KU LEUVEN

DEPARTMENT OF COMPUTER SCIENCE

Shared Internet-of-Things Infrastructure Platform (SIoTIP)

Part 1A

Requirements Analysis

Geens–Jochems–Salim

Academic Year 2017–2018

H09B5B: Software Architectuur
H07Z9B: Software Architecture

Tomas Geens (r0597792)
Stijn Jochems (r0587702)
Afraz Salim (r0439731)

1. Utility tree of ASRs

	Quality Attribute	**Attribute Refinement**	**Summary** **Rationale *Business Value*** **Rationale *Architectural Impact***
1	Availability	Server uptime	The server should be up and running for 99.99% of the time.
			H: *SIoTip supports real time applications. a high uptime is evident.*
			L: *It's an agreement with the server provider.*
2	Security	Security on the channel	Third parties should not be able to listen to data sent over the channel from gateway to online service or start sending data over this channel when not authorised to do so. Successful attacks should be detected within 5 minutes.
			H: *Privacy of data is a big concern and we do not want unauthorised messages on the channel.*
			L: *This should be solved by the service agreement with the telecom operator.*
3		Unauthorised hardware	When an unauthorised piece of hardware (e.g. a sensor not from microPnP) is plugged into a mote, the infrastructure owner should be notified.
			L: *Hardware is sold via the SIoTip corporation, this hardware is authorised.*
			M: *The architecture needs to notify the infrastructure owner.*
4		DoS attack	A third party application should not be able to deny service of other applications by flooding the online service with data. Such flooding should be detected within 1 minute.
			H: *DoS attacks entail that some applications would not be processed anymore.*
			M: *The platform should be able to deal with such attacks.*
5		Fend off successful attacks	An attacker has got somehow access to the server/database. Such an attack can provide the attacker passwords, usernames and as a consequence the attacker might have access to all the accounts. All the passwords and usernames residing on the server/database should be encrypted and must not provide any information except the owner of the account.
			L: *As the infrastructure owner has no idea how data is stored so it can have low business value.*
			M: *Implementing this technique would require the SIoTIP team to rehash all the sensitive data on the database.*
6	Interoperability	Lossless information	Data sent across the channel should not be lost and should be received within 5 seconds.
			H: *It is essential that there is no loss of data or inconsistencies.*
			L: *This should be solved by the service agreement with the telecom operator.*
7	Modifiability	new type of hardware	new Hardware by a (new) manufacturer should take 2 man-weeks to implement and become available to the infrastructure owners.
			H: *At some point the hardware will become outdated.*
			M: *Our ability of dealing with this can separate us from rival companies.*

	Quality Attribute	Attribute Refinement	Summary Rationale ***Business Value*** Rationale ***Architectural Impact***
8		Change dashboard UI	When the dashboard UI becomes outdated, we should implement a new UI within 5 man-weeks.
			H: *At some point the UI will become outdated, and we want the dashboard to be easy to use.*
			L: *A UI doesn't have a big architectural impact.*
9		Change server provider	Switching server providers should not be noticeable for the users. No server downtime or data-loss during the switch between providers.
			L: *It's not the plan to change server providers, and it is not a selling point.*
			L: *This only means data is transferred to a different server.*
10		Extra functionality on the server.	The server needs to be updated. This update can involve adding extra functionality such as some machine learning techniques or some other functionality.Adding extra functionality should require minimal changes. The system should not require any changes to the code that is already written.
			M: *Business impact depends on the type of extra functionality being implemented.*
			M: *Changing the code can require the change in architecture.*
11		Change telecom operator.	Switching telecom operators should not be noticeable for the users. The communication during the switch between gateway and online service should stay up, no communication should be lost.
			L: *It's not the plan to change telecom operator, and it is not a selling point.*
			L: *The telecom operator has little architectural significance.*
12	Usability	CO (un)subscribes to application	When a CO subscribes or unsubscribes to an application it should be hassle free (in the dashboard it should take no more than a few clicks to (un)subscribe).
			H: *One of the main selling points of the platform is the ease of use.*
			M: *This does not have high architectural impact.*
13		Plug and play w/o effort	If a user plugs a new sensor/actuator/mote in, this should be automatically registered to the gateway with default values. The user should not have to search for, or select the newly plugged in sensor/actuator/mote on the platform, this way the human-error should be zero.
			H: *One of the main selling points of the platform is the ease of use.*
			M: *Some architectural impact because the platform needs to be updated.*
14	Monitorability	Defect Sensor,Actuator or Mote.	A sensor, actuator or mote is damaged and does not respond anymore. Such an item should be detected by the online-service within one heartbeat. The heartbeat received by the online-service from the mote does not contain the ID of the damaged sensor or actuator. If the online-service does not receive a heartbeat, then the mote is damaged.
			H: *It's important for the infrastructure owner to know if his hardware is defect.*
			L: *Online services can detect the missing hardware in one routine.*
15	Performance	Server throughput.	Many gateways can send data at the same time. The server should be able to process all the requests within 1 second. Requests processing speed of the server must be proportional or higher than the incoming data from gateways and the requests from the infrastructure owner.

	Quality Attribute	Attribute Refinement	Summary Rationale ***Business Value*** Rationale ***Architectural Impact***
			M: *Customer organisations want to have a fast response.*
			M: *As this requirement describes how the requests should be processed.*
16		Gateway can handle its connections.	The gateway can handle at least 25 motes. If there are too many connected, then the gateway should give a notification.
			L: *If many motes are required, the infrastructure owner can install multiple gateways.*
			L: *The infrastructure owner is required to install the new gateway and register it.*
17	Development Distributability	Development of system by different teams.	It should be possible to develop the system in different teams and different modules should have compatible interfaces.
			L: *Not important to the stakeholders.*
			H: *This requirement imposes a specific type of design.*
18	Mobility	Battery management	Battery of the battery-powered motes should last at least 1 month. When the battery is below 10%, the infrastructure owner is notified.
			L: *Not every mote uses batteries (some are plugged in sockets).*
			M: *The architecture needs to notify the infrastructure owner.*
19	Scalability	Server expansion	When the company grows bigger the platform will require more storage, We should reserve enough storage to accommodate double the planned amount of customers. Initially this is 200 and 50 different applications.
			L: *The users expect enough servers, it is not a selling point.*
			L: *The architecture doesn't change much. It just requires more space.*
20	Safety	Shared and non-shared components	It should be possible to allocate the different hardware components from same infrastructure owner to different customer organisations and same hardware, except those which are supposed to shared, should not be allocated to different customer organisations.
			H: *Dynamically allocating the hardware can be a selling proposition and can have medium to high business impact.*
			H: *This requirement has a higher architectural impact as it defines the blue print of the subsystem.*
21	Testability	Sandbox testing	The sandbox environment that tests new applications should finish testing within one day. This time contains the time necessary for human review by a SIoTIP system administrator when initial tests fail.
			M: *testing of applications should be fast but thorough.*
			L: *The speed of the testing has a low impact.*

2. Quality Attribute Scenarios

2.1 Performance: Server throughput.

The server receives requests from the infrastructure owner and the data from the gateways simultaneously.Multiple gateways can also try to synchronise the data at the same time. The server must be capable of dealing with all these requests in real time and without losing any data from the gateways.

- **Source:** Gateways,infrastructure owners and customer organisations.
- **Stimulus:**
  - Multiple gateways send the data simultaneously.
  - Many infrastructure owners want to adjust the settings at the same time.
  - Customer organisations want to consult their invoices or they want to manage the roles of the users.
- **Environment:** At run-time or design time.
- **Artifact:** Processing of the data.
- **Response:**
  - The system must be able to handle a large number of requests at the same time.
  - These requests should be executed independently and without any delay.
  - It should be possible to carry out read and write operations simultaneously on the database.
  - This doesn't affect that we want lossless information.
- **Response measure:**
  - Requests processing speed of the server must be at least proportional to the number of incoming requests.
  - Sever must be able to process at least $10,200$ requests per second initially.(50 different apps,200 initial market players, 200 dashboards of infrastructure owners).

2.2 Monitorability: A sensor, actuator, mote or gateway does not respond anymore.

A sensor,actuator or mote is damaged and does not respond anymore. Such a defect item should be detected by the gateway or Online service. A defect gateway should also be detected by the online service. Infrastructure owners must be notified upon detecting such components.

- **Source:** Mote, sensor, actuator or gateway.
- **Stimulus:**
  - The heartbeat received by the gateway from the mote does not contain the ID of a specific sensor or actuator.
  - Gateway has not received any data from the mote for a certain period of time.
  - A specific gateway has not synchronised the data with the online service for a certain period of time.
- **Artifact:** Processing of the data.
- **Environment:** At run-time.
- **Response:**

  - Gateway or the online service should detect the missing hardware component.
  - The infrastructure owner must be notified upon detecting such defect hardware.
  - This doesn't affect any working sensors, actuators, motes or gateways.
- **Response measure:**
  - It must be possible to detect the missing sensor within one periodic exchange of information between gateway and mote.

2.3 Usability: Customer organisation (un)subscribes to application

When a CO subscribes or unsubscribes to an application it should be hassle free (in the dashboard it should take no more than a few clicks to (un)subscribe.).

- **Source:** A customer organisation.
- **Stimulus:**
  - The customer organisation wants to subscribe to an application.
  - The customer organisation wants to unsubscribe from an application.
- **Artifact:** The customer organisation's dashboard and the database.
- **Environment:** At runtime.
- **Response:**
  - First the customer organisation finds a list of applications in it's dashboard, here it selects the application and after confirming that it is not a robot, it can click to subscribe.
  - The new subscription or un-subscription is be updated in the database.
  - The customer's organisation's dashboard updates.
  - If the customer organisation doesn't have all the required hardware, the infrastructure owner is notified.
  - This doesn't affect any existing subscriptions.
- **Response measure:**
  - The amount of clicks should be minimal. Max. two to find a list of applications, six to confirm it is not a robot and two to confirm the subscription.
  - The time between the final click and the update in the database and dashboard should be no more than two seconds.

2.4 Modifiability: Change dashboard UI

When the dashboard UI becomes outdated, a new UI should be implemented within 5 man-weeks.

- **Source:** The SIoTIP Corporation.
- **Stimulus:**
  - There is a demand for a new dashboard UI by the customer organisations.
  - The SIoTIP Corporation wants a new UI for the dashboard.
- **Artifact:** Code.
- **Environment:** At design time.
- **Response:**
  - Pinpoint the bad aspects of the old UI through user complaints or surveys.

  - Design a new UI and steadily roll it out to the client base. Ask for feedback on the new design.
  - Tweak and finalise the design and roll it out to the whole user base.
  - This doesn't affect the functionality on the dashboard.

- **Response measure:**
  - Designing and implementing the new UI should not take longer than 5 man-weeks.
  - The user satisfaction should go up after enrolling the new UI, measured with a survey.
  - Code of the other functionalities should not be changed while updating the dashboard.