



**Multi-layered
Security
Technologies**
for hyper-connected
smart cities

D3.1: M-Sec Requirements Analysis –

first version

February 2019



Grant Agreement No. 814917

Multi-layered Security technologies to ensure hyper-connected smart cities with Blockchain, BigData, Cloud and IoT

Project acronym	M-Sec
Deliverable	D3.1 M-Sec Requirements Analysis – first version
Work Package	WP3
Submission date	February 2019
Deliverable lead	Orfefs Voutyras (ICCS) / Koumoto Takafumi (NII)
Authors	Orfefs Voutyras (ICCS), Antonis Litke (ICCS), George Palaiokrassas (ICCS), Koumoto Takafumi (NII), Arturo Medela (TST), Jin Nakasawa (KEIO), Andrés Iglesias (WLI), Vanessa Clemente (WLI), Aamir Bokhari (YNU), Kenji Tei (WU), Mathieu Gallisot, Levent Gurgun (CEA), Keiko Doguchi (NTTE), Sonia Sotero Muñiz (AYTOSAN)
Internal reviewer	Arturo Medela (TST), Jin Nakasawa (KEIO)
Dissemination Level	Public
Type of deliverable	R

Worldline



TST



YNU



NTT DATA
Trusted Global Innovator



The M-Sec project is jointly funded by the European Union's Horizon 2020 research and innovation programme (contract No 814917) and by the Commissioned Research of National Institute of Information and Communications Technology (NICT), JAPAN (contract No. 19501).





Version history

#	Date	Authors (Organization)	Changes
V0.1	15 November 2018	Antonis Litke (ICCS)	Full ToC
V0.2	30 November 2018	Orfefs Voutyras (ICCS)	Guidelines for all Sections
V0.3	13 December 2018	Koumoto Takafumi (NII)	Sections 3.2 & 3.3
V0.4	14 December 2018	Arturo Medela (TST)	Sections 2.2, 2.7 & 3.4
V0.5	15 January 2019	Andrés Iglesias (WLI)	Sections 3.5, 3.8 & 3.9
V0.6	30 January 2019	Jin Nakasawa (KEIO)	Sections 2.4, 3.10, 3.11 & 3.12
V0.7	01 February 2019	Vanessa Clemente (WLI)	Section 2.3
V0.8	06 February 2019	Jin Nakasawa (KEIO)	Section 2.5
V0.9	06 February 2019	Arturo Medela (TST)	Section 3.5
V0.10	07 February 2019	Aamir Bokhari (YNU)	Section 3.13
V0.11	11 February 2019	Antonis Litke (ICCS)	Sections 1 & 3.14
V0.12	12 February 2019	Andres Iglesias (WLI)	Sections 2.3 & 3.7
V0.13	15 February 2019	Kenji Tei (WU)	Sections 3.16 & 3.17
V0.14	18 February 2019	George Palaiochrassas (ICCS)	Section 3.15
V0.15	19 February 2019	Mathieu Gallisot, Levent Gorgen (CEA)	Sections 3.18 & 3.19
V0.16	20 February 2019	Keiko Doguchi (NTTE)	Section 2.6
V0.17	21 February 2019	Sonia Sotero Muñiz (AYTOSAN)	Updates in Section 2
V0.18	22 February 2019	Orfefs Voutyras (ICCS)	Section 4, Initial Review
V0.19	26 February 2019	Jin Nakasawa (KEIO)	Internal Review
V0.20	26 February 2019	Arturo internal (TST)	Internal Review
V1.0	28 February 2019	Orfefs Voutyras (ICCS)	Final version



Table of Contents

Version history.....	3
Table of Contents	4
List of Tables	8
List of Figures.....	9
Glossary	10
1. Introduction	12
1.1 Scope of the document	12
1.2 Relation to other WPs and Tasks.....	12
1.3 Overall methodology followed	13
2. Overview of the use cases	14
2.1 Introduction - Methodology	14
2.2 Overview of Use Case 1 (SAN-UC1)	15
Reliable IoT devices with multi-layered security for a smart city	15
Stakeholders involved and specific needs and contributions.....	15
Requirements summary.....	16
2.3 Overview of Use Case 2 (SAN-UC2)	16
Home Monitoring & Wellbeing Tele-assistance for active and independent ageing people.....	16
Stakeholders involved and specific needs and contributions.....	18
Requirements summary.....	19
2.4 Overview of Use Case 3 (FUJ-UC3)	20
Secure and Trustworthy Urban Environment Monitoring with Automotive, Participatory, and Virtual Sensing Techniques.....	20
Stakeholders involved and specific needs and contributions.....	20
Requirements summary.....	21
2.5 Overview of Use Case 4 (FUJ-UC4)	21
Secure and Trustworthy Hyper-connected Citizen Care	21
Stakeholders involved and specific needs and contributions.....	22
Requirements summary.....	23
2.6 Overview of Use Case 5 (CB-UC5).....	23





A marketplace of IoT services for effective decision making	23
Stakeholders involved and specific needs and contributions.....	24
Requirements summary.....	25
2.7 Overview of Use Case 6 (CB-UC6).....	25
Citizens as sensor	25
Stakeholders involved and specific needs and contributions.....	25
Requirements summary.....	26
3. Project background and Technology assets brought by the partners.....	27
3.1 Introduction - Methodology	27
3.2 Asset 1 [NII]: Security Analysis Tools	28
Technology asset description	28
Relation to M-Sec.....	28
Requirements summary related to M-Sec.....	28
3.3 Asset 2 [NII]: Development Method for a secure service	28
Technology asset description	28
Relation to M-Sec.....	29
Requirements summary related to M-Sec.....	29
3.4 Asset 3 [TST]: TSmarT platform & TST IoT devices	30
Technology asset description	30
Relation to M-Sec.....	30
Requirements summary related to M-Sec.....	30
3.5 Asset 4 [TST]: Santander Pace of the City.....	31
Technology asset description	31
Relation to M-Sec.....	32
Requirements summary related to M-Sec.....	32
3.6 Asset 5 [WLI]: Mobile Wallet	32
Technology asset description	32
Relation to M-Sec.....	33
Requirements summary related to M-Sec.....	34
3.7 Asset 6 [WLI]: Connected Assistance.....	34
Technology asset description	34



Relation to M-Sec.....	34
Requirements summary related to M-Sec.....	35
3.8 Asset 7 [WLI]: Balena.io (FKA resin.io)	35
Technology asset description	35
Relation to M-Sec.....	36
Requirements summary related to M-Sec.....	36
3.9 Asset 8 [WLI]: Node-RED	36
Technology asset description	36
Relation to M-Sec.....	36
Requirements summary related to M-Sec.....	36
3.10 Asset 9 [KEIO]: KEIO Mobile Sensing Platform	37
Technology asset description	37
Relation to M-Sec.....	37
Requirements summary related to M-Sec.....	38
3.11 Asset 10 [KEIO]: KEIO SOX	38
Technology asset description	38
Relation to M-Sec.....	39
Requirements summary related to M-Sec.....	39
3.12 Asset 11 [KEIO]: Fujisawa MinaRepo.....	40
Technology asset description	40
Relation to M-Sec.....	40
Requirements summary related to M-Sec.....	41
3.13 Asset 12 [YNU]: YNU Honeypot (IoTPOt)	41
Technology asset description	41
Relation to M-Sec.....	42
Requirements summary related to M-Sec.....	42
3.14 Asset 13 [ICCS]: Blockchain framework.....	42
Technology asset description	42
Relation to M-Sec.....	44
Requirements summary related to M-Sec.....	45
3.15 Asset 14 [ICCS]: Quorum Blockchain framework	46



Technology asset description	46
Relation to M-Sec.....	48
Requirements summary related to M-Sec.....	48
3.16 Asset 15 [WU]: Modal Transition System Analyser (MTSA)	48
Technology asset description	48
Relation to M-Sec.....	49
Requirements summary related to M-Sec.....	49
3.17 Asset 16 [WU]: Runtime Environment Model Updater (REMU)	49
Technology asset description	49
Relation to M-Sec.....	50
Requirements summary related to M-Sec.....	50
3.18 Asset 17 [CEA]: Secured components for devices and gateways.....	50
Technology asset description	50
Relation to M-Sec.....	51
Requirements summary related to M-Sec.....	51
3.19 Asset 18 [CEA]: Eclipse sensiNact platform and Studio.....	51
Technology asset description	51
Relation to M-Sec.....	52
Requirements summary related to M-Sec.....	53
4. Elicitation of M-Sec Requirements	54
4.1 Methodology	54
4.2 M-Sec Platform Functional Requirements	54
4.3 M-Sec Platform Non-Functional Requirements	55
4.4 Consolidation and Coding of M-Sec requirements.....	56
5. Conclusions	69



List of Tables

Table 1. Stakeholders' needs and contribution in SAN-UC1	15
Table 2. Stakeholders' needs and contribution in SAN-UC2	18
Table 3. Stakeholders' needs and contribution in FUJ-UC1	20
Table 4. Stakeholders' needs and contribution in FUJ-UC2	22
Table 5. Stakeholders' needs and contribution in CB-UC1.....	24
Table 6. Stakeholders' needs and contribution in CB-UC2.....	26
Table 7: Consolidated and Coded M-Sec Platform Generic Functional requirements	56
Table 8: Consolidated and Coded M-Sec Use Case specific Functional requirements	57
Table 9: Consolidated and Coded M-Sec Assets Functional requirements.....	59
Table 10: Consolidated and Coded M-Sec Non-Functional Security & Privacy requirements	62
Table 11: Consolidated and Coded M-Sec Non-Functional Scalability requirements.....	65
Table 12: Consolidated and Coded M-Sec Non-Functional Performance requirements	65
Table 13: Consolidated and Coded M-Sec Non-Functional Reliability & Availability requirements.....	66
Table 14: Consolidated and Coded M-Sec Non-Functional Manageability & Flexibility requirements	66
Table 15: Consolidated and Coded M-Sec Non-Functional Openness & Extensibility requirements.....	67
Table 16: Consolidated and Coded M-Sec Non-Functional Design & Implementation requirements	67



List of Figures

Figure 1—1: M-Sec requirements analysis methodology	13
Figure 2—1: Home Monitoring UML diagram.....	17
Figure 2—2: Social isolation and wellbeing monitoring UML diagram	17
Figure 2—3: Use Case 4 UML diagram	22
Figure 2—4: Use Case 5 UML diagram	24
Figure 3—1: Security Analysis Tool	28
Figure 3—2: Process of the development method for a secure service	29
Figure 3—3: Development Process for a secure and privacy-friendly cloud service	29
Figure 3—4: An example of TST's IoT boards	30
Figure 3—5: Pace of the City event cycle	31
Figure 3—6: Mobile Wallet - adding cryptocurrency to the balance and getting entitlement for reproducing content	33
Figure 3—7: Storyboard connected assistance	34
Figure 3—8: Resin.io workflow (balena.io)	35
Figure 3—9: Flow editing in Node-RED	36
Figure 3—10: Automotive Sensing Trucks in Fujisawa City.....	37
Figure 3—11: Screenshot of garbage bag counting mechanism.....	38
Figure 3—12: Overview of Sensor over XMPP platform	39
Figure 3—13: Overview of Fujisawa MinaRepo	40
Figure 3—14: IoT Devices Security	41
Figure 3—15: Hyperledger component diagram.....	42
Figure 3—16: Hyperledger Composer REST server	43
Figure 3—17: Hyperledger Explorer User Interface	44
Figure 3—18: Overview of Quorum Framework.....	46
Figure 3—19: Private Transaction Flow in Quorum	47
Figure 3—20: Example of application of the MTSA to a reactive system	49
Figure 3—21: High-level view of REMU.....	49
Figure 3—22: Examples of Secure Components	51
Figure 3—23: Overview of sensiNact Platform and Studio	52





Glossary

API	Application Programming Interface
App	Application
BT	Bluetooth
CB	Cross-border
CoAP	Constrained Application Protocol
CPU	Central Processing Unit
CRUD	Create, Read, Update, Delete
D	Deliverable
DDoS	Distributed Denial of Service
EU	European Union
FUJ	Fujisawa
GDPR	General Data Protection Regulation
H2020	Horizon 2020
HTML	Hypertext Markup Language
HTTP	Hypertext Transfer Protocol
IBFT	Istanbul Byzantine Fault Tolerance
ID	Identifier
IoT	Internet of Things
JP	Japan
(L)GPL	(Lesser) General Public License
M	Month
MAC	Media Access Control
MQTT	Message Queuing Telemetry Transport
OS	Operating System
PIPA	Personal Information Protection Act
PKI	Public Key Infrastructure
PM	Particulate Matter
QoL	Quality of Life
REST	Representational State Transfer





SAN	Santander
SPI	Serial Peripheral Interface
SSH	Secure Shell
T	Task
UC	Use Case
UML	Unified Modelling Language
USB	Universal Serial Bus
UV	Ultraviolet
WP	Work Package
XMPP	Extensible Messaging and Presence Protocol



1. Introduction

1.1 Scope of the document

The current document is the deliverable 'D3.1 M-Sec Requirements Analysis – first version' which comprises the first major outcome of the task 'Task 3.1 – System level and User level Requirements analysis'. Task T3.1 is in charge of identifying system and use cases related requirements that reflect real citizen's needs, and consolidating all of them in a way that will facilitate the design of the overall M-Sec system. As such, this deliverable contains an extensive elicitation of user and technical side requirements.

The primary audience of this document consists of the members of the consortium that will participate in the design and development of the components and modules of the M-Sec system as well as of the system per se. Additionally, the document is of wider interest to stakeholders that are active in the domains of smart cities, IoT, security and Big Data, including researchers participating and contributing to H2020 projects under the aforementioned topics.

This deliverable is a live document following an iterative approach and thus it is going to have a final version on M24 which will include the updated requirements captured by the different user groups engaged during the pilots.

1.2 Relation to other WPs and Tasks

Task 3.1 receives input from WP2 and in particular from Tasks "T2.1 M-Sec use case description" and "Task 2.2 M-Sec Pilots: definition, setup and citizens involvement" through the corresponding deliverables (D2.1 and D2.2). More specifically, these deliverables provide in a holistic way an overview of the use cases description along with particular details on their implementation within the pilots. To do so, D2.1 provides a full analysis of the use cases that will be covered in the M-Sec project. D2.2 then describes the different actors/stakeholders involved, the functions and conditions that need to be offered, how, when and where the pilots will be set up, and the plan for engaging and committing the participation of the citizens and the stakeholders.

At the same time, Task 3.1 is in close alignment to and receives input from Task "T5.3 GDPR compliance" so as to include input related to GDPR compliance in the overall requirements of the project.

Regarding the exploitation of the output of Task 3.1, having as an aim the full definition and consolidation of the M-Sec requirements, the Task provides its results to Task "T3.2 M-Sec Architecture" by contributing to the overall design and specification of the M-Sec system and to Task "T3.3 Risks and security elements for a hyper-connected smart city" by supporting the further study and analysis of the risks and security elements that affect modern and future smart cities.

The outcomes of this deliverable will be also used as input to the technical deliverables of WP4 as well as the overall integration plans of WP2, as they will be described through Task "T2.3 Overall Integration".



1.3 Overall methodology followed

The following figure gives an overview of the methodology that is being followed to complete Task 3.1.

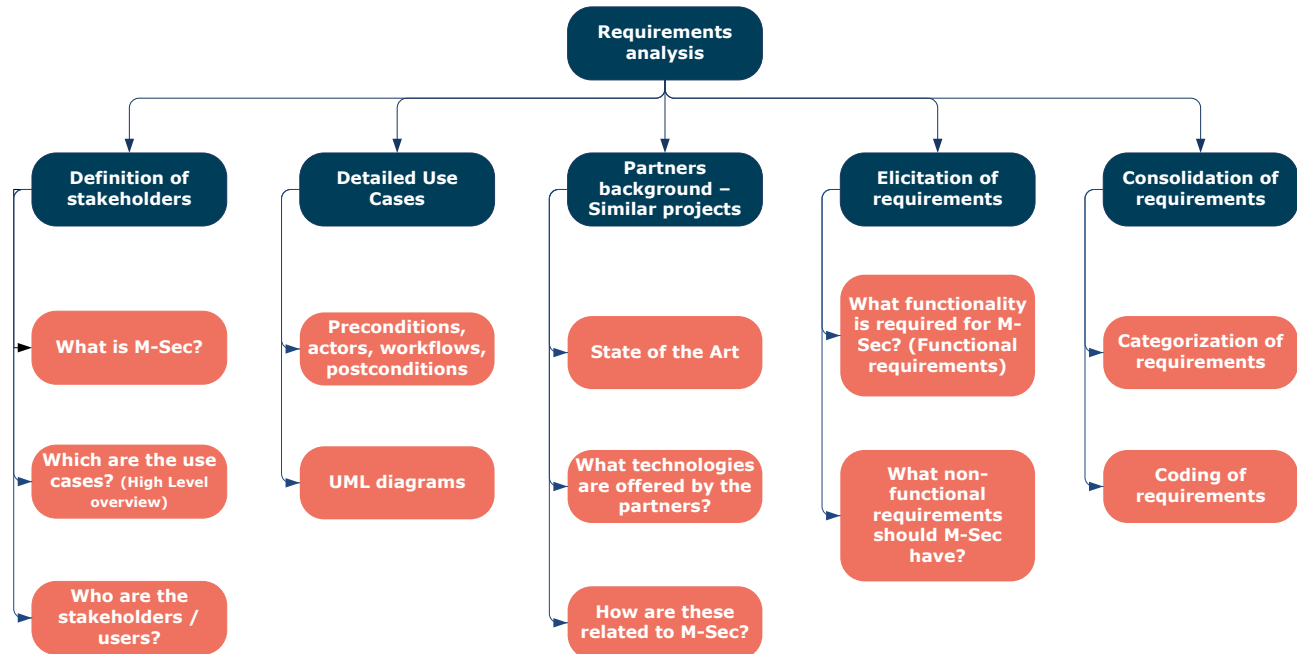


Figure 1—1: M-Sec requirements analysis methodology

The analysis starts with a definition of the M-Sec concept, including an overview of the use cases, use case diagrams and the stakeholders involved so as to understand the context of the project and identify the various stakeholders. These specific steps (1st and 2nd column in Figure 1—1) have been implemented within Task 2.1 and Task 2.2 and are reflected in the corresponding deliverables.

As a next step (and based on the previous one), the analysis focused on requirements from potential end-users of the M-Sec platform, including both corporate users and citizens (4th column in Figure 1—1). A variety of modalities was exploited towards eliciting requirements, including review of the state-of-the-art services and direct contact with all stakeholders that comprise the M-Sec value chain. Direct contact with stakeholders was pursued based on the partners' business networks, involving experts from the large industrial partners of the consortium. These results are presented in **Section 2**.

In parallel with this step, the consortium partners gave an overview of the technologies that are going to be involved in the project and the perspective of using them in order to implement the M-Sec concept. Similar projects and background from previous projects are also mentioned in order to present the state-of-the-art and the previous achievements that can be used as a starting point (3rd column in Figure 1—1). These results are presented in **Section 3**.

In the sequel, the elicitation of requirements is derived from the definition of the desired functionality of the M-Sec system given from the perspectives of the functional components and the non-functional attributes that the final system has to expose (4th column in Figure 1—1). Finally, the requirements are gathered and consolidated into one list, grouped and coded accordingly in order to comprise the reference for the design, implementation and validation phases of the project (5th column in Figure 1—1). These results are presented in **Section 4**.



2. Overview of the use cases

2.1 Introduction - Methodology

The requirements analysis process relies heavily on the involvement of the stakeholders in the whole value chain that the project brings. It should be noted that the M-Sec consortium includes all necessary stakeholders of the M-Sec value chain. In particular, the consortium includes smart city infrastructure providers, technology providers as well as service providers and integrators (i.e. the technical partners from EU and JP side) and end users as these are going to be recruited in WP5. This approach allows for a credible validation of the M-Sec concept, along with different deployment configurations and services operations plans.

The first group of requirements is provided directly from an overview of the use cases:

- SAN-UC1: Reliable IoT devices with multi-layered security for a smart city (**subsection 2.2**)
- SAN-UC2: Home Monitoring & Wellbeing Tele-assistance for active and independent ageing people (**subsection 2.3**)
- FUJ-UC1: Secure and Trustworthy Urban Environment Monitoring with Automotive, Participatory, and Virtual Sensing Techniques (**subsection 2.4**)
- FUJ-UC2: Secure and Trustworthy Hyper-connected Citizen Care (**subsection 2.5**)
- CB-UC1: A marketplace of IoT services for effective decision making (**subsection 2.6**)
- CB-UC2: Citizens as sensor (**subsection 2.7**)

In order to extract the requirements from the use cases of the project more easily, the following subsections are included for each use case, focusing on input/needs of potential end-users of the M-Sec platform, including corporate users and citizens:

- **Use Case description:** This subsection presents briefly the main use case per se (what is the “problem”/scenario expected to be tackled and what are its parameters).
- **Stakeholders involved and specific needs and contributions:** This subsection provides a list of the stakeholders involved in the use case as well as their main needs that should be covered by the use case and the project. It also describes the stakeholders’ direct or indirect contributions to the project. More specifically, for each stakeholder their Scenario specific needs and Security/Privacy specific needs are presented as well as their potential contribution to the project. It should be noted that a distinction is made between security/privacy needs and other (scenario specific) needs for the easier categorisation of the corresponding requirements. Although the presentation of security/privacy needs throughout the use cases may be repetitive or too generic, it is necessary so as to identify the vulnerable actors that M-Sec will attempt to protect.
- **Requirements summary:** More technical details about needs/contributions are given in this subsection through a list of Scenario specific and another list of Security/Privacy specific requirements. Some of these requirements are quantitative. This summary of requirements is the main input provided to **Section 4**.



2.2 Overview of Use Case 1 (SAN-UC1)

Reliable IoT devices with multi-layered security for a smart city

Governments around the world are devoting significant effort and resources to the management of the environment. Likewise, the city of Santander is also involved in this activity and is trying to carry out an effective policy for environmental management through the signing of agreements that aid improvements in air quality and QoL (Quality of Life) for its citizens. A key element in undertaking this task is the noise and temperature level measurements.

Currently, noise measurements are taken by the Engineering Department of the Council in a timely manner; generally, in response to complaints, abnormal operation of some service, etc. However, it is an important issue for all cities to provide a real time heat map and periodic reports of noise pollution levels which allows monitoring of the city. In addition, another interesting parameter from an environmental perspective is temperature monitoring at different points across the city. Indicatively, its study could provide measurements useful for research on the interaction of the traffic and pollutants with the environment and global warming.

In this sense, it may also be of high interest for both citizens and the City Council to know at every time the 'occupancy' levels of certain spots in the city, such as the beaches during summertime. If the corresponding heat map suggests that there are a lot of people in a specific area, users could pick other spots and diminish the traffic and pollutants, while the Municipality will be able to sketch specific initiatives based on these data.

Stakeholders involved and specific needs and contributions

The stakeholders involved in this use case and the main needs that should be covered by the use case and the project as well as the direct or indirect contributions of the stakeholders' to the project are presented in the following table:

Table 1. Stakeholders' needs and contribution in SAN-UC1

Stakeholder	Scenario specific needs	Security/Privacy specific needs	Contribution
Citizens/ Tourists/ Visitors	<ul style="list-style-type: none">• Check temperature and noise levels in certain city spots• Get real-time information about the 'occupancy' of some city spots	<ul style="list-style-type: none">• Hide sensitive data of users checking this information	<ul style="list-style-type: none">• Report high-noise situations• Send mobile phone information to create "occupancy" heat map
Municipality	<ul style="list-style-type: none">• Receive updated temperature and noise information• Receive real-time information about 'occupancy' of selected city spots	-	<ul style="list-style-type: none">• Create strategies and offer services adapted to the data received• Engagement of end-users



- Get generation of statistics on top of the received data

Solution provider

- Contribute with new devices that will provide new and reliable sources of city information to be integrated with existing ones.
- Handle properly mobile phone IDs of people present at the selected city spots
- Development and evaluation of new technologies in a real environment

Requirements summary

The Scenario specific requirements for this use case are the following:

- Deployed devices should not impact negatively the scenario nor affect the daily operations as they are before their deployment.
- The service should facilitate the visualisation of real-time information in a map or in a list (physical sensing information).
- The service should facilitate the visualisation of historical information in a map or in a list (physical sensing information).
- The application should provide a tool to analyse data and extract statistics in a simple and easily understandable way for the city economic development division and event organisers.
- The service should offer the option to users to publish Events.
- The associated web application should gather satisfaction information from the users.
- The local architecture should be scalable and integrated with others.

The Security/Privacy specific requirements for this use case are the following:

- The mobile phones IDs of end-users interacting with the city spots where the pilot is carried out should be anonymised.
- The associated web application should protect the privacy of the end-user, propose several levels of management of personal data, and give the option of modifying the privacy parameters any time.

2.3 Overview of Use Case 2 (SAN-UC2)

Home Monitoring & Wellbeing Tele-assistance for active and independent ageing people

Use case 2 carried out in Santander city intends to face the main challenge of the rapid increase of elderly population during the past years caused by the increase of life expectancy due to medical, social and economic advances. Ageing people may feel isolated due to the lack of close family ties or the result of living alone. Additionally, many ageing citizens live with a constant fear of falling or becoming unwell without being detected or helped by others for a long time. Therefore, the consortium aims to provide a solution that already covers some issues related to wellbeing, safety at home and fear of isolation. The solution will be implemented through two different pilots as it involves two different user profiles and contexts.



The first pilot (Pilot 2.1) is going to focus on home activity monitoring through the use of sensors such as presence sensors, temperature sensors, smart plugs, etc. This pilot has the aim to digitalise some of the current analogic-based, tele-assistance service provided by the Social Services department of the Santander City Council through a third-party operator.

The second pilot (Pilot 2.2) is going to focus on isolation and wellbeing monitoring and will be based on providing devices that e.g. count steps or monitor sleep, and tools providing access to available city activities. In addition, elderly citizens and the care giving network will be provided with a mobile app featuring communication capabilities to fight social exclusion and isolation (chat, video-chat and call).

The two following UML diagrams better illustrate the two different pilots.

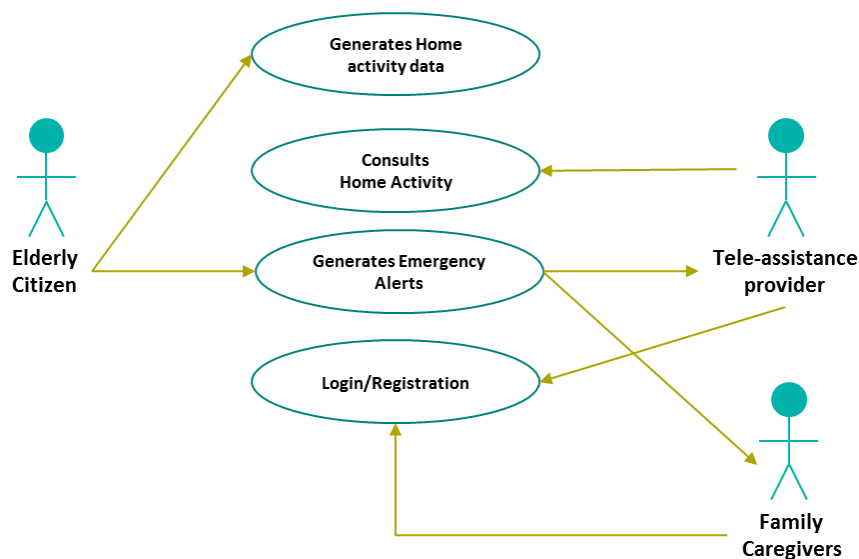


Figure 2—1: Home Monitoring UML diagram

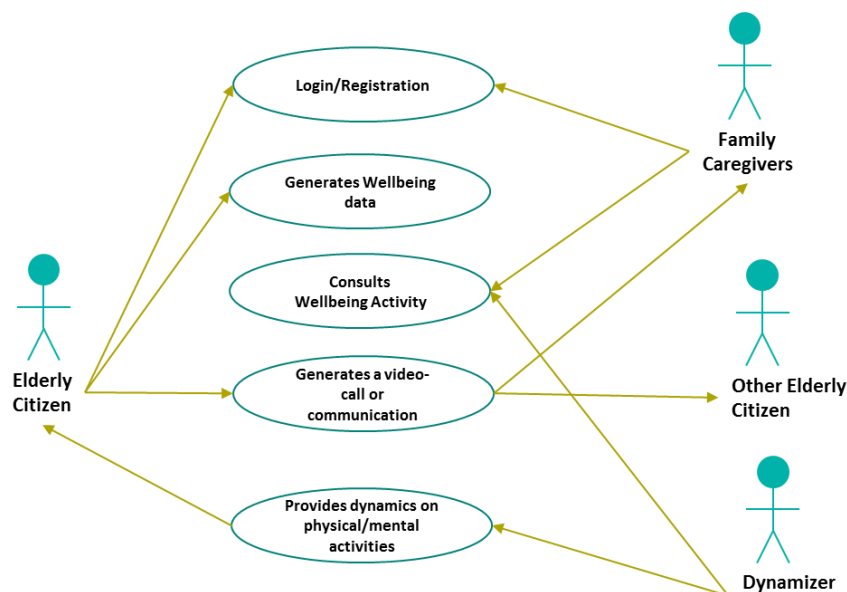


Figure 2—2: Social isolation and wellbeing monitoring UML diagram



Stakeholders involved and specific needs and contributions

The stakeholders involved in this use case and the main needs that should be covered from the use case and the project and their direct or indirect contributions to the project are presented in the following table:

Table 2. Stakeholders' needs and contribution in SAN-UC2

Stakeholder	Scenario specific needs	Security/Privacy specific needs	Contribution
Ageing People	<ul style="list-style-type: none">• Information of the wellbeing parameters monitored such as steps, sleep, weight (pilot 2.2)• Information regarding city events/activities (pilot 2.2)• Access to the video call, call tool (Pilot 2.2)		<ul style="list-style-type: none">• Evaluating the provided services (like/dislike)• Creating a network with family or caregivers to be controlled (pilot 2.1 and 2.2)
Relatives	<ul style="list-style-type: none">• Information of the wellbeing parameters monitored (pilot 2.2)• Alert warning of unusual or dangerous events from the user (pilot 2.1 and 2.2)• Information regarding city events/activities (pilot 2.2)• Access to the video call, call tool (Pilot 2.2)	<ul style="list-style-type: none">• Protection of personal data collected by the devices (pilot 2.1 and 2.2)• Protection of personal data from the registration process (Pilot 2.2)• User data authentication (Pilot 2.2)• Protection of communications (Pilot 2.2)	<ul style="list-style-type: none">• Evaluating the provided services (like/dislike) (Pilot 2.1 and 2.2)• Supporting the use of the solution to their ageing relatives (Pilot 2.2 and 2.2)• Providing advice or help on the use of the devices (Pilot 2.2)• Taking care of the parameters monitored (Pilot 2.1 and 2.2)
Caregivers	<ul style="list-style-type: none">• Information regarding the physical activity of ageing people within their network (pilot 2.2)• Alert warning of unusual or dangerous events from the user (pilot 2.1)• Access to the video call, call tool (pilot 2.2)		
Social Services/Tele-assistance service providers	<ul style="list-style-type: none">• Information of the wellbeing parameters monitored through presence sensors, window/door sensors, emergency alert (pilot 2.1)	-	<ul style="list-style-type: none">• Evaluating the provided services (like/dislike)• Taking care of the parameters monitored (Pilot 2.1)• Engagement of end-users



Dynamizer (City of Santander)	<ul style="list-style-type: none">• Access to the platform to include available activities for ageing citizens	-	<ul style="list-style-type: none">• Engagement with end users of the provided solution• Facilitator of public activities taking place in the city
Solution provider	<ul style="list-style-type: none">• Integration of the specific needs of the stakeholders mentioned in this table	<ul style="list-style-type: none">• Ensure GDPR compliance when personal data is involved	<ul style="list-style-type: none">• Digitalization of some of the current analogic-based tele-assistance service and new app focused on elderly people.

Requirements summary

The Scenario specific requirements for this use case are the following:

- The system should let users collect information about their wellbeing status (e.g. steps per day).
- The system should let users collect information from sources not directly attached to their Body Area Network, such as sensors in the room where the user is.
- The system should store massive information (such as temperature per second) in an efficient way, taking into account that not all data are expected to be recorded/stored forever.
- The system should store information (such as access to data) in a way that ensures this information will not be forgotten, and with mechanisms that allow third parties to verify that this information is correct and true.

The Security/Privacy specific requirements for this use case are the following:

- The system should have an access control policy, binding the users with different profiles, each with different access privileges to data (the owner of the data, person assigned by them to consult their data –family member or professional-, software administrator, technical support, security officer...)
- The system should have several policies of access to the data, depending on the role of the user and the data the user is trying to access to. Anonymous users should be given no access to any data related to users.
- The system should have a dashboard for matching roles to policies and privileges of access.
- The system should support an easy to use mechanism that enables the owner of the data to grant privileges of access to other users, in an understandable way.
- The system should support mechanisms for authentication and authorisation of the users, including updates of the users' proofs of access privileges.
- The system should include security measures which protect data transmitted over the network at application level against eavesdropping (encryption and peer authentication).
- If there is a role that can assign privileges of access to users on behalf of the owner of the data, the system should enable a way to record the user decision about this assignation (such as signing a consent form prior to assigning those privileges).



- The system should show and record the user consent about the usage of their data. The consent should be CRUD at any moment, and logs of the decisions of the users about their data should be kept.
- The system should keep logs of the interaction that the users have with the system, especially when getting access to data (not the data themselves). The logs should include the profile the data was accessed from, as well as the action(s) performed (authentication, CRUD on data, access granting/validation/ revocation/ removal, signing of consent, etc.).
- The system should store sensitive information (users' data and detailed logs that could reveal information about users or secrets about the system itself) in a way that this information could be permanently deleted.
- The system should support replying to queries concerning Data Protection of the users that employ the system. If this request cannot be satisfied in real-time, the system should guarantee that an answer will be provided in a reasonable time frame.

2.4 Overview of Use Case 3 (FUJ-UC3)

Secure and Trustworthy Urban Environment Monitoring with Automotive, Participatory, and Virtual Sensing Techniques

This use case provides a client application that allows urban environment monitoring entities (for example local governments) to visualise spatially and temporarily dense environmental data. Using the application, the entities are enabled to better serve their citizens with sophisticated environment monitoring. In this scenario, an automotive sensing platform is used to generate real-time environmental sensor data streams from all over the city of Fujisawa leveraging a hundred mobile sensing trucks.

This use case illustrates how the M-Sec platform secures such a mobile sensing platform by meeting the following objectives. Firstly, the heterogeneous components involved in the data stream dissemination are secured so that they are not cracked by malicious attackers. Secondly, the data streams are secured so that the data are not tampered in the network between their source and destination. Finally, the data streams do not harm citizens' privacy, thus an automated privacy protection mechanism should be provided.

Stakeholders involved and specific needs and contributions

The stakeholders involved in this use case and the main needs that should be covered from the use case and the project and their direct or indirect contributions to the project are presented in the following table:

Table 3. Stakeholders' needs and contribution in FUJ-UC1

Stakeholder	Scenario specific needs	Security/Privacy specific needs	Contribution
Local government	<ul style="list-style-type: none">• Data streams regarding city environment, such as air pollution, road roughness, etc.	<ul style="list-style-type: none">• Protection of the data streams from malicious attacks at the IoT devices• Protection of the data	<ul style="list-style-type: none">• Offering a service to citizens regarding city environment monitoring• Offering a service to





	<ul style="list-style-type: none">• Distributed cloud platform through which data streams are disseminated• IoT devices that generate the data streams	<ul style="list-style-type: none">streams from malicious attacks at the distributed cloud platform• Permission of access to the data streams only to those who have valid rights	<ul style="list-style-type: none">municipal officers regarding city environment monitoring
Citizens	<ul style="list-style-type: none">• Information regarding city environment• Access to the distributed cloud platform to acquire data	<ul style="list-style-type: none">• Receiving untampered environment data streams	<ul style="list-style-type: none">• Evaluating the provided services (like/dislike)

Requirements summary

The Scenario specific requirements for this use case are the following:

- The system should collect environment sensor data.
- The system should be able to handle a large number of data streams concurrently.
- The system should be able to transfer the data streams in real time.

The Security/Privacy specific requirements for this use case are the following:

- The system needs to secure the heterogeneous components involved in the data stream dissemination, so that they are not cracked by malicious attackers.
- The system needs to secure the data streams, so that the data are not tampered in the network between their source and destination.
- The system should protect the data streams from malicious attackers at the edge and distributed cloud platform.
- The system should disseminate the environment data stream to citizens securely.
- The system should not harm citizens' privacy; thus, an automated privacy protection mechanism should be provided.
- The system should grant access only to those with valid access rights.

2.5 Overview of Use Case 4 (FUJ-UC4)

Secure and Trustworthy Hyper-connected Citizen Care

In this use case, "Hyper-connected citizen care applications" will be created for a range of different purposes and for different stakeholders. On one hand, a government officers' application will collect city related data, (such as urban waste generation per household, pedestrian flow or traffic flow data, etc.) through the M-Sec architecture and analyse the data to produce value-added data that affect citizens efficiently. Citizens' applications, on the other hand, will consume that value-added data to empower their decision on related topics towards better (physical, mental or social) wellbeing or QoL.

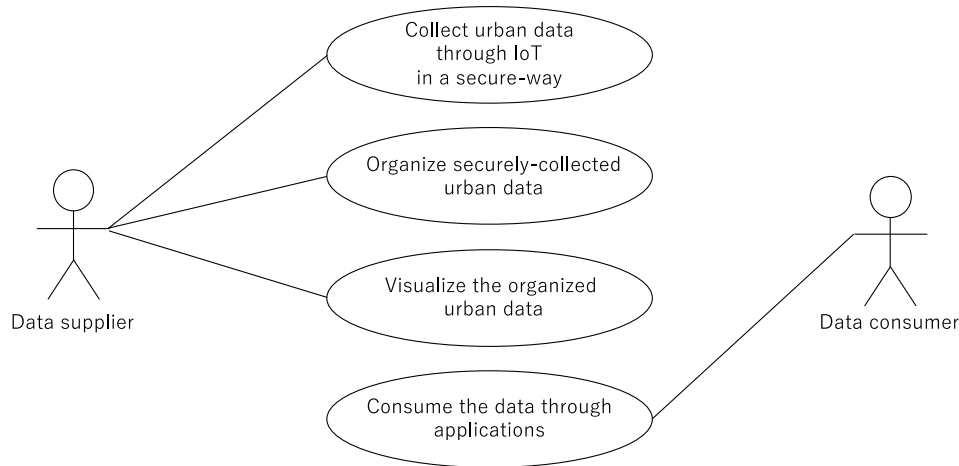


Figure 2—3: Use Case 4 UML diagram

Stakeholders involved and specific needs and contributions

The stakeholders involved in this use case and the main needs that should be covered from the use case and the project and their direct or indirect contributions to the project are presented in the following table:

Table 4. Stakeholders' needs and contribution in FUJ-UC2

Stakeholder	Scenario specific needs	Security/Privacy specific needs	Contribution
Local government	<ul style="list-style-type: none">• Data streams regarding citizens life, such as urban waste generation per household, pedestrian flow, traffic flow, etc.• Distributed cloud platform through which data streams are disseminated• IoT devices that generate the data streams	<ul style="list-style-type: none">• Clarifying whether or not the devices are compromised, enabling consumers to know whether the data they see are safe/authentic• Protection of the end-to-end data streams from malicious attacks• Protection of privacy information which may be contained in images	<ul style="list-style-type: none">• Offering a service to citizens regarding citizens care• Offering a service to municipal officers regarding citizens care
Citizens	<ul style="list-style-type: none">• Receiving live data streams regarding citizens life• Access to the distributed cloud platform to acquire data streams	<ul style="list-style-type: none">• Receiving untampered data streams• Citizens privacy information, such as the amount of one's urban waste generation, their private car numbers and their faces captured in an image, is not leaked	<ul style="list-style-type: none">• Evaluating the provided services (like/dislike)



Requirements summary

The Scenario specific requirements for this use case are the following:

- The system should collect heterogeneous data on citizens' life real time.
- The system should be able to handle a large number of data streams concurrently.
- The system should be able to transfer the data stream real time.
- The local architecture should be scalable and can be integrated with others.
- Deployed devices will not impact negatively in the scenario nor affect the daily operations as they are before their deployment.
- The associated web application should provide and visualise environment information collected over the city.
- The application should provide a tool to analyse data and extract statistics in simple and easily understandable way for the city environment division and citizens.

The Security/Privacy specific requirements for this use case are the following:

- The system needs to secure the heterogeneous components involved in the data stream dissemination, so that they are not cracked by malicious attackers.
- The system needs to secure the data streams, so that the data are not tempered in the network between their source and destination.
- The system should not harm citizens' privacy; thus, an automated privacy protection mechanism should be provided.
- The system should disseminate the data stream to municipalities and citizens securely
- The system should protect the data streams from malicious attackers at the edge and distributed cloud platform
- The system should grant accesses from whom own a valid access right
- The cloud system should store the data securely so that they are not disclosed to any party without permission

2.6 Overview of Use Case 5 (CB-UC5)

A marketplace of IoT services for effective decision making

The aim of this use case is to construct a marketplace between EU and Japan to distribute data by ensuring Confidentiality, Integrity, Availability, and Privacy of data following GDPR/PIPA regulations, so that people or organisations in EU and Japan can utilise the data more effectively.

Recently, foreign visitors are increasing around the world. Business opportunities are expected in various situations. In such circumstances, data distribution between countries needs to take place safely and smoothly done to make the data effective enough to contribute in "building" the smart city.

Along with the development of the Internet in recent years, since cyber-attacks are becoming increasingly complicated and sophisticated, provision of a secure data distribution method between countries is an essential task for smart cities.



The aim of this use case is to construct a marketplace where data integrity is present or tamperproof data can be securely distributed.

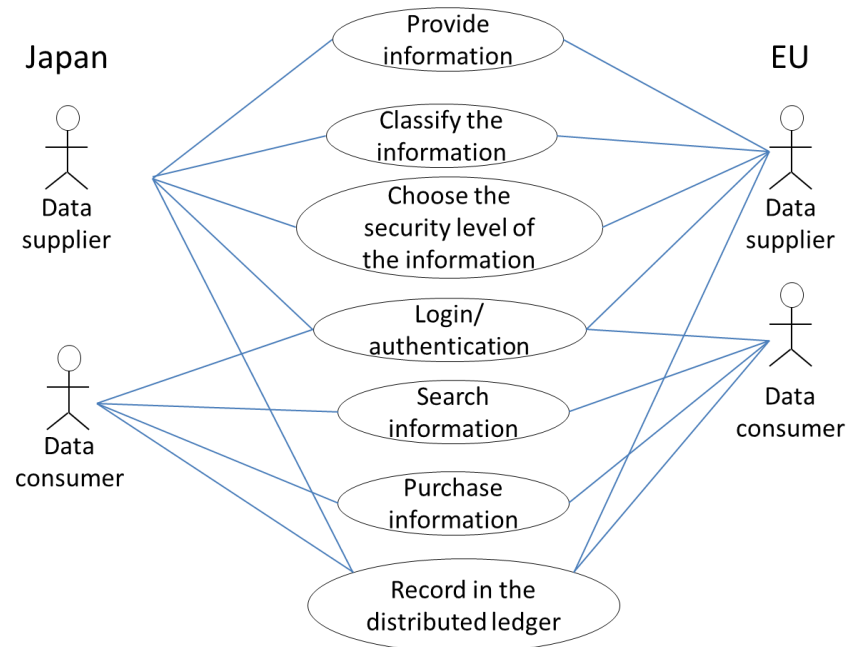


Figure 2—4: Use Case 5 UML diagram

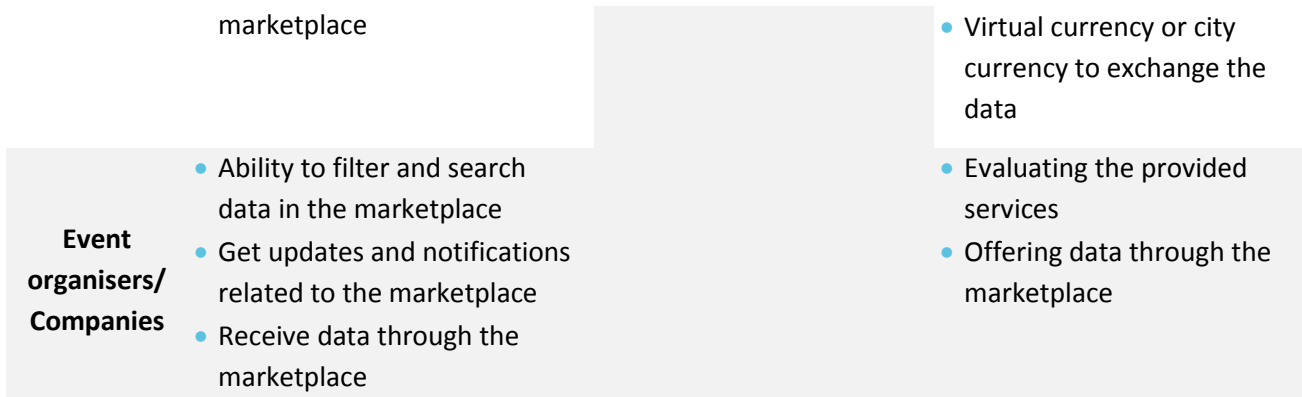
Stakeholders involved and specific needs and contributions

The stakeholders involved in this use case and their main needs that should be covered from the use case and the project as well as their direct or indirect contributions to the project are presented in the following table:

Table 5. Stakeholders' needs and contribution in CB-UC1

Stakeholder	Scenario specific needs	Security/Privacy specific needs	Contribution
Citizens/ Tourists/ Visitors	<ul style="list-style-type: none">• Ability to filter and search data in the marketplace• Get updates and notifications related to the marketplace• Receive data through the marketplace	<ul style="list-style-type: none">• Protection of personal data collected by the devices• Protection of personal data from the registration process	<ul style="list-style-type: none">• Evaluating the provided services• Offering data through the marketplace
Municipality	<ul style="list-style-type: none">• Ability to filter and search data in the marketplace• Get updates and notifications related to the marketplace• Information regarding city events/activities• Receive data through the	<ul style="list-style-type: none">• User data authentication for secure access• Permissioned blockchain for privacy	<ul style="list-style-type: none">• Evaluating the provided services• Generating internal statistics to keep track of the city evolution• Offering data through the marketplace





Requirements summary

The Scenario specific requirements for this use case are the following:

- The devices should be used to collect users' data such as behaviour characteristic data.

The Security/Privacy specific requirements for this use case are the following:

- The local architecture should be processed securely.
- The cloud system should store the data securely and should be accessed by both EU and Japan.
- The block chain technology should be utilised to secure the data.

2.7 Overview of Use Case 6 (CB-UC6)

Citizens as sensor

In this scenario, users utilise their mobile phones to share information and make reports about incidents and events that take place throughout the city (e.g. some street lights in a certain neighbourhood are not working properly or the road presents some unexpected bumps). This information is pushed to the M-Sec platform. Users can also subscribe to services such as "Pace of the city" (see **subsection 3.5**) or "MinaRepo" (see **subsection 3.12**), where they can get alerts for specific types of events currently occurring in the city. The users will receive the notifications on the occurred events via a smartphone application in the preferred language.

All users interested in receiving the notifications have to register with the service, create a personal profile (including e.g. the preferred language) and select the information they are interested to. This subscription may be done via web interface or directly through the application. If the Council wants to provide the service also to users without web access, it can provide a phone number of a help desk to be called in order to subscribe to the service with operator support.

Stakeholders involved and specific needs and contributions

The stakeholders involved in this use case and the main needs that should be covered from the use case and the project and their direct or indirect contributions to the project are presented in the following table:



Table 6. Stakeholders' needs and contribution in CB-UC2

Stakeholder	Scenario specific needs	Security/Privacy specific needs	Contribution
Citizens / Tourists/ Visitors	<ul style="list-style-type: none">• Subscription to certain events happening in the city• Get updates and notifications related to events the user is subscribed to• Ability to filter and search for events	<ul style="list-style-type: none">• Hide sensitive data of citizen using the app	<ul style="list-style-type: none">• Send reports of incidents to municipal services• Evaluate proper resolution of those incidents
Municipality		-	<ul style="list-style-type: none">• Solve incidents as reported by citizens and visitors• Generate internal statistics to keep track of the city evolution
Solution provider	<ul style="list-style-type: none">• Integration of the specific needs ensuring security	<ul style="list-style-type: none">• Handle properly IDs of people reporting incidents	<ul style="list-style-type: none">• An improved communication channel between citizens and city council

Requirements summary

The Scenario specific requirements for this use case are the following:

- The app should enable the users to publish Events.
- The app should offer users the option to Subscribe/Unsubscribe to specific types of events occurring in the city.
- The app should let users search for events filtering by date, type or location.
- The app should facilitate the visualisation of historical information in a map or in a list (physical sensing information).
- The application should provide a tool to analyse data and extract statistics in a simple and easily understandable way for the municipal services and citizens.

The Security/Privacy specific requirements for this use case are the following:

- The IDs of citizens and visitors reporting incidents should be anonymised.
- The cloud system should store the data securely so that they are not disclosed to any party without permission.



3. Project background and Technology assets brought by the partners

3.1 Introduction - Methodology

In this section, the technical partners of the consortium provide an overview of the technologies that can be involved in the project and the perspective of using them in order to implement the M-Sec concept. Similar projects and background from previous projects are also mentioned in order to present the state-of-the-art and the previous achievements that can be used as a starting point.

In an attempt to identify dependencies between these technologies as well as the candidate use cases that they will support and identify more technical requirements relevant to the project, the following subsections are included for each asset:

- **Technology asset description:** This subsection includes one photograph/picture and one use case diagram (maximum).
- **Relation to M-Sec:** This subsection provides a small presentation of the relation of the technology asset to the M-Sec project concept. In some cases, based on the input provided in **Section 2** as well as the corresponding deliverables of WP2, the use cases needs that this asset can cover are presented in more detail.
- **Requirements summary:** In this subsection the various functionalities that should be covered by the asset and its limitations (e.g. handling X amount of data per day) are presented.

It should be noted that the description of these assets will also be used as first input to Task “T3.2 M-Sec Architecture”. As it was explained in **Section 1**, this deliverable is a live document following an iterative approach and thus it is expected to be further enriched in its next version, probably with more assets that will be deemed necessary for the full coverage of the projects’ requirements and more details about how these assets are mapped (as solutions) to specific challenges.



3.2 Asset 1 [NII]: Security Analysis Tools

Technology asset description

NII brings in the project a Security Analysis Tool contributing to the Assessment of Business Processes by Checking Transaction Documents for Inconsistency Risks

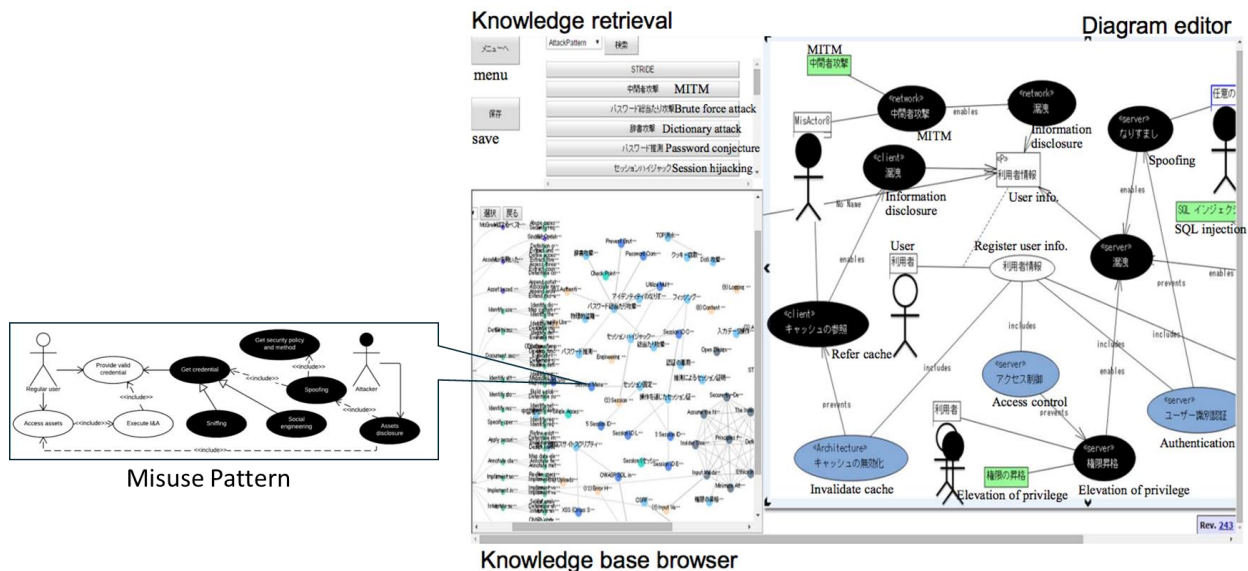


Figure 3—1: Security Analysis Tool

Relation to M-Sec

This asset contributes to:

- Engineering at a Multi-Layer Security and Privacy Analysis framework for hyper-connected smart cities.
- Security Risk assessment and Security Specification of use cases and M-Sec architecture

Requirements summary related to M-Sec

The requirement that this asset fulfils is:

- providing Security Analysis tool to build IoT applications

3.3 Asset 2 [NII]: Development Method for a secure service

Technology asset description

To carry out secure design, we propose an application to design software systems with verification of security patterns using model testing. Our method provides extended security patterns. Once developers specify threats and vulnerabilities in the target system, our method can verify whether the security patterns are properly applied and assess if the vulnerabilities are resolved.

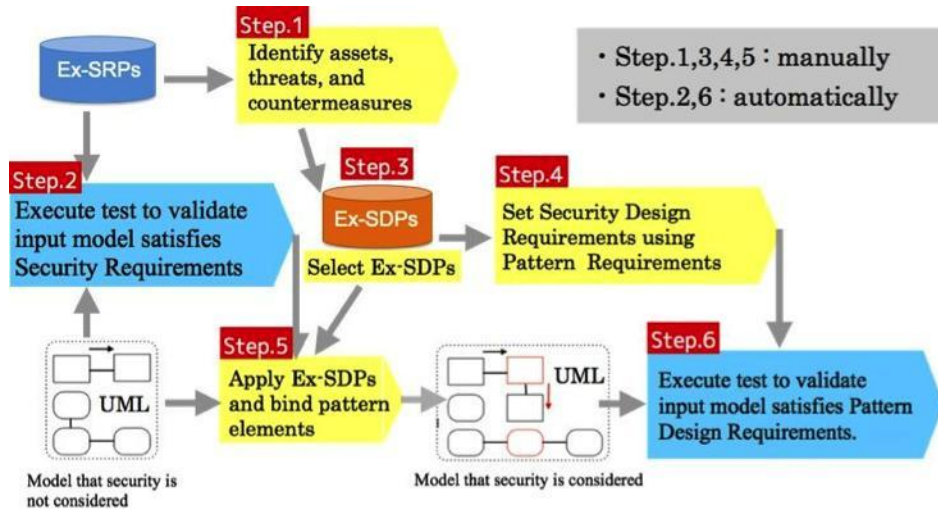


Figure 3—2: Process of the development method for a secure service

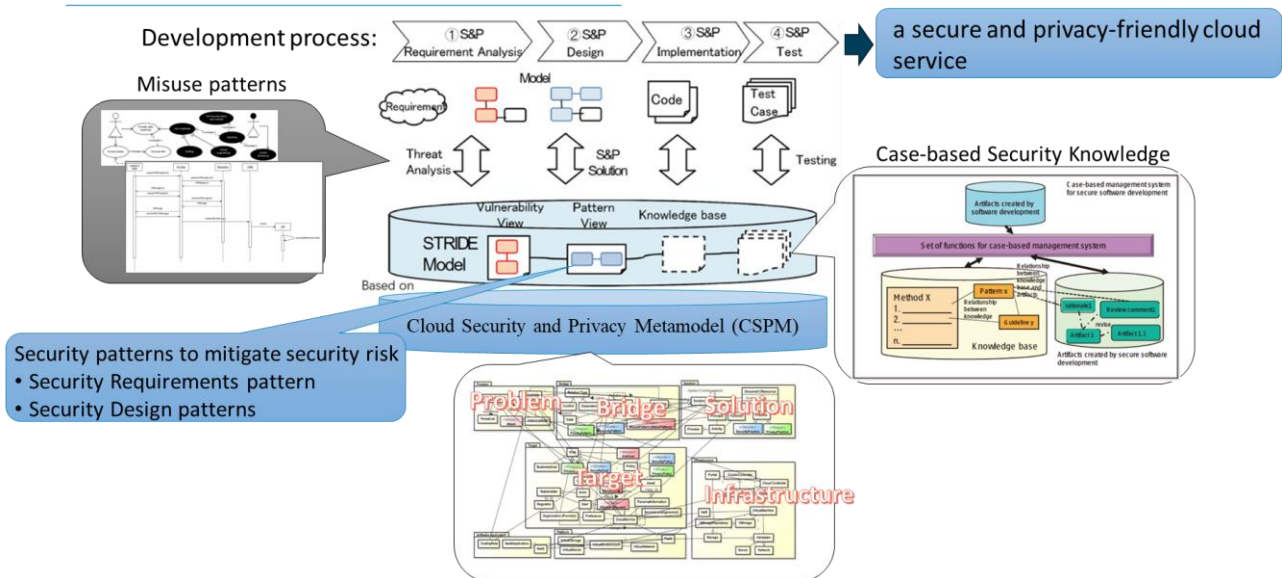


Figure 3—3: Development Process for a secure and privacy-friendly cloud service

Relation to M-Sec

Our development method contributes to engineering new levels of security and trust in large scale autonomous and trust-less multipurpose smart city platforms at the design level of M-Sec's architecture.

Requirements summary related to M-Sec

The requirements that this asset fulfils are:

- Providing a Development Process for a secure and privacy-friendly cloud service
- M-Sec platform needs to disseminate environment data stream to citizens securely
- M-Sec platform needs to protect the data streams from malicious attackers



3.4 Asset 3 [TST]: TSmarT platform & TST IoT devices

Technology asset description

TST can offer its TSmarT platform. TSmarT is a TST-owned modular wireless communications platform designed to facilitate the development and implementation of IoT monitoring and remote control applications. In addition, TST has an extensive background in designing and developing custom TST IoT devices. TST has a vast experience carrying out specific projects that require the design and development of those custom IoT devices that comprise of different sensors and make use of diverse communication technologies such as ZigBee, Wi-Fi, NB-IoT or BLE. Figure 3—4 below shows a couple of examples of the electronic design of the boards that serve as the foundation for the aforementioned IoT devices.

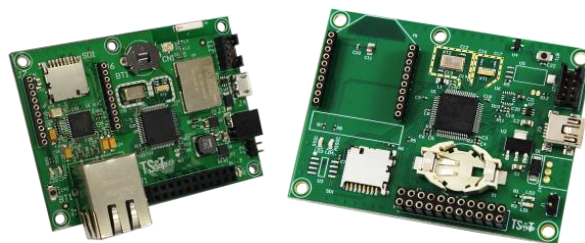


Figure 3—4: An example of TST's IoT boards

In order to provide data generated by these IoT devices, TST uses the MQTT protocol in them which simplifies the collection of sensor data, the publication of the different values obtained and the remote configuration of nodes.

Relation to M-Sec

TST IoT devices will be designed, developed and employed in Use Case 1, where they will be in charge of providing environmental information, such as temperature and noise levels. They will be deployed in emblematic parks and/or buildings which will provide a valid and useful test bed for the Municipality and the local services. In addition, other IoT devices will be specifically created to generate heat maps of specific city spots, selected in agreement with the City Council representatives.

Regarding Use Case 6, TST will rely on its software design and development capabilities working on multiple platforms. This way, TST developers will rely on their knowledge to develop an app in both Android and iOS Operating Systems (OSs), generating impact over a wider audience.

Requirements summary related to M-Sec

The aforementioned assets cover the following M-Sec requirements:

- IoT environmental sensing devices will be capable of sensing temperature, humidity, luminosity and noise.
- IoT devices will be capable of 'listening to' Bluetooth (BT) and/or Wi-Fi signals emitted by cell phones and register the corresponding MAC addresses to generate a 'heat map' reflecting the 'occupancy' at a specific city spot.



In their turn these assets have the following limitations:

- The IoT environmental sensing devices run on batteries. Their battery life will vary depending on the frequency (agreed between the involved parties) of sending data.
- IoT devices should implement a secure link to send data and avoid tampering or hacking.

3.5 Asset 4 [TST]: Santander Pace of the City

Technology asset description

The Pace of the City app was designed to report events within Santander city, including social, cultural or recreational events, as well as any incidences in the city such as malfunctioning of municipal services, reporting of damaged urban fixtures or traffic incidents. These latter events are managed by the Municipality through an internal app.

Within this context, the event life cycle is as follows (depicted in Figure 3—5):

- First, a citizen/visitor reports an event through the Pace of the City app, by choosing a category and subcategory and adding information such as the event date, a picture and the geo-location of the event.
- This event is then received by the Municipality, specifically by the Citizen Participation Service, through the internal application. This service is charged with classifying and sending the event to the specialised Municipality service.
- The designated municipal service should assign an operator within the service who will be charged with solving the reported incident.

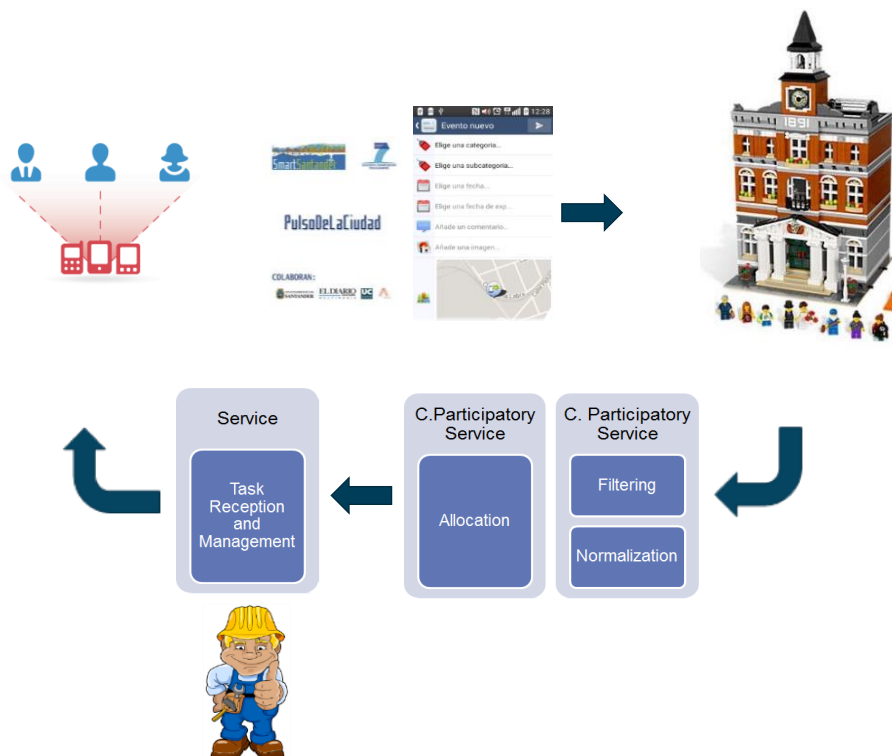


Figure 3—5: Pace of the City event cycle



All the events reported by the Pace of the City app and actions by the Municipality to solve incidences are accessible to all app users. Therefore, it is important to highlight this visibility in order to focus the municipal staff efforts correctly so as to ensure an efficient event management.

The activity related to this app was discontinued a few years ago even though it is considered a useful tool. Therefore, the goal now consists in putting it again into action, securing citizens confidence on it, integrating security features and including a novel set of functionalities, such as creating an interactive game among users in Spain and Japan, through this very same tool or making it converge with MinaRepo (see **subsection 3.12**), with both applications sharing the rewarding mechanisms and creating cross-border virtual competitions.

Relation to M-Sec

This topic is tightly related to Use Case 6 “Citizen as sensor”. TST will rely on its software design and development capabilities working on multiple platforms to create a novel app based on the old application features and functionalities, adapting it to today’s context and providing innovation in aspects such as security. This way, TST developers will rely on their knowledge to develop an app in both Android and iOS Operating Systems (Oss), generating an impact over a wider audience.

Requirements summary related to M-Sec

This assets covers the following M-Sec requirements:

- The Pace of the City server can store the data securely so that they are not disclosed to any party without permission.
- The Pace of the City application can provide and visualise information collected over the city.

3.6 Asset 5 [WLI]: Mobile Wallet

Technology asset description

In order to allow content consumers to buy access to media produced by content creators in a participatory way (without excluding but also without needing the media industry and their classic distribution channels), Mobile Wallet enables users to manage directly their transactions and get access to the content.

For doing so, the HTML5 app manages the entitlement that the user has with respect to content and cryptocurrency. The concept of Smart Contract has been used and the app is ready to use either Ethereum ether or an ad-hoc currency.

The app works without needing to store the user’s private keys in any other system, thus avoiding unlicensed transactions.

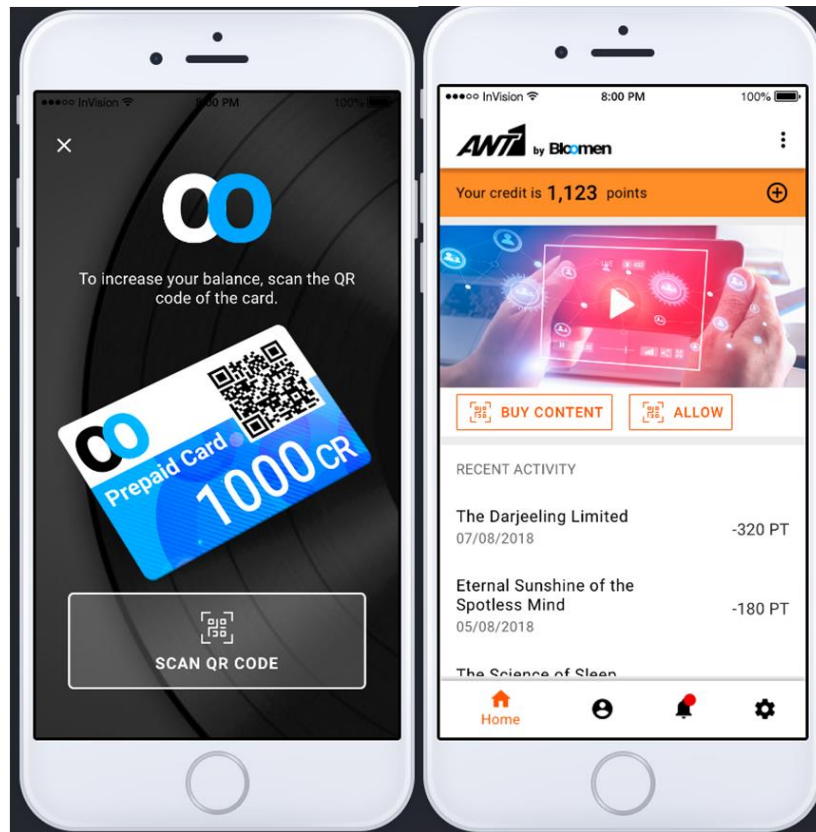


Figure 3—6: Mobile Wallet - adding cryptocurrency to the balance and getting entitlement for reproducing content

Relation to M-Sec

The same concepts presented above can be used in the context of getting access to health/wellbeing data of users. The current system employs direct grant of access in two cases: when the patient enables other users to get access to their information, and when the pairing of the sensors with the user profile takes place. However, this is not desirable, since a more decentralised approach would allow reaching new levels in the effective care of the people under the Use Case 2 “Healthcare & Wellbeing Tele-assistance for active and independent ageing people” umbrella.

Regarding the granting of access to individual profiles, it is hard for the user to manage their options in a one-by-one basis. Under some circumstances related to ageing condition or to digital literacy the user may not have the full mental map of the system. And even in the best case, it is cumbersome to manage the access this way, which provokes either a lack of filling the data or a ‘yes to all’ approach that compromises the user’s privacy. Adding Mobile Wallet capabilities to the system will allow the creation of a trust mechanism in which e.g. a Doctor of Medicine can access to the patients’ data using his/her private keys, getting the entitlement for accessing because the smart contract of the patient allows this possibility, and adding the access of data (not the actual data) to the ledger for audition purposes.

With regard to sensors, some of them can gather data from several users at the same time (e.g.: fall detector in a bathroom). It is hard to manage this directly with the current system that is designed to get measurements from personal sensors such as the heart rate sensor of a wrist band. Using the Mobile Wallet approach, every user can get the rights to get access to several sensors seamlessly.



Requirements summary related to M-Sec

Some of the requirements and limitations regarding the use of this asset from the project are the following:

- M-Sec should implement Smart Contracts in order to take advantage of this asset.
- M-Sec should have a notion of currency.
- Mobile Wallet does not guarantee real-time access to data¹.

3.7 Asset 6 [WLI]: Connected Assistance

Technology asset description

Worldline Connected Assistance is a networked care platform for patients and their caregivers (professional and informal) facilitating remote health monitoring and management to enhance independent living and a better quality of care at home.

This asset provides an IoT-centred business solution enabling patient health data self-management through the use of wearables and medical devices. Once captured, health data are securely transmitted to health providers for analysis and monitoring.

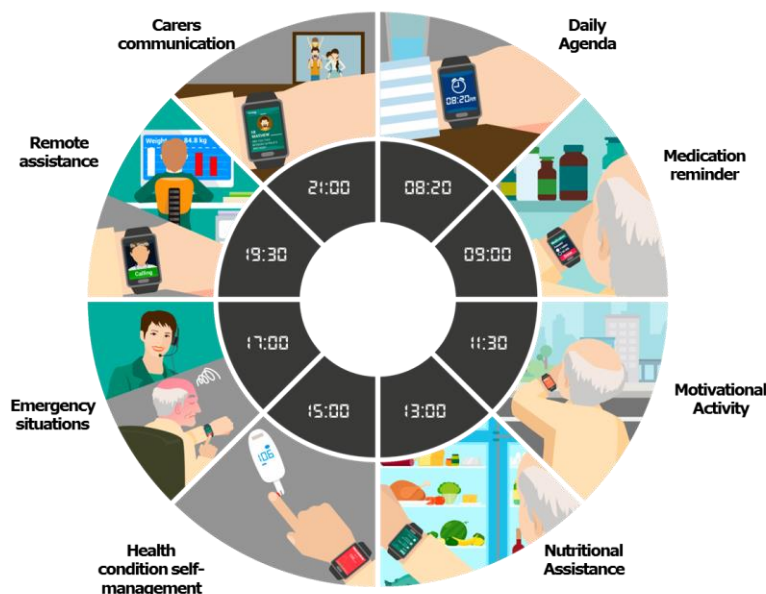


Figure 3—7: Storyboard connected assistance

Relation to M-Sec

Connected Assistance can be reused mainly within Use Case 2 “Home Monitoring & Wellbeing Tele-assistance for active and independent ageing people”. Worldline will use its Connected Assistance platform

¹ It depends on the Blockchain system employed. For example, Bitcoin creates a new block every 10 minutes, and Ethereum varies from 10 to 20 seconds.





as a starting point, a solution that already covers some issues related to health and wellbeing. Additionally, a range of functionalities will be added in order to offer a complete solution not only on terms of wellbeing monitoring but also by making ageing people to feel safe at home (through smart home sensors) and less isolated (through a video-call /chat tool and some push notifications regarding individual status and public activities organized by the City Council of Santander).

Requirements summary related to M-Sec

This asset presents the following requirements/limitations:

- It does not contain end-to-end security mechanism. This limitation should be resolved by M-Sec security mechanism.
- It does not comply with current law of GDPR. A mechanism regarding right of deletion, right to be forgotten and so on should be implemented.
- It does not provide a secure storage of data collected.
- It does not provide secure communication for video call/call.

3.8 Asset 7 [WLI]: Balena.io (FKA resin.io)

Technology asset description

Balena is a mechanism used to update all of the code in external devices at the same time. The devices can range from smart TVs to drones as long as they are able to install a dedicated operating system (a minimal OS but with Docker-like capabilities). WLI has experience with resin.io, the project that became balena.io.

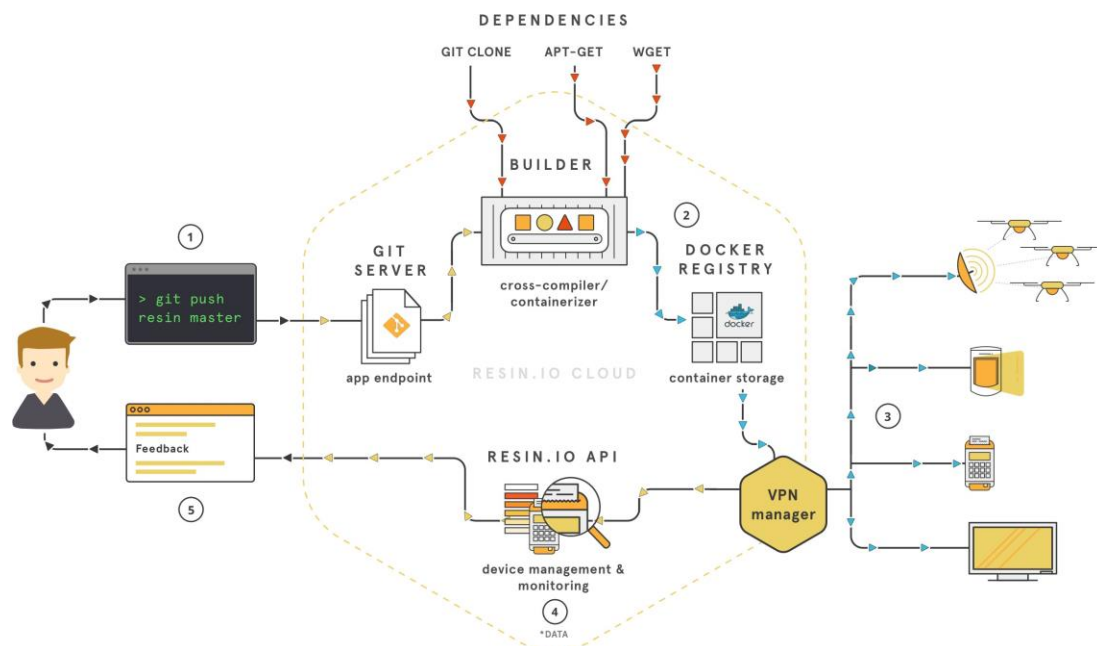


Figure 3—8: Resin.io workflow (balena.io)²

² Ryan M Harrison <https://hackernoon.com/im-a-resin-io-fanboy-and-you-should-be-too-2b70e90b0710>



Relation to M-Sec

Several Use Cases are planning to deploy a series of IoT devices to act as sensors. Once the devices are deployed, updating the code can be a challenge. With this asset, the complexity of updating is reduced.

Requirements summary related to M-Sec

This asset presents the following requirements/limitations:

- The devices need to install balena.io.
- The devices need some connectivity for their code to be updated.

3.9 Asset 8 [WLI]: Node-RED

Technology asset description

Node-RED is a flow editor that compiles directly in Raspberry Pi and is able to cross-compile in Arduino. It generates node.js code. WLI has worked with this editor during the agile-iot project³.

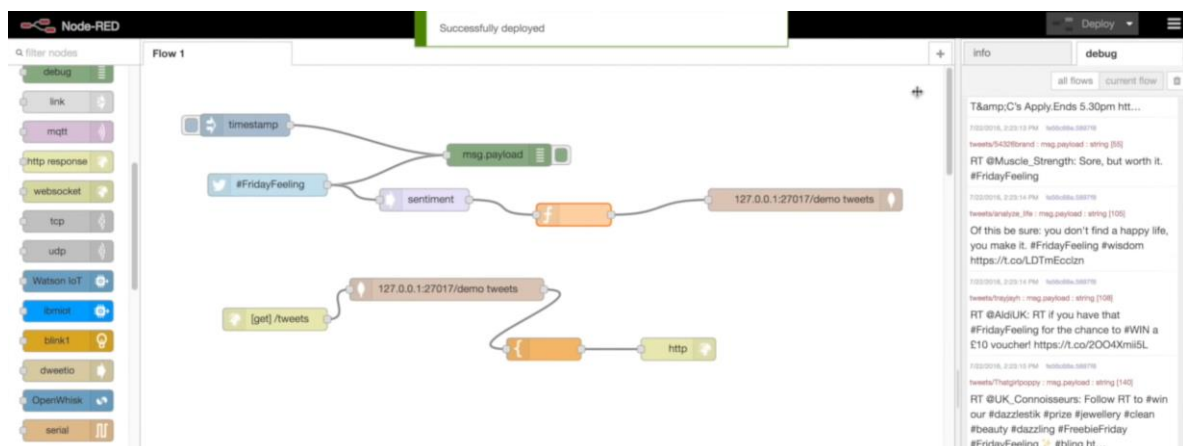


Figure 3—9: Flow editing in Node-RED⁴

Relation to M-Sec

Several Use Cases are planning to deploy a series of IoT devices to act as sensors. This flow editor helps in the creation of Rapid Prototypes, thus speeding up the development cycle.

Requirements summary related to M-Sec

This asset presents the following requirement/limitation:

- The IoT devices should support node.js. Alternatively, the devices can support cross-compiling.

³ AGILE (Adaptive Gateways for dIverse muLTiple Environments) <http://agile-iot.eu/>

⁴ IBM Node-RED team, https://www.youtube.com/watch?time_continue=197&v=vYreeoCoQPI





3.10 Asset 9 [KEIO]: KEIO Mobile Sensing Platform

Technology asset description

The mobile sensing platform is capable of sensing acceleration, angular velocity, humidity, temperature, atmospheric pressure, PM2.5, UV-A, and location. It consists of about 70 garbage collection trucks in Fujisawa, on each of which a sensor box is mounted. Among them, PM2.5, UV-A, temperature, humidity, and atmospheric pressure are valuable for citizens to monitor the environment, while acceleration and angular velocity are valuable for local government to monitor road surface. In addition to these sensors, all the trucks have two cameras, one in its front and the other in its rear. These cameras can be used to visually monitor road surfaces and the amount of collected garbage. The trucks cover the whole city; they visit all the houses in the city since in this city garbage collection is operated on a house-by-house basis, instead of a garbage collection stations one.

Sensor data are transmitted at most 100 times/second using XMPP to acquire temporarily dense sensor data. The data streams are handled at SOXFire (next subsection) server based on a publish/subscribe policy. A sensor on a truck publishes the sensor data stream to a virtual sensor node configured in the server, and the stream is routed to entities (e.g. applications and services) that have subscribed to this virtual sensor node. This platform is a distributed system consisting of edge computers (sensors) and a cloud system (SOXFire server).



Figure 3—10: Automotive Sensing Trucks in Fujisawa City

Relation to M-Sec

The mobile sensing platform plays a critical role in the M-Sec project due to its sensing capability. Firstly, Use Case 3 "Secure and Trustworthy Urban Environment Monitoring with Automotive, Participatory, and Virtual Sensing Techniques" requires sensors that capture the city environment. Since the platform contains a number of different sensors to acquire city data, this requirement is well-covered by this platform. Secondly, the platform's capability of transmitting at most 100 sensor data in a second suites the requirement for live sensor data stream in Use Case 3. Such live sensor data streams are transferred to citizens to enable them to optimise their behaviour. Thirdly, in Use Case 4 "Secure and Trustworthy Hyper-connected Citizen Care", monitoring citizens urban waste generation becomes possible with this asset. The rear camera captures garbage bags to be collected. This enables garbage bag counting by a garbage bag counting mechanism



included in this platform, which can in turn be used for estimating the urban waste generation in a fine-grained way.



Figure 3—11: Screenshot of garbage bag counting mechanism

Requirements summary related to M-Sec

The aforementioned asset covers the following M-Sec requirements:

- The platform can estimate the amount of urban waste generation in a fine-grained way using images captured by camera.
- The platform can collect environment sensor data. Also, it can deliver the environment data stream to citizens.

However, the asset has the following limitations:

- The system may harm citizens' privacy, since the images may contain people walking in the city. This limitation should be resolved by the M-Sec privacy mechanism.
- The sensing system in this platform does not have any security support. This limitation should be resolved by M-Sec security mechanism.

3.11 Asset 10 [KEIO]: KEIO SOX

Technology asset description

Sensor over XMPP (SOX) is a distributed platform that enables scalable collection and dissemination of real world data with standardised metadata using a publish/subscribe method. The technology is open at the website of KEIO and it has been in operation by KEIO for years. As depicted in Figure 3—12, SOX can handle heterogeneous sensor data streams. It has also been clarified by research from KEIO that a SOX server can handle at least 10,000 sensor data streams concurrently. Opposing to the sensing side is the applications side where a number of services and applications consume those differing sensor data streams. Considering the existence of data stream producers and consumers between which connections are established in unpredictable and dynamic timings, the platform needs to lower mutual dependency between them. To do so, SOX is based on a publish/subscribe mechanism. The sensors publish data streams towards





corresponding virtual sensor nodes in a SOX server, while the applications subscribe to virtual sensor nodes relevant to their purpose. Publishers and subscribers are indirectly connected via SOX while staying mutually independent.

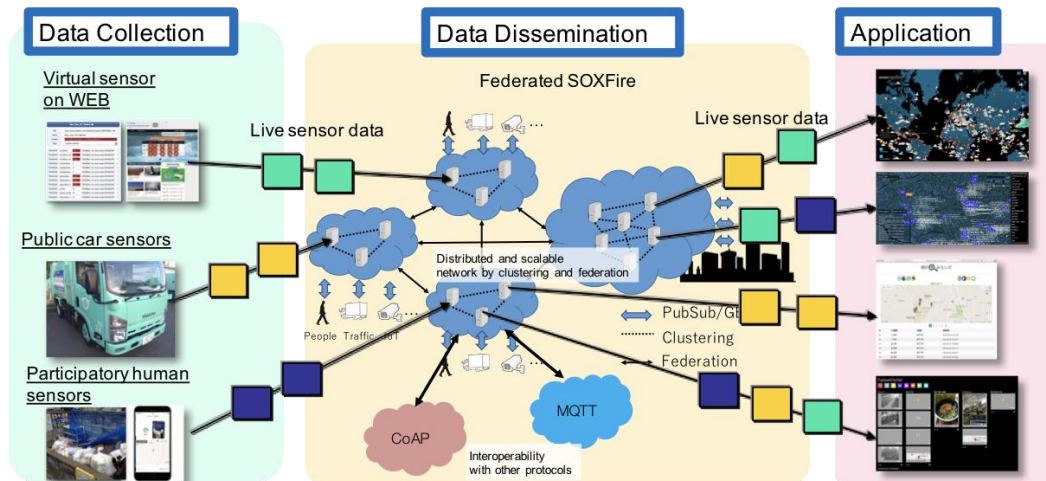


Figure 3—12: Overview of Sensor over XMPP platform

Relation to M-Sec

SOX plays a major role in the M-Sec project since it provides an infrastructural data dissemination platform. More specifically, in Use Cases 3 and 4, a number of sensor data streams of different types are supposed to be produced and consumed by citizens and local government officers. The connection between these producers and consumers is arranged by SOX. Also, since SOX has a user accounting mechanism, it can allow users with a valid right to access virtual sensor nodes. On the other hand (and not surprisingly), access requests from users without a valid right are denied.

However, the major limitation of SOX is that it does not provide end-to-end data security. Data streams are not encrypted, thus can be sniffed along the link between an edge and the SOX server. Even if the connection between them is secured by the use of, for example, Transport Layer Security (TLS), the data are in the form unciphered text in the server. This means that if the server is hacked, the data may be stolen. These limitations should be tackled by the M-Sec project.

Requirements summary related to M-Sec

The aforementioned asset covers the following M-Sec requirements:

- SOX can deliver data streams to citizens.
- SOX can handle at many streams of different types concurrently.
- SOX can deliver data streams real-time.

However, the asset has the following limitation:

- SOX does not contain an end-to-end security mechanism. This limitation should be resolved by M-Sec security mechanism.



3.12 Asset 11 [KEIO]: Fujisawa MinaRepo

Technology asset description

Fujisawa MinaRepo is a participatory sensing platform that leverages human sense to acquire information from the real world. Currently, it is operated in Fujisawa city with 50 city officers as participants. The data include images, the location, and the metadata of city problems like illegal garbage, graffiti, potholes, etc. It has been in operation since October 2017, and till now over 6,000 reports, including the aforementioned events, have been submitted by the participants. Fujisawa MinaRepo consists of two major components. One is MinaRepo App, implemented as a smartphone (iOS/Android) native application and a web application. Participants can report their findings using one of these applications. The reports are delivered via SOX to the other major component, which is MinaRepo server. The MinaRepo server is a set of a front-end server process and a backend database. The front-end is developed as a web server that opens an API against the Internet. The backend database stores all the reports and allows clients to retrieve reports selectively.

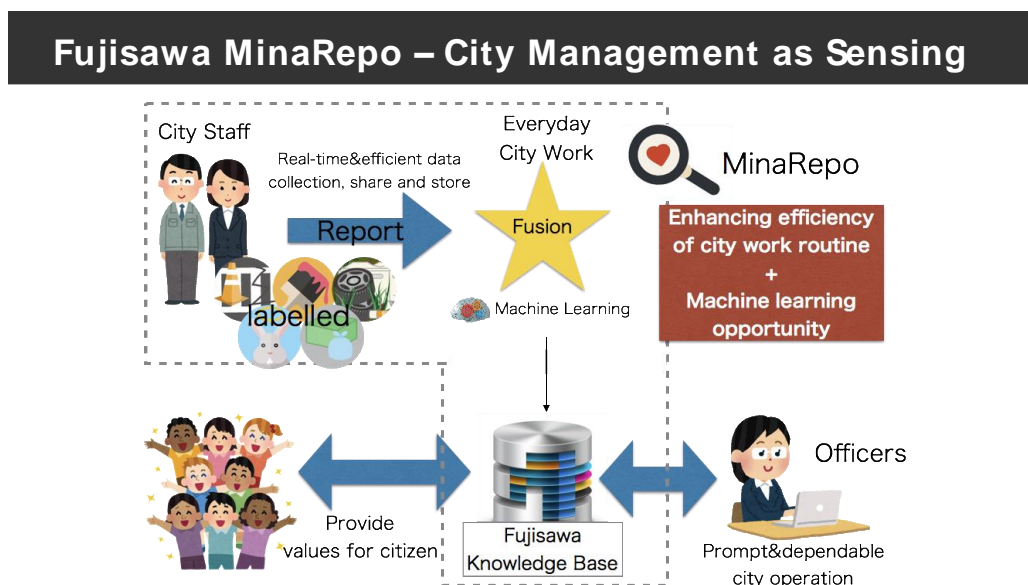


Figure 3—13: Overview of Fujisawa MinaRepo

Relation to M-Sec

Use Case 6 “Citizens as sensor” is the major user of this asset in the project. MinaRepo allows citizens who wish to volunteer to send information, make use of any of the categories (beaches, parks and gardens, transport, public roads, culture, sports, etc.), insert images, and include comments, date of the event, expiration date, etc. The events, under the responsibility of the Municipal Services of the Town Hall, are sent to the Town Hall as incidences for their resolution, enabling the citizens to know their state at **any moment**.

Information can be sent anonymously (thus not requiring a user registration process requesting personal data) or through the employment of a virtual ID created specifically for this service. In the latter case, a rewarding mechanism will be created to grant prizes to the most participatory citizens, considering this is a proper means to incentivise participation.



Requirements summary related to M-Sec

This asset covers the following M-Sec requirements:

- The MinaRepo server can store the data securely so that they are not disclosed to any party without permission.
- The MinaRepo application can provide and visualise information collected over the city.

3.13 Asset 12 [YNU]: YNU Honeypot (IoTPOT)

Technology asset description

Due to the rapid growth of various devices capable of communicating through the internet, the playfield for attacking such devices has also exponentially grown. The “Mirai” malware attack (2016)⁵ using compromised Internet of Things (IoT) devices has shown the world how vulnerable IoT devices are, and how malicious intent can be used to disrupt businesses of large enterprises on the Internet. The security professionals use a computer system that analyses various attack patterns by attracting malicious entities to attack devices placed in an isolated environment. Such a computer system is called “Honeypot”. By examining the attack methods, we can come up with appropriate detection and protection mechanisms. In order to understand various attack vectors and malicious codes, a YNU asset of honeypot (IoTPOT) will be utilised. Its key purpose will be as follows:

- Obtaining knowledge on the attack vectors of IoT devices, such as IP cameras and sensors.
- Using this knowledge to protect the IoT devices.
- Enabling Intrusion Detection System (IDS) to detect and report attacks.

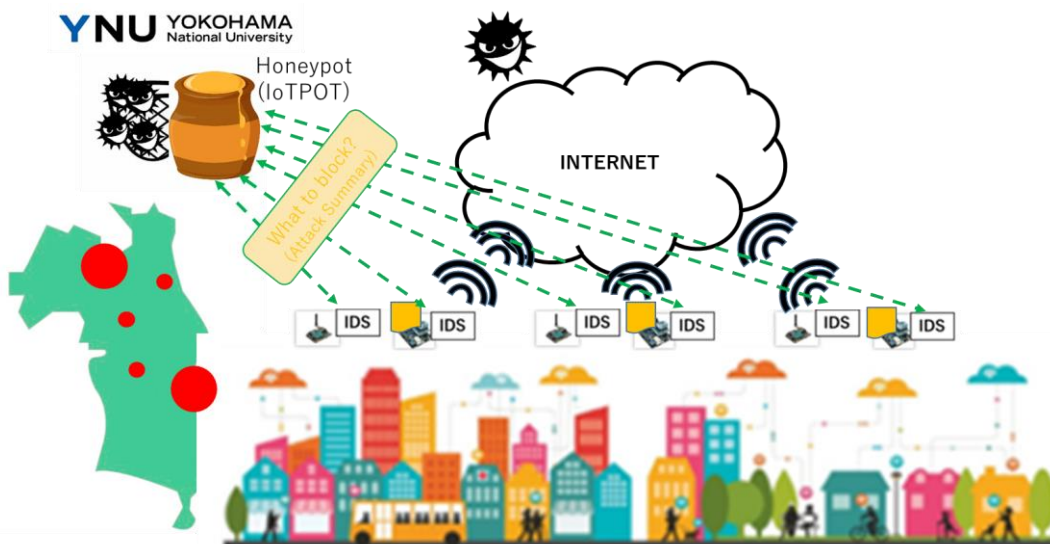


Figure 3—14: IoT Devices Security

⁵ <https://www.theguardian.com/technology/2016/oct/26/ddos-attack-dyn-mirai-botnet>



Relation to M-Sec

This asset will be used for obtaining attack vectors needed in setting up information security protection for IoT devices (sensors and cameras) used in Use Case 3 “Secure and Trustworthy Urban Environment Monitoring with Automotive, Participatory, and Virtual Sensing Techniques” and Use Case 4 “Secure and Trustworthy Hyper-connected Citizen Care”. This asset will help in designing the IoT data security for the IoT devices and end-to-end protection.

Requirements summary related to M-Sec

This asset covers the following M-Sec requirement:

- IoTPOT will allow us to capture latest attack vectors for finding appropriate detection solution in IDS.

A limitation of this asset is the following:

- The data security solution will depend upon the available resources in the IoT gateway devices.

3.14 Asset 13 [ICCS]: Blockchain framework

Technology asset description

The main technical asset that will be used as the blockchain implementation is the Hyperledger framework. Below is a detailed diagram, which describes the different Hyperledger components and the interactions between them.

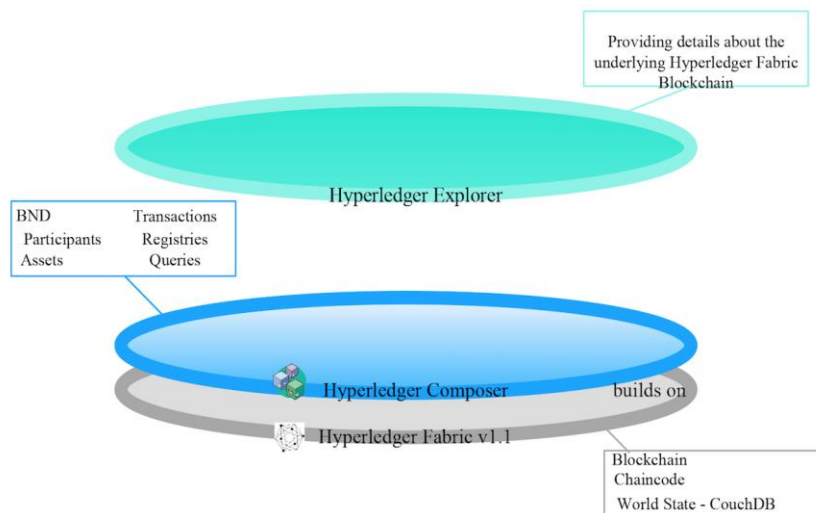


Figure 3—15: Hyperledger component diagram

In M-Sec three basic Hyperledger components will be used:

- Hyperledger Fabric, providing the permissioned blockchain environment.
- Hyperledger Composer, providing RESTful API.
- Hyperledger Explorer, providing visualized information about the blockchain.



Hyperledger Composer is an extensive, open development toolset and framework that supports the underlying Hyperledger Fabric v1.0.4 blockchain infrastructure and runtime. In particular, it provides the way to quickly model the demo business network starting from the business level by defining the corresponding assets, administration rights, the participants and their permissions and finally the transactions. It is of vital importance that Hyperledger Composer exposes a REST Server API to invoke transactions that create, delete and update assets and transfer them between participants on the underlying Hyperledger blockchain. In general, Hyperledger Composer functions as the middleware service between the Demo application and the Hyperledger Fabric blockchain.

Hyperledger Composer REST server		
org_acme_sample_BuyNewsTransaction : A transaction named BuyNewsTransaction	Show/Hide	List Operations Expand Operations
org_acme_sample_ChangeUrl : A transaction named ChangeUrl	Show/Hide	List Operations Expand Operations
org_acme_sample_Credit : A transaction named Credit	Show/Hide	List Operations Expand Operations
org_acme_sample_Newsdata : An asset named Newsdata	Show/Hide	List Operations Expand Operations
org_acme_sample_Payment : A transaction named Payment	Show/Hide	List Operations Expand Operations
org_acme_sample_PublishTransaction : A transaction named PublishTransaction	Show/Hide	List Operations Expand Operations
org_acme_sample_User : A participant named User	Show/Hide	List Operations Expand Operations
GET /org.acme.sample.User	Find all instances of the model matched by filter from the data source.	
POST /org.acme.sample.User	Create a new instance of the model and persist it into the data source.	
GET /org.acme.sample.User/{id}	Find a model instance by {id} from the data source.	
HEAD /org.acme.sample.User/{id}	Check whether a model instance exists in the data source.	
PUT /org.acme.sample.User/{id}	Replace attributes for a model instance and persist it into the data source.	
DELETE /org.acme.sample.User/{id}	Delete a model instance by {id} from the data source.	
Query : Named queries	Show/Hide	List Operations Expand Operations
System : General business network methods	Show/Hide	List Operations Expand Operations

Figure 3—16: Hyperledger Composer REST server

The **Hyperledger Explorer** component works on top of the underlying blockchain data structure and provides a visualized way of the information stored in it. The provided details are the following:

- Overall information related to a channel, like the number of peers, blocks, transactions and chaincode.
- Specific block information.
- Peer list.
- Chaincode list.
- Transaction information.



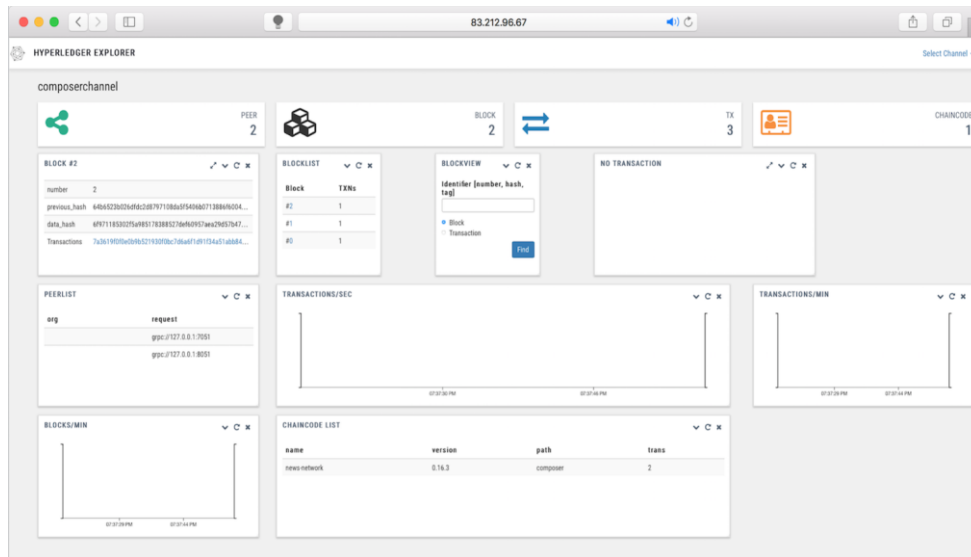


Figure 3—17: Hyperledger Explorer User Interface

Relation to M-Sec

The blockchain is an authenticated data structure forming a long ledger of transactions shared by participants of the network. A full copy of the blockchain at any time contains all records of every transaction ever completed in the network. All participants in the blockchain can maintain their own copy of this ledger of transactions, though ideally, the amount of data stored would vary based on capability, need and preference. Every block contains a hash of the previous block. This enables the blocks to be traced back even to the first, the genesis block. Cryptography is used to verify transactions and keep information on the blockchain private. Blocks are generated by a consumption of a scarce resource in a process called mining. Mining allows nodes to reach **a secure, tamper-resistant consensus**. Mining is also the mechanism used to introduce new digital coins or tokens of transaction into the system. There are various mining algorithms, primarily based on Proof of Stake or Proof of Work approaches. In M-Sec, a Proof of Work mining algorithm will be applied but without the escalating difficulty in the mining process which is introduced to restrict the issuance of currency (a tactic used to avoid inflation problems). For instance, blockchain can be used in M-Sec to prevent distributed denial of service (DDoS) attacks by implementing a cryptographic currency system that would be consumed as a "fee" in each transaction. Each element of the system would have access to a credit line that would allow it to operate within the system and could always increase the balance if necessary. This type of solutions makes it unfeasible for an attacker to put the system at risk by generating an unlimited number of transactions.

Moreover, M-Sec will explore the implementation of a novel **marketplace where IoT devices can exchange information and services through the use of virtual currencies**, allowing real-time matching of "supply and demand" (meaning "supply of IoT device capabilities" and "demand of IoT device capabilities by citizens and other smart city stakeholders"), thus enabling the creation of liquid markets with profitable business models of the smart city stakeholders.

The core peer-to-peer architecture in M-Sec will be formed by the decentralised edge nodes which are high-end devices (e.g. servers) with significant computational and storage capabilities, capable of hosting



marketplaces. A marketplace would potentially require payment services, analytics solutions, demand supply matching solutions, etc. These peer nodes are also potential repositories for a complete copy of the blockchain and would provide blockchain analytical services. The size of blockchains can rapidly increase in a world where every city or community may have millions or hundreds of millions of IoT devices. With the blockchain being the trusted source of information pertaining to all transactions, it is important to be able to access it at a regional or community level going back in time, in some cases from the start of transaction history.

Requirements summary related to M-Sec

The following generic requirements have been identified for the blockchain platform and technical framework:

- **Permissioned blockchain for privacy** in order to hide an organisation's blockchain network activity from the public (i.e. unauthorised parties). In permissioned (or private) blockchains, only authorised members can enter the blockchain network.
- **Smart contract support** in order to automatically execute transactions (i.e. without intermediate's approval), apply specific execution and functionality under given conditions and apply other automated procedures, etc. Smart contracts are an interesting application over blockchains that can enable high levels of automation and support for decentralised apps.
- **Transactions per second (TPS)** should be from 10-15 to 1,000-2,000 in order to serve multiple requests at the same time. The platform should scale that much to process all kinds of transactions, such as simple transactions, simple smart contracts or many that call one another, etc.
- **Single profile/account with digital wallet** available to every participant (e.g. end-users, companies, website owners, content providers, etc.) in order to be able to make purchases and payments and participate in an ecosystem of value exchange over IoT infrastructures in smart cities, etc.
- **Anonymous Identity (ID)** for every party that wants to hide their real/private information from transaction tracing (not all Use Cases need global implementation). In order to provide anonymity, the blockchain platform should be aware of the private information of a party, providing an anonymous ID and ensuring the party's transactions take place only through this without revealing any other information in the permissioned blockchain network.
- **Mining algorithm** in our platform in order to provide incentive for the end users to be active and not malicious in the blockchain network.
- **Privacy and anonymisation requirements:** Privacy by design is an essential implementation principle of the project; user private data will not be revealed outside the blockchain network as the latter will be permissioned. Additionally, in order to anonymise the user identity, external certificate authorities might be deployed (such as public bodies) who can verify the true identity of the user (particularly useful for the Know Your Customer procedures⁶) and provide them a user hash-ID which will be the only identification used to interact further with the system. Off-chain data storage and linking with on-chain transactions will be needed to increase the privacy notion within M-Sec.

⁶ <http://business.rediff.com/report/2010/oct/18/perfin-why-kyc-is-mandatory-now.htm>





3.15 Asset 14 [ICCS]: Quorum Blockchain framework

Technology asset description

A technical asset that could be used as a blockchain implementation is the Quorum Platform⁷. Quorum supports private transactions within a permission group of known participants. It is an open source platform, GPL/LGPL licensed just like Ethereum, supported by a growing community of users and developers. Since it is designed to develop and evolve alongside Ethereum, it is able to incorporate the majority of Ethereum updates quickly and seamlessly. It is actually a fork of go-ethereum⁸ and it is updated in line with go-ethereum releases.

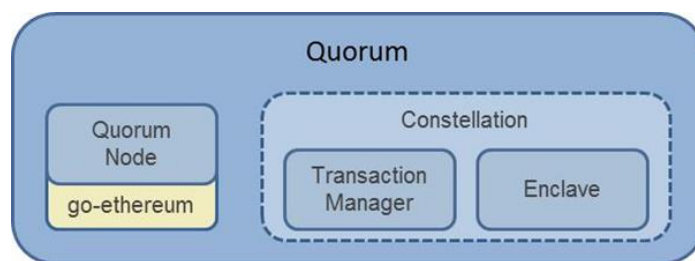


Figure 3—18: Overview of Quorum Framework

Some features of Quorum over go-ethereum, important for the M-Sec project are:

- **Privacy:** Quorum supports private transactions and private contracts through public/private state separation, and utilises peer-to-peer encrypted message exchanges for directed transfer of private data to network participants.
- **Alternative Consensus Mechanisms:** since Quorum is a permissioned network, it offers more choices for the consensus protocols besides Proof-of-Work and Proof-of-Stake such as: Raft-based Consensus and IBFT.
- **Permissioned Network:** Quorum is a permissioned network and only known parties can join the network.
- **Higher performance:** It leverages the work that the Ethereum developer community has undertaken and achieves higher performance.

Quorum consists of the following components:

- **Quorum Node:** it is designed to be a lightweight fork of geth and includes modifications to geth among which:
 - Consensus is achieved with the Raft or IBFT consensus algorithms instead of using Proof-of-Work.
 - It is permissioned and it supports private transactions.
 - Transaction creation has been modified to allow for transaction data to be replaced by encrypted hashes in order to preserve private data where required.

⁷ <https://www.jpmorgan.com/global/Quorum>

⁸ <https://github.com/ethereum/go-ethereum>





- **Constellation/Tessera - Transaction Manager:** It is responsible for Transaction privacy. It stores and allows access to encrypted transaction data, exchanges encrypted payloads with other participant's Transaction Managers but does not have access to any sensitive private keys. It utilises the Enclave for cryptographic functionality (although the Enclave can optionally be hosted by the Transaction Manager itself). The Transaction Manager is restful/stateless and can be load balanced easily.
- **Constellation/Tessera – Enclave:** Distributed Ledger protocols typically leverage cryptographic techniques for transaction authenticity, participant authentication, and historical data preservation (i.e. through a chain of cryptographically hashed data.) In order to achieve a separation of concerns as well as to provide performance improvements through parallelisation of certain crypto-operations, much of the cryptographic work including symmetric key generation and data encryption/decryption is delegated to the Enclave. The Enclave works hand in hand with the Transaction Manager to strengthen privacy by managing the encryption/decryption in an isolated way, holds private keys and is isolated from other components.

Below is a detailed diagram, which describes how Transaction Processing works, introducing the notion of Public Transactions and Private Transactions, extending Ethereum Transaction Model which now includes new parameters.

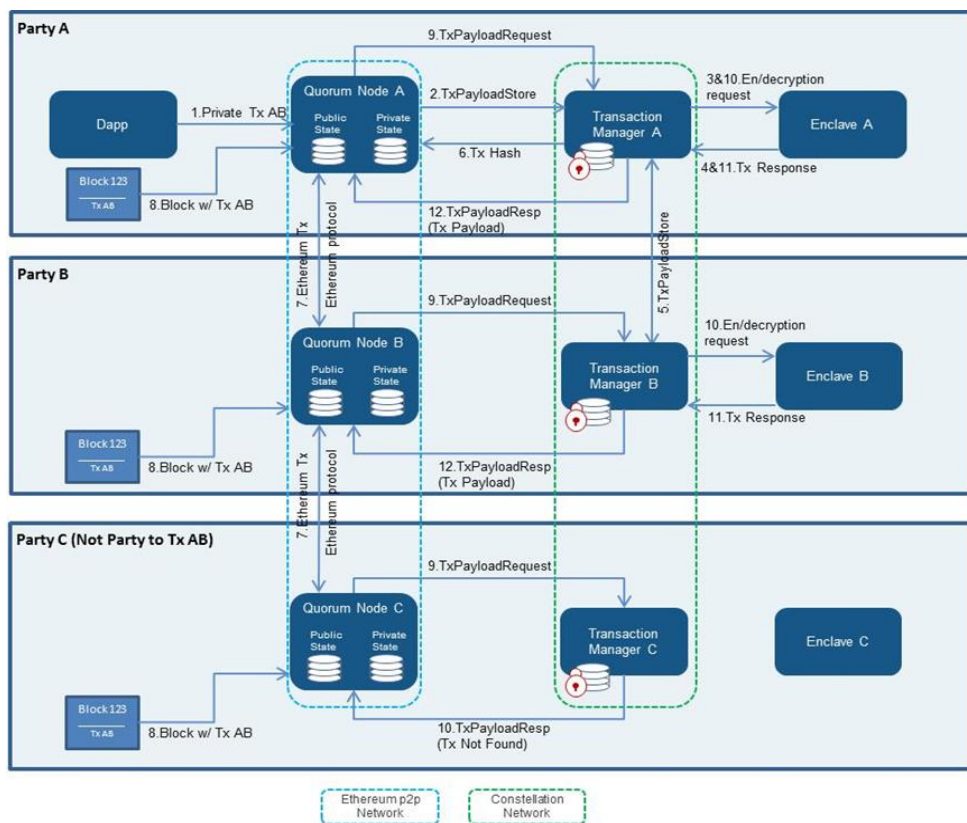


Figure 3—19: Private Transaction Flow in Quorum



Figure 3—20: Example of application of the MTSA to a reactive system

Relation to M-Sec

The MTSA helps in designing behaviour specifications in each Use Case. For example, considering the home activity monitoring system in Use Case 2, there may exist the requirement that the system can sense the activity of elderly citizen in the house. MTSA can synthesise the correct specification of switching sensors through modelling the interactions between the sensors and the environment and formalising the requirements for each sensor.

Requirements summary related to M-Sec

The asset has the following limitations:

- It is required to model the reactivity of an environment in a labelled transition system. It is also needed to distinguish the actions in the model into controllable or uncontrollable ones.
- The MTSA cannot address non-functional requirements such as “X must happen in Y seconds”.
- The MTSA is based on discrete event systems. Therefore it is required to translate continuous values like sensing data into discrete events.
- MTSA cannot synthesise a controller if requirements are too strict. In that case, the requirements or assumptions on the environment should be relaxed.

3.17 Asset 16 [WU]: Runtime Environment Model Updater (REMU)

Technology asset description

REMU is a tool to automatically update an environment model efficiently, accurately and at runtime in response to changes in the environment. Reactivity of an environment may change at runtime. This may cause inconsistency with an environment model and the environment and a formal guarantee of correctness of a behaviour specification may be broken. REMU uses a difference learning technique to reflect environmental changes to environment model accurately, in fast settling time, and with low computational overhead. REMU is designed to be used with MTSA. Then, REMU takes an environment model in the form of a labelled transition system (LTS) and execution traces of events used in the LTS.

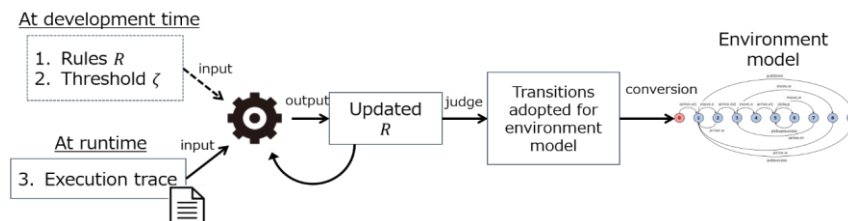


Figure 3—21: High-level view of REMU



Relation to M-Sec

In some Use Cases, a system may operate under a changing environment. REMU enables the system itself to reflect the changes in the environment model to achieve a new correct behaviour specification by using MTSA with the updated environment model.

Requirements summary related to M-Sec

The asset has the following capabilities/limitations:

- REMU can update environment model at runtime.
- REMU is designed to be used with MTSA.
- This technique needs the reactivity of an environment to be modelled in the form of Labelled Transition System.
- REMU takes execution traces in the form of a sequence of events.
- REMU requires to model possible reactivity as rules, which should be determined in the development time.



3.18 Asset 17 [CEA]: Secured components for devices and gateways

Technology asset description

Secure components are a common countermeasure for preventing hardware attacks from connected products. These components store sensitive information and make sensitive operations in an area protected from direct attacks, such as side-channel or fault injections.

CEA has developed a toolbox for integrating various forms of such secure components into new or existing platforms. The goal is to provide the following functions on IoT systems:

- Storing sensitive information such as private SSH keys, disk decryption key, passwords, pin numbers and others.
- Performing sensitive operation in a more secured way, such as hash functions or random number generation.
- Verifying the integrity of a system by monitoring the bootchain and state of the device.

	
<i>Secure component in the form of a trusted anchor on a USB device for securing legacy</i>	<i>Secure component located on an extension (in white on the picture) for a computer type Raspberry PI 3 for a</i>



<i>devices.</i>	<i>check boot and an encryption of the SD card.</i>
-----------------	---

Figure 3—22: Examples of Secure Components

The toolbox associated with these components consists of tools for provisioning components for certificate initialisation and the possible association with a PKI, the initialisation of standardised interfaces such as PKCS11⁹ as well as tools for the verification of integrity when booting and attaching removable media to platform instances.

Secure components play a vital role in the project because they are the primitive low-level security functions that will be propagated to create trust over applications. In addition to the use of securing products, secure components offer an authentication capability with applications and blockchain, which can be likened to a hardware wallet.

Relation to M-Sec

In the M-Sec project, this asset can be used to link IoT device security to cloud-based applications to authenticate cloud data from their sources. With the use of a blockchain, these components have the ability to serve as hardware wallet on the one hand, and can also affirm the role of oracle gateways or some IoT device. In addition, security primitives make it possible to assert and reinforce the privacy of data, in particular by managing the encryption of data and communications.

Requirements summary related to M-Sec

The asset has the following capabilities:

- It provides encryption engine, hashing engine, true random number generator and secure storage.
- It is compatible with U-boot, Linux Kernel and OpenSSL.
- It enables PKCS11 and X509.

Regarding the asset's limitations:

- Hardware ideally requires an on-board integration (SPI or USB).
- Firmware and OS should be compatible with security.

3.19 Asset 18 [CEA]: Eclipse sensiNact platform and Studio

Technology asset description

The Eclipse sensiNact project consists of a software platform enabling the collection, processing and redistribution of any data relevant to improving the quality of life of urban citizens, and programming of interfaces allowing different modes of access to data (on-demand, periodic, historic, etc.) and application development and deployment to easily and rapidly build innovative applications on top of the platform.

⁹ <https://www.oasis-open.org/standards#pkcs11-base-v2.40>

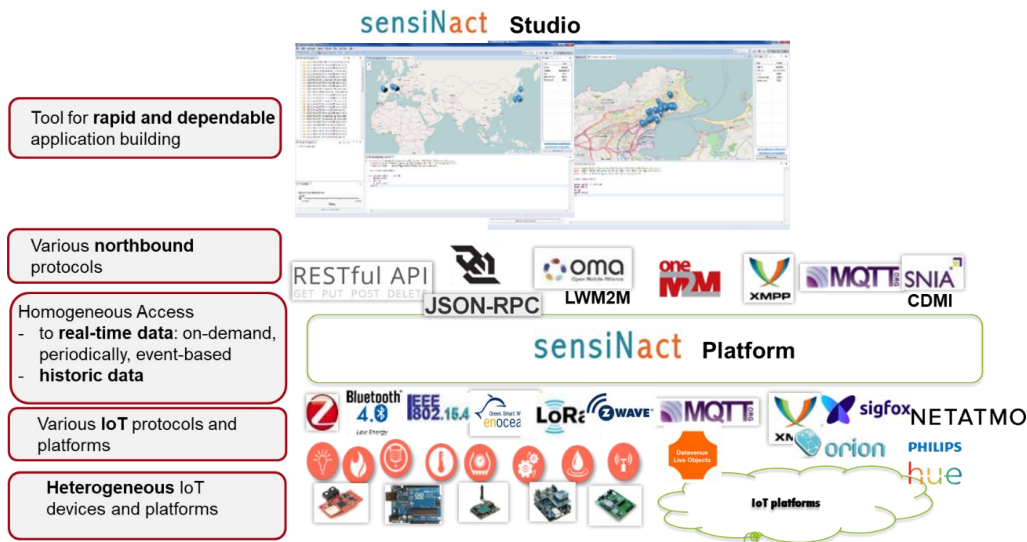


Figure 3—23: Overview of sensiNact Platform and Studio

At the heart of sensiNact lies its service-oriented approach in which IoT devices expose their functionalities in terms of services (temperature service, presence detection service, air quality monitoring service, alarm service, etc.). Each service then exposes one or several resources such as sensor data or actions. Thus, building applications becomes a matter of composing sensing services with actuation services. Loose coupling between the devices and the services they implement makes the composition of services more dynamic and adaptable to the changing context, not only in the software environment (increasing CPU or memory usage, low battery, reducing quality of measures, etc.) but also in the physical environment (replacing sensors, changing localization, etc.).

Relation to M-Sec

The sensiNact platform deals with one of the main today's issues in every urban environment: uninteroperability of heterogeneous data sources and protocols to access them. Emerging IoT devices, legacy systems, increasing number of social networks, mobile applications, open data repositories and web data are the potential exploitable data sources in urban environments. sensiNact provides connectivity support to those data sources including today's IoT protocols and platforms such as LoRa, Zigbee, IEEE 802.15.4, Sigfox, enOcean, MQTT, XMPP, HTTP, CoAP, etc. With its modular approach, connectivity support for new protocols can be rapidly developed and dynamically added to the platform, even at run-time.

sensiNact can be used in each use case that involves heterogeneous data sources. It is particularly adapted for the cross-border EU-JP use cases, in which it can provide interoperability among the platforms used by the pilot sites. The sensiNact platform provides a unified view over those heterogeneous sources of data and action. With the lack of a de facto standard data model today in the IoT domain, sensiNact adopts a generic and extensible data model to facilitate building adapters for various protocols. Its core model is based on 4 types of resources: sensor data, action, state variables, and properties. Those resources are accessible by generic and easy to use APIs providing synchronous (on demand) and asynchronous (periodic or event based) access to data/actions of IoT devices as well as access to historic data.

On top of the platform, sensiNact proposes a management and development tool (sensiNact Studio) that allows managing devices/services connected to the platform and rapidly creating applications and deploying



them to the platform. With a Domain Specific Language, the developers can express the application logic in terms of Event-Condition-Action rules, which is verified and validated by the tool before its deployment into the sensiNact platform. sensiNact Studio can be used to test and deploy applications related to use cases described in the **Section 2**.

Requirements summary related to M-Sec

The asset covers the following use case requirements:

- The sensiNact platform can address requirements related to the data connection from heterogeneous data sources in every use case.
- sensiNact Studio addresses requirements related to secure management of IoT devices, rapid prototyping of rule-based applications that can exploit data and actuation capabilities provided by IoT devices, and the lifecycle management of the deployed applications.



4. Elicitation of M-Sec Requirements

4.1 Methodology

This section provides a consolidated view of the requirements described above so as to conclude to a unique coding of all individual requirements in order to be tracked throughout the lifecycle of the project.

The tables presented in **subsection 4.4** provide an overview of the elicited requirements of the M-Sec project. It has to be underlined that the requirements are derived not only from the use cases, but also from a range of other factors including the partners' background projects, state-of-the-art projects and requirements expressed by the project stakeholders. The selection of requirements has been also driven by:

- The need to address, design and implement the innovative features of the M-Sec framework.
- The functionalities that have to be offered in the use cases.
- Non-functional features that the final M-Sec system has to include.

The first column of these tables provides a unique coding for each requirement. That way, each and every requirement will be traceable throughout the whole lifecycle of the project. The second column gives the description of the requirement while the third column describes any dependencies or comments. The fourth and fifth columns provide the details on the origin of the document and the relevant stakeholder respectively, thus linking the requirement with the corresponding summaries as these have been provided by the partners in the previous sections of this document. For a better traceability of the requirements from their inception to their implementation, the requirements in some cases are mapped to specific assets' coverage.

In some tables the third and/or fourth column are not present, because for the specific reported requirements the origin is the same. In these cases a note is provided at the bottom of the tables.

These tables will be updated and extended further in the next versions of this document. For example, they will contain an additional column which will be focused on the prioritization of the requirements, distinguishing them between *High* (critical for the implementation of many other functionalities within the project), *Medium* (can be implemented in a second iteration) and *Low* (must be implemented but it is not critical for other components) priority, based on whether the specific requirement is to be addressed in the first version of the prototype release or in next iterations.

4.2 M-Sec Platform Functional Requirements

The functional view of a requirements analysis process focuses on what the system must do to produce the required operational behaviour. It includes inputs, outputs, states, and transformation rules. The functional requirements are the primary sources of the requirements that will eventually be reflected in the system specification. Indicatively, the functional view information includes:

- System functions;
- Tasks or actions to be performed;



- Inter-function relationships;
- Hardware and software functional relationships;
- Functional constraints;
- Interface requirements.

M-Sec is a complex project, with many different components that should work together to offer interesting services to the users. In order to construct an integrated system, the development process must fulfil some requirements. These functionality requirements are intended for developers and integrators as well as for final users.

In turn, the non-functional requirements are divided in three groups:

- **M-Sec Platform generic functional requirements:** this group includes functional requirements that are directly related to M-Sec platform itself (Table 7).
- **M-Sec Use Case specific functional requirements:** this group includes functional requirements that are related to the Use Cases reported in **Section 2** (Table 8).
- **M-Sec assets functional requirements:** this group includes functional requirements that are related to the assets presented in **Section 3** (Table 9).

4.3 M-Sec Platform Non-Functional Requirements

The non-functional view of a requirements analysis process focuses on what other technical features the system must have in order to facilitate the service provision. These include, among others (Tables 10-16):

- **Security**
- Scalability
- Performance
- Reliability and availability
- Manageability and flexibility
- Modularity
- Openness and Extensibility Requirements

Non-functional requirements are very important in setting the foundation of a complex integrated system as is the case of M-Sec. Due to the nature of the project, the analysis that takes place focuses mainly on Security.



4.4 Consolidation and Coding of M-Sec requirements

Table 7: Consolidated and Coded M-Sec Platform Generic Functional requirements

Category: R1 Functional Requirements		
Group 1: M-Sec platform generic functional requirements		
Code	Description	Relevant Stakeholders
R1.1.1	M-Sec should be able to collect data from heterogeneous data sources (open data from the city, real-time traffic information, localisation of users, etc.)	Service providers & Integrators, IoT infrastructure providers
R1.1.2	The platform should be able to be integrated with existing sensor networks.	Service providers & Integrators, IoT infrastructure providers
R1.1.3	The platform should be able to access data from sensors on demand and through subscriptions.	Service providers, IoT infrastructures
R1.1.4	The platform should be able to access online data, e.g. from web sites.	Service providers & Integrators
R1.1.5	M-Sec should provide means to push data from the users and relevant stakeholders into the platform.	Service providers & Integrators, IoT infrastructure providers
R1.1.6	The platform should be able to collect and store data.	Service providers & Integrators
R1.1.7	The platform should be able to provide stored historical data.	Service providers & Integrators
R1.1.8	The platform should provide real-time data processing functionalities.	Service providers & Integrators
R1.1.9	The platform should provide edge processing functionalities.	Service providers & Integrators, IoT infrastructure providers
R1.1.10	The platform should provide big data analytics functionalities.	Service providers & Integrators
R1.1.11	The platform should provide a dashboard in order to present results of analysis.	Service providers & Integrators
R1.1.12	M-Sec should be able to issue notifications to end users when interesting events occur.	Smart Cities, End Users
<p>Dependencies with other requirements and Comments: The generic functional requirements of the M-Sec platform have been derived as system requirements from the technical partners of the consortium after analysing the use case requirements from the end users. For this reason there is a strong dependency with requirements under Group 2.</p> <p>Origin of requirement: All use cases.</p>		



Table 8: Consolidated and Coded M-Sec Use Case specific Functional requirements

Category: R1 Functional Requirements				
Group 2: Use Case specific requirements				
Code	Description	Dependencies & Comments	Origin	Relevant Stakeholders
R1.2.1	Deployed devices should not impact negatively the scenario nor affect the daily operations as they are before their deployment.		SAN-UC1/ FUJ-U4	IoT infrastructure providers
R1.2.2	The local architecture should be scalable and able to be integrated with others.	R2.2.*	SAN-UC1/ FUJ-U4	Service providers & Integrators
R1.2.3	The citizens should install a smartphone app as an interaction with the services based on the M-Sec platform.		SAN-UC1/ CB-UC6	Smart Cities, End Users
R1.2.4	The service should facilitate the visualisation of real-time information in a map or in a list (physical sensing information).	R1.2.8, R1.2.17 R1.2.5	SAN-UC1	Service providers
R1.2.5	The service should facilitate the visualisation of historical information in a map or in a list (physical sensing information).	R1.2.8, R1.2.17 R1.2.4	SAN-UC1	Service providers
R1.2.6	The service should facilitate user published Events.	R1.1.5	SAN-UC1/ CB-UC6	Service providers, Smart Cities, End Users
R1.2.7	The associated web application should provide means to gather satisfaction information from the user.	R1.2.3	SAN-UC1	Service providers, End Users
R1.2.8	The application should provide a tool to analyse data and extract statistics in a simple and easily understandable way for the city economic development division and for the event organisers.	R1.1.7, R1.1.11	SAN-UC1/ FUJ-IC4	Service providers, Smart Cities, End Users
R1.2.9	The system should let users collect information about their wellbeing status.	R1.2.10	SAN-UC2	End Users
R1.2.10	The system should let users collect information from sources not directly attached to their Body Area Network, such as sensors in the room where the user is.	R1.1.1	SAN-UC2	IoT infrastructure providers, End Users



R1.2.11	The system should store massive information (such as temperature per second) in an efficient way, taking into account that not every data is expected to be recorded forever.	R1.1.6, R1.1.7	SAN-UC2	Service providers & Integrators
R1.2.12	The system should store information (such as access to data) in a way that allows not to forget this information, and with mechanisms that allow third parties to verify that this information is correct and true.	R1.2.11 R1.1.6, R1.1.7	SAN-UC2	Service providers & Integrators
R1.2.13	The system should collect environment sensor data.	R1.2.1	FUJ-UC3	IoT infrastructure providers, Smart Cities
R1.2.14	The system should be able to handle a large number of data streams concurrently.	R2.4.*	FUJ-UC3/4	Service providers & Integrators
R1.2.15	The system should be able to transfer the data streams in real time.	R2.3.*	FUJ-UC3/4	Service providers & Integrators
R1.2.16	The system should collect heterogeneous data on citizens' life in real time.	R1.1.1	FUJ-UC4	IoT infrastructure providers, Smart Cities
R1.2.17	The associated web application should provide and visualise environment information collected over the city.	R1.2.7, R1.2.8, R1.2.20	FUJ-UC4/ CB-UC6	IoT infrastructure providers, Service providers & Integrators
R1.2.18	The application should provide a tool to analyse data and extract statistics in simple and easily understandable way for the city environment division and citizens.	R1.2.20, R1.2.24	FUJ-UC4	Service providers & Integrators, Smart Cities, End Users
R1.2.19	The devices should be used to collect users' data such as behaviour characteristic.	R1.2.24	CB-UC5	IoT infrastructure providers
R1.2.20	The app should facilitate the visualisation of historical information in a map or in a list (physical sensing information).	R1.2.7, R1.2.8, R1.2.17	CB-UC6	Service providers & Integrators, Smart Cities, End Users
R1.2.21	The app should offer users the option to Subscribe/Unsubscribe to specific types of events occurring in the city.	R1.1.3	CB-UC6	Smart Cities, End Users
R1.2.22	The app should let users get notified of the occurrence of an event of a type the users are subscribed to.	R1.1.12	CB-UC6	Smart Cities, End Users
R1.2.23	The app should let users search for events	R1.2.22	CB-UC6	Smart Cities, End Users



	filtering by date, type or location.			
R1.2.24	The application should provide a tool to analyse data and extract statistics in simple and easily understandable way for the municipal services and citizens.	R1.2.19	CB-UC6	Service providers & Integrators, Smart Cities, End Users
R1.2.25	Users should be able to report through smart phones various incidents.	R1.2.6, R1.1.5	CB-UC6	Smart Cities, End Users

Table 9: Consolidated and Coded M-Sec Assets Functional requirements

Category: R1 Functional Requirements		
Group 3: M-Sec platform assets requirements		
Code	Description	Relevant Asset
R1.3.1	The IoT environmental sensing devices run on batteries. Their battery life will vary depending on the frequency (agreed between the involved parties) of sending data.	3
R1.3.2	IoT devices should implement a secure link to send data and avoid tampering or hacking.	3
R1.3.3	M-Sec should implement Smart Contracts in order to take advantage of this asset.	5
R1.3.4	M-Sec should have a notion of currency.	5
R1.3.5	Mobile Wallet does not guarantee real-time access to data.	5
R1.3.6	Connected Assistance does not contain end-to-end security mechanism. This limitation should be resolved by M-Sec security mechanism.	6
R1.3.7	Connected Assistance does not comply with current law of GDPR. A mechanism regarding right of deletion, forgotten and so on should be implemented.	6
R1.3.8	Connected Assistance does not provide a secure storage of data collected.	6
R1.3.9	Connected Assistance does not provide a secure communication for video call/call.	6
R1.3.10	The devices need to install balena.io.	7
R1.3.11	The devices need some connectivity to be updated.	7
R1.3.12	The IoT devices should support node.js. Alternatively, the device can support cross-compiling.	8



R1.3.13	The platform can estimate the amount of urban waste generation in a fine-grained way using images captured by camera. However, the system may harm citizens' privacy, since the images may contain people walking in the city. This limitation should be resolved by the M-Sec privacy mechanism.	9
R1.3.14	The platform can collect environment sensor data. Also, it can deliver the environment data stream to citizens. However, the sensing system in this platform does not have any security support. This limitation should be resolved by M-Sec security mechanism.	9
R1.3.15	SOX can deliver data streams to citizens. However, it does not contain end-to-end security mechanism. This limitation should be resolved by M-Sec security mechanism.	9
R1.3.16	The data security solution will depend upon the available resources in the IoT gateway devices.	10
R1.3.17	The data security solution provided by YNU Honeypot will depend upon the available resources in the IoT gateway devices.	12
R1.3.18	Permissioned blockchain for privacy: in order to hide organization's blockchain network activity from public (i.e. unauthorized parties). In permissioned (or private) blockchains, only authorized members can enter the blockchain network.	13/14
R1.3.19	Smart contract support: in order to automatically execute transactions (i.e. without intermediate's approval), apply specific execution and functionality under given conditions and apply other automated procedures, etc. Smart contracts are an interesting application over blockchains that can enable high levels of automation and support for decentralized apps.	13/14
R1.3.20	Transactions per second (TPS) should be from 10-15 to 1000-2000: in order to serve multiple requests at the same time; our platform should scale that much to process all kinds of transactions, such as simple transactions, simple smart contracts or many that call one another, etc.	13/14
R1.3.21	Single profile/account with digital wallet available to every participant (e.g. end-users, companies, website owners, content providers, etc.): in order to be able to make purchases and payments and participate in an ecosystem of value exchange over IoT infrastructures in smart cities etc.	13/14
R1.3.22	Anonymous Identity (ID) for every party that wants to hide their real/private information from transaction tracing (not all Use Cases, needs global implementation): in order to provide anonymity, the blockchain platform should be aware of the private information of a party, providing an anonymous ID and allowing the party's transactions to happen only through this without revealing any other information in the permissioned blockchain network.	13/14
R1.3.23	Mining algorithm in our platform; providing incentive for the end users to be active and not malicious in the blockchain network.	13/14



R1.3.24	Privacy and anonymization requirements; privacy by design is an essential implementation principle of the project; user private data will not be revealed outside the blockchain network as the latter will be permissioned; additionally, in order to anonymize the user identity, external certificate authorities might be deployed (such as public bodies) who can verify the true identity of the user (particularly useful for the Know Your Customer procedures) and provide them a user hash-ID which will be the only identification used to interact further with the system. Off-chain data storage and linking with on-chain transactions will be needed for increasing the privacy notion within M-Sec.	13/14
R1.3.25	It is required to model the reactivity of an environment in a labelled transition system. Besides, it is also needed to distinguish the actions in the model into a controllable one or uncontrollable one.	15
R1.3.26	The MTSA cannot address non-functional requirements such as “X must happen in Y seconds”.	15
R1.3.27	The MTSA is based on discrete event systems. Therefore it is required to translate continuous value like sensing data into discrete events.	15
R1.3.28	MTSA cannot synthesize a controller if requirements are too strict. In that case, the requirements or assumptions on the environment should be relaxed. It is required to model the interactions between a system and an environment in labelled transition system. In addition it is also needed to distinguish the actions in the model into controllable one and uncontrollable one.	15
R1.3.29	REMU requires to model possible reactivity as rules, which should be determined in the development time.	15
R1.3.30	This technique needs that reactivity of an environment is modelled in the form of Labelled Transition System.	15
R1.3.31	REMU takes execution traces in the form of a sequence of events.	15
R1.3.32	REMU is designed to be used with MTSA.	16
R1.3.33	The technique needs the reactivity of an environment to be modelled in the form of Labelled Transition System.	16
R1.3.34	REMU takes execution traces in the form of a sequence of events.	16
R1.3.35	REMU requires to model possible reactivity as rules, which should be determined in the development time.	16
R1.3.36	Hardware ideally requires an on-board integration (SPI or USB).	17
R1.3.37	Firmware and OS should be compatible with security.	17



Table 10: Consolidated and Coded M-Sec Non-Functional Security & Privacy requirements

Category R2: Non-functional Requirements				
Group 1: Security & Privacy				
Code	Description	Dependencies & Comments	Origin	Relevant Stakeholders
R2.1.1	The M-Sec platform should be capable of managing different users' profiles distinguishing between stakeholders, actors and roles.	R2.1.2	All use cases	All
R2.1.2	The platform should store privacy covered data in a protected way. Access to protected data should be possible only to authorised users.	R2.1.3	All use cases	All
R2.1.3	The applications and technologies used in M-Sec must respect all regulations concerning the ethical aspects, especially those related to data protection and privacy.		All use cases	All
R2.1.4	M-Sec security and privacy parameters should be (re-) configurable.	R2.1.4	All use cases	Smart Cities, End Users
R2.1.5	The ID of citizens and visitors mobile phones attending the city spots where the pilot is carried out should be anonymised.	R2.1.3	SAN-UC1	Service providers, End Users, Smart Cities
R2.1.6	The associated web application should protect the privacy of the end-user and propose several levels of management of personal data, and give the possibility of modifying the privacy parameters any time.	R2.1.4	SAN-UC1	Service providers, End Users
R2.1.7	The system should have an access control policy, binding the users with different profiles, each with different access privileges to data (the owner of the data, person assigned by them to consult their data –familiar or professional-, software administrator, technical support, security officer...)	R2.1.9	SAN-UC2	Service providers, End Users
R2.1.8	The system should have several policies of access to the data, depending on the role the user presents and the data the user is trying to access to. Anonymous users should be given no access to any data related to users.	R2.1.9	SAN-UC2	Service providers, End Users



R2.1.9	The system should have a dashboard for matching roles to policies to privileges of access.		SAN-UC2	Service providers, End Users
R2.1.10	The system should support an easy to use mechanism that enables the owner of the data to grant privileges of access to other users, in an understandable way.	R2.1.11	SAN-UC2	Service providers, End Users
R2.1.11	The system should support mechanisms for authentication and authorisation of the users, including updating these users' proofs of access privileges.	R2.1.9	SAN-UC2	Service providers, End Users
R2.1.12	The system should include security measures which protect data transmitted over the network at application level against eavesdropping (encryption and peer authentication)		SAN-UC2	Infrastructure Providers, Service providers & Integrators
R2.1.13	If there is a role that can assign privileges of access to users on behalf of the owner of the data, the system would enable a way to record the user decision about this assignation (such as signing a consent form prior to assign those privileges)		SAN-UC2	Service providers, End Users
R2.1.14	The system should show and record the user consent about the usage of their data. The consent can be CRUD at any moment, and the decisions of the users about their data logged.	R.2.1.4	SAN-UC2	Service providers & Integrators, End Users
R2.1.15	The system should log the interaction that the users have with the system, especially the access to data (not the data themselves). The logging should include the profile the data was accessed from, as well as the action(s) performed (authentication, CRUD on data, access granting/validation/revocation/removal, signing of consent...)		SAN-UC2	Service providers & Integrators, End Users
R2.1.16	The system should store sensitive information (users' data and detailed logs that could reveal information about users or secrets about the system itself) in a way that this information could be permanently deleted.	R2.1.4	SAN-UC2	Service providers & Integrators, End Users
R2.1.17	The system should support replying to queries concerning Data Protection of the users that employ the system. If this request cannot be satisfied real-time, the system should guarantee that an answer	R2.1.4	SAN-UC2	Smart Cities, End Users



	will be provided in a reasonable time range.			
R2.1.18	The system needs to secure the heterogeneous components involved in the data stream dissemination, so that they are not hacked by malicious attackers.	R2.1.19 R2.1.20 Asset 2	FUJ-UC3/4	Infrastructure Providers, Service providers
R2.1.19	The system should protect the data streams from malicious attackers at the edge and distributed cloud platform.		FUJ-UC3/4	Infrastructure Providers, Service providers & Integrators
R2.1.20	The system needs to secure the data streams, so that the data are not tampered in the network between their source and destination.		FUJ-UC3/4	Infrastructure Providers, Service providers & Integrators
R2.1.21	The system should not harm citizens' privacy, thus an automated privacy protection mechanism should be provided.	R2.1.4	FUJ-UC3/4	ALL
R2.1.22	The system should disseminate the environment data stream to citizens/municipalities securely.	R2.1.24	FUJ-UC3/4	Service providers & Integrators, Smart Cities, End Users
R2.1.23	The cloud system should store the data securely so that they are not disclosed to any party without permission.		FUJ-UC4	Service providers & Integrators
R2.1.24	The cloud system should store the data securely and should be accessed by both EU and Japan.		CB-UC5	Smart Cities
R2.1.25	The block chain technology should be utilised to secure the data.		CB-UC5	Service providers & Integrators
R2.1.26	The ID of citizens and visitors reporting incidents should be anonymised.	R1.1.5	CB-UC6	Service providers, End Users
R2.1.27	The cloud system should store the data securely so that they are not disclosed to any party without permission.		CB-UC6	Service providers & Integrators
R2.1.28	M-Sec should cover with state-of-the-art technologies all the aforementioned security aspects.	R2.1.*	All use cases	IoT infrastructure providers, Service providers & Integrators



Table 11: Consolidated and Coded M-Sec Non-Functional Scalability requirements

Category R2: Non-functional Requirements			
Group 2: Scalability			
Code	Description	Dependencies & Comments	Relevant Stakeholders
R2.2.1	The M-Sec system must be able to scale with respect to more input sensors. In this way, the number of edge nodes within an M-Sec system should not be limited and more edge nodes should be deployed to handle more sensors.	R1.1.1, R1.1.2	Infrastructure and Service Providers, Integrators
R2.2.2	The big data analytics engine should be able to scale to more input data from more edge nodes.	R1.10	Infrastructure and Service Providers, Integrators
R2.2.3	The dashboard should be able to respond to user queries from the use cases, and moreover should be able to handle multiple users of the use cases at once, such that results are obtained quickly even under load.	R1.11	Infrastructure and Service Providers, Integrators
<i>Origin: All Use Cases</i>			

Table 12: Consolidated and Coded M-Sec Non-Functional Performance requirements

Category R2: Non-functional Requirements			
Group 3: Performance			
Code	Description	Dependencies & Comments	Relevant Stakeholders
R2.3.1	The M-Sec system deploys various big data analytics frameworks that have demands in computational power. They should be regularly evaluated during development, such that they are shown to be accurate with real-time data.		Infrastructure and Service Providers, Integrators
R2.3.2	Statistics and reports that should be displayed through the dashboard is a process with high performance needs.	R1.1.10	Infrastructure and Service Providers, Integrators
<i>Origin: All Use Cases</i>			



Table 13: Consolidated and Coded M-Sec Non-Functional Reliability & Availability requirements

Category R2: Non-functional Requirements			
Group 4: Reliability and availability			
Code	Description	Dependencies & Comments	Relevant Stakeholders
R2.4.1	The M-Sec system should have a high availability and reliability (e.g. more than 98% in regular operation during the pilots) that can be monitored, measured and audited.		Infrastructure and Service Providers, Integrators
R2.4.2	In case of failures, measures have to be taken in order to overcome these in short notice and additional measures for preventing their occurrence.		Infrastructure and Service Providers, Integrators
<i>Origin: All Use Cases</i>			

Table 14: Consolidated and Coded M-Sec Non-Functional Manageability & Flexibility requirements

Category R2: Non-functional Requirements			
Group 5: Manageability and flexibility			
Code	Description	Dependencies & Comments	Relevant Stakeholders
R2.5.1	The M-Sec system should have a high manageability and flexibility even for users that are not considered experts.		Infrastructure and Service Providers, Integrators
R2.5.2	Common management attributes such as add/delete/update should be intuitive and easy to be performed.	R1.1.11	Infrastructure and Service Providers, Integrators
R2.5.3	The M-Sec modularity level should allow enough independence of all modules so as if any module needs to be replaced, this will have no consequences to the other modules.		Infrastructure and Service Providers, Integrators
<i>Origin: All Use Cases</i>			



Table 15: Consolidated and Coded M-Sec Non-Functional Openness & Extensibility requirements

Category R2: Non-functional Requirements			
Group 6: Openness and extensibility			
Code	Description	Dependencies & Comments	Relevant Stakeholders
R2.6.1	The various components of M-Sec should be ideally portable across major operating systems.		Service Providers, Integrators, OS Community
R2.6.2	The various components of M-Sec should be interoperable with other services implementing common and open standards		Service Providers, Integrators, OS Community
R2.6.3	The core components of the M-Sec framework should be extensible to new unforeseen types of sensors and events captured.	R1.1.2	Service Providers, Integrators, OS Community
R2.6.4	M-Sec APIs should rely on open standards and built upon other existing open standards where possible.		Service Providers, Integrators, OS Community
R2.6.5	M-Sec should provide programming interfaces for application developers to gather real-time and historic data	R2.6.4	Service Providers, Integrators, OS Community
<i>Origin: All Use Cases</i>			

Table 16: Consolidated and Coded M-Sec Non-Functional Design & Implementation requirements

Category R2: Non-functional Requirements			
Group 7: Design and implementation requirements			
Code	Description	Dependencies & Comments	Relevant Stakeholders
R2.7.1	M-Sec should reuse existing open source software and tools, where it is appropriate and possible according the license.	R2.6.*	Service Providers and Integrators
R2.7.2	The architecture of M-Sec must be layered, supporting modularity and high customisation per use case and application.		Service Providers and Integrators
R2.7.3	The components of M-Sec must be developed following accepted good programming practices.		Service Providers and Integrators, OS Community



R2.7.4	The components should be developed using proven and trusted languages.	R2.6.4	Service Providers and Integrators, OS Community
Origin: All Use Cases			



5. Conclusions

Requirements analysis plays an important role for the whole lifecycle of the M-Sec project. It is the input for the M-Sec specification and overall architecture as well as for the validation of the final system and its evaluation against the desired functionality.

In this document, we have presented the requirements analysis methodology along with the defined functional and non-functional requirements of the M-Sec system.

The analysis starts with a definition of the M-Sec concept, including an overview of the use cases and the stakeholders involved so as to understand the context of the project and identify the various stakeholders who represent a holistic value chain.

Similar projects and background from previous projects are also mentioned in order to present the state-of-the-art and the previous achievements that can be used as a starting point.

In the sequel the elicitation of requirements is derived from the definition of the desired functionality of the M-Sec system given from the perspectives of the functional components and the non-functional attributes that the end system has to expose. The M-Sec system should have specified functionality to facilitate the use cases and a rich set of non-functional requirements such as scalability, reliability, availability, manageability, flexibility, modularity, openness, etc.

The document has concluded with a consolidated and coded requirements list that will be the reference for the design activities and the implementation and validation of the M-Sec system. The list has been divided into groups of requirements, indicating dependencies between the various requirements but also the relevant stakeholders that are affected.

The current deliverable will be one of the main reference documents for the design and specification of the M-Sec system.