POLITECNICO

MILANO 1863

Computer Science and Engineering

Software engineering 2

Data4Help DD

Design Document

Version 1.0

14/11/18

Diego Piccinotti, Umberto Pietroni, Loris Rossi

Table of Contents

1 Introduction

1.1 Purpose

The purpose of this document is to further analyze the design and architectural choices for the system to be. Whereas the RASD presented a general view of the system and its features, this document will detail those concepts by showing components of the system, its run-time behavior, deployment plan and algorithm design.

The document will therefore contain a presentation of:
- Overview of the high-level architecture
- Main components and their interfaces provided one for another
- Runtime behavior
- Design patterns
- Algorithm design of the most critical parts of the application
- Implementation plan
- Integration plan
- Testing plan

1.2 Scope

(An SDD shall identify the design stakeholders for the design subject.
An SDD shall identify the design concerns of each identified design stakeholder.
An SDD shall address each identified design concern.

Stakeholders are those people, groups, or individuals who either have the power to affect,
or are affected by the endeavor you're engaged with.
)

1.3 Definitions, Acronyms, Abbreviation

1.3.1 Definitions

- **User**: individual who allows *Data4Help* to monitor his location and health status.
- **Third party**: individual or organization registered to *Data4Help* which can request users' data.
- **Data collection**: gathering of users' data through a wearable device.
- **Anonymized data**: data about more than 1000 users whose personal information has been previously removed so that they are not directly relatable to the system's users.
- **Elderly**: user who is subscribed to *AutomatedSOS* and is older than 60 years old.
- **Risk threshold**: Set of boundary health parameters defined for each elderly. If monitored values of the user's health parameters get below these boundaries, an SOS request is placed to an external ambulance provider.
- **Athlete**: user who participates in a run.
- **Run organizer**: third party who can manage runs for athletes.
- **Spectator**: non-registered individual who follows a run through a map with runners' positions.
- **Wearable**: a personal device provided with biometric sensors and GPS given to each user for free after the registration process.

1.3.2 Acronyms

- **API**: Application Programming Interface
- **DB**: Database
- **DBMS**: Database Management System
- **DD**: Design Document
- **GUI**: Graphical User Interface
- **RASD**: Requirements Analysis and Specifications Document
- **GPS**: Global Positioning System
- **GSR**: Galvanic Skin Response

1.3.3 Abbreviations

- **Gn**: n^{th} goal.
- **Dn**: n^{th} domain assumption.
- **Rn**: n^{th} functional requirement.
- **Rn-NF**: n^{th} nonfunctional requirement.

1.4 Reference Documents

- IEEE Standard on Software Design Descriptions (*IEEE Std 1016-2009*)

1.5 Document Structure

2 Architectural Design

2.1 Overview:

(High-level components and their interaction)
The main architecture on which this system will be based upon is the three-tier
architecture. The presentation tier is re
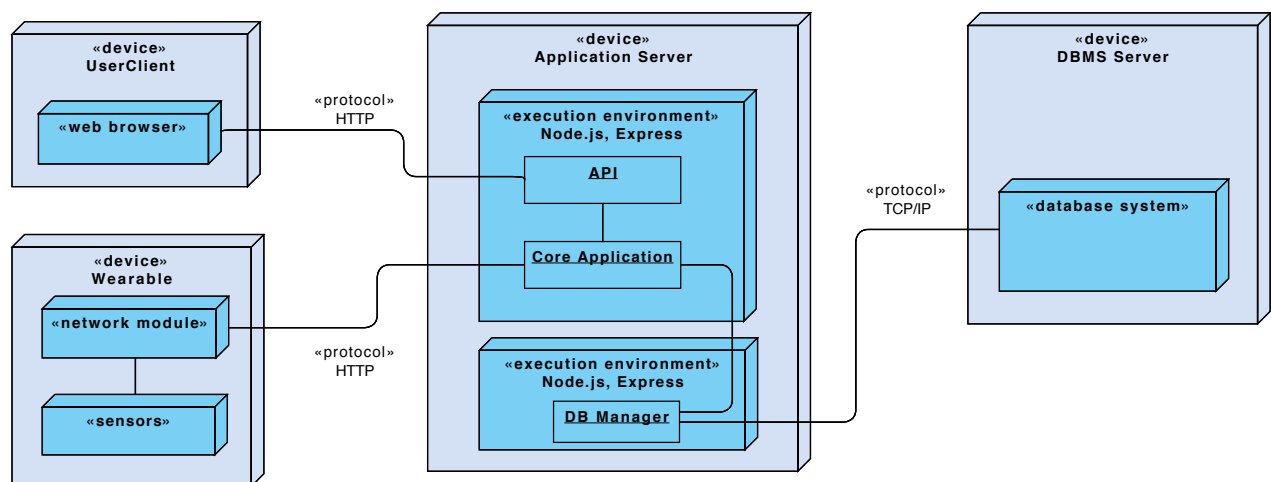
2.2 Component View

2.3 Deployment View

The deployment diagram shows the physical topology of the system.
We can see the three tiers architecture explicitly.

In the Application Server, there is a dedicated node to deploy the DB Manager. This way it
is possible to maintain and update the DB Manager without shutting off the whole
Application Server.

(more to be added)

2.4 Runtime View

(You can use sequence diagrams to describe the way components interact to accomplish specific tasks typically related to your use cases)

2.5 Component Interfaces

2.6 Selected Architectural Styles And Patterns

(Please explain which styles/patterns you used, why, and how)

2.7 Other Design Decisions

3 User Interface Design

(Provide an overview on how the user interface(s) of your system will look like; if you have included this part in the RASD, you can simply refer to what you have already done, possibly, providing here some extensions if applicable.)

4 Requirements Traceability

(Explain how the requirements you have defined in the RASD map to the design elements that you have defined in this document.)

G1: The user can be recognized by providing a form of identification.

- DataCollector **[R2]**
- User

G2: Allow third parties to monitor data about location and health status of individuals.

G3: Allow third parties to access data relative to specific users.

G4: Allow third parties to access anonymized data of groups of users.

G5: Allow third parties to offer a personalized and non-intrusive SOS service to elderly people so that an ambulance arrives to the location of the customer in case of emergency.

G6: Allow athletes to enroll in a run.

G7: Allow organizers to manage runs.

G8: Allow spectators to see on a map the position of all runners during the run.

5 Implementation, Integration And Test Plan

(Identify here the order in which you plan to implement the subcomponents of your system and the order in which you plan to integrate such subcomponents and test the integration.)

6 Effort Spent

(In this section you will include information about the number of hours each group member has worked for this document.)

6.1 Piccinotti Diego

Description of the task	Hours
Purpose, Scope, Definition	1
High-level Components and their Interaction	3
Component View	2
Deployment View	
Runtime View - Sequence Diagrams	
Selected Architectural Styles and Patterns	
Component Interfaces	
Algorithm Design	
User Interface Design	
Requirements Traceability	
Implementation, Integration and Test Plan	

6.2 Pietroni Umberto

Description of the task	Hours
Purpose, Scope, Definition	1
High-level Components and their Interaction	3
Component View	2
Deployment View	
Runtime View - Sequence Diagrams	
Selected Architectural Styles and Patterns	
Component Interfaces	
Algorithm Design	
User Interface Design	
Requirements Traceability	
Implementation, Integration and Test Plan	

6.3 Rossi Loris

Description of the task	Hours
Purpose, Scope, Definition	1
High-level Components and their Interaction	3
Component View	2
Deployment View	1.5
Runtime View - Sequence Diagrams	
Selected Architectural Styles and Patterns	
Component Interfaces	
Algorithm Design	
User Interface Design	
Requirements Traceability	
Implementation, Integration and Test Plan	

7   References