Qualitative and Quantitative Methods in Libraries (QQML) 5: 79- 88, 2016

Developing the Web Portals of the German National Library of Science and Technology: Tools and Workflows Used

Sven Strobel

Competence Centre for Non-Textual Materials, German National Library of Science and Technology (TIB), Welfengarten 1B, 30167 Hannover, Germany

Abstract: Teams working on the development of software systems use certain tools and workflows of product and knowledge management. These tools and workflows help them plan, monitor and control the product at all stages of the product lifecycle as well as capture, develop and share project-related knowledge. This paper discusses the tools and workflows development teams of the German National Library of Science and Technology (TIB) use when developing the web portals. The use case illustrated here is a project, in which the interaction concept and screen design of the video portal of TIB (TIB|AV-Portal) had to be adapted to the new look and feel of the main portal (TIB-Portal). In the course of the project, the team gathered and structured preliminary requirements in the enterprise wiki, created an Axure user interface prototype, evaluated the prototype and specified the requirements using sprints from the Scrum framework, commissioned external screen designers and software developers to create a new screen design based on the prototype and implement the requirements, tested the implementations in a collaborative workflow and approved their deployment after debugging.

Keywords: product management; knowledge management; web portal; library; user interface prototyping; software engineering; sprint; Scrum; collaborative working; workflow

1. Introduction

Development teams of the German National Library of Science and Technology (TIB) work in small groups, which cover all the necessary competences to carry out their projects. TIB teams dealing with the development of the web portals normally require at least a conceptual developer, IT specialist, Usability expert, and project manager. However, the teams often cannot accomplish their projects

Received: 23.1.2016 Accepted: 21.3.2016 ISSN 2241-1925

without external service providers. This probably holds true for the whole library sector, especially when developing complex software.
This paper discusses the tools and workflows TIB teams use when developing the web portals. The use case illustrated here is a project, in which the video portal of TIB (TIB|AV-Portal[i]) had to be adapted to the new web site of the main portal (TIB-Portal[ii]) in terms of interaction concept and screen design. The TIB|AV-Portal was relaunched together with the main portal on 4 January 2016.
The team, which worked on the relaunch of the TIB|AV-Portal, involved four persons representing the four roles mentioned above. First of all, the team collected all available requirements (cf. section 2). The conceptual developer and Usability expert created a user interface prototype based on these requirements (cf. section 3). The team evaluated the prototype collaboratively using sprints from the product management framework Scrum and defined the requirements more precisely (cf. section 4). The team commissioned external screen designers to create a new screen design – including an HTML click dummy – based on the prototype. Having received the screen designers' click dummy, the team commissioned external software developers to implement the new interaction concept and screen design (cf. section 5). The team tested the implementations in a collaborative workflow and summarised the identified bugs in a test report. After debugging, the implementations were deployed in the production system (cf. section 6).

2. Collecting the Requirements

The team began by clarifying which components of the interaction concept and screen design of the TIB|AV-Portal had to be adapted to the new appearance of the TIB-Portal. The team compared the status quo of the TIB|AV-Portal with the user interface prototype and requirements for the TIB-Portal. In the enterprise wiki, it gathered requirements for the interaction concept such as key pages, user interface elements, functionalities, clicking patterns, and wording. It also gathered requirements for the screen design such as icon set, layout grid, typeface, font colour and font size of links, headlines and running text, usage and placing of logos, and colour values for buttons. With respect to the two portal concepts, the team pursued the goal to represent matching items in the same way and deviating items – if possible – in a similar way. Furthermore, the team created mind maps for the page structure of the portal and for specific use cases (cf. figure 1). The mind maps served as a model for the click paths of the user interface prototype.

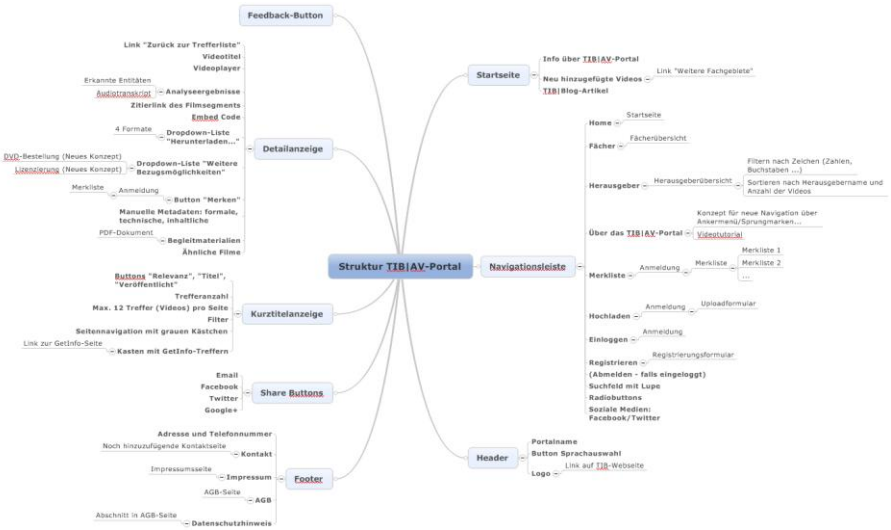

Figure 1 *Page Structure of the TIB|AV-Portal in a Mindjet Mind Map*

3. Creating the Prototype

The conceptual developer and Usability expert created a user interface prototype (click dummy) based on the collected requirements. The prototype allows to construct, evaluate and improve issues of the user interface, even before a running system exists (cf. Richter and Flückiger 2013: 52). The conceptual developer and Usability expert used Axure RP, which enables the creation of detailed and interactive prototypes. When creating the prototype, they focused on function rather than design. The prototype contained the general structure and rough screen layout, 47 pages, numerous links, information display, important operating elements, and use cases such as registration or video upload. Figure 2 shows the start page of the TIB|AV-Portal, as conceptualised in the prototype.

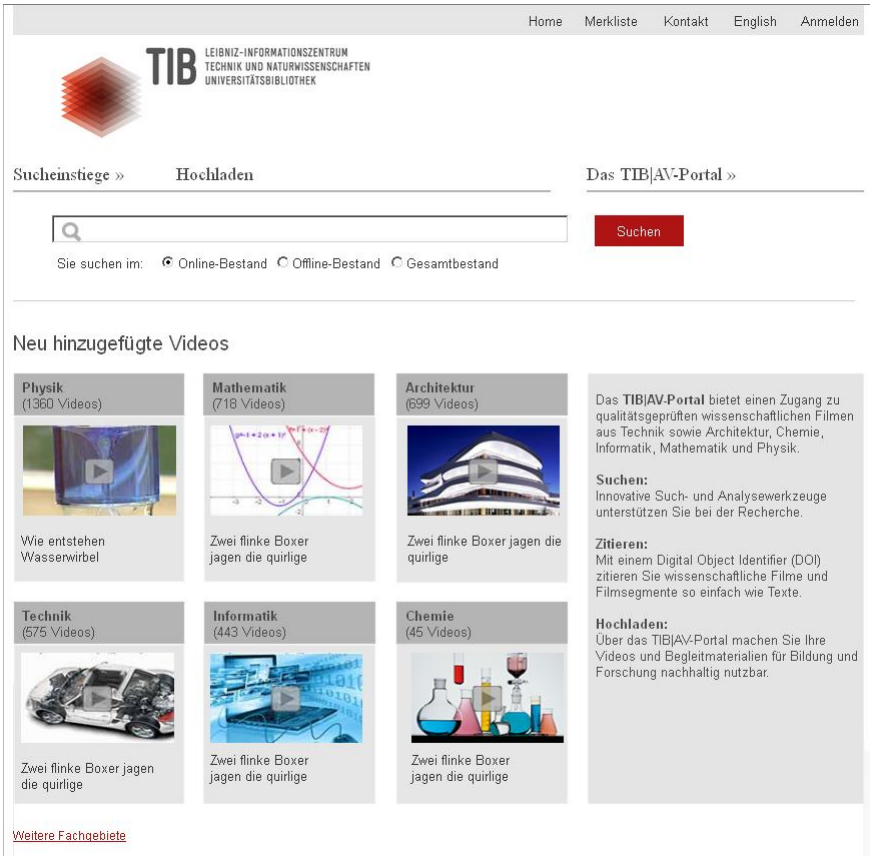

Figure 2 *Start Page of the TIB|AV-Portal (Axure Prototype)*

When specifying the requirements, the prototype can be used for the following purposes: illustrating range of function, clarifying functionality, identifying user interface elements, displaying navigation and interaction, estimating the realisation effort, clarifying the data exchange with other systems, and determining the provision of specific contents such as texts or images (cf. Richter and Flückiger 2013: 58).

The Usability expert checked if the user interface was easy to use and understand. She examined the following issues: Are there any obstacles for users? Is the navigation efficient? Do users find the desired information? Can users grasp the displayed information? Is the error handling simple? Do the site elements follow users' expectations? Do users get enough feedback from the system? etc.

The prototype was a determination of the requirements that had so far been collected. In the next step, the team had to evaluate the prototype and specify the requirements. The team did that in a collaborative manner using sprints.

4. Specifying the Requirements Using Sprints

Sprints originate from the agile product management framework Scrum (cf. Takeuchi and Nonaka 1986, Deemer et al. 2012, Schwaber and Sutherland 2013). At TIB, we use a special sprint concept with a workflow and roles that deviate from Scrum (cf. Strobel 2015). In other words, we rather work with the sprint idea than in the Scrum framework. According to our definition, a sprint is a short working cycle (2 - 4 weeks), in which a part of a concept is collaboratively and mainly virtually developed and approved. The overall concept is developed incrementally by the repeated execution of sprints. The sprints help to accelerate complex decision making within the team.

The conceptual developer had written 17 specifications describing header, footer, navigation, hit list, detailed view or use cases such as registration. Every specification was taken into a sprint and collaboratively worked on. The sprint workflow of TIB is illustrated in figure 3.

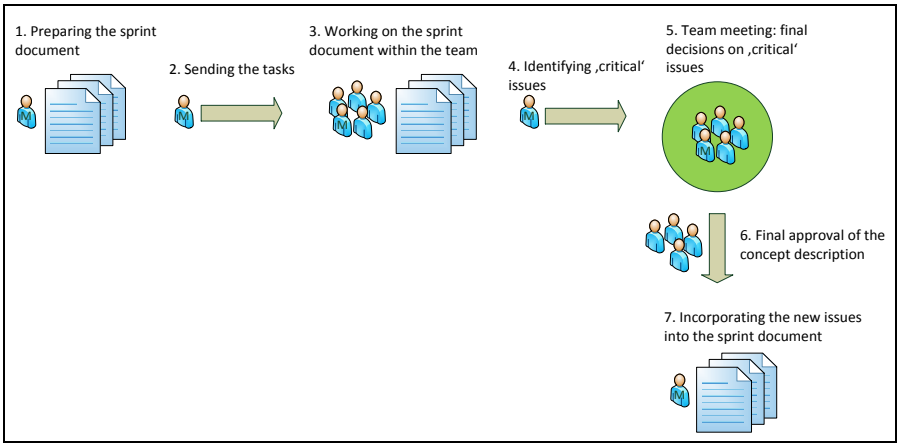

Figure 3 *Sprint Workflow of TIB*

The workflow can be summarised as follows: The sprint master prepared the sprint document in the enterprise wiki. At TIB, we use the wiki software Confluence by Atlassian. In the sprint document, the sprint master determined start and end of the sprint, defined the sprint goal, wrote a detailed description of the concept and assigned scheduled tasks to the sprint participants. The concept description contained the requirements, which were illustrated by screen shots of the user interface prototype.

The sprint master plans, steers and controls the sprint workflow.[iii] In this project, the conceptual developer took on the role of the sprint master. The sprint master sent the scheduled tasks to the participants via wiki function, which marked the beginning of the sprint. The participants worked off their tasks, commented on the concept description and suggested supplements and modifications. The

sprint master identified ‘critical’ issues in the sprint document about which decisions needed to be made and planned a team meeting. The team members made final decisions on the ‘critical’ issues at that meeting. After the final decisions had been made, the sprint participants approved the concept and the sprint master incorporated the new issues into it. As a result, the team members collaboratively worked out a consistent concept and agreed on it. The total set of requirements was incrementally produced by applying the sprint cycle several times.

5. Commissioning the Screen Design and Functional Implementation

The sprints enabled the team to specify the requirements and refine the user interface prototype. The team commissioned external screen designers at *pixelcreation*[iv] to create a new (responsive) screen design based on the prototype. The screen designers worked on positions and distances of interface elements, orientation of labels, grouping of elements, colours and contrasts, use of fonts, style and representation of icons and other graphical elements. Figure 4 illustrates the screen designers’ recommendation for the start page (compare with figure 2).

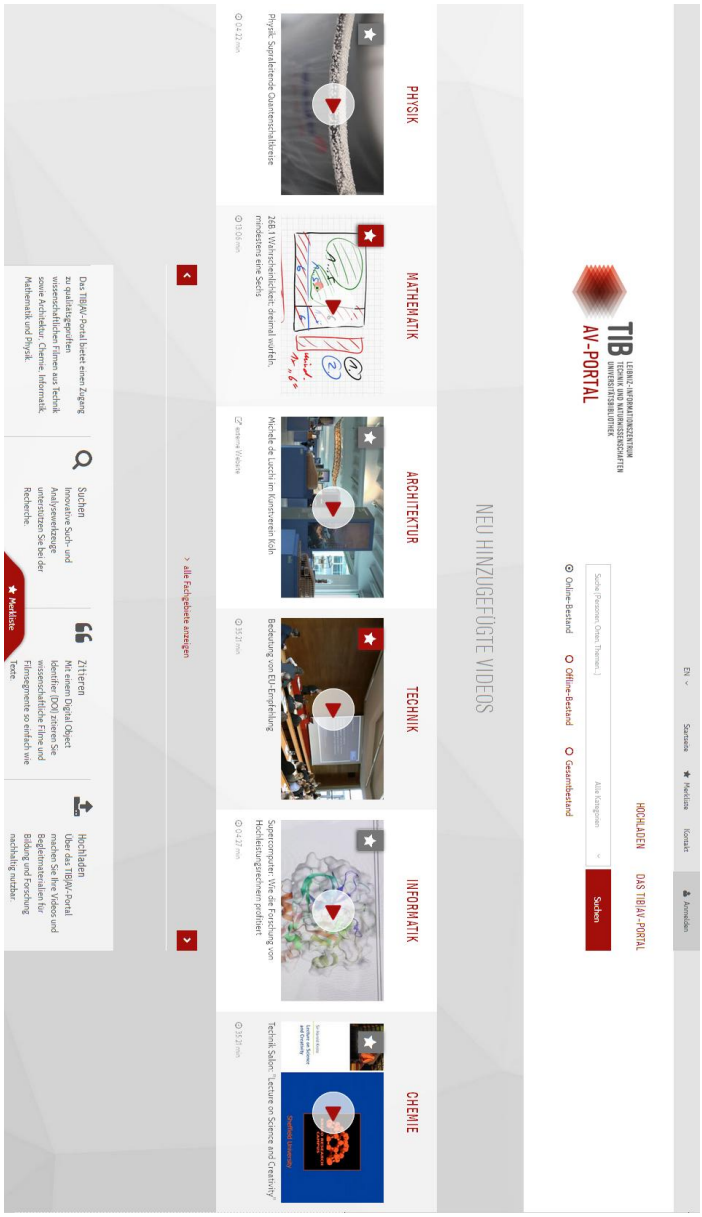

Figure 4 *Start Page of the TIB|AV-Portal (HTML click dummy)*

The screen designers supplied an HTML click dummy including CSS and JavaScript. The team commissioned software developers at *yovisto*[v] to implement the new screen design plus interaction concept defined by the

requirement specifications. After the software developers had implemented the requirements, the implementations had to be tested.

6. Testing, Debugging and Deploying

Tests are a measure of quality management for software development. The conceptual developer and IT specialist of the project team acted as test managers, who planned, organized and monitored the testing workflow. They grouped the test cases into two scenarios: *requirement specifications* – Are all requirements implemented and do they work?; and *free testing* without specific objectives. The test scenarios contained a total of 58 test cases. For every test case, the test managers created a wiki document, which lists the *title of the test case*, *tester*, *browser version*, *status*, *result*, *test goal*, *test steps*, and *test documentation*. Some of these properties (*title, tester, browser, status, result*) of the single documents were passed by wiki function to the testing main page so that the test managers obtained an overview of the whole testing process (cf. figure 5).

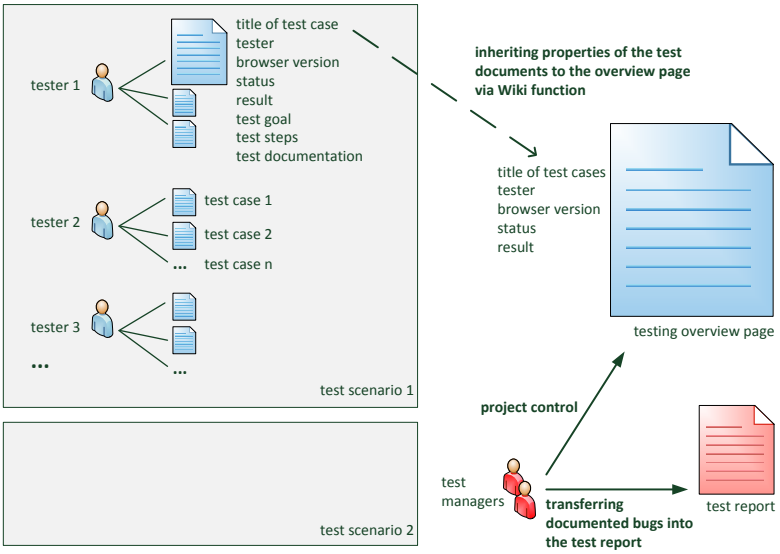

Figure 5 *Testing Workflow*

All issues were tested in the test system. The testing period was constrained to one week. The testing workflow involved 10 testers and 2 test managers. The testers were supposed to complete their test cases, which had been assigned to them by the test managers. The testers had to read and capture the test goal, execute the test steps, and document bugs with detailed descriptions and screen shots. Having finished the test case, they specified the status (*done*) and result (*passed* vs. *failed*) of the test case as well as the browser version they had used.

The test managers' task involved outlining the test concept, briefing the testers, monitoring the testing, reproducing reported bugs, summarising all reproducible bugs in a test report, and discussing the results with the external software developers. They used the testing main page in the wiki as their essential instrument to control the testing process. This page reflected which testers had to be reminded to carry out their tests and which test cases included documented bugs.

The external software developers received the test report for debugging. They fixed the defects in the test system. The test managers tested these issues again by going through the test report and eventually approved the implementations. The new interaction concept and screen design were deployed in the production system with the successful go-live of the TIB|AV-Portal on 4 January 2016.

7. Conclusion

The most important tool in this project was the enterprise wiki of TIB. The wiki helped the team to realise all phases of the project and, most importantly, to accomplish collaborative work packages. The workflows for sprints and testing were planned, steered and controlled by the workflow managers (sprint master, test managers) with great support from the wiki. Axure RP allowed the team to create a detailed and interactive user interface prototype. The prototype was the first representation of the collected requirements. By repeatedly evaluating the prototype, the team was able to continuously specify the requirements. The external screen designers and software developers not only received the textual requirement specifications but also the final prototype for purposes of illustration. All requirement specifications and screen shots of the prototype were archived in the wiki as part of the project documentation.

References

Deemer, P. et al., (2012). *The Scrum Primer: A Lightweight Guide to the Theory and Practice of Scrum*. Version 2.0, InfoQ.com.

Richter, M. and Flückiger, M., (2013). *Usability Engineering kompakt: Benutzbare Produkte gezielt entwickeln*. Springer, Berlin/Heidelberg.

Schwaber, K. and Sutherland, J., (2013). *The Scrum Guide™, the Definitive Guide to Scrum: The Rules of the Game*. scrum.org.

Strobel, S., (2015). Einsatz von Sprints in der Produktentwicklung der Technischen Informationsbibliothek. *BuB – Forum Bibliothek und Information*, 67, 713 – 715.

Takeuchi, H. and Nonaka, I., (1986). The New New Product Development Game. *Harvard Business Review*, 64, 137 – 146.

[i] av.tib.eu

[ii] tib.eu.

[iii] Sprint master is a special role of the TIB sprint workflow. The term 'sprint master' was inspired by 'scrum master' from the Scrum framework. However, note that sprint master is a totally different role.

[iv] pixelcreation.de/typo3-oxid-internetagentur-hannover
[v] yovisto.com/imprint