A bit of "Persona", a bit of "Goal", a bit of "Process" ... a recipe for Analyzing User Intensive Software Systems

Chiara Di Francescomarino, Chiara Leonardi, Alessandro Marchetto, Cu D. Nguyen, Nauman A. Qureshi, Luca Sabatucci, Anna Perini, Angelo Susi, Paolo Tonella, and Massimo Zancanaro

Fondazione Bruno Kessler, Povo, Trento, Italy
{dfmchiara,cleonardi,marchetto,cunduy,qureshi,sabatucci, perini,susi,tonella,zancana}@fbk.eu

Abstract. The centrality of users in the design and development of complex systems, such as service-based applications, calls for new methodologies and techniques to extract and represent user needs and to translate them into real processes.
In this short paper, we describe the integration of concepts and analysis techniques of different approaches, namely Goal-Oriented Requirements Engineering, User-Centred Design and Process-Oriented Modeling, that are being developed in the context of two projects related to Ambient Assisted Living and Internet of Services.

Key words: Goal-Oriented paradigm, User-Centred Design, Requirements Engineering, Business Process Modeling

1 Introduction

The central role of the users in the design and evolution of complex systems, such as service-based applications for the Internet of Services (IOS[1]) or Ambient Assisted Living systems, has been widely recognized in the last years [1, 2]. Thus, to stress this aspect, we refer them as "user intensive" systems.

Goal-oriented Requirements Engineering (GORE) plays a fundamental role in the development of this kind of systems, enabling reasoning about the domain features with the aim of identifying conflicts and of checking for validity of functional and non-functional requirements. This technique has been exploited in the ACube project[2], whose goal is to study technologies for monitoring complex environments that can be applied in areas such as assisted living homes to help personnel, as well as to support the independence and safety of users.

Moreover, the key for the operationalization of goal-oriented system requirements in terms of services is the definition of a set of rules for associating process modeling concepts to goal-oriented ones. This enables designers to trace business

[1] http://www.future-internet.eu
[2] ACube is funded by the Autonomous Province of Trento. http://acube.fbk.eu/

processes of service-based systems back to intentional elements of the user model (e.g, goals, preferences, roles). The experience in this area comes from the Internet of Service project (IOS[3]), whose aim is to push the "Internet of Services" for real services, rather than for software services. Studies in different areas are conducted within this project, such as the area of service usage, representation, engineering, and delivery.

Nevertheless, we experienced that the two approaches, more than sharing the language of goals, require an effective way to center the design on the users of the system. The aim of this short paper is to propose an integrated methodology in which goal-oriented analysis [3], user-center design [4] and process modeling [2] may cooperate for a continuous communication between requirements engineers, stakeholders and designers, thus reducing the risk of misunderstanding the domain, missing important requirements, and resulting in an increase of the final value of the product.

2 From users' needs to requirements

The strength of the Goal-Oriented techniques in modelling the domains can be still enhanced by coupling the engineering perspective with a creative perspective typical of User-Centred Design approaches. The *User Centered Design* exploits a series of well-defined methods and techniques coming from social sciences and psychology for analysis, design, and evaluation technologies. Contextual inquiries, personas and scenarios are widely employed for obtaining a rich picture of the context (organizational, social, physical), and easily communicating it to stakeholders in order to envision acceptable and innovative technological solutions. Additional values emerge from this collaboration at different phases of the process.

Our proposal is to integrate Tropos [3] with user-centred design techniques in order to guide requirements engineering teams toward effective collection of user requirements and the envisaging of the design of complex software system infrastructures, while helping to fill the gap between end users and developers. Basic principles of this integration are: (i) early focus on users, tasks and environment, (ii) the active involvement of users in the design process, (iii) allocation of functions between user and system, (iv) the incorporation of user-derived feedback into system design, (v) iterative design whereby a prototype is designed, tested and modified.

The result of this integration is a process that encapsulates activities from both the two methodologies, promoting a very close collaboration between teams and an easy exchange of data. The process begins with the *investigation of the domain* in order to understand the organizational setting of the domain and to derive possible needs and services that the system could provide to users. The contextual inquiry produces a rich collection of data in a narrative format. The *data interpretation* phase provides the necessary abstraction to create a

[3] A joint Research project at FBK IRST-CIT. http://se.fbk.eu/en/node/15

believable model of the domain, but avoiding to loose important details typical of a narrative analysis. Tropos early requirement plays a central role at this step by providing the semi-formal language for describing the domain. The *data consolidation* filters data to focus on relevant characteristics: activity scenarios and personas allow designers to have insight the system, providing an anchor to the real domain and the real users. Finally, the *envisioning phase* lets the analysis team to reason on the system-to-be in order to expand designers' prospective, to look at the problems from different points of view, to figure out how their ideas can work in a real context, to identify design criticalities, and finally, to generate requirements. Brainstorming and other creative techniques are used with the result to shift from the Tropos early requirement phase toward the late requirement phase, thus obtaining a list of requirements for the system. The process terminates with the validation of requirements with customers, essential for evaluating the value of the services the system will provide [1].

The key for the integration is the use of *Scenarios* and *Personas* along the whole process. Whereas the use of Scenarios in RE is pretty established as an instrument to describe instances of behavior of the system, Personas — from social science — is still going to be consolidated in RE. Their conjunct usage may increase the ability to envision the system [5], to identify requirement problems and exceptional cases [6] and to help in discovering system functionalities. In particular, Personas are powerful instruments for creating descriptive models of system-to-be users based on behavioral data, gathered from many actual users encountered in ethnographic interviews [4]. Personas' descriptions contain the empathy with users and their personal motivations within a scenario; the cognitive and emotional dimensions are important factors since they help the designer to take decisions in the design process.

In our integrated process, scenarios are stories about people (personas) performing activities; they describe a context in which personas act with the aim of summarizing, clarifying and reasoning about the collected information. The aim is the validation with stakeholders and the technical team. They are presented as narrative or visual stories easy to understand even for non technical people [6].

Among the possible limits of the approach, there is the need of keeping the huge amount of data, usually collected by contextual inquiry and generated by the scenarios-based design, always synchronized with the Tropos diagrams; this also represents a challenge from a theoretical and practical point of view. In the future work, we intend to refine the approach and to investigate the requirements of a tool that support a multi-disciplinary team in this respect.

3 Aligning Goal and Process Models for Real Services

A real service is a combination of actual services and software services that provide electronic access to and monitoring of the actual services [7]. An example of real service is the application that we use when we plan to attend a concert at "Arena di Verona": its realization involves both software services (e.g., the online ticket booking), and actual services (e.g., the actual transportation service needed for reaching Verona).

In real services the user plays a key role: she expresses preferences (e.g., a cheap transportation mean); the service is adapted to her needs (e.g., if she has a meeting when her train should leave, a different actual transportation service is used); the real service is composed taking into account her perspective (e.g., the travel and the concert performance are part of the same real service, though in reality they are independent). Moreover, real services are context-aware, which may result in instant changes and timely responses of the user. Hence the user's preferences, needs and decisions guide the composition of real services, and they, in turn, impact and modify the user's assets (e.g., user's money, user's agenda). For example, buying the ticket for the concert, decreases the user's money.

Due to the "user intensive" nature of these systems, our high level purpose is to stress the centrality of the user along their life cycle. Hence, in this phase, our goal is to move from the user centred requirements (elicited as described in the previous Section) to the system design and validation, while preserving the central role of the user (and her assets). In detail, our work aims at defining a modeling framework that integrates the goal-oriented paradigm (specifically Tropos modeling methods [3]) and process modeling (in particular BPMN [8]), enabling the designers to capture the intentional elements of the user (e.g., goals, preferences, roles, assets), as well as the operational aspects of the real service (e.g., its control flow description, the effect of business activities on user assets). The modeling framework, moreover, includes an ontology for representing user assets and asset modifiers (i.e. activities characterizing the real service that can modify the value of a user-asset), thus capturing their semantics and making them available for supporting the system realization/execution (e.g., the service composition according to the user needs/preferences).

More precisely, the framework rests on the following iterative steps for designing a real service [2]:

- Ontology construction. An ontology modeling *user-assets* and *asset modifiers*is defined, thus capturing the semantics of the concepts, their relationships and constraints.
- Goal model construction. It starts with the analysis of the domain involved in the target system and the system requirements (from the previous requirement phase) and results in the definition of a goal model of the target system.
- Business process model construction. The process model of the target system is defined, deriving part of the information from the goal model. The process is described in the BPMN language and it details the process realization in terms of relevant activities and their execution control flow.
- Dynamic semantics definition of process activities. The dynamic semantics of the asset modifiers is defined, thus allowing to capture the impact of such activities on the user-assets.
- Process model annotation. The generated process model is enriched with semantic annotations [9] taken from the ontology modeling the asset modifiers.

In [2] we illustrated the proposed design methodology along an exemplar case-study. In our future work we intend to extend the modeling framework in the following directions: (i) adding scenarios to goals in the goal model in order to have a more detailed and precise description of goals, thus allowing to better reason about the domain and its concrete operationalization; (ii) further investigating how user preferences (expressed as soft goals in the goal model) affect the user assets, thus allowing to reason about "good" alternatives, to be possibly suggested to the user, during the system process execution; (iii) combining the business process semantics with the dynamic semantics of user assets, thus supporting the reasoning on feasible executions and possible user recommendations. Moreover, the design phases will be complemented with early validation and implementation phases. Early validation will be realized using a simulation system in which runs of real services, events, and operating contexts are simulated for testing purposes.

References

1. Leonardi, C., Sabatucci, L., Susi, A., Zancanaro, M.: Ahab's leg: Exploring the issues of communicating semi-formal requirement to final users. In: Proceedins of CAiSE 2010. (2010)
2. Marchetto, A., Nguyen, C.D., Francescomarino, C.D., Qureshi, N.A., Perini, A., Tonella, P.: A Design Methodology for Real Services. In: 2nd International Workshop on Principles of Engineering Service-Oriented Systems, PESOS, May 2-8, 2010, Cape Town, South Africa. (2010)
3. Penserini, L., Perini, A., Susi, A., Mylopoulos, J.: High variability design for software agents: Extending tropos. ACM Transactions on Autonomous and Adaptive Systems (TAAS) **2**(4) (2007)
4. Cooper, A., Reimann, R., Cronin, D.: About face 3: the essentials of interaction design. Wiley India Pvt. Ltd. (2007)
5. Rolland, C., Salinesi, C.: Supporting Requirements Elicitation through Goal/Scenario Coupling. In: Conceptual Modeling: Foundations and Applications, Springer (2009) 416
6. Sutcliffe, A., Maiden, N., Minocha, S., Manuel, D.: Supporting scenario-based requirements engineering. IEEE Transactions on Software Engineering **24**(12) (1998) 1072–1088
7. Pistore, M., Traverso, P., Paolucci, M., Wagner, M.: From Software Services to a Future Internet of Services. In: Towards the Future Internet - A European Research Perspective. IOS Press (2009) 183–192
8. Business Process Management Initiative (BPMI): Business process modeling notation: Specification. `http://www.omg.org/spec/BPMN/1.2/` (2009)
9. Di Francescomarino, C., Ghidini, C., Rospocher, M., Serafini, L., Tonella, P.: Reasoning on semantically annotated processes. In: International Conference on Service-Oriented Computing. (2008)