D4.1 – Cloud-based enablers for LBS provision

Deliverable ID	**D4.1**
Deliverable Title	**Cloud-based enablers for LBS provision**
Work Package	**WP4**
Dissemination Level	**PUBLIC**
Version	**1.0**
Date	**2018-10-31**
Status	**Submitted**
Lead Editor	**CNET**
Main Contributors	**ISMB, FIT**

Published by the GOEASY Consortium

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 776261.

Document History

Version	Date	Author(s)	Description
0.0	2018-10-01	Peter Rosengren	First Draft with TOC
0.1	2018-10-02	Peter Rosengren	Basic descriptions
0.2	2018-10-06	Peter Rosengren	Added architecture diagram
0.5	2018-10-15	Peeter Kool	Added API descriptions
0.8	2018-10-19	Mikael Mölder	Added OGC format descriptions
0.9	2018-10-29	Peter Rosengren	Edited version ready for peer review
0.95	2018-10-30	Giuseppe Pacelli ISMB Leonardo Fringuelli,GreenApes	Peer reviewed and approved with comments
1.0	2018-10-31	CNET	Final version, ready for submission.

Table of Contents

1 Introduction

1.1 Scope

This deliverable is the first from task 4.1 which will focus on the developing the enabling technology for the backend cloud needed to support the envisioned location-based services. This includes real-time event processing of incoming messages and observations, big data storage technologies for the management, structuring and aggregation of geo-based data. In general, this will result in the development of components, built upon existing scalable cloud open enablers, suitable to store, selectively retrieve and analyse the data under real-time constraints provided by reference applications. Development work in this task will be structured to enable release and future re-use of developed enablers.

Document sections are presented as follows. Section 2 summarises the GoEasy overall architecture while section 3 describes and discusses the specific cloud architecture aspects. Sections 4 presents an initial set of cloud-based enabler. Section 5 describes a first prototype, while section 6 summarises our conclusions.

1.2 Related documents

ID	Title	Reference	Version	Date
[RD.1]	D2.2 Initial GOEASY Platform and Pilots Reference Architecture_2.0			

2 Architecture design

2.1 GoEasy Architecture

The overall GOEASY concept is described in Figure 1.

- GOEASY Devices are trusted physical internet-connected devices.
- Application and services running on-board devices can access the trusted GOEASY platform via the open GOEASY APIs, which allow access to core GOEASY services such as the end-to-end authentication of position information, the trusted measurement and exchange of position information, dependable LBS or privacy-aware Database Management System (DBMS).
- To provide its services, the core GOEASY enabled devices and platform depends naturally on GNSS services. Moreover, GOEASY is externally supported by third-party services federated with the platform, cloud-based applications, beyond interacting directly with devices with GOEASY components on-board, can also access GOEASY services via the open GOEASY application API.

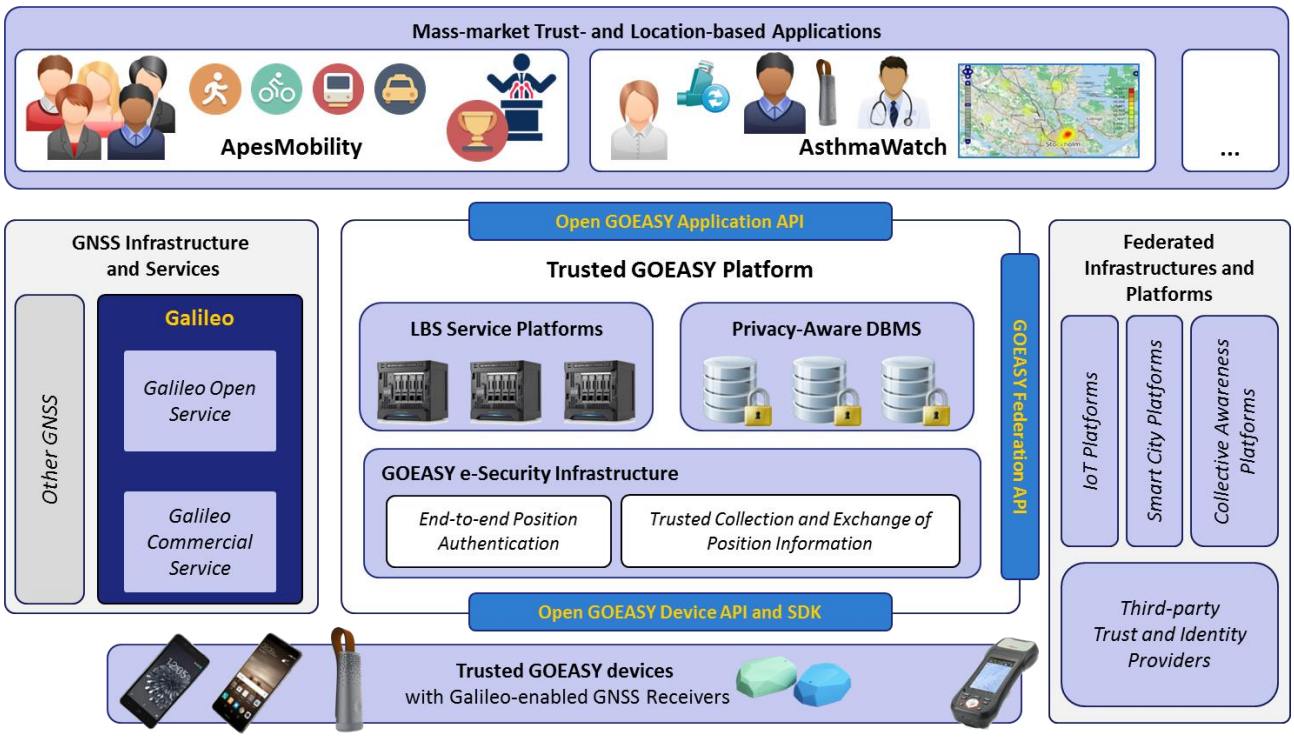

Figure 1. The GOEASY high-level conceptual architecture

Figure 2 presents the major components and the functional architecture of GoEasy. More details are available in deliverable D2.2 "Initial GOEASY Platform and Pilots Reference Architecture".

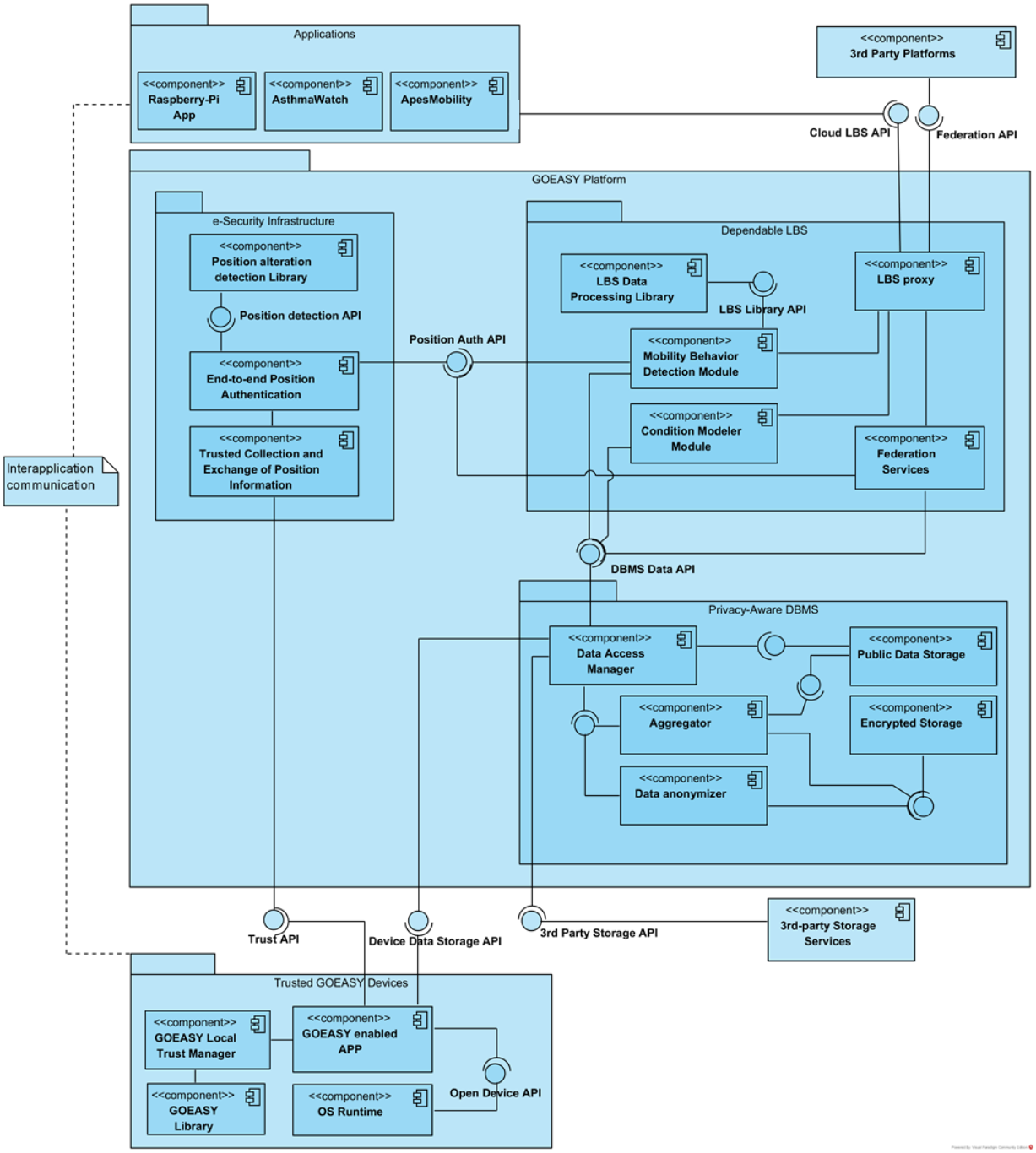

Figure 2. GOEASY Functional Architecture - 1st Release

3 The GoEasy Cloud Architecture

We have chosen a message oriented architecture in GoEasy based on a message broker. The selected message broker is RabbitMQ which implements the concepts of brokers, messages, producers, exchanges, queues and consumers. A publisher – an application that produces messages - sends a message to an exchange, where it is routed to one or more queues. In GoEasy this will typically be the gateway or smart phones that will publish messages. The message is then pushed to (or pulled by) a consumer – an application that processes messages - for processing. The topology of the message routing is controlled by

the publisher and consumer, which allows for very flexible communication design. Exchanges and brokers are access-controlled, which allows for fine-grain security control over the communication.

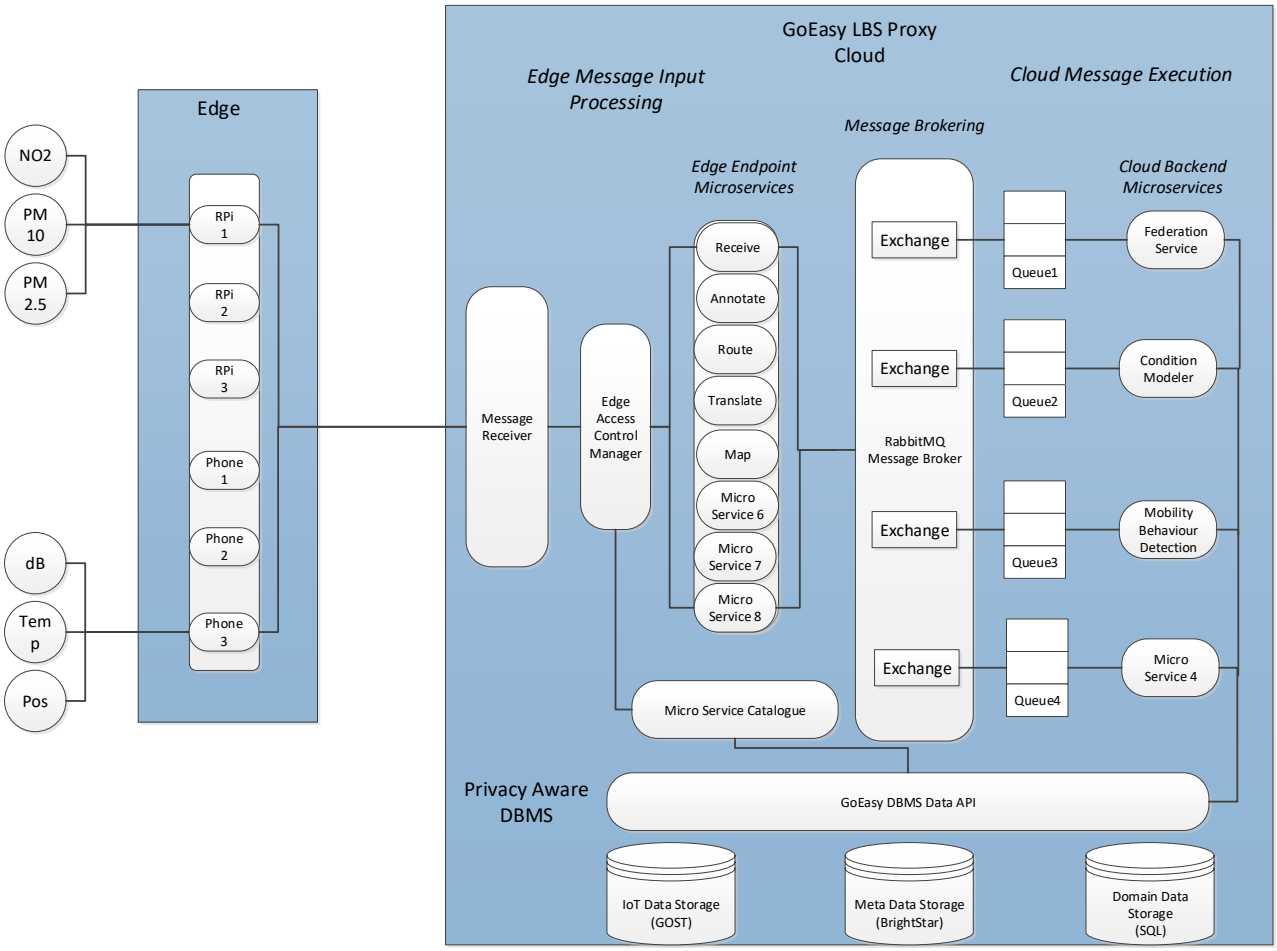

Figure 3 The GoEasy Cloud Architecture

To increase the scalability of the solution we will implement microservices which each has a dedicated task to perform. This will make it more efficient to process the different message queues and to scale out queue processing if necessary. Furthermore, we will implement a Fog/Edge Computing Architecture where processing can take place both at the edge level (i.e. the gateways) and in the cloud back end.

4 Initial Cloud-based Enablers

This module is focused on the enabling technology for the backend cloud needed to support the envisioned location-based services. It consists of a set of enablers. This includes real-time event processing of incoming messages and observations, big data storage technologies for the management, structuring and aggregation of geo-based data. In general, this will result in components, built upon existing scalable cloud open

enablers, suitable to store, selectively retrieve and analyse the data under real-time constraints provided by reference applications. In addition to the overall cloud enabling architecture we will develop:

- Real-time event processing enablers.
- Dedicated enablers suitable to support privacy-oriented features.
- Specific enablers to support the AsthmaWatch pilot.
- Specific enablers to support the ApesMobility pilot.

In order to enable the creation and growth of third-party ecosystem we will define standardized, easy-to-use and open APIs for developers. We intend to base our APIs for the IoT-data on the OGC (Open Geospatial Consortium) SensorThings format, which defines a data model for exchange of IoT data. There will also be an API for storing and updating domain data needed by the different applications. This API will be elaborated in the next version of this deliverable.

The API will provide a high-level semantic layer for app developers to use. Different existing ontologies will be evaluated to foster use of common concepts. In addition to the open cloud-based LBS APIs, the following will also be developed:

- Automatic tools for cross-platform API definition and documentation.
- Privacy-related extensions to the API i.e. sub-set of meta-APIs to gather, annotate and control how different privacy related aspects are handled in the remaining API resources.
- Alignment of APIs with location-based collective awareness platform.

4.1 Edge Message Input Processing

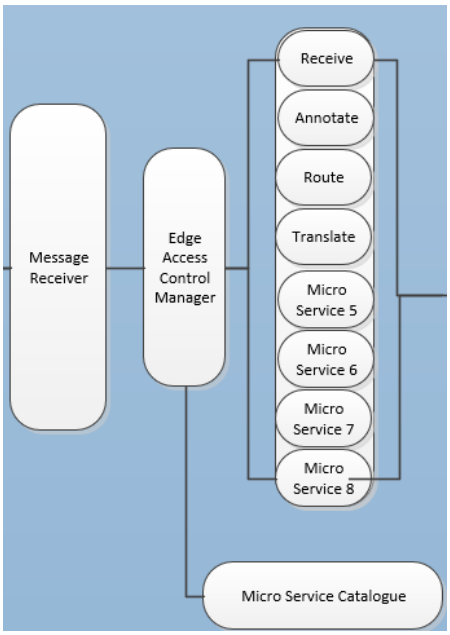

Figure 4 The processing of incoming messages is managed by a set of microservices

4.1.1 Message Receiver

The Message Receiver is the endpoint to which the publishers send their messages. A typical incoming message in the OGC format is shown below:

```
{
        "resultTime": "2018-10-30T08:00:00+01:00",
```

```
                "Datastream": {
                "@iot.id": 2206
                },
                "phenomenonTime": "2018-10-30T08:00:00+01:00",
                "result":
                        {
                                "valueType": "NO2",
                                "Position":
                                        {
                                        "type": "Point",
                                        "coordinate": [
                                        59.316006853996,
                                        18.057807343668
                                        ]
                                        },
                        "response":
                                {
                                "value": "10.3983"
                                }
                        }
        }
```


Edge Access Control Manager

5 Conclusions

We have described the design of the GoEasy Cloud-based Enablers and the architecture to support it. We have chosen a message-oriented architecture in GoEasy based on a message broker. The selected message broker is RabbitMQ which implements the concepts of brokers, messages, producers, exchanges, queues and consumers.

To increase the scalability of the solution we will implement microservices which each has a dedicated task to perform. This will make it more efficient to process the different message queues and to scale out queue processing if necessary. Furthermore, we will implement a Fog/Edge Computing Architecture where processing can take place both at the edge level (i.e. the gateways) and in the cloud back end.

The message format chosen for IoT-data is the OGC SensorThings.

6 References

[1] IEEE, *ISO/IEC/IEEE 42010:2011, Systems and software engineering — Architecture description,* 2011.

[2] N. Rozanski and E. Woods, Software Systems Architecture: Working With Stakeholders Using Viewpoints and Perspectives, Addison-Wesley Professional, 2011.

[3] E. Commission, “REGULATION (EU) 2016/679 OF THE EUROPEAN PARLIAMENT AND OF THE COUNCIL of 27 April 2016on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, and repealing Directive 95/46/EC (General Dat,” April 2016. [Online]. Available: https://eur-lex.europa.eu/legal-content/IT/TXT/?uri=CELEX%3A32016R0679.

[4] A. Hern, “The Guardian,” 28th January 2018. [Online]. Available: https://www.theguardian.com/world/2018/jan/28/fitness-tracking-app-gives-away-location-of-secret-us-army-bases.

[5] Raw Mwasurements Task force, “Using Gnss Raw Measurements,” GSA, Prague, 2017.

Acronyms

Acronym	Explanation
DoA	Description of Action
GEP	GOEASY platform
LBS	Location-Based Services
API	Application Programming Interface
IOT	Internet of Things
GNSS	Global Navigation Satellite System
DBMS	Database Management System
XMPP	Extensible Messaging and Presence Protocol
MQTT	Message Queuing Telemetry Transport
STOMP	Simple Text Oriented Message Protocol
AMQP	Advanced Message Queuing Protocol
OGC	Open Geospatial Consortium

Appendix A: Application requirements – ApesMobility

Key	Summary	Status	Labels
GOEAS-65	As a citizen, I want to be able to delete identifiable location data to protect my privacy.	OPEN	ApesMobility UC-APES-08
GOEAS-64	As a citizen, I want to be able to enable and disable location tracking on a regular basic automatically to protect my privacy.	OPEN	ApesMobility UC-APES-08
GOEAS-63	As a citizen, I want to be able to enable location tracking only for a specific time frame to protect my privacy.	OPEN	ApesMobility UC-APES-08
GOEAS-62	As a citizen, I want to be able to en-/disable location tracking manually to protect my privacy.	OPEN	ApesMobility UC-APES-08
GOEAS-61	As a citizen, I want to subscribe to specific areas or topics only in order to get relevant information only.	OPEN	ApesMobility UC-APES-07
GOEAS-60	As a citizen, I want to be notified about new challenges in order to improve my sustainable behavior.	OPEN	ApesMobility UC-APES-06 UC-APES-07
GOEAS-59	As a citizen, I want to see additional information about registered locations/ events in order to decide whether to go there or not.	OPEN	ApesMobility UC-APES-06 UC-APES-07
GOEAS-58	As a citizen, I want to be able to find registered POIs and challenges in order to collect additional rewards by check-in (and share).	OPEN	ApesMobility UC-APES-06 UC-APES-07
GOEAS-57	As a citizen, I want to be notified about a detected, certified activity in order to not miss additional rewards.	OPEN	ApesMobility UC-APES-01
GOEAS-55	As a citizen, I want to be able to share my certified activities in order to motivate fellow citizens.	OPEN	ApesMobility UC-APES-01
GOEAS-20	As a citizen, I want to have full control of the location tracking in order to protect my privacy.	OPEN	ApesMobility UC-APES-01 UC-APES-08
GOEAS-15	As a citizen, I want to join challenges to get motivated and to support the community.	OPEN	ApesMobility UC-APES-07
GOEAS-14	As an organization, I want to offer challenges for users in order to familiarize them with sustainable topics.	OPEN	ApesMobility UC-APES-06

Key	Summary	Status	Labels
GOEAS-13	As a citizen, I want to get my location certified to get additional rewards.	OPEN	ApesMobility UC-APES-05
GOEAS-12	As a citizen, I want to share my location to inspire others and/or be additionally rewarded.	CLOSED- DUPLICATED	ApesMobility UC-APES-04
GOEAS-11	As a citizen, I want to join activities shared by others to get inspired for sustainable behavior.	CLOSED OUTOFSCOPE	ApesMobility UC-APES-03
GOEAS-10	As a citizen, I want to get my activity certified to get additional rewards.	OPEN	ApesMobility UC-APES-02
GOEAS-9	As a citizen, I want to be able to confirm my certified activity to be additionally rewarded.	OPEN	ApesMobility UC-APES-01

Appendix B: Application requirements – AsthmaWatch

Key	Summary	Status	Labels
GOEAS-16	As a user I want to know about the condition in an area to decide if I will go there or not.	OPEN	AsthmaWatch UC-AW-01 UC-AW-02
GOEAS-17	As a user I want to know the alternative routes based on a specific criteria in order to avoid an asthma attack.	OPEN	AsthmaWatch UC-AW-03 UC-AW-04 UC-AW-05 UC-AW-06
GOEAS-18	As a user I want to keep track of my health in order to react in time on changes.	OPEN	AsthmaWatch UC-AW-07 UC-AW-08
GOEAS-19	As a user I want to inform specific persons about an asthma attack in order to get immediate help.	OPEN	AsthmaWatch UC-AW-09 UC-AW-10 UC-AW-11
GOEAS-21	As a user I want to define rules for a specific criteria in order to be warned in relevant situations.	OPEN	AsthmaWatch UC-AW-01

Key	Summary	Status	Labels
GOEAS-22	As a user I want to be notified when I am within a certain range of a dangerous spot.	OPEN	AsthmaWatch UC-AW-01 UC-AW-02
GOEAS-23	As a user I want to be able to view information of a single measuring station to decide if I will go to this area or not.	OPEN	AsthmaWatch UC-AW-02
GOEAS-24	As a user I want to be able to define a preferred route in order to get advised whether to take it or not.	OPEN	AsthmaWatch UC-AW-03
GOEAS-25	As a user I want to be able to import a route from 3rd party app(s) in order to reduce redundant tasks (defining the same route in different applications).	OPEN	AsthmaWatch UC-AW-03
GOEAS-26	As a user I want to get alternative routes suggested automatically if conditions along my preferred route exceed my rules in order to avoid an asthma attack.	OPEN	AsthmaWatch UC-AW-05 UC-AW-06
GOEAS-27	As a user I want to be able to view details of any suggested route in order to avoid an asthma attack.	OPEN	AsthmaWatch UC-AW-06
GOEAS-28	As a user I want to be able to export a picked route in order to start navigation with a 3rd party app.	OPEN	AsthmaWatch UC-AW-06
GOEAS-29	As a user I want to be able to enter health information manually to keep track of my health.	OPEN	AsthmaWatch UC-AW-07
GOEAS-30	As a user I want to be able to add health information automatically by a connected device to keep track of my health.	OPEN	AsthmaWatch UC-AW-07
GOEAS-31	As a user I want to be able to enter health information to the application in order to keep track of my health.	OPEN	AsthmaWatch UC-AW-07
GOEAS-32	As a user I want to get immediate feedback after adding health information in order to react in time on changes.	OPEN	AsthmaWatch UC-AW-07
GOEAS-33	As a user I want to be able to view historical data in order to know about the development of my health.	OPEN	AsthmaWatch UC-AW-08
GOEAS-35	As a user I want to control the map by known interaction patterns to get the required information as fast and easy as possible.	OPEN	AsthmaWatch UC-AW-02 UC-AW-03 UC-AW-06
GOEAS-36	As a user I want to be able to specify multiple persons as emergency contacts to increase the probability to get help.	OPEN	AsthmaWatch UC-AW-09

Key	Summary	Status	Labels
GOEAS-37	As a user I want to be able to activate an alarm in case of an asthma attack in order to get immediate help.	OPEN	AsthmaWatch UC-AW-10
GOEAS-38	As a user I want to inform my emergency contacts about my position when I activated the alarm to get help as fast as possible.	OPEN	AsthmaWatch UC-AW-10 UC-AW-11
GOEAS-39	As a user I want my phone to play an alerting sound when I activated the alarm to get immediate help by people around me.	OPEN	AsthmaWatch UC-AW-10
GOEAS-40	As a user I want to view all outgoing calls in the app due to an activated alarm to know who was informed and who I should update about my situation now.	OPEN	AsthmaWatch UC-AW-10 UC-AW-11
GOEAS-41	As a user I want to be bale to cancel a call at any time to avoid unnecessary attempts of contacts.	OPEN	AsthmaWatch UC-AW-10 UC-AW-11
GOEAS-42	As a user I want to be able to select text messages to send to all emergency contacts in case I cancelled a call in order to let them know about my current situation.	OPEN	AsthmaWatch UC-AW-10 UC-AW-11
GOEAS-43	As a user I want to be able to create and update templates for text messages to inform people faster about my current situation.	OPEN	AsthmaWatch UC-AW-10 UC-AW-11
GOEAS-44	As a user I want to be able to determine the order in which contacts are contacted to prioritize based on criteria.	OPEN	AsthmaWatch UC-AW-09