API endpoints and their functions for CampusWatch

By Diya Neupane

The core API endpoints required and their functions for the firmware update distribution system are given below:

- **Cameras('/cameras')**

1. GET '/cameras': Endpoint to get all the cameras connected to the system. This endpoint can be used to view all the surveillance footage collectively or to get some data analytics about the cameras.
2. GET '/cameras/{id}': Endpoint to get all the metadata about a specific camera. This endpoint can be used to carry out some actions on a specific camera.
3. POST '/cameras': Endpoint to register/add a new camera to the system. Used when new cameras are added around the Campus.
4. PUT '/cameras/{id}': Endpoint to update the data about any specific camera. This endpoint can be used when camera details, such as location, name need to be updated.
5. DELETE '/cameras/{id}': Endpoint to delete information regarding a camera. This endpoint can be used when a camera is removed from the system.

- **Firmware('/firmware')**

1. GET '/firmware': Endpoint to get information about all the available firmwares, including their version numbers and latest updates.
2. GET '/firmware/{id}': Endpoint that fetches all the information about a specific firmware version.
3. POST '/firmware': Endpoint that can be used to create a new version of firmware.
4. POST '/firmware/update': Endpoint to update the new version of firmware in the system.

5. GET '/firmware/status':Endpoint to get the update status of a new firmware being updated to the system

6. DELETE '/firmware/{id}': Endpoint to remove a certain version of firmware from the system.

● **Incident report ('/incident')**

1. GET '/incident': Endpoint to fetch all the incidents related to firmware updates and failures.

2. GET '/incident/{id}': Endpoint to retrieve details related to a particular incident.

3. POST '/incident': Endpoint to report /add a new incident automatically(from backend) or manually(by admin).

4. PUT '/incident/{id}': Endpoint to update the status or notes of a particular incident.

5. DELETE '/incident/{id}': Endpoint to remove the data about a particular incident.

● **Alerts('/alerts')**

1. GET '/alerts': Endpoint to get all the alerts in system regarding firmware update success or failures.

2. GET '/alerts/{id}': Endpoint to get the information (firmware update success or failure referenced) of a particular alert.

3. POST '/alerts': Endpoint to create and add alerts.

4. DELETE '/alerts/{id}': Endpoint to remove the information of a particular alert.