

Towards an Agile User Experience Virtual Assistant and Management Platform

M. Patterson, R. Bond, M. Mulvenna,
C. Reid, F. McMahon, P. McGowan
Ulster University
Belfast, N. Ireland

{m.patterson | rb.bond | md.mulvenna |
ci.woodside | fc.mcmahon | p.mcgowan}@ulster.ac.uk

J. McGarry, H. Cormican
Cirdan Imaging Ltd.
Lisburn, N. Ireland
{jmcgarry |
hcormican}@cirdan.com

Much research has been carried out into the integration of User Experience and the Agile software development methodology. However, there has been no formal attempt to develop a computer-based management system to aid in the integration of these two concepts. This paper outlines the initial development of such a system; UX-Vision. It includes how the system operates, the design of its user interface, and the data structure. There is also discussion on how the system could be developed further to include a virtual assistant to indicate when and how UX activities should be carried out within Agile software development.

User Experience, Agile, Virtual Assistant, Management Platform.

1. INTRODUCTION

User experience (UX) is a field which has been gathering interest within the IT industry for a number of years. Developers are beginning to see the value of involving users in the development of their software products. This has led to the development of methodologies such as User-Centred System Design (UCSD) (Guliksen et al., 2003) amongst others (Myers, 1996). These methodologies encourage a user-focused approach to system design where users are involved at every stage of the process. In recent years Agile has become a popular method for software development as a departure from the more rigid traditional methods of software development such as Waterfall, and has been proven as three times more effective (Hastie & Wojewoda, 2015). Agile provides a more flexible methodology for development with input from users also being included. However, the main focus of Agile is to provide a more light-weight approach to software development, rather than traditional, “documentation driven, heavyweight software development processes” (Beck et al, 2001). There are several frameworks for carrying out the Agile methodology, with one of the most popular being Scrum (Schwaber, 1997). It should be noted that, even though contact with users is encouraged within Agile, it does not guarantee that a positive overall UX approach is achieved.

Research has been carried out into the relationship between Agile software development and UX design methods, and, in particular how they can be better integrated. Larusdottir, Guliksen, and Cajander (2017) point out how the Agile process has much to gain from integrating UCSD

techniques. They state, “Agile software development projects do not support UCSD activities simply through the process itself, nor does it prevent it”. It is, therefore, important that this integration is deliberate and well considered. Attempts have been made to integrate these two processes such as the development of U-Scrum (Singh, 2008), however these have not been widely adopted. Despite these attempts, there has been no formal attempt to develop a computer-based management system to aid software architects in integrating these two concepts (namely UX and Agile). This paper outlines the initial development of such a system; UX-Vision. It is hoped that it will act as a central hub which will provide information about both Agile as a development framework, and UX as a design methodology. It aims to provide resources for carrying out various UX activities, as well as a way of archiving the data collected from completing these activities. This data will be stored against a particular software development project, allowing the system to be utilised across an entire organisation. For example, by logging activities in the software development lifecycle, the system can act as a virtual assistant by providing recommendations for appropriate UX activities at pertinent stages in the project.

2. IMPLEMENTATION

The base of the system has been built using WordPress as a content management system. A custom theme was developed to control how the content is displayed using custom Cascading Style Sheets (CSS), Hypertext Markup Language 5 (HTML5), Hypertext Preprocessor (PHP), and jQuery in addition to those files included in the

WordPress platform. WordPress login credentials are used by administrators to manage the content within the system. A separate login system was implemented for 'front-end' users to access the site content, create projects, and upload activity data.

2.1 Workflow

The system has several potential workflows. The first involves the system being used as an educational resource for learning about UX design methods, as shown in Figure 1. A number of UX activities are listed within the system. For each activity, a guide has been developed to provide information about that activity; its history, what it is used for, how to carry out that activity, and how the

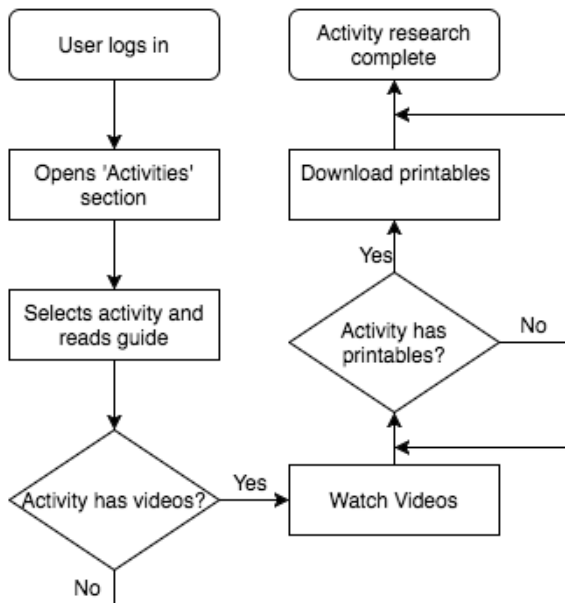


Figure 1: Flow diagram illustrating educational workflow of system

data obtained from the activity can be utilised within the Agile model for software development. Where appropriate, videos have also been created to illustrate the activity being carried out, as well as tutorials for any accompanying software. In preparation for carrying out the activities, 'printables' of consent forms, recording sheets, and other physical resources are provided.

The second workflow involves recording the data

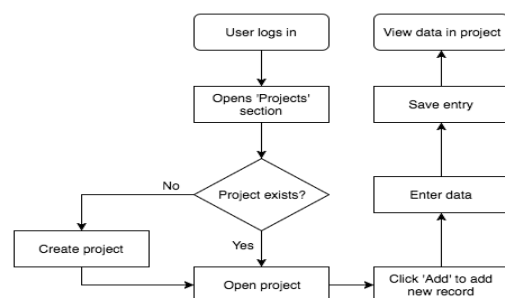


Figure 2: Flow diagram illustrating data entry workflow of system

collected from carrying out the various UX activities, as shown in Figure 2. These are recorded against a specific project, which can be opened if it exists, or created if it does not. These project sections act as an archive for all the UX activities carried out in the development of that project. Activities are displayed as a timeline, showing when each activity took place. The activity can then be opened to view the collected data. This data can take different forms; PDFs of documents, video recordings of interviews or usability tests, spreadsheets of numerical data, and so on. By displaying the activities as a timeline, any gaps in UX research for the project can be easily identified by either a user or a virtual agent.

2.2 User Interface

The user interface has been divided into four sections.

(1) Development process. This illustrates the Agile process being used via an interactive diagram. Clicking on the various sections of the diagram will cause it to expand and display more information about that stage of the development process. A screenshot of this can be seen in Figure 3.

(2) Activities. Many UX activities are listed along with links to their guide, videos, and 'printables' as appropriate.

(3) Software. The system provides links to pieces of external software which can be used either within Agile development, or to carry out some of the UX activities.

(4) Projects. This final section provides access to existing projects and the ability to set up new ones. Within each project, the UX activities which have been carried out can be viewed on a timeline. The data for each activity can also be

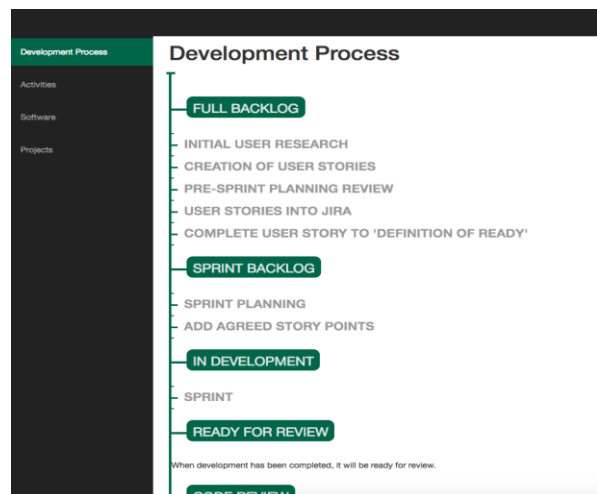


Figure 3: Screenshot of the Development Process viewed.

2.3 Data Model

WordPress' table structure was used for much of the content. Additional tables were added to hold the data of the front-end users (i.e. not the content managers), as well as the project information and result data. The relationships between these tables can be seen in Figure 4. As shown, a user can be involved in many projects, with projects being editable by many users. A project can contain many results, with each result only appearing on one project. A user can upload many results, with each result only being uploaded by a single user.

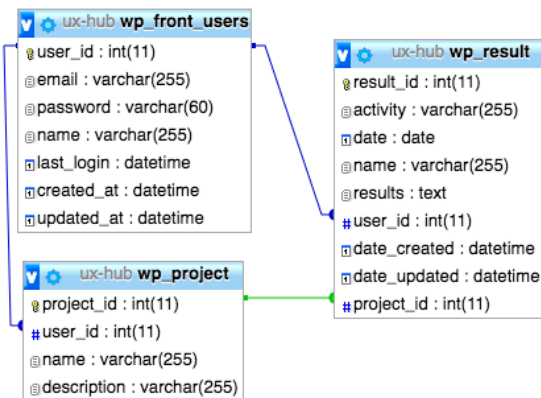


Figure 4: Image of the relational database schema

3. DISCUSSION AND FUTURE WORK

In an attempt to integrate the Agile development process with UX design methods, a platform has been designed which has attempted to document the general Agile process, provide information and resources to aid carrying out various UX activities, and can act as an archive for holding the results of carrying out these activities within a particular project. It is hoped that by providing this information and functionality in a central hub, it will be simpler for UX to be utilised within a company employing Agile software development processes, such as Scrum.

In the future, it is hoped that the system can be developed further to incorporate a guide and virtual assistant for when and how UX activities should be carried out. Logic could be used to examine what activities have been carried out so far, and a simple recommender engine could suggest as to what steps or UX activities could be carried out. This could be presented as a 'virtual assistant', guiding the user through the development process. There has been an increase in the use of virtual assistants in a variety of disciplines including healthcare (Teixeira et al., 2014), sales (Beldad, Hegner, & Hoppen, 2016), and aiding in research (Hasler, Tuchman, & Friedman, 2013).

Development of frameworks for the design of virtual assistants are being researched (Eisman, López, & Castro, 2012) and, in particular, how this could be applied with the Agile methodology (Biswas et al., 2012). While this research is still in an early stage, the progress being made seems promising.

As the UX activities within a project are also being documented, there is potential for further study into how the quantity and frequency of UX research activities being carried out within a software development project affects the overall final quality of the product. Thus, an analytics dashboard could detail which UX activities or combination of activities correlated with the most and least successful projects.

4. REFERENCES

- Beck, K., Cockburn, A., Jeffries, R., and Highsmith, J. Agile Manifesto. <http://www.agilemanifesto.org>. 2001. 12-4-2002.
- Beldad, A., Hegner, S. and Hoppen, J., 2016. The effect of virtual sales agent (VSA) gender–product gender congruence on product advice credibility, trust in VSA and online vendor, and purchase intention. *Computers in human behavior*, 60, pp.62-72.
- Biswas, P., Lin, R., Hanumanthgari, R. and Vojjala, S.B., 2012. Development of a Virtual Teaching Assistant System applying Agile methodology. In *American Society for Engineering Education*. American Society for Engineering Education.
- Eisman, E.M., López, V. and Castro, J.L., 2012. A framework for designing closed domain virtual assistants. *Expert Systems with Applications*, 39(3), pp.3135-3144.
- Gulliksen, J., Göransson, B., Boivie, I., Blomkvist, S., Persson, J. and Cajander, Å., 2003. Key principles for user-centred systems design. *Behaviour and Information Technology*, 22(6), pp.397-409.
- Hasler, B.S., Tuchman, P. and Friedman, D., 2013. Virtual research assistants: Replacing human interviewers by automated avatars in virtual worlds. *Computers in Human Behavior*, 29(4), pp.1608-1616.

- Hastie, S. and Wojewoda, S., 2015. Standish Group 2015 Chaos Report-Q&A with Jennifer Lynch. *Retrieved*, 1(15), p.2016.
- Larusdottir, M., Gulliksen, J. and Cajander, Å., 2017. A license to kill—Improving UCSD in Agile development. *Journal of Systems and Software*, 123, pp.214-222.
- Myers, B.A., 1998. A brief history of human-computer interaction technology. *interactions*, 5(2), pp.44-54.
- Schwaber, K., 1997. Scrum development process. In *Business Object Design and Implementation* (pp. 117-134). Springer London.
- Singh, M., 2008, August. U-SCRUM: An agile methodology for promoting usability. In *Agile, 2008. AGILE'08. Conference* (pp. 555-560). IEEE.
- Teixeira, A., Hämäläinen, A., Avelar, J., Almeida, N., Németh, G., Fegyó, T., Zainkó, C., Csapó, T., Tóth, B., Oliveira, A. and Dias, M.S., 2014. Speech-centric multimodal interaction for easy-to-access online services—a personal life assistant for the elderly. *Procedia computer science*, 27, pp.389-397.