An Approach for Modeling and Analyzing Emotional Requirements

Supervisor: Mohamad Gharib (mohamad dot gharib at ut dot ee)

The success of any software system is determined based on whether it meets its users' and stakeholders' needs and expectations, where these needs and expectations serve as a key source for specifying the requirements for the system [1]. In the Requirements Engineering (RE) community, requirements are, traditionally, classified into functional and non-functional (quality) requirements, and a successful system should satisfy both types of these requirements [2]. While engineering methodologies have matured to handle functional and various types of quality requirements (e.g., usability, reliability) [3], emotional requirements have received relatively less attention from the RE community. Emotional requirements capture how users should feel when using the system [4]. Consequently, inadequate consideration of such requirements can result in end-users' reluctance to use the system [5]. Therefore, capturing and addressing emotional requirements of end users during system design is essential for a successful system as it improves its acceptance and usability. However, this is not an easy task as existing approaches for dealing with emotional requirements (e.g., motivational goal modeling (MMs) [3, 4]) are relatively scarce, they capture emotional requirements at a high abstraction level, they lack a systematic process for the refinement of emotional requirements, and, most importantly, they do not provide any formal semantics that allows any kind of automated analysis.

The aim of this research is tackling this problem by proposing a model-based approach for modeling and analyzing emotional requirements. The contributions of this research include: (1) A UML profile [6] for modeling emotional requirements along with other requirements (functional and quality) of the system to be. (2) An automated analysis support that allows for verifying emotional requirements models. More specifically, several properties of the design, represented as OCL (Object Constraint Language [7]), will be formulated to verify the models. (3) A tool that allows emotional requirements models to be generated and verified depending on OCL constraints. (4) An engineering methodology to assist designers while dealing with emotional requirements. The approach will follow a Design Science Research (DSR) approach [8], which identifies the problem that needs to be solved, motivates the development of the solution as a design artifact, and evaluates the application of the solution.

References

[1] Sommerville, Ian. "Software engineering (ed.)." America: Pearson Education Inc (2011).

[2] Chung, Lawrence, et al. Non-functional requirements in software engineering. Vol. 5. Springer Science & Business Media, 2012.

[3] Miller, Tim, et al. "Emotion-led modelling for people-oriented requirements engineering: the case study of emergency systems." Journal of Systems and Software 105 (2015): 54-71.

[4] Iqbal, T., Anwar, H., Filzah, S., Gharib, M., Mooses, K., & Taveter, K. (2023, May). Emotions in Requirements Engineering: A Systematic Mapping Study. In 2023 IEEE/ACM 16th International Conference on Cooperative and Human Aspects of Software Engineering, pp. 111-120. IEEE.

[5] Mendoza, Antonette, et al. "The role of users' emotions and associated quality goals on appropriation of systems: two case studies." (2013).

[6] Fuentes-Fernández, Lidia, and Antonio Vallecillo-Moreno. "An introduction to UML profiles." UML and Model Engineering 2.6-13 (2004): 72.

[7] Cabot, Jordi, and Martin Gogolla. "Object constraint language (OCL): a definitive guide." International school on formal methods for the design of computer, communication and software systems. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012. 58-90.

[8] Hevner, Alan R., et al. "Design science in information systems research." Management Information Systems Quarterly 28.1 (2008): 6.

A Modeling Language for Designing Web-based Applications

Supervisor: Mohamad Gharib (mohamad dot gharib at ut dot ee)

Web applications become more complex consisting of various software components, languages, and interfaces that not only provide stand-alone functions to the end user, but are also integrated with corporate databases and business applications [1]. Although web applications, like traditional software, require development processes/ methodologies, existing traditional software methodologies are not efficient for the development of web applications because they leave out relevant features that this kind of system has [2]. Specifically, web applications have distinguished characteristics that cannot be captured properly by existing traditional methods including: volatile requirements, multiple end-users and stakeholders, short deadlines, and limited resources, etc. [3]. As a result, various development methodologies have emerged, such as Object-Oriented hypermedia (OO –H method) [4], Scenario-based object-oriented hypermedia design methodology (SOHDM) [5], Object-oriented hypermedia design model (OOHDM) [6], UML-based web engineering (UWE) [7], and Web modeling language (WebML) [8] to mention a few. However, these methodologies did not see the level of adoption that their counterparts in traditional software applications have seen. A main drawback in most of these methodologies is their proposed modeling languages that are either too complex to be used, or very abstract to be practically useful.

To this end, a novel modeling language for web application design is required, and the main aim of this thesis is to propose it. This will be achieved by (1) critically reviewing available modeling language for designing web applications; (2) identifying their strength and weaknesses, (3) deriving a set of key requirements that the modeling language to be developed needs to meet, and (4) developing the modeling language according to the previously mentioned requirements. The developed modeling language will be evaluated by peer experts, and validated by applying it to a real/realistic case study/scenario.

References

[1] Huang, Yen-Chieh, and Chih-Ping Chu. "Developing web applications based on model driven architecture." International Journal of Software Engineering and Knowledge Engineering 24.02 (2014): 163-182.

[2] Molina-Ríos, Jimmy, and Nieves Pedreira-Souto. "Comparison of development methodologies in web applications." Information and Software Technology 119 (2020): 106238.

[3] Al-Salem, Lana S., and Ala Abu Samaha. "Eliciting Web application requirements–an industrial case study." Journal of Systems and Software 80.3 (2007): 294-313.

[4] Muller, Pierre-Alain, et al. "Platform independent Web application modeling and development with Netsilon." Software & Systems Modeling 4 (2005): 424-442.

[5] Escalona, M. José, and Nora Koch. "Requirements engineering for web applications–a comparative study." Journal of web Engineering (2003): 193-212.Escalona, M. José, and Nora Koch. "Requirements engineering for web applications–a comparative study." Journal of web Engineering (2003): 193-212.

[6] Schwabe, Daniel, and Gustavo Rossi. "The object-oriented hypermedia design model." Communications of the ACM 38.8 (1995): 45-46.

[7] Koch, Nora, and Andreas Kraus. "The expressive power of uml-based web engineering." Second International Workshop on Web-oriented Software Technology (IWWOST02). Vol. 16. CYTED, 2002.

[8] Ceri, Stefano, Piero Fraternali, and Aldo Bongio. "Web Modeling Language (WebML): a modeling language for designing Web sites." Computer Networks 33.1-6 (2000): 137-157.

Towards a Usable Privacy Notice

Supervisor: Mohamad Gharib (mohamad dot gharib at ut dot ee)

``I have read and agree to the Privacy Policy". This can be described as one of the biggest lies in the current times, and that is all what a service provider needs to acquire what can be called ``informed consent", which allows it to do as it pleases with your Personal Information (PI). Although many developed countries have enacted privacy laws and regulations to govern the collection and use of PI as a response to the increased misuse of PI, these laws and regulations rely heavily on the concept of informational self-determination through the ``notice" and ``consent/choice" model. This model is inefficient for acquiring informed consent simply because current mechanisms for presenting the notice and obtaining the consent are deeply flawed as indicated by many researchers because they are neither useful nor usable. More specifically, most notices are long and complex [1]; hard to be understood by ordinary people [2], and do not help much in assessing the consequences and risks related to consenting [3]. Consequently, most data subjects (DSs), usually, blindly accept such notices [4]. However, if a data subject did not read, understand and comprehend the consequences and risks of consenting, such consent cannot be informed [5], accordingly, it is not valid.

The challenge of providing ``usable" privacy notice has been recognized as an open challenge [6], and suggestions to improve it are scattered over the literature. This thesis aims at solving an answering an important Research Question (RQ) ``*How can we identify potential risks and/or consequences of consenting and make DSs aware of them*" resulting in a more ``usable" privacy notice. This thesis will use the ontology proposed in [7] as a baseline, aiming at extending and refining its concepts to properly answering the RQ. The developed ontology will be evaluated with the help of some software tools, lexical semantics experts, and privacy and security researchers.

References
[1] Alfssandro Acquisti and Jens Grossklags. Privacy and rationality in individual decision making. IEEE Security and Privacy, 3(1):26–33, 2005.
[2] Frederik Zuiderveen Borgesius. Informed consent: We can do better to defend privacy. IEEE Security and Privacy, 13(2):103–107, 2015.
[3] Bart Custers, Francien Dechesne, Wolter Pieters, Bart Schermer, and Simone van der Hof. Consent and privacy. Technical report, 2018.
[4] Bart W. Schermer, Bart Custers, and Simone van der Hof. The crisis of consent: How stronger legal protection may lead to weaker consent in data protection. Ethics and Information Technology, 16(2):171–182, 2014.
[5] Ra´ul Pardo and Daniel Le M´etayer. Analysis of privacy policies to enhance informed consent. In IFIP Annual Conference on Data and Applications Security and Privacy, volume 11559 LNCS, pages 177–198. Springer Verlag, 2019.
[6] Schaub, Florian, et al. "A design space for effective privacy notices." Eleventh symposium on usable privacy and security. 2015.
[7] Gharib, Mohamad. "Toward an architecture to improve privacy and informational self-determination through informed consent." Information & Computer Security 30.4 (2022): 549-561.

Towards a Novel Taxonomy for Requirements Interdependencies

Supervisor: Mohamad Gharib (mohamad dot gharib at ut dot ee)

It is well acknowledged in Requirements Engineering (RE) community that most requirements are related to one another [1]. According to Carlshamre et al., [2], only a fifth of the requirements are not related to or influenced by any other requirements. Requirements interdependencies capture how requirements relate to and affect each other [3], and it is one of the main problems to be handled for delivering quality requirements, and in turn, quality system/software. This said, requirements interdependencies cannot be overlooked or ignored and must be properly handled. In particular, requirements dependencies influence several design, development, and implementation decisions [4], and inappropriate handling of such interdependency can lead to software/system development failures [5].

Although various interdependencies among requirements have been considered in the literature (e.g., requires, refines, similar, or conflicts [6]), they are not able to cope with the advancement on the requirements side. More specifically, systems are becoming more complex, leading to more complex interdependencies among their requirements, which available interdependencies might not be able to capture them. Moreover, different types of requirements (e.g., emotional, awareness) have been introduced in the last two decades, and their dependencies cannot be appropriately captured with available dependencies types. Finally, research on dependencies is mainly focused on general dependencies overlooking the special types of some requirements.

The objective of this study is to synthesize existing knowledge in available literature concerning requirements interdependencies to answer the following three research questions (**RQs**): **RQ1:** What are the available types of requirements interdependencies? **RQ2**: How available types of requirements interdependencies are used among the different types of requirements? **RQ3**: Which key aspects of requirements interdependencies (e.g., traceability, prioritization) cannot be captured by available types of requirements interdependencies? Based on the answers to the aforementioned three questions, a new taxonomy of requirements interdependencies will be developed.

References

[1] Dahlstedt, Åsa G., and Anne Persson. "Requirements interdependencies: state of the art and future challenges." Engineering and managing software requirements (2005): 95-116.
[2] Carlshamre, Pär, et al. "An industrial survey of requirements interdependencies in software product release planning." International Symposium on Requirements Engineering. IEEE, 2001.
[3] Tabassum, Mirza Rehenuma, et al. "Determining interdependency among non-functional requirements to reduce conflict."International Conference on Informatics, Electronics & Vision (ICIEV). IEEE, 2014.
[4] Robinson, William N., Suzanne D. Pawlowski, and Vecheslav Volkov. "Requirements interaction management." ACM Computing Surveys (CSUR) 35.2 (2003): 132-190.
[5] Noviyanto, Fiftin, Rozilawati Razali, and Mohd Zakree Ahmad Nazree. "Understanding requirements dependency in requirements prioritization: a systematic literature review." International Journal of Advances in Intelligent Informatics 9.2 (2023): 249-272.
[6] Gharib, Mohamad, et al. "Privacy requirements: findings and lessons learned in developing a privacy platform." International Requirements Engineering Conference (RE). IEEE, 2016.

An Approach for Deriving "Integration Requirements" for Enterprise Systems

Supervisor: Mohamad Gharib (mohamad dot gharib at ut dot ee)

Over the past decades, Enterprise Systems (ESs) (also called Enterprise Information Systems (EISs)) have emerged as a promising tool for integrating and extending functionalities and services at both intra-organizational and inter-organizational levels [1]. In particular, ESs are, usually, the result of integrating two or more existing organizational systems into broader intra/inter-organizational ES [2]. The software engineering community has always been faced with different challenges in designing and developing larger complex systems such as ESs. One of these challenges is specifying the requirements for ESs, especially, the requirements related to the integration aspects of the ES (integration requirements), e.g., they may not be easily identified because they may manifest themselves in a vague way within the overall set of the ES requirements, they might be overlooked, ignored or considered irrelevant, they might exist in different forms (e.g., functional or non-functional requirements), etc. However, integration requirements are a vital success factor for any ES, they may have an impact on the overall performance of the ES since if they are incorrect, incomplete, and/or inconsistent; the ES-to-be is likely to suffer from serious issues [3]. Although much effort has been devoted to engineering requirements for ESs (e.g., [1, 4, 5]) with a particular focus on "architecturally significant requirements" (e.g., [2, 6]), no specialized approach has been developed for dealing with "integration requirements". This results in dealing with such important requirements in an ad-hoc manner.

The objective of this thesis is to fill the aforementioned gap by developing an approach for identifying (e.g., eliciting, specifying, and/or deriving), analyzing, and validating integration requirements for ESs. This objective will be achieved by (1) critically reviewing available literature on ES's requirements; (2) reviewing available requirements engineering practices, identifying their strength and weaknesses for dealing with integration requirements; and based on best practices (3) proposing approach for identifying, analyzing and validating integration requirements. The approach will be validated by applying it to a real/realistic case study/scenario.

References

[1] L. D. Xu, "Enterprise systems: State-of-the-art and future trends," IEEE Trans. Ind. Informat., vol. 7, no. 4, pp. 630–640, Nov. 2011.

[2] Niu, N., Da Xu, L., Cheng, J.R.C. and Niu, Z., 2013. Analysis of architecturally significant requirements for enterprise systems. IEEE Systems Journal, 8(3), pp.850-857.

[3] B. Nuseibeh and S. Easterbrook, "Requirements engineering: A roadmap," in Proc. Future Softw. Eng., Limerick, Ireland, 2000.

[4] H. M. Kim, S. Lu, J. S. Kim, and B.-D. Kim, "Parallel, multistage model for enterprise system planning and design," IEEE Syst. J., vol. 4, no. 1, pp. 6–14, Mar. 2010.

[5] M. Fonoage, I. Cardei, and R. Shankar, "Mechanisms for requirements driven component selection and design automation," IEEE Syst. J., vol. 4, no. 3, pp. 396–403, Sep. 2010.

[6] Chen, L., Babar, M.A. and Nuseibeh, B., 2012. Characterizing architecturally significant requirements. IEEE Software, 30(2), pp.38-45.