

# A model-driven method for gesture-based interface requirements specification

Otto Parra González

PROS Research Centre, Universitat Politècnica de València, Spain  
otpargon@upv.es  
Computer Science Department, Universidad de Cuenca, Ecuador  
otto.parra@ucuenca.edu.ec

**Abstract.** Currently there are several software development suites that include tools for user interface design and implementation (mainly by programming source code). Some of the tools are multi-platform and multi-style; that is, allowing the specification of devices, e.g. computer, notebook, smartphone, and user interaction styles, e.g. based on gestures, voice, mouse and keyboard. Among the styles, gesture-based interaction is neglected, despite the proliferation of gesture-recognizing devices. Given the variety of styles of human-computer interaction currently available, it is necessary to include information on these styles in software requirements specification to obtain a complete specification prior to code generation. In this paper, we propose the design of a model-driven method and tool that allows specifying gesture-based interactions and then generates gesture-based interface requirements specification. We intend to our proposal be interoperable with existing methods and tools. The research method follows design science and we plan to validate our proposals by means of technical action-research.

**Keywords:** requirements engineering, gesture-based interface requirements specification, model-driven method, gesture-based interaction, user interface.

## 1 Motivation

The software requirements specification is one of the most important tasks in software development [1]. This is one of the first activities in the software development life cycle, which allows you to define the characteristics of such software based on the needs of users [2]. Later, the design of user interfaces implies a trade-off between the tasks being supported, the most appropriate styles of presentation and interaction, the previous experience of users and the available devices [3].

Nowadays, both software requirements specification and user interface design need to be performed in a coordinated manner so as to ensure the consistency, completeness, usefulness and usability of the resulting software [2] [4] [5] [6] [7] [8] [9]. However, previous proposals for user interface modelling do not take into account the user interaction style and the available devices.

Currently, the development of devices has produced the existence of two main interaction styles in the field of user interfaces. The first known as WIMP (Windows, Icons, Menus, Pointer) is related to conventional interaction based on the use of keyboard and mouse that uses the GUI (Graphical User Interface) in desktop computers and notebooks. The second known as post-WIMP is associated with non-conventional interaction employing other interaction styles that are currently available and results in what has been called NUI (Natural User Interface) [10].

In the field of user interfaces, there are several aspects of human-computer interaction that need to be modelled (related with interaction using keyboard and mouse, gestures or voice) [3] together with software components. These aspects, together with the user interaction styles in a device (i.e. gestural, vocal and conventional) paves the way for developing multimodal systems. On the whole, these elements give the user an opportunity for choosing the most natural interaction pattern depending on the task, context and, preferences and skills [11]. Therefore, they must be included in the software requirements specifications [2].

This thesis proposes extending existing model-driven practices to allow for gesture-based interface requirements specification.

Section 2 further elaborates the problem statement that is confronted in this thesis, including the expected contributions. Section 3 describes the research questions. Section 4 outlines the research methodology. Section 5 overviews the proposed method and tools. Section 6 shows the progress of the thesis.

## 2 Problem Statement

Currently gesture-based interface development lacks proposals for proper interaction specification and interface design. If a model-driven development is intended, such proposals are even more essential because the models must include complete requirements to create the software product using model-transformation and code-generation tools.

This suggests that it would be more effective to include these interaction specifications so that the software meets the requirements of users and also provides an interaction according to the type of task to be performed with the software, for instance, a software may include conventional interaction, which is still valid, and the gesture-based interaction to perform computer tasks using their hands.

Our work aims to *design a model-driven method and tool for specify gesture-based interface requirements*. We aim to make the contributions cited in Section 5.

## 3 Research Questions

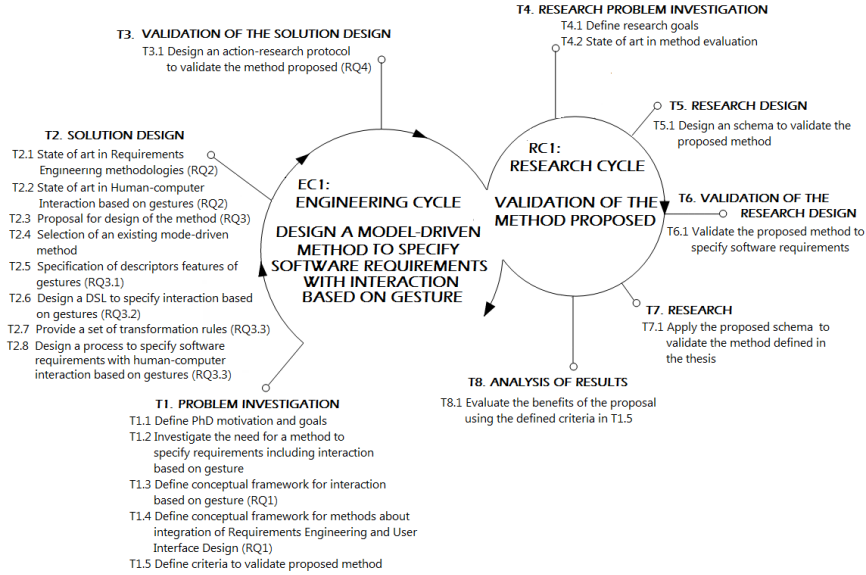
In this thesis, we aim at gathering new knowledge and producing useful artefacts; thus, we opt for a design science approach. We classify the research questions as either knowledge problems (KP) or design problems (DP), based on the definitions by Wieringa [12]:

- **RQ1** (KP): What elements should be considered for the definition of a method for the software requirements specification considering human--computer interaction based on gestures? The answer to this question will establish a conceptual framework to help in the process of defining the method mentioned.
- **RQ2** (KP): What methods exist for software requirements specification and user interfaces with human--computer interaction based on gestures? The answer to this question should establish the state of the art regarding model-driven methods existing at present for software requirements specification which consider human--computer interaction.
- **RQ3** (DP): Design a method for the specification of software requirements that considers the human--computer interaction based on gestures. The answer to this question is related to the main objective of this thesis.
  - **RQ3.1** (KP): Determine the characteristics of a gesture that are representative to be used as descriptors of human--computer interaction.
  - **RQ3.2** (DP): Define a tool to represent a gesture in the specification of the interaction in user interface.
  - **RQ3.3** (DP): Establish techniques and tools to facilitate the use of the method proposed!
- **RQ4** (KP): What advantages and disadvantages has the model-driven method for software requirements specification that considers the human--computer interaction based on gestures? The answer to this question should establish a validation scheme to measure feasibility, sensitivity, advantages and disadvantages of the method proposed through the use of technical action-research [13]. We plan to apply the results of RQ3 to project “Capability as a Service in Digital Enterprises - CaaS”, Project of European Commission FP7 (ref. 611351).

## 4 Research Methodology

The type of research methodology used corresponds to design science framework since its purpose is the design a new artefact: a model-driven method to define gesture-based interface requirements specification. The research methodology is explained by means of regulative cycles [12] that were conceived in order to answer the research questions. This methodology proposes: 1) to perform an initial problem investigation that characterizes the problem to solve; 2) to provide a solution design suitable to solve those problems; and 3) to validate if the proposed solution satisfies the problematic phenomena previously analyzed.

The main cycle of the research methodology is an engineering cycle (EC1: Design a model-driven method to specify software requirements with interfaces that include human--computer interaction based on gestures) since this proposal focuses on the development of a new artifact (method). A research cycle has been defined (RC1: Validation of the proposed method), where the process that will be developed to validate the proposed method is described. This process corresponds to a case study provided in the project CaaS. **Fig. 1** presents the research methodology described, where the regulatory cycle can be observed.



**Fig. 1.** Overview of the research methodology

The research methods to be used in the development of this work are:

- A literature review to know the current state of the conceptual framework using bibliographic sources as IEEE, ACM, and Scopus will be done. In this case, a conceptual framework will be defined. (RQ1)
- A literature review to develop state of art on the issues related with this work. (RQ2)
- A set of metamodels is used to design the proposed method because it raise the level of abstraction, it is platform-independent and facilitates the process of models transformation. The transformation rules will be defined in a language based on Meta Object Facility (MOF) Model to Text Language (MTL) by Object Management Group (OMG) and restrictions will be defined in OCL (Object Constraint Language). Guidelines about the definition of metamodels will be included in this work. (RQ3)
- The validation of the design solution using TAR (Technical Action Research) will be made by the process suggested in [13]. (RQ4)

## 5 Solutions and Contributions

This work proposes a model-driven method that provides the gesture-based interface requirements specification. A model-driven schema is used because the problem is modeled at a high level of abstraction, it automates the development process and by MDE goals: portability, interoperability and reusability. The software requirements specification obtained will be expressed in the Requirements Interchange Format

(ReqIF), defined by OMG. ReqIF is a standardized open, generic, non-proprietary and tool-independent exchange format based on XML [14]. ReqIF is platform-independent and it can be used to define software requirements specifications to any device with conventional and gesture-based interaction.

This work proposes an extension of an existing method [15] which includes the users' characteristics (user model), the current domain of application (domain model), and the tasks they commonly perform (task model). The task model is obtained using a transformation from domain model to task model according to [16]. The three models are combined and added other functions to create concrete interaction objects (CIO) which are transformed to final interaction objects (FIO), the last step is the code generation to obtain the user interface and application code (Fig. 2). The task model is defined according to ConcurTaskTrees notation [17].

The proposal of this work consists on the definition of: (i) device model based on the Model-View-Controller (MVC) design pattern, the device contains the interfaces; (ii) interaction style model; and (iii) gesture representation model. The first step of the proposed method (Fig. 3) is a M2M transformation between device model and CIO model (obtained in the existent method) which produces a model that contains the definition of the interface of a device with concrete interaction objects added. The next step is a M2M transformation between gesture model and interaction style model to obtain the gesture-based interaction style model. The last step is a model-to-text (M2T) transformation between interface model and gesture-based interaction style model which generate the gesture-based interface requirements specification. This proposal is shown in (Fig. 3).

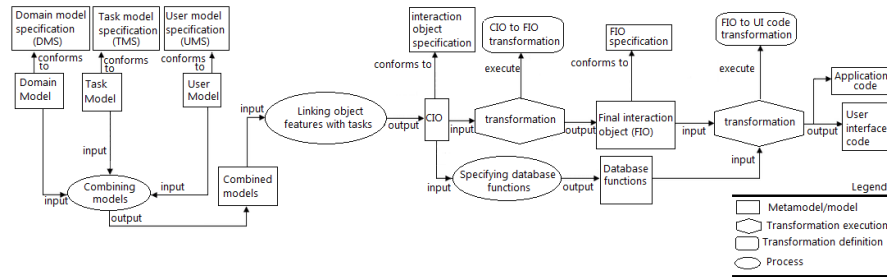


Fig. 2. Schematic diagram of the existent method

Therefore, the main contribution to be expected from this proposed work is a model-driven method to specify the gesture-based interface requirements. Other contribution is a tool to specify gestures-based interaction.

## 6 Progress of the Thesis

Issues related to integration processes of software requirements specification and design of user interfaces, also on gestural representation and methods for the specification of user interfaces (RQ1) were studied in 2013. A review of the related literature

was conducted with these issues and finally the state of art of each of these topics (RQ2) was performed.

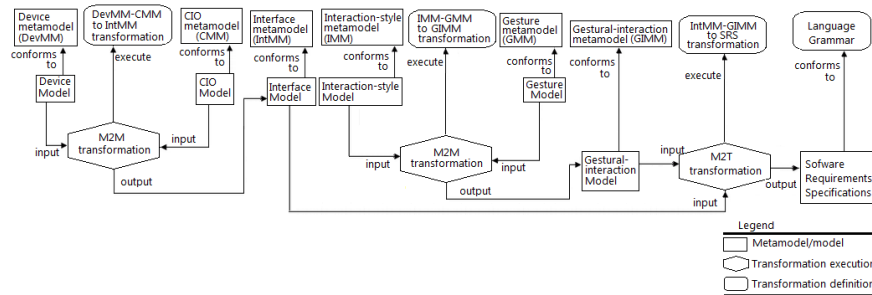


Fig. 3. Schematic diagram of the proposed method

It has also selected an existing method for designing user interfaces in order to extend it by including the specification of the interaction style, specifically two types of interaction: conventional and gestural. Have been defined some metamodels necessary for the proposed process.

In 2014 we plan to specify descriptors of gesture features (RQ3.1) for the design of the planned tool (RQ3.2) for the definition of gesture-based interaction, we plan to define the transformations rules (RQ3.3) to include in the proposed method and we also plan to finalize the specification of the proposed method.

En 2015 we plan to establish the guidelines of use and we plan to validate the method proposed (RQ4). In 2016 we plan to finalize writing the PhD thesis document.

## Acknowledgements

My acknowledgements to my supervisors Sergio España and Óscar Pastor for their invaluable support and advices. This work has been supported by Secretaría Nacional de Educación Superior, Ciencia y Tecnología – SENESCYT, and Universidad de Cuenca, which are institutions of the Republic of Ecuador. Research partially supported by the Spanish Ministry of Science and Innovation project PROS-Req (TIN2010-19130-C02-02), the Generalitat Valenciana project ORCA (PROMETEO/2009/015), European Commission FP7 Project CaaS (611351).

## References

- [1] R. Butkiene and R. Butleris, "The approach for user requirements specification," in *5th East-European conference ABDIS-2001*, Lituania, 2001.
- [2] I. Antovic, S. Vlajic, M. Milic, D. Savic and V. Stanojevic, "Model and software tool for automatic generation of user interface based on use case and data model," *IET Software*, vol. 6, no. 6, pp. 559-573, 2012.
- [3] I. C. Society, Software Engineering Body of Knowledge (SEBOK), Abran, A.; Moore, J.; Bourque, P., 2013.

- [4] S. España, I. Pederiva, J. I. Panach, S. Abrahao and O. Pastor, "Linking requirements specification with interaction design and implementation," *IFIP - Human Work Interaction Design: Designing for Human Work*, vol. 221, no. Springer, pp. 123-133, 2006.
- [5] L. Constantine, R. Biddle and J. Noble, "Usage-Centered Design and Software Engineering: Models for Integration," in *ICSE Workshop on SE-HCI*, 2003.
- [6] P. Anitha and B. Prabhu, "Integrating Requirements Engineering and User Experience Design in Product Life Cycle Management," in *UsARE*, Zurich, Switzerland, 2012.
- [7] H. Fischer, "Integrating Usability Engineering in the Software Development Lifecycle Based on International Standards," in *Proceedings of the 4th ACM SIGCHI symposium on Engineering interactive computing systems - EICS'12*, Copenhagen, Denmark, 2012.
- [8] J. I. Panach, N. Juristo and O. Pastor, "Including Functional Usability Features in a Model-Driven Development Method," *Computer Science and Information Systems*, vol. 10, no. 3, pp. 999-1024, 2013.
- [9] K. Nebe and V. Paelke, "Key Requirements for Integrating Usability Engineering and Software Engineering," in *Human-Computer Interaction. Design and Development Approaches, 14th Int. Conference*, USA, 2011.
- [10] D. Wigdor and D. Wixon, *Brave NUI World: Designing Natural User Interfaces for Touch and Gesture*, UK: Morgan Kaufmann Publishers, 2011.
- [11] K. Kvale and W. N. D., "Multimodal Interfaces to Mobile Terminals – A Design-For-All Approach," in *User Interfaces*, Vukovar, Croatia, Intech Europe, 2010, pp. 207-228.
- [12] R. Wieringa, "Design Science as Nested Problem Solving," in *DESRIST'09*, Malvern, PA, USA, 2009.
- [13] R. Wieringa and A. Morali, "Technical Action Research as a Validation Method in Information Systems Design Science," in *7th International Conference - DESRIST 2012*, Las Vegas, USA, 2012.
- [14] OMG, *Requirements Interchange Format (ReqIF) v 1.1*, USA: OMG, 2013.
- [15] V. Tran, J. Vanderdonckt, M. Kolp and S. Faulkner, "Generating User Interface from Task, User and Domain Models," *2009 Second International Conference on Advances in Human-Oriented and Personalized Mechanisms, Technologies, and Services*, pp. 19-26, 2009.
- [16] C. Pribeanu, "An Approach to Task Modeling for User Interface Design," *Proc. of World Academy of Science, Engineering and Tech.*, vol. 5, pp. 5-8, 2005.
- [17] F. Paternó, "The ConcurTaskTrees Notation," in *Model-Based Design and Evaluation of Interactive Applications*, London, UK, Springer-Verlag London Limited, 2000, pp. 39-66.