**“Software Engineering”**

**Course**

**a.a. 2018-2019**

**Template version 1.0**

**Deliverable #2**

**Lecturer: Prof. Henry Muccini (henry.muccini@univaq.it)**

**Dashboard Monitoraggio Ambientale**

|  |  |
| --- | --- |
| **Date** | 23/12/2018 |
| **Deliverable** | 2 |
| **Team (Name)** | 404 Not Found |

|  |  |  |
| --- | --- | --- |
| **Team Members** | | |
| **Name & Surname** | **Matriculation Number** | **E-mail address** |
| D’Ercole Dario | *244615* | [dario.dercole@student.univaq.it](mailto:dario.dercole@student.univaq.it) |
| Carestia Alessandro | *244236* | [alessandro.carestia@student.univaq.it](mailto:alessandro.carestia@student.univaq.it) |
| De Flaviis Manuel | *231100* | [manuel.deflaviis@student.univaq.it](mailto:manuel.deflaviis@student.univaq.it) |
| Pennacchia Francesco | *247848* | [francesco.pennacchia@student.univaq.it](mailto:francesco.pennacchia@student.univaq.it) |
| Avallone Andrea | *179324* | [andrea.avallone@student.univaq.it](mailto:andrea.avallone@student.univaq.it) |

**Project Guidelines**[do not remove this page]*This page provides the Guidelines to be followed when preparing the report for the Software Engineering course. You have to submit the following information:*

* *This Report*
* *Diagrams (Use Case, Component Diagrams, Sequence Diagrams, Entity Relationships Diagrams)*
* *Effort Recording (Excel file)*

***Important:***

* ***document risky/difficult/complex/highly discussed*** *requirements*
* *document decisions taken by the team*
* ***iterate****: do not spend more than 1-2 full days for each iteration*
* ***prioritize*** *requirements, scenarios, users, etc. etc.*

Project Rules and Evaluation Criteria

***General information:***

* *This homework will cover the 80% of your final grade (20% will come from the oral examination).*
* *The complete and final version of this document shall be* ***not longer than 40 pages*** *(excluding this page and the Appendix).*
* *Groups composed of five students (preferably).*

*I expect the groups to submit their work through GitHub*

***Use the same file to document the various deliverable.   
Document in this file how Deliverable “i+1" improves over Deliverable “i".***

**Project evaluation:**

*Evaluation is not based on “quantity” but on “quality” where quality means:*

* *Completeness of delivered Diagrams*
* *(Semantic and syntactic) Correctness of the delivered Diagrams*
* *Quality of the design decisions taken*
* *Quality of the produced code*

Table of Contents of this deliverable

Sommario

**Digitare il titolo del capitolo (livello 1) 1**

Digitare il titolo del capitolo (livello 2) 2

Digitare il titolo del capitolo (livello 3) 3

**Digitare il titolo del capitolo (livello 1) 4**

Digitare il titolo del capitolo (livello 2) 5

Digitare il titolo del capitolo (livello 3) 6

List of Challenging/Risky Requirements or Tasks

*<In this section, you should describe using the table below the most challenging or discussed or risky design tasks, requirements, or activities related to this project. Please describe when the risk arised, when and how it has been solved.>*

PLEASE FILL IN THIS TABLE AT EACH DELIVERABLE

|  |  |  |  |
| --- | --- | --- | --- |
| **Challenging Task** | **Date the task is identified** | **Date the challenge is resolved** | **Explanation on how the challenge has been managed** |
| Formato memorizzazione del dato; | 14/11/2018 |  |  |
| Gestione comportamento dei sensori (in caso di dati anomali); | 20/11/2018 | 30/11/2018 | Prevediamo che il Sistema potrà fornire un feedback al sensore sul suo stato così da regolarne la frequenza di invio dei dati |
| Trovare tool adeguati alla gestione delle fasi di progettazione; | 20/11/2018 |  | Project Libre è scomodo, troveremo un altro tool per il secondo deliverable |
| visualizzazione del giusto livello di dettaglio da parte di un gestore | 30/11/2018 | 03/12/2018 | È giusto che un gestore veda i dati di tutto il sistema e quindi di ogni singolo sensore? Ha senso che veda solo se nel suo livello inferiore vi sono anomalie |
| elaborazione del dato in base al livello di gestione | 30/11/2018 | 03/12/2018 | Il dato lo dobbiamo aggregare per poterlo vedere anche nei livelli superiori oppure abbiamo bisogno di un semplice alert di eventuali anomalie? |
| Rischio tutti i sensori aumentano la loro frequenza di invio | 30/11/2018 |  |  |
| **Challenging Task** | **Date the task is identified** | **Date the challenge is resolved** | **Explanation on how the challenge has been managed** |

A. Requirements Collection

*In this section, you should describe both the application* ***features/functional*** *requirements as well as the* ***non functional*** *ones. You shall also document* ***constraints*** *and* **rules***, if they apply.*

A.0 Detailed Scenarios

*<Describe in extreme details at least 3 scenarios. Use those scenarios to better understand what the system has to do in much detail. Use the information coming from this study to refine the functional requirements below>*

* Il gestore di distretto effettua il login, entrerà quindi nella schermata dedicata alla visualizzazione dello stato di tutti gli edifici appartenenti al suo ditretto;
* Il gestore di città effettua il login, entrerà quindi nella schermata dedicata alla visualizzazione dello stato di tutti I ditretti presenti nella sua città; ne compare uno dove sono presenti delle anomalie, la segnalazione delle anomalie avverrà tramite l’utilizzo di segnali colorati, può quindi entrare nel livello di distretto per poter capire quale è l’edificio che presenta anomalie;
* Il gestore di edificio effettua il login, entrerà quindi nella schermata dedicate alla visualizzazione delle aree in cui è suddiviso il suo edificio. Se in un’area è presente un’anomalia, questa verrà sengalata tramite l’utilizzo di segnali colorati. Può quindi entrare nel livello dedicato alla sua area per poter capire il tipo di anomalia rilevata dai sensori e dal sistema.

A.1 Functional Requirements

*<List, in bullet points, all the functional requirements your system shall implement>   
<Describe functional requirements and stakeholders through Use Case Diagrams>. <Then, prioritize them, and provide a table-based description of the most important requirements>*

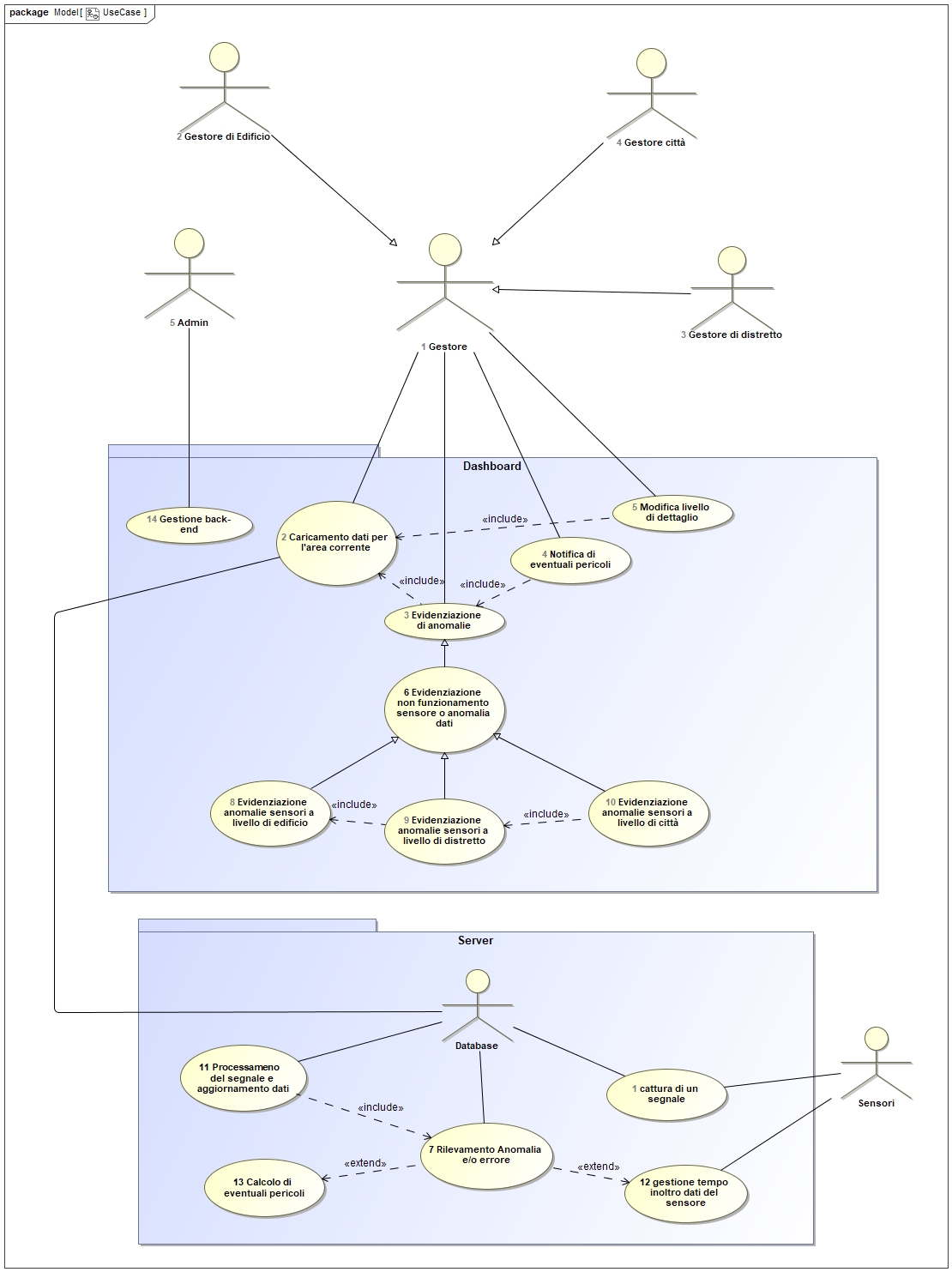
*<Provide a table-based description of the most important requirements, using the Alistair Cockburn Use Case Template>*

*PLEASE COPY HERE: (i) ALWAYS the diagram to be discussed, (ii) the text explaining the DECISIONS taken when creating the diagram (that is, do not spend time in EXPLAINING the details written in the diagram, but provide a small synthesis and focus on the decisions taken to create the diagram).*

PRIORITIZE THEM

FOR EACH FIGURE (Use Case D, Class, Sequence, etc) add a number and a label to it (e.g., Figure 1: Sequence diagram of the xxx scenario)

. Please do not submit the model alone: add a textual description for each model!!



Alcune note sul diagramma:

1. Questa prima versione comprende esclusivamente requisiti ad alto livello, i requisiti di sistema e i corrispondenti scenari/casi d’uso saranno introdotti nel secondo deliverable, ciò include le funzionalità lato server, qui riportato come semplice attore. (Motivazione: Ci sono stati grandi ritardi nella prima fase)
2. La gestione in back-end sarà gestita nel prossimo deliverable, quando daremo anche un modello per i dati.
3. Abbiamo preferito separare nei 3 livelli di astrazione (edificio, distretto, città) il caso d’uso per le anomalie dei sensori dal momento che i dati (dati singoli vs dati aggregati) e i processi coinvolti (threeshold basato su sensori di backup rimanenti vs threeshold basato su rapporto sensori funzionanti / sensori totali) sono troppo eterogenei per riassumerli sotto un unico use-case.
4. Le relazioni di inclusione tra i casi d’uso 10 -> 9 , 9 -> 8 saranno probabilmente eliminate in quanto i casi d’uso 9 e 10 utilizzeranno dati che sono stati prodotti dai casi d’uso 8 e 9 ma lo faranno in modo asincrono: i trigger per questi casi d’uso sono la ricezione di nuovi dati rispetto all’area monitorata, non la disponibilità di nuovi dati relativi alle anomalie sui sensori.
5. Le relazioni di inclusione tra casi d’uso 3 -> 2, 4 -> 3 indicano un trigger: L’aggiornamento dei dati in dashboard (#2) fanno scattare il controllo sulle anomalie (#3), mentre al termine del controllo sulle anomale (#3), se ne sono state rilevate, verrà effettuato il controllo sui possibili pericoli (#4); La relazione di inclusione tra casi d’uso 5 -> 2 è invece in senso stretto, dal momento che per focalizzarsi su una sotto-area, la dashboard farà richiesta al server dei dati relativi a quella sotto-area come 1-step routine.

**REQUISITI FUNZIONALI:**

* Cattura del Segnale;
* Interpretazione dei segnali;
* Memorizzazione dei dati;
* Caricamento/Aggiornamento dati in dashboard;
* Modifica del livello di dettaglio;
* Evidenziazione anomalie sensore;
* Evidenziazione dati anomali;

A1.1 GUI Requirements (da riempire a partire dalla Versione 2)

<*Please report here GUI-specific Requirements>*

A1.2 Business Logic Requirements (da riempire a partire dalla Versione 2)

<*Please report here Business-logic specific Requirements>*

A1.3 DB Requirements (da riempire a partire dalla Versione 2)

<*Please report here DB-specific Requirements>*

A.2 Non Functional Requirements

*<List and describe here the* ***most important*** *non functional requirements.>   
BE CAREFUL NOT TO MAKE CONFUSION AMONG DIFFERENT NON FUNCTIONAL REQUIREMENTS. PLEASE FOCUS ON NO MORE THAN THE 3-4 MOST IMPORTANT NON FUNCTIONAL REQUIREMENTS- PLEASE JUSTIFY THEM!!!*

**-Performance:**

* il sistema deve supportare lo storage e il processing di almeno 150.000 segnali al minuto;
* il sistema deve riuscire a gestire l'accesso in parallelo di 50 gestori;

**-Scalabilità**: il sistema deve supportare future espansioni;

**-Affidabilità**: Il sistema deve essere disponibile H24 365/anno;

**-Usabilità:** I gestori devono riuscire di usare il sistema senza nessuna formazione specifica;

Note:

1. Il requisito di Affidabilità lo abbiamo assunto dal momento che trattandosi di un sistema di monitoraggio ambientale che deve notificare eventuali pericoli, dovrà essere minimizzato il tempo offline.
2. Nel caso peggiore, il requisito di performance sui segnali da processare può raggiungere un picco di 300.000 segnali al minuto (infatti supponiamo la frequenza con cui i sensori inviano dati raddoppi in caso di anomalia)

A.3 Excluded Requirements

*<List and describe here the* ***most important******excluded*** *functional requirements.>*

* **Installazione dei sensori:** se ne occupa un’altra organizzazione, noi dobbiamo solamente assicurarci che i sensori di cui l’ID non sia ancora registrato in DB vengano aggiunti automaticamente alla ricezione del primo segnale a loro corrispondente.
* **Sostituzione / riparazione di sensori guasti o malfunzionanti:** ci è richiesto soltanto di notificare le anomalie sulle dashboard dei gestori.

A.4 Assumptions

*<Briefly document, in this section, the most relevant requirement assumptions/decisions you had to made during your project>*

* I sensori avvertiranno di essere guasti per qualsiasi tipo di problema che si presenti, anche se i dati continuano ad essere veritieri
* I 3 tipi di gestore saranno distinti al momento del login sul sistema
* Chi installa i sensori provvederà ad assegnarli gli ID strutturati nel modo seguente: <ID città><ID distretto><ID edificio><ID interno>
* I sensori non omnidirezionali (come ad esempio i sensori di luminosità) andranno installati tenendo conto degli standard previsti rispetto alla zona in cui appartengono (ad esempio i sensori di luminosità per ambienti esterni devono essere puntati in modo che la luce solare non causi “false” anomalie.)

***A.5 Prioritization***

<List here all the requirements, in prioritized order>

1. **Rilevazione anomalie**: ci sono ancora molti punti da chiarire sulle responsabilità dei sottosistemi e sull’aggregazione dei dati provenienti dai livelli di astrazione inferiori, specialmente per le anomalie dei sensori.
2. **Performance**: nel gruppo c’è tendenza ad ottimizzare, per questo lo piazzo al primo posto tra i requisiti non funzionali.
3. **Scalabilità:** è forse uno dei requisiti non funzionali più complicati per noi dal momento che richiede da parte nostra uno sforzo aggiuntivo per lo studio di tecnologie che rendano il sistema scalabile (come ad esempio l’utilizzo di un database non relazionale, fin’ora mai trattati da nessun membro del gruppo).
4. **Cattura e processamento di un segnale:** c’è una questione aperta su come i sensori possano sapere di essere in uno stato anomalo o meno così da aggiustare la loro frequenza di invio dei segnali, si è valutato di introdurre un feedback dal sistema nello UseCase relativo a questo requisito.
5. **Tutti gli altri requisiti:** stileremo una lista aggiornata al prossimo deliverable, che includerà anche gli eventuali requisiti di sistema

B. Software Architecture   
*<Report here both the static and the dynamic view of your system design, in terms of a Component Diagram, Class Diagrams and their related Sequence Diagrams >*

C.1The static view of the system: Component Diagram

*AVOID TO MAKE IT TOO COMPLEX AND FINE GRAINED. FOCUS MORE ON “HOW”*

*THE COMPONENTS SHALL INTERACT IN ORDER TO SATISFY THE REQUIREMENTS*

*ADD INTERFACES and their parameters*

C.2 The dynamic view of the software architecture: Sequence Diagram

*BE SURE THAT THE STRUCTURE IS SYNCHRONIZED WITH THE DYNAMIC VIEW*

C. ER Design

<Report here the Entity Relationship Diagram of the system DB>

D. Class Diagram of the implemented System

E. Design Decisions   
<Document here the **5** most important design decisions you had to take. You can use both a textual or a diagrammatic specification.>

THIS IS A VERY IMPORTANT PART. BE SURE TO DOCUMENT THE 5 MOST IMPORTANT DECISIONS (related to your requirements and design) YOU MADE

F. Explain how the FRs and the NFRs are satisfied by design

*<Report in this section how the design you produced satisfies the FRs and the NFRs>*

G. Effort Recording



***PERT****Make a PERT documenting the tasks and timing you expect to spend on the deliverable. Try to be as precise as possible. Check, after the deliverable deadline, if and how you satisfied (or not) the deadlines.*

***Logging*** *As you are working on the assignment, record what you are doing and how long you spent. As a rule of thumb, you should add a log entry every time you switch tasks. For example, if you do something for two hours straight, that can be one log entry. However, if you do two or three things in half an hour, you must have a log entry for each of them. You do not need to include time for logging, but should include the time spent answering the other parts of this question.*

*For this purpose, please use the* ***LogTemplate.xls*** *file.*

***Categorization*** *When logging the time spent on the project, please create different sub- categories.**Specifically, it is important to clearly distinguish between two main categories: the time spent for “****learning****” (the modeling languages, the tools, etc.) from the time needed for “****doing****” (creating the models, taking the decisions, …). Learning tasks are in fact costs to be paid only once, while doing costs are those that will be repeated through the project.*

*For each category, please define sub-categories. Examples follow. You may add other sub-categories you find useful.*

|  |  |
| --- | --- |
| ***Learning***   * ***Requirements Engineering*** * ***Non functional Requirements*** * ***Use Case Diagrams*** * ***Tool study*** | ***Doing:***   * ***Requirements discovery*** * ***Requirements Modeling (UC diagrams)*** |

***Summary Statistics****Based on the attributes defined above, calculate the summary statistics of the time spent for “learning”, the time spent for “doing”, and the total time.*

***Note: this Deliverable report shall document only the Summary Statistics for the different deliverables (D1, D2, and Final). Detailed information shall be reported in the Excel file.***

***COPY HERE (computed from the spreadsheet): i) the total number of hours spent by the group (that is, hours per task X number of people working on that task), ii) the time spent for LEARNING and for DOING***

Appendix. Code   
*<Report in this section a* ***documented*** *version of the produced code. I do not need a copy&cut of your code, but rather an explanation of how your code satisfies the Functional and Non functional requirements.  
<Show some screenshots of the code behavior>*

*<please upload your executable code in the dropbox folder>*