

Capturing Multimedia Requirements Descriptions with Mobile RE Tools

Rick Rabiser¹

Norbert Seyff²

Paul Grünbacher^{1,2}

Neil Maiden³

*Christian Doppler Laboratory for
Automated Software Engineering¹
Johannes Kepler University Linz
4040 Linz, Austria
rabiser@ase.jku.at*

*Johannes Kepler University Linz²
Systems Engineering & Automation
4040 Linz, Austria
nseyff@sea.uni-linz.ac.at*

*City University London³
Centre for HCI Design
London EC1V 0HB, UK
n.a.m.maiden@city.ac.uk*

Abstract

As tools for requirements engineering (RE) become available on mobile devices using their multimedia capabilities to capture requirements descriptions is an obvious opportunity. This paper reports on two different approaches enabling mobile analysts and end-users to add multimedia descriptions to requirements. Based on our mobile tool for scenario-based RE we compare a solution based on the COTS package Microsoft Pocket Word with a novel plug-in solution providing more flexibility for tool users.

1. Introduction

The capabilities of mobile devices have evolved significantly in the last few years. Such devices now provide faster processing, increased storage, and improved connectivity. Mobile Computing is a maturing and growing field and appears to become a dominant computing paradigm [6]. The advancing capabilities of mobile devices like Personal Digital Assistants (PDAs) have led to the development of mobile RE tools supporting novel approaches to requirements capturing. E.g., such tools allow stakeholders to capture requirements directly in the workplace of future system users with benefits for the RE process [4]. Modern mobile devices typically provide multimedia features like microphones or digital cameras. Therefore using such devices to enrich requirements descriptions with sketches, audio notes, pictures, or videos is an obvious opportunity.

Mobile RE approaches are promising to complement already established RE processes and tools. For instance, mobility and multimedia can significantly enrich scenario descriptions and thereby support requirements elicitation [1, 4]. Mobile approaches and tools can be used to walk through scenarios directly in the work context of future system users. Working on-site enables mobile requirements engineers to observe

current system behaviour and interact with system users. In many cases notebooks do not fulfil the needs of mobile analysts and end-users as they often have to walk around during an elicitation. Mobile devices like PDAs provide excellent support for capturing rich media input on-site in manifold ways. Most mobile devices have only limited keyboard functions to enter text. Instead such devices often provide a touch screen. In combination with a stylus users can easily make drawings or sketches. Many devices are also equipped with built-in microphones and digital cameras enabling users to record audio and video. Mobile devices therefore provide a great opportunity to investigate mobility and multimedia for requirements elicitation.

Traditionally, requirements are captured as natural language statements [7]. However, such textual descriptions of real world situations can get very extensive [1]. Furthermore natural language is prone to misinterpretation [7].

The focus of this paper is to present and analyze two options supporting capturing multimedia descriptions of requirements with a mobile scenario tool. We report preliminary results on how these options fulfil the needs of mobile stakeholders based on our experience and initial evaluation studies. We explore which option is more amenable to different user types. As little is known about capturing multimedia requirements descriptions using mobile RE tools, this comparison can stimulate further research in this field.

The rest of this paper is structured as follows: In Section 2 we briefly describe ART-SCENE, a proven scenario-based technique for requirements discovery and the Mobile Scenario Presenter (MSP) [3], our mobile tool for ART-SCENE. Section 3 reports and analyzes two ways how the MSP facilitates multimedia descriptions of requirements. Section 4 discusses lessons learned and further research issues.

2. ART-SCENE and the MSP

ART-SCENE [2] (Analysing Requirements Trade-offs: Scenario Evaluations) is a scenario-driven technique for requirements discovery and documentation. A scenario generated by ART-SCENE consists of a list of normal course events and a list of alternative course events. The normal course represents events relevant to a specific use case. For each normal course event ART-SCENE generates a list of alternative course events ('what-if' questions), which enable stakeholders to define requirements for unexpected or unusual system behaviour.

The ART-SCENE Scenario Presenter tool supports walkthroughs and is mainly used in face-to-face meetings. Stakeholders using the Scenario Presenter are guided by a facilitator [2]. However, formal requirements sessions taking place out of the stakeholders work context potentially diminish the effectiveness of RE techniques [8]. To overcome this issue two options have been analyzed: (1) The ART-SCENE Scenario Presenter has been extended with multimedia scenario descriptions. Such image, video, and audio descriptions are suitable to depict an environment or work context and the human actors in it. As described in [10] multimedia scenarios give stakeholders more recognition cues than text-based scenarios. (2) As stakeholders cannot directly interact with the actors we have been developing the Mobile Scenario Presenter (MSP), a mobile scenario-based tool augmenting ART-SCENE in several iterations.

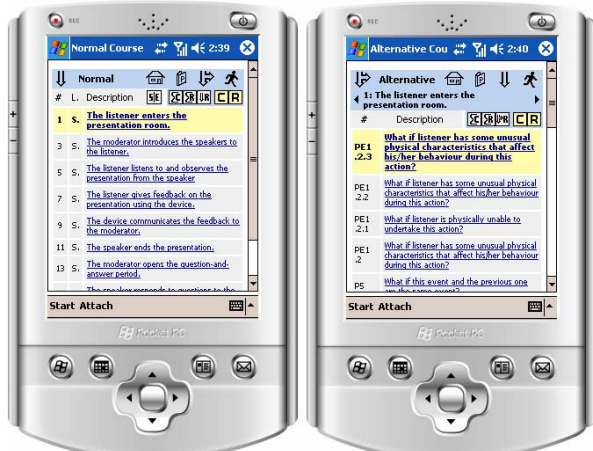


Figure 1: A scenario normal course (left) and generated alternatives for the selected normal course event (right) on the MSP

The MSP is a Personal Digital Assistant (PDA)-based ASP.NET web application enabling mobile stakeholders to undertake scenario walkthroughs on-

site. It uses a mobile browser and wireless access to connect the mobile stakeholder to server-side ART-SCENE scenario and requirements databases. Using the MSP mobile stakeholders can discover and document requirements systematically in the work context using structured scenarios generated by ART-SCENE. For example, the MSP has been used to discovering requirements for a conference presentation support system at ICSE'04 [5]. During an ongoing presentation stakeholders walked through the normal and alternative course events shown in Figure 1. Starting with the event – *the listener enters the presentation room* – they captured requirements or browsed through automatically generated alternative course events like - *what if listener has some unusual physical characteristics that affect his/her behaviour during this action?*

After a mobile scenario walkthrough, the analysts have the possibility to synchronize the PDA with a desktop computer system. Where necessary, they can formulate more detailed requirements descriptions based on the information gathered with the MSP.

A number of features distinguish the MSP from the desktop Scenario Presenter [3] including capturing multimedia requirements descriptions as discussed in the remainder of this paper.

3. Capturing Multimedia Descriptions

When developing the MSP and analyzing its usage it became clear that it is useful to distinguish between stakeholders who are experienced in RE and stakeholders with little or no experience. From a usability perspective trained analysts need different tool support compared to non-specialist end-users. Easy-to-use tools providing detailed guidance and help have been requested by inexperienced end-users. Trained analysts instead preferred more powerful and specialized tools.

When developing the MSP our goal was to create a balanced solution for both experts and non-specialists. We consider multimedia features as useful for both user groups. For example, both can sketch their thoughts about future system requirements using a sketch tool. Users and experts can benefit from a drawing to highlight concepts or from pictures describing a problem they encountered. Beyond that, analysts can benefit from audio or video recordings of interviews conducted with end-users in the workspace.

We designed and developed two different technical solutions supporting capturing multimedia requirements descriptions. In the following we discuss and compare these options.

3.1 MSP I: Microsoft Pocket Word

Our first attempt to realize multimedia requirements descriptions was based on the integration of the Mobile Scenario Presenter with Microsoft Pocket Word. This word processing application is shipped as part of Microsoft Pocket PC operating systems. Besides normal text entry the Pocket Word application also allows audio recording through the PDA's built-in microphone and drawing of sketches using the PDA's stylus.



Figure 2: Multimedia requirements descriptions with Microsoft Pocket Word

When developing the first prototype (MSP I) our goal was to support mainly mobile analysts. Using this first solution, analysts can create a new requirement starting with a short text-based description (left side of Figure 2). A word icon in the Microsoft Pocket Internet Explorer (PIE) allows analysts to launch Pocket Word and to benefit from all its features. These include writing textual descriptions, drawing sketches, and recording audio notes (right side of Figure 2). The created Pocket Word document is linked to the requirement it belongs to using a unique identification number.

The first MSP prototype was used in several studies¹ that focused on the usability and utility of the tools [5]. At the end of each evaluation test users were interviewed about their experiences in working with the prototype. They successfully used the MSP to capture requirements and comments. Although the results were encouraging we revealed several issues:

Users complained about missing feedback when working with the MSP. For example, they never knew whether a requirement they entered was successfully added. Furthermore missing error recovery features required tool-experts to support the users on-site. Serious usability drawbacks originated from using Pocket

Internet Explorer to display the MSP. Users frequently caused errors by clicking on the 'back', 'forward', or 'stop' buttons included in PIE's user interface. PIE's address bar also wastes screen space and the already mentioned buttons were not needed for the MSP.

The studies also revealed critical issues concerning the usage of Microsoft Pocket Word. Only a few test users started Pocket Word to add multimedia requirements descriptions. Three main problems were discovered in interviews with test users:

- Adding audio notes or sketches to requirements by starting Pocket Word is not very intuitive. Users were not aware of Pocket Word's features and therefore did not start it in the first place.
- Pocket Word is not seamlessly integrated with the MSP. Too many clicks are necessary to add a document containing a multimedia object (in the best case it takes seven clicks to add an audio note).
- Pocket Word is rather complicated, especially for novice PDA users. The application has many features offering a lot of options but this makes it even more difficult to use for inexperienced users.

MSP I was an attempt to support analysts working out in the field. Working with the MSP I gave rise to the idea that it could also support end-users eliciting requirements on-site, e.g., by sketching their ideas in their work environment. End-user could then analyze and finalize these preliminary ideas together with an analyst. As this approach seemed to be promising we started to evolve the MSP and developed an updated prototype that better supports the needs of end-users. Taking the preliminary evaluation results into account we also tried to overcome the main usability issues reported in the evaluation of the tool [5].

3.2 MSP II: Extendable Plug-In Solution

A second prototype of the Mobile Scenario Presenter (MSP II) was developed to address the lessons learned and to improve support of end-users through a set of novel features:

- (1) Wizards that guide the user through the creation of requirements and comments step-by-step. This makes it much easier for end-users to understand which information they should enter.
- (2) A tutorial scenario to learn the handling of the MSP without help from tool-experts.
- (3) Improved feedback about errors and important events (e.g., 'requirement successfully added') to increase user confidence in the application.
- (4) A settings menu that allows deactivating the features 1-3 for expert users.

¹ We used a HP iPAQ 5450 Pocket PC, 64 MB, 400MHz, Microsoft Pocket PC 2003 at 240x320.

(5) Replacement of Pocket Internet Explorer with a full-screen browser application. This application is based on the .NET compact framework [9] and uses the functionality of PIE to display the MSP. It has its own user interface without an address bar and without buttons. The user interface consists of two elements, the web view displaying the MSP and a small menu on its bottom. In this menu the URL from which the MSP shall be loaded can be set and the MSP can be re-started.

(6) A new feature for facilitating multimedia requirements descriptions with the potential to overcome the issues discovered when using Pocket Word. The rest of this subsection describes this new concept, a plug-in solution, in more detail.

The implemented plug-in solution allows the integration of arbitrary applications that facilitate capturing of different media formats. The browser application implements a plug-in host interface. Additional applications implement an appropriate plug-in client interface and simply have to be put into the same directory as the browser application to be recognized by it. Each recognized application is automatically integrated into the menu and can be started from there.

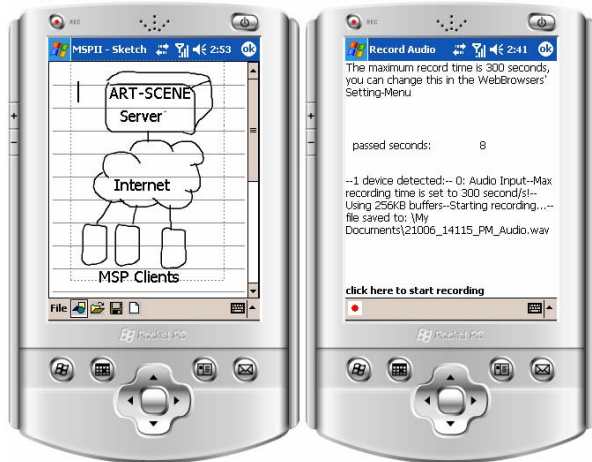


Figure 3: Multimedia requirements descriptions with the plug-ins NoteTaker (left) and AudioRecorder (right)

We implemented two prototype plug-ins (cf. Figure 3), one for the creation of drawings (NoteTaker) and the other for recording audio (AudioRecorder). Files saved with the NoteTaker plug-in are compatible to Microsoft's Pocket PC/Windows Mobile application "Notes" and can easily be exported to desktop PCs. Recordings made with AudioRecorder are saved in the common and widespread WAV-format supported by many applications. In order to support novice users the user interfaces of both plug-ins were designed as sim-

ple as possible. E.g., only a few clicks are necessary to add an audio recording.

When creating a requirement within the MSP, files created with a plug-in application are linked to that requirement with a special ID included in the filename. Similar to the usage of Pocket Word the user can synchronize the PDA with a desktop system and elaborate more detailed requirements in subsequent activities based on the multimedia information gathered with the Mobile Scenario Presenter. The MSP does not automatically convert multimedia descriptions of requirements into text based descriptions. The plug-in solution allows to easily integrate new applications, e.g., for making photos or video recordings. The user is not depending on the functionality of Pocket Word anymore, although he still has the possibility to use it.

4. Pocket Word vs. Plug-In Solution

This section compares the two approaches based on selected quality criteria (partially informed by ISO 9126).

Suitability: This boils down to the question: How well can multimedia requirements descriptions be created? Both options have their benefits and limitations. Microsoft Pocket Word is a proprietary application but with many powerful features. It is not integrated with the MSP and difficult to use for inexperienced users. The in-house developed plug-in solution is seamlessly integrated with the MSP, expandable, scalable, and adaptable. Its biggest disadvantage is the implementation and test effort for creating new plug-ins. While Pocket Word provides built-in functionality to add text, sketches, and audio recordings the plug-in solution can support arbitrary media formats through plug-ins.

Understandability: This aspect is mainly interesting from the perspective of understanding the integration of the capturing tool into MSP. The loose integration of Pocket Word results in a non-intuitive way to create multimedia descriptions. This can be much easier using custom plug-ins where each plug-in is designed to offer only very few functions following the same mode of interaction.

Learnability: The plug-in approach relieves users from having to learn 'yet another application' (if all plug-ins follow the same mode of interaction). This is particularly important for novice users.

Changeability: Pocket Word is a proprietary, commercial application; there is no easy way to modify its features. The in-house developed plug-in solution allows to easily adapt already existing functions and to add new plug-ins if necessary.

Adaptability: Both approaches are rather limited in this respect. Pocket Word is only available on Micro-

soft's Pocket PC OS. The plug-in solution relies on the .NET compact framework.

Time and resource behaviour: The plug-ins have shorter start-up times and lower resource consumption compared to Pocket Word.

Development Effort: COTS-based Microsoft Pocket Word does not require additional development and the application is mature and powerful. The plug-in solution approach perfectly suites the needs of mobile stakeholders but requires additional design, development, and test effort when providing new plug-ins.

5. Conclusions and Further Work

The advancing capabilities of mobile devices offer the possibility to support requirements engineers working out in the field with proper tools. The MSP was presented as an example showing how to complement existing text-based requirements descriptions using multimedia elements. Initial descriptions created with such mobile tools can be used as a first idea for describing more precisely formulated requirements in later phases. Because pure text-based requirements are often prone to misinterpretation we propose users to augment their requirements with multimedia descriptions.

In this paper we described and compared two approaches to enhance our MSP tool with multimedia descriptions of requirements: We presented an approach based on Microsoft Pocket Word to attach documents containing, i.e., audio notes or sketches, to textual requirements descriptions. Evaluation showed that integrating Pocket Word turned out to be non-intuitive for many users. As an alternative we then presented an extendable plug-in solution also facilitating multimedia requirements descriptions but with a better adaptability to needs of non-specialist stakeholders. Finally, we compared the two different approaches showing that each has its benefits and limitations. Pocket Word perfectly supports users which prefer to work with a fully-fledged word processor. Seamlessly integrating such a COTS product is challenging. A plug-in solution is a more flexible and adaptive solution providing just the features most users need. The contribution provided by this paper can help researchers to select an appropriate solution how to facilitate multimedia requirements descriptions.

Although we have preliminary evaluation results on using Pocket Word, such results are missing for the plug-in solution. It is necessary to conduct another evaluation also comparing the two solutions. For the future of the MSP the plug-in solution seems to be more promising. New ways of documenting requirements can easily be added and existing ones can easily be modified to satisfy user needs.

References

- [1] Haumer, P., K. Pohl, K. Weidenhaupt, "Requirements Elicitation and Validation with Real World Scenes", *IEEE Transactions on Software Engineering*, Vol. 24(12), IEEE CS, 1998, pp. 1036–1054.
- [2] Maiden, N.A.M., "Systematic Scenario Walkthroughs with ART-SCENE", in *Scenarios, Stories, Use Cases: Through the Systems Development Life-Cycle*, I. Alexander and N.A.M. Maiden (eds.), John Wiley & Sons, 2004, pp. 161–178.
- [3] Maiden, N.A.M., N. Seyff, P. Grünbacher, "The Mobile Scenario Presenter: Integrating Contextual Inquiry and Structured Walkthroughs", in *Proc. of the 2nd Int. Workshop on Distributed and Mobile Collaboration (DMC 2004)*, within the 13th IEEE Int. Workshops on Enabling Technologies: Infrastructures for Collaborative Enterprises (WETICE-2004), IEEE CS, 2004, pp. 115–120.
- [4] Maiden, N.A.M. and S. Robertson, "Developing Use Cases and Scenarios in the Requirements Process", in *Proc. of the 27th Intl. Conference on Software Engineering (ICSE'2005)*, IEEE CS, 2005, pp. 561–570.
- [5] Maiden, N.A.M., N. Seyff, P. Grünbacher, O. Otojare, K. Mitteregger, "Making Mobile Requirements Engineering Tools Usable and Useful", *14th IEEE Int. Requirements Engineering Conference (RE'06)*, Minneapolis, Minnesota, USA, September 2006.
- [6] Myers, B.A. and M. Beigl, "Guest Editors' Introduction: Handheld Computing". *IEEE Computer*, Vol. 36(9), IEEE CS, 2003, pp. 27–29.
- [7] Sutcliffe, A., *User-Centered Requirements Engineering*, Springer, London, 2002.
- [8] Viller, S. and I. Sommerville, "Social Analysis in the Requirements Engineering Process: From Ethnography to Method", in *Proc. of the 4th IEEE Int. Symposium on Requirements Engineering*, IEEE CS, 1999, pp. 6–13.
- [9] Wigley, A., M. Sutton, R. MacLoed, R. Burbidge, S. Wheelwright, *Microsoft .NET Compact Framework (Core Reference)*, Microsoft Press, 2003.
- [10] Zachos, K., N. Maiden, A. Tosar, "Rich-Media Scenarios for Discovering Requirements", *IEEE Software*, Vol. 22(5), IEEE CS, 2005, pp. 89–97.