Personas and Multiple Layers in Interface Patterns

Plinio Thomaz Aquino Junior, Lucia Vilela Leite Filgueiras
Escola Politécnica da Universidade de São Paulo,
Departamento de Engenharia de Computação e Sistemas Digitais
São Paulo, SP, Brasil, 05508-900
{plinio.aquino, lucia.filgueiras}@poli.usp.br

Abstract
The accommodation of the diversity of users’ profile in the interfaces design of systems is a problem in the designers’ activities. This issue causes problems in the systems interface. The user finds barriers during the use of interfaces that were not developed for their profile. This work presents a solution for the accommodation of the diversity. The interface patterns are created with multiple layers and they are organized according to the user’s profile. The interface patterns support the designer on the search for solutions, because they represent a recurring problem and an abstract solution to the problem, in a way that this same solution may be used in many different levels of the same problem. The patterns have a multiple layer structure to allow the accommodation of the user’s diversity. The layers are grouped by personas as a way of typify the user. The IPPL processes are defined to create unique procedures that are followed by the interface designers that use the concept searching for the solution of their problems with accommodation of the users’ profiles diversity.

Keywords: Human-computer interaction, interface patterns, user profile, universal usability, multi-layer interface.

Resumo
A acomodação da diversidade de perfil de usuários no projeto de interface de sistemas é um problema existente nas atividades do projetista. Essa dificuldade causa problemas na criação da interface de sistemas e, por este motivo, o usuário encontra barreiras ao utilizar as interfaces que não foram produzidas para o seu perfil. Este trabalho apresenta uma solução para a acomodação da diversidade. Os padrões de interface são criados em camadas e organizados de acordo com o perfil do usuário. Os padrões de interface apóiam o projetista no direcionamento de soluções, pois representam um problema recorrente e uma solução abstrata para o problema, de tal modo que esta solução pode ser aplicada em várias instâncias diferentes do mesmo problema. Os padrões possuem estrutura em camadas múltiplas para possibilitar a acomodação da diversidade de usuários. As camadas são indexadas por personas como meio de caracterização do usuário. Os processos IPPL são definidos neste trabalho para apoiar o projetista de interfaces no uso do conceito, na busca de soluções para seus problemas na acomodação da diversidade dos perfis de usuário.

Palavras-Chaves: Interação humano computador, padrões de interface, perfil de usuário, usabilidade universal, interface em camadas.
1. Introduction

This research studies the users’ accommodation problem in the interactive system. Several problems can be found in the greater part of interfaces that are daily used. Many times the users find themselves using interfaces that do not fulfill their needs and aren’t able to solve the problem that had motivated them to look for the execution of the task in the first place. When performing their tasks the users might get confused while reading texts, insecure about the given data and not confident whether they have executed the correct steps.

The difficulties that users find during the interfaces use are caused by systems that are not properly projected for these users. In this case the interactive systems are created with inadequate language, complex steps sequence, no user’s feedback, among other problems.

These problems happen when the interface designer doesn’t exactly know the user who will use the system that is being developed. A common method is the definition of a generic profile according to the designer’s preliminary concept of the user. This generic profile supplies a generalized interface design in relation to the direction of the solution. When the interface designer is able to define a users’ profile with closer precision, new problems such as the unawareness of the representativeness of that profile, how many users’ types and which are the variables of the profile that change throughout time appear. The user’s accommodation problem happens due to the users’ diversity that interacts independently in the computational systems. The users’ diversity is directly linked to the profile differences. The user profile is the information that helps to typify the users in relation to their experiences and behaviors.

This research’s purpose is to define a possible way to create a design able to deal with the user variety into an interactive system project. This feature is based on the creation of an interface pattern group with solution in layers, organized according to the user’s profile represented by an archetype. As its secondary objective, it wants to provide a process that supplies interfaces developed according to several types of system target-users. This process allows the creation and maintenance of an interface patterns group with solutions in layers and organized by the user’s profiles.

The interfaces are created in conformity to the identified profiles and the designers use interface patterns to increase the quality of their products, in other words, it becomes possible to create standardized interfaces that assist the different characteristics of a people group.

The application in the electronic government's case illustrates that the proposed process solves the problem of the accommodation of a large user’s diversity in an interface design. The electronic government's services request the acceptance of a large user’s number, becoming a subject that motivates the application of the studies of this work.

The process should reach all users types and several technologies types: from users with little computational experience to experienced users, from computers screens to mobile interfaces, from computers with low processing capacity and antique software’s to last generation equipments. This goal is strongly related to the design patterns definition, assuring that there is abstraction in the definitions of the interaction patterns in the intention of usage in several domains.

The concept presented in this work is defined as Interface Patterns in Personas Layers (IPPL) that support the creation of interfaces in layers with the accommodation of the user’s diversity. The section 1 introduces the subjects related with this research, demonstrating previous researches and origin of the involved concepts. The section 2 describes the IPPL since the creation until the application in real cases, considering an adaptative web site, in the application process. The section 3 presents the research results and conclusions.

1.1. User’s Profiles Represented by Personas

Knowing the user is an extremely important aspect to reach the usability of a product or service. This can be done in several ways: research with the user, data collected in the usability test, interviews, inspections, among other techniques. The collected information about the user should be used by the designers in an efficient and practical way. The utility of the information about the user can be amplified with the use of a method of the user's representation, the personas [4].

Personas are fictitious illustrations made to represent the behaviors and motivations of a real users group, representing them during the development design process [7]. A persona is the representation of users as an instrument to help making project decisions, to make easier the communication with the designers and to select project rules adapted to the real proposal. The persona composition can be based on imagined information, demographical and biographical attribute of the represented personality. The personas have names as they were real people and they can be represented with an image, by a picture to add realism.
Cooper and Reimann [8] defined the persona technique as part of the concept of project driven by goals. The personas are traditionally created from user's researches, as for instance, interviews, contextual verification and other qualitative methods. The necessary information to create the base of the personas can also be collected among people from a team or corporation. The personas can be improved with truthful details or imaginary histories [13].

During the development of an interface design the personas can represent a real person, motivating the emphasis in the user, especially among the designers focused in technology. On the other hand, precaution is a must so that the user's representation with personas doesn’t become very realistic, because the personas tend to motivate the excess of the real projection and the concrete thought, excluding the abstraction.

The personas are used in this work as a way to direct the solution suggested by an interface pattern to the correct user type. This way, the personas behave as solutions application layers.

### 1.2. Interface Patterns

The patterns concept used nowadays in the computer’s area started with the theory of Christopher Alexander, architect, which was subsequently published as the book “Notices on the Synthesis of Form” [2]. Later, in “A Pattern Language” [3] the author supplies the patterns definition and presents detailed patterns collection as a dominant language of the urban architecture. The idea appeared in the constructions architecture however many areas use this concept, such as the Software Engineering (SE) and the Human-Computer Interaction (HCI).

A pattern should have a solution for a recurring problem in a project in a generic format, easy to understand and readable. The patterns have been proving to be a satisfactory way to communicate design experiences and the values of their authors' projects. The patterns can be used as a learning tool for professionals with little experience. They can be used for a general reading to help in the creation of ideas or a specific reading in the search of a solution for a specific problem. This way, the use of the patterns in the development activities of systems projects can be used as way of communication and discussion of ideas and solutions. Patterns are not ready components; they are not simple rules or heuristics. The patterns are abstract solutions nucleus for problems that appeal in different contexts, but they find the same forces at all moments, and the current implementation of a solution varies with each application.

The use of patterns in HCI happened as a natural sequence application of the patterns in Software Engineering, although there are differences among them [5]. The patterns have specialized vocabulary and structure for each private application field. It is verified that there isn’t a conformity in the HCI community of which are the representative patterns of the area, the format that should be followed to be presented, which is the purpose of each patterns group is and how they should be used [6]. As for instance, Tidwell [16] defines patterns as descriptions for the attempt of reaching good solutions for a common project problem according to a context, describing the constant and common characteristics in all the solutions. Borchers defends that design patterns detain a solution in evidence for a recurring project problem, using an easy to understand format [5].

Researchers of the area recognized the need of a unified structure for HCI patterns and specified the first PLML (Pattern Language Markup Language) version to ease the implantation of a standardized format [12]. PLML is the first language used to unify the patterns structure. The language is composed using a DTD (Document Type Definition) in common XML. PLML can be used to make patterns more solid in the terminology, favoring a larger organization and relationship success, besides allowing the errors recognition and contradictions in a pattern language.

The interface patterns are used in this work to formalize the presentation of solutions for recurring problems. The solution is organized in layers according to the identified personas.

### 1.3. Multiple Layer Interface

The multiple layers interface design comprehends the development of a group of interfaces, each one with a characteristics predetermined group with growing complexity in their functionalities [15].

The multiple layers concept was defined by Fekete [9] as the way to use several layers to take apart the graphic entities involved in the objects visualization that are manipulated in the interface. Later Shneiderman [14] defined the multiple layers concept as an approach structured in levels. A project structured in levels includes two or more interfaces, each one containing a pre-determined functions group. The users have the option of selecting an interface, but they don’t select which functions appear in the interface [14].

The interface design of a multiple layer application concentrates a limited group of basic characteristics and simplified manipulation on the first layer. Inside this concept, the system can have a limited group of layers that are increased together with the complexity of new functionalities and interface features. It is a possible to users of complex
systems to develop their knowledge as they manipulate the system and in their own rhythm. The users have the autonomy of working with the new characteristics layers when they have time or need new features to execute their tasks. The creator defends that such projects make possible for the inexperienced users to improve their capacities according to the personal rhythm of learning, while the experienced users can choose the use of basic functionality layers to make easier the use of more frequent operations [15].

The interfaces design with multiple layers promotes the universal usability, because it lets users with several types of knowledge and computational experience achieve success in the use of the interface. Shneiderman [15] encourages the increase of researches and the development of interface designs in multiple layers giving the user the control of the group of available characteristics in a certain use state of the system.

The interface layers justify the application of the patterns organized for personas in the construction of interfaces customized by user’s types. The interfaces organization by users makes applications layers with interfaces objects addressed by profile.

2. Interface Patterns in Personas Layer

The proposal of this work tries to solve the problem of the users’ accommodation in the interactive design, based on the problems created by the users’ diversity and the lack of a process that helps the interface designer during the development of systems that assist the universal usability requirements.

This work considers that the patterns are useful for the documentation of solutions about constant interface problems. The projects that use interface patterns reach better usability results. The projects that should assist a diversity of users usually don’t reach good usability results, however the projects that are focused on the user’s profile usually present an appropriate usability rate. The interface in layers gathers the appropriate functionality to a usage profile represented by personas, as the persona is an appropriate manner to represent the users’ profiles that are involved in an interface design [4], [10] and [11].

The project feature is composed by the definition of the essential IPPL processes, composed by the creation process (item 2.2), the consumption process (item 2.3) and the use and maintenance process (item 2.4). The consumption processes, use and maintenance are presented as a way of validating the concept.

The International Standard ISO/IEC 12207 and NBR ISO/IEC 12207 Software Life Cycle Processes [1] regulation establishes as its fundamental processes, the development process (activities of the developer), the operation process (activities of the user, including software operation and operational support to the users) and the maintenance process (activities of the responsible for the maintenance).

The sequence of processes defined in this project follows the definition of the Software Life Cycle Processes regulation [1] with three related phases. The first is the development process of IPPLs, defined in this work as creation process. The creation process is based on the development process of the Software Life Cycle Processes. The second is the operation process of the concept in the interfaces construction in layers, defined in this work as consumption process. The third is the maintenance process of the layers’ characterization with the users’ data, defined in this work as the use and maintenance process.

The objective of the IPPLs essential process is to establish a common structure for all phases of creation, use and maintenance of the patterns, with a well defined terminology for the implantation in diversified types of interface designs. The structure has processes, activities and tasks that can be applied during the acquisition of a IPPLs base, regardless of the application field and the technology restrictions involved. The goal of the essential process for IPPLs involves from the acquisition of data to the maintenance of the information with the profile evolution, behavior, among other elements of the definition structure.

2.1. IPPL Framework

The IPPLs are interface patterns that have a presentation structure organized in layers, and these layers are built and classified according to the personas. The IPPLs main structure is based on the elements that compose the PLML. For the analysis of the viability of the proposal, only basic elements to define the IPPL structure were used, which are shown in Table 1. All PLML elements can be inserted in the IPPL structure. The detailed structure in Table 1 lets the insertion of several layers (named with layer) of personas with the description of customized solutions of these users’ needs, besides the demonstration of an illustration of the application of the solution. The illustration can be enriched with examples of assisted pattern or non assisted pattern.
Table 1 – IPPL structure elements

<table>
<thead>
<tr>
<th>Element</th>
<th>Description of the structure element</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pattern id</td>
<td>Declaration of the identifying code for a quick reference to the pattern (the code syntax declaration may help the recognition of the pattern type and the number of layers). A single value for each pattern. A syntax where the code gives information on the pattern may be used.</td>
</tr>
<tr>
<td>Name</td>
<td>It’s the name defined for the pattern. It name should be a simple and relevant noun so as to remember the details of the pattern. It can’t cause doubts nor be redundant with other existing names in the active group of patterns. The name will be used as a communication method among de designers and also as a way to ease the exchange of information and the decision about the projects.</td>
</tr>
<tr>
<td>Problem</td>
<td>Describes pattern usage situation. Explains the problem that the pattern intends to solve.</td>
</tr>
<tr>
<td>Context</td>
<td>Presents the panorama that characterizes the situations in which the problem might happen and is the environment where the pattern can be successfully used. It is the list of conditions that must be reached before using the pattern. The context justifies the situation in which the problem occurs.</td>
</tr>
<tr>
<td>Forces</td>
<td>Presents the project tensions involved in the problem.</td>
</tr>
<tr>
<td>Layer - Persona</td>
<td>Layer identifier. It’s a name or image, defined for a type of user who belongs to the group of defined personas to represent the types of users in the system.</td>
</tr>
<tr>
<td>Layer - Solution</td>
<td>Describes the elements that compose the project, its relations, responsibilities and cooperation. The solution does not describe the implementation or the concrete project in particular, because a pattern is like a mold that might be used in different situations.</td>
</tr>
<tr>
<td>Layer - Illustration</td>
<td>Composed of representation that solves the problem according to the described scenery, based on the documented solution to the identified persona. The example must present illustration for two situations:</td>
</tr>
<tr>
<td></td>
<td>- Approved pattern: example of the interface implementation that solves the problem, in other words, example of pattern implementation according to the persona and the layer solution;</td>
</tr>
<tr>
<td></td>
<td>- Not approved pattern: example of implementation that does not meet the required pattern, in other words, examples contrary to the pattern, the persona and the layer solution.</td>
</tr>
</tbody>
</table>

The Figure 1 represents the example personas considered in the contextualization of concept. Considering the existence of two personas (Figure 1) identified as representatives of the type of user that will use the system in development, it becomes possible to build an IPPL based on an interface pattern.

Jack
“J’ve got little time to learn new things that are not related to my activity as a pharmacist”
42 years old
Free lancer
Income ranging from R$3,000.00 to R$ 5,000.00
Web navigation only to read e-mails and pay bills
Favorite software application: spreadsheets
Use of computers only at work
Low computational experience

Dave
“I like to have freedom and do things my way...”
26 years old
Post Graduation student
Income ranging from R$900.00 to R$ 1,800.00
Web navigation for most of his daily activities
Favorite software: games in general
Use of computers at home and at school
High computational experience

Figure 1 – Personas group identified to use the interface
Table 2 presents an example of Interface Pattern in Personas Layers used for to resolve problems in the Web, which demonstrates the composition of an IPPL with the interface pattern defined as “Map”.

<table>
<thead>
<tr>
<th>Pattern id: 001</th>
<th>Name: Map</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Problem</strong></td>
<td>The user has reached an objective, after surfing on many interface levels. The user doesn’t know “where” is or how reached that point in the interface. The user also doesn’t know how to come back (in the web surfing case, with no use of the “back” key).</td>
</tr>
<tr>
<td><strong>Context</strong></td>
<td>The non-specialist user needs to execute an occasional task to reach your target, consisted of surfing in different interface levels. The user finds what wants, but does not know how to go back to the original category.</td>
</tr>
<tr>
<td><strong>Forces</strong></td>
<td>The user has an objective, but maybe is not familiar with or interested in the paths to find a necessary task. To reach a certain level, the user needs to stay in contact with the previous level (for example, necessary documents to perform the task), but the user gives up surfing that way. To reach the objective, several steps will be taken, but the exact number of necessary steps may vary, due to the decisions taken in previous steps.</td>
</tr>
</tbody>
</table>

**Layer – Persona Jack**

According to the choices of the user, only the path to be used should be shown. Referring to the current location the user can only see where he is and go back to the levels themselves wants. The user can see the group of levels their chose to get to the current level. It is possible to go back two or more levels through the navigation map.

**Layer – Illustration Persona Jack**

If the designer identifies that the interface to be implemented will be used by users with behaviour recognized as the persona “Jack”, it should use the identified pattern as “001” (Pattern id attribute) and name “Map” (name attribute) to solve the support problem for the user in relation to his position in the structure of the interface. As the persona “Jack” has low computational experience and low learning interest, the pattern recommends that the interface demonstrates the path used by the user with the possibility of returning only through the same path. On the other hand if the identified persona was “Dave”, the pattern recommends the exhibition of all of the navigation possibilities emphasizing the path chosen by the user.
This recommendation is based on the computational experience, with the need of navigation freedom of this user type. It is verified that the same problems exist for all types of users; however the solutions can be adapted according to the profile and the users’ behaviour. In Table 2, the red arrows indicate the user access possibilities during the navigation.

The creation and maintenance of the IPPLs group requires the existence of an interface application type (web, critical systems, office tools, among others), and a personas group that represent the users types involved in this application type. The IPPLs group can contribute to several interface types, when it is defined in a generic personas group, in other words, representative personas of the whole possible population. These two requirements collaborate with the characterization of several elements of the structure of the pattern presentation.

It is verified in the context of this project that the interface patterns should be built and electronically published to use the hyperlinks and multimedia features, obecting the easiness of the search, the visualization and the maintenance of examples. The definition of the IPPLs group should follow the steps determined by the creation process (item 2.2).

2.2 Creation Process

The creation process is a sequence of ordered activities with the objecting to reach the creation goal of a IPPLs group, based on data that characterize the system’s target users group. The creation process defines the acquisition activities to be performed by the interface development designers. The creation process purpose is to obtain a patterns and personas group that identifies solutions for problems addressed to the users’ profiles of the target product or the products types that are developed by the team. The process begins with the identification of an interface patterns group (creation of new patterns, usage of patterns already published, etc.) and it finishes with the publication of the IPPLs group for the project and development team.

This process requires as an entrance the users’ data with information that allow the behavior recognition, experience and ethnographic data. The beginning of the cycle should contain the definition of the storage base and the policies of constant updating of the users’ data. The users’ data should be composed by attributes that characterize the user’s profile under the socioeconomic aspects and the use of the interface. These data can be originated in a data repository, log file, database, with attributes about the users of the target system.

![Figure 2 – IPPL Creation Process](image)

The user can be characterized by registration of personal data, questionnaires directed to the computational experience and ethnographic data, and later, after the use of the interface, can be characterized by the actions during the navigation and experience declared by the user and observer.

The information acquisition contained in the data repository is used for the generation of knowledge about the product target population, demonstrating difficulties, behaviors, tasks and associated profiles.

With this identified knowledge it becomes possible to mold representative characters of a large part of the target population. The characters are modeled according to the personas concepts, which are used by the usability specialists in the creation of interface patterns in layers.

Traditionally an interface pattern presents the solution for recurrent problems; however the proposal of the patterns in layers is to add solutions focused on the types of the profile of identified personas. With that, the designer possesses features to implement the interface, according to the personas that use that application type.

The interface patterns are defined according to the characteristics of users’ profile, being associated and classified in several layers.
Table 3 – Elements consumed and produced by the creation process

<table>
<thead>
<tr>
<th>Element Types</th>
<th>Elements</th>
</tr>
</thead>
<tbody>
<tr>
<td>External</td>
<td>Data with the behavior and ethnographic information</td>
</tr>
<tr>
<td></td>
<td>Interface Patterns Groups</td>
</tr>
<tr>
<td></td>
<td>Personas Groups</td>
</tr>
<tr>
<td>Final</td>
<td>Group of Interface Patterns in layers by each persona</td>
</tr>
</tbody>
</table>

The usability specialist is the actor of the creation process and carries out the following activities:

- Analysis of the problem through the type of activity to be performed by the users (the objective of feedback the information that are requirements for the definition of the user’s data collection);
- Definition of the procedure of user’s data collection and storage structure according to the focus information in the classification of personas;
- Analysis of the user’s data for the personas characterization;
- Selection or creation of the interface patterns group;
- Creation of personas (description and illustration);
- Identification or construction of IPPLs.

The users’ stored data are an essential feature for the creation process. The storage and the structuring of the data are necessary to allow the data acquisition. The data can be captured automatically through components of data collection or through the user’s observation.

2.3 Consumption Process

The consumption process organizes activities for the construction of the interface in layers, using the final element produced in the creation process, in other words, consuming the interface patterns in layers. The actor of that process is the web designer that should use the interface patterns to develop interaction solutions supplying paths and different features for differentiated types of users and represented by personas.

![Figure 3 – IPPL Consumption Process](image)

Table 4 – Elements used and produced by the consumption process

<table>
<thead>
<tr>
<th>Element Types</th>
<th>Elements</th>
</tr>
</thead>
<tbody>
<tr>
<td>External</td>
<td>Personas groups and interface patterns group in layers by personas</td>
</tr>
<tr>
<td>Intermediate</td>
<td>Does not exist</td>
</tr>
<tr>
<td>Final</td>
<td>Interface in Layers</td>
</tr>
</tbody>
</table>

8
The designer develops the following activities:

- Uses the of personas group to identify the target users of the interface;
- Uses the layer patterns to build an interface of the product;
- Creates interfaces in layers driven to the user’s profile (characterized by personas);

The multiple layer interfaces should have the maximum number of layers according to the formula below:

| Number of Interfaces in Multiple Layers | = Number of Personas identified for the service | + | 1 (one) Default Layer |

The default layer is used when it is not an identified and characterized persona, in other words, in some project situations, a persona can be defined as “Default Persona”, and therefore used on the default layer. The default layer should present characteristics of use of medium complexity. Starting from the default layer, the user can be identified by basic or advanced personas.

### 2.4 Use and Maintenance Process

This process defines the behavior activities of the system during the use of the interfaces by the users and the maintenance of the profile by the automated data collection. After the implementation of the interfaces produced in the consumption process, the application is available for the user's access. With the development of multiple layer interfaces, the user will always use the features and the content of a specific layer.

For web interfaces, if the service to be performed by the user is not characterized by a continuous contact with the site, the user's identification for a simple and single navigation in the content or execution of tasks is not necessary. In opposite situations, during the first access the user should supply information for characterization.

Even with the supply of data before the first use of the interface, the user will use an interface layer named default. During the use of the interface, the mechanisms of automated data collection will feed the repository of users’ data, with information about the accessed content and use strategy. Through the collected information the user can be qualified in one of the used personas to compose the application layers. If the user is not qualified among the modeled personas, that information is stored and it will continue to use the interface of the pattern layer.
On the following accesses the user can identified and use the interface layer of the persona that characterizes them, in case the adaptive system fits the profile of the user to a persona. Even with defined interface layer, the automatic data collection mechanisms, feed the repository of the users' data with information of the performance use. With the updating of the user's profile and used data it is possible that the system evaluates the alteration of the user's profile.

The use of the interface promotes the continuous updating of the users' data allowing the user's to fit in other personas. The usage also supports the maintenance of IPPLs, verifying the alterations in the characteristics that define the original personas, allowing the addition of new personas or removing personas that don't represent the most representative profiles (saving effort in the maintenance of the interfaces). The table 6 presents the actors and their respective roles in the level of use of the interface in layers of the application model.

Table 5 - Consumed and produced components by the use and maintenance process

<table>
<thead>
<tr>
<th>Components Type</th>
<th>Components</th>
</tr>
</thead>
<tbody>
<tr>
<td>External</td>
<td>Interface in Layers</td>
</tr>
<tr>
<td>Intermediate</td>
<td>User’s Identification Survey Data</td>
</tr>
<tr>
<td></td>
<td>Data of the Collected Usage Data</td>
</tr>
<tr>
<td>Final</td>
<td>Updating and maintenance of user’s data</td>
</tr>
</tbody>
</table>

Table 6 - Actors and roles in the use level of the interface in layers of the application model

<table>
<thead>
<tr>
<th>Actors</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>User</td>
<td>Uses the interface through the default layer;</td>
</tr>
<tr>
<td></td>
<td>Supplies information for questionnaire, form, interviews, that helps to characterize the user as a persona;</td>
</tr>
<tr>
<td></td>
<td>Uses the interface through the layer associated to a persona;</td>
</tr>
<tr>
<td></td>
<td>When accomplishing actions in the interface, supplies the usage information that are being collected</td>
</tr>
<tr>
<td>Adaptive Interface of the Site</td>
<td>Collects information declared by the users in the questionnaire and forms;</td>
</tr>
<tr>
<td></td>
<td>Collects the usage information based in the users’ actions (feeds the repository of the users’ data);</td>
</tr>
<tr>
<td></td>
<td>Qualifies a user recently registered in a persona;</td>
</tr>
<tr>
<td></td>
<td>Evaluates possible alterations in the user's profile according to the current persona;</td>
</tr>
<tr>
<td></td>
<td>Evaluates behavior alterations of the defined personas for the interface;</td>
</tr>
</tbody>
</table>

3. Conclusion

This paper presented the complete process for creation, use and application of Interface Patterns in Personas Layer. According to preliminary researches, IPPLs defined concept in this work enables the interfaces quality assurance in electronic government [10] and supports the interface adaptive development (Figure 4). The results were applied in an electronic government's context, because such systems should consider that all people are the same, without distinction of any nature, reaching a great diversity.

This work contributed with a solution for the interface profile diversity problems, assisting the market needs, due to the growing amount of people that have access to technology. The work defined a project feature that the designer interface can use to accommodate the users’ diversity. The processes of creation, application, use and maintenance were defined to create unique procedures that are followed by the designers that use the concept searching for the solution of their problems. The universal usability is possible by the use of personas that represent many types of users, when they are taken in consideration in interface designs. The interface design developed in layers with different levels of complexity content and actions becomes a more personalized method for certain types of users.

The application of this project feature brings benefits for both users and interface designers. The interface designers can reach the accommodation of the users’ diversity, keep the knowledge of the profile and the behavior of the system users, besides disseminating acquired knowledge for other projects and other team members. The users use interfaces especially designed for their profile and needs, increasing the usage satisfaction and their independence in the accomplishment of activities in computational systems.

The results have as a main goal to provide the improvement of life quality of interactive systems users, regardless of the profile type which they belong to, reaching the social inclusion with the support of the digital inclusion.
References


[7] Cooper, A. The inmates are running the asylum: Why high-tech products drive us crazy and how to restore the sanity. Indianapolis, Ind.: Sams, 1999.


