EAI Endorsed Transactions
on e-Learning

Research Article **EAI.EU**

ThesesDB – blended self-service and supervision of students' theses

Fabian Ball*,†, Thomas P. Hummel*,†, Andreas C. Sonnenbichler,†, Victoria-Anne Schweigert,†

Karlsruhe Institute of Technology, Kaiserstraße 12 76131 Karlsruhe, Germany

Abstract

e-learning can be seen as service creation process between students and supervisors. In order to automate and enhance thesis and seminar processes at our university, we aim to increase transparency, traceability, communication, and success for our students and lecturers. Therefore, we suggest standardized process templates for students' theses and their implementation with an IT support system. We start analyzing and designing processes for bachelor, master, and PhD theses as well as bachelor and master seminars and suggest process templates to cover these processes. To implement these standard processes, we design and implement an enhanced IT web application platform, ThesesDB. By providing a centralized view, we offer a self-service for students and supervisors in the process. Our approach is evaluated by the Computer System Usability and USE Questionnaire.

Received on 20 January 2017; accepted on 06 June 2017; published on 20 June 2017

Keywords: e-learning, mentoring, process, service enhancement, supervised student publication, tutoring, workflow

doi:10.4108/eai.20-6-2017.152746

1. Introduction

With the digitization of education, new concepts of learning and teaching are being developed. On the one hand, there are new digital courses and materials that allow to learn online at any time and from any place [e.g. 1]. On the other hand, there are traditional educational institutions with face-to-face supervision and lectures, sometimes even with compulsory attendance. And, finally, there are many combined approaches that extend online services with face-to-face offers [e.g. 2] or universities that enhance their portfolios with digital content, also called blended learning.

As a university having a focus on both, top research and high-quality teaching for our students (and teachers), our department follows closely the development on the e-learning field trying to apply new ideas in our environment. We see teaching as a special kind of service being provided for our students coming along with some specialties. The attribute "special" arises from the setting that our "client" in universities' teaching services are mainly students who are assessed by the service provider, e.g. by examinations. In more typical services, like a hair-stylist or a restaurant, the client assesses the service provider and not vice versa[1]. Seeing teaching and e-learning as services, it seems reasonable to apply (research) principles of service management to this kind of service, too. C. Grönroos is a very influencing researcher teaching the so-called "Nordic School" with focus on high-quality service creation and delivery and considering external consequences of decisions made by the service provider (cf. [3, 4]). Grönroos (e.g. in [5]) distinguishes three basic service types for every service a service provider can offer to its clients:

Core services are services that are basically the reason for a service provider to be in the market. For a hair-stylist this is styling hairs, for universities this is giving lectures, and so forth.

*Corresponding authors: ball@kit.edu, thomas.hummel@kit.edu
†All authors contributed equally

[1] We are aware that lecture assessments are a increasingly used instrument to establish the client (student) assessment of the service provider (lecturer), however, at least in Germany, the consequences of bad or even good assessments for a lecturer are very limited.

Enabling services are services which must be offered in order to enable the core services to be able to function, e.g. the paying services or offering examination for lectures.

Enhancing services are services enhancing the service experiencing of the clients providing advantages in competition for the service provider, e.g. offering a cup of coffee to the client at the coiffure.

Looking at university lectures with this kind of mindset, we realized, that a lot of e-learning activities are tailored towards the core services. Identifying this gap, we decided to improve the enabling and enhancing services in e-learning.

In this paper we describe an innovative approach how to improve the interaction between students and lecturers. We focus on the creation, support and supervision process for student theses, such as bachelor thesis, master thesis, PhD thesis and student seminars. Lecturers at our department are supervising quite a number of these types of works simultaneously and, despite their best efforts, sometimes lose track, what in detail has been going on in every specific thesis/seminar. Consequently, our research targets for this work are

- analyze and formalize the process for creating and supervising seminars, bachelor/master/PhD theses (on a general level)
- design, implement, and introduce an IT tool to support these processes. We named the IT tool *ThesesDB* (theses database).
- evaluate (or at least start the evaluation) of the ThesesDB.

To our knowledge, no such effort has been undertaken so far.

2. Related Work

A study of McFarland and Hamilton in 2005 [6] compares the performance of offline and online students. They conclude that no major difference can be found on a general level, but that (structured) online material can have positive effects on the success of students.

Johnson, Killion and Oomen [7] put forward success factors for online courses. These factors are design (target group), flexibility, contact (make contact to lecturers easy), student-student interaction, monetary support (for the online systems), and orientation (which we interpret as process-focus).

Alanazi and Abbod [8] did a general research on the needs for e-learning repository systems at Saudi Universities. They analyze different type of media (e.g. source materials, videos, audio) and how these materials can be connected/linked. However, apart from the collection and sharing process, they avoid to elaborate on processes that are necessary for such a repository system, like versioning, diffs, notification of further materials for students and teachers (e.g. during a course when material is added), lifetime analysis, archiving and so on.

Eybers and Giannakopoulos [9] do research on the engagement of students in an online environment and compare this to the face-to-face perspective. They conclude that "the teachers' roles, the students' needs, the administration must satisfy e-learning criteria providing a student centered collaborative approach which could lead to student satisfaction and thus get a more engaged student" [9, p.74]. We see this as a strong argument towards IT systems supporting collaborative and structured processes in e-learning environments; like our ThesesDB.

Caione et al. [10] analyze the effectiveness of e-learning environments related to their respective goal. They do this by an example in the agri-food sector of unstructured information being analyzed with the help of an ontology. The feedback analysis is being planned to be implemented in our work.

A virtual assistant to provide aid in e-learning in environments for students is tested by Harvey et al. [11]. They use rule-based systems for an avatar-based FAQ system for university services at a London university. Radhamani et al. [12] suggest a virtual lab for biotechnology students at the Amrita School in India. Following a standardized process (selecting an experiment, protocol standardization, virtualization, sketching of story board, and value platform), the authors analyze the effectiveness in supporting the students to increase their active learning process. The result was, that "virtual labs [...] ensured a better performance during evaluations" [12, p.145].

However, supervision, advising, and writing a thesis may be a special case, different from a typical subject-based e-learning situation. We focus on the structured and process-oriented way, on how this approach was conducted.

3. Key Features

We propose the platform ThesesDB that supports the creation process of students' work from the different views of authors, supervisors or administrators. Based on customizable processes (e.g. for bachelor, master, PhD theses or seminar papers), the platform stores all related information from announcements of open theses through official dates for presentations to the final archiving step.

Due to the integrated rights management and personalized views, each involved party can track the overall progress of the work and its own responsibilities and tasks. New processes can be created from scratch or extended by additional steps. Every step can point to additional material or it can require actions and activities from the users. When a process reaches a stage that requires attention of a specific role, notifications can be sent via e-mail and reminders appear in the users' overview sites.

In addition to explicitly modeled actions and activities in every process step, it is possible to upload attachments and notes. This allows to track meetings, add supplementary materials or keep track of relevant information for future reference.

External systems and services can be connected through custom code. For example, this can be used to file the final work with sources and references to a separate archive system.

4. Processes and Examples

Analyzing the typical papers created by students of our faculty (processes and their related participants), showed us high grades of similarities: The sequence of actions and participants are almost the same, with only PhD theses being a bit special compared to the other processes. As we want to support those processes by our IT tool, we chose to opt for customizable process templates supporting each of the above processes. We found three different process roles (responsibilities) reoccurring in all process templates: author, supervisor, and administration. It is possible to add new processes or roles for special cases, for example when a thesis is supervised jointly with a third party. Utilizing these templates, the whole process of allocation, editing, submitting feedback, archiving, and accounting will become more transparent to all process partners and enable each party to see the current process state at one glance.

As an example, we depict our current process template for the bachelor thesis (Figure 1) with the four activities *Announcement & Assignment*, *Editing*, *Presentation & Correction*, and *Finalization*. Each activity possibly consists of several (sub-)activities and/or atomic actions. These activities encapsulate atomic actions, e.g. the necessity to organize a presentation date. As process organization may differ between organizational units or universities, the described process could be modeled in another way. ThesesDB is capable of dealing with such requirements.

Here, the wording differs between UML and the concrete implementation. In ThesesDB, UML (sub-)activities are called *(sub-)process steps* and UML actions are split up into *activities* and *actions*. Details on that are provided in section 5.

The process of a bachelor thesis can be summarized as follows:

1. Announcement & Assignment: As a first step, the supervisor publishes the topic of the bachelor thesis. Interested students can apply.

2. Editing: The student (author) works on the thesis, during that time a short presentation will take place, organized by the administration. Irregular meetings between author and supervisor are not modeled explicitly.

3. Presentation & Correction: In this process step, the author uploads the final submission. The submission includes the thesis as PDF and LaTeX, presentation slides and used literature. After the supervisor corrected the work, he gives feedback to the author.

4. Finalization: Evaluation, archiving of materials and recording of results.

In Figure 2 we provide an example of the user interface depicting the sub-process "Delivery" and its assigned activities (we use the wording of section 5). We now explain the process and the functionality of ThesesDB by the example of a bachelor thesis:

A student, here named "Author Demo", applies for a bachelor thesis with the working title "Single-Source of Information and Workflow Support for Students' Work". The supervisor is named "Supervisor Demo", the responsible contact person in the administration office is named "Office Demo".

4.1. Announcement & Assignment

As a first step, the supervisor chooses one process template available. This template is then instantiated as a copy making it independent from its template. This is important as templates might change over time (e.g. an additional step is added) which must not influence processes already being worked on or, even worse, finished process instances. When instantiating a process, the supervisor is also able to customize the template. The ThesisDB is also connected to other IT systems of the university, e.g. the faculty's seminar application system enabling batch data import for seminars with several participating students.

Example: "Supervisor Demo" chooses the template "bachelor thesis", sets the topic "Single-Source of Information and Workflow Support for Students' Work" and adds a small description. The supervisor creates a PDF announcement with a 1-2 page description of the thesis and makes it available for the students, e.g. by publishing it on the faculty's web site. Next, interested students discover the newly available thesis, read the topic and description and apply for it. Side note: The application process is currently not implemented in the ThesesDB. However, there are no technical

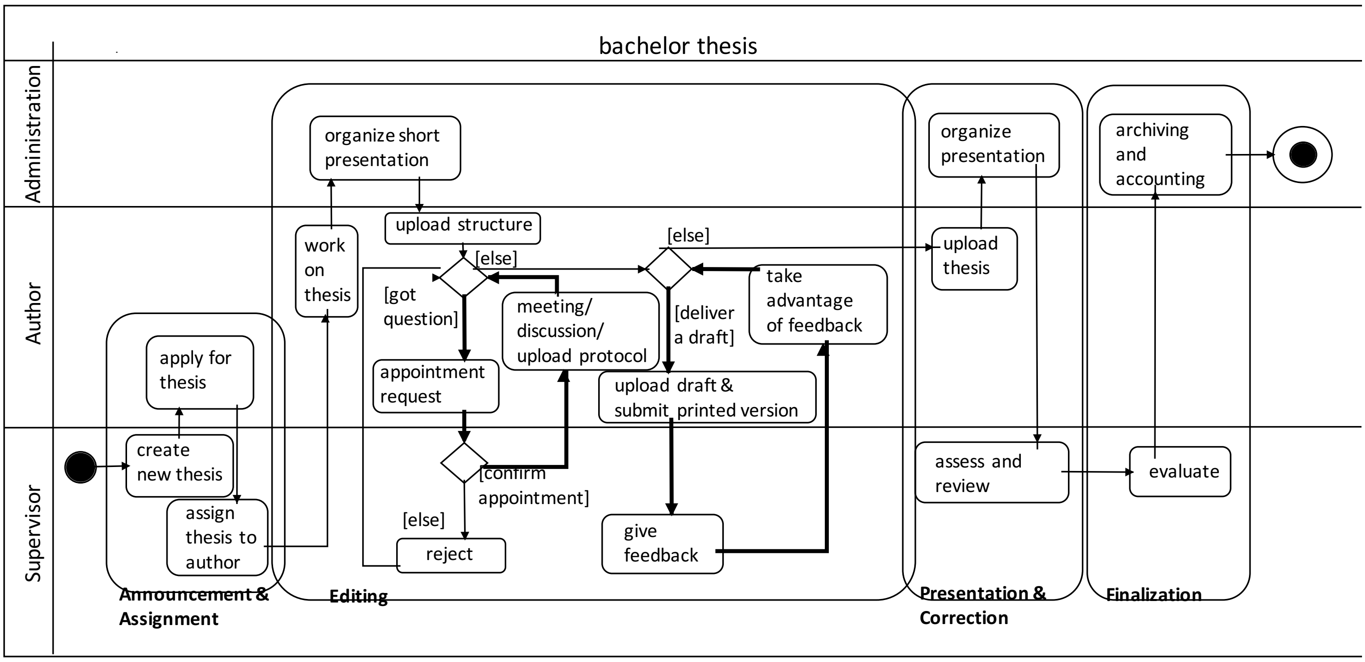


Figure 1. UML activity diagram for a bachelor thesis. The four major activities are described in the text. Feedback loops in bold.

limitations to extend ThesesDB with this functionality. When the student first logs in to the system with the personal university credentials, student information is copied from our central LDAP service. Then, the student "Author Demo" is assigned the author role for this process instance. As last step in this process part, the supervisor assigns "Office Demo" as contact person / responsible role for administrative purposes. As a result, only supervisor, author and assigned administrative contact person are able to see the details of the process instance.

4.2. Editing

This activity consists of the actions "Editing Details", "Short Presentation" (added as special action differently from the process template for this special process instance), "Elaboration", and "Delivery". If the author changes anything in the ThesesDB the supervisor gets a notification by e-mail.

Example: The student works on her topic. First she gives a short presentation. The event will be organized by the administration. Afterwards, the author uploads the structure of the thesis. As she got some questions she decides to ask for an appointment with her supervisor suggesting a specific date, time, and location utilizing ThesesDB. The supervisor confirms the request and the meeting takes place. The student uploads the meeting notes to the ThesesDB making them accessible for all participants. Of course, also the supervisor could have made the appointment or uploaded the notes.

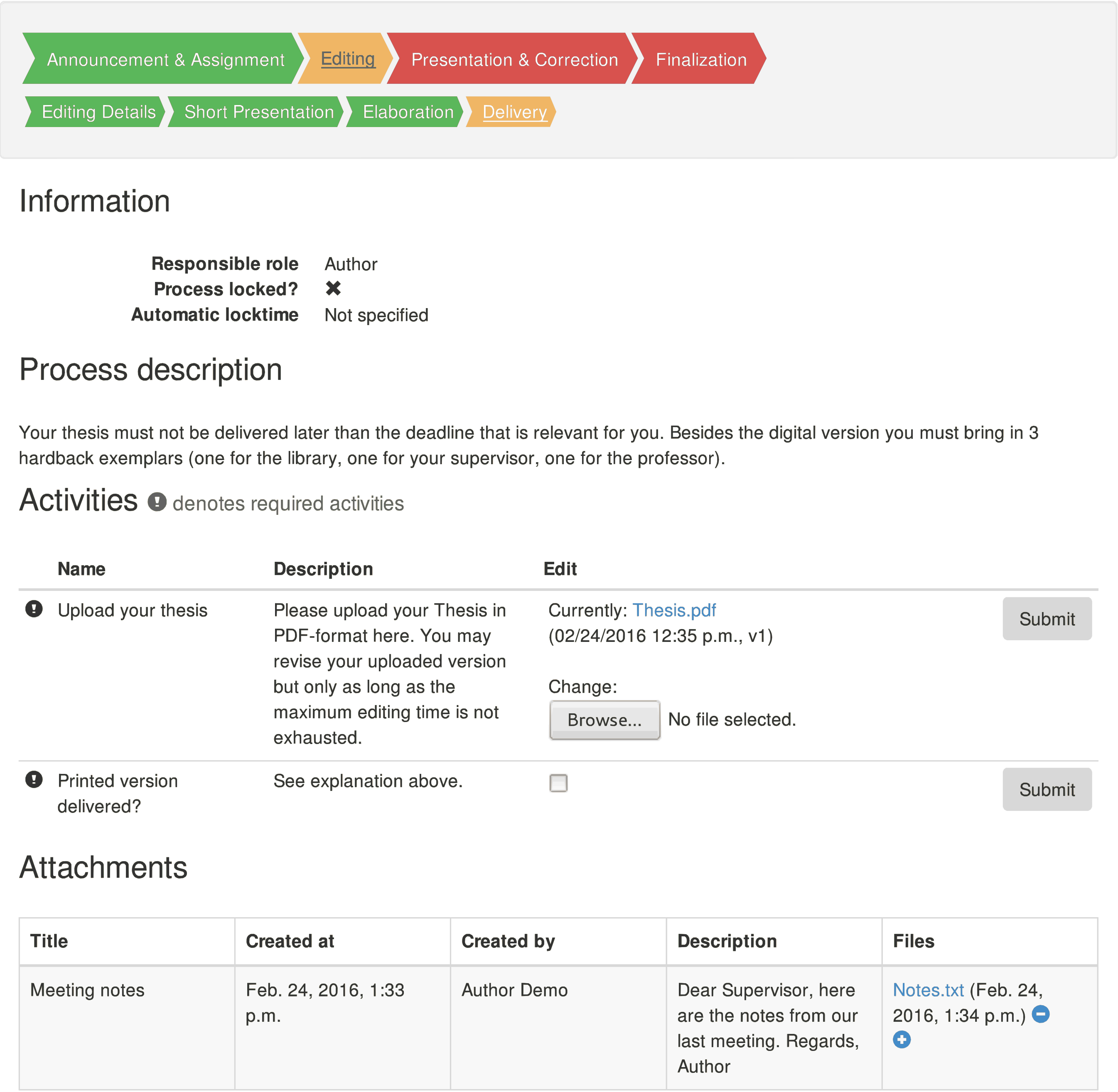
Figure 2 depicts an example of a part of the user interface in the process step "Editing" and subprocess "Delivery".

We see, that our author uploaded his first version of the thesis as a PDF. Attached are also some notes of the last meeting. The reminder (check box) to hand in a printed version is unchecked and remains as a to-do for the author. After the correction by the supervisor, the supervisor gives some feedback to the student. The student takes an advantage of the feedback and has some more time to finalized her work. If wished, the supervisor is able to set a date for the last possible time of submission. After this date the final submission is no longer possible.

4.3. Presentation & Correction

In this process step, the author uploads the final submission. The submission includes the thesis as PDF and LaTeX, presentation slides, and references. The administration organizes a presentation. After the supervisor corrected the work, he gives a feedback to the author.

Example: The author uploads the final version of her bachelor thesis and all needed attachments. By submitting the final version, the student has to upload the presentation slides and her references as BibTex, too.



Information

Responsible role	Author
Process locked?	✖
Automatic locktime	Not specified

Process description

Your thesis must not be delivered later than the deadline that is relevant for you. Besides the digital version you must bring in 3 hardback exemplars (one for the library, one for your supervisor, one for the professor).

Activities ❗ denotes required activities

	Name	Description	Edit	
❗	Upload your thesis	Please upload your Thesis in PDF-format here. You may revise your uploaded version but only as long as the maximum editing time is not exhausted.	Currently: Thesis.pdf (02/24/2016 12:35 p.m., v1) Change: Browse... No file selected.	Submit
❗	Printed version delivered?	See explanation above.	□	Submit

Attachments

Title	Created at	Created by	Description	Files
Meeting notes	Feb. 24, 2016, 1:33 p.m.	Author Demo	Dear Supervisor, here are the notes from our last meeting. Regards, Author	Notes.txt (Feb. 24, 2016, 1:34 p.m.) ⊖ ⊕

Figure 2. The subprocess "Delivery" from the supervisor's point of view. Only the main part of the interface is shown, the page header (including global navigation bar and thesis title), sidebar on the right (including links to other views and information on the thesis), and the form to add new attachments at the bottom are omitted.

4.4. Finalization

The last process step supports the evaluation, the archiving of the work and the used literature, and the accounting of the results. To support the archiving, we extended the ThesisDB with an interface to a literature database of our institute enabling the supervisor to upload the used files (plus references) to our archive system. This avoids failures made by students and reduces the editing time of the supervisors.

Example: The bachelor thesis and the used literature are archived and the student gets a result (mark). After uploading the thesis and all corresponding materials, the supervisor controls the documents and uses the interface to the archive system (literature database, LitDB) to archive the documents. The administration accounts the results. If wished, author and supervisor meet again for a final feedback meeting. The whole process is completed.

5. Implementation

This section describes the translation of the key features (see section 3) into software, from the abstract concept down to some implementation details. In order to stay flexible in designing our processes (see section 4), we divide the different specialized components into more general elements. These are outlined in the following as well.

Furthermore, we distinguish between design time and runtime of a process, well comparable with classes and objects in object-oriented programming. Each time the start of a process is triggered (e.g. a new bachelor thesis is made available), an instance of a previously designed process is generated.

Instances are linked to the "template" process. However, changes (e.g. due to process redesign) do not affect the instances. This is an important feature as theses finished in the past or currently running must not be altered.

5.1. Design time

We reduced our process model to the three building blocks *process steps*, *activities* and *actions*. As an overall structure we chose to model business processes in a tree-like manner:

- A unique *root process step* defines the process. Different business processes have different root processes.
- Each process step that is not a root process has a parent process step and an ordinal number which defines the ordering of process steps having the same parent. Such process steps are called *sub process steps*.
- *Activities* and *actions* also have process steps as parent. *Activities* are also ordered. These two elements represent the "dynamic" part of our process model whereas the process steps can be seen as "static" part.

By "dynamic" we mean points of user interaction with the system. *Activity* and *action* are distinguished as follows:

- An activity is a predefined and reusable building block that requires a user to perform some simple task. These tasks can be e.g. entering a date/text/url, uploading a file or simply ticking a checkbox to signal that the manual task related to the activity is done. The semantic of an activity is given by its descriptive text which is assessed by the process designer.
- An action is much more complex than an activity and covers a special task that cannot be accomplished by an activity or a combination of several activities. Other than an activity, an action must be explicitly implemented by a programmer to perform this special task. An example is given in section 5.3.

To complete the business process definition, we introduce special (user-)roles that are attached to process steps: On the one hand, there is a visibility relation that permits restricting visibility to only those users which have an appropriate role (not limited to one role). On the other hand, there is a responsibility relation that assigns exactly one role to be responsible for this process step.

5.2. Runtime

When a process is initiated, copies of the above defined elements are made and encapsulated within a "thesis" instance. This data model holds all meta-information (name, start/end dates, . . .) that are needed at runtime and have nothing to do with the abstract business process definition. The instance elements' names are then prefixed with *thesis* (e.g. *thesis process step*). To assign to a user in a thesis, the user gets a role for the thesis. The user then can see all process steps visible for this role and can interact via activities and actions.

Furthermore, it is possible for every involved user to add *attachments* to a *thesis process step*. This allows the exchange of documents (e.g. meeting notes) that are not directly part of the business process itself. To prevent later changes (e.g. after some deadline due), *thesis process steps* can be locked manually or automatically if some point in time is reached. Locked process steps are read-only. To keep track of changes, users can subscribe to several events which allows them to be notified via e-mail.

5.3. Implementation Details

We implemented our process model with the *Django web framework*[2] which has a powerful object-relational mapper (ORM) that allows quick modeling of structural data and rapid development overall. The design and runtime components are decoupled by using separate modules (called *apps* in the Django domain). The user interface is mainly written in plain *HTML* which fits best with Django's template engine and makes debugging easy.

[2] https://www.djangoproject.com/ (accessed February 2016)

As an example for an action, we implemented the submission procedure of a student's thesis: The student uploads the thesis' PDF file and the LaTeX source files. The related bibtex file is parsed and the student can upload the cited literature. The supervisor reviews the uploads and afterwards triggers the archiving process. The archiving process is done in a completely separate system and therefore includes invoking multiple web service calls which makes up the increased complexity that disallows the use of "simple" activities.

5.4. A Word on Workflow Management Systems

When we started working on ThesesDB, no one had in mind where the way would lead us. The main purpose was to create a light-weight tool to support everyone involved (but mainly students) in theses at our local chair. Therefore, nobody came up with the idea to have processes designed and executed in a specialized environment, mostly known as workflow management system. The Workflow Managament Coalition (WfMC) presented a reference architecture [13] over 20 years ago. Few years later, they issued another report [14] where many workflow patterns are described in detail. Picking up on these, van der Aalst et al. [15] came up with their own definitions and refined them over the years, also fully available to the public[3].

A workflow management system (WfMS) contains a workflow engine that executes instances of modeled process (the workflows) and allows monitoring the current state. Furthermore, workflow design/definition and execution are decoupled, a user and role model exists and a user interface permits interaction with the system (cf. [13, p. 11]).

These patterns provide a powerful tool set to model control and data flow explicitly and many available[4] WfMSs implement at least the most important (splitting and merging control flow) of those patterns.

All this sounds very promising for our ThesesDB, but after reviewing some WfMSs and comparing the "basic control flow patterns" and others with the capabilities of ThesesDB, we concluded *not* to include a WfMS into our system for several reasons:

- On the one hand, processes in ThesesDB are by definition sequential and do not allow branching and hence "parallel" execution. On the other hand are the emerging processes relatively simple and only few roles are involved therein. Furthermore, the whole process of a thesis is very linear (cf. Figure 1) and process steps involving e.g. a feedback loop do not necessarily require such a strict equivalent in the modeled process.
- Workflow management systems are intended to cover a fairly big amount of running instances (e.g. think of the management of customer inquiries at a large company). Managing students' theses is a comparably (very) small scale.
- The intention of ThesesDB is not to "industrialize" the writing of theses but to support students with their work, inform them how the overall process looks like and which steps have to be taken (now and in the future).
- The process of writing and supervising a thesis is mostly human work and only few points of automation exist.
- Using a WfMS in an (existing) software system is not trivial. We see the main reason for that concern mostly in the need to build the software *around* the WfMS with all its data structures, subsystems and components rather than simply including it side by side.

The mentioned arguments *against* using a workflow management system for ThesesDB are not meant to be against such systems in general. They allow for standardized and controlled operations, especially when many different parties have to contribute for the success of a process. And all that in an at least medium sized environment.

6. Evaluation

In order to evaluate the usability of the proposed system, we conducted a study with 5 participants of our bachelor seminar in the winter term 2015/2016. The seminar takes place in the field of economics and computer science. We used self-reported metrics, as explained by Tullis and Albert [16, pp.121-162] and selected the Computer System Usability Questionnaire (CSUQ) [17] as well as the USE Questionnaire [18].

The CSUQ measures *Information Quality*, *System Usefulness*, *Interface Quality* as well as *Overall Satisfaction*. The USE questionnaire covers similar dimensions with *Ease of Use*, *Ease of Learning*, *Usefulness* and *User Satisfaction*. Both instruments should lead to similar results.

The users rated their agreement on both scales on a seven-point Likert scale from *strongly agree* to *strongly disagree*. The score for the particular dimensions is calculated by the mean over the respective items and ranges from 0 to 1, where 1 represents full agreement.

As shown in Table 1, the reactions are mainly positive, since dimensions range between 0.65 and 0.86. In the free text answers students mainly criticize technical difficulties at the beginning but they emphasize the good structure and helpful guidelines provided by the system. Figure 3 visualizes both metrics as radar plots.

[3] http://www.workflowpatterns.com/ (accessed January 17)

[4] See e.g. http://www.workflowpatterns.com/evaluations/

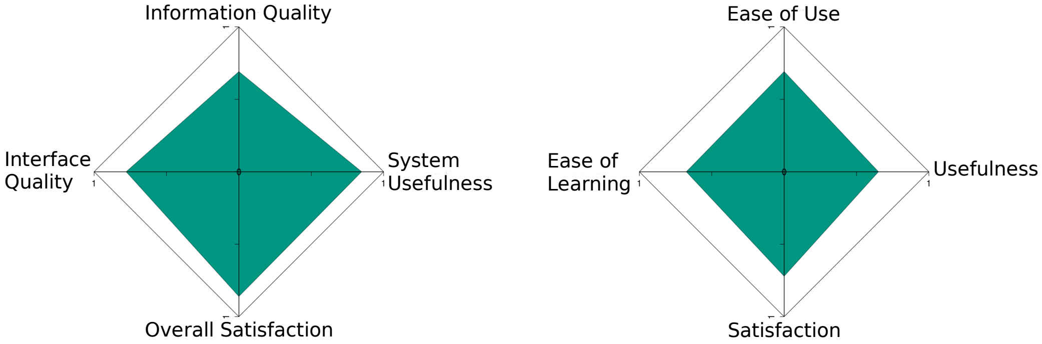


Figure 3. Radar Plots of the Evaluation Scores. Left: CSUQ, Right: USE

CSUQ	System Usefulness	Information Quality	Interface Quality	Overall Satisfaction
	0.85	0.69	0.78	0.86
USE	Usefulness	Ease of Use	Ease of Learning	Satisfaction
	0.65	0.69	0.68	0.72

Table 1. Scores of the CSUQ (top) and USE (bottom) scales

A time saver for supervisors is that students upload their (digital) bibliography in a structured way. Students request less organizational help than before, because they are pointed to relevant information in the system.

Multiple supervisors can access that data and do not have to maintain a local file system structure for each student or work. They see the student's progress at a glance and do not have to keep track of it on their own.

During discussions, members of the e-education community criticize missing auditing features. In certain environments, it is requested to keep track of the time it takes for an employee to complete a task. With the help of that, one can compare the efficiency of supervision of different research groups or identify problems in the processes. [see publications on analytics information systems in education, e.g. 19]

7. Conclusion and Future Work

In this work we treat the university as a service provider for its students. In particular, we look at the business process that represents the creation and supervision of students' theses or term papers.

Therefore, we propose a system that supports the creation of process templates during design time and the management of a concrete work in runtime. It provides information for supervisors, authors, and administrators and can be extended to work with additional tools or services.

Future extensions could include better integration with existing systems such as libraries or other repositories. Meeting dates could be exported to personal calendars or presentation dates could be published on a website, to name only few examples. Application handling and the selection of candidates could be integrated as well. Advanced metrics would provide better insights in the efficiency of students and staff in order to improve the processes or react to (predicted) problems. Furthermore, the use of a graphical process design would enhance the functionality and the ease-of-use even more.

References

[1] Kay, J., Reimann, P., Diebold, E. and Kummerfeld, B. (2013) MOOCs: So Many Learners, So Much Potential .... *IEEE Intelligent Systems* **28**(3): 70–77. doi:10.1109/MIS.2013.66.

[2] Murphy, W.M. (2011) From E-Mentoring to Blended Mentoring: Increasing Students' Developmental Initiation and Mentors' Satisfaction. *Academy of Management Learning & Education* **10**(4): 606–622. doi:10.5465/amle.2010.0090.

[3] Berry, L.L. and Parasuraman, A. (1993) Building a new academic field – The case of services marketing. *Journal of Retailing* **69**(1): 13–60. doi:10.1016/S0022-4359(05)80003-X.

[4] Gummesson, E., Lehtinen, U. and Grönroos, C. (1997) Comment on nordic perspectives on relationship marketing. *European Journal of Marketing* **31**(1-2): 10–17.

[5] Grönroos, C. (2007) *Service Management and Marketing Third Edition: Customer Management in Service Competition* (Chichester, West Sussex, England; Hoboken, NJ: John Wiley & Sons), 3rd ed.

[6] McFarland, D. and Hamilton, D. (2005) Factors Affecting Student Performance and Satisfaction: Online versus Traditional Course Delivery. *Journal of Computer Information Systems* **46**(2): 25–32. doi:10.1080/08874417.2006.11645880.

[7] Johnston, J., Killion, J. and Oomen, J. (2005) Student Satisfaction in the Virtual Classroom. *Internet Journal of*

Allied Health Sciences and Practice **3**(2): 1–7. URL http://nsuworks.nova.edu/ijahsp/vol3/iss2/6. Article 6.

[8] Alanazi, A. and Abbod, M. (2014) E-Learning Repository System for Sharing Learning Resources Among Saudi Universities. In Vincenti, G., Bucciero, A. and de Carvalho, C.V. [eds.] *E-Learning, E-Education, and Online Training*, no. 138 in Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering (Springer International Publishing), 10–18. doi:10.1007/978-3-319-13293-8_2.

[9] Eybers, S. and Giannakopoulos, A.P. (2015) Student Engagement in an Online Environment: Are We Trying to Mimic Contact Education? A South African Perspective. In Vincenti, G., Bucciero, A. and de Carvalho, C.V. [eds.] *E-Learning, E-Education, and Online Training*, no. 160 in Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering (Springer International Publishing), 68–75. doi:10.1007/978-3-319-28883-3_9.

[10] Caione, A., Guido, A.L., Paiano, R., Pandurino, A. and Pasanisi, S. (2015) E-learning Project Assessment Using Learners' Topic in Social Media. In Vincenti, G., Bucciero, A. and de Carvalho, C.V. [eds.] *E-Learning, E-Education, and Online Training*, no. 160 in Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering (Springer International Publishing), 92–100. doi:10.1007/978-3-319-28883-3_12.

[11] Harvey, P.H., Currie, E., Daryanani, P. and Augusto, J.C. (2015) Enhancing Student Support with a Virtual Assistant. In Vincenti, G., Bucciero, A. and de Carvalho, C.V. [eds.] *E-Learning, E-Education, and Online Training*, no. 160 in Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering (Springer International Publishing), 101–109. doi:10.1007/978-3-319-28883-3_13.

[12] Radhamani, R., Sasidharakurup, H., Sujatha, G., Nair, B., Achuthan, K. and Diwakar, S. (2014) Virtual Labs Improve Student's Performance in a Classroom. In Vincenti, G., Bucciero, A. and de Carvalho, C.V. [eds.] *E-Learning, E-Education, and Online Training*, no. 138 in Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering (Springer International Publishing), 138–146. doi:10.1007/978-3-319-13293-8_17.

[13] Workflow Management Coalition (1994) *The Workflow Reference Model*. Tech. Rep. TC00-1003, Workflow Management Coalition. http://www.wfmc.org/standards/docs/TC00-1003_10_1994.pdf.

[14] Workflow Management Coalition (1999) *Terminology & Glossary*. Tech. Rep. WFMC-TC-1011, Workflow Management Coalition. http://www.wfmc.org/standards/docs/TC-1011_term_glossary_v3.pdf.

[15] van der Aalst, W.M., Ter Hofstede, A.H., Kiepuszewski, B. and Barros, A.P. (2003) Workflow Patterns. *Distributed and parallel databases* **14**(1): 5–51. doi:10.1023/A:1022883727209.

[16] Tullis, T. and Albert, B. (2013) *Measuring the User Experience: Collecting, Analyzing, and Presenting Usability Metrics* (Amsterdam; Boston: Elsevier/Morgan Kaufmann), 2nd ed.

[17] Lewis, J.R. (1995) IBM computer usability satisfaction questionnaires: Psychometric evaluation and instructions for use. *International Journal of Human-Computer Interaction* **7**(1): 57–78. doi:10.1080/10447319509526110.

[18] Lund, A.M. (2001) Measuring Usability with the USE Questionnaire. *Usability Interface* **8**(2): 3–6.

[19] Guitart, I. and Conesa, J. (2015) Analytic Information Systems in the Context of Higher Education: Expectations, Reality and Trends. In *2015 International Conference on Intelligent Networking and Collaborative Systems*: 294–300. doi:10.1109/INCoS.2015.71.