

Defining a notation for usability-oriented interaction and navigation modeling for interactive systems

Anna Beatriz Marques
*USES Research Group, Institute of
 Computing*
Federal University of Amazonas
 Manaus, Brazil
 anna.beatriz@icomp.ufam.edu.br

Simone Diniz Junqueira Barbosa
Informatics Department
PUC-Rio
 Rio de Janeiro, Brazil
 simone@inf.puc-rio.br

Tayana Conte
*USES Research Group, Institute of
 Computing*
Federal University of Amazonas
 Manaus, Brazil
 tayana@icomp.ufam.edu.br

Abstract—Modeling the interaction and navigation of an interactive system can assist designers in making decisions about how the users will be able to achieve their interaction goals. However, there is a lack of proposals to: (1) deal with interaction and navigation in an integrated way and (2) deal with usability features in interaction and navigation modeling. In this paper, we propose a usability-oriented interaction and navigation model to improve the quality in use of interactive systems. We evaluate the feasibility of the model through a study with three participants with experience in using models in industry, teaching models and carrying out academic research about models. Our main contributions are: (1) a knowledge base about the existing solutions for the problem, (2) the USINN (Usability-oriented Interaction and Navigation) model, (3) a preliminary evaluation about the feasibility of USINN, (4) the evolution of the USINN notation based on the results of the feasibility study, and (5) the definition of the USINN metamodel.

Keywords— *usability-oriented design, interaction model, navigation model, software quality.*

I. INTRODUCTION

Usability is a strategic factor that offers several benefits, such as reducing training costs, improving customer satisfaction, and reducing user errors [1]. Usability can be described in terms of functional features that affect not only the interface, but also the whole user-system interaction, such as the possibility of providing commands to the user for undoing actions, validating user requests and providing appropriate feedback [2]. If these aspects are not considered in the early stages of software development and the usability problems are detected after the software codification, there is a high probability of rework to include them later [3]. Therefore, usability should be considered in the design decisions early in the development process, as well as other quality criteria such as performance and reliability [4][5][6].

Functional usability features (FUFs) are being integrated into models used in Software Engineering, such as the UML (Unified Modeling Language) use case, sequence, and class diagrams [5][6]. However, those models do not detail the user-system interaction in terms of user actions, system responses, data validation, and alternative interaction paths [7][8]. Consequently, even if the usability mechanisms are implemented, the interaction design may not be adequate to

support their use of by the application users, causing difficulties at interaction time.

In the Human-Computer Interaction (HCI) area, interaction models allow us to represent the user-system interaction, with focus on the user interaction goals [9]. Interface mockups can be developed based on interaction models [10]. However, through a case study, we identified that when using interaction models as the basis for the creation of mockups, the designers face difficulties to identify specific aspects of the interface, such as the navigation between the mockups [10].

Navigation models basically comprise navigation nodes (a set of information or functionalities that will be presented to users) and navigation flows between nodes [11]. Traditionally, the navigation model is derived from the structural model of the software to organize the content to be made available to the user. However, modeling the navigation according to the way in which the user will explore the application, i.e. according to the user interaction, helps to obtain navigational paths with higher usability [12].

Navigation and interaction are interrelated aspects of design, since navigation flows are consequences of the interaction between the user and the system. For instance, when a user requests to buy a product, the system should direct the user to see the information about the purchase and provide feedback about the system status. Conversely, not all interactions trigger navigation. For instance, when a user views the list of products and decides to bookmark a product as a favorite one, the system does not need to direct the user to another navigation node.

When we conducted a systematic mapping of the literature about interaction and navigation models, we identified gaps regarding: (1) models that represent the interaction and navigation in an integrated manner; and (2) models that consider functional usability features related to user navigation and interaction. Considering functional usability features in interaction and navigation models can help the designer think about how usability mechanisms can support the user interaction and navigation on a specific system, improving its quality of use.

To fill these gaps, this paper presents a solution to support the interaction and navigation modeling oriented by usability mechanisms: the USINN (USability-oriented INteraction and Navigation) model. This paper describes a feasibility study

conducted in order to analyze the USINN acceptance from the point of view of participants with previous experience in adopting, teaching, and researching models. We also evaluated the USINN effectiveness with regards to a set of requirements that were defined for it to meet. Based on the results of the study, we made improvements in the USINN notation and we developed the USINN metamodel. The evolution of USINN and the USINN metamodel are also presented in this paper.

This paper is organized as follows: Section II presents the concepts and existing solutions regarding interaction and navigation modeling. Section III presents an analysis of usability in interaction and navigation models. In Section IV, we describe the creation of the USINN notation, as well as the initial version of USINN. Then, Section V presents the feasibility study we conducted and the obtained results. Section VI discusses the USINN improvements based on the results of the feasibility study, showing a new version of USINN notation. Finally, in Section VII, we discuss our final considerations and future work.

II. BACKGROUND AND RELATED WORK

In this section, we present the concepts related to our research and related work.

A. Interaction and Navigation Modeling

An interaction model describes the communication between the user and the system, specifying [10][9]: (a) when the user can perform specific tasks to achieve certain goals; (b) when the user can input some data; and (c) when the system can process the user input and show the appropriate contents and feedback. The focus of an interaction model is to represent the communication processes between the user and the system, but not necessarily the details about the specific elements of the user interface [13]. However, there are interaction models which represent the interaction and strictly related it to the components of the application interface, such as the Diamodl [14] and IFML [15] models. In such models, the user interaction is represented in terms of events over the interface

components, such as clicking a button, selecting an element, among others.

The focus of our research is on the interaction models that do not consider the concrete aspects of the interface. This decision was made since such models can be adopted in the initial phases of system development, in which the interface design decisions have not yet been made [10]. As suggested by Beaudouin-Lafon [16], we believe that one way to improve the quality of the user interfaces is to change the focus from interface design to interaction design, since the interface is the means through which the interaction occurs.

Modeling the system navigation aligned with the user-system interaction can help to define navigational paths aligned with the user interaction goals [12]. Navigational models are usually composed by nodes and links [11]. A node represents a set of information or a functionality that will be presented to users. Links are used to join nodes, representing the possibility of navigating from one node to another. A navigation model defines how the available content and functionality will be associated, indicating valid navigational paths [17].

In order to characterize the state of the art regarding the existing solutions for interaction and navigation modeling, we conducted a systematic mapping [18]. Unlike informal literature reviews, where the researcher does not follow a defined process, a systematic mapping follows a pre-established protocol subject to review and repetition [19]. As a result, we have identified 14 different notations for modeling the interaction or navigation of interactive systems. We carried out a comparative analysis with regard to the interaction and navigation elements considered in each notation, as shown in Table I.

To define the interaction and navigation elements described in Table I, we used the above definitions of interaction [10][9] and navigation models [11] as basis. We highlight that the notations may use different terminologies for the elements described in Table I. Thus, we analyzed the semantics of the notation, identifying the meaning of each element.

TABLE I. SUMMARY OF THE CHARACTERISTICS OF THE IDENTIFIED NOTATIONS FOR INTERACTION AND NAVIGATION MODELING.

Notation	Type of system	Interaction elements				Navigation elements	
		User actions	System responses	Data validation	Interaction flow	Navigation nodes	Links
TADEUS [20]	Generic	X			X	X	X
PUAN [21]	Generic	X	X	X			
Lean Cuisine + [22]	Generic	X	X				
DIGBE [23]	Digital control system	X					
MoLIC [9]	Generic	X	X	X	X		
OCD [24]	Generic	X	X	X	X		
CRITON [25]	Hypermedia system					X	X
PSDM [7]	Web system	X			X		
CTDM [26]	Generic	X	X	X	X		
OntoUCP [27]	Generic	X	X		X		
NIM [28]	Web system	X			X	X	X
Diamodl [29]	Generic	X	X		X		
CIAN [30]	Colaborative system	X	X		X		
DSM [31]	Multiplatform system	X			X		

With regards to interaction, the MoLIC [9], OCD[24], and CTDM [26] notations represent all the elements to be considered in interaction modeling. Regarding navigation, TADEUS [20], CRITON[25], and NIM [28] represent all the elements related to navigation. However, we did not identify a notation that integrates both aspects, representing all the necessary elements to model both interaction and navigation.

The identified notations consider different elements as the basis for designing models. The TADEUS [20] and CRITON [25] notations use interface presentation units as basis (for example, views or web pages). On the other hand, the PUAN [21], OCD [24], CTDM [26], OntoUCP [27], PSDM [7], and CIAN notations are task-oriented. The DSM [27] notation is based [30] states, which group related tasks during the interaction. The MoLIC [9] model is based on user interaction goals. Lean Cuisine+ [22], DIGBE [23], and Diamodl [29] consider interface elements as the basis for the interaction modeling. Finally, the NIM [28] model considers use cases as the basis for modeling. This variety of the basis for modeling the interaction makes it difficult to integrate interaction and navigation, in which one has to define the navigation nodes organizing content or features.

We identified few empirical studies to evaluate the existing notations; only MoLIC [9] and OCD [24] were evaluated through empirical studies. This lack of evidence on the benefits of the notation can make their adoption by the software industry difficult, since companies do not have indicators that show the benefits of applying the proposed notations.

B. Usability-oriented Systems Design

To deal with usability as a non-functional requirement does not provide sufficient information to develop features that meet the usability requirements [4]. Juristo et al. [2] identified a set

of functional usability features (named FUFs) that affect the user interaction with the system, not only the user interface (Table II). To verify how much FUFs affect interactive systems design, the FUFs were incorporated in various real object-oriented systems and the following metrics were analyzed: number of expanded use cases affected; number of new classes; complexity of the created methods; coupling of new classes with other domain classes [2].

Although the impact of incorporating FUFs is directly related to the specific functionalities of each system, it was possible to conclude that mechanisms such as cancel operations and system feedback for the user have high impact on the user interaction with an interactive system [2]. In order to assist the identification of the necessary FUFs to be implemented in an interactive system, Juristo et al. [32] elaborated guidelines for usability requirements elicitation. As each FUF can be implemented in different ways, Juristo et al. [32] also defined subtypes for each FUF, called usability mechanisms (described in Table II). Furthermore, for each usability mechanism, Juristo et al. [32] elaborated guidelines for functional usability requirements elicitation and specification. The guidelines are composed of questions to assist the practitioners when conducting the requirements elicitation with stakeholders, even if they are not usability specialists.

Carvajal et al. [5] developed guidelines for incorporating usability mechanisms from the FUFs in use cases, classes, and sequence diagrams. The main motivation for incorporating usability in models is related to the improvement of the quality in use of the system. However, the employed models do not detail the user interaction with the system, in terms of user actions, system responses, data validation, and alternative paths, and do not explore the interaction in terms of usability.

TABLE II. FUNCTIONAL USABILITY FEATURES (FUFs) AND THEIR ASSOCIATED USABILITY MECHANISM (BASED ON [2] AND [32]).

Functional Usability Features	Description [2]	Usability mechanism	Goal (adapted from [32])
Feedback	To inform users about what is happening in the system	System status	To inform users about the internal status of the system
		Interaction	To inform users that the system has registered a user interaction
		Warning	To inform users of any action with important consequences
		Long action feedback	To inform users that the system is processing an action that will take some time to complete
Undo / Cancel	To undo system actions at several levels. To cancel the execution of a command or an application	Global undo	To undo system actions at several levels
		Object-specific undo	To undo several actions on an object
		Abort operation	To cancel the execution of an action or the whole application
		Go back	To go back to a particular state in a command execution sequence
User input errors prevention/correction	To improve data input for users and software correction	Structured text entry	To help prevent the user from making data input errors
Wizard	To help do tasks that require different steps with user input	Step-by-step execution	To help users do tasks that require different steps with user input, and to correct such input
User profile	To allow the users to customize the system according to their preferences.	Preferences	To record each user's options for using system functions
		Personal object space	To record each user's options for using the system interface
		Favorites	To record certain parts of the system that are of interest for the user
Help	To provide different help levels for different users	Multilevel help	To provide different levels of help to different users
-	-	Command aggregation	To express possible actions to be taken with the software through commands that can be built from smaller parts

III. USABILITY IN INTERACTION AND NAVIGATION MODELS

Interaction models have the advantage of including usability mechanisms, since these models focus on user-system interaction. Incorporating usability mechanisms in models that help the designer to think about the user interaction can maximize the benefits in the final product quality. However, none of the interaction and navigation models identified through our systematic mapping focuses on representing functional usability features. In order to evaluate whether and how the identified interaction and navigation modeling notations (Table I) allow the representation of usability mechanisms, the first author analyzed their elements and verified which elements allow to model the usability mechanisms described in Table II.

The investigation was based on the publications related to the notations that we identified in our systematic mapping. The PUAN [21], DIGBE [23], and Diamodl [29] notations were not included in the analysis because their authors did not provide enough details about their elements. As a result of the analysis, we obtained a mapping between usability mechanisms and notation elements. This mapping was reviewed by the two co-authors of this paper. Table III illustrates the mapping between usability mechanisms and interaction and navigation models, indicating which models allow modeling usability mechanisms.

TABLE III. USABILITY MECHANISMS IN NOTATIONS FOR INTERACTION AND NAVIGATION MODELING.

Usability mechanism	Interaction and Navigation Model										
	TADEUS	Lean Cuisine +	MoLIC	OCD	CRITON	CTDM	OntoUCP	NIM	PSDM	CIAN	DSM
System status	X	X	X		X	X		X	X	X	X
Interaction											
Warning			X								
Long action feedback			X								
Global undo	X		X						X		X
Object-specific undo	X		X						X		X
Abort operation			X								
Go back	X		X		X						X
Structured text entry			X	X		X	X		X		
Step-by-step execution	X		X		X	X					X
Preferences	X	X	X	X	X	X	X	X	X		X
Personal object space	X	X	X	X	X	X	X	X	X		X
Favorites								X			
Multilevel help	X	X	X	X	X	X	X	X	X		X
Command aggregation	X	X	X	X	X	X	X	X	X		X

We identified that the *interaction*, *warning*, *abort operation*, *step-by-step execution*, *personal object space*, and *favorites* usability mechanisms cannot be represented by most of the notations. On the other hand, the *system status* and *multilevel help* usability mechanisms can be represented by most of the models. When carrying out this analysis, it is important to discuss which usability mechanisms have a higher impact on the user interaction and navigation. Juristo et al. [2] explored the impact of the FUFs on the system functionalities. Starting from this analysis, we can associate the interaction and navigation with the system functionalities.

With regards to the interaction, we can associate the user interaction with the system functionality, since the system functionalities must meet the user interaction goals. The navigation is also defined by the functionalities available in the system, since the user can navigate to the functionalities available in a given moment of the interaction.

Juristo et al. [2] showed that the *cancel* and *feedback* FUFs have a high impact on the system's functionality. Such FUFs are associated to the *system status*, *interaction*, *warning*, *long action feedback*, *undo*, *abort operation*, and *go back* usability mechanisms. Among these mechanisms, only the *system status*, *undo*, *abort operation*, and *go back* are supported by four or more notations. Conversely, not all of the usability mechanisms require specific elements in the notations to be represented because they are similar to common features of the system: to define the configuration preferences, to view help content, to create a macro of the system commands.

The *preferences*, *personal object space*, *multilevel help*, and *commands aggregation* usability mechanisms fit in this case, in which specific elements are not necessary to represent them. The interaction elements themselves can be used to represent such usability mechanisms when necessary. The NIM model differs from other models regarding the *preferences* and *personal object space* mechanisms, its data collection element allows to store the user data preferences.

As a result, although the focus of the identified models in the systematic mapping does not explicitly consist in supporting design considering usability, the models partially support the representation of usability features. However, we identified usability mechanisms that have a strong impact on interaction and that cannot be represented by the notations.

We can highlight MoLIC as a that covers the majority of usability mechanisms. However, in previous studies [10][33] we conducted, the participants did not perceive MoLIC as a model to support usability. Actually, MoLIC is a notation based on Semiotic Engineering theory [34] and its elements are not directly related to usability aspects. Besides that, the MoLIC elements are not focused on representing navigation aspects.

Although navigation and interaction are interrelated aspects of design, the existing solutions for interaction and navigation modeling consider the representation of these aspects separately. Thus, the designers need to adopt different notations to model these aspects. Maintaining the different interaction and navigation models consistent among themselves is not easy, because the interaction models use

different elements as their basis, while navigation models in general are based on presentation units of the interface.

Based on this perspective, we have proposed a notation for interaction and navigation modeling, incorporating usability mechanisms, called USability-oriented INteraction and Navigation model (USINN). Our goal is for USINN to be adopted in interactive systems design before the construction of the interface prototypes. The interaction and navigation modeling can be an artifact used as basis for creating prototypes with higher usability.

IV. CREATING A NOTATION FOR USABILITY-ORIENTED INTERACTION AND NAVIGATION MODELING

In our previous work, we conducted exploratory empirical studies about interaction modeling [10][33]. The data collected in these studies allowed us to identify an initial set of requirements for a model that integrates interaction and navigation aspects. In this section, we present the set of requirements identified in these experiments and the process of developing a new notation for usability-oriented interaction and navigation modeling.

A. Identifying requirements for a new notation

Two experiments provided qualitative data about existing solutions: (1) a case study to investigate whether and how interaction models and interface prototypes can be used together during design [10]; and (2) a comparative study between different interaction models [33]. In these experiments, the participants answered questionnaires, providing data about their perception of the interaction models. We applied qualitative analysis procedures [35] to the collected data and generated a set of requirements for a new model (Table IV). We organized the requirements in three categories: learnability of the model, support for interaction and navigation modeling, and support for interface design.

In the **requirements for the learnability of the model** category, we identified features that the model must meet in order to facilitate its learning by novice designers. In the **requirements for supporting interaction and navigation modeling** category, we identified features that the model must have to enable the modeling of all aspects relevant to the interaction and navigation. The **requirements for supporting interface design** category describes features that the model must possess in order to be a resource for interface design, aligning the interaction and navigation solutions to the interface solutions.

B. USINN: USability-oriented INteraction and Navigation Model

Considering that an interaction and navigation model can be used as the basis for interface design, if usability mechanisms are adequately represented in these models, the interface can reflect the usability features incorporated in those models. Therefore, based on the requirements identified and the usability mechanisms discussed in Section II, we have proposed a new model named USINN (USability-oriented INteraction and Navigation).

TABLE IV. REQUIREMENTS FOR A NOTATION THAT INTEGRATES INTERACTION AND NAVIGATION ASPECTS.

Requirements for the learnability of the model	
R1	The model must be easy to learn.
R2	The model must be easy to understand.
R3	The model must facilitate the learning of the interaction elements.
Requirements for supporting interaction and navigation modeling	
R4	The model must detail the user and system actions during the interaction.
R5	The model must represent the possibilities of errors and alternative interaction paths.
R6	The model must help to identify business rules.
R7	The model must clearly define the actions and data involved in the interaction.
R8	The model must represent the manipulated data during the interaction.
R9	The model must represent the relationship between the interaction goals.
R10	The model must support the definition of the application navigation.
R11	The model must provide an overview of the application behavior.
Requirements for supporting interface design	
R12	The model must assist in the identification of the necessary prototypes.
R13	The model must help to start prototyping.
R14	The model must help to identify the content of the interface.
R15	The model must represent aspects of the interface layout.
R16	The model must support the development of prototypes with a high degree of usability.
R17	The model must help the complete development of the prototypes.

USINN is intended to provide elements to visually represent the interaction and navigation of interactive systems and the associated usability mechanisms. We expect that when USINN is used as the basis for the creation of interface prototypes, the usability of the final product will be improved.

The USINN elements can be organized according to their purpose: to represent interaction aspects, to represent navigation aspects, or to represent usability mechanisms. We highlight that some elements can have more than one purpose (e.g. to represent interaction and usability aspects simultaneously). Fig. 1 illustrates the elements of USINN.

To represent the navigation of an interactive system, USINN has the following elements:

State: it is the basis of the navigational structure of the system and it groups user actions to achieve the interaction goal described by the state description.

Navigation: it is a relationship between the system states. It indicates that the user can navigate to the next state and/or return to the previous one. It defines how each state will be accessible to the user.

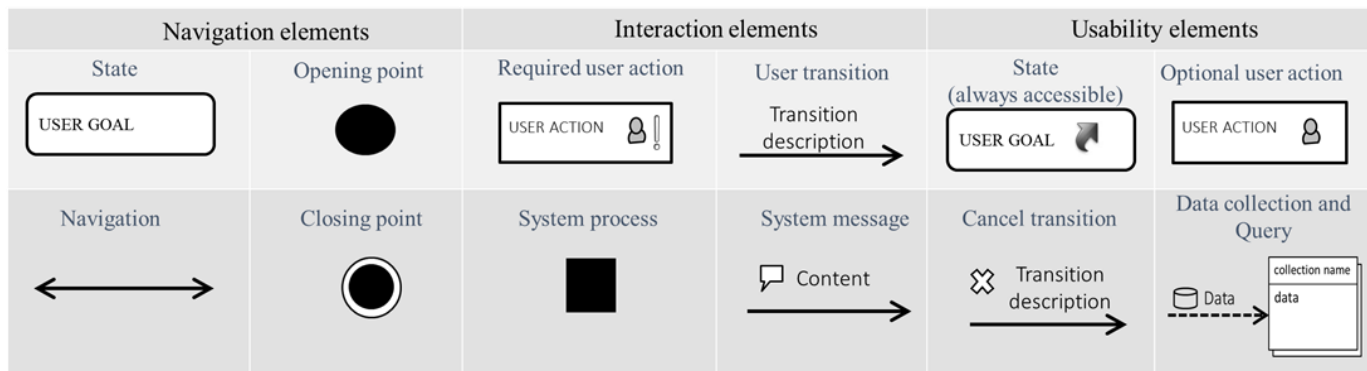


Fig 1. USINN model elements to represent interaction, navigation, and usability.

Opening point: It indicates the start of the user navigation/interaction with the system, that is, when the user accesses the system.

Closing point: It indicates the end of the navigation/interaction, that is, when the user exits the system.

The user interaction with the system is represented by the following elements of the USINN notation:

Required user action: It describes a necessary action to achieve an interaction goal.

User transition: It represents a user intention to proceed with the interaction.

System process: It represents an internal system verification after a user request.

System message: It describes the system feedback provided in response to a user request. It occurs after a system process.

The USINN usability-related elements are aligned with the usability mechanisms discussed by Juristo et al. [32], since they affect the user interaction with the system. In order to represent these usability mechanisms, we noticed that additional elements were necessary. Thus, the USINN has the following specific elements to represent usability mechanisms:

State (Always accessible): It represents a state that must be always accessible during the interaction.

Optional user action: It describes an optional user action to achieve an interaction goal.

Cancel transition: It describes that there is a possibility for the users to undo or cancel actions.

Data collection: This element contains the manipulated data during user interaction. The user personal preferences and favorites can be stored in a data collections.

Query: It represents an operation at the user interface that manipulates the collected data.

The only usability mechanisms that we could not consider in the USINN notation were: *interaction* and *structured text entry*, because to include them, we would need to consider the concrete aspects of the interface, such as a mask in a text field (defining the format for data input) or a color change on the

edge of a button [5]. As an interaction model, USINN does not deal with concrete aspects of the interface, so that it can be adopted in the early stages of software development, where decisions about the interface have not already been made.

Fig. 2 illustrates a USINN diagram related to a system to manage support services of an institution, named AcadSupport. The AcadSupport is an application that teachers can use to request support services in an institution, while technicians can use it to manage the support demands. The support requests are received by the institution's support technicians. After the beginning of the request fulfillment, the support technicians update the status of the request. In Fig. 2 we can see a partial interaction of a technician with the AcadSupport system.

The user navigation occurs through the following *states*: "access AcadSupport", "view requests history", "view details of a request", and "exit the AcadSupport". At any moment during the interaction, the user can navigate to the following *states*: "view requests history" or "exit the AcadSupport", because these are states that are always accessible.

When detailing the interaction, one can see that, for accessing the system, the user must perform the "inform registration and password" action, a *required user action*. The system validates the data by consulting the "user data" *data collection* and then provides feedback to the user, informing whether an error occurred during data input (*message* "invalid login") or the inputted data is valid (*message* "valid login"). In the former case, the user is led back to the "inform registration and password" action; in the latter, the user navigates to the "view requests history" *state*.

In the "view requests history" *state*, the user views the list of support requests. The data of these requests is detailed in the *query* that consults the "support request" *data collection*. The user can "select a request to view its details", which is an *optional action*. If the user selects a request, the user navigates to the "view details of a request" *state*, in which the user views the details, and can update the status of the request. In order to update the request's status, the system validates the new status and provides appropriate feedback to the user.

In order to evaluate whether the USINN meets the defined requirements and allows representing the discussed usability mechanisms, we decided to conduct a feasibility study.

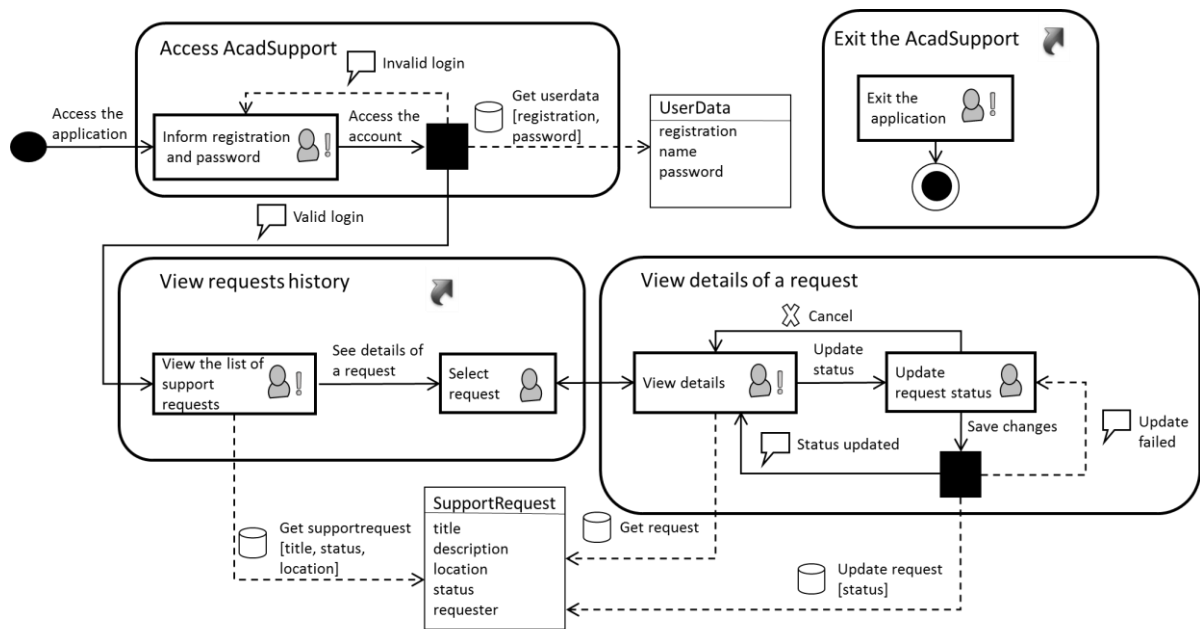


Fig 2. Example of a USINN diagram.

V. FEASIBILITY STUDY OF THE USINN MODEL

In the context of empirical studies, a feasibility study aims to verify whether a new technology is feasible and whether the time spent is well used [36]. A feasibility study is not intended to obtain a definitive answer, but to obtain data to refine the solution and generate hypotheses associated with its use [36].

A. Planning of the Feasibility Study

In this step, we have defined the following definitions, materials and procedures:

Participants: we selected three doctoral students with a degree in Computer Science to participate in the feasibility study. The participants had different degrees of experience with: (a) the use of HCI and SE models in the industry; (b) teaching HCI and SE models; and (c) carrying out research on HCI and SE models. Table V shows the characterization of the participants from the feasibility study.

Tasks: the participants had to: (i) create a model using the USINN notation for a given scenario; and (ii) create interface prototypes based on the previously created model.

Scenario: the used scenario described a knowledge management system of a research group. The user interaction goals described in the scenario were: (i) to edit the researcher’s personal information; (ii) to view data about the researcher; (iii) to register knowledge assets; (iv) to edit knowledge assets; and (v) to view one’s contribution ranking. In the scenario, we did not explicitly describe usability mechanisms to be represented in the model, because our goal was to see whether the USINN elements would guide the participants in modeling usability features, even if the scenario did not explicitly describe usability mechanisms.

TABLE V. CHARACTERIZATION OF THE PARTICIPANTS.

Experience with HCI models			
Participant	Experience using HCI models in industry	Experience researching about HCI models	Experience teaching HCI models
P1	L	H	L
P2	L	L	L
P3	L	L	-
Experience with SE models			
Participant	Experience using SE models in industry	Experience researching about SE models	Experience teaching SE models
P1	H	M	L
P2	H	M	M
P3	L	H	L

Footnote: L (up to 1 year); M (between 1 to 2 years); H (more than 2 years); - (No experience).

Setting: we conducted the feasibility study in a research lab.

Metrics: in order to analyze the feasibility of USINN, we used objective metrics about the quality of the produced artifacts by the participants and subjective metrics about the participants’ perception of the USINN model. We defined the objective metrics based on the quality objectives of conceptual models discussed in [37]:

- **Completeness:** to what extent a model presents the necessary information according to the purpose of the modeling. This metric is related to the semantics of the model. The omission of information reduces completeness.
- **Correctness:** to what extent a model: (i) correctly employs the elements and relationships, according to the syntax notation, and (ii) correctly describes the application domain according to the available

information. The incorrect use of the notation elements and inconsistencies between the contents of the model and the available domain information reduces the correctness.

The subjective metrics were based on the Technology Acceptance Model (TAM) [38] and the previously defined requirements (see Table IV).

- Perceived usefulness: to what extent a person believes that using a particular technology would improve her/his performance in certain tasks.
- Perceived ease of use: to what extent a person believes that using a particular technology would be free of effort.
- Behavioral intention to use: to what extent a person believes that (s)he would use a particular technology in the future.
- Perceived effectiveness: to what extent a person believes that the USINN model meets the requirements (see Table IV) for which it was proposed.

In order to collect data about the subjective metrics, we developed a questionnaire with statements about each metric. The statements we employed to measure the perceived usefulness, perceived ease of use, and behavioral intention to use were adapted from Davis's [39] original proposal of the TAM questionnaire. The statements are described in Table VI. To measure the perceived effectiveness, we have adapted the requirements defined for USINN (see Table IV), turning them into statements. For example, the requirement "the model must help to identify business rules," was modified to "the model helps to identify business rules".

In the questionnaire, we employed a six-point ordinal scale ranging from totally agree to totally disagree. As suggested by Laitenberger and Drayer [38], the neutral point (neither agree nor disagree) was not used in the ordinal scale, since it does not allow to identify the degree (either positive or negative) of agreement of the participants.

Additionally, we included some open questions about the USINN model in the questionnaire. These questions are described as follows:

- (1) Comment about positive and negative aspects of the use of this model for interaction and navigation modeling. Your comments will help us to improve the model.
- (2) Were there any interaction and/or navigation aspects that you failed to represent using the model's elements?
- (3) Would you recommend this model for practitioners who work with the design and development of interactive systems?
- (4) Do you consider that this model would help design the interaction and navigation with a focus on usability?

Such questions would provide qualitative data to help us understand which aspects of the model would influence its acceptance or not by its users.

Materials: to support the conduct of the feasibility study, we elaborated: (i) a consent form; (ii) a characterization form; (iii) the scenario describing the knowledge management system; (iv) a document containing the set of tasks to be performed by the participants; (v) supporting material containing the USINN notation and an example of a USINN diagram; and (vi) the post-study questionnaire previously described.

B. Feasibility Study Execution

We conducted the feasibility study individually with each participant. At the beginning of the study, the participant received the consent form and answered the characterization form. After that, the participant received the set of tasks, the scenario and the support material about USINN.

TABLE VI. STATEMENTS TO MEASURE THE PERCEIVED USEFULNESS, PERCEIVED EASE OF USE AND BEHAVIORAL INTENTION TO USE.

Perceived usefulness
PU1. Using USINN in my job would enable me to model interaction and navigation more quickly.
PU2. Using USINN would improve my job performance in interaction and navigation modeling.
PU3. Using USINN to model interaction and navigation would increase my productivity. (I believe I will have identified a higher number of elements in a shorter time).
PU4. Using USINN would enhance my effectiveness in interaction and navigation modeling (I believe I will have identified a higher number of elements with this model).
PU5. Using USINN would make it easier to model interaction and navigation.
PU6. I would find USINN useful in interaction and navigation modeling.
Perceived ease of use
PEOU1. Learning to model interaction and navigation with USINN would be easy for me.
PEOU 2. The USINN elements are clear and understandable.
PEOU 3. I would find it easy to use the USINN elements for interaction and navigation modeling.
PEOU 4. It was easy to become skillful using USINN for interaction and navigation modeling.
PEOU 5. It is easy to remember how to model interaction and navigation using USINN.
PEOU 6. I would find USINN easy to use for interaction and navigation modeling.
Behavioral intention to use
BI1. Assuming USINN would be available on my job, I predict that I will use it on a regular basis in the future.
BI2. I would prefer to use USINN over other models (e.g. UML models) for interaction and navigation modeling.

The participants did not receive previous training about the USINN model. They only received information about the study goal. The researcher responsible for the study explained to the participant that, after completing the tasks, he/she should answer a questionnaire about the experience of using the

model. We did not set a time limit for the completion of the tasks.

VI. RESULTS ANALYSIS OF THE FEASIBILITY STUDY

After the execution of the study, we analyzed the artifacts elaborated by the participants (models and prototypes) and the data collected through the post-study questionnaires. In this subsection, we discuss the obtained quantitative and qualitative results.

A. Objective Metrics: Results

In order to calculate the objective metrics, we analyzed the artifacts (models and prototypes) produced by the participants. Regarding the completeness of the models, two researchers examined whether the elaborated models met the following criteria for completeness:

- C1. For each functionality, was a state defined?
- C2. Is the state accessible during the interaction?
- C3. Are the necessary actions to achieve the goal represented in the state?
- C4. Are the user requests validated by the system and is the adequate feedback provided?
- C5. Is the data that is manipulated during actions or during system processing represented?

The scenario used as basis for the modeling tasks contained five functionalities. To model the “view data of the researcher” and “view contribution ranking” functionalities, it was not necessary to use data validation. Therefore, the C4 completeness criteria was not applied to those functionalities.

First, we calculated the completeness of the modeling of each functionality and, subsequently, we calculated the completeness of the model as a whole. The completeness of a functionality was calculated considering the percentage of the completeness criteria met in the modeling of each functionality. The completeness of the model was estimated by the average completeness of the modeled functionalities. The analysis about the completeness of the models elaborated by the participants is detailed in Table VII.

All participants represented the “edit the researcher’s personal information”, “register knowledge assets” and “edit knowledge assets” functionalities completely. However, participants P1 and P2 did not represent the manipulated data in the “view data of the researcher” functionality. Participant P1 did not represent the *navigation* to reach the *state* related to the “view contribution ranking” functionality”. Moreover, participant P3 did not model the “view contribution ranking” functionality. Therefore, we observed a higher incidence of omission with regards to the representation of the manipulated data (use of the *data collection* and *query*).

We calculated the correctness of the models as the percentage of USINN elements which were correctly used by the participants in the models. Table VIII details the analysis of the correctness of the models elaborated by the participants. We observed that the *required user action* element was incorrectly used by all three participants. It is possible that the participants had forgotten to use the exclamation icon, which indicates that an action is required (i.e., not optional).

Participants P1 and P3 incorrectly employed the *state* element. The participants represented different user interaction goals in the same state. For example, in the *state* named “view researcher information”, participant P1 also modelled the *user actions* to “edit the researcher information”. Additionally, Participant P3 used a *user transition* without description. The *always accessible state* element was used in an invalid context by participant P2, i.e. in a state that should not always be accessible. Finally, participant P1 used the *cancel transition* from an operation that could not be undone.

Analyzing which usability mechanisms had been represented in the created models, we observed that only the *system status*, *warning*, and *abort operation* mechanisms were represented by all the participants. The *undo* mechanism was represented only by one participant, as well as the *long action feedback* mechanism. The *step-by-step execution*, *preferences*, *personal object space*, *favorites*, *multilevel help*, and *command aggregation* mechanisms were not represented. This result indicates that the elements of the USINN by themselves did not guide the designers to think about modeling usability aspects of the application.

TABLE VII. ANALYSIS ABOUT THE COMPLETENESS OF THE CREATED MODELS.

Criteria	P1						P2						P3					
	C1	C2	C3	C4	C5	Total	C1	C2	C3	C4	C5	Total	C1	C2	C3	C4	C5	Total
edit researcher’s personal information	✓	✓	✓	✓	✓	100%	✓	✓	✓	✓	✓	100%	✓	✓	✓	✓	✓	100%
view data of the researcher	✓	✓	✓	n/a	✗	75%	✓	✓	✓	-	✗	75%	✓	✓	✓	n/a	✓	100%
register knowledge assets	✓	✓	✓	✓	✓	100%	✓	✓	✓	✓	✓	100%	✓	✓	✓	✓	✓	100%
edit knowledge assets	✓	✓	✓	✓	✓	100%	✓	✓	✓	✓	✓	100%	✓	✓	✓	✓	✓	100%
view contribution ranking	✓	✗	✓	n/a	✗	75%	✓	✓	✓	-	✓	100%	✗	✗	✗	n/a	✗	0%
Completeness	90%						95%						80%					

TABLE VIII. ANALYSIS OF THE CORRECTNESS OF THE CREATED MODELS.

USINN notation element	P1	P2	P3
State	✗	✓	✗
Navigation	✗	✓	✓
Opening point	✓	✓	✓
Closing point	✓	✓	✓
Required user action	✗	✗	✗
User transition	✓	✓	✗
System process	✓	✓	✓
System message	✓	✓	✓
Always accessible state	✓	✗	✓
Optional user action	✓	✓	✓
Cancel transition	✗	✓	✓
Data collection	✓	✓	✓
Query	✓	✓	✓
Correctness	69%	85%	77%

B. Subjective metrics: Results

In this section, we present the obtained results for the subjective metrics. The graphs illustrated in Fig. 3, Fig. 4, Fig. 5, and Fig. 6 detail the perception of the participants provided in the questionnaires. The data provided in the open questions allowed us to further analyze the obtained results.

Fig. 3 details the results regarding perceived usefulness. We did not identify negative perceptions with regards to the statements about perceived usefulness, which indicates a positive result for the USINN usefulness. Analyzing the participants' responses to the open questions, we identified some quotes that we can relate to the **perceived usefulness of USINN**:

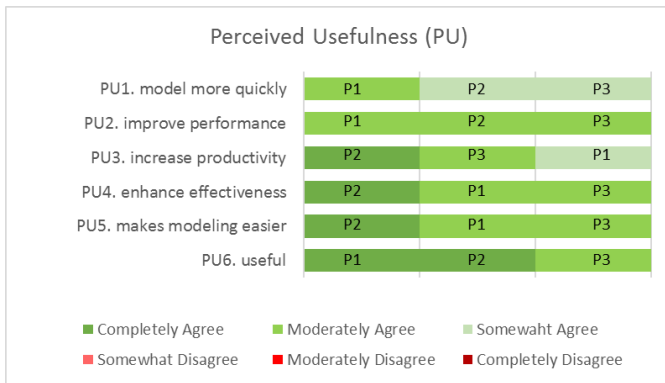


Fig 3. Perceived usefulness from the point of view of the participants.

Participant P3 stated that: *“If we used it (the model) with mockups, it can improve the user interaction”*. P3 also commented that *“Maybe the model may assist developers during implementation”*. Furthermore, Participant P2 pointed out that: *“I liked that the model involves the interface content by means of tables that can assist the creation of a database”*. Finally, participant P1 stated *“The model allows the clear understanding of the user and system actions”*.

In Fig. 4, we can see the detailing of the results for perceived ease of use.

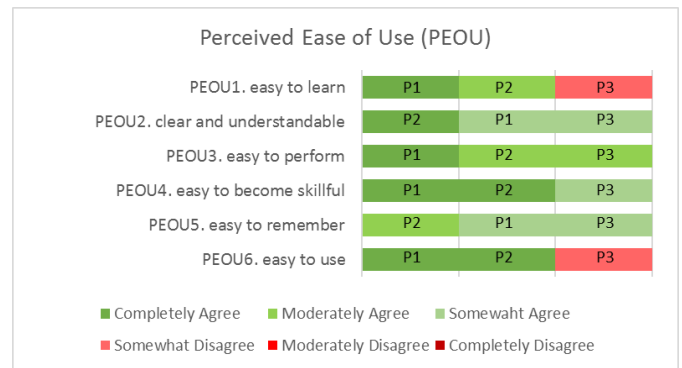


Fig 4. Perceived ease of use from the point of view of the participants.

We noticed that participant P3 disagreed with statement PEOU6, which affirms that the model is easy to use; and disagreed with statement PEOU1, which states that the model is easy to learn. We identified some quotes that we related to the **use of USINN**.

Participant P1 pointed out some difficulties such as: *“I had a doubt regarding whether, in order to represent a particular warning to the user, we had to use a state element, or we could use the system message”*. Participant P2 also indicated some difficulties in using USINN, such as: *“I got confused about representing actions that are not sequential, I don't know if I have done it right”*.

We identified quotes about the **learnability of the model**. For instance, participant P3 pointed out that *“In the beginning of the modeling task, it was confusing because the USINN elements are very similar to each other”* and *“The model is based on elements which other models already have, so we don't need to spend a lot of time learning the notation”*.

Fig. 5 shows the participants' responses regarding **behavioral intention to use USINN** in the future. We achieved positive results with regards to this indicator. The fact that the participants have prior experience with other analysis and design models strengthens the result for statement BI2, which deals with the preference for applying USINN.

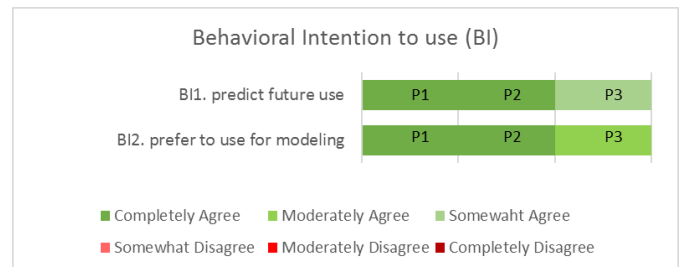


Fig 5. Behavioral intention from the point of view of the participants.

Fig. 6 details the results regarding the perceived effectiveness by the participants. Participant P3 disagreed with some statements describing the requirements of USINN. Participants P1 and P2 agreed with all of the statements, although in some cases the degree of agreement was not strong.

With regards to the perceived effectiveness of USINN, we identified quotes related to the **elements of the model**:

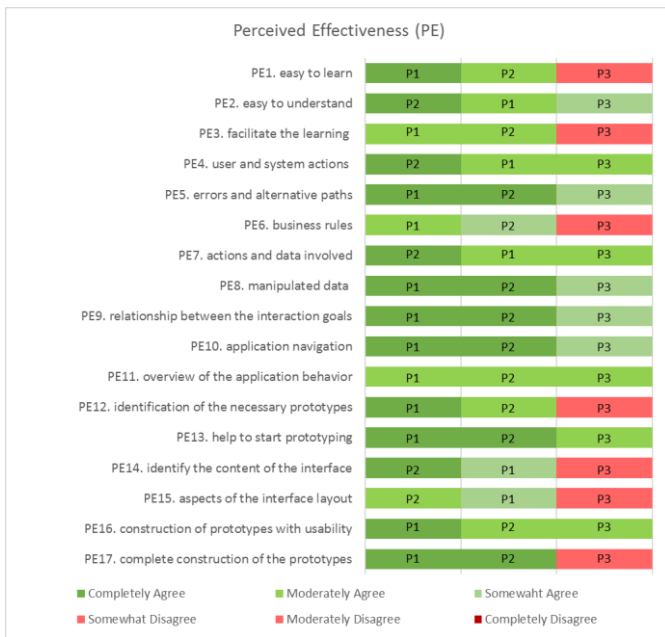


Fig 6. Perceived effectiveness from the point of view of the participants.

Participant P1 indicated that “The model helps in understanding the elements that must always be available for the user” and “It is possible to view the user actions which change the state of the system”. P1 suggested: “The USINN could have a ‘rule’ element, in which we could describe some rule for the state element. Thus, the reader could understand why the element is or not available”.

Participant P2 commented: “The model has very important elements for interaction and navigation modeling. I really appreciated the data collection element because we can know what information is being presented to the user and the query allows us to view what is going on”. Similarly, to participant P1, P2 suggested a specific element for modeling business rules “The USINN could have an element like the data collection element, but focused on business rules”. P2 also suggested that “... USINN could involve the relationships between data collections”.

Participant P3 pointed out that some interaction aspects were identified during the development of the interface prototypes: “I had a doubt when I was representing a possible user input after which we could have two interaction flows. I represented only one interaction flow. However, when I was creating the mockup I realized that I had to represent another interaction flow. So, I had to go back in the model and correct it”.

Finally, regarding the **support for including usability aspects**, participant P1 indicated that “Through the USINN model it is possible to represent the actions that the user can do (and undo), the elements in the interface that are always available, and it is possible to inform the user (feedback) about a particular action”. Conversely, P2 commented: “I think it is necessary that the designer understands the concept of usability for better modeling focusing on this aspect (usability)”. In turn, P3 stated that “I think that the USINN

model already helps to design usability, but in a ‘smooth way’, without the designer noticing it”.

Analyzing the statements related to perceived effectiveness, we observed that for attending PE15 and PE17, we must to incorporate interface aspects in the interaction modeling. However, this could force the interaction designer to make design decisions that are not related to his/her role. Thus, in future studies, we will reduce the list of requirements for supporting interface design.

The participants’ comments indicate that USINN still does not provide sufficient guidance for the designer to model the interaction and navigation considering usability, even if the usability mechanisms are not explicitly described. The comments also indicated that there is a need for a better definition of the model elements and guidelines for its use. These guidelines can help the designer understand how to adequately use the elements.

VII. EVOLVING USINN BASED ON THE FEASIBILITY STUDY RESULTS

Based on the feasibility study results, we made some improvements in USINN in order to improve its perceived usefulness, ease of use and effectiveness.

A. Refinement of the USINN elements

Analyzing the models created by the participants, we noticed that only the *system status*, *warning*, and *abort operation* usability mechanisms were represented by all participants. Thereby, we analyzed whether the model allowed to represent the other usability mechanisms and we identified that it was necessary to refine the notation.

We decided to change the name of the *system message* element to *system feedback* and the name of the *state* element to *presentation unit*. Also, we extended the *system process* element to represent the long action feedback. Additionally, we included two new elements to allow the adequately representation of warnings: *confirmation warning* and *notification warning*. Fig. 7 illustrates the elements of the refined USINN notation.

The elements that were changed or included are described below:

Presentation unit: it is the basis of the navigational structure of the system, restricting the interaction that a user can perform at the interface in a given moment.

Presentation unit (Always accessible): it is a presentation unit that must always be accessible during the interaction.

User transition: it is a dialog where the user chooses how to proceed with the interaction between a set of actions provided by the system. The transition may or may not have a description, which indicates that the user also receives feedback from the system when choosing how to proceed with the interaction.

Navigation elements		Interaction elements		Usability elements		
Presentation unit	Opening point	User action	User transition	Presentation unit (always accessible)	Optional user action	Warning
Navigation	Closing point	System process	System feedback	Cancel transition	Data collection and Query	System process with progress feedback

Fig 7. USINN elements to represent interaction, navigation and usability - new version after feasibility study.

System process: it represents an internal system processing after a request from the user. It may or may not inform the user about its progress. If it is necessary to provide progress feedback, the process displays a progress indicator and the element is represented differently, as Fig. 7 illustrates.

During a system process, the user can: (a) cancel a request through the cancel transition element; (b) wait for the system to complete the process and to direct to the next presentation unit; or user action through the system feedback element.

System feedback: it is a response provided by the system to a user request. If a request is successfully executed, the arrow is continuous. Otherwise, a dashed arrow with a description informs the problem that occurred.

Confirmation warning: it represents a warning that the system may display after user transition that must be confirmed

by the user. After a confirmation warning, the user can either confirm or cancel the request.

Notification warning: it represents a warning that the system can display during the user-system interaction, due to certain conditions. The notification warning will not interrupt the user interaction, neither will require a response from the user. However, the notification warning must be used to present useful information that will influence the user interaction.

Fig. 8 illustrates a new USINN diagram for the AcadSupport system, represented with the refined notation. In the “access AcadSupport” presentation unit, after the “access the account” user transition, the system displays a progress feedback during the user data authentication, during which the user can cancel the operation.

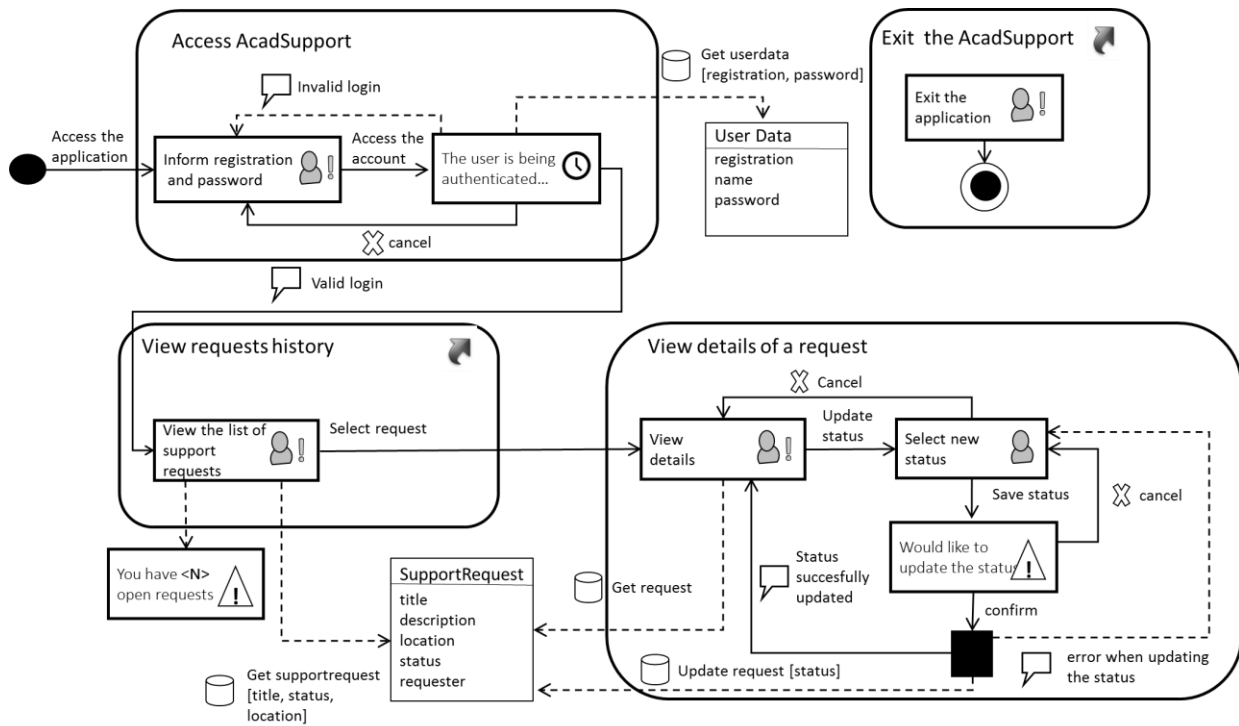


Fig 8. Example of a USINN diagram - after the refinement of the notation.

When the user views the list of support requests, the system can send a *notification warning* to the user, indicating that there are new requests. In the “view details of a request” *presentation unit*, after the “save status” *user transition*, the system displays a *confirmation warning* for the user to confirm the status update before the system processes the operation. After processing the new status, the system can send a *system feedback* confirming the status update (“status successfully updated”) or informing about an operation error (“error when updating the status”).

B. Development of the USINN Metamodel

In order to support the theoretical basis of USINN, we created the USINN metamodel. To represent the metamodel, we used a UML class diagram. The metamodel aims to represent the USINN concepts (classes and attributes) and relationships. Fig. 9 illustrates the USINN metamodel. We describe the main concepts and relationships next.

The *presentation unit* is the basis for the navigational structure of the system. The *presentation unit* enables/disables *user actions* over the interface. Presentation units are associated through the *navigation* element.

The *navigation* allows the user to navigate to the different *presentation units* during the interaction. Also, if there is a *navigation* between two *presentation units*, it indicates that the user can navigate between the two units during interaction.

The *user actions* are dialogs in which: (i) the user provides domain information to the system in order to achieve interaction goals; or (ii) the user views domain information presented by the system. The *user action* has an attribute that indicates whether the action is required or not to achieve an interaction goal.

The *user transition* is a dialog in which the user chooses how to proceed with the interaction among a set of actions

provided by the system. In these transitions, the user can request domain information from the system. In these cases, the *user transition* must have a description indicating what the user is requesting from the system. The *user transitions* are enabled according to the *user actions*.

The *system process* represents an internal system processing after a user transition. Some *user transitions* require a *system process*, while others do not. When the user chooses what to do from a set of possible actions, the system may need to validate the user’s choice, analyzing whether the user can proceed with the intended interaction flow. The *system process* has an attribute indicating whether the progress feedback is necessary or not. In the case of requests that may take some time to be completed by the system, it is important to inform the user so (s)he can be aware of the request progress. This decision is made by the interaction designer. When the system process has a progress indicator, a *cancel transition* is enabled while the operation is not completed.

The *system process* can enable *user actions* or change the current *presentation unit*. In some cases, the *system process* does not change the *presentation unit* and only enables *user actions*. In other cases, the *system process* may direct the user to a *presentation unit* different from the one (s)he is currently in.

The *data collection* contains the data presented and used during the interaction, by either *user actions* or *system processes*.

The *notification warning* may be necessary for informing users during or before the execution of actions in the system. The *notification warning* will not interrupt the user interaction, nor require a response from the user. However, the *notification warning* must be used to present useful information to the user, which can influence the user interaction.

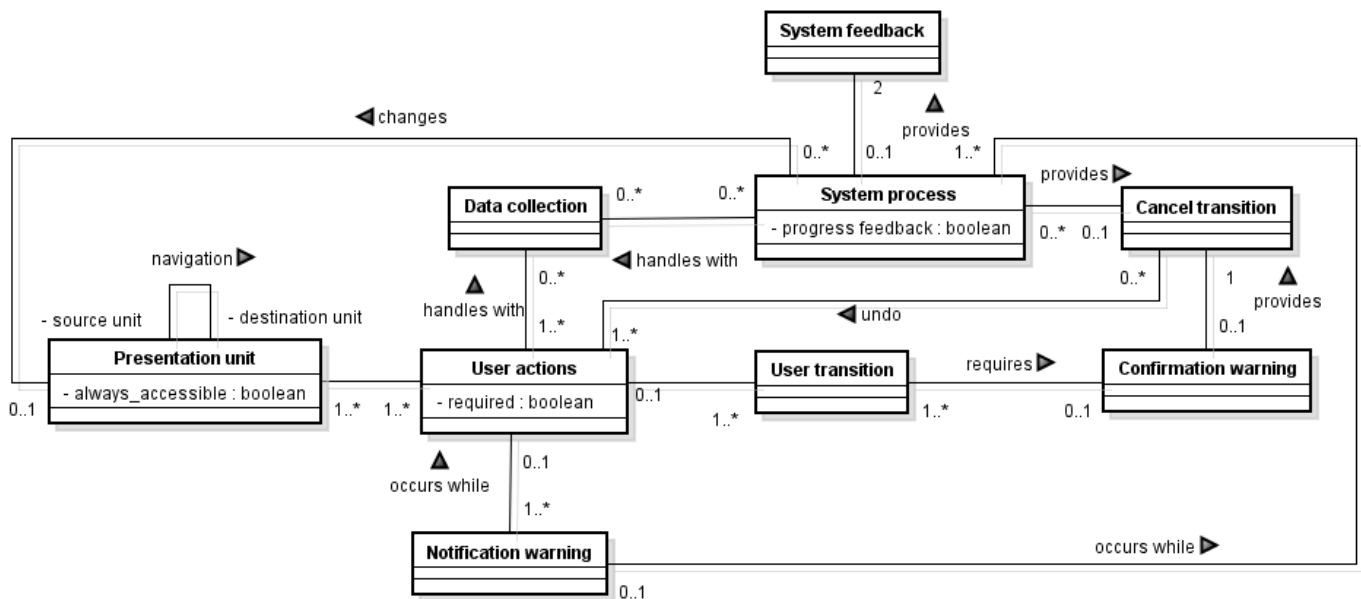


Fig 9. USINN Metamodel.

The *confirmation warning* may be necessary after a *user transition*. It consists of a notification to inform users about any action with important consequences. The *confirmation warning* enables a *cancel transition*, which allows the user to cancel the last request, and a *user transition*, which allows the user to confirm the last request.

The development of the USINN metamodel allowed us to describe the semantics and syntax of the proposed notation. Defining the meaning of each element and the possible associations between them helps to define the syntax rules of the notation.

The metamodel can assist practitioners willing to adopt the model in understanding and using it. In addition, the metamodel can guide the development of a support tool for creating USINN diagrams.

VIII. CONCLUDING REMARKS AND FUTURE WORK

In this paper, we presented the USINN model, which is intended to model the interaction and navigation of interactive systems oriented to usability. The paper also reports a feasibility study of the model with three participants, with different degrees of experience with the use of models in the industry, with conducting research about models, and with teaching models.

The quantitative results provide indications that USINN was perceived as useful, easy to use, and effective. However, one participant indicated disagreement with regards to some statements that evaluated these aspects.

The participants indicated that USINN allows the clear understanding of the user and system actions, USINN may assist developers during implementation and USINN already helps to design usability. On the other hand, through the qualitative analysis, we identified improvements to be made in USINN, such as the need for inclusion and adaptation of elements to allow the effective representation of usability mechanisms. Considering the comments from the participants, we have refined the USINN notation and we decided to develop the USINN metamodel to represent its theoretical basis.

The metamodel enabled us to strengthen the conceptual basis of USINN and to define the possible relationships between its elements. Thus, the rules regarding the creation of USINN diagrams were described more clearly and the metamodel may allow us to get better results about the perceived ease of use.

To better investigate whether USINN supports the usability mechanisms representation, we intend to conduct a second feasibility study, in which the scenario used as basis for modeling tasks will describe functional usability requirements. In this study, we will use the objective and subjective metrics defined in this first feasibility study, in order to verify the impact of the improvements carried out in USINN. As future work, we also intend to conduct comparative studies of USINN with existing interaction and navigation models, as well as to investigate the adoption of USINN in the software industry

With this paper, we intend to contribute to the improvement of the quality of use of interactive systems, since including functional usability features at the design stages can provide benefits. In addition, USINN can help designers to think on different solutions that provide a positive user experience while using interactive systems.

ACKNOWLEDGMENT

The authors acknowledge the financial support granted for this research by CAPES process 175956/2013 and CNPq processes: 309828/2015-5 and 423149/2016-4.

REFERENCES

- [1] A. Abran, A. Khelifi, W. Suryn, and A. Seffah, "Usability meanings and interpretations in ISO standards," *Software Qual. J.*, vol. 11, pp. 325-338, November 2003.
- [2] N. Juristo, A.M. Moreno, and M-I. Sanchez-Segura, "Analysing the impact of usability on software design," *J. Syst. Software*, vol. 80, pp. 1506-1516, September 2007.
- [3] J.I. Panach, N. Aquino, and O. Pastor, "A proposal for modelling usability in a holistic MDD method," *Sci. Comput. Program*, vol. 86, pp. 74-88, June 2014.
- [4] L. Bass, and B.E. John, "Linking usability to software architecture patterns through general scenarios," *J. Syst. Software*, vol. 66, pp.187-197, June 2003.
- [5] L. Carvajal, A.M. Moreno, M-I.Sanchez-Segura, and A. Seffah, "Usability through Software Design," *IEEE T. Software Eng*, vol. 39, pp. 1582-1596, November 2013.
- [6] F.D. Rodríguez, S.T. Acuña, and N. Juristo, "Design and programming patterns for implementing usability functionalities in web applications," *J. Syst. Software*, vol. 104, pp. 107-124, July 2015.
- [7] D. Costa, L. Nóbrega, and N.J. Nunes, "An MDA approach for generating web interfaces with UML ConcurTaskTrees and canonical abstract prototypes," *In Proc. of Tasks models and diagrams for users interface design (TAMODIA'06)*, pp. 137-152, October 2006.
- [8] J.I. Panach, N. Juristo, F. Valverde, and O. Pastor, "A framework to identify primitives that represent usability within Model-Driven Development methods," *Inform. Software Tech*, vol. 58, pp. 74-88, February 2015.
- [9] S.D.J. Barbosa, and M.G. de Paula, "Designing and Evaluating Interaction as Conversation: a Modeling Language based on Semiotic Engineering," *In Proc. of Int. Workshop on Design, Specification, and Verification of Interactive Systems (DSV-IS'2003)*, pp. 16-33, June 2003.
- [10] A. Lopes, A.B. Marques, S.D.J. Barbosa, and T. Conte, "Evaluating HCI Design with Interaction Modeling and Mockups: A Case Study," *In Proc. of International Conf. on Enterprise Information Systems (ICEIS 2015)*, pp. 79-87, April 2015.
- [11] F. Molina, and A. Toval, "Integrating usability requirements that can be evaluated in design time into Model Driven Engineering of Web Information Systems," *Advances in Engineering Software*, vol. 40, pp. 1306-1317, December 2009.
- [12] M. González, L. Cernuzzi, O. Pastor, "A navigational role-centric model oriented web approach MoWebA," *Int. J. Web Eng. Technol.*, vol. 11, pp. 29-67, April 2016.
- [13] B.S. Silva, O.A. Martins Netto, and S.D.J. Barbosa, "Promoting a Separation of Concerns via Closely-Related Interaction and Presentation Models," *In Proc. of El Congreso Latinoamericano de la Interacción Humano-Computadora (CLIHC'05)*, pp. 170-181, October 2005.
- [14] H. Trætterberg, "Integrating Dialog Modeling and Domain Modeling – the Case of Diamodl and the Eclipse Modeling Framework," *Journal of Universal Computer Science*, vol.14, pp. 3265-3278, November 2008.
- [15] M. Brambilla, A. Mauri, E. Umuhoza, "Extending the interaction flow modeling language (IFML) for model driven development of mobile

- applications front end,” *Int. Conf. on Mobile Web and Inf. Systems (MobiWIS 2014)*, pp. 176-191, August 2014.
- [16] M. Beaudouin-Lafon, “Designing interaction, not interfaces,” *Working conference on Advanced visual interfaces (AVI’04)*, pp.15-22, May 2004.
- [17] J.I. Panach, N. Condori-Fernández, F. Valverde, N. Aquino, and O. Pastor, “Towards an Early Usability Evaluation for Web Applications,” In *Software Process and Product Measurement*, J.J. Cuadrado-Gallego, R. Braungarten, R.R. Dumke, & A. Abran, 2008, pp. 32-45.
- [18] A.B. Marques, A. Lopes, A.C. Oran e T. Conte, “TR-USES-2015-001 Modelagem de Interação e Navegação de Sistemas Interativos: Protocolo de um Mapeamento Sistemático da Literatura,” *Relatório Técnico do Grupo de Usabilidade e Engenharia de Software (USES)*, 2015. Disponível em: <https://uses.induspam.com/relatorios-tecnicos/>
- [19] B. Kitchenham, S. Chartes, “Guidelines for performing systematic literature reviews in software engineering,” *EBSE Technical Report EBSE-2007-01*, Software Engineering Group Department of Computer Science Keele University, 2007.
- [20] T. Elwert, “Continuous and explicit dialogue modelling,” In *Proc. of Conf. on Human Factors in Computing Systems (CHI’96)*, pp. 265-266, April 1996.
- [21] M. Du, and D. England, “Temporal Patterns for Complex Interaction Design,” In *Proc. of Int. Workshop on Design, Specification, and Verification (DSV-IS’01)*, pp. 114-127, June 2001.
- [22] C.J. Scogings, and C.H.E. Phillips, “Linking tasks, dialogue and GUI design: A method involving UML and Lean Cuisine+,” *Interact. Comput*, vol. 14, pp. 69-86, December 2001.
- [23] R.R. Penner, and E.S. Steinmetz, “Model-Based Automation of the Design of User Interfaces to Digital Control Systems,” *IEEE T. Syst. Man Cyb*, vol. 32, pp. 41-49, August 2002.
- [24] H. Kim, and W.C. Yoon, “Supporting the cognitive process of user interface design with reusable design cases,” *Int. J. Hum-Comput St*, vol. 62, pp. 457 – 486, April 2005.
- [25] P. Avgeriou, and S. Retalis, “CRITON: A hypermedia design tool,” *Multimed. Tools Appl*, vol. 27, pp. 5-21, September 2005.
- [26] V. López-Jaquero, and F. Montero, “Comprehensive Task and Dialog Modelling,” In *Proc. of Int. Conf. on Human-Computer Interaction (HCI’07)*, pp. 1149-1158, July 2007.
- [27] S. Kavaldjian, “A model-driven approach to generating user interfaces,” In *Proc. of Joint Meeting on European Soft. Engin. Conf. and the ACM SIGSOFT Symp. on the Found. of Soft. Engin. Companion Papers (ESEC-FSE Companion’07)*, pp. 603-606, September 2007.
- [28] X. Lu, J. Wan, and J. Hou, “A Model Based Heuristic Design of Web User Interface,” In *Proc. of Int. Conf. on Information Reuse and Integration (IRI’07)*, pp. 6-11, August 2007.
- [29] H. Trætteberg, “Integrating Dialog Modeling and Domain Modeling – the Case of Diamodl and the Eclipse Modeling Framework,” *J. Univers. Comput. Sci*, vol. 14, pp. 3265-3278, November 2008.
- [30] A.I. Molina, M.Á. Redondo, and M. Ortega, “A methodological approach for user interface development of collaborative applications: A case study,” *Sci. Comput. Program*, vol. 74, pp. 754-776, July 2009.
- [31] E. Saleh, A. Kamel, and A. Fahmy, “A Model Driven Engineering Design Approach for Developing Multi-Platform User Interfaces,” *WSEAS Transactions On Computers*, vol. 9, pp. 536-545, May 2010.
- [32] N. Juristo, A.M. Moreno, and M-I. Sanchez-Segura, “Guidelines for Eliciting Usability Functionalities,” *IEEE T. Software Eng*, vol. 33, pp. 744-758, November 2007.
- [33] A.B. Marques, S.D.J. Barbosa, and T. Conte, “A Comparative Evaluation of Interaction Models for the Design of Interactive Systems,” In *Proc. of Symp. on Applied Computing (SAC 2016)*, pp. 173-180, April 2016.
- [34] C. Souza, “The Semiotic Engineering of Human-Computer Interaction”, The MIT Press, 2005
- [35] A. Strauss, and J. Corbin, “Basics of Qualitative Research: Techniques and Procedures for Developing Grounded Theory,” SAGE Publications, 1998.
- [36] F. Shull, J. Carver, and G.H. Travassos, “An empirical methodology for introducing software processes,” In *Proc. of the European Soft. Engin. Conf. held jointly with ACM SIGSOFT Int. Symp. on Found. of Soft. Engin. (ESEC/FSE-9)*, pp. 288-296, September 2001.
- [37] M.F. Granda, N. Condori-Fernández, T.E.J. Vos, and O. Pastor, “What do we know about the defect types detected in conceptual models?,” In *Proc. of the Int. Conf. on Research Challenges in Information Science (RCIS 2015)*, pp. 88-99, June 2015.
- [38] O. Laitenberger, and H.M. Dreyer, “Evaluating the usefulness and the ease of use of a Web-based inspection data collection tool,” In *Proc. of the Int. Symposium on Software Metrics (METRIC’ 1998)*, pp. 122-132, March 1998.
- [39] F.D. Davis, “Perceived Usefulness, Perceived Ease of Use, and User Acceptance of Information Technology”, *MIS Quarterly*, vol. 13, pp. 319-340, Septemeber 1989.