Interactive scenario visualization for user-based service development

Jan-Willem van ’t Klooster¹, Bert-Jan van Beijnum¹
¹Remote Monitoring and Treatment
University of Twente
Enschede, The Netherlands
{j.w.vantklooster,
,b.j.f.vanbeijnum}@utwente.nl

Anton Eliens¹,²
²Department of Computer Science
Free University
Amsterdam, The Netherlands
eliens@cs.vu.nl

Hermie Hermens¹,³
³Roessingh Research and Development
Enschede, The Netherlands
h.hermens@rrd.nl

Abstract—Scenarios are commonly used to develop new systems in multidisciplinary projects. However, written scenarios are sequential, not dynamic and often too abstract or difficult to understand for end users. The goal of this paper hence is to extend the use of scenarios in design methodologies, using an interactive scenario visualization (ISV) approach. After discussing scenario-based design, we show that ISV can be used beneficially to develop a new ICT system and that ISV aids in reflection upon the design trajectory. Comparing two software platforms, we found it was possible to develop such ISVs inexpensively, rapidly and with good visual quality. As a case study, we demonstrate the use of a home care telemedicine system in 3D for discussion and development purposes¹. An evaluation among n=22 professionals illustrates that ISVs can prove useful in design, aid in clarification of new systems, are suitable to demonstrate system functionalities, and aid in articulating feedback. Finally, we discuss generalization of the use of ISVs.

Keywords—interactive scenario visualization; scenario-based design; scenarios; home care system design.

I. INTRODUCTION

In situations where the exact use of a new system is not known a priori, scenarios are used to conceptualize and discuss the system’s properties, the behavior of people interacting with the system or the interaction context [1-2]. Written scenarios may however be difficult to understand, especially when they concern abstract concepts. Moreover, their stories are sequential and non-interactive. Also, knowledge barriers may induce unfamiliarity with the proposed technologies, which hampers input of domain experts and end users [3]. Finally, the visualization and imagination of the scenario has to be performed by the reader, based on the given (lengthy) text. When multiple stakeholders are involved this easily leads to interpretation differences or incomplete understanding.

In this study therefore, we developed an interactive scenario visualization technique, to extend the use of scenarios in design methodologies. The goal of visualization is two-fold: to clarify the system to be designed and to facilitate discussion with stakeholders. This paper discusses the results of this explorative study using a case study; a multi-disciplinary design effort to develop an ICT-mediated home care system for independent elderly living at home.

¹The ISV of the case study is available online at http://rmt-server.ewi.utwente.nl/scenvis

The remainder of this paper is presented as follows. Section II discusses the use of scenarios and scenario visualization in the development lifecycle. Section III explains our interactive scenario approach. Section IV discusses the results of the case study. Section V presents the outcomes of a small-scale panel evaluation, which is discussed in Section VI. Section VII presents an outlook to discuss the generalization of our approach. Section VIII finally presents the conclusion.

II. BACKGROUND

A scenario is a story about the concrete use of a (future) service or technology and the consequences of the usage. [4-6]. Scenarios are especially useful in multidisciplinary development projects to perform needs analysis, to guide the development of novel systems [7-8] and to articulate constraints from different stakeholders [9]. Hence, scenarios are used for understanding requirements and they aid in prototyping and evaluation of design concepts.

The scenario product is a description of who does what for what purpose [10] on a specified abstraction level, depending on the goal of the scenario and the moment of delivery in the development lifecycle. In early design stages, the scenario takes on a user perspective revealing PACT (people, activities, context of use and technology used) properties [11].

As the development effort continues, the scenario corpus (i.e. the set of scenarios) increases. Scenarios become more detailed and system-oriented, reflecting the FICS [9,11] properties (functions, interactions, content and service) of the system. The functions describe the functionalities or abilities of the system. The interaction description describes the interaction and modalities the system is used by its users. The content reveals what (measurement) data, information and knowledge should be stored in the system. These 4 properties impose functional and technical requirements on the system, or, as frequently is the case nowadays, on the distributed service.

The FICS properties in turn aid to detail the MVC (Model, View, Control) components of the system. The MVC-model [22,23], used widely in ICT development to manage, separate and re-use components, distinguishes the information model of a service, the logic or algorithmic (control) that controls the service, and the user interface (view) which represents (a state
of) the information in the model. From left to right, Figure 1 shows that a scenario-based design method can be used to iterate from an abstract, user-centered PACT-scenario, via a system-oriented and more concrete FICS-scenario, to a technical service decomposition in terms of information representation (model), user interface (view) and algorithmic (control) of the service. Iterations may take place both within and after the different steps involved, for example as a result of ambiguities, requirements changes or details becoming more clear along the process.

![Scenario decomposition diagram](image)

**From Scenario corpus to Service decomposition**

Figure 1. From abstract, user-centred, to detailed, design-centred scenarios, aiding service decomposition and service development using MVC [24,25].

Scenarios are produced in different modalities: apart from a written storyline they can be in mockup, sequence of maps, animation, theatre, audio-visual or interactive format [1,2,4,13]. Whereas visualized modalities have impact on understandability and consumability of the scenario [12], they come at the price of translation and interpretation of the storyline. The resulting product may be less flexible: e.g. a video version is difficult to modify as the scenario evolves. Moreover, they tend to be time-consuming and expensive to produce or modify. This is also the reason that video productions sometimes are pre-produced (pre-visualization) in virtual environments [23].

Interactive scenario visualizations, like audio-visual visualizations, also have interpretations of the storyline, but have interesting properties regarding understandability, dynamicity and, using current toolkits, limited production effort. In interactive scenario visualization, the story becomes dynamic and reversible, and may hence be traversed in the user’s order of choice, e.g. useful in educational situations or serious gaming [14]. The ISV tempts to activate its audience and is not ‘consuming-only’ anymore. Like audio-visual scenarios, ISVs can be streamed over the internet to reach a large audience. Unlike movies however, interactive scenarios can be linked to data sources, web services and news feeds to increase the actuality of the scenario, to store metadata, or for reuse later in the development lifecycle. An example is the visualization of physical exercises, before using these visualizations in physical exercise software. Yet despite the common use of scenarios and the potential of ISV, ISVs for health-related development are still, as far as we know, very limited.

III. METHODOLOGY

This section describes the methods used in designing a case study ISV, aimed at visualization of a home care system. Therefore this section describes (i) the scenario creation, (ii) the comparison between two software packages used to develop the visualizations, and (iii) the content selection process to create the visualization itself. After the development, a validation has been conducted using questionnaires in small focus groups. This is discussed in Section V.

To develop the scenario, 4 engineers performed a user-based requirements elicitation process [18]. This process consisted of interviewing caregivers and caretakers to find and clarify their desires for a new care system containing services such as medication guidance, telemonitoring and an activity agenda. Their desires were formulated into a visionary scenario. This scenario was validated by a stakeholder workshop, and used to illustrate the direction of development of the new system. To this end, it was visualized.

To develop the visualizations of that scenario, we scoped ourselves and considered two freely available software packages: Alice and Unity. Alice [15] is an easy-to-use model-driven engineering drag-and-drop programming environment aimed at teaching Object-Oriented programming. This aim is achieved by means of storytelling. Using characters in 3D virtual worlds [16], the designer can create a program that tells a story, e.g. visualize a scenario. As of version 3.0, the output code of Alice can be exported to Java source code for further manipulation or addition. Characters of ‘The Sims’ can be used in the created virtual world, so the result can be appealing, albeit somewhat cartoonesque.

Unity [17] on the other hand has a steeper learning curve but offers large flexibility in terms of graphics, story design and realistic representation of daily activities in 3D virtual worlds. Behavior can be extended through adding C# code. The package is meant for (serious) game design instead of storytelling / learning to program, as is the case for Alice. In Unity, it is easy to produce interesting and entertaining results, hence it is widely used for serious games, 3D games and visualizations.

For the content of the ISV, the existing written scenario from [18] were used. The use of a telemedicine system in a nursing home is demonstrated, showing the graphical interface for both a caregiver and a caretaker. The story in the virtual environment is based on an interaction graph, or interactive story graph. The interaction graph represents the different parts of the scenario and define the choices that are available to follow the scenario using different routes. Interaction graphs ease the administration of choices in the story design and the tracking of logical sequences, moreover it serves as an aid to check if all parts of the scenario text are also visualized. An example is shown in simplified form in Figure 2. Diamonds (choices) refers to interactivity in the scenario; sequential (movie-like) scenes may start in the act that follow (eg. ‘take medication’).
Figure 2. Simplified flow chart representing the interaction graph of the scenario visualization with choices (diamonds) and acts (rectangles). When the user finds a screen, a todo list is displayed with activities to conduct. The screen in the scenario visualization is visible in Figure 5, inset on the left.

The control flow of the visualization becomes clear through the use of the interaction graph. However, the available options should be presented to the viewer of the ISV in a consistent and clear way as well [19]. Hereto, different dialogue box designs were tried.

IV. RESULTS

We developed an interactive virtual environment, inspired by the actual nursing home ‘Parc Hoogveld’ in Sittard, The Netherlands. It shows two elderly apartments, a nurse office and a corridor connecting these three spaces, as visible in Figure 5 (end of paper). This Figure shows the result of the development in Unity. The interactive scenario evolves around the role of one of the inhabitants in his apartment; he is able to use the installed home care system freely. The concept of this home care system is an easy to access touch screen on the wall, offering services like medication reminders, agenda, social interaction and aid in doing exercises. The system can also be accessed through the TV in the living room, as indicated by the two screens in the inset on the right. Once the user is in the vicinity of the screen, the options of the home care system become usable as shown by the Figure’s inset on the left. The graphical interface in the virtual world is similar to the actually prototyped version [20].

Figure 3. Virtual nursing home in Alice. Two nursing homes and a connecting corridor are shown. Interactivity (shown in the inset) can not be added in Alice itself, but only through modifying the underlying Java source code.

In the ISV, different functions of the home care system can be explored from the viewpoint of both care receiver and caregiver. Interactive parts are combined with sequential scenes to show these different perspectives. An important point of attention in this regard is the self-explaining property of an ISV: depending on the intended use of the ISV, the visualization should be more or less self-contained. This means that for our discussion purposes, audio and visual comments were added alongside the visualization. However for standalone operation, these comments should indeed replace a person presenting the ISV. In that case extensive hints for the operation of the ISV should be added.

V. EVALUATION

We performed an evaluation to validate interactive scenario visualization by means of the case study. The ISV made in Unity was presented in St. Etienne, France and Enschede, The Netherlands in small groups to n=22 professionals working in healthcare development areas (7) healthcare R&D; (10) academia; (3) ICT consultancy industry; (2) healthcare management). All of the inquired people dealt with healthcare innovation on a daily basis, either from development or from practical perspective.

After an explanation and demonstration of the visualization, a questionnaire was handed out to get the opinion of the panel on this method and to find out how valuable they think it is over other techniques. Only the ISV made in Unity was used in the evaluation: partly due to the mentioned quality constraints of Alice, but also because of time availability of some of the professionals.

Using a 5-point Likert scale (totally disagree codified ‘1’; totally agree codified ‘5’), we measured whether or not scenario visualization:

1) Clarifies innovative care scenarios;
2) Gives insight in the care platform’s proposed functionality;
Both conceptual and finished systems. The creation of ISVs is such a scenario. Indeed, visualizations are useful to clarify the system (eg. nurses) and that it should be recommended to also use the visualization for end users of the scenario visualization stakeholders in the design process on the development insight in the proposed functionality and eases giving feedback attitudes: the (clarification, functionality and feedback) scored positive was expected.

So a more technical requirements. Disagreed on construct 3 varied. This indicates the implementation of the visualization to clarify technical details, 

The results of the evaluation are shown in Figure 4. Scores on construct 3 varied. This indicates the interviewees disagreed on whether such visualizations give insight in the technical requirements. It was not in the aim and implementation of the visualization to clarify technical details, so a more negative attitude in general among the stakeholders was expected. On the other hand, constructs 1, 2, and 4 (clarification, functionality and feedback) scored positive attitudes: the ISV clarifies innovative care scenarios, gives insight in the proposed functionality and eases giving feedback on the development to better involve end-users and stakeholders in the design process.

Comparing the perceived value of ISV over other techniques, we asked how the professionals would use scenario visualization themselves:

- The most common answer (9 people) was usage for requirements elicitation and validation.
- 5 said they would use it to articulate their design ideas.
- 2 people said they would use it for usability testing.
- 6 people said they would not use it.

Remarks given by the evaluations subjects included the recommendation to also use the visualization for end users of the system (eg. nurses) and that it should be easy to create such a scenario. Indeed, visualizations are useful to clarify both conceptual and finished systems. The creation of ISVs is enabled by modern, model-driven software environments.

Another remark concerns the fact that making a visualization potentially improves the scenario text, because one is forced to think about details and eliminate ambiguities in the storyline.

VI. DISCUSSION

According to 14 of 20 interviewees, ISV is useful to facilitate design (requirements elicitation and validation, usability, clarification of concepts). However, the right toolkit should be used: it was clear that Alice does not produce high quality visualizations (so it was decided not to show it to the panel) whereas Unity does - within a development period in the order of days, of course depending on the scale of the result.

The value of ISV over mockups seems that it shows more of the high level process steps and not only the interface. Hence besides application usage, organizational usage is shown too, eventually involving different actors. This is all not the case in mockups. In terms of MVC, it shows, in addition to the user interface (view), also the added value and working of the system (control). Showing organizational usage is also possible in movies, but only 1 storyline can be told. Besides, interaction lacks and production (especially modification) is costly. Using modern toolkits like Unity allow ISVs to be made in the order of days without actors or producers.

In short, benefits, based on this case study, seem to be clarification of a system to be designed, the functionality of such a system and the use for getting stakeholder feedback. ISV can prove useful in design. It is not directly comparable to mockups and certainly has advantages over movies. However, a more detailed comparison should fortify the results using a larger test panel and multiple scenario text vs. visualization comparisons.

VII. OUTLOOK

Although this paper focuses on interactive scenarios for discussion and development purposes, ISVs may just as well play a role in later phases of the product lifecycle. In this section, we discuss two challenges based on expert talks with the professionals that took part in the evaluation:

(i) using ISVs to train users in the operation of an actual system once implemented, and

(ii) connecting ISVs to data sources, to log behavior of users using the scenario for evaluation purposes.

For many years, visualizations are used for training purposes, e.g. in flight simulation. Hence, an ISV can be used to demonstrate usage, and train users of the system. A challenge in this regard is to train end-users, eg. in our case study elderly and nurses, in using the system by demonstrating the functions and by showing them the overall functioning of the system. In this way, users may gain understanding on what the system offers to the different parties involved. Using the ISV, it is e.g. shown how the home care system reminds the client on medicine intake using one of the screens, or how the system notifies care professionals in case of emergencies.
On the other hand, ISVs can also be extended to make use of other information sources such as motion capturing data for realistic movements, web services, electronic client dossiers or training programs. This way, interactive visualizations can be used for more applications and become more directed towards the audience. Also, such a connection enables tracking of users and the evaluation of the ISV when it is shown to multiple users.

VIII. CONCLUSION

This paper proposes interactive scenario visualization (ISV) as an extension of scenario-based design. We showed how interactive scenario development can be used beneficially in the process to develop a new telemedicine system, presenting a methodology that links firstly PACT scenarios, then FICS properties and finally MVC-based service decomposition.

ISVs make scenarios more dynamic in storytelling and they are more flexible than other visualization techniques such as mockups and movies. Scenario visualizations can be created rapidly nowadays, are appealing to work with, and can be extended or modified easily without the need of producers or experienced professionals.

After developing and comparing prototyped ISVs in both Alice and Unity packages, Unity shows to enable more realistic visualizations, better extensibility and better graphical possibilities. Unity, offering visual, model-driven design and scripting, however has a steeper learning curve compared to Alice. In both environments, ISVs can be made in the order of hours to days. Because the results in Alice are poor, we only evaluated the ISV case study developed in Unity.

In the evaluation, we noticed that ISV was received well with respect to getting feedback, clarification and functional aspects: It seems to be promising to use interactive scenarios to clarify innovations, to give insight into system functionalities and to gain feedback of stakeholders. However discussing technical requirements of a system to be designed is more difficult; such discussions do not benefit from scenario visualization directly. Attention should be paid to the degree to which the ISV is self-contained: a visualization that runs autonomously requires more audiovisual guidance than the ISV developed in the present study. The current state-of-the-art of software packages like Unity enables rapid design of scenario visualizations and enables the use of ISVs to involve stakeholders in the design process effectively. ISV can prove useful in design but is not directly comparable to mockups or movies: besides application usage, organizational usage is shown, involving different actors/viewpoints, unlike mockups. In contrast to movies, ISV provides interaction, facilitated production, modifiability and dynamicity. These considerations should be taken into account by designers when using scenario-based methods to get stakeholder feedback.

ACKNOWLEDGMENT

Fred Vuijk and Herjan Treurniet, Douwe Bart Mulder and Jan Kolkmeier (students at the Remote Monitoring and Treatment Group vs. Creative Technology educational program) are acknowledged for their contributions to the creation of the visualizations.

This work is part of the IOP GenCom U-Care project (http://uicare.ewi.utwente.nl), sponsored by the Dutch Ministry of Economic Affairs under contract IGC0816.

REFERENCES

Figure 5. Virtual nursing home created in Unity. The large photo shows 2 apartments and a nurse office. The inset on the right shows 1 apartment, its inhabitant (the main character operated by the user) and the two screens (1 on the wall and 1 as TV in the lower right) of the home care system. In the inset on the left, the screen on the wall is active, showing Agenda, Photo Album, Television and Video Telephony options to explore by the user.