

# A LifeLike Experience to Train User Requirements Elicitation Skills

Silvia De Ascaniis<sup>1</sup>(✉), Lorenzo Cantoni<sup>1</sup>, Erkki Sutinen<sup>2</sup>,  
and Robert Talling<sup>2</sup>

<sup>1</sup> USI – Università della Svizzera italiana, Lugano, Switzerland  
{silvia.de.ascaniis,lorenzo.cantoni}@usi.ch

<sup>2</sup> University of Turku, Turku, Finland  
{erkki.sutinen,robert.talling}@utu.fi

**Abstract.** User Requirements Elicitation (URE) is a critical stage in the development of software systems. It is aimed at defining the information needs the system has to fulfill and the services it is expected to provide. The term “elicitation” points out the delicate role of the analyst, who has to take an active listening attitude in the dialogue with system stakeholders and intended users, being able to seek, uncover and elaborate requirements. The success of the process largely depends on the analyst’s communication skills and expertise, since URE is communicative, interdisciplinary and practical in nature. Despite a variety of techniques and approaches to URE are proposed, there is not at the moment a systematic training method. In the paper, a behavioral simulator reproducing a lifelike URE conversation is presented, which was developed exactly to train URE skills. The didactical idea backing the simulator is an interaction between user and game, based on a narrative and relational model developed by one of the leading companies in the field. The effectiveness of the simulator was verified through an experiment, whose design, implementation and results are described. The experiment intended to verify the internal validity, that is if users playing systematically with the simulator improved their performance with the simulator itself, as well as the external validity, that is if users also enhanced their URE skills. Results showed users’ improvements in both aspects.

**Keywords:** User requirements elicitation · Requirements engineering · Online applications · Behavioral simulator · Soft skills training

## 1 Introduction

The first step when designing an online application is to define the information needs it intends to fulfill and the services it is expected to provide. Contents and functionalities have to be implemented according to the typology of users the online application is mainly intended to reach. In order to use and enjoy it, in fact, users require the application to be able to satisfy their requests. The website of an enterprise devoted to baby care, for instance, will probably be visited mostly by parents of young children, parents-to-be, and people somehow related to the previous two groups; therefore, to meet the expectations of such users, the website must provide contents related to

infants' needs like products available, location of stores, information about different stages of children's growth, and functionalities like buying online or having the complete view of expensive products (e.g. car seats). The process of defining users' requirements for online applications and, in general, for any type of software system, is called User Requirements Elicitation (URE). URE is one stage in the requirements engineering process, which comprises other activities such as requirements prioritization and operationalization, and it is, according to many, the most critical stage for the success of the project (Hofmann and Lehner 2001; Hickey and Davis 2002). The term "elicitation" points out the delicate role of the analyst, who has to take an active listening attitude in the dialogue with system stakeholders and with intended users, being able to seek, uncover and elaborate requirements. Literature about requirements engineering, however, does not provide a uniform presentation of the steps involved, not even a shared definition of this activity itself. It is commonly recognized, though, that URE is about learning and understanding the intentions of clients in developing the system as well as the needs of the users, with an important role being played by invention and creativity (Robertson and Robertson 1999; Maiden et al. 2004). The success of the process largely depends on the analyst's communication skills and expertise, since URE is (a) communicative, (b) interdisciplinary, and (c) practical in nature, as explained below.

- (a) It can be seen as a dialogue among three actors: the client, whose intentions and expectations about the software have to be precisely clarified and made explicit; the intended users, whose characteristics have to be pointed out in order to understand their requirements; the analyst, who needs to collect the more elements as possible to realize the client's desires and fulfill users' needs.
- (b) URE is performed in a variety of settings, from the development of websites and mobile applications to the design of complex pieces of software, from the implementation of enterprise systems to the development of market product lines. Different techniques are employed and approaches adopted, depending on the specific context of the project; such techniques and approaches have been borrowed and adapted from different disciplines, such as the social sciences (e.g. communication sciences, marketing), organizational theory, knowledge engineering, group dynamics. Only a few of them, though, have been developed specifically for URE (Zowghi and Coulin 2005).
- (c) URE is an early but critical stage in the development of software systems, since in many cases, apart from a set of shared fundamental goals, functions, and constraints for the system or an explanation of the problems to be solved, contents and functionalities have still to be discussed, clarified, even identified. Stakeholders, then, might have different positions, or might be blind towards real users' needs or technical abilities. Moreover, most of the times stakeholders and analysts come from widely different professional areas, and do not share enough common understanding of concepts and terms; it is also often the case that the analyst has not enough familiarity with the problem, or that stakeholders do not realize what is actually feasible or realistic. All these drawbacks only emerge during the practice, that is in the actual system engineering, and require practical solutions.

As a communicative activity, URE is influenced by the multifaceted, unexpected, unforeseeable variables occurring in human interactions, and it is, thus, subject to a large degree of error. The establishment of a collaborative and positive attitude among the participants is essential to reduce incorrect, incomplete, inconsistent collection of information, and to overtake misunderstandings and misalignments between the parties involved. Despite the many techniques and approaches developed to perform URE, as well as classifications to organize such plethora of techniques and help analysts to select the most suitable one for their case, the training of novices still largely relies on practice (learning by doing) and experts' emulation (learning by observing). The need to work towards reducing the gap between experts and novices has been widely denounced, but "requirements elicitation still remains more of an art than a science" (Zowghi and Coulin 2005).

In this paper, an experiment is reported, which was designed to validate the effectiveness in training URE skills of a behavioral simulator, exactly developed for this task. The simulator derived from a joined idea between an academic stakeholder, who teaches and makes research in the area of online communication, and an industrial stakeholder, who is one of the leading international companies in the design, construction and validation of behavioral simulators to train soft skills in different working settings.

In what follows, we first set the state of the art of current techniques and approaches to URE, then we present the framework to understand and analyze online communication that led the design of the simulator. The experiment will be later described, results discussed, and future developments proposed.

## 2 Techniques, Methodologies and Models of URE

A wide variety of techniques, approaches and tools have been used for requirements elicitation. Zowghi and Coulin (2005) selected a core group of eight of them, which they claimed are representative of those that are both state of the art and state of practice. These eight 'families' of techniques are the following ones: interviews, in which stakeholders are interviewed by an analyst in a structured, unstructured or semi-structured manner and are best applied when there is a limited understanding of the domain from the part of the analyst; domain analysis, where related documentation and applications/competitor systems are examined; group work, implying a direct commitment of the stakeholders and cooperation among them; ethnography, which is particularly effective to investigate intended uses of the system and the addressed public; prototyping, implying building prototypes of the system to support the stakeholders and analysts to collaborate on possible solutions, which may be expensive but is extremely useful when entirely new applications have to be developed; goal based approaches, in which high-level goals of the system are decomposed and elaborated into sub-goals; scenarios, that are narratives describing expected interactions between users and the system; viewpoints, where the domain is modeled from different perspectives (such as from its operation, its interface, its competitors, its users) in order to have a complete description of the system.

The most commonly referenced empirical studies, when researching the effectiveness of elicitation techniques, are case studies (Takamoto and Carroll 2004; van Velsen et al. 2009; Martin et al. 2012) and experiments (Pitts and Browne 2004; Agarwal and Tanniru 1990). Several attempts have been made to aggregate the empirical research regarding URE to compare the effectiveness of elicitation techniques (Zhang 2007; Dieste and Juristo 2011; Davis et al. 2006).

Interviewing has emerged as the most used URE technique (Hadar et al. 2014; Davey and Cope 2008). Compared to other methods, interviewing is an interactive and engaging activity and enables the analysts to easily change the questioning strategy based on the received responses (Davis et al. 2006; Zowghi and Coulin 2005). Disadvantages of interviews are that they can be very resource demanding and require a longtime to collect answers from all the relevant stakeholders. Additionally, interviews are usually not exhaustive and do not explore all possible scenarios.

Usually, more than one technique or approach is adopted, because of the multifaceted and iterative nature of the URE process. Methodologies are reported in the literature, which propose combinations of approaches and techniques to achieve the best possible results in specific situations and environments (Checkland and Scholes 1990; Goguen and Linde 1993). When choosing a methodology, it needs to be tested in the context of use, because it can either enhance or constrain communication (Coughlan and Macredie 2014).

Process models have been proposed over the years to describe the different stages of the URE and, thus, to guide the selection of the techniques to be used (Sommerville and Sawyer 1997; Hickey and Davis 2002). Also frameworks for systemizing requirement elicitation methods have been developed (Carrizo et al. 2014). However, these models and frameworks only sketch generic roadmaps of the process; their inability “to provide definitive guidelines is a result of the wide range of tasks that may be performed during requirement elicitation, and the sequence of those activities being dependent on specific project circumstances” (Zowghi and Coulin 2005: 23).

A growing trend in software development is agile development, which poses new challenges on traditional requirement elicitation processes (Paetsch et al. 2003; Balasubramaniam et al. 2010). In agile development, an iterative URE process is used, where requirements emerge and are validated during software development. The focus is on continuous user interaction and user-centered design. A major advantage of iterative URE is the ability to adapt faster to changing requirements, but a major concern is that non-functional requirements (qualitative requirements) easily get neglected in the process (Ramesh et al. 2010).

Tools have been developed to aid URE process. A tool that provides the analyst with cognitive support is, for instance, Requirement Apprentice, which gives automated assistance for requirements acquisition (Reubenstein and Waters 1991). Other tools have been created for easier collaboration, like EasyWinWin, a collaborative user requirement negotiation software tool (Gruenbacher 2000). Some other tools aim at engaging stakeholders in the elicitation processes, as iThink, which has applied gamification to the URE process (Fernandes et al. 2012): it rewards the user with points for suggesting and analyzing user requirements.

Unfortunately, the adoption of new techniques and tools is limited, due to the lack of scientific evidence that they could provide cognitive support and domain knowledge to the analyst (Zowghi and Coulin 2005).

When committed to develop online communication projects, our research team also employs a mix of techniques to perform URE, including interviews, meetings and focus groups with stakeholders, as well as user scenarios. An original methodology developed by USI's team exactly for URE, is called User Requirements with Lego (URL) (Cantoni et al. 2009a, b). URL is based on Lego® Serious Play®, “an experiential process designed to enhance innovation and business performance” (Lego n.d.), by ‘giving your brain a hand’; in fact, the core idea is doing while thinking, in order to stimulate and enhance creativity. URL is a sort of extension of Lego Serious Play, designed to support the definition of Information Architecture and content strategies in online communication. In particular, URL helps in finding tacit, difficult to grasp communicative requirements that usually do not emerge with other techniques. For this reason, URL has to be intended as an additional methodology, used besides formal and structured strategies (such as interviews and focus groups) to uncover and define user requirements.

The development of the URL methodology as well as the construction of the behavioral simulator presented in this paper, were led by the Online Communication Model (OCM), a model which ideally represents all the components of online communication artifacts – hereafter referred to with the generic term ‘online applications’ – like websites or mobile applications, and constitutes the framework of our understanding of URE. Differently from other models used to map URE, which are mostly based on processes and domain knowledge, OCM looks at computer-based systems from the point of view of communication, adopting a holistic approach.

In the dialogue between analysts and system stakeholders, then, a number of communication misunderstandings and misalignments may occur, which make it harder to design an online application that is able to satisfy the expectations of stakeholders and the needs of intended users. For this reason, after describing the OCM, some of the most common misunderstandings will be discussed, which were considered in the development of the simulator as well as in measuring the performances of students who took part in the experiment.

### **3 A Framework to Understand and Analyse Online Communication**

#### **3.1 Online Communication Model (OCM)**

OCM (Cantoni and Tardini 2006, 2010; Tardini and Cantoni 2015) was developed to ideally consider all the elements and actors involved in a communicative activity taking place online. It goes beyond a naive dichotomy that sees online applications either as mere technological artefacts, to be handled by engineers or, on the opposite side, as advertising tools, to be managed by visual communication experts. The one or the other interpretation, in fact, are only partially true, and both suggest that online applications are static objects. OCM considers them as dynamic entities with a proper life and

typical activities, like a shop or a press agency, and groups their constitutive elements in four dimensions or pillars (Fig. 1):

1. *contents and services/functionalities*: the more or less structured ensemble of information pieces and services provided in the application, such as information provision, news reporting, buying, chatting, product or service reviewing;
2. *accessibility tools*: the collection of technical instruments, which make the contents and services accessible, like hardware, software, and interface;
3. *people who manage*, who are the group of people who design, implement, maintain and promote the application;
4. *users/clients*, who are the group of people who access and use the application

The first two dimensions are related to ‘things’, while the other two ones are related to ‘people’. There is, then, a fifth dimension that completes the framework:

5. *ecological context or relevant info-market*, which gives to every element of the application its precise meaning, value and place within the broader context of the web.

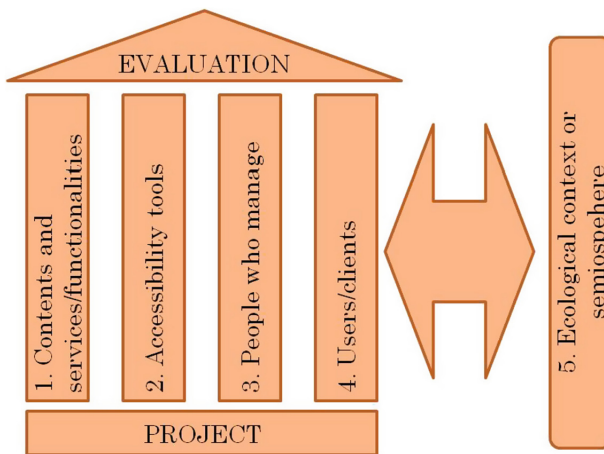


Fig. 1. Online communication model (adapted from Cantoni and Tardini 2006).

### 3.2 Communication Mistakes

As a result of a broad research they conducted with different teams of web design and development, Cantoni and Piccini (2004; see also Cantoni and Tardini 2006) defined the most common misunderstandings or mistakes between analysts and stakeholders of an online application referring them to three main scopes: (a) the *speakers* that are those who take part in the design dialogue; (b) the *relationship between the online application and the “world”* it is referred to; (c) the *time* spent to design and develop the application. Within each scope typical communication mistakes might occur, in particular:

1. concerning the speakers, it might happen:

- a *mistake of person* that is when analysts and developers undertake the process of URE, design and development of the system with a person, who does not adequately represent the client, in the sense that has not a decision-making role or an adequate knowledge of the company's characteristics or needs;
- that *each other's role and expertise are not valued*, because there are reciprocal prejudices, stereotypes or impatience;
- that one "*can't see the forest for her trees*", because the client and/or the analyst only focuses on one aspect of the system, which is not the most relevant, neglecting its final goal and the whole picture;
- *to believe that the other speaker thinks and works as one does*, uses the same language, has the same knowledge of a field, follows the same procedures;
- *to believe that one has to think as the other speaker does*, which can be the case both for the analyst, when s/he pushes the client to change some aspect of the company because s/he has a technical solution for it, as well as for the client, when s/he starts suggesting technical or implementation solutions s/he is not really expert about;
- that one or the other of the speakers keeps *silence*, because s/he believes s/he has to think as the interlocutor but realizes s/he is not able to do so, and thus prefers to avoid the risk of failing or giving a bad impression;
- that one of the speakers, usually the analyst, has a sense of *shyness*, due to the common belief that "the client is always right", and does not engage him/herself in an active dialogue with the client but passively obeys to his/her requests;

2. concerning the relationship between the "world" and the online application, three types of communicative mistakes can be distinguished:

- *simulation* that is when the description made by the client of the "world" s/he wants to represent with the online application – which might be a company represented by a website, a service implemented on an online platform or a smartphone application – does not correspond to the reality, trying to make such reality more attractive than it is;
- *dissimulation*, which is the opposite of simulation, and occurs when the client tries to hide aspects of the "world" that are considered weak or negative;
- *utopia of the twofold* that is when the speakers – usually the client – believe that the online application will be a digital counterpart of the "world" it should represent, exactly reproducing its complexity;

3. concerning for the time factor, two are the most common types of communicative mistakes:

- *haste* to have the online artifact ready, neglecting the time needed for its design, prototyping, implementation and testing, maybe just considering the ease of its technical production;
- *hesitation*, mostly from the part of the client, who underestimates the time needed to realize the online application and continues to ask for changes and improvements.

## 4 Behavioural Simulators to Train Soft Skills

### 4.1 The Concept

Simulators are well known in education. Usually they are employed to train hard skills, which are specific teachable abilities that can be defined and measured, like typing, accounting, using a software. Simulators are also very popular to train procedural knowledge that is how to do things, like driving a car, flying a plane, operating a patient (see, for instance, the wide literature on surgical simulators or machine drive simulators). On the contrary, simulators for training soft skills are less known. Soft skills are personality-driven abilities, related to the emotive and communicative sphere, like patience, interpersonal relation, teamwork, communicative attitude. They can make a difference in the way a person performs in certain activities and at work, but are difficult to be defined, taught and, thus, measured.

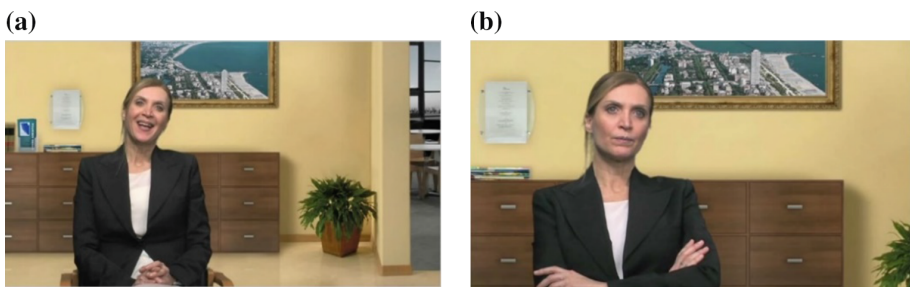
The simulator validated in our experiment was designed to train URE skills, which comprise a number of soft skills, many of them related to people's communicative attitude, like negotiation style, self-control, empathy, complaints management. It was realized thanks to a joined idea of eLearning Lab, the laboratory of Università della Svizzera italiana in charge of improving the quality of teaching through the use of ICTs, and LifeLike Interaction, an international enterprise that designs and realizes behavioral simulators to improve job outcomes that rely in the interaction among people (LifeLike [n.d.a](#)). LifeLike simulators are based on the idea that "our brain does not accumulate data, rather, it memorizes experiences in the form of stories" (LifeLike [n.d.b](#)); different types of behavioral simulators have been developed by LifeLike for specific tasks, like to enhance customers' needs understanding, team management, communication effectiveness, sales closing, managing critical relationships, gathering information. In the next section, the simulator developed to train URE skills is presented.

### 4.2 How It Works

The didactical idea behind the simulator is an interaction between user and game, based on a narrative and relational model developed by LifeLike (the LifeLike Interaction®). The user plays an interactive game-interview, in which s/he plays the role of an online communication consultant, who meets a client willing to create a website for her enterprise. The game experience is highly realistic and based on emotion. In order for the dialogue to be as much realistic as possible, the situation has to be credible and the characters must have a professional as well as a psychological and private profile. The client, in our case, is Mrs. Manuela Cesticchi, the manager of a family-driven hotel on the Adriatic coast (Italy), who wants a website for her hotel, because she knows that her competitors also have a website and understands the potentiality of ICT as new promotion channels. The character of Manuela is played by a professional actor and the video sequences are built with advanced cinematographic techniques. All LifeLike simulators are built following the same technique. Actors are required a fine performance, since reading the script is not enough: they have to interpret a number of



different moods and attitudes, which are communicated mostly non-verbally, with kinesics, facial expression, tone of voice (see Figs. 1 and 2). Real-time is mandatory, given that time has a real impact on the evolution of the interview and the reaction of the client; in fact, there is no pause button. Starting from a ‘dream dialogue’, that is the perfect dialogue between an analyst and a client, every dialogical turn was modified so to include a potential problem or an error in the URE process. The interview the user plays is the result of a unique combination of modified dialogical turns, according to his/her choices during the meeting with the client. About 16 million combinations are possible. The dream dialogue was elaborated by eLab, who has expertise in the field of online communication, URE and usability in particular; the dream dialogue was then modified by LifeLike according to its script rules.



**Fig. 2.** (a) Mrs. Cristicchi in a welcoming attitude (b) Mrs. Cristicchi in an annoyed attitude

The game is articulated in the following steps:

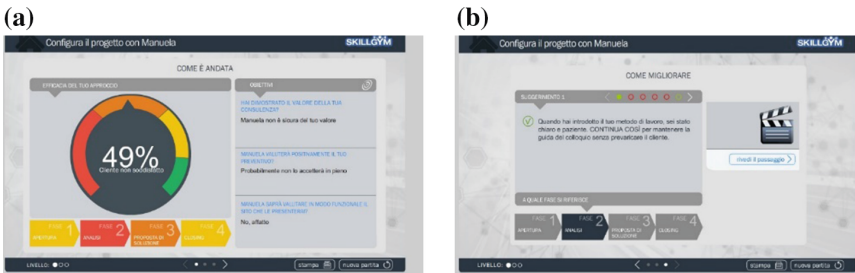
1. the user logs into the system and is introduced into the simulator with a short video; s/he also receives a short description of Manuela Cristicchi’s professional and personal profile;
2. the user plays the game: at each dialogical turn, s/he has to choose among three to five alternative statements, which might be either questions for or answers to Mrs. Cristicchi (see Fig. 3);
3. at the end of the game, the user gets: a score in percentage representing his/her performance in the URE dialogue according to the four stages of the game (opening, requirements analysis, solution proposal, closing) (see Fig. 4), an analytical feed-back on different stages of his/her dialogue with the possibility to check the game sequence (Fig. 5), a comment by Manuela, in the form of a telephone call she makes to her son and during which she reports her impression of the meeting and her decision;
4. the user is invited to play again to improve his/her performance.

The feed-back is based on the following indicators, weighted by the system to give the overall game score (see Fig. 5):

- negotiation style: refers to the user’s spontaneous approach to the task. It can be assertive, aggressive or appeasing;



**Fig. 3.** Alternative choices in a dialogical turn of the LifeLike game.



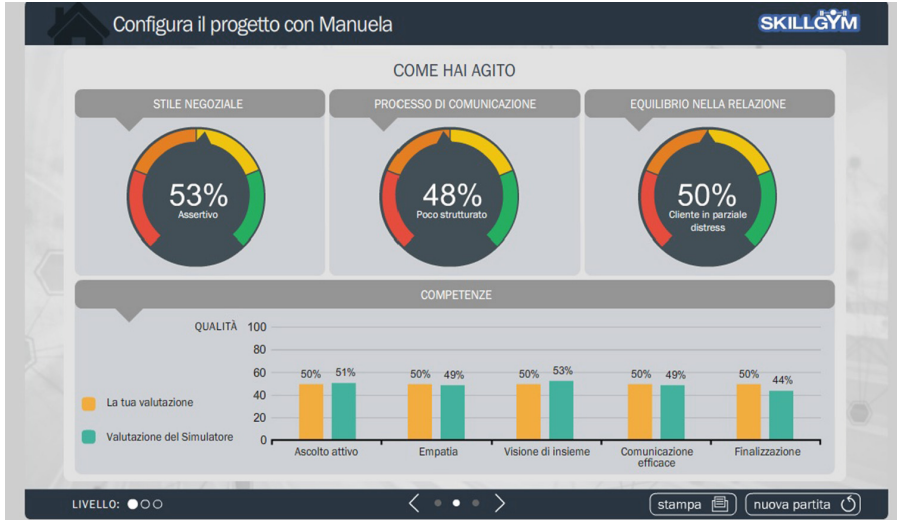
**Fig. 4.** (a) Overall game performance weighted according to the 4 stages of the dialogue (b) Analytical feed-back on each stage of the dialogue

- process of communication: indicates the extent to which the user was able to follow the ideal dialogical path, managing difficulties and unexpected situations;
- quality of relation: refers to the atmosphere created by the user during the dialogue, that is the extent to which s/he was able to put the client in a comfortable state;
- focus on the other part: is about the ability of the user at taking the right decision and accomplish the task without wasting time and disappointing the interlocutor;
- self-control: refers to the perceived consistency of the user’s behavior;
- finalization: refers to the ability of “closing” the negotiation/dialogue.

At the end of the game, before the simulator gives its feedback, the user is asked to evaluate him/herself, through a set of questions related to the above indicators, to which s/he has to assign a score in percentage. Figure 5 reports the comparison between the scores assigned by the user to him/herself and the scores assigned by the system.

Like a muscle, soft skills need continuous and systematic training. As one single training session is not enough to strengthen a muscle, in the same way, one single game

is not enough to enhance the desired skills. LifeLike Interaction fine-tuned a training protocol for behavioural simulators to be effective, which requires the trainees to play a certain number of games over one year, and their performances being monitored thanks to periodical check-ups.



**Fig. 5.** Game performance according to the system indicators and comparison between the user's self-evaluation and the system evaluation

## 5 Validation of the LifeLike Simulator to Train URE Skills

### 5.1 Research Questions

The experiment was driven by two research questions:

- Do users playing systematically with the LifeLike simulator improve their performance with the simulator?
- Do users playing systematically with the LifeLike simulator enhance their user requirements elicitation skills?

Question (a) refers to the ability of players to reach better results practicing with the simulator, that is if they are able to understand the underlying rationale of the simulator and learn from it to improve their scores. If a selected group of trainees does not improve its performance with a tool/game by systematically training with it, this means either that trainees have some kind of learning difficulty – due, for instance, to a wrong estimate of the knowledge level required – or that the tool/game presents design failures – like unclear rules or inconsistent rationale.

Question (b), on the other side, represents the core of the experiment, as well as the success or failure of the simulator to accomplish the task it was developed for.

The experiment was conducted at USI with a class of 69 students, attending the second year of a Bachelor program in Communication Sciences, in the second semester of the academic year (17th February 2014–30th May 2014). The class was divided into two groups, the experimental and the control group, which trained with the simulator according to the schedule in Fig. 6. Students were randomly assigned to one or the other group, paying attention to have a balance between men and women. The experimental protocol followed five main steps described below. Figure 7 is a representation of the experimental procedure for the two groups.

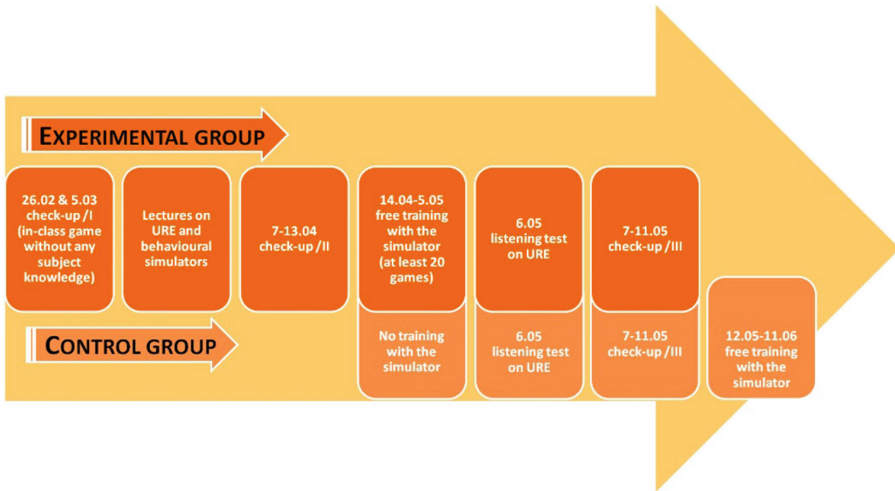


Fig. 6. Experimental protocol

## 5.2 Experimental Protocol

- 1<sup>st</sup> check-up: both groups played 4 games over a 2 h class, after an overview of the experiment and a brief introduction to the simulator. Students were assumed to have the same level of specific knowledge (i.e. related to URE and online communication) and to have no experience with the practice of URE. Goal of the first check-up was to measure the individual as well as the group starting performance.
- 2<sup>nd</sup> check-up: both groups played 4 games over one week (the game location being not relevant), after attending classes on URE and receiving details about the design, features and use of the simulator. Students were taught about different techniques and approaches to URE, case studies were discussed, and the OCM was presented as theoretical framework that drove the design of the simulator and constituted the point of reference for the URE practice. Goal of the second check-up was to measure the individual as well as the group improvement in playing with the simulator, after receiving field-specific and tool-related knowledge.

After the second check-up, students were split in groups. The experimental group was required to freely train with the simulator over 4 weeks (from 14th April to 5th May), playing a minimum of 20 games. After that period, accounts of students belonging to the experimental group were blocked.

3. 1<sup>st</sup> Listening test: after the training period, on May 6<sup>th</sup> both groups underwent a ‘listening test’ that is a test specifically designed to measure if students’ URE skills benefited from the training. The name recalls the core of the elicitation activity, which implies the analyst to actively listen to his/her client, in order to uncover and clarify the requirements of the online application at stake. The test consisted in a video showing a lifelike dialogue between an analyst and a client willing to re-design the website for her luxury cruises enterprise (see Fig. 7a). Students had to carefully look – and listen! – at the video and annotate on a pre-set scheme three categories of elements:
  - (a) what they learnt from the dialogue in terms of the 5 elements of the Online Communication Model (e.g. contents and functionalities of the website, people internal or external to the enterprise devoted to manage produce and manage it); this part was awarded 3 out of 10 points;
  - (b) what else they need to know to successfully complete the URE, but that did not emerge from the dialogue, again in terms of the 5 elements of the OCM; this part was awarded 3 out of 10 points;
  - (c) communication mistakes between the client and the analyst, listed according to the types described in Sect. 3.2 that are: problems regarding the speakers, problems regarding the world and its representation, and the time factor (Cantoni and Piccini, 2004); this part was awarded 4 out of 10 points.
4. 3<sup>rd</sup> check-up: the day after the listening test (May 7th), a final check-up was made, in order to test if the experimental group improved its scores thanks to the systematic training with the simulator. From May 8th until the exam, held on June 11th, the control group was given the possibility to freely train with the simulator, in order for those students to reach the same level of preparation for the exam as the students of the other group. Playing with the simulator did not directly grant students a higher score in the exam, but was supposed (as the experiment in fact later showed) to support their preparation to that part of the exam devoted to URE.
5. 2<sup>nd</sup> listening test: one part of the exam, held on 11th June 2014, was constituted by a second listening test, aimed at testing if students improved in understanding URE, thanks to the knowledge acquired during the course and to the activities proposed. An ad-hoc lifelike video (see Fig. 7b) was shot for the second listening test, showing a URE conversation between an analyst and the communication manager of a cultural association devoted to restoration of pieces of art in Venice (Italy), who wished to develop a website to let interested people know about the association activities and do fund-raising.

It has to be noted that students had access to the feedback part of the game only during the systematic training, while the check-up games did not give them any feed-back. This was to avoid that the control group could learn from its mistakes by listening to the feedback.



**Fig. 7.** (a) A frame of the first listening test (b) A frame of the second listening test

## 6 Results

### 6.1 Training with the LifeLike Simulator and Improvement in Game Performance

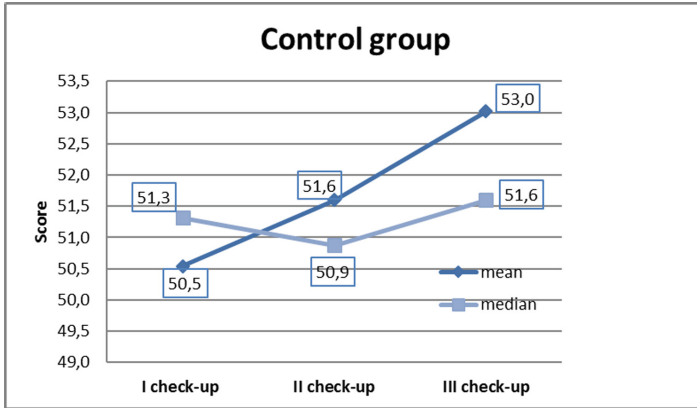
Charts 1 and 2 represent the values of mean and median of the control group (c.g.) respectively the experimental group (e.g.) across the three check-ups. The fact that the evolution of the median values followed the evolution of the mean values in every check-up shows that the two groups were normally distributed.

At the first check-up (c.g.  $N^{\circ} = 30$ ; e.g.  $N^{\circ} = 34$ ), the two groups performed almost the same (median = 51.3% and mean = 50.5% for the c.g.; median = 51.4% and mean = 50.6% for the e.g.), meaning that they had the same starting level of confidence with the use of the simulator.

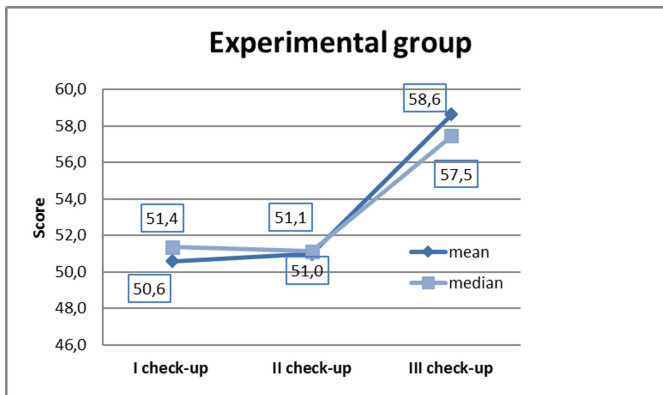
After in-class lectures on URE and a familiarization with behavioral simulators, both groups improved a bit their performance: at the second check-up the mean value of the control group ( $N^{\circ} = 28$ ) reached 51.6% and that of the experimental group ( $N^{\circ} = 31$ ) reached 51.0%. The median values of both groups, however, slightly decreased. This result suggests that knowledge of the domain and of the type of tool alone do not influence game performance.

At the third check-up – that is at the end of the systematic training of the experimental group with the simulator over three weeks – a clear difference between the two groups was observed: experimental group ( $N^{\circ} = 25$ ) dramatically improved its performance, with a mean value that reached 58.6% and a median value that reached 57.5% (i.e. gaining 7.0% if compared to the first check-up). The control group ( $N^{\circ} = 22$  students) that did not play, on the opposite, had only a slight increase in the values: the mean value moved to 53.0% and the median value moved to 51.6% (i.e. gaining 0.3% if compared to the first check-up).

In order to verify if the difference in the groups' performance was statistically significant, a T-test between independent samples was performed for each check-up. The result of a T-test at 95% confidence level showed that the difference of the mean values was not significant on either of the three check-ups ( $p = 0.768$  at the 1st check-up;  $p = 0.498$  at the 2nd check-up;  $p = 0.118$  at the 3rd check-up). It is, thus, not possible to state that a systematic training with the simulator allows to improve own



**Chart 1.** Mean and median values of control group across the three check-ups



**Chart 2.** Mean and median values of experimental group across the three check-ups

performance with the simulator itself, even if this is what actually happened in the considered sample.

For the 3rd check-up, a T-test between independent samples was done also for specific indicators used by the simulator to calculate the performance. The difference between mean values resulted to be statistically significant for the following indicators: negotiation style ( $p = 0.07$ ), self-control ( $p = 0.048$ ), and finalization ( $p = 0.049$ ).

## 6.2 Training with the LifeLike Simulator and Improvement in URE

Table 1 reports both the overall results (mean and median) of the two groups' performances in the first and the second listening test, and the values they obtained in each of the three parts of the listening tests.

The experimental group performed better than the control group in the first listening test, with a mean advantage of nearly one out of ten points (5.98 for the exp. g. against 5.04 for the c.g.). If the median value is considered, the performance is even higher: the value separating the higher half of the experimental group is 6.6 against 4.9 of the control group.

**Table 1.** Results of the experimental and control group at the 1<sup>st</sup> and the 2<sup>nd</sup> listening test

	I listening test <i>(only the experimental group trained with the simulator)</i>		II listening test <i>(both groups trained with the simulator)</i>	
	Experimental group (N = 23)	Control group (N = 24)	Experimental group (N = 23)	Control group (N = 24)
Mean	5.98/10	5.04/10	6.43/10	6.44/10
Median	6.60/10	4.90/10	6.23/10	6.47/10
Mean part (a) + (b)	3.11/6 3.40/6	2.64/6 2.50/6	3.61/6 3.78/6	3.55/6 3.52/6
Median part (a) + (b)				
Mean part (c)	2.86/4	2.36/4	2.82/4	2.88/4
Median part (c)	3.10/4	2.40/4	3.00/4	3.00/4
Mean part (a)	1.67/3	1.57/3	2.04/3	2.01/3
Median part (a)	1.80/3	1.60/3	2.00/3	2.00/3
Mean part (b)	1.44/3	1.08/3	1.57/3	1.54/3
Median part (b)	1.70/3	0.80/3	1.70/3	1.60/3

The experimental group performed better in each of the three parts of the listening test, in particular, the median value of the second part (“what else they need to know”) showed a relevant difference of nearly one point out of four (exp. g. 1.7 against c.g. 0.8). This result might mean that training with the simulator, users became more sensitive to the need of collecting all the relevant information to develop a successful application and, conversely, more alert in detecting the missing information.

Results of the second listening test showed a slight improvement for the experimental group – both overall and in the single parts of the test – but a dramatic improvement of the control group performances. It has to be remembered that after the first listening test the experimental group was denied access to the simulator. Instead, the control group systematically trained with the simulator: its mean and median scores raised of about 1.50 points out of ten. The median reached 6.47 against the previous value of 4.90, and the mean reached 6.44 against 5.04. If the single parts are taken into account, users’ performances improved especially in the first two parts of the listening test (“what they learnt” and “what else they need to know”), more than in the third part (“communication mistakes”), confirming the above observation.

T-test for independent samples were performed also for the two listening tests, in order to verify if the difference in the mean values between the two groups was



statistically significant. The T-test for the first listening test showed a significant difference between the two groups ( $p = 0.025$ ) that is that a systematic training with the LifeLike simulator can improve URE skills. The T-test for the second listening test, instead, did not show a significant difference ( $p = 0.67$ ).

## 7 Conclusion

Behavioral simulators are a new type of systems developed to train soft skills, which are personality-driven abilities related to the emotive and communicative sphere. In the paper, the design process, the development and the first results of an experiment aimed at validating a behavioral simulator to train user requirements elicitation skills were presented. The experiment involved students of a second-year Bachelor attending a course on Online Communication. Both their performance in playing with the simulator and their ability in the practice of URE were measured in order to verify the effectiveness of the simulator. Results showed that after a period of systematic training, the median values of the experimental group increased more than those of the control group that did not train with the simulator, even though such observation is not statistically significant, thus cannot be generalized. Results of the listening tests, which were designed to test students' URE skills out of the simulator environment, also showed an improvement in the performance of the experimental group, which was, instead, statistically significant; it is, thus possible to state that a systematic training with LifeLike simulator promoted an improvement in URE skills. In particular, users performed better in the parts of the test related to the requirements identification (i.e. what they learnt from the client's words), and to the unsatisfied requirements need (i.e. what else they need to know that did not come out from the dialogue with the client).

Some factors that probably influenced the experiment, and that are worth to be considered more carefully in a second experimental round, are the following ones: the size of the sample, the drop-out given to the type of users (i.e. students), the timeframe of the training that is usually much longer in non-experimental contexts (LifeLike suggests to play 20 games along 4 weeks). Further analysis, then, should be performed to test the correlation between performance and users' gender.

While, on the one hand, the promising results of the study open new educational opportunities to train designers in URE, on the other hand they raise several opportunities to enhance the technical design of the novel URE training platform. The student's performance can be observed not only by recording his/her choices on the screen, but by analyzing face muscles, by eye tracking, or by learning about his/her movements, breathing, skin, or voice. All the data collected can be mined for tracing the student's patterns and compared to those within a larger student community. The more a given student's behavior is observed and analyzed and opened transparently to the student, the more s/he can learn from his/her learning process and improve performance.

By having the student to talk to the simulator, it is possible to automatically transcribe the communication and analyze the student's oral communication, by using natural language processing, including sentiment analysis. These technologies would

probably help the student to identify and self-analyze communication mistakes and thus improve the current system, which did not yet help students in this particular area.

## References

- Agarwal, R., Tanniru, M.R.: Knowledge acquisition using structured interviewing: an empirical investigation. *J. Manag. Inf. Syst.* **7**(1), 123–140 (1990)
- Balasubramaniam, R., Cao, L., Baskerville, R.: Agile requirements engineering practices and challenges: an empirical study. *Inf. Syst. J.* **20**(5), 449–480 (2010)
- Cantoni, L., Piccini, C.: Il sito del vicino è sempre più verde. La comunicazione fra committenti e progettisti di siti internet. Franco Angeli, Milano (2004)
- Cantoni, L., Tardini, S.: Internet. Routledge, London, New York (2006)
- Cantoni, L., Botturi, L., Faré, M., Bolchini, D.: Playful holistic support to HCI requirements using LEGO bricks. In: Kurosu, M. (ed.) *HCD 2009*. LNCS, vol. 5619, pp. 844–853. Springer, Heidelberg (2009a). doi:[10.1007/978-3-642-02806-9\\_97](https://doi.org/10.1007/978-3-642-02806-9_97)
- Cantoni, L., Marchiori, E., Faré, M., Botturi, L., Bolchini, D.: A systematic methodology to use LEGO bricks in web communication design. In: *Proceedings of the 27th ACM International Conference on Design of Communication*, Bloomington, Indiana, USA, 05–07 October 2009, pp. 187–192. ACM, New York (2009b)
- Cantoni, L., Tardini, S.: The Internet and the Web. In: Albertazzi, D., Cobley, P. (eds.) *The Media. An Introduction*, 3rd edn, pp. 220–232. Longman, New York (2010)
- Carrizo, D., Dieste, O., Juristo, N.: Systematizing requirements elicitation technique selection. *Inf. Softw. Technol.* **56**(6), 644–669 (2014)
- Checkland, P., Scholes, J.: *Soft Systems Methodology in Action*. Wiley, New York (1990)
- Coughlan, J., Macredie, R.D.: Effective communication in requirements elicitation: a comparison of methodologies. *Requir. Eng.* **7**(2), 47–60 (2014)
- Davey, B., Cope, C.: Requirements elicitation – what’s missing? *Issues Inf. Sci. Inf. Technol.* **5**(1), 53–57 (2008)
- Davis, A., et al.: Effectiveness of requirements elicitation techniques: empirical results derived from a systematic review. In: *Proceedings of the IEEE International Conference on Requirements Engineering*, pp. 176–185 (2006)
- Dieste, O., Juristo, N.: Systematic review and aggregation of empirical studies on elicitation techniques. *IEEE Trans. Softw. Eng.* **37**(2), 283–304 (2011)
- Fernandes, J., et al.: iThink: a game-based approach towards improving collaboration and participation in requirement elicitation. *Procedia Comput. Sci.* **15**, 66–77 (2012)
- Goguen, J.A., Linde, C.: Techniques for requirements elicitation. In: *1993 Proceedings of IEEE International Symposium on Requirements Engineering*, pp. 152–164. IEEE, January 1993
- Gruenbacher, P.: Collaborative requirements negotiation with EasyWinWin. In: *Proceedings - International Workshop on Database and Expert Systems Applications, DEXA 2000*, pp. 954–958 (2000)
- Hadar, I., Soffer, P., Kenzi, K.: The role of domain knowledge in requirements elicitation via interviews: an exploratory study. *Requir. Eng.* **19**(2), 143–159 (2014)
- Hickey, A.M., Davis, A.M.: The role of requirements elicitation techniques in achieving software quality. In: *Proceedings of the 8th International Workshop of Requirements Engineering: Foundation for Software Quality*, Essen, Germany, 9–10 September 2002
- Hofmann, H.F., Lehner, F.: Requirements engineering as a success factor in software projects. *IEEE Softw.* **18**(4), 58–66 (2001)

- Lego: Lego Serious Play (n.d.). <http://www.seriousplay.com>. Accessed 7 Feb 2017
- LifeLike: Behavioral simulation (n.d.a). <http://www.lifelikeinteraction.com/en/index.html>. Accessed 7 Feb 2017
- LifeLike: Simulators behavior (n.d.b). <http://www.lifelikeinteraction.com/en/concept.html>
- Maiden, N., Gizikis, A., Robertson, S.: Provoking creativity: imagine what your requirements could be like. *IEEE Softw.* **21**(5), 68–75 (2004)
- Martin, J.L., et al.: A user-centred approach to requirements elicitation in medical device development: a case study from an industry perspective. *Appl. Ergon.* **43**(1), 184–190 (2012)
- Paetsch, F., Eberlein, A., Maurer, F.: Requirements engineering and agile software development. In: *Proceedings of the Twelfth IEEE International Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprises, WET ICE 2003*, pp. 308–313 (2003)
- Pitts, M.G., Browne, G.J.: Stopping behavior of systems analysts during information requirements elicitation. *J. Manag. Inf. Syst.* **21**(1), 203–226 (2004)
- Reubenstein, H.B., Waters, R.C.: The requirements apprentice: automated assistance for requirements acquisition. *IEEE Trans. Softw. Eng.* **17**(3), 226–240 (1991)
- Robertson, S., Robertson, J.: *Mastering the Requirements Process*. Addison Wesley, Great Britain (1999)
- Sommerville, I., Sawyer, P.: *Requirements Engineering: A Good Practice Guide*. Wiley, Great Britain (1997)
- Takamoto, Y., Carroll, J.M.: Designing a mobile phone of the future: requirements elicitation using photo essays and scenarios. In: *18th International Conference on Advanced Information Networking and Applications, AINA 2004*, vol. 2, pp. 475–480 (2004)
- Tardini, S., Cantoni, L.: Hypermedia, internet and the web. In: Cantoni, L., Danowski, J.A. (eds.) *Communication and Technology*, pp. 119–140. De Gruyter Mouton, Berlin (2015)
- van Velsen, L., van der Geest, T., ter Hedde, M., Derks, W.: Requirements engineering for e-Government services: a citizen-centric approach and case study. *Gov. Inf. Q.* **26**(3), 477–486 (2009)
- Zhang, Z.: *Effective Requirements Development—A Comparison of Requirements Elicitation Techniques*, p. 9. Tampere, Finland (2007). INSPIRE
- Zowghi, D., Coulin, C.: Requirements elicitation: a survey of techniques, approaches, and tools. In: Aurum, A., Wohlin, C. (eds.) *Engineering and Managing Software Requirements*, pp. 19–46. Springer, Heidelberg (2005)