system in place, but now you need always-on monitoring of your car lots and show rooms in the offices of key managers and salesmen. In no time at all, your IT guy is able to configure Streamie for each office, bringing the cameras online.

Customers, cars and all of the details of the sales process are now easily monitored.

Tie Streamie into your sales management system, and present live sales stats in the employee break room to keep everyone motivated and on their game.

## Gym

Liability concerns is what keeps gym owners awake at night. Let Streamie help. Let your customers see your CCTV system as they walk in the door. The front desk staff can keep an eye on every corner of the gym without leaving their post. The gym manager knows what's going on without leaving his office. You can check in on things while away, review recordings remotely and receiving alerts when there's after-hours motion.

## Home

You've got three different cameras at home made by three different companies, and thus you have three different apps on your phone, none of which have all of the features you want. Streamie works with each of your cameras. You can watch the crib from your home office while your baby sleeps, see who just rang the doorbell at a glance with your kitchen iPad, get alerts when movement occurs around the house at night, have local 24/7 recordings to review later if something happened. You pick the cameras. Streamie brings the peace of mind.

## Estate

You've worked hard throughout your career and your estate shows your worth. You've got every angled covered in your video security system. With just the click of a button, you can pull up any of your cameras in any room with Streamie. You only display a few cameras at a time, and the Loop feature cycles through the offscreen cameras. Your security guards have an iPad loaded with Streamie with read-only mode enabled, allowing them to monitor, but not edit any parameters. Audible alerts announce to security the presence of people and vehicles at the perimeter.

Your main video wall uses the Focus feature to display your favorite cameras combined with the cameras with recent motion, cycling cameras onto the screen as the staff and visitors move around the property, allowing you to effortlessly know what's going on at a glance.

# Features

ONVIF	RTSP	Google Nest	HomeKit
Attached Cameras	USB Cameras	Ubiquiti	YouTube
NAS Recording	Cloud Recording	Camera Sharing	WebAPI
Remote Streaming	Tech Support	Home Automation	Permissions
Access Controls	Web Portal	Camera Layout	Modules
Functions	Integrations	Device Pairing	Event History
Members	Scalable Video Walls	Event Actions	Groups

## ONVIF

Streamie supports ONVIF for discovering cameras on the local network, initializing RTSP streams, receiving event notifications, controlling pan, tilt and zoom, controlling focus and other image parameters and logging license plates.

Many cameras and NVRs feature built-in support for ONVIF, yet there are also many compatibility issues between them. Streamie is optimized for working around these compatibility issues arriving at a place where cameras “just work” in Streamie.

Streamie’s built-in ONVIF support is completely custom, and developed with a large test bench of cameras and feedback from many Streamie users.

## RTSP

Streamie’s RTSP support started with the open source library Live555. Due to camera compatibility issues, Streamie shifted to using FFmpeg. Due to stability and scalability issues, Streamie eventually migrated to its own completely custom RTSP streaming core.

There are many cheap RTSP cameras on the market, and many run poor, old or buggy RTSP servers. Additionally, network connections are not perfectly reliable, particularly over wifi. Adding this all up, a custom RTSP streamer was necessary to meet user expectations that Streamie would be able to reliably stream their cameras for months or years without issue.

Issues arise though, which made the focus of the RTSP streaming core to carefully and reliably recover from unexpected errors. In lieu of a good solution, Streamie should always do the least obnoxious thing from the user’s perspective.

Much work has gone into improving compatibility with countless RTSP cameras and their unique behaviors, ultimately allowing Streamie to reliably support a wide mix of cameras simultaneously.

# Google Nest

Google Nest cameras are accessed via RTSP and (more recently) via WebRTC. Streamie supports both protocols for Google Nest camera streaming.

Due to the architecture of WebRTC, Streamie is able to record WebRTC video, but not audio. Streamie can also relay / share Google Nest cameras.

Due to the insanity which is Google's general outlook on what camera ownership actually means, the enduser setup process to be able to access your own cameras via Streamie is a pages-long ordeal. This feature is supported by Streamie, but does not live up to the ease-of-use expectation that one should have for a typical camera.

# HomeKit

Only marginally better than Google Nest is Apple's HomeKit. The setup process is completely painless, but the end user is completely forbidden from actually accessing streaming data from his own camera. HomeKit will display video, but it does so via a separate process which is inaccessible to the host app. Audio remains entirely unsupported in 3rd party apps. Overall, it is a blackbox disaster.

# Attached Cameras

Some host devices, such as an iMac, MacBook Pro, iPhone and iPad, having built-in (aka, "attached") cameras. Streamie supports these video sources, which provide uncompressed, unencoded raw video frames.

Internally, Streamie is able to encode these video frames for recording, remote streaming, etc. Streamie can also (obviously) display the attached camera video stream to the screen.

Audio, via built-in microphones, is also supported.

Audio and video data from attached sources is handled by the core router just like all other input sources.

# USB Cameras

Some host devices support USB cameras. Typically this might be in the form of a traditional webcam. Streamie supports audio and video streaming from USB cameras much like attached cameras.

# Ubiquiti UniFi Protect

Ubiquiti UniFi Protect is a video security product line of hardware and software with a lot of parallels to ONVIF. Streamie supports Protect in much the same way it supports ONVIF. Once Streamie is configured with authorized access to a Protect system, Streamie can discover cameras, establish RTSP connections and monitor for system and camera events.

Ubiquiti's RTSP support has grown over the years from H.264 to its own possibly custom H.265 encoder that is dynamically adaptive to network congestion. Interestingly, that has led to video stream corruption issues in situations where Streamie launches streams for many cameras simultaneously. Streamie has code in place to identify the exact corruption that occurs in this situation, and to seamlessly handle it with minimal user impact.

Ubiquiti is working on an official API. Until that is released, API support is based on 3rd party reverse engineering. It tends to be reliable, but new updates to the Protect software have been known to break backwards compatibility with 3rd party software.

## YouTube

YouTube features multiple live streaming ingest protocols, such as RTMP (Adobe Flash) and MPEG-DASH. I wrote an extensive technical document on the "adventure" which was establishing a reliable MPEG-DASH connection with YouTube relying on their minimal documentation and non-existent technical support. In fact, at this time I remain unaware of any 3rd party software other than Streamie that supports YouTube's MPEG-DASH ingest system.

YouTube integration in Streamie functions as an output destination to the stream router, receiving H.264 video and AAC audio, which are transcoded on-the-fly (if necessary), to comply with YouTube's live stream requirements. Whether one, both or neither stream has to be transcoded, timing information is correctly conveyed to keep audio-video synchronization correct all the way from the source input to the YouTube live stream.

With a single Apple TV and a display, a church, school or other organization can reliably host a YouTube live stream with minimal effort and minimal expense.

## NAS & Cloud Recording

Streamie can record your stream sources to a local NAS (via SMB), to any local or remote storage service that is S3 compatible, to Azure Blob Storage, or to the built-in Streamie Cloud service.

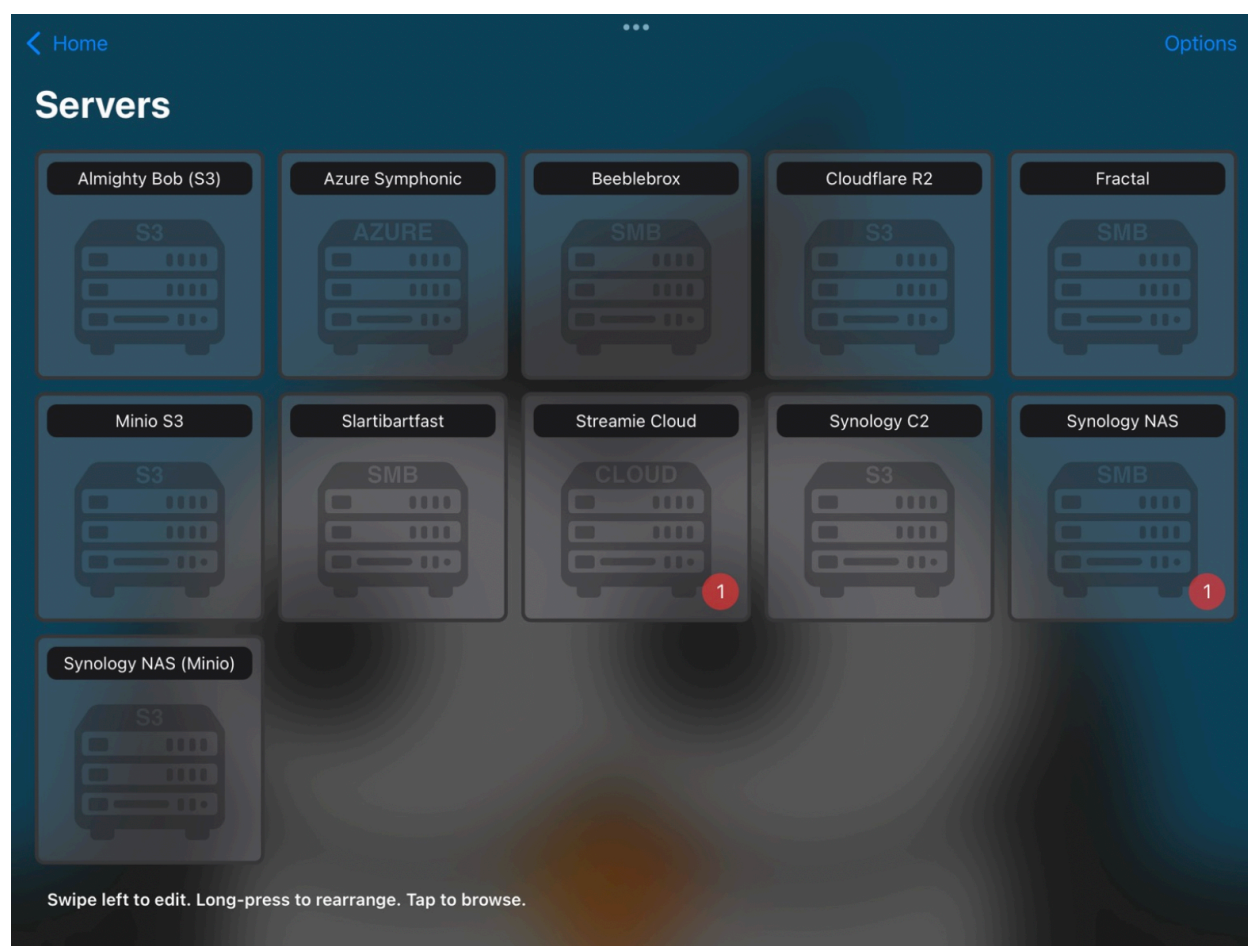
User-customizable parameters are available to tune Streamie's performance for the particular storage service including thread concurrency, max retries, request timeout and retry wait period. Each of these settings is critical to correctly support the nuances of various 3rd party storage services.

Recording files are created on the local device using a secure, proprietary format called Streamie Packet Format (SPF). This format is largely based on the internal representation of raw stream data and is thus optimal for recording at scale. The format of the SPF file is optimized both for writing and streaming, eliminating random writes, and minimizing random reads.

Each SPF file includes a table of contents and an appendix. The appendix includes low-res snapshots that were accumulated periodically during the duration of the recording, making it possible to quickly preview a recording file without downloading the entire recording. Each write to the file includes a complete Group of Pictures. Each recording file begins with a key frame and ends with the completion of a Group of Pictures. The table of contents includes byte offsets and the associated time code for the start of each GoP, making scrubbing within a recording file, particularly in low bandwidth situations, a pleasant user experience.

The recording functionality can be configured to output MOV or MP4 files in addition to SPF, although scalability is reduced.

For users interested in recording more so than live monitoring, Streamie features a recording-optimized mode. While running in this mode, camera streams are only updated on the screen periodically, significantly reducing the computational overhead. Ultimately, the network bandwidth available to the host device becomes the primary recording bottleneck.



## Camera Sharing

The Streamie backend, known as Penguin, enables the synchronization of account configuration details, manages all account information and imposes security and account restrictions. The Penguin API servers coordinate the sharing of cameras between accounts, enforce limits and (when all else fails) act as a relay server.

A common user request is to be able to share a camera. This could be a crib cam being viewed by grandparents or security cameras being monitored by a 3rd party. Each Streamie device has access to a default camera, "Streamie HQ", which is pointed out into a boring yard. A surprising number of people watch this camera.

Streamie's camera sharing feature makes it as simple as sending a URL to a friend. After you configure a camera, you can create a URL for that camera. Copy-paste it, and anyone else with Streamie can watch that camera.

Some use cases involve automatically enabling and disabling shared camera access on a schedule, so Streamie supports custom, per camera, sharing schedules for just that purpose.

For the camera sharing feature to function, you must have an "always on" Streamie device that has direct network access to the shared camera. Ideally, that device should be on a network with UPnP enabled, but that is not a requirement.

## Web API

The Web API feature is a server-side mechanism that can be used by scripts, other programs, cron jobs, etc., to securely route commands to your Streamie devices. For instance, you might want to grab and save a snapshot from a particular camera. You might have a motion sensor near your swimming pool that activates a program that launches Streamie on your office Apple TV and then instructs Streamie (via Web API) to stream the pool camera (this is a real use case).

The Web API provides an API key that you can use outside of Streamie to access and control Streamie.

## Remote Streaming

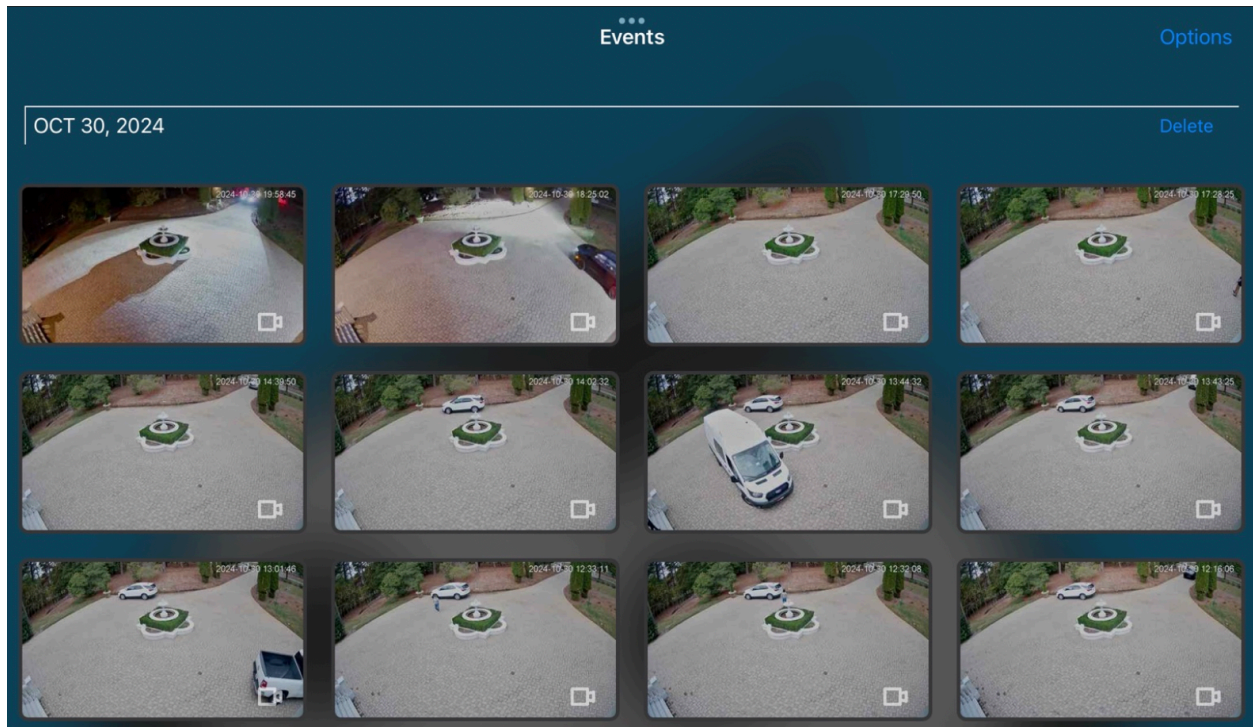
Remote streaming is a mechanism that attempts to as transparent as possible, that enables your remote devices to access your cameras without any special effort by the user. If cameras can be accessed directly, that will happen first. If cameras are not locally accessible, the Penguin API server guides the Streamie client with details of another Streamie device on your account that advertises having direct access to the desired camera(s). With that information, your remote Streamie device then starts attempting to find a way to connect with the Streamie device with local camera access. This all happens transparent to the user.

## Event History

ONVIF and UniFi Protect cameras support smart motion and other events. Streamie can receive these events and perform actions including capturing a snapshot, a brief recording and notifying you of the event.

Captured events are stored in the cloud and are immediately accessible to any of your devices in the dedicated Events tab.

Streamie "learns" your camera system's supported events by monitoring the cameras. Once an event type is seen for the first time, Streamie records that information and it becomes accessible within Streamie for the user to add customized event handlers.



## Device Pairing

Coordination of account settings such as 3rd party integrations, camera connection details, modules, file servers, recording settings and so forth are all stored securely and made accessible to Streamie via the Penguin API server.

To simplify the deployment and management of a Streamie installation, multiple devices can be “paired” together into a single account. When one device makes an account change, that change is immediately accessible to all of the other devices on the account.

More details on the benefits of this feature can be found in the **Device Management & Permissions** section in this document.

## Integrations

Integrations is a framework by which 3rd party services and other external features can be accessed by, and integrated with Streamie. These integrations include:

- Google Nest
- UniFi Protect
- YouTube Live Streaming
- S3 Storage
- Azure Blob Storage
- Web API
- Rhombus
- AI Classify

Descriptions of each of these integrations and the associated functionality can be found elsewhere in this document.

## Functions

Functions are a JavaScript-based feature. User-created Functions are executed directly on the device. Functions have direct access to internal data and features that a user might typically access manually. Common actions can be scripted with a Function.

Functions can be manually executed or associated with an event triggered by a camera or NVR.

One customer use case involved a camera with a proprietary API for triggering an alarm. When any camera in the facility triggered an event that Streamie received, subject to certain schedule and mode restrictions, Streamie could use that proprietary API to notify each camera with an attached alarm and strobe to activate. All of this was scripted in the Function itself.

In another use case, a Function could be triggered by an event, grab a snapshot from a series of cameras and push that snapshot to one or more configured file servers.

Functions are the mechanism by which truly custom functionality can be added to Streamie without requiring a new build and app distribution.

## Modes

Modes is a mechanism by which functionality can be manually or automatically enabled or disabled. Given a large number of cameras, each configured to capture and notify motion events, it would be ideal to disable that functionality during business hours.

Modes operate on schedules and can also be manually enabled or disabled. When configuring an action on a camera event, that action can be tied to one or more modes. The action will not execute unless at least one of its associated modes is active.

Modes are also tied to camera sharing, as mentioned elsewhere in this document. Businesses want to share cameras with their customers, but only during business hours. Modes accomplish this automatically, with the flexibility to manually override set schedules.

## Integrated Support

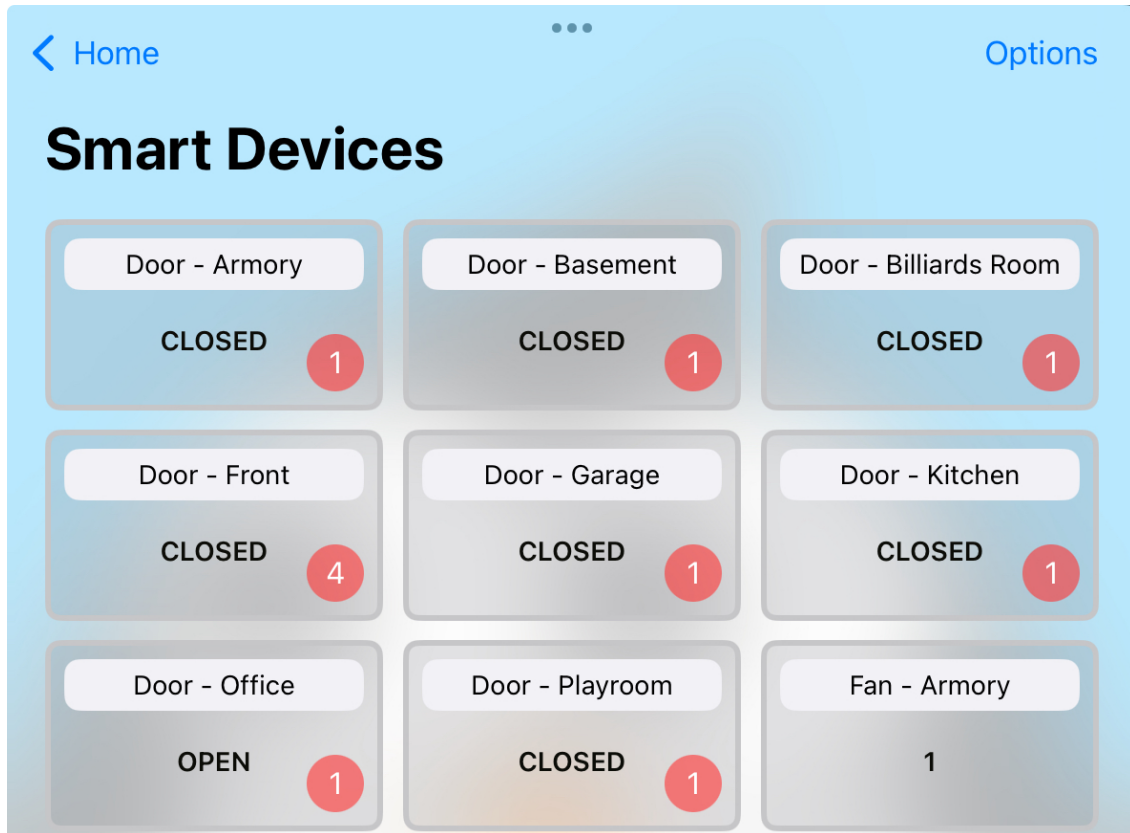
The best way to help customers is by making it convenient for them to find answers, or ask for assistance. The Help tab in Streamie does both. Articles, categorized by topic, and accessible at any time. If a user still has questions, a built-in support ticket system enables the user to reach Streamie Support right within the app.

## Home Automation

Streamie supports Hubitat for home automation integration using a custom Hubitat integration that directs Hubitat to send events to one or more of your Streamie devices.

With Hubitat integration enabled, Streamie immediately starts learning all about your home automation system as events occur (much as it learns about camera events as they occur).

Once an event is recognized, the user can add custom actions to that event, such as audible alerts, push notifications, camera snapshots or functions.



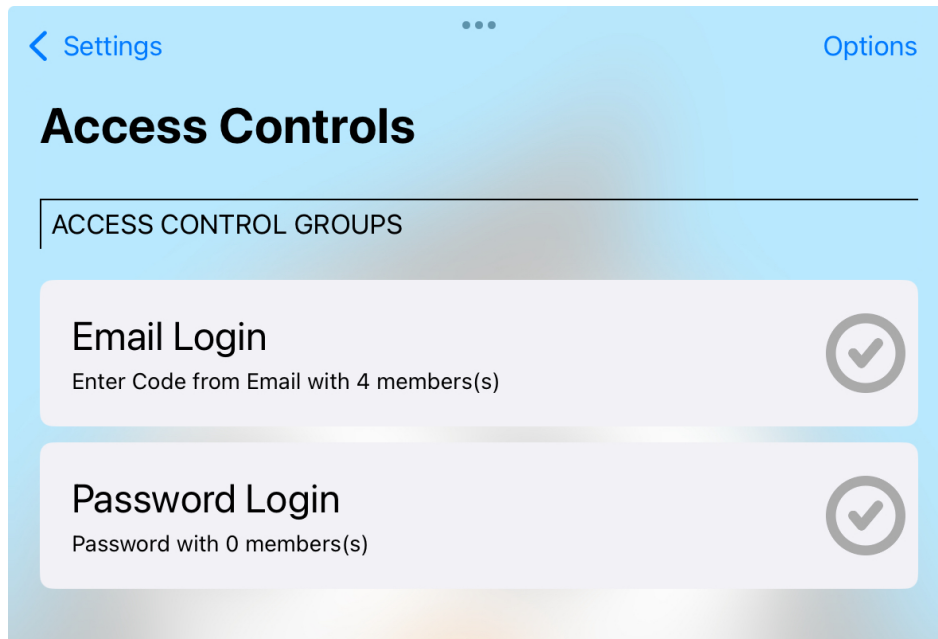
## Device Management & Permissions

Granular permissions in Streamie are key to letting people have access to devices on your Streamie account while controlling exactly which data and what functionality the user has access to.

In one case, security guards are issued an iPad running Streamie. They are able to view and stream certain exterior security cameras. They receive audible alerts when motion occurs. They cannot capture images. They cannot view or edit camera configurations. Their iPad is locked down and all write features are disabled.

Additionally, as an IT administrator, you may have many devices on many accounts that you need to manage. A traditional setup includes one account per physical location. From the convenience of your administrative Streamie device, you can switch to the account associated with a particular office, modify cameras, add or remove devices, view the screens of any of the Streamie devices and remotely control those same devices. This functionality reduces the number of on-site visits required and allows the administrator to see what Streamie is doing without relying on local employees to describe a situation or to handle changes.

# Access Controls



The Access Control system in Streamie adds an authentication mechanism on top of the app. An administrator device can create an access control group along with a set of authentication mechanisms, such as a security code via text message or email. This group is restricted to specific domains. A whitelist and blacklist enable easily adding or prohibiting certain users.

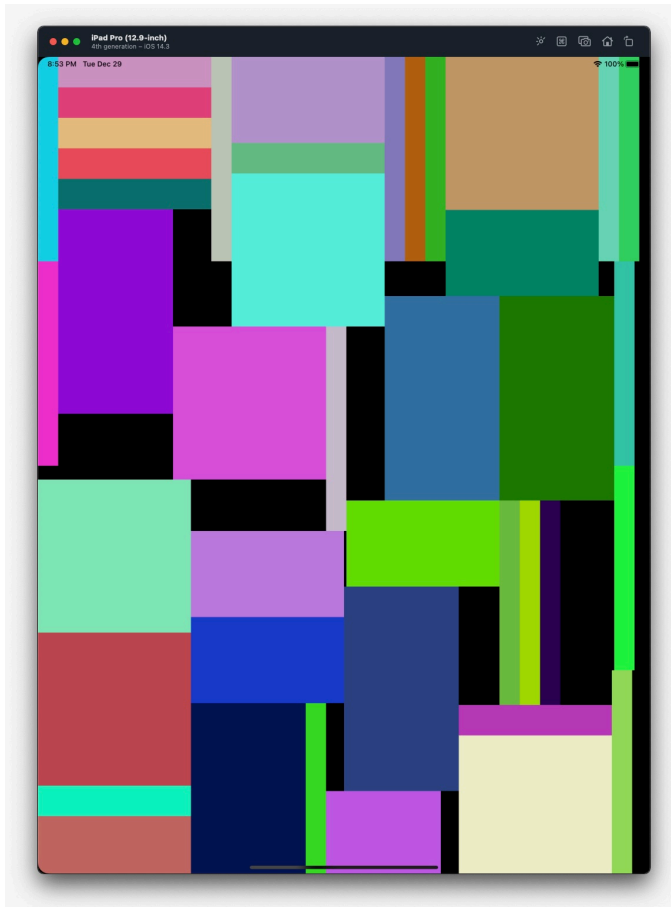
When a user wants to unlock a Streamie device, that user must first authenticate based on an available access control group.

## Web Portal

The Web Portal is an interface that is hosted by a Streamie device that gives a user web-based access to recordings and cameras. Instead of needing to run the app on a device, a user can conveniently access a limited amount of Streamie functionality from any supported web browser.

# Advanced Camera Layout

Staying true to our roots of thoroughly designing and testing core features, the advanced camera layout system was developed and tested using randomly generated rectangles to (hopefully) reflect an impossible-to-replicate-in-the-real-world scenario, proving that anything a customer can throw at Streamie, Streamie will support with ease.



## Modules

Modules are a mechanism for non-video on-screen content. Built-in modules include:

- Weather
- Clock
- Message
- Slideshow
- Custom

Modules are rendered from JSON and displayed on the screen much like a camera. Additionally, modules can be rendered as an overlay on an individual camera, or the screen as a whole. For instance, you can customize the weather to just be a temperature and overlay it on your outdoor camera. A convenience store could display “CCTV IN USE” along the bottom

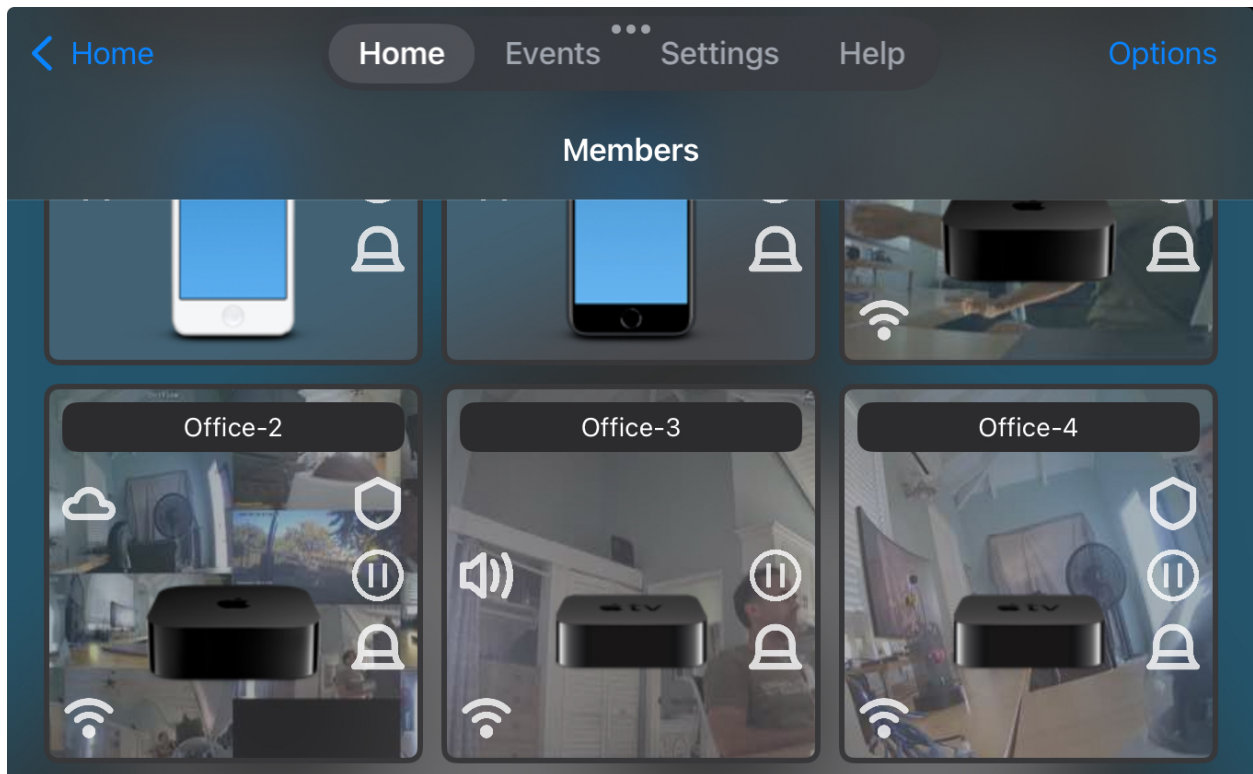
of a TV showing a grid of cameras. A CNC machine camera could have an overlay showing the machine's current status.

Custom modules are possible by providing a URL that returns supported JSON. A Streamie user might use Modules to display current sales stats on a video wall in a sales office. A network administrator might use Modules to display system information around a data center.

The content is entirely customizable by the end user. The rendering occurs on-device and the device itself makes the requests for external content.



# Members



A Streamie account can have one or more members, subject to subscription limitations. Each member can be remotely managed. Each member has a number of individual settings that characterize its performance and behavior.

**Debug Logging** — A Streamie user will typically enable this when instructed to when Streamie Support is diagnosing unexpected behavior. Streamie will log all console output which will be periodically pushed to the Penguin API server where they can be examined. Once enabled, debug logging will automatically disable after one week.

**Audible Alerts** — You may choose to use audible alerts for some motion events or other features. Sometimes you just want to make sure that Streamie does not interrupt something important. This option disables all spoken alerts.

**Proxy Server** — When enabled, this Streamie device can act as a proxy server for other Streamie devices to access cameras. This functionality is used in two scenarios: (1) remote camera access, (2) local proxy client access. Both of these scenarios are described in detail elsewhere in this document.

**Proxy Client** — The proxy client feature is also known as “scalable video walls”. It is not uncommon for a Streamie customer to want to display cameras in more locations than the camera can support in terms of concurrent streams. By establishing one or more Proxy Server devices, a Streamie user can limit other devices to being Proxy Clients. Those devices will no longer attempt to directly connect to cameras; they will

instead require the use of a Proxy Server device. A small amount of latency is added in this situation, but the number of video walls can scale significantly without putting additional load on a camera or camera system.

**Notifications** — When certain events occur, a Streamie user may enable push notifications to be alerted of the event. If the user needs a quick way to disable all of those events on a particular device, this option will accomplish that.

**Stop Offscreen Cameras** — This is a performance related tuning option which is enabled by default. When enabled, all connections and features associated with a camera are stopped if that camera is not actively being streamed, visible on the screen.

**Streaming Rotation Lock** — A rotation lock is applied when this feature is enabled, that prevents Streamie from rotating (on devices that support rotation) once streaming has begun.

**Device Offline Alert** — Some Streamie devices can be critical to your security operations and you may want to be alerted if one of them goes offline for any reason. This feature, when enabled for a particular device, will alert other devices when it goes offline.

**Background Keep-Alive** — When enabled Streamie will attempt to keep itself alive while running in the background.

**Web Portal** — The web portal is a feature that enables web browser access to some Streamie functionality such as reviewing NAS recordings or live streaming cameras.

**Autostream** — You can choose a camera, module or group to be streamed automatically when Streamie starts up.

# Scalable Video Walls

This feature is not actually specific to video walls, but that is the common use case. Many popular security cameras support no more than 3 - 5 concurrent streams. It is sometimes the case that a Streamie user wants more video walls (each requiring a concurrent camera stream) than what the cameras support.

Streamie can work around concurrent stream limitations in a simple, intuitive way. Choose one or more Streamie devices to function as Proxy Servers. Enable this feature on each device in the Settings tab. These devices will directly connect to cameras. For all other Streamie devices, enable the Proxy Client feature in the Settings tab. These devices will refuse to directly connect to cameras, and will instead require a Proxy Server device for camera access. The number of concurrent, direct camera streams will be limited to the number of Proxy Server devices that are running. Meanwhile, all local Proxy Client devices will “piggy-back” off the Proxy Server devices, keeping the cameras happy while also not incurring any significant latency.

This feature is limited to local network connections.

# Event Actions

Event Actions are a mechanism by which Streamie can automatically perform certain tasks. Event Actions are triggered by cameras (ONVIF or Ubiquiti) and home automation hubs (Hubitat).

Streamie “learns” the capabilities of a camera or home automation system by monitoring for events and recording each type of event. Once an event type has been recorded, a Streamie user can assign a set of custom Event Actions to the event.

Event Actions fall into the following categories:

**Speak Text** — The devices on which this action runs will audibly alert you with the text you provided using the built-in text-to-speech capability of the device.

**Play Sound** — Choose from one of several built-in sound options to notify you of a particular event.

**Log Event** — The devices on which this action runs will immediately grab a snapshot from the camera and will trigger a 30 second log recording that begins ~7 seconds prior to the start of the event. The metadata, snapshot and recording will be pushed to the API server and made available to Streamie users in the Events tab. A push notification with the snapshot is sent to your Streamie devices as well.

**Save Snapshot** — Grab a snapshot of the camera and save it to an iCloud photo album.

**Save Recording** — Grab a 30 second recording that begins ~7 seconds prior to the start of the event. Save the recording to the iCloud photo album.

**Run Function** — Look at the Functions section of this document for more details. This feature allows an event to trigger a function. An example use case for this is a camera

event from a license plate reader camera. The event itself contains license plate information. Combined with a logging Function, the license plate information can be stored in a web service.

**Focus Camera** — A powerful feature for passively monitoring more cameras than can reasonably fit on one screen. Briefly, an off screen camera will be moved on screen when a focus action is run on that camera. See the Groups section elsewhere in this document.

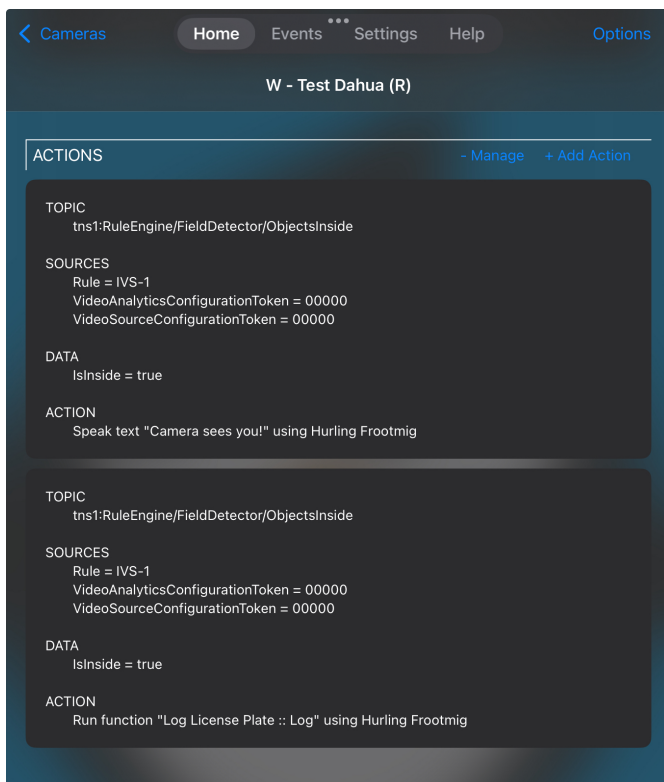
**Notify Me** — Send a push notification to your Streamie devices.

Each Event Action can be further customized in the following ways:

**Modes** — Modes are a simple mechanism for manually or automatically (based on a schedule) enabling or disabling functionality, such as event actions.

**Members** — Define exactly which Streamie devices will execute an event action.

**Timeout** — Define the timeout period in which duplicate events will be dropped. This prevents insanity when events occur rapidly.



# Groups

The Groups feature allows you to define a set of Cameras and Modules, along with a specific scale, independent item sizing, various options, layout settings and behaviors. The Groups functionality in Streamie is where real video security decisions are made. Let's dig in.

**Group Scale** — Each Group has a primary scroll axis which is either horizontal or vertical. Perpendicular to that scroll axis, the Group Scale defines how many equal divisions of the screen size, the layout is comprised of.

**Item Scale** — Each item (camera or module) included in the group has its own scale which defines how many units of the Group Scale that item will consume. A group might have several larger, primary cameras, and many smaller secondary cameras.

**Group Layout** — In addition to the horizontal and vertical layout mentioned previously, a group can also have an automatic layout. When automatic layout is enabled, a group does not scroll, and instead adjusts the Group Scale adequately to fit all items on the screen, and it adjusts the Item Scale to maximally fill the screen.

**Group Overlay** — Similar to how each camera can have an Overlay associated with it, a Group can have an overlay as well. A Group overlay can extend across the entire screen. A common use case is “CCTV Monitoring in Progress” across a customer-facing video wall.

**Group Function** — A Group can function in two different modes. The normal mode live streams the cameras. The record mode only periodically updates the screen (perhaps just once a minute), optimizing its behavior for scalable recording. In this mode, very little video decoding would take place, allowing Streamie to record many more cameras than what it could otherwise handle.

**Group Full Screen** — If you configure Streamie with many cameras, you may have to choose low resolution streams to work within the GPU limits of the device. When you full-screen a low-res stream, you will see compression artifacts. With this option set to High-Res, Streamie will look for a high resolution stream and automatically start that stream when switching from group view to full screen view.

The following options relate to each item in a group, primarily camera items. These features allow you to customize the behavior of the group when streaming cameras that support events. These settings apply specifically to the \*location\* where the camera is positioned in the group, and not the camera itself.

**Item Normal Mode** — As you might expect, an item location in a group with a normal mode setting will not do anything special on its own.

**Item Looper Mode** — With the Looper setting applied to an item location, off screen cameras will be cycled through that looper location allowing the user to regularly see more cameras than will fit on the screen.

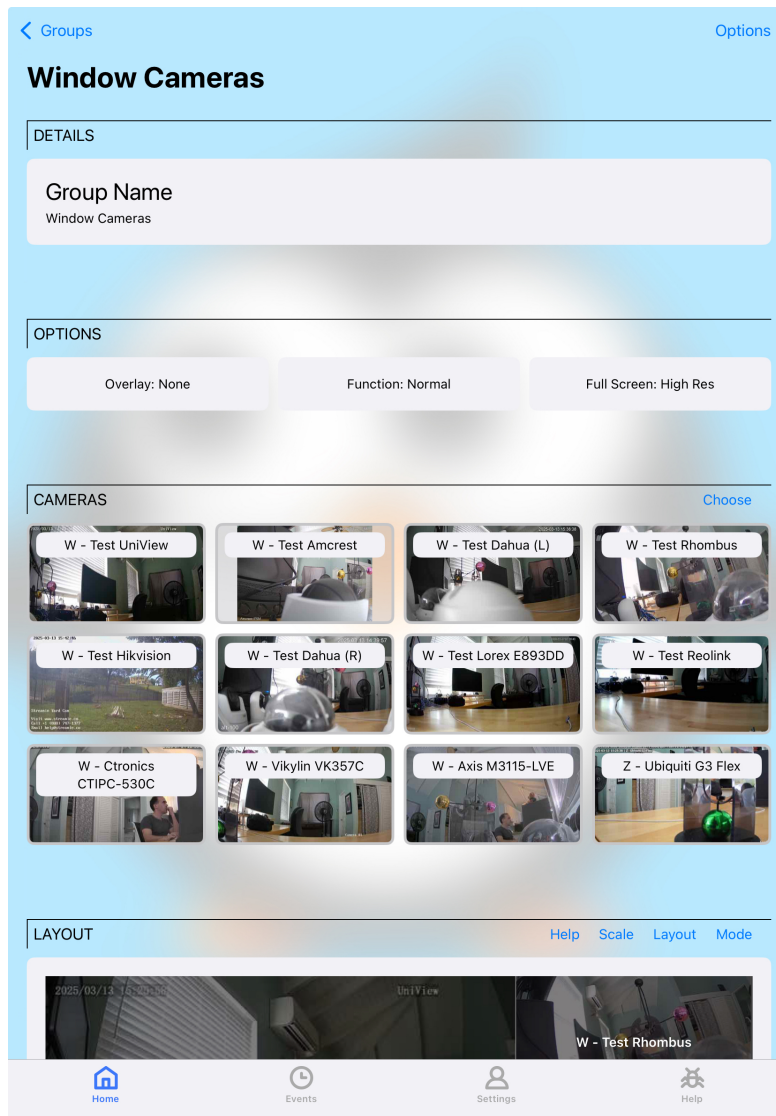
**Item Focus Mode** — The Focus mode is what really sets Streamie apart. Each camera that supports events can have a user-added Focus event associated with some of the

camera events, for instance a motion event. When the focus event occurs for an off screen camera, that camera will be moved on screen to one of the Focus item locations. After a timeout period where no further events occur, the focus item will be returned to its off screen location.

**Item Locked Mode** — When an item position is locked, the camera or module configured for that location will never change due to Looper or Focus events.

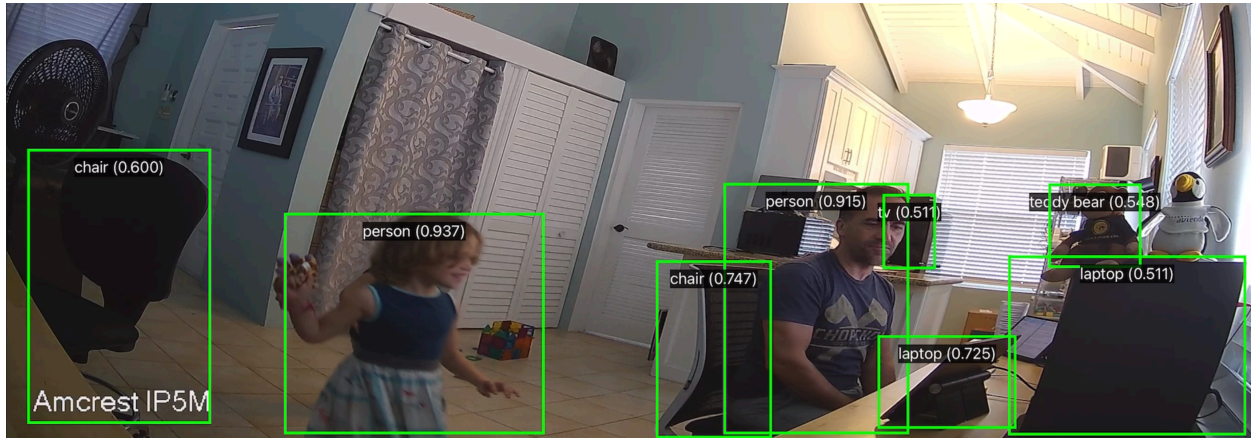
Summarizing the Item Mode feature, a user could set up a Group with a 3x3 grid of items, including many more than nine cameras in the group. Several cells could be fixed, requiring those cameras to remain in position at all times. Another position might loop, periodically showing you off screen cameras. Finally, several positions could be used for focus events, allowing you to immediately see cameras (including those positioned off screen) that detect motion.

These advanced features allow a single screen showing relatively few cameras to be effective at monitoring a large number of cameras with little or no intervention by a person.

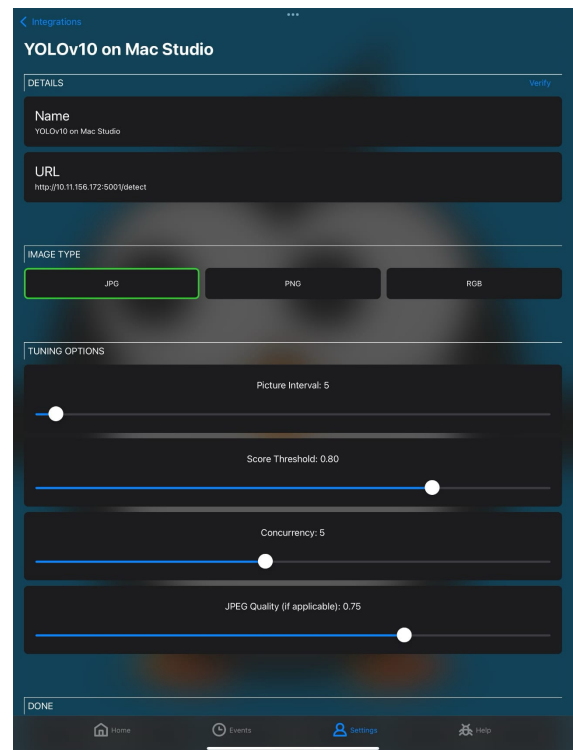
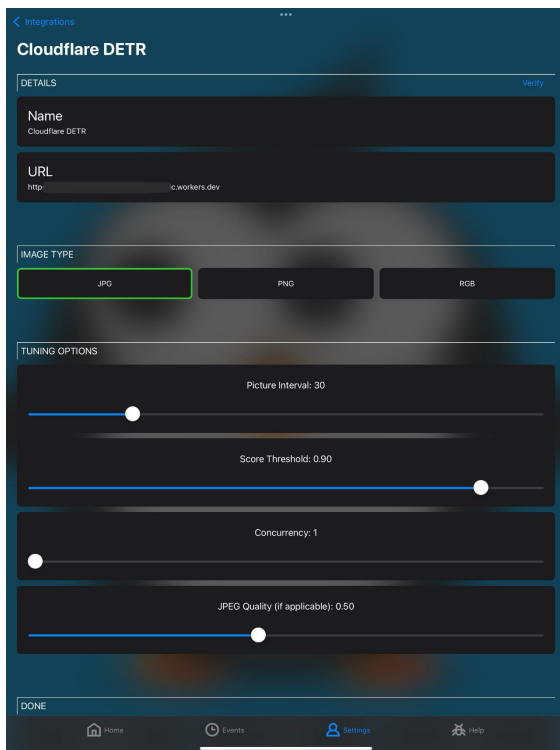


# AI Classify

The AI Classify feature is a mechanism by which Streamie can be directed to send an active video stream to an external service for real time image recognition (or other processing) and (optionally) to draw metadata on top of the video stream. The simplest use of this is drawing a box around people moving through a scene, or around objects that are recognized.



Extensive instructions exist for enabling this integration ([https://help.streamieapp.com/a/ha\\_IQrm6a3HQA](https://help.streamieapp.com/a/ha_IQrm6a3HQA)) which can work with services such as Cloudflare Workers AI (facebook/resnet-detr-50) or with YOLOv10 running directly on your own hardware.



# Learn More

You can learn more about Streamie by visiting the Streamie Help site at <https://help.streamieapp.com>.

All changes to Streamie are published online and accessible directly in the app. You can find the full history of release notes here: <https://help.streamieapp.com/t/Release%20Notes>

The latest version of this document is available here: <https://help.streamieapp.com/media/Streamie-Overview.pdf>

For all other questions, contact us directly: [help@streamie.co](mailto:help@streamie.co) or +1 (888) 747-1377.



# Summary

Streamie is the definitive solution for modern video security, surveillance, and streaming needs, offering an unmatched blend of reliability, scalability, and technological depth. Designed for Apple platforms, Streamie powers everything from single-camera home monitoring to enterprise-grade multi-location deployments, ensuring precise synchronization, zero frame loss, and seamless remote access.

For **sales professionals**, Streamie's market reach spans schools, businesses, law enforcement, and security firms, each needing a robust, scalable video monitoring solution. Its ability to integrate with AI-based analytics, cloud and NAS storage, and third-party automation platforms makes it an easy sell to organizations that demand top-tier security and automation.

For **investors**, Streamie represents a high-value opportunity in a rapidly growing sector. The video surveillance market is projected to exceed \$100 billion by 2030, and Streamie is positioned to capture significant market share with its advanced features, extensive device compatibility, and proprietary technology stack. With recurring revenue models from cloud storage, enterprise licensing, and white-label partnerships, Streamie delivers sustainable and scalable revenue growth.

For **technology licensors**, Streamie's flexible architecture supports deep customization, allowing companies to white-label or integrate Streamie's core streaming and recording technology into their own products. Whether it's rebranding for a dedicated security service, embedding proprietary AI analytics, or deploying custom API integrations, Streamie's licensing model ensures partners can differentiate their offerings while leveraging a rock-solid, battle-tested video streaming backbone.

With proprietary RTSP technology and ONVIF compatibility custom built for reliability in real-world circumstances, as well as cloud-native architecture, and API extensibility, Streamie isn't just another video security app—it's the platform that smart businesses choose to build on. If you're in the business of selling security, investing in high-growth technology, or licensing best-in-class video streaming software, you should be working with Streamie.

# Curtis Jones

It all started with a passion for programming, sparked by a HyperCard book and a Mac Plus. That passion turned into Streamie, initially designed to stream home security cameras, evolving from a basic script to a robust streaming platform. This journey highlights the importance of innovation, adaptation and growth in technology.

Our founder's background includes pivotal roles at leading organizations such as Apple, The Wall Street Journal, Bloomberg, and Chick-fil-A. This experience spans the development of macOS applications, mobile apps, data routing systems, and wireless technologies. This diverse expertise allows us to tackle complex technical challenges and deliver high-quality, dependable solutions.

We've had the privilege of working alongside major camera manufacturers, law enforcement agencies, educational institutions, and enterprises. These partnerships have not only enhanced Streamie's capabilities but have also provided us with valuable insights into the unique challenges you face daily.

At Streamie Security, Inc., our mission is to bridge the gap between cutting-edge technology and practical application. We continuously innovate to provide solutions that enhance operational efficiency and strengthen security measures for professionals in the IT, engineering, and security sectors.

