## Stakeholders

Development team, the company and the end-users.

Open source community, third party service,

## Three functional requirements

The system must be able to gather and save video data from a camera.

The system must be able to evaluate the input data against data of the same type from the local repository.

The system must be able to report threats.

## Three desired quality attributes

It should be able to install new data collection sources within 3 minutes.

Data should never be lost between collection and the database.

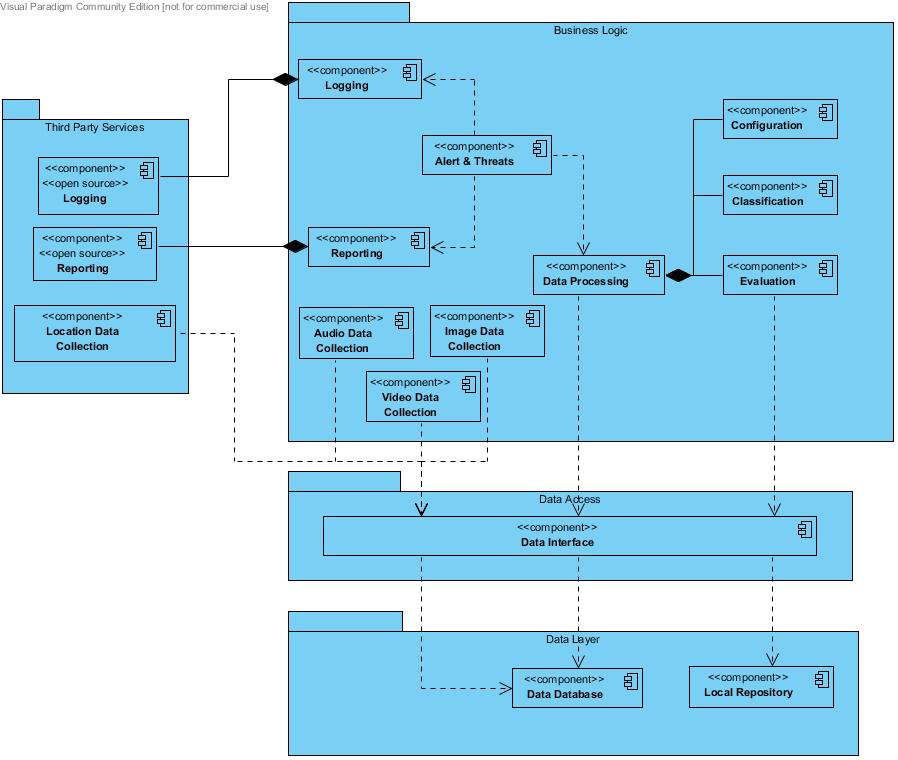
The system should return to functional state in at most 5 minutes if it goes down.

**Architectural drivers (goal based on the qualities and requirements)**

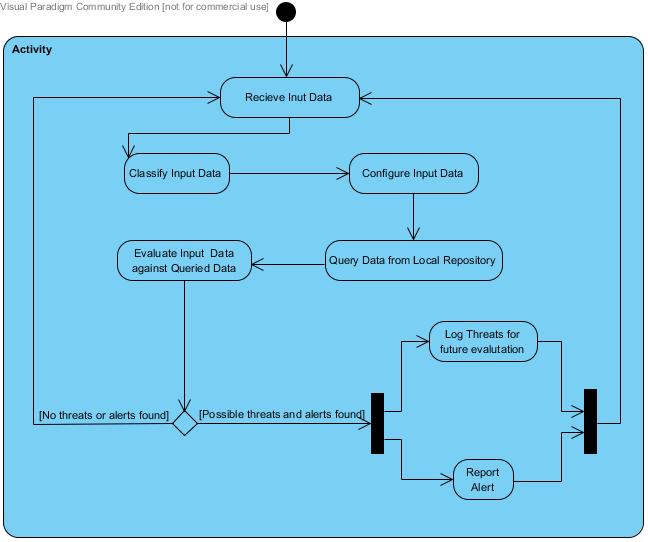
Maintainability is vital to the system since a security system should be easy to set up, change, and service. Within maintainability we also see reusability and scalability as two important attributes, since the ways of compromising the security might change, the system should also be able to change with it, and to be able to do that it’s important that it’s reusable. One easy way to do that is by creating the system in a scalable way.

Security is also one of the main attributes. Since the system is a security system it should be secure in more than one way. Firstly, the system should fulfill its purpose. Secondly it must be secure from intrusions and sabotage to the system itself, since if the system is altered the integrity of the entire system could be compromised.

The system needs to be reliable to function as a security system. If left alone, the user needs to be able to trust the system not to fail doing its tasks or crash.

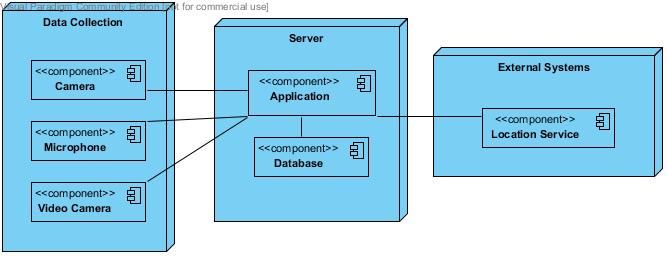


# Process view:

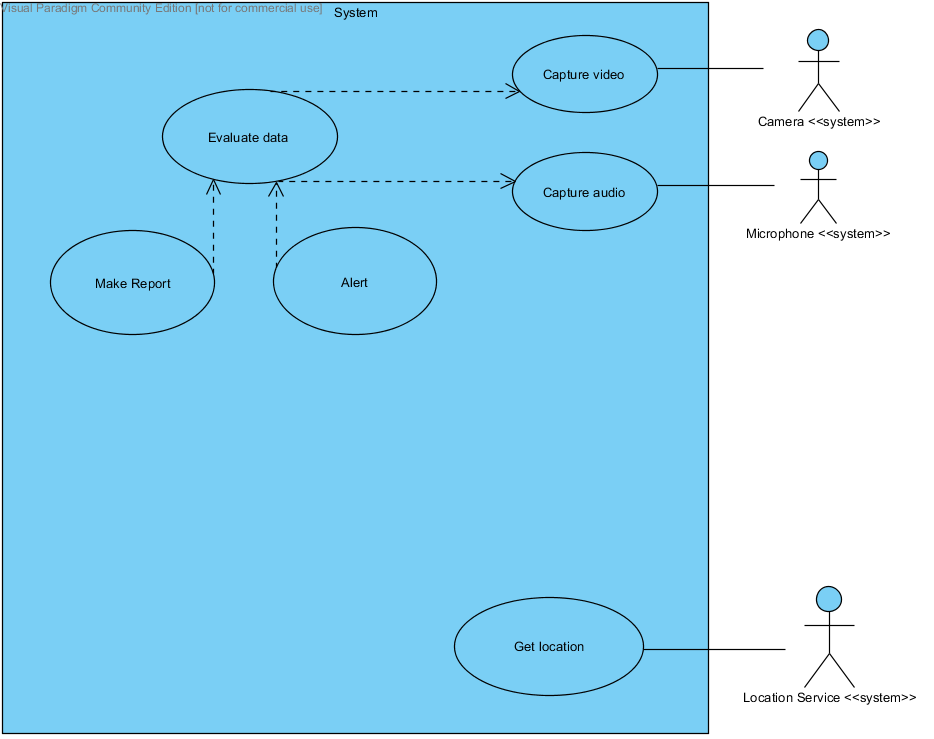


The process view was useful for getting the same idea of what the system should do and in what order. It makes it much clearer what should happen and where the input comes from or output goes to from the various parts.

# Physical view:

The physical view was useful for deciding where the various parts go. And is useful for quickly seeing what was decided. For instance how the database is on the same machine as the application which collects the various data.

## Scenario View



Basic use-case of the monitoring system.

1. The use-case begins when the system is turned on.
2. The system collects data in the form of audio, video, images and location from various sources.
3. The data is processed and evaluated against input data from a local repository to find possible threats and alerts.
4. If any threats or alert are found, report and log them for future evaluation; continue from 2.

Alternate Flow:

1. At 4: If there are no threats or alerts; continue from 2.