

Dronology Use Cases

Revision History

Date	Author	Description of change
5/16/2017	Jane Huang	Create Use Case

Use Case: Detect Physical Drones

ID: UC-04

Description

Detects physical UAVs that are deployed on the ground prior to flight.

Primary Actor

UAV Flight Operator

Supporting Actors

IT/Hardware support specialist
UAV handlers

Stakeholders and Interests

Fire department engaged in river rescue
FAA concerned with flight regulations
General public

Pre-Conditions

All UAVs to be detected have previously been registered.
Correct telemetry hardware to communicate with UAVs on the ground is installed and ready-for-use.

Post Conditions

All UAVs which are detected and have passed arming checks are activated in Dronology.
The status of all UAVs that were detected is depicted (i.e. activated, failed to pass arming test).

Success end condition

At least X% of UAVs that were placed on the ground are activated in Dronology and ready to fly.
All activated UAVs are visually depicted on the displayed map.
The user is informed of the number and percentage of UAVs which were not activated.

Failure end condition:

Multiple UAVs share a single telemetry signal.
More than X% of the UAVs deployed on the ground are not activated.

Commented [J1]: I think we should tolerate some failure as it is inevitable.

Dronology Use Cases

Trigger

The UAV technicians place UAVs at their launching positions and arm them for flight.
The UAV Flight operator launches the Detect and Activate UAV UI.

Main Success Scenario

1. The UAV Technician places UAVs on the ground and activates them.
2. The UAV Flight Operator requests Dronology to detect UAVs.
3. Dronology retrieves a list of registered **drones**.
4. Dronology attempts to ping each of the registered drones.
Repeat steps 5-8 for each registered UAV which responds to the ping
 5. Dronology pings the UAV to request its type.
 6. Dronology performs an initial health status check on the UAV.
 7. Dronology retrieves inventory information from the UAV registry.
 8. Dronology retrieves meta-data corresponding to the type of the UAV.
9. Dronology returns a list showing the number of UAVs detected and the coordinates, meta-data, and health of each detected UAV.
10. The UI establishes a perspective for the map that includes the positions of all detected UAVs and the default scale of the map.
11. The UI displays a map of the selected area.
12. The UI displays the detected UAVs and their associated data on the map.

Exceptions

- 8a. In step 8, One or more of the UAV's actual types is not found in the UAV **metadata**.
 - 9a.1 Dronology sends a list of problem cases to the UI.
 - 9a.2 Dronology deactivates all UAVs of types without stored meta-data.
 - 9a.3 The UI reports the problem to the operator and shows the UAVs as deactivated.
 - 9a.4 The UI offers the option to **edit meta-data**.
- 5a. In step 5, Dronology fails to make contact with any **UAVs**.

Future Variations

Frequency:

Whenever an instance of Dronology is run with physical UAVs.

Assumptions

Physical hardware needed to establish communication with UAVs is available and setup.

Non-Functional Requirements

Performance

Commented [J2]: We have to think about whether this is actually necessary. We'll have to do some experiments to figure out what we really need to know in advance. Another option is just to cycle through all available telemetry hardware on all known USB/COM ports and see what we find.

Also – I don't know how to ping the drone for its equipment – but we can assume there is a way.

Commented [JC3]: Could we pull metadata from the drone? If so, we could just ask the operator if s/he want's to add that metadata to the system...

Commented [J4R3]: No we can't. Metadata includes manufacturers specs which aren't available on the drone.

Commented [J5]: Will revisit after glossaries are finalized.

Commented [J6]: We need to investigate reasons for this failure and fault tolerance solutions.

Commented [J7]: We need to brain-storm NFRs (I'm not focusing on these yet).

Dronology Use Cases

Issues

To do

Run some simple tests to see exactly what information we can get from each UAV about its serial number etc. For each piece of equipment we put on the UAV we must know how to figure out if that equipment is on board.