Handling Nonfunctional Requirements for Smart Cities

Isabel Sofia Brito[1[0000-0002-7556-4367]], Ana Moreira[2[0000-0003-2046-2766]]
João Araújo [3[0000-0001-5914-1631]]

[1] UNINOVA-CTS, Caparica, Polytechnic Institute of Beja, Beja, Portugal
isabel.sofia@ipbeja.pt

[2,3] NOVALINCS, Computer Science Department, Universidade NOVA de Lisboa, Portugal
[amm|joao.araujo]@fct.unl.pt

Abstract. The complexity of large-scale systems combined with their intrinsic Non-Functional Requirements (NFRs) (e.g., interoperability, scalability) constitute some of the challenges imposed by smart cities goals. Several proposals exist for eliciting and specifying NFRs, but not to deal with the smart cities core elements such as cloud-computing, physical devices and communication issues. Given the importance and complexity of NFRs, and their often conflicting nature, this research describes a Requirements Engineering approach to guide the elicitation and specification of NFRs based on a smart cities' framework. This approach to handle NFRs was developed in the context of the U Bike project.

Keywords: NFRs, smart cities framework, requirements specification.

1 Introduction

The goal of a smart city is to optimize technologies to create a sustainable and self-aware city that improves both the quality of governmental services and the citizens welfare. Thus, a city with numerous ICT (Information and Communications Technology) projects is not necessarily a smart city [1]. One of the challenges is to deal with the NFRs associated with the smart city core elements, such as cloud computing, physical devices, and communication. In the era of cloud computing, one of the challenges software providers may face is to accurately capture the real requirements from stakeholders, be them functional (e.g., software services) or non-functional (e.g., service availability). The data collected from an Internet of Things (IoT) platform, typically produced by a huge number of physical devices, can be stored and processed in a cloud environment using big data [2]. This highlights the need to support NFRs such as scalability, or interoperability to cope with the data exchanged between multiple elements. Finally, an effective communication between different network nodes (e.g., user devices, sensors, actuators) is a mandatory requirement for a smart city system.

In summary, NFRs play a significant role in a smart city system. When smart city stakeholders are looking for a service, typically they have some functional requirements in mind, as they may not be familiar with qualities and constraints. Given the importance and complexity of NFRs, and their often conflicting impact on other smart city system requirements, it is important to reason about NFRs early in the development. Some works study the different types of NFRs relevant to smart cities systems (e.g. [3] [4] [5] [6] [2]), but they do not provide a process to specify NFRs. This is why we are proposing a Requirements Engineering (RE) approach based on the smart city framework [7] to specify and reason about NFRs. This approach contributes to an understanding of a smart city system goals, by identifying stakeholder needs, specifying NFRs, and managing conflicting situations triggered by negative contributions among NFRs or between those and other system requirements. Moreover, our approach bridges the gap between NFRs analysis and design, as the framework guides the identification and specification of requirements based on the two fundamental smart city system parts, **Structure** and **Operation**. The **Structure** part describes four different components: Physical Layer, Communication Layer, User Applications, and External Services. The **Operation** part consists of three different phases used by the smart city components for interactions: collecting; transmitting; and processing, managing and utilization.

The rest of this paper is organized as follows. Section 2 describes the smart city framework used to guide our requirement approach. Section 3 presents a RE approach to identify and specify NFRs, and to identify conflicts. The process is explained using parts of the U-Bike project. This project aims to promote sustainable mobility behavior by offering electrical and conventional bikes to academic communities. Section 4 discusses related work, highlighting similarities and differences with our proposal. Finally, Section 5 concludes and suggests directions for future work.

2 The smart city framework

The smart city IoT framework guides the development of a smart city system [7] and is composed of the **Structure** and **Operation** parts. Fig. 1 summarizes the elements of the two parts of this framework.

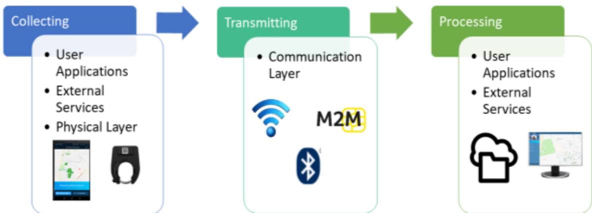

Fig. 1. Smart city framework components, from [7].

The Structure is composed by the Physical Layer, Communication Layer, User Applications, and External Services. The Physical Layer consists of many types of sensors (e.g., ultrasonic, pressure, proximity), cameras, smartphones and tablets, for example, through which the required data (e.g., environmental data) are gathered. The heterogeneity of the equipment coexisting in this layer reinforces the need for interoperability. The Communication Layer allows the transmission of data, between framework components. The User Applications component collects and manages data.

It is one of the systems’ front ends, providing a range of possible applications to achieve the stakeholders’ goals. Finally, the External Services are functionalities provided by external entities responsible for processing and managing the obtained data to support smart city requirements and stakeholders' goals (e.g., an interface between the Physical Layer and the User Applications, or the means to store and process large amounts of data collected by the Physical Layer and the User Applications).

The Operation consists of three phases, Collecting, Transmitting, and Processing, Managing and Utilization. Collecting senses the physical environment, collecting real-time and non-real-time data, and constructing a general perception of it. Data acquisition is performed by using different technologies and devices from the Physical Layer. The Physical Layer and the User Application component achieve the objectives of this phase. Transmitting includes mechanisms to “send” the data collected by the Physical Layer to the User Applications or the External Services, and from/to the User Applications and the External Services. Finally, the processing, managing and utilization phase uses the User Applications and the External Services components to process and analyze data/ information flows, and to provide feedback to control the applications. It is also responsible for critical tasks such as device discovery, device management, data management (filtering and aggregation).

3 The NFR RE Approach for Smart Cities

The NFR RE approach described here was created to handle NFRs in the context of the U-Bike project[1]. U-Bike provides classic and electric bicycles to the academic community (e.g., students, teachers and academic staff) in the Instituto Politécnico de Beja. U-Bike encourages a sustainable urban transportation in the city of Beja, Portugal, by fostering healthier lifestyles while reducing the CO2 footprint in the environment and saving money. The project is coordinated by the Institute of Mobility and Transport, I.P. (IMT), and is co-funded by the Portugal 2020 program, through the Operational Programme for Sustainability and Efficient Use of Resources (POSEUR). A small subset of the U-Bike requirements, used here for illustration purposes only, are:

- The bike should be used by academic users.
- The system should have the means to assess bike monitoring and maintenance (by means of bike data), in particular electric bikes.
- The system must analyze the effects of the project regarding its objectives, targets and expected results (metrics) by means of reports. These metrics are CO2 reduction, km travelled per week by each user (i.e. user and ride information), adherence rate of the academic community population to the project (by type of target audience) and the gain of healthy habits.
- The system must be available 24/7.

The approach follows an iterative and incremental process for each of the components of the Structure. It is composed of three main tasks: Elicit NFRs, Specify NFRs, and Identify conflicts.

[1] https://www.u-bike.pt/sobre/

Elicit NFRs (task 1). This task gathers NFRs from different sources, including stakeholders' opinions, to understand the system goals from different perspectives. To perform this task, we need to identify sources of domain documentation, the relevant stakeholders, and reuse information from existing catalogues and frameworks.

Identify sources (task 1.1). The smart city adoption is expected to be strongly driven by the city needs and its stakeholders. This task collects information and documentation about smart city goals, such as process models or standards, catalogues and user (or other) manuals of existing systems to help with the identification of requirements. In the U-Bike project, the main information was collected from IMT and POSEUR documentation[2].

Identify stakeholders (task 1.2). Because the RE process focuses on the stakeholders' needs, we can use the questions in [8] to identify all the persons, organizations and other systems that have a direct or indirect interest in the system under study. The U-Bike stakeholders are academic users (students and teachers), academic institution (IPBeja) who is the data owner and collects incomes (receives funds from POSEUR), and project managers (academic staff) that are involved in service monitoring, collecting incomes and managing the services. Stakeholders not belonging to the academy are project coordinator (IMT and POSEUR) who is involved in service performance measurement and funding the project, communication sector representatives and service providers or cloud service providers who sell services, and local administration (Beja municipality) who collects incomes.

Reuse knowledge from catalogues and frameworks (task 1.3). This task guides the stakeholders to identify NFRs based on existing projects, frameworks and catalogues, such as NFR catalogues [9] and the SynchroniCity technical framework [10]. Such catalogues and frameworks promote reusability. According to the U-Bike stakeholders, Interoperability, Scalability, Privacy, Security, and Availability are the NFRs this project should consider.

Specify NFRs (task 2). For each NFR, this task identifies responsibilities, contributions to other NFRs, and stakeholders' priorities.

Identify responsibilities (task 2.1). This task provides a description of the NFR intended behavior. For each framework Structure components (Physical Layer, Communication Layer, User Applications, and External Services), it is necessary to identify the required NFRs. Each NFR is described using five elements: a keyword to describe the NFR in upper case; the name of the component element; the representation of preference of the NFR ("must" for obligation, "could" for strong suggestion, "should" for suggestion without imposing or "will not" be accomplished). The action (collect, transmit, process, manage and execute, based on Operation phases of the smart city framework). Finally, the core elements handled by the system in terms of data, applications and services. The NFR description structure is:

```
<keyword> <component element> <must|could|should|will not>
<collect, transmit, processing, execute> <data, service, application>
```

Note that the elements do not follow any predefined order. The important thing is to make sure that the responsibility makes sense. Let us consider the Interoperability NFR. It needs to be *satisficed* [9] because the system consists of **heterogeneous**

[2] https://poseur.portugal2020.pt/pt/candidaturas/avisos/poseur-07-2015-31-aviso-projeto-u-bike-portugal/

devices, services and applications from different sellers and service providers using different communication technologies and formats for data exchange [11]. Using the above elements, the responsibilities are:

IntOp_Resp01: `HETEROGENEOUS smartphones (Physical Layer) must collect and transmit user bike data`

IntOp_Resp02: `HETEROGENEOUS user apps (User Applications) must collect, transmit, and process user bike data, ride data`

IntOp_Resp03: `Cloud computing (External Services) could collect, process bike data, user data`

IntOp_Resp04: `The network (Communication Layer) must transmit user bike, bike and ride data between bike, user app, project management web site, cloud computing`

Now, let's focus on Scalability, as the system must **handle a growing number of** resources and requests, due to the volume of sensor data flowing, the volume of data to be stored in databases, the number of (heterogeneous) devices handled by the management system, the amount of data processed by services and applications (cloud computing), and the number of applications and users. Therefore, scalability refers to the ability of being extensible without negatively affecting the quality of the service [11]. Some scalability responsibilities are:

Scale_Resp01: `The project management website (User Application) should HANDLE A GROWING NUMBER OF processing bike, user bike, ride data`

Scale_Resp02: `The project management website (User Application) should HANDLE A GROWING NUMBER OF execute user application`

Scale_Resp3: `The cloud computing (External Services) must HANDLE A GROWING NUMBER OF collecting, processing bike data, user data`

As a final example, let us look at Privacy, important because the platform collects, transfers, stores, and processes sensible data (e.g., personal information related to their habits and interactions with other people and services) from the city and citizens. Data **protection** and privacy issues should be addressed in several levels, from the physical layer (devices and sensors) to specific end-user applications. Some responsibilities are:

Priv_Resp01: `User app (User Applications) should PROTECT user bike personal data WHEN collect, transmit`

Priv_Resp02: `Project web site (User Applications) must PROTECT user bike personal data WHEN collecting, processing`

Priv_Resp03: `Cloud computing (External Services) must PROTECT user bike personal data WHEN collecting, transmitting, processing`

Priv_Resp04: `The network (Communication Layer) must protect user bike personal data DURING communication between User app, Project website, cloud computing`

Identify contributions between NFRs (task 2.2). NFRs impact each other. This is expressed by a contribution relationship that can be collaborative (or positive, helping the affected NFR) and represented by a "+" sign, or damaging (or negative, obstructing the affected NFR) and represented by a "-" sign. Contribution relationships maybe have different strengths (e.g. + and ++, or - and - -). Some of these relationships can be found in catalogues (e.g., [9] [12]). In the U-Bike context, Interoperability contributes negatively to Privacy.

Identify stakeholders' priorities (task 2.3). A priority expresses a stakeholder's importance to an NFR. This is relevant for conflict solving. We use a qualitative 5-values scale: Very Important (the stakeholder cannot do without it), Important (the stakeholder does not want to be without it), Medium (the stakeholder would appreciate it), Low (the stakeholder would accept its absence), Very Low (the stakeholder accepts its absence) and Don't Care (the stakeholder does not need it). In the U-Bike, the stakeholders allocated a low priority to Scalability because the number of participants is known (corresponds to the number of bikes). On the other hand, they consider Privacy to be very important to comply with the General Data Protection Regulation (GDPR), and similarly to Interoperability.

Identify conflicts (task 3). A conflict occurs any time two or more NFRs contribute negatively to each other, and are needed by the same framework component. For example, we identify a conflict between Interoperability and Privacy with respect to the user applications and external services components. To help handling this our stakeholders set a high priority to Privacy with respect to the User Applications, considering data protection, and high priority to Interoperability with respect to External Services, considering the heterogeneity of resources.

To demonstrate how these requirements were addressed, we present a brief description of the implemented system based on the framework components:

1. Physical Layer: the system has a diverse range of devices, such as iOS and Android smartphone and a SmartLock, from different sellers cooperating to achieve common goals. Therefore, the Interoperability NFR was ranked as Very Important and special care was taken to *satisfice* it (see points 3 and 4).
2. User Applications: the system has two applications for the stakeholders. The IPBeja's project managers are responsible for the system and use the U-Bike web functionalities. The academic users "own" the bike and use the mobile application (app) functionalities (see Fig. 2, right) to: start a ride by inserting her "bike number" or scanning a QR Code placed on the bike (see Fig. 2, left). This operation allows users to automatically unlock the bike; view data for each ride (bike ID, distance travelled and date) and the respective paths represented on the city map (see Fig. 2, middle). The app and web site use encryption to guarantee data privacy.

Fig. 2. (left) Bike with a SmartLock device' collects user location data and ride data; (middle) Bike monitoring system: Reservation; Tracking; Monitoring; (right) Mobile App: Unlock bike; Manage rides; Visualize rides.

3. External Services: the system uses the google API for location and AWS cloud-computing platform, which is a full management service at any scale. Interoperability and scalability are satisficed by the AWS platform.

4. Communication Layer: the system must support communications between all the components elements. Interoperability requirements are accomplished by different technologies: Wi-Fi, Bluetooth, GPRS/3G/4G, LTE and M2M. For example, the SmartLock device along with communication technologies (M2M and Bluetooth) are used to (un)lock the bike.

In summary, the NFRs can be supported in different ways for each component of a smart city system (e.g. Interoperability) and can affect design decisions. It is therefore important to have an approach that identifies, describes and reasons about NFRs early in the software development process to guide the decision process.

4 Related Work

In [5], the authors propose a template and checklist to analyze quality attributes (i.e. NFRs) for IoT based on a traffic management system. Due to the diverse set of NFRs, it is difficult to bridge the gap between requirement analysis and software design. This work inspires the identification task of our process. The works that follow helped us to identify relevant NFRs to support the specification task of our proposal. For example, the goal in [3] performed a literature review and identified 17 NFRs. Security, Scalability, Privacy are the most cited NFRs. This study also indicates that requirements are specified for different components of smart city systems such as middleware, software platforms, data solutions, and business components. Similarly, in [6] the authors present a survey to discover relevant NFRs for smart city systems. They identify that the most important NFRs are: Interoperability, Usability, Authentication and Authorization, Availability, Recoverability, Maintainability, and Confidentiality. They also propose a set of questions to identify domain stakeholders, where Municipality, Private and Public sector, Government, service provider are the most cited actors in the smart city context. These questions inspired our proposal for the stakeholders' identification. Identifying stakeholders and priorities and creating policies are one of the challenges faced in the smart cities initiatives. Jayasena et al [8] show how to identify stakeholders and their positive or negative impact to the smart cities. The authors relate them (and theirs impact) with stages of smart city development. Finally, [2] analyzes the functional requirements and NFRs extracted from the 23 software platforms for smart cities. This paper shows that NFRs for smart city systems, such as scalability, adaptation, and interoperability, are related to large, heterogeneous distributed systems. Other NFRs are related to the manipulation of critical and personal data from citizens, such as security and privacy.

The main difference between the approaches presented here and our proposal is the NFR specification task. The NFR specification is complete and consistent as it is described considering all the components of a smart city. This allows us to obtain an integrated perspective of the system.

5 Conclusions and Future Work

This paper presents an approach to handle NFRs in the smart city system context during requirements engineering. First, it proposes the use of catalogues to help the identification and specification of NFRs. The catalogues we have mainly used are the NFR framework and SynchroniCity technical framework. Second, it specifies NFRs by using a set of rules based on the smart city framework. Third, it identifies conflicting situations during the specification phase. The process described in this paper emerged from our partnership in the U-Bike project, these ideas are at an early stage of research. Therefore, we need to validate it further in other case studies. For future work, we need to define the best approach to resolve conflicts, for example how to deal with three or four NFRs simultaneously, with different contributions among them; refine NFRs to be mapped onto functional requirements; and handle sustainability issues (economic, environmental and technological) considering that one of the smart city goal is to optimize modern, useful technologies to create a sustainable and self-aware city.

Acknowledgment

This work was partially financed by Portuguese Agency "Fundação para a Ciência e a Tecnologia" (FCT), in the framework of project UID/EEA/00066/2019, and NOVA LINCS (ref. UID/CEC/04516/2019).

References

1. P. Neirotti, et al, "Current trends in Smart City initiatives: Some stylised facts", *Cities,* vol. 38, pp. 25-36, 2014.
2. E. Santana, et al, "Software Platforms for Smart Cities: Concepts, Requirements, Challenges, and a Unified Reference Architecture", *ACM Computing Surveys (CSUR),* vol. 5, pp. 1-37, 2018.
3. V. Bastidas, M. Helfert and M. Bezbradic, "A Requirements Framework for the Design of Smart City Reference Architectures", in *HICSS-51 - Hawaii International Conference on System Sciences*, Hawaii, 2018.
4. M. Maggio, et al, "SynchroniCity: Delivering an IoT enabled Digital Single Market for Europe and Beyond", 2018.
5. S. N. Mahalank, K.B. Malagund and R.M. Banakar, "Non-Functional Requirement Analysis in IoT based smart traffic management system", in *International Conference on Computing Communication Control and Automation (ICCUBEA)*, India, 2016.
6. G. Kakarontzas, et al, "A conceptual enterprise architecture framework for smart cities: A survey-based approach", *11th International Conference on e-Business (ICE-B),* Austria, 2014.
7. I.S. Brito, et al, "Leveraging IoT Framework to Enhance Smart Mobility. The U-Bike IPBeja Project", in *Smart Systems Design, Applications, and Challenges*, IGI Global, 2020.
8. N. Jayasena, et al, "Stakeholder Analysis for Smart City Development Project: An Extensive Literature Review", *MATEC Web Conf,* p. 266 06012, 2019.
9. L. Chung, et al, Non-Functional Requirements in Software, Kluwer, 2000.
10. SynchroniCity, "Technical Framework", 2019. Available: https://synchronicity-iot.eu/tech/.
11. S. Ravidas, et al, "Access control in Internet-of-Things: A survey", *Journal of Network and Computer Applications,* vol. 144, pp. 79-101, 2019.
12. N. Nurmuliani, D. Zowghi, D. Mairiza, "Towards a Catalogue of Conflicts Among Non-functional Requirements", *Fifth International Conference on Evaluation of Novel Approaches to Software Engineering* (ENASE), pp. 20-29, 2010.