A Generic Model of Deliverable Artifacts through the Development of Graphical User Interfaces

Laura C. Rodríguez-Martínez[1], Ricardo Mendoza-González[1], Héctor Durán[2] and Mario A. Rodríguez-Díaz[1]

[1]Instituto Tecnológico de Aguascalientes, Aguascalientes, Ags., México
[2]Universidad de Guadalajara, Zapopan, Jal., México
lrodriguez@mail.ita.mx
Paper received on 14/07/12, Accepted on 02/08/12.

Abstract. Every product requires a process model to guide its construction; to define such a model it is required the specification of its components (i.e. its phases, activities, deliverable artifacts, agents (people and tools), and roles). The construction of Graphical User Interfaces (GUI), as a product, is not an exception. In the Human-Computer Interaction (HCI) research area the development of GUIs is driven by a set of principles, guides and tools. However, the development process is presented in a general way (i.e. current approaches donnot present a detailed development process), and in most of cases the development process is not explicitly presented. In recent proposals for constructing deliverable atifacts, it is suggested the use of tools like UML, schetching, prototyping among others. Nevertheless, such proposals donnot specify the deliverable artifacts to be constructed neither its order of construction. This paper proposes a generalized model of deliverable artifacts of a GUI development Process (GDP). Our main aim is reaching an explicit and detailed definition of a GDP that indicates the deliverable products and their order of construction. The contributions of this work are: (1) the generalizad model of deliverable artifacts (or products) of the GDP through the normalization of the deliverable products proposed in ten selected GDPs; (2) a general description of each deliverable artifact of such a generalized model. Finally, the validation of this proposal is based on the robustness of the revised theoretical base, the arguments for the generalization of the proposed model, and by comparing comparing our approach with the International standard ISO 13407.

Key words: Human-Computer Interaction, Graphical User-Interface, GUI Development Process, GUI Design Process.

1 Introduction.

One of the main objectives to model the GUI development processes (GDPs) is to reach deliverable artifacts with quality in two senses. First, the quality of the development process for GUIs, and second, the quality of the proper interfaces developed with the specific GDP. On the other side, GUI design is considered a critical task for the successful development of software systems. Thus, we believe that

Advances in Intelligent and Information Technologies.
Research in Computing Science 60, 2012, pp. 175 -187.

while the System Development Life Cycle (SDLC) literature scarcely addresses the task of GUI design, such type of studies are required and can enhance the required experience in modern highly interactive software systems to design GUIs. We consider that by including the "HCI experience" in the used SDLC, it can be reached a more "systematic application" of such an experience and it can be avoided the commonly intuitive GUI design realized by developers. Then, primarily it is required to know what is to be constructed and how, and as part of this knowledge it is necessary the generic and explicit definition of deliverable artifacts into the GDP to be modeled in this paper.

We review ten selected and relevant GUI design methodologies reported in the Software Engineering (SE) and Human-Computer Interaction (HCI) literature. Each one of these proposals, defines explicit or implicitly several artifacts as products of the process of design and implementation of GUIs in software systems. We do not intend to address the great open problem of HCI. However, we believe that an analysis of such a SwE and HCI literature from a methodological point of view, will contribute to advance our scientific knowledge on how to improve the usual SDLC on regarding the GUI design.

This paper continues as follow: in section II we explain the theoretical background about well-defined processes, and a previous work about normalized activities of GDP. Section III provides a cross-grained description of the deliverable artifacts of the ten selected GDPs (named from H1 to H10). Section IV describes the generalized model proposed. In section V, the paper concludes with recommendations for further work and limitations.

2 Theoretical background.

A well-defined process model defines its phases, activities, artifacts, roles, and tools, for building a software system [10]. Since 1970 several SDLCs have been proposed in the SwE area. According to [11] each new SDLC evolves from previous ones fostered by two main drivers: (i) the knowledge-gap driver and (ii) the technological-change driver. For instance, the emergence of the Unified Process (or its most known version Rational Unified Process) can be explained by the need of counting with an Object-oriented SDLC (e.g. the knowledge-gap driver) given that availability of Object-oriented technologies (e.g. the technological-change driver). A similar situation occurred for the SDLCs based on flow of data and functions (e.g. there were technologies based on functional paradigm). Following a similar logical argumentation, we claim that in the SwE area there is a gap referring to a plausible well-defined process to design and implements GUIs as part of a normal SDLC. Then, by capitilizing the experience of HCI on SDLC we claim that the study of GDPs is necessary and important too by the following consideration presented in [12]: "*We support the notion of that the Software System development cannot be separated of the development of its respective GUI. In 1987, Rosson identified a discussion point about the benefits to separate such task or keep it on the other systems' functionalities. Its study revealed that there is not one unified and systematic GUI design decision concerning to its realization through a separate or included GUI design process from the system design process. In such a study on a sample of*

19 software developers, 8 designers recognized such separation as feasible and necessary, including the design of GUI in early development tasks, but the remainder 11 designers indicated that GUI must not be considered as a separate element of the system. Furthermore, they expressed that they were not able to imagine the difficulty of such a separation. Rosson concluded that if the system functionality is a product of an integrated design process of the system with its GUI, then this integration will provide help for supporting one user-centered design approach with the benefit of reaching a correct early user interaction in less number of design iterations. This notion of a non-separation of the GUI design task of the SLDC has been also supported by other studies. For instance, Harston & Hix reports that such a separation introduces a greater level of design difficulty rather than its integrated execution. Thus, since the GUIs expose functionality of software-systems and they are strongly linked to the total system functionality, both processes of design should not be separated. Hence, the concern of a GUI separated or included task in a SLDC is a relevant issue that deserves further study."

We have revised ten selected GDPs, shown in Table 1, and the survey of GDPs presented in [12]. This survey presented a table with ordered and normalized activities for ten GDPs that are shown in Fig. 1. Such selcted GDPs are named H1..H10.

Table 1. Ten GUI Development Processes (GDP) down study.

H1	"Designing for Usability: Key Principles and What Designers Think" Gould & Lewis, 1985 [4]
H2	"Designing for Designers: Analysis of Design Practice en the Real World" Rosson, Maass, & Kellogg, 1987 [13]
H3	"Human-Computer Interface Development: Concepts and Systems for its Management" Hartson & Hix, 1989 [6]
H4	"The Usability Engineering Life Cycle" Nielsen, 1992 [9]
H5	"Iterative Methodology and Designer Training in Human-Computer Interface Design" Bailey, 1993 [2]
H6	"Usability Engineering Turns 10" Butler, 1996 [3]
H7	"Design Methodology and Design Practice" Lowgren & Stolterman, 1999) [7]
H8	"Guía de actuación en el desarrollo de interfaces de usuario según la metodología centrada en el usuario INTEGRAM" Losada, López, & Martínez, 2004 [8]
H9	"Diseño de Sistemas Interactivos Centrados en el Usuario" Granollers, Lorés, & Cañas, 2005 [5]
H10	"A Survey on HCI in the Software Development Life Cycle: from Practitioner's Perspective" Abd Majit, Md Noor, Wan Adnan & Mansor, 2009 [1]

We have revised this theoretical base as a means to identify the gaps on SwE related to a systematic and detailed process for developing GUIs.

3 Description of deliverable artifacts of the ten GDPs.

This section presents the deliverable artifacts for each of the GDPs assigned to the normalizad activities presented in Figure 1. The artifacts are proposed in each of the GDPs in an explicit or implicit way. Such artifacts are contained in Table 2 to Table 11.

Each table shows in the first column, the activities proposed by each GDP (one table corresponds with one GDP); and the second column shows the deliverable arti-

facts. Each deliverable artifact proposed by the corresponding GDP is located on the row corresponding to the activity that produces it. For example, the deliverable artifact "Preliminary Specification of the User Interface" (located in the second row in Table 2 for GDP H1), is produced by the activity "Preliminary Specification of the User Interface". It is important to note that differend levels of detail are presented in Tables H1..H10, since the corresponding GDPs are defined at different levels of detail.

Identification of artifacts and and relating them to particular activities of each GDP is not trivial since several of the artifacts are included in some proposals in a implicit way. Furthermore relating artifacts to activities is based on the description (implicit or explicit) of both, artifacts and activities in the GPD.

<table>
<tr><th>Macro-phase</th><th>Phase</th><th>H1 (1985) Gould</th><th>H2 (1987) Rosson</th><th>H3 (1989) Hartson</th><th>H4 (1992) Nielsen</th><th>H5 (1993) Bailey</th><th>H6 (1996) Butler</th><th>H7 (1999) Lowgren</th><th>H8 (2004) Losada</th><th>H9 (2005) Granollers</th><th>H10 (2009) Abd Majid</th></tr>
<tr><td rowspan="4">Definition</td><td rowspan="3">Requirements</td><td>Organize the Work.</td><td>Other Activities (i.e. marketing activities)</td><td>---</td><td>---</td><td>---</td><td>---</td><td>---</td><td>---</td><td>---</td><td>Project selection and planing</td></tr>
<tr><td>Preliminary Specification of the User Interface.</td><td>---</td><td>---</td><td rowspan="3">Predesign</td><td>---</td><td rowspan="3">Analysis</td><td rowspan="3">Design Contextualization</td><td rowspan="3">Analysis: task analysis / functional analysis, requirements specification</td><td>Analysis of Needs</td><td rowspan="3">Analysis</td></tr>
<tr><td>Collect Critical Information About Users.</td><td>---</td><td>Requirements Specification</td><td>---</td><td>Requirements Analysis</td></tr>
<tr><td>Develop Behavioral Goals.</td><td>---</td><td>Functional Analysis of Tasks</td><td>---</td><td>User Analysis and Task Analysis + Functional Analysis</td></tr>
<tr><td rowspan="3">Development</td><td>Design</td><td rowspan="3">Iterative Development Phase</td><td>Design</td><td>Formal Conceptual Design</td><td rowspan="2">Design</td><td>Design</td><td>Design</td><td>Detailed Design</td><td>Design: conceptual design/formal design</td><td>Design</td><td>Design</td></tr>
<tr><td>Construction</td><td rowspan="2">Implementation</td><td>Prototyping</td><td rowspan="2">Prototyping</td><td rowspan="2">Construction</td><td rowspan="2">Implementation</td><td rowspan="2">Prototyping</td><td rowspan="2">Prototype</td><td rowspan="2">Implementation</td></tr>
<tr><td>Evolution</td><td>Operation</td><td>Interface Construction</td><td>Evaluation</td><td>Evaluation</td><td>Postdesign</td><td>Evaluation</td><td>Evaluation</td><td>Evaluation</td><td>Evaluation</td></tr>
</table>

Figure 1. Normalized activities of the ten GDPs.

Table 2. Artifacts deliver for the activities of the GDP H1 [4].

Activity	**Deliverable Artifact**
Organize the work	---
Preliminary Specification of the User Interface	Preliminary Specification of the User Interface
Collect Critical Information About User	Documentation of user characteristics (cognitive, conductual, antrophometric and actitudinal characteristics)
Develop Behavioral Goals	Behavioral objective definition: - Behavioral User description. - Task to be perfomed, and the circunstances in which they should be performed. - Definition of interesting metrics, such as learning time, errors, number of help requisitions. Definition of metrics criteria.
Iterative Development Phase	Presentation, review and verify. Prototyping Empirical testing.

Table 3. Artifacts deliver for the activities of the GDP H2 [13].

Activities	**Deliverable Artifacts**
Other activities (i.e. marketing activities)	---
Design	---
Implementation	---
Evaluation	Evaluation

Table 4. Artifacts deliver for the activities of the GDP H3 [6].

Activities	**Deliverable Artifacts**
Requirements specification	---
Functional analysis of tasks	Task analysis: Task list, Task hierarchi, Task description
Format conceptual design	Structural description: Dialog objects (entradas, validaciones, mensajes) Interfaces representations: - Detalils of the form - Content - Sequence (using interfae representation schemes) Dialog transactional model that includes constituent objects of UI (static or dynamic dispaly objects, input objects and dialog computational objects). Presentation component: lexical and syntactic criteria that implies the relation between objects as transaction, interaction and actions. Input Dialog. Output Dialog. Dialog control component (dialog between human and computer)
Prototyping	Application interface model.
Interface construction	Interface Construction
Evaluation	Evaluation

Table 5. Artifacts deliver for the activities of the GDP H4 [9].

Activities	**Deliverable Artifacts**
Predesign	Know user: - individual user characteristics, - user' model of the task - análisis de funciones. Competitive analysis. Definition of Usability Objectives.
Design	Developers' model of the task (including methapors). Prototypes. Consistency that includes the definition of standards and the definition of the product identification. Definiton of the guidelines of usability for heuristic evalution: - general guidelines - guidelines of specific category - guidelines of specific product
Postdesign	Empírical testing (with prototype)

Table 6. Artifacts deliver for the activities of the GDP H5 [2].

Activities	Deliverable Artifacts
Design	---
Prototyping	Prototyping
Evaluation	Evaluation

Table 7. Artifacts deliver for the activities of the GDP H6 [3].

Activities	Deliverable Artifacts
Analysis	User's conceptual model of system functions: User's work Concepts, Data, Processes, Task Analysis, Hierarquical Decompostion, Contextual Analysis System functions and requirements of technology
Design	Computing specifications: 1. Functional descomposition (objects + operations + interfaces) : - Mapeo del "trabajo del mundo real" (real-world work) - Computational specifications assigned to objects: for each object is defined its layout, appareance, control and beha-vior. 2. Layout, appearance, control and behavior of the presentation.
Construction	Prototypes (since low fidelity even high fidelity): 1. User interface model 2. First prototypes with less interactivity. 3. Prototype evolvable using reusable components. Deliverable application
Evaluation	Evaluation: - Metrics of evaluation - Registry of objective data. - Registry of subjective data. - Assesing of suitability of design for supporting the planned task improvements, of to recommend how to improve design. - Diagnostics of problems.

Table 8. Artifacts deliver for the activities of the GDP H7 [7].

Activities	Artifacts
Design Contextualizacion	Function list, clasification of functions in principal, necessary, desirables, and innecessary)
Detailed Design	---
Implementation	---
Evaluation	---

Table 9. Artifacts deliver for the activities of the GDP H8 [8].

Activities	Artifacts
Analysis: task analysis / functional analysis, requirements specification	User Model of Task (HTA – Hierarquical Task Analysis)
Design: conceptual design / formal design	Dialog Model.
Prototyping	- Early prototype with presentation and navigation models which represented as primitive presentation. - Implementation of operative version.
Evaluation	Verification of design integrity, Logs, Problem identification.

Table 10. Artifacts deliver for the activities of the GDP H9 [5].

Activities	Deliverable Artifacts
Analysis of Needs	- Plan. - Functional goals and objectives.

Requirements Analysis	- Scenario análisis that represent task and procedures. - Platform Analysis. - Usability Objectives. - Interaction Style (boceto + storyboard). - Contextual Analysis.
User Analsysis and Task Analysis + Functional Analysis	- Stackeholders analysis. - Task Analysis. - Hierarquical task analysis (HTA) - Task-Object-Rol-Agent-Event . - Functional Analysis.
Design	- Metaphors Analysis. - GUI Design (paper prototype + storyboard). - Navigation. - Information Architecture, that includes the layout and graphical design. - GUI design (digital mokap + storyboard navegacional).
Prototype	Software prototype with minimal functionality called horizontal prototype, and/or disposable prototypes with different technological alternatives.
Evaluation	Heuristic evaluation, trail observation, field observation, interviews, questionaries, test, focus group, thinking aloud.

Table 11. Artifacts deliver for the activities of the GDP H10 [1].

Activities	**Deliverable Artifacts**
Project selection and planning	Project Plan
Analysis	- User Requirements: Context Analysis, User Analysis, Task Analysis, Evaluation Metrics - Formative Evaluation
Design	- Interface specification: Metaphor Design, Dialog Design, Media Design - Presentation Design
Implementation	Coding and Formative Evaluation

4 Generalized model of deliverable artifacts (products) of the GUI development process.

Products of the ten GDPs are normalized based on the activities of each GDP, – showed above on from Table 2 to Table 11– and on the location of these activities into the generic phases showed in the second column in Table 12. The products of the ten GPD were normalized taking as a base the activities and their location within the generic phases (shown in figure 1) as well as the location of the deliverables products of each GDP within the activities of the GPD.

Table 12. Generalized model of deliverable artifacts to develop a GUI.

Macro-phase	**Phase**	**Generalized Deliverables**	**Description**
Definition	**Requirements**	Plan	This deliverable contains the project plan to be executed.

		Definition of behavioral objectives. Requirements Specification	The definition of behavioral objectives is a document that describes the user's model: - Stakeholders' analysis and their cognitive, conductual, antropomhetric and actitudinal characteristics. - Task and procedure analysis of the user: task list, hierarquy of task, task description. - Context Analysis (Scenario Analysis): that shows wakness and streghtness of the user to work (mails, much thaan enough time invertid, etc.) - Fucntional Analysis - Platform Analysis Requirements specification is a document that defines: 1. The conceptual and contextual model of user' system-functions: - data, - processes - System task analysis - Hierarquical decomposition of functions 2. Functional objectives and goals: - Functions list. - Functions clasification as principals, necessaries, desirables or unnecesaries. 3. Usablity Objectives 4. Interactional Style (sketch + storyboard) 5. Evaluation Metrics 6. Formative Evaluation
		Preliminary Design (sketched)	This deliberable includes: 1. The GUI design: - Task Analysis = hierarquical analysis (HTA) of Task-Object-Rol-Agent-Event - Functional Analysis - Methapor's Analysis - Paper prototyping and storyboard 2. Definition of behavioral objectives of the interface: - User' description - Task to be performed, and the circunstances in which they should be performed - Definition of metrics of interest, such as learning time, errors, number of help requestings. Definition of criteria to be reached to each metric. 3. Presentation, review and verify 4. Structural description: - Dialog objetcs (inputs, validations, messages) - Interface representations (dtails of the form, content) 5. Sequence
Development	**Design**	Design	This deliverable includes: 1. Developers' model of the task 2. Computing specifications: a) Functional descomposition (objects + operations + interfaces). This artifact maps the real-world work with computacional specifications assigned to objects. In it is defined how the objects sould to behave in the display, in a predecible, and consistent way, according to the conceptual model of the user. It artifact defines layout, appareance, control, behavior for each object. b) Presentation:

			- layout - appearance (graphical design) - control - comportamiento b) Interface specification: - Metaphor Design - Dialog Design - Media Design - Presentation Design
	Construction	First Prototype / Operative Prototype / Final System	It is refering to several system models, since a first prototype to the final system.
Evolution	**Operation = Postdesign**	Operation Register Evaluation Documents	Operative Register is refering to registrer of operative metrics. Evaluation Documents is referring to register of analysis of such metrics resumed in assesment or decisions about the evaluation to evolving.

5 Mapping with the international standard ISO-13407.

In this section, we map the activities-products of the process defined in the standard ISO-13407 "Human-centered design process for interactive systems" to the new proposed generic GDP.

5.1 Overview of ISO-13407

ISO-13407 is an international standard established in 1999. The standard provides guidance on human-centred design activities throughout the life cycle of computer-based interactive systems. The standard aims at "those managing design processes" and does not provide detailed coverage of methods and techniques. ISO 13407 describes user-centered design from four aspects:

Rationale. It describes the benefits of usable systems; as training, support and cost reduction, and more satisfied and productivity user.

Planning. It provides guidelines to introduce the activities of the process design into the SDLC.

Principles. It identifies four general principles that characterize user-centered design that can be followed over all design process of User Interfaces: (1) The active involvement of user and a clear understanding of user and task requirements; (2) an appropriate matching and allocation of functions between user and technology; (3) iteration of design process of the User Interfaces; and (4) Multi-disciplinary design.

Activities. It refers to the activities of the design process of User Interfaces showed in Fig. 2, and briefly described in Table 13. The model in Fig. 2 shows two circles involving the part where the usability guidelines and principles are applied to the design process of User Interfaces.

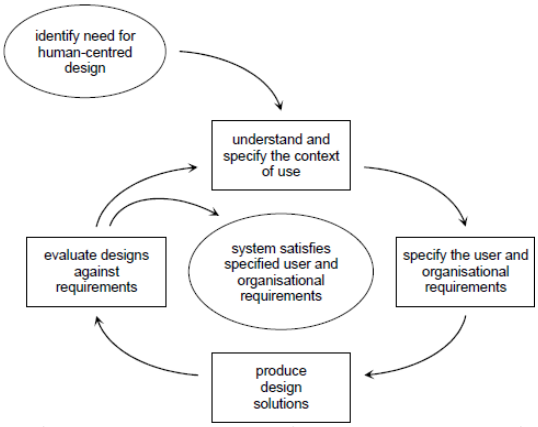

Figure 2. Activities of the standard ISO-13407 for the design process of User Interfaces.

Table 13. Description of Activities of the standard ISO-13407 for the design process of User Interfaces.

Activity	Description
Understand and Specify Context of Use.	Know the user, the environment of use, and the tasks that he or she uses the product for.
Specify the User and Organizational Requirements.	- Determine the success criteria of usability for the product in terms of user tasks, e.g. how quickly a typical user should be able to complete a task with the product. - Determine the design guidelines and constraints.
Produce Design Solutions.	Incorporate HCI knowledge (of visual design, interaction design, and usability) into design solutions.
Evaluate Designs against Requirements.	The usability of designs is evaluated against user tasks.

Table 14 shows the products of each activity (similar on how they are presented for H1 to H10 in Tables 2 to 11), but with the description of the artifact.

Table 14. Activities, and its product (deliverable artifacts) with their descriptions proposed in the standard ISO-13407 for the design process of User Interfaces.

Activity	Deliverable Artifact	Artifact definition
Understand and Specify Context of Use.	Listo of user and their relevant characteristics	The relevant characteristics of the user can include knowledge, skill, experience, education, training, physical attributes, habits, preferences and capabilities.
	List of Tasks the user perform; and their description.	The description should include the overall goals of the use of the system. Tasks should not be described solely in terms of the functions or features, but in terms of usability.
	Environment of Use	The environment includes the hardware, software and materials to be used. Their description can be in terms of a set of products, one or more of which can be the focus of human-

		centred specification or evaluation, or it can be in terms of a set of attributes or performance characteristics of the hardware, software and other materials. Relevant characteristics of the physical and social environment should also be described. These can include relevant standards, attributes of the wider technical environment, the physical, ambient, legislative and the social and cultural environment.
Specify the User and Organizational Requirements.	Requirements Analysis	- Required performance of the new system against operational and financial objectives; - Relevant statutory or legislative requirements, including safety and health; - Co-operation and communication between user and other relevant parties; - The user' jobs (including allocation of tasks, user' well-being, and motivation); - Task performance; - Work design and organization; - Management of change, including training and personnel to be involved; - Feasibility of operation and maintenance; - The human-computer interface and workstation design.
	Design Metrics	The specification of user and organizational requirements should provide measurable criteria against which the integrated design can be tested.

5.2 Mapping the activities and deliverable products of ISO-13407 with the generic GDP

We donnot pretend to cover all points of view of the standard. We have carried out a mapping of activities and deliverable products from the process point of view. The mapping was carried out against the standard ISO-13407 and the mapping is only focused on its proposed activities and the products of these activities. This mapping is shown in the Table 15.

Table 15. Mapping of activities of the standard ISO-13407 with the proposed generic GDP.

Macro-Phase	Phase	Generalized Deliverables	Standard ISO Products	Standard ISO Activities
Definition	Requirements = Predesign	Plan	Plan (implicit in the activity)	Planning (in an implicit way as an aspect defined on the standard)
		Definition of behavioral objectives.	List of user and their relevant characteristics List of Tasks the user perform; and their description. Environments of Use	Understand and Specify Context of Use.
		Requirements Specification	Requirements Analysis Design Metrics	Specify the User and Organizational Requirements.
		Preliminary Design (sketched)		

Development	**Design**	Design	Design (implicit for the activity)	Produce Design Solutions.
	Construction	First Prototype / Operative Prototype / Final System	Prototyping (in an implicit way for user-centered design abroad by the standard)	
Evolution	**Operation = Postdesign**	Operation Register Evaluation Documents	Evaluation (implicit for the activity)	Evaluate Designs against Requirements.

We can see that the standard ISO-13407 matches with the proposed GDP respect deliverable artifacts.

6 Conclusions and Future Works.

The issue about separation versus not separation of such processes is considered -in several studies- as a complex task; for example Zang [14] presents an integrated methodological proposal. In [12] several arguments are presented about the alternatives of keeping GUI activities in the SDLC against to have two separate processes. We claim that an integration of SDLC and GDP enables us to avoid duplication of tasks mainly in user analysis, task analysis and functional analysis. On the other hand, a well-defined process for developing GUIs enables us to simplify the GUI design of experts and support the GUI design of novices. We highlight the importance of integration of SDLC with GDP, for evolving toward a better process to reach successful software applications. Also, we claim this study would to be useful as a base to propose a Model Driven Me-thodology as a future work, to develop a GDP.

This study paves the way towards an explicit GDP and a better understanding on how a well-constructed GUI can be developed. Also this study represents some progress towards a systematization of the GDP to enable the integration of it into an SDLC. This work focuses on the methodological part of the design process of GUI. Finally, as future work we are looking at: (a) a generic GDP that can be easily and explicitly integrated with either model of SDLC; (b) the integration of such generic GDP with the related generic model of deliverable artifacts; and finally (c) a complete and generic GDP that includes roles and tools to reach a well-constructed GUI.

References

1. Abd Majid, R., Md Noor, N. L., Wan Adnan, W. A., Mansor, S.: A Survey on HCI in the Software Development Life Cycle: from Practitioner's Perspective. ACM 978-1-60558-710-3/09/11. pp. 1-4. (2009)
2. Bailey, G.: Iterative Methodology and Designer Training in Human-Computer Interface Design. ACM INTERCHI'93. pp. 198-205. (1993)
3. Butler, A.: Usability Engineering Turns 10. ACM Interactions. pp. 59-75. (1996)
4. Gould, J., Lewis, C.: Designing for Usability: Key Principles and What Designers Think. Communications of the ACM Volume 28 Number 3. pp. 300-311. (1985)
5. Granollers, T., Lorés J., Cañas, J. J.: Diseño de Sistemas Interactivos Centrados en el Usuario. Editorial UOC. España. (2005)
6. Hartson, H.R., Hix, D.: Human-Computer Interface Development: Concepts and Systems for its Management. ACM Computing Surveys Volume 21 Number 1. pp. 5-92. (1989)
7. Lowgren, J., Stolterman, E.: Design Methodology and Design Practice, ACM Interactions. pp. 13-20. (1999)
8. Losada, B., López, D., Martínez, J.: Guía de actuación en el desarrollo de interfaces de usuario según la metodología centrada en el usuario INTEGRAM. Depto. de Lenguajes y Sistemas Informáticos, Facultad de Informática. Universidad del País Vasco. pp. 1-10. (2004)
9. Nielsen, J., Bellcore: The Usability Engineering Life Cycle. IEEE. pp. 11-22. (1992)
10. Oktaba, H., Ibargüengoitia, G.: Software Process Modeled with Objects: Static View. Computación y Sistemas Vol. 1 No. 4 CIC-IPN ISSN 1405-5546. pp. 228-238. (1998)
11. Rodríguez, L. C., Mora, M., Alvarez, F. J.: Process Models of SDLCs: Comparison and Evolution, Chapter on Handbook of Research con Modern Systems Analysis and Design Technologies and Applications. Minnesota State University. Mankato, USA. pp. 76-89. (2008)
12. Rodriguez-Martinez, L. C., Mora, Manuel, Muñoz, J., Mendoza-Gonzaáez R.: A Descriptive-Comparative Study of Activities of GUI Development Processes and their Evolution. CONISOFT 2012. Abril 25-27, 2012. pp. 17-24. (2012)
13. Rosson, M., Maass, S., Kellog, W.: Designing for Designers: Analysis of Design Practice en the Real World. ACM CHI+GI 87. pp. 137-142. (1987)
14. Zhang, P., Carey, J., Tremaine, M.: Integrating Human-Computer Interaction Development into the Systems Development Life Cycle: A Methodology. *C*ommunications of the Association for Information Systems Volume 15, Conference. June 14-16, 2005. pp. 512-543. (2005)