

Use Case: View Live Feed

Essential Use Case:

Participants: Home-owner

Pre-conditions: Home-owner has devices set up.

Typical Course of Events:

Actor Intention

1. Access application
3. Choose option to view feed
5. Select device
7. View feed
8. Exit feed

System Responsibility

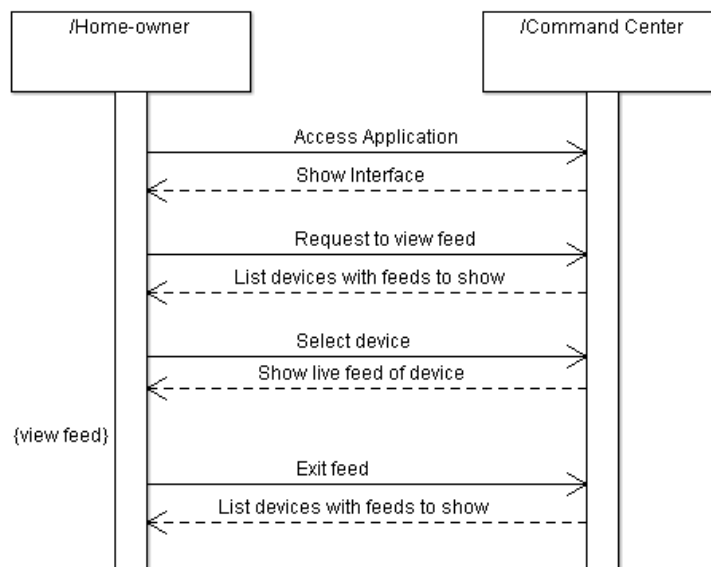
2. Show interface
4. List devices with feeds to show
6. Show live feed of device
9. List devices with feeds to show

Scenarios:

1. Home-owner wants to view the camera feed. The system shows the live view from the camera.
2. Home-owner wants to check what levels various sensors are reading at. The system shows the current (and expected) values for any the home-owner wishes to see.
3. The home-owner has no devices active or connected. The system displays a message that there are no live feeds available.

Post-condition: Home-owner has viewed feed (if available) and exited the feed.

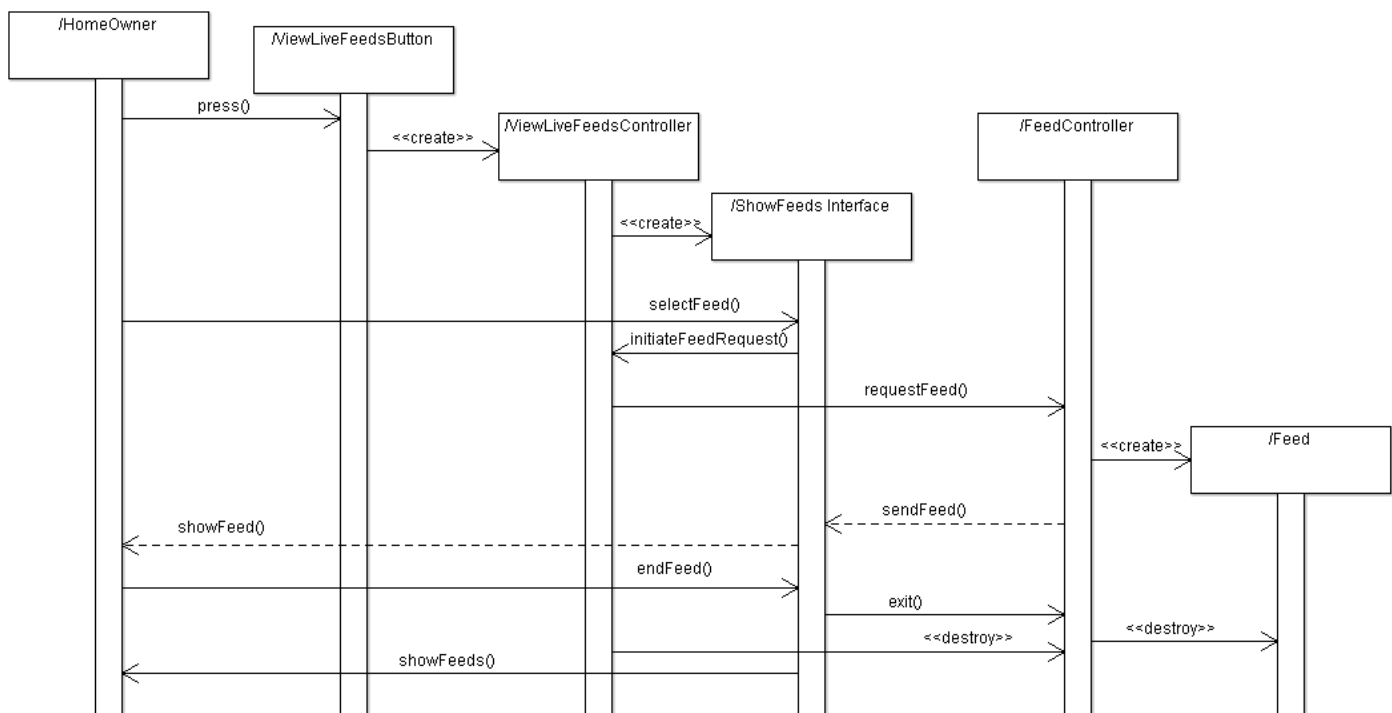
High Level System Sequence Diagram:



Concrete Use Case:

<i>Use case name</i>	ViewLiveFeed
<i>Entry condition</i>	1. The HomeOwner accesses the CommandCenter and presses the “View Live Feeds” button.
<i>Flow of events</i>	2. The CommandCenter displays an interface that lists the available devices to view the live feeds for. There is also an option to just view all current readings. 3. The HomeOwner selects the devices that they would like to view the feed for by pressing its respective button. The device’s live results are then accessed by the CommandCenter. 4. The CommandCenter displays an interface that shows the live feed for the device. If the device is a camera, then the camera feed is shown. If it is a sensor monitoring some levels, it shows the current reading, expected reading, and a graph showing the level variation over the last few hours. If the user chooses the option to view all current readings, a similar display to this is shown that includes the data for all of the monitoring sensors.
<i>Exit condition</i>	5. The HomeOwner exits the live feed.

Detailed System Sequence Diagram:



Use Case: Contact Emergency Services

Essential Use Case:

Participants: Home-owner, Emergency Services

Pre-conditions: System sends a notification to the home-owner about a potential reason to contact emergency services.

Typical Course of Events:

Actor Intention

3. Review the notification.
4. Tell system to contact emergency services.

System Responsibility

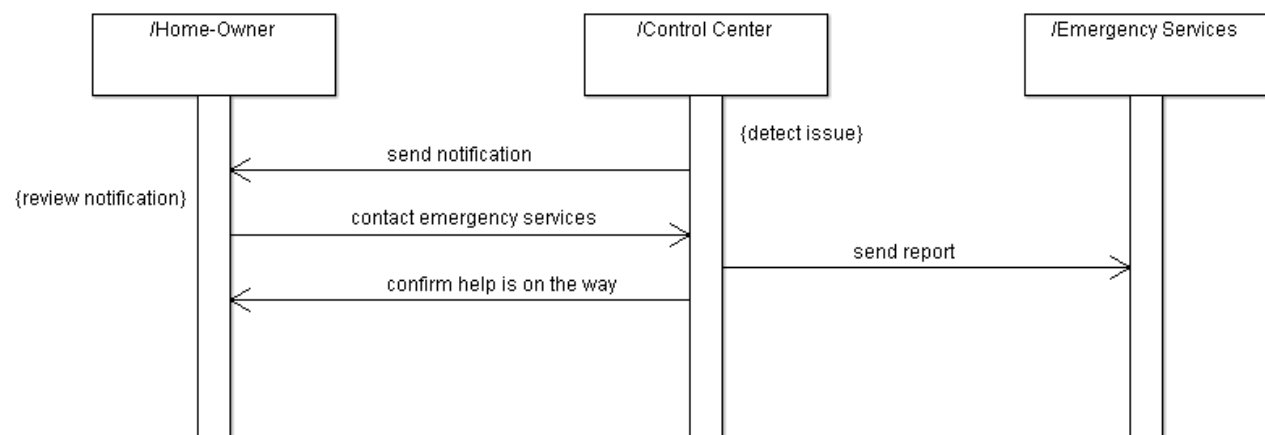
1. A sensor detects an issue that requires attention.
2. Notify the home-owner of the problem and recommend action.
5. Contact emergency services and send a report to assist them.
6. Tell home-owner help is on the way.

Scenarios:

1. System detects an intrusion. Home-owner notified, home-owner validates that it is not a mistake. Emergency services called.
2. System detects dangerous levels. Warns home-owner, and contacts emergency services.
3. System detects an intrusion. It is not an actual problem, home-owner does not contact emergency services.

Post-condition: Emergency services have been called.

High Level System Sequence Diagram:



Concrete Use Case:

<i>Use case name</i>	ContactEmergencyServices
<i>Entry condition</i>	1. The CommandCenter identifies an issue that may require emergency services to be called.
<i>Flow of events</i>	2. The CommandCenter notifies the HomeOwner of the issues that it identified. 3. The HomeOwner reviews the issue and decides whether to call for help. Typically, the HomeOwner will review the issue and decide that emergency services should be called. 4. The CommandCenter contacts EmergencyService and sends an EmergencyReport that lists the issue and provides as much assistance to the EmergencyService as needed. The EmergencyService then confirms that help is being sent to CommandCenter. 5. CommandCenter notifies HomeOwner that help is on the way.
<i>Exit condition</i>	6. EmergencyService is on the way.

Detailed System Sequence Diagram:

