

Framework of User Participation for the Interface Design in the SaaS Layer

Zayra Jaramillo¹, Miguel Vargas-Lombardo^{1,2}

¹Technological University of Panama, Panama City, Panama; ²Research Group in e-Health and SuperComputer (GIses), Research Center for Development, Innovation of Technology Information and Communication (CIDITIC), Panama City, Panama.
Email: zayra.jaramillo@utp.ac.pa, miguel.vargas@utp.ac.pa

Received March 3rd, 2013; revised April 6th, 2013; accepted April 14th, 2013

Copyright © 2013 Zayra Jaramillo, Miguel Vargas-Lombardo. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

ABSTRACT

The development of software as a service represents a great advantage for the users of cloud computing, such as users of small, medium or large organizations. All these users possess different characteristics, which must be taken into account as reference for the interface development of a software product. This paper considers *the element of a user experience model*, with the purpose of getting to know the different features that must be taken into account during the development of software as a service. This also involves the participation of different types of users of the SaaS layer model together with the users' interaction skills level. This paper offers a framework of the involvement actors in the design of a software interface as a service, with the objective of identifying their participation outside and within the organization. The purpose is to create an interface centered on the user, involving clients and experts to work collaboratively; and get usable software as a service application.

Keywords: Cloud Computing; Interaction; Interface; SaaS; User Design; User Experience

1. Introduction

Cloud computing is defined as a large scale distributed computing, which is virtualized, dynamically scalable, managed computing power and the services are offered to external clients over the internet [1].

The information technologies are an essential part of any business and it is commonly accepted that technological solutions are needed. More and more software applications are supported by the cloud computing requiring greater effort to attract users to interact more and better with everyday environments that arise from different contexts. Therefore, it requires new approaches to satisfy the users [2]. Users must have an important role in the development of software products. Usable products can be designed by incorporating attributes known to benefit users in a particular context of use [3].

Companies are choosing this new cloud computing technology to offer their application services called software as a service (SaaS), which is why companies are experimenting with the use of multiple software applications in their departments, which have little or no control over the applications [4]. Also, different designs of

the SaaS layer are being created so it can be customized.

The SaaS layer offers its services in a software model of a web based application, and its objective is to serve a large quantity of users [5]. The SaaS layer can be used by users with several levels of experience and tasks.

The interface designs create a good communication and collaboration between users, which is why the design must be centered on the user. Interface designs use rules that include topics such as [6]: Industrial and educational psychology, graphic design, writing techniques, ergonomics, and sociology, among others. An effective interface lowers frustration, fear and user mistakes when he interacts with software. Therefore, if the characteristics of the users are known, product based on their necessities can be developed, improving organization productivity. The experience of the user with the software must be a cohesive, intuitive and pleasant [7].

In this paper, we proposed to take into account the users in the development of an interface for the SaaS layer. The design must be centered on the user, which is why we take into account the elements of the user experience model [7] because the users are the main purpose in the development of the different stages of the interface mo-

del.

In Section 2, general details of the SaaS layer are shown. In Section 3, the different types of users that participate and their importance are shown; in Section 4 the elements of user experience layer model will be described. In Section 5, a framework of the participants in the development of the interfaces of a SaaS product will be defined. Finally, in Section 6 we mention some conclusions.

2. Software Services in the Cloud

Cloud computing offers three layers of services, infrastructure as a service (IaaS), platform as a service (PaaS) and software as a service (SaaS) [1,8]. SaaS is the layer that immediately visualizes the user in the cloud, so it represents the user interface [8].

The SaaS layer has changed the way to store and access software applications, since they are already stored in remote servers and are accessed through the internet [9]. But, the traditional software applications are installed directly on the client's computer.

The SaaS layer is divided into two main categories, business oriented services and client oriented services. Business oriented services sell software to companies, through a subscription. These services are related to business process such as product or application management services for customer relationship management. The other type of SaaS service, client oriented services, offers software for the general public, this services are acquired through a subscription or free of charge [9].

Cloud computing [10] defines some characteristics based on the resources to offer SaaS. These characteristics are the following:

- Details of the resources are unknown for the users, for example: the hardware or software version.
- The lack of resources is not a reason for concern for the user.
- In some cases, payments are made for the use of the resources.

These resources are necessary so the user can utilize SaaS. SaaS layer is designed to offer services at any time or place, in which the SaaS provider uses necessary resources and provides the adequate maintenance [11].

SaaS has been successful, counting with several software applications such as video conference, customer relationship management, accounting, emailing, and human resources among others [12].

SaaS application work for clients that are interested on improving their applications integration system, therefore, SaaS applications can be used by several users [13,14].

3. Users

SaaS is a multi-user software model, which permits dif-

ferent clients, from a single user up to a group of users from a large company, to use the same software [13].

The user represents the main actor and any other user registered or unregistered in software as a service.

3.1. SaaS Layer Users

The SaaS layer offers applications that the client can interact with, the main types of SaaS users are [15]:

- *Organizational*: user that accesses business interest services and these services are administrated by a client user.
- *Client*: represents the client subscriber, he is the service administrator for a specific application, accesses and configures the software services. Also, the client assigns the user privileges, in a company scenario.
- *Visitors*: user that utilizes a given time to test software. Counts with access to some applications that are found in the SaaS layer. This type of users can register or authenticate themselves to be able to use cloud services.

Figure 1 shows SaaS users, these users are defined according to the tasks they can perform in the SaaS layer.

3.2. Interaction Skill Levels Users

The user is the main element for interface design; therefore it is necessary to know his level of interaction with interfaces and area of domain of an application. The users [6] can be classified as:

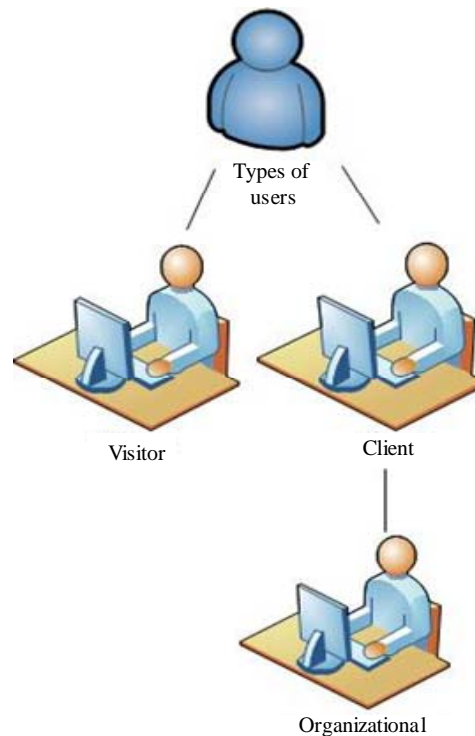


Figure 1. Types of SaaS layer users.

- *Novice*: knows little about the concepts of tasks and interfaces, or is a first-time user.
- *Knowledgeable intermittent*: represents the client subscriber, he is the service administrator for a specific application, access and configures the software services. Also, the client assigns the user privileges, in a company scenario
- *Expert frequent*: he is completely familiarized with the concepts of tasks and interfaces and is capable of modifying the software customization for a more efficient performance.

After analyzing the different types of users of the SaaS layer and interaction skill levels users, we can determine the user participation of SaaS application, as shown in **Table 1**. The *client user* in the SaaS layer must have a high level of interaction, because he is the representative who will buy software, so he must know if it is convenient to make the purchase. The *organizational user* is represented by the employee that utilizes the software, he can be novice, knowledgeable intermittent or expert frequent, and this depends on the user experience. The *visitor user* utilizes the software on a trial period or first time; also some visitors have experience with the use of software interface.

The user plays an important role in the designing of a software interface, so their main tasks help to know which elements are relevant for the interface design.

3.3. User Importance

The user experience and needs are taken into consideration for interface design by organizations, therefore, these characteristics increase the possibilities of a successful software product development [6,16]. A software developer carefully considers the business necessities and knows the specific requirements of the non-technical users of the organization. Therefore, the developers make their designs centered on the user [17]. A design centered on the user takes into account the user in each stage of the product software development [7].

If the user is taken into account, it is necessary to know their characteristics through user profiles. User profiles are defined as a set of characteristics that represent types or classes of users [6,18]. The set of character-

Table 1. Relationship between SaaS layer users and interaction skill level users.

Interaction skill level users	<i>Novice</i>	<i>Knowledgeable intermittent</i>	<i>Expert frequent</i>
SaaS layer users			
<i>Client</i>	X	√	√
<i>Organizational</i>	√	√	√
<i>Visitors</i>	√	√	X

istics is much related to the business objectives that will be implemented on the product software [18].

By the previous considerations, it is necessary to create user roles. The roles of the users describe their tasks, work description, responsibilities and other factors related to the functionalities of interaction with the interface [18]. As we can see, the user enjoys performing tasks when they can interact easier with the software interface.

4. The Elements of User Experience Layer Model

In [7], a layer model is defined as a guide for products or services development, based on user experience. These products or services represent the development of web sites, so, they can show information or manipulate certain functionalities. These functionalities can be electronic commerce, social network, and financial services among other areas that the users may require.

This model is completely developed on the user experience, so it takes into account the users actions and understanding the user's expectations in each step he makes during the development of a task.

The objective is to take into account what the user can do and understand in a web environment and also to create functional products.

The model is divided into product as functionality and product as information, at the same time, it has five planes based on a set of decisions about how the web application will look like, how it will behave and what it allows him to do.

4.1. Product Functionalities in the Elements of User Experience

The planes of the elements of user experience model allow us to know specific elements for the development of applications in the web. However, for the development of web based software, one must take into account specific elements such as functionality product, **Figure 2**.

The Elements of user experience model [7]. As shown in **Figure 2**. Functional products are tools or a set of tools to do one or more tasks. According to Garrett [7], the elements of user experience as functional products are focused on tasks. These tasks are processing in each plane and how they are done by developers. Next, the planes of the elements of user experience model will be explained and their main characteristics:

1) *Strategy Plane*: it is interested in the user needs and product objectives.

a) User needs: these are the specific objectives of people that use the interface; these objectives are found outside of the company.

b) Product Objectives: these are the business goals.

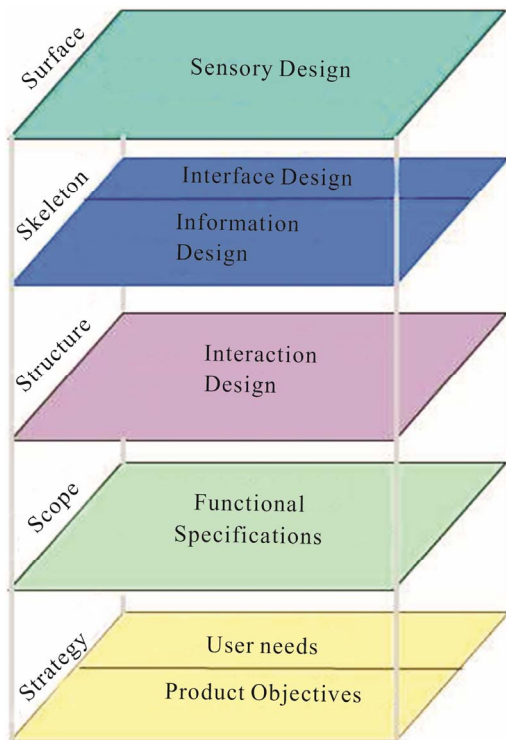


Figure 2. The Elements of user experience model [7].

They describe the internal strategic objectives of the company or organization.

2) Scope Plane: this plane creates functional specifications. Functional specifications describe the characteristics of a product.

The participants are the developers, designers of the software product, whose development is based on accomplishing the objectives of the previous layer. It is recommended to use the brainstorming exercise with the purpose of obtaining new possibilities for the project.

3) Structure Plane: Determines the user experience with the system. The system responds to the user is known as interaction design. The interaction design performs and completes the tasks through the behavior and thought of the users.

4) Skeleton Plane: takes into account the information design and the interface design.

a) Information design: it consists on how to present the information, in a way that the users may use and understand it with ease.

b) Interface design: selects the correct elements of an interface to perform a task, which must be easy to use for the user.

5) Surface Plane: Considers the sensory design, which determines the visual structure of the product. The sensory design is manifested in the sense of the user, being the sense of the sight the most important. This is because the user can visualize the software as a final product.

These planes model allows the development of prod-

ucts based on user experience, so the user is the main actor in software development as a product.

5. Framework of User Participation According to the Interaction Context

The framework of user participation according to the context of interaction is based on the main actors in the development of software interfaces as a service, helping developers and designers to take into account from the start. They are creating functional products for the people, which will allow them to have a pleasant experience while performing their tasks.

This framework shows the participation of users inside and outside of an organization, and an intermediate participation. As shown in Figure 3, users have a high level of participation in taking decisions of specific elements for the development of software. Also, developers and designers of software as a service will be there during the entire development of the product.

Below, we will be defining the users that participate in the development of software as a service, in which the elements of user experience planes model is taken into account, the types of users of the SaaS layer and the types of users of the different levels of interaction are described as follows:

- *User A*: represents the users within an organization or company. According to the framework, in Figure 3, the users have a high level of understanding and experience in the usage and knowledge of software. Also, these users participate in the definition of tasks and in the accomplishment of software product objectives. So, it will cover a large scope of tasks that must be considered for the software product.

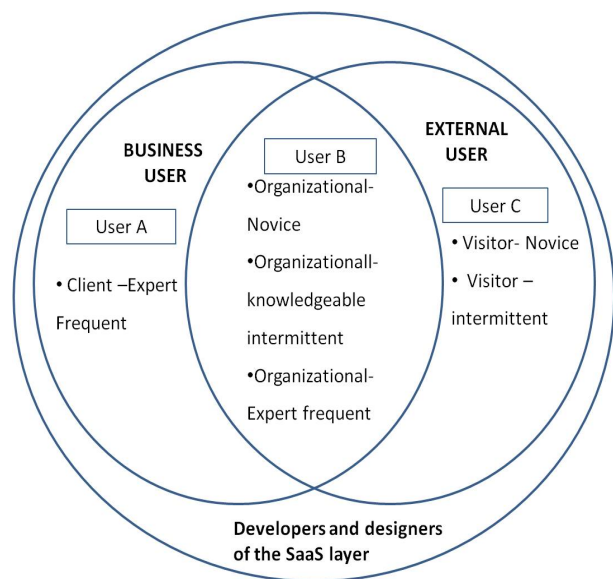


Figure 3. User participation in a development of a SaaS.

- *User B*: represents the users that can be in or out of the organization. Actors such as the organizational-novice, organizational-knowledgeable intermittent, organizational-expert frequent, novice-visitor and knowledgeable intermittent-visitor; contribute in the definition user needs for the software development.

To know these necessities the users must perform tasks under similar conditions to standardize the interactions while executing tasks [6]. These interactions can use support tools such as surveys or interviews which help collect information about users' aptitudes and perceptions. Also, the user evaluations and field studies, allow understanding specific aspects of the behavior and interaction of the user with the interface [7].

With the development of software as a service the functional characteristics of the software must be known, because the users must interact in an efficient with these functions. The actors involved are the organizational-novice and organizational knowledgeable intermittent. According to [7], to know the functional specifications of the product, the most experienced users are not considered in this step because they do not contribute with new ideas for the product design.

It's clear that knowing the necessities of the users, helps accomplish the objectives of the product with an effective communication to improve task performance.

- *User C*: represents the users that are out of the organization, since they do not know the objectives that are needed to reach the development of the Software as a Service and contribute with the necessary elements for an adequate interaction.

In [6], the designers take into considerations some elements of the interface such as:

- The utilization of signs, which help to understand where the user is and where he could go. Some examples of signs are icons, labels and typography.
- The feedback from the interaction of a novice user, showing errors messages when the interaction has not been satisfactory.
- To increase the users understanding, vocabulary information must be reduced to familiar and consistent concepts [19]. The users have to list in a very detailed manner the technical language that they use to perform their tasks [20].
- The elements such as dialogs and online help, prevent the knowledgeable intermittent user can execute his tasks in a satisfactory manner, if the interface has a sequence of consistent actions, significant messages and a guide patter of frequent use. Then, the expert frequent requires quick answers, brief feedbacks and shortcuts, to execute actions using a series of key

combinations from the keyboard or small selections.

As part of the framework, the users that participate in a SaaS application and their main tasks have been defined. This is the framework of the user participation according to the interaction context in a SaaS application, see **Figure 4**. First, we must clearly identify users who are involved in the development of SaaS applications and categorize them according to category A, B or C, so users can contribute to complete each layer. The first plane is Strategy; in the sub plane user needs it requires the participation of external users. Therefore, we can get the user requirements to contribute in the design of a SaaS application and user can achieve their tasks when the final product is complete. Then, the next sub plane is about product objectives, it has the participation of user within the company. In this sub plane, it is important to assess the business goals in order to know what users' needs will be taken into account. The second plane is Scope, it requires the participation of users A and B, because they define task that have developed in a SaaS application, so it is necessary users that are within and out of the company contribute to complete this layer. The third plane is Structure; it requires the participation from users out of the company, because the users give opinion about how to interact with the interface. The behavior of the interface will be assessed by the user if it responds a request satisfactorily. That is the reason, feedbacks help us to do improvements, so the final product will make user feel comfortable during the interaction. The fourth plane is Skeleton, which contains two sub planes that are important in the interface design. This plane requires the participation of user out (User C) out of the company. It is necessary to know certain elements from the user, for example language and writing directions; in order to develop a SaaS application properly. The last plane is Surface; this is the most important layer because with this

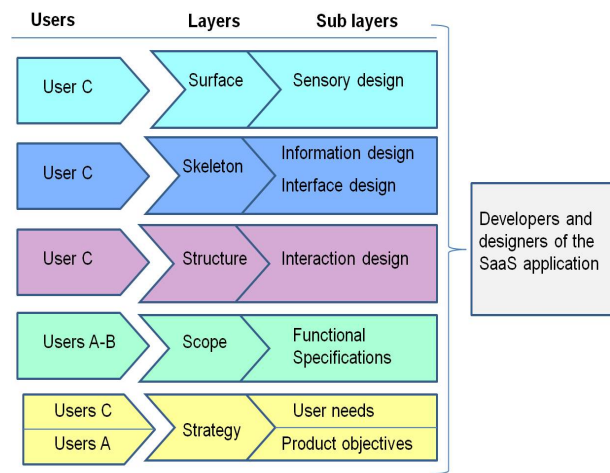


Figure 4. Framework of the user participation according to the interaction context in a SaaS application.

layer the interface design in the SaaS application is completed. It requires the participation of users out of the company who assess the final product. Thus, the evaluation gives different suggestions to improve the interface design. The SaaS application is expected to have a high scale of acceptance by users.

Developers and designers work collaboratively in the development of a software product, since the necessary information based on the user experience has been collected.

This framework permits and emphasizes on knowing what the user clearly requires, from an organization for the creation of SaaS application interfaces, as well as the user that requires working with the application and does not belong to the organization.

Besides, we can see that the organization must have a relationship with the users that work with the software product, in this manner they both benefit by providing a tool that covers the necessary functions for the optimum operation of the company and that the clients or users can perform their tasks in an efficient and enjoyable manner.

The clients are complicated, their individual needs and expectations can not be standardized, so the context and the situation of each client must be taken into account for the development of a product as a service [20,21].

According to [21], the design of services represent the perspective of the client, to design services it is important to establish a good understanding of the objectives, motivations and client needs. Besides, the design of services allows us to define the interactions of the clients with the system, which must be understood and used by the clients in an easy manner filling their expectations.

The knowledge of the detail of the elements, interactions and information that compose the interfaces makes them usable and easy to understand.

The study of this framework allows the participation of experts and clients in the design process of a SaaS product, with users of various levels of experience.

6. Conclusion

In this paper, a framework of user participation according to the interaction context in a SaaS application is proposed for designing usable SaaS application. This framework uses *the element of user experience model* that is used by many developers in the development of software applications because it recollects the user experience, and the framework shows the user participation in each layer of the model. In the identification of users, we search types of SaaS users and interactions skills level users. Therefore, we identify three types of users, internal (User A), external (User C) and internal/external (User B) of the company. Internal users determine the functional specifications of the application, the external users collaborate in the design and interactions of the SaaS appli-

cation's interfaces, and the internal/external users contribute in the design of some elements and limit the development of the project.

This framework is a guide for SaaS application developers and designers, to create usable products because it focuses on the user experience. Also, the framework takes into account the user needs and the functional goals of a software product.

It is very important to keep in mind that the user is the one who can contribute to make improvements in the development of a SaaS application. If people feel comfortable when they are interacting with an interface, that product is usable.

REFERENCES

- [1] Foster I., *et al.*, "Cloud Computing and Grid Computing 360-Degree Compared," *Grid Computing Environments Workshop*, Austin, 12-16 November 2008, pp. 1-10.
- [2] S. K. Garg, S. Versteeg and R. Buyya, "A Framework for Ranking of Cloud Computing Services," *Future Generation Computer Systems*, Vol. 29, No. 4, 2012, pp. 1012-1023.
- [3] N. Bevan, "International Standards for HCI," *Encyclopedia of Human Computer Interaction*, 2006, p. 362.
- [4] H. Hai and S. Sakoda, "SaaS and Integration Best Practices," *Fujitsu Scientific and Technical Journal*, Vol. 45, No. 3, 2009, pp. 257-264.
- [5] M. Hoogvliet, "Saas Interface Design," Bachelor Thesis, Rotterdam University, Rotterdam, 2008.
- [6] B. Shneiderman and C. Plaisant, "Designing the User Interface," Pearson Education India, Delhi, 2010.
- [7] J. J. Garrett, "The Elements of User Experience: User-Centered Design for the Web and Beyond," New Riders Pub, 2010.
- [8] C. Weinhardt, A. Anandasivam, B. Blau, N. Borissov, T. Meinel, W. Michalk and J. Stöber, "Cloud Computing—A Classification, Business Models, and Research Directions," *Business & Information Systems Engineering*, Vol. 1, No. 5, 2009, pp. 391-399.
[doi:10.1007/s12599-009-0071-2](https://doi.org/10.1007/s12599-009-0071-2)
- [9] I. Menken and G. Blokdijk, "SaaS and Web Applications Specialist Level Complete Certification Kit—Software as a Service Study Guide Book and Online Course," Emereo Pty Ltd., 2009.
- [10] F. Leymann, "Cloud Computing," *IT—Information Technology*, Vol. 53, No. 4, 2011, pp. 163-164.
- [11] G. Clair, "Software-as-a-Service (SaaS)," 2008.
- [12] T. Han and K. Sim, "An Ontology-Enhanced Cloud Service Discovery System," *Proceeding of the International Multi-Conference of Engineers and Computer Scientists*, Hong Kong, 17-19 March 2010.
- [13] G. Blokdijk, "SaaS 100 Success Secrets—How Companies Successfully Buy, Manage, Host and Deliver Software as a Service (SaaS)," Emereo Pty Ltd., Brisbane, 2008.

- [14] M. Turner, D. Budgen and P. Brereton, "Turning Software into a Service," *Computer*, Vol. 36, No. 10, 2003, pp. 38-44. [doi:10.1109/MC.2003.1236470](https://doi.org/10.1109/MC.2003.1236470)
- [15] J. Lopes, "SaaS (Software as a Service)-Models and Infra-Structures," 2009.
- [16] J. Nielsen, "Usability Engineering," Morgan Kaufmann, Burlington, 1994.
- [17] A. Cooper, R. Reimann and D. Cronin, "About Face 3: The Essentials of Interaction Design," Wiley-India, 2007.
- [18] J. Pruitt and T. Adlin, "The Persona Lifecycle: Keeping People in Mind throughout Product Design," Morgan Kaufmann, Burlington, 2006.
- [19] J. Johnson, "Designing with the Mind in Mind: Simple Guide to Understanding User Interface Design Rules," Morgan Kaufmann, Burlington, 2010.
- [20] A. Metzger, *et al.*, "Variability Modeling to Support Customization and Deployment of Multi-Tenant-Aware Software as a Service Applications," *Proceedings of the 2009 ICSE Workshop on Principles of Engineering Service Oriented Systems*, Washington, 2009, pp. 18-25.
- [21] S. Moritz, "Service Design: Practical Access to an Evolving Field," M.Sc. Thesis, KISD, Killeen, 2005.