

Insert here your thesis' task.

CZECH TECHNICAL UNIVERSITY IN PRAGUE
FACULTY OF INFORMATION TECHNOLOGY
DEPARTMENT OF SOFTWARE ENGINEERING



Master's thesis

BOZP portal - eLearning module

Bc. Martin Náhloušký

Supervisor: Ing. Jiří Chludil

12th January 2016

Acknowledgements

I would like to express my gratitude to my supervisor Ing. Jiří Chludil for continuous support, the useful comments, remarks and guidance while writing this master thesis. Furthermore I would like to thank my friend Silvr McFly, who has been a constant support of encouragement and enthusiasm. I would like to thank Cody Rhoads who personally contributed to this thesis, helping me with text correction. His family has been also a huge support while I was learning English. My sincere thanks also goes to Mario Chacón Rivas. M.Sc. for supervising my thesis during my study at Instituto Tecnológico de Costa Rica. And finally I would like to thank my family, who have supported me throughout my entire life and study.

Declaration

I hereby declare that the presented thesis is my own work and that I have cited all sources of information in accordance with the Guideline for adhering to ethical principles when elaborating an academic final thesis.

I acknowledge that my thesis is subject to the rights and obligations stipulated by the Act No. 121/2000 Coll., the Copyright Act, as amended, in particular that the Czech Technical University in Prague has the right to conclude a license agreement on the utilization of this thesis as school work under the provisions of Article 60(1) of the Act.

In Prague on 12th January 2016

.....

Czech Technical University in Prague

Faculty of Information Technology

© 2016 Martin Náhlovský). All rights reserved.

This thesis is school work as defined by Copyright Act of the Czech Republic. It has been submitted at Czech Technical University in Prague, Faculty of Information Technology. The thesis is protected by the Copyright Act and its usage without author's permission is prohibited (with exceptions defined by the Copyright Act).

Citation of this thesis

Náhlovský), Martin. *BOZP portal - eLearning module*. Master's thesis. Czech Technical University in Prague, Faculty of Information Technology, 2016.

Abstrakt

Cílem této práce je analyzovat, navrhnout a realizovat náhradu za současný eLearning modul BOZP portálu s pomocí integrace Moodle platformy. Studie zahrnuje detailní analýzu funkcí Moodle, které jsou nezbytné pro vytváření testů, export dat a možnost přihlásit se pomocí Shibboleth. Ukázalo se, že je potřeba vytvořit vlastní webservisu pro možnost spravování eLearningu přes Moodle. V důsledku toho byl rozšířen BOZP portál o funkce pro poskytování rozhraní na vzdálenou komunikaci.

Klíčová slova BOZP, BOZP portál, eLearning, Shibboleth

Abstract

The objective of this thesis is to analyze, design and develop a replacement for the existing eLearning module in BOZP portal by integrating Moodle platform. The study covers detailed analysis of Moodle features that are necessary to provide test creation, export data and ability to sign in with Shibboleth. It has been shown that custom web service has to be developed to manage the learning process via Moodle. Consequently, BOZP portal was extended by functionalities to provide interface for remote communication.

Keywords OSH, OHS, BOZP portal, eLearning, Shibboleth

Contents

Introduction	1
1 Task description	3
1.1 Analyze the existing solution of the eLearning form of training, identify its drawbacks and get the latest requirements from the OSH workers	3
1.2 Analyze Moodle platform and verify its usability to the needs of BOZP portal	3
1.3 Design an integration of Moodle into BOZP portal for training tests, add the following functionalities	4
1.4 Implement changes into BOZP portal	4
1.5 Perform automated and user testing of implementations	5
2 Analysis	7
2.1 Current form of eLearning	7
2.2 Trade study	8
2.3 Moodle	10
2.4 New solution proposal	13
2.5 Requirements	14
2.6 Technologies	16
3 Design	19
3.1 User roles	19
3.2 Use cases	20
3.3 Moodle integration	25
3.4 Mockups	27
3.5 KOSapi integration	31
3.6 Architecture	32
4 Implementation	35

4.1	BOZP portal	35
4.2	Moodle configuration	36
4.3	Moodle integration	38
4.4	Web service	38
4.5	Shibboleth	39
5	Testing	41
5.1	User testing	41
5.2	Automated testing	46
	Conclusion	49
	Evaluation of completion performance	50
	Future work	51
	Bibliography	53
	A Acronyms	55
	B Mockups	57
	C Scenarios for testing	61
C.1	Scenario 1 - Create an event of eLearning type - regular term . .	61
C.2	Scenario 2 - Sign in to the event and finish a quiz in Moodle . .	61
C.3	Scenario 3 - Review received grades and create an attendance list	62
C.4	Scenario 4 - Grant training to the students who has signed on the attendance list	62
C.5	Scenario 5 - Review your result of the finished training	63
C.6	Scenario 6 - Create an event of eLearning type - resit	63
	D User Manual	65
D.1	Adding an eLearning event	65
D.2	Managing events	66
D.3	Generating an attendance list	67
D.4	Granting a training	67
D.5	Signing in to an eLearning event	68
D.6	Setting up Moodle in BOZP portal	68
D.7	Creating a template in Moodle	69
	E Contents of enclosed CD	73

List of Figures

2.1	CourseSites demo - adding a new question to quiz	9
2.2	eFront demo - course dashboard	10
2.3	Business process model	14
3.1	Use case diagram relationships	20
3.2	Create event use case diagram	21
3.3	Sign in to event use case diagram	22
3.4	Acquire results data use case diagram	23
3.5	Event list mockup	28
3.6	Create event mockup	30
3.7	Event detail mockup	31
3.8	Database scheme	34
B.1	Homepage mockup	57
B.2	Moodle settings mockup	58
B.3	Send e-mail notification mockup	59
D.1	Create event form	66
D.2	Event list	67
D.3	Event detail	68
D.4	Available events	69
D.5	Settings from for Moodle connection	69
D.6	Course's dashboard in Moodle	70
D.7	Adding quiz in Moodle	71
D.8	Question manager of quiz in Moodle	72

List of Tables

3.1	Create event form fields	29
5.1	Description of tasks for user the testing	43

Introduction

Nowadays systems are very demanding on user experience, stability and availability. People have mobile devices at their fingertips whether it is using a smart phone or a tablet. Immediate access whenever and wherever is required. BOZP portal is considered falling within this category because of its use by hundreds of students. It is a very challenging project, which will examine a programmer's knowledge and improve his skill in many ways. Specifically focus of this project will be on the eLearning part, which will have a demand on getting to know a new system and accordingly develop suitable solution for its mutual integration.

The beginning of this thesis will be discussing the variety of possible solutions with help of different available systems in the same area of scope. The analysis will put bases for the following design. This will be resolved asking three main questions; how both systems will work together, data export and user management. The next step—implementation—is the key point as system installation will be performed as well the development of the necessary parts of the system. In the final part of the thesis, the newly developed functions will be verified by users as well as automated testing with unit and integration tests.

BOZP portal

BOZP portal is an online application designed to facilitate the managing and reporting procedures of Occupational safety and health (OSH) program at the Czech Technical University (CTU) in Prague. It is used by employees and also students. The portal is being developed to replace the original paper form, which was difficult to handle within the time efficiency and archive in order to later retrieval. This project covers complete design, build of website and maintenance. It is mainly developed within the student's diploma projects.

Task description

The content of this thesis is defined by tasks which are also the most important coordinators for the following work. Below is described each one of the given tasks.

1.1 Analyze the existing solution of the eLearning form of training, identify its drawbacks and get the latest requirements from the OSH workers

At the very first version of BOZP portal, the project has begun to start with an implementation of its own eLearning system. Former member of the BOZP portal team—Lukas Jeshke [1]—created a basic training system that was sufficient for the time being. After finishing the entire system, the eLearning module was not further developed and as time was passing the system was getting old and was not usable for the current needs. All the current functionalities have to be now summarized and taken for revision in order to design a better solution that will suit new demands of nowadays technologies. There is a need to be taken into account all available systems that are used at school and design such systems enabling convenient development. New requirements will be obtained by the OSH worker Ing. Jiří Chludil.

1.2 Analyze Moodle platform and verify its usability to the needs of BOZP portal

Moodle is a web-based application that provides online courses. It has been used at CTU for couple years and it is showing itself as a fitting system for school activities. Although Moodle is a very developed software consisting of many modules, it has not been used for BOZP portal due to a lack of

important functionalities. Therefore the analysis of Moodle has to be done to find out the possibility of integration with BOZP portal.

1.3 Design an integration of Moodle into BOZP portal for training tests, add the following functionalities

Next step is designing the actual working parts of the system that will together build one functional application. As the project is focusing on integration of BOZP portal and Moodle, I have to come up with a solution for an authentication throughout the systems, data export and entire communication interface. In the following sections is described each required functionality:

- **Authentication through Shibboleth, respectively through BOZP portal**
The majority of applications at CTU use the Shibboleth authentication method. It is a convenient sign-in system facilitating the cross visit of websites associated with the school. This way of authentication has been implemented into BOZP portal therefore to keep consistency and unnecessarily not confusing users, any sign in processes with Moodle should be conducted the same way.
- **Test generation for declared trainings**
The key aspect of the eLearning module. There is a need to find out the solution, how to generate new tests on behalf of cooperating system with BOZP portal. Based on previous Moodle analysis the best solution is chosen and design will be created.
- **Attendance lists generation**
Users who successfully passed a training test are included in generated attendance lists. Generation takes place after test results are exported from the supporting system; in this case it is Moodle.
- **Automated exports of test results into BOZP portal**
As mentioned in previous point results need to be exported in order to generate attendance lists. Exportation has to be automated and current data should be always presented to the user of BOZP portal.

1.4 Implement changes into BOZP portal

The changes will be made in the latest version of BOZP portal continuing and extending the work by Bc. Kamil Falta [2] who has worked on the regular training module. However, not all changes will necessarily involve BOZP

portal. The outcome of Moodle analysis can show need to strike into its implementation because, for example, a lack of functionalities.

1.5 Perform automated and user testing of implementations

In order to prevent undefined system state or occurrence of some security holes therefore the application must be conducted to the testing process. Automated testing ensures application correctness and consistency of behavior when any changes in code are performed. And user testing will examine the application from the view of usability.

Analysis

2.1 Current form of eLearning

At the beginning of the BOZP portal project, there was an effort to make one complex application which is self-sufficient, needing no outside support by any other third-party system. The eLearning part was no exception. The portal has now implemented fully manageable eLearning system. It includes administration of questions and following creation of tests for different categories of users. Administrator is able to add new materials such as links or files and use it afterwards for test questions.

2.1.1 Solution drawbacks

Although using eLearning module over couple years has shown up that it is missing some features and there is a need for extension. It is particularly the user interface and functions that are key indicators of successful application. In the following points I break up into the drawbacks of BOZP portal, mainly gathered over the years by the OSH worker.

- **Low user interface:** Current solution does not allow its users to work continuously. Every action such as editing or adding is displayed onto a different page which is indirect, fragmented and not clear to work with. Nowadays technologies have progressed rapidly and offer more user interaction with applications through AJAX, for example, drag and drop operations or pop-up windows.
- **Low functionality:** In BOZP portal users work with big amounts of information that need to be manageable. Most tests are made with dozens of questions and their preparation is not quite simple. Therefore the data importing comes in handy. The list of questions can be prepared using common text editors and simply be imported into the system.

Also as a backup functionality user can use the exportation of previously added data, for example, in case of the system migration or distribution.

- **Complicated sign in process:** Students all over the faculty must sign in to the portal every time they are obligated to take a test. Login credentials are not specifically unified and it leads to confusion. At school now exists the single-on login method which does not require the filling in of the login and password every time.
- **Test attendance list:** Until now, attendance lists were generated with names of all students from a class. That included the students who either didn't attend the test, didn't pass according to the grade system, or even the ones who no longer study at the CTU. The new solution should help the responsible person to keep better organization by sorting out the students who are relevant and belong to the list.

Like any other system, with the progressing time things get older and new things are developed. BOZP portal and particularly the eLearning module is older than three years which can be seen also in the web design. The school's infrastructure is getting renewed which influence this project too to keep up with the new technologies.

2.2 Trade study

Despite the fact that the final eLearning system is specified in the task list of this project, it is valuable to mention other solutions. Knowing competitors is helpful to keep up with current trends and can be inspirational for solving the problems of this project.

To narrow down the area of the system variety, I have focused the study on the systems only including some common features as the current eLearning solution. Primarily involving tests builders and related grading and reporting system. After searching for terms such as "Learning Management Systems (LMS)" or "moodle alternatives", several results came out that included *CourseSites*, *eFront* and *Google Classroom*. In the following section is described each platform.

2.2.1 Moodle alternatives

2.2.1.1 CourseSites

CourseSites is a web based free version of popular eLearning platform Blackboard Learn. It is aimed at individual instructor form and allows instructors to have up to five course websites, share open education resources, access to manage grades, integrate multimedia into class content, and engage student in social learning [3]. CourseSites is even available for mobile devices through

Blackboard Mobile Learn app. It's a very user-friendly system with similar control panel layout as Moodle. Also, there is a better question management that allows to link the questions and easily and quickly change any of their settings. Nevertheless CourseSites is missing some of the functionalities of BlackBoard paid version which is unusable for purposes of this project. These include custom web services to deliver remote control and single sign-on login method. To conclude, Moodle is currently more suitable platform, but we can't exclude possibility that CourseSite will add in the future the needed functionalities.

Website: <https://www.coursesites.com>

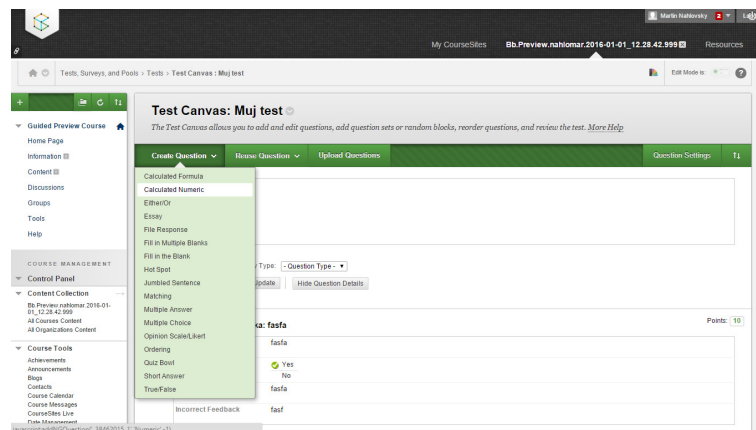


Figure 2.1: CourseSites demo - adding a new question to quiz

2.2.1.2 eFront

Another system with the same domain as Moodle is eFront. It is an open source eLearning system with cloud based solution. eFront includes an intuitive icon-based interface, course creation, test builder, assignments management, reporting and others. Although, the interface is well designed and very simple to use, there is a possibility for some users having hard time when finding its full functional capability that leads to opening the user manual. eFront has a different approach to creating course content than Moodle. Its courses consist of lessons that are built from content-based units which can serve as place, for example, tests [4]. Other content like assignments or forum sit outside of the lesson structure. By comparison, eFront is a simple teacher-student oriented learning system with not that many functionalities, but suitable for the most situations.

Website: www.efrontlearning.com

2. ANALYSIS

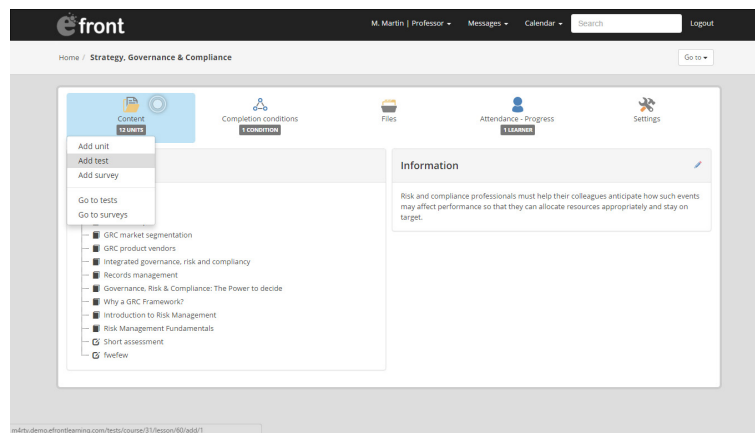


Figure 2.2: eFront demo - course dashboard

2.2.1.3 Google Classroom

Google has developed their own version of eLearning system, offering free, but only available to accredited academic institutions (those with a Google Apps for Education account) [5]. Classroom integrates with other Google applications like Docs, YouTube or Drive. Easy intuitive look and feel, as well as basic course functionalities, makes this system as good option for school eLearning system. Although, Classroom does not have automated grading system, it is not major drawback that could not replace the current BOZP portal. Already, many teachers from CTU encourage students to work with Google products like Docs or Sheets. This could possibly simplify the transition of teaching students with a new system.

Website: <https://classroom.google.com/>

2.2.2 Conclusion

Each of the above solutions has its pros and cons, having better application in different areas. The eFront platform appears to be the closest to the project demands because of its various customization through modules. It provides an XML API for communicating with third-party applications that is necessary in this project for passing test results.

2.3 Moodle

As mentioned above, Moodle has been chosen as the replacement for the current eLearning solution. Moodle platform is a free open source project with a big support of the online community. People around the world contribute

and help to develop the code base. It is mainly used to host online classes, but also companies find its usage too. Because of its easy modifiability, many universities have implemented Moodle into their infrastructure. It is a flexible and easily expendable system based on PHP.

2.3.1 Features

Some of the keys features in Moodle are user and group management, course management and web service support. Moodle modular capability allows to enrich these features with various different types of activities and resources. In the following text I will point out important functionalities of each feature:

- User and group management supports a variable range of user authentication methods throughout plugin extensions, allowing integration with existing systems. The basic method is the e-mail user registration. Students create their own login accounts. E-mail addresses are then verified by confirmation [6]. Next, Moodle offers usage of LDAP for creating new accounts login based on the checking against a LDAP server. Remote user data can be mapped onto users profiles. Similarly Shibboleth authentication method works on the user data exchange that helps define user info.
- Each online class is represented as a course that can take the format settings such as by week, by topic, social or single activity. Within the course, admin can set up activities and add new resources. Students are either manually enrolled into a course or use predefined password for access.
- Moodle supports standard web services across the code base and exposes Moodle functionalities to external applications [6]. Control over system access is secured by generated token and also by defined set of rules allowing or restricting function use. Web service communication with Moodle from outside is either XML-RPC, REST or SOAP.

2.3.2 Integration with BOZP portal

The goal of this project is to have two independently running applications working together, specifically Moodle and BOZP portal. BOZP portal acts as a consumer of Moodle services. Moodle integration with other existing system stands on exchanging messages through HTTP communication. For this purpose, there is an option to develop an external web service plugin when existing services are insufficient. To approach the good integration, there has to be figured out, how to replace the current functionalities of BOZP portal such as user accounts, training testing workflow and their evaluation.

First, we take a look at user registrations. In BOZP portal users have been imported through KOSapi and the credentials were validated against the LDAP server. This method is not no longer used at the school because of the new authentication method - Shibboleth. Moodle has solved this problem for us and provides full support for the Shibboleth integration. Admin can specify mapping fields of user data. This is the important point when avoiding of username duplication. Both systems then can easily reference to each other without need of repetitive user signing in.

Next, we discuss training tests. Alternative to training tests in BOZP portal is Moodle's type of activity called quizzes. They consist of questions of various type. Quizzes are automatically graded according to the defined settings with the outcome of a feedback. So far, Moodle provides web services only for course management, but not including quizzes. To manage quizzes remotely there has to be developed custom web service.

Tests and relevant results are stored in the same database for BOZP portal which makes easy to work with test results. As Moodle results resist on the different system and database, we need to send these data to BOZP portal. Similarly to managing quizzes, we can define web service that returns as result the exported data. Moodle provides standard XML data format.

2.3.3 Test generation

Moodle supports training tests in the form of quizzes with the management through the custom web services, but how can be tests (quizzes) created on the behalf of Moodle. First, we take into consideration adding new quizzes. There are two options.

The easy method is to duplicate already existing quiz and "only" modify some fields. We need to keep the list of the quizzes acting as templates. This can be achieved by setting up a course in form of topics and dedicate one topic to these quiz templates. Also, Moodle provides an option to hide the topics which allows to hide templates to be available only for instructors.

Another option is to create new quiz from a scratch. A new quiz requires to set up the feedbacks, the grading method and other minor necessary options. Also, adding questions to quiz and set each of their numeric grading. From the view of managing all the setups through the web service, it is the clean method, but more programmer challenging.

2.3.4 Data export

Data export is necessary for getting results of students who passed a quiz. On the behalf of Moodle students can take quizzes that are afterwards graded according to predefined settings. BOZP portal receives these grades and process them in order to generate attendance list for later compulsorily signature.

In Moodle any module can trigger an event with attached data [7]. In this way, any submitted quiz can trigger sending message to the other system. When a quiz attempt is submitted, an event is built and broadcasted to the other parts of the system. Then the event handling has to be defined in plugin by providing the communication with the outside world. The main drawback of this method brings need to specify the outgoing sources for the communication.

On the other hand, Moodle can server as a service provider and return data only on the request. The other side–BOZP portal–has to implement a way to automatically send requests to Moodle. It is more efficient way to save many connection between the systems by grabbing more quiz results at the same time. Server has to allow to enable cron or any other automatic time-based job scheduler.

2.4 New solution proposal

2.4.1 Training test workflow

As the starting point for the next analysis, I have created a high level view on the business process, including two main participants: BOZP portal and Moodle. BOZP is the initiator of requests and Moodle provides the automated work. Both systems communicate together and send data in order to progress in workflow.

Figure 2.3 shows workflow from the view of the training process for a test participant (i.e students or school employees). The story starts when the OSH worker announces a training event. In BOZP portal admin creates a new event of type eLearning. System will then send a request to Moodle where the new test is automatically created and ready to use by a trainer. Students obligated to take the test, has to sign in to Moodle, fill in presented question and submit results. The outcome is the final grade assigned to students which will decide the next action. BOZP portal collects all these data and orders students by the final grade. If student passed the test, then his name is included in the final attendance list, otherwise in case of failed test, student will be send an invitation to the resit test.

2.4.2 Class example testing

To better imagine, how system should correspond to a real-world training, the process is described on a class example. The OSH responsible person lists trainings for a course including the parallel classes. Each class has assigned teacher who will perform the testing process. Teachers are provided a password as the access to a quiz in Moodle. However, this password is valid only

2. ANALYSIS

for stated period of time. Students can access the specific quiz of the class, what they attend, only with the password from teacher and that time. This should eliminate any cheating efforts and enable everybody the same conditions. There should not be specifically defined student attendance list in each class. Any student who is registered to the course can possibly attend any class in order to take the quiz.

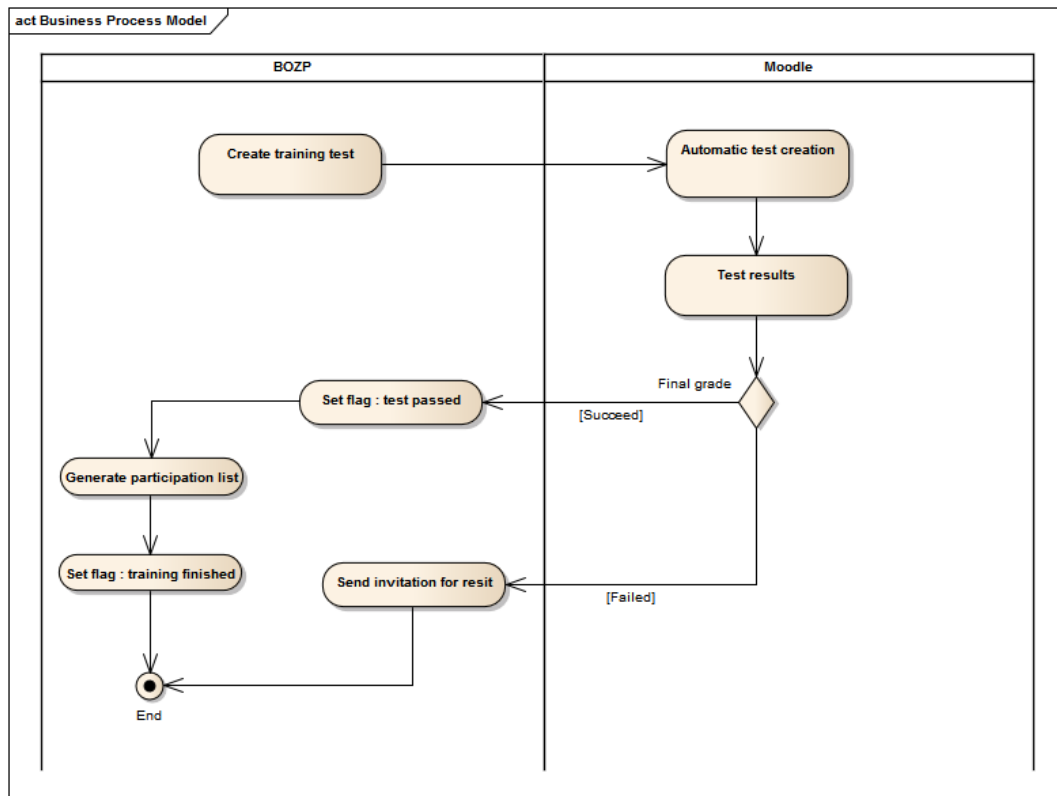


Figure 2.3: Business process model

2.5 Requirements

This section will cover behaviors of the required features on the system. It provides the specification of what is expected of the application, what it should do, and how it should work. This information was provided by the OSH worker. On top of the identified needs, I have added items that are also necessary and completed the full working progress of the proposed functionalities.

2.5.1 Functional requirements

The following is a list of requirements that must be delivered in the final implementation of eLearning module.

- **Authentication:** User will be able to sign in to the application by using his official credentials for CTU given by the student department. The same process is applied for signing in to the external training system. User accounts are paired via e-mail address. Authentication is provided by Shibboleth login service.
- **Create quiz:** User shall be able to create an event for the eLearning module that afterwards initiates creation of quiz in external system by choosing template for questions, setting up start and end time, filling in the name and the access password. Quiz is either type of regular term or resit. When resit than particular students are notified about this event via e-mail.
- **Edit quiz:** User shall be able to edit values of previously added event and its connected quiz from external application.
- **Archive quiz:** User shall be able archive previously created event. This event and relevant quiz will not be visible to common users. Archived quizzes can be reverted to active state.
- **Delete quiz:** User shall be able delete quiz from the external system using BOZP portal interface. The system is responsible for notifying user about finished process.
- **Show quiz detail:** User shall be able view details of quiz including event's name, start and end time of testing, room, person in charge of testing and list of users grades.
- **Import grades:** BOZP portal shall provide import of quiz grades from the external system into the database. The grades involves data of students who either passed or did not.
- **Generate participant list:** User shall be able to generate list of users who attended an event and successfully passed the training. The list is in a printable form that can be use for signatures of participants.
- **Show list of events:** User shall be able to view list of available events with action buttons for signing in to quizzes and viewing the detail of event. Additionally, user with admin role is shown event control buttons such as for editing.

- **Send notification:** BOZP portal shall notify user whenever automatic system import is processed. These information will be shown to the admin in the dashboard among other existing notifications.
- **E-mail alerts:** BOZP portal shall be able to send an e-mail alert about upcoming events. In case of resit training events, admin can choose manually required receivers. Users will then receive an invitation to the resit training with time and place of event.
- **Approve training:** User shall be able to approve training based on either successful completion of test or proof of previous completion.
- **Set up connection:** User shall be able to change settings of connection to the external system. This should include every necessary parameter to have ability connect other system of same kind in case of, for example, migration.

2.5.2 Non-functional Requirements

The following is a list of non-function requirements on BOZP portal.

- **Availability:** BOZP portal will be sending and receiving request within a communication with other systems, it should be always available to process a request. Any system downtimes could possibly cost a lost of data and therefore incomplete event in the application.
- **Response time:** Integration of two systems stands on time that takes each system to process an action, if one of them gets delayed, it can have an impact on the other. Thus there should be ensured that connection establishment and data transmission are performed in reasonable time and systems enable fluent user experience.
- **Maintainability:** Application should be well built by following recommended rules of used technologies to allow easy maintenance when problem occurs. Also adding new features should not cause any system downtimes.
- **Security:** With the consideration that BOZP portal processes private information of students and teachers, there should be provided sufficient and qualitative security with an effort to eliminate any sensitive data leak or system hack.

2.6 Technologies

List of technologies are briefly outlined to have clear view of already used platforms and also the ones that are newly proposed. BOZP portal runs on

Apache based web server with PHP and MySQL database, similarly configuration applies to Moodle. These common and well known technologies are excluded from the list.

2.6.1 Moodle

New technology added to BOZP portal. As mentioned above, it will be set up as an eLearning platform and provide data import of results to the parent application. At this time Moodle is available in version 3.0.0 with requirement of PHP version 5.4 and above.

2.6.2 Shibboleth

Shibboleth is an open source project, providing single sign-on authentication method across organization applications. The key components are identity provider and service provider. BOZP portal represents a service provider within user can authenticate with his or her organizational credentials, and the school identity provider passes back the identity information. It has been widely used in CTU environment for many applications. Another essential point is that Shibboleth is enabled for Moodle as an authentication extension plugin.

2.6.3 Nette

PHP framework that is continuously developed with a support of large online community. The very first BOZP portal was built on Nette. Since that time, there has been released newer versions and BOZP portal always keeps up with them, following new programming techniques to improve performance and stability. The update to new Nette is a part of Kamil Falta's master thesis [2], which is an important prerequisite for this project. Not only code migration includes the update, as well there will be an effort to replace old components for fresh one that Nette fully supports like datagrids or localization plugins.

2.6.4 Composer

Nette Framework is following global trends that includes also Composer, tool for dependency management. Composer pushes programmers to separate third-party libraries from their own code, having them defined in one place. It is very easy to upgrade any used dependency by simply changing version in the description file.

Design

The issue focused at system integration will be clarified in this section, as well as user interface for the new web parts in BOZP portal. Equally important is the architecture and decision, how every part will be structured and cooperate together.

3.1 User roles

To begin with use cases which describe the functional part of system, the user rules has to be first defined and explained of their position in the system.

- **OSH worker**

As the OSH worker I want to manage events and have full control of the process from their creation to extinction. I want the application to display list of users who passed a training in order to print the participant's document. Also, modify settings regarding Moodle server, receive notifications and grant trainings.

- **Student & school employee**

This role is the prior consumer of created content in BOZP portal. Both students and school employees are eligible to work with events. They can view list of available events and also see its details. As the student role I want the application to show my current trainings and its date of expiration.

- **System**

The system role performs operations automatically without user intervention. Calling data import, executing commands by other systems. It represents BOZP portal, Moodle and also a web server.

3.2 Use cases

The uses cases presented in this chapter describe the application functionalities from the view of the system roles. They are based on the requirements in Section 2.5 and extend their specification of use in the application. Figure 3.1 shows the use case diagram of user scope of allowed actions grouped by roles.

In the following section are listed the important use cases, visualized by activity diagram which allow to see interaction between the system roles supplemented by a textual description of the the ongoing actions. The main focus is at behavior of user towards BOZP portal and its cooperating system Moodle. Firstly, I will mention the use cases that are different and interesting, then only a brief list of the rest of the use cases.

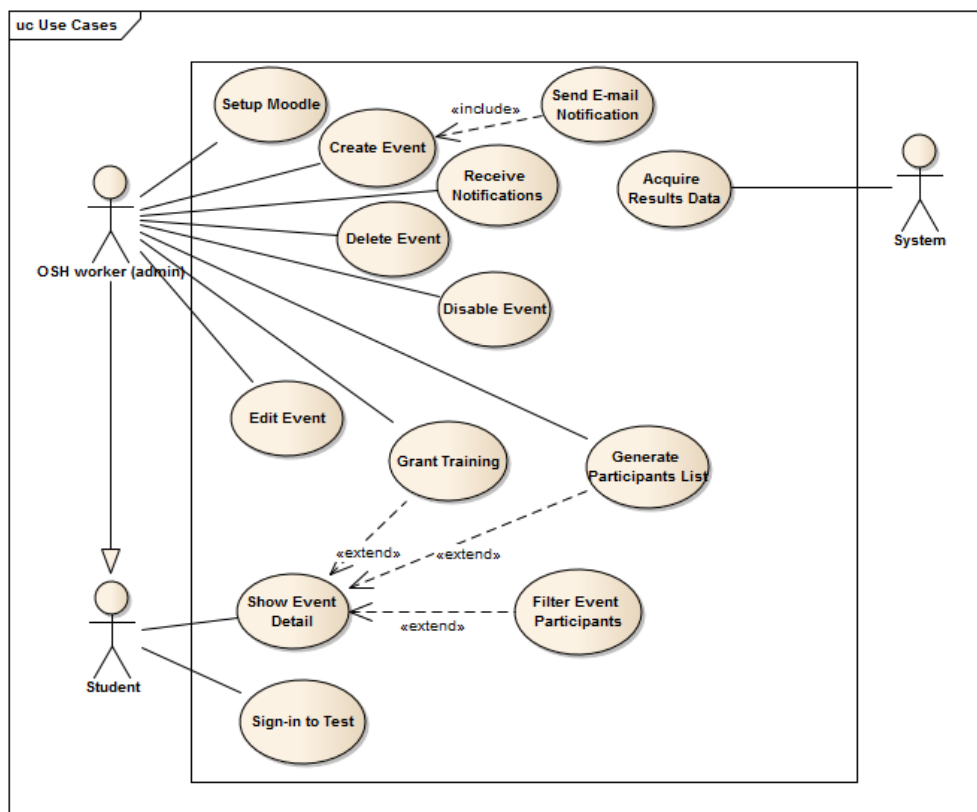


Figure 3.1: Use case diagram relationships

3.2.1 Create event use case

In this use case figures three roles that are user, BOZP portal and Moodle. User is the initiator of the action and interacts with BOZP portal. Whereas

Moodle receives requests from BOZP portal. The prerequisites of this use case assume user is signed in and has privileges for creating new events. The whole communication process is shown in Figure 3.2.

There are two important activities in this use case. First, the creation of a new quiz on Moodle's behalf that is preceded by duplicating an existing quiz (resp. template) which is selected by user in BOZP portal. Both systems should have consistent data, including list of templates, and also notify each other in order of any change. Second, if quiz is successfully created then its results need to be retrieved later. For this reason the new event is queued, representing job that will initiate the data retrieval. This process is described in use case "Acquire result data" in Section 3.2.4.

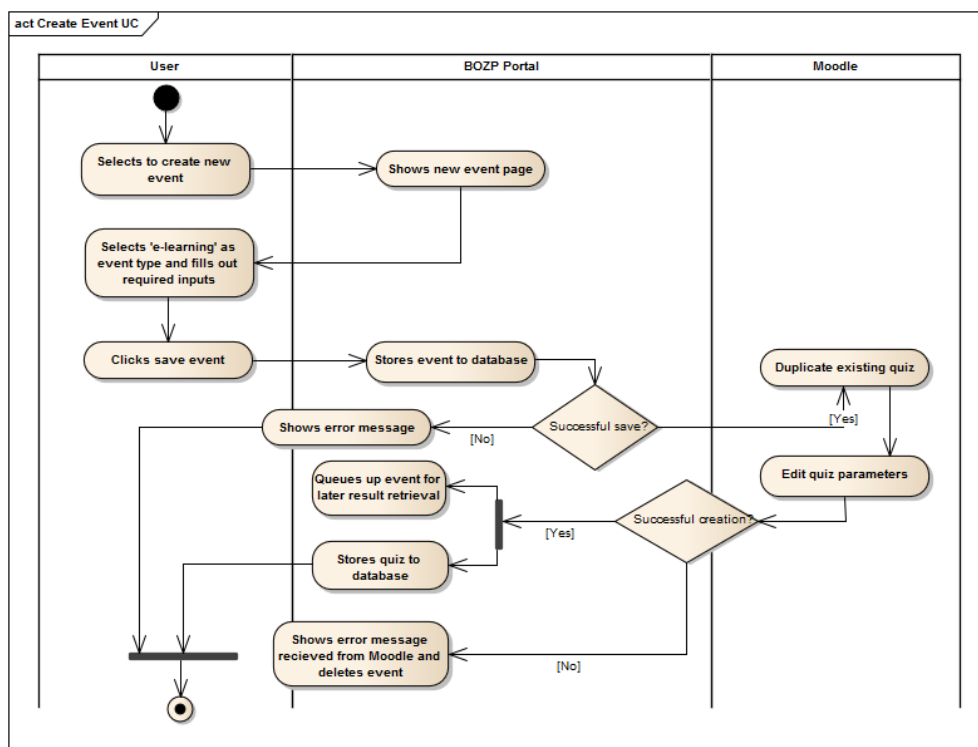


Figure 3.2: Create event use case diagram

3.2.2 Send e-mail notification use case

This use case is related to the "Create event" use case, the next step after user clicks on the "Save" button. The preconditions assume that the user selected resit as the quiz type which is an option for creating training events for students who did not pass the previous trainings. This use case involves user and BOZP portal. User requests portal to send e-mail notifications to

users who has been chosen as recipients. The notification includes a text and a link, used to alert user of the announced event.

3.2.3 Sign in to test use case

This use case involves four actors, user, BOZP portal, FIT Shibboleth gate and Moodle. The FIT Shibboleth gate is an independent application providing user authorization. The activity diagram of this use case is shown in Figure 3.3. The preconditions assume user has an active account and active credentials by CTU. The use case steps show, how can user sign in to an event, beginning at BOZP portal, authenticating through Shibboleth and finally processing login in Moodle.

There are two important problems that this use case points at. The user always has to access BOZP portal first, in order to sign in to any training test. Any attempt of accessing Moodle directly will end up into redirection the back to portal's homepage. Next, what is worth of mentioning is how Moodle handles its new and existing users. When user tries to access Moodle, after clicking on the sign in button, then checking of the Shibboleth authorization is done first. If the credentials are correct, Moodle will look up the logged user in its database, if user is not found then a new account is created. Otherwise Moodle will proceed the authentication as a standard user account. This assures no redundant user account appears in Moodle system.

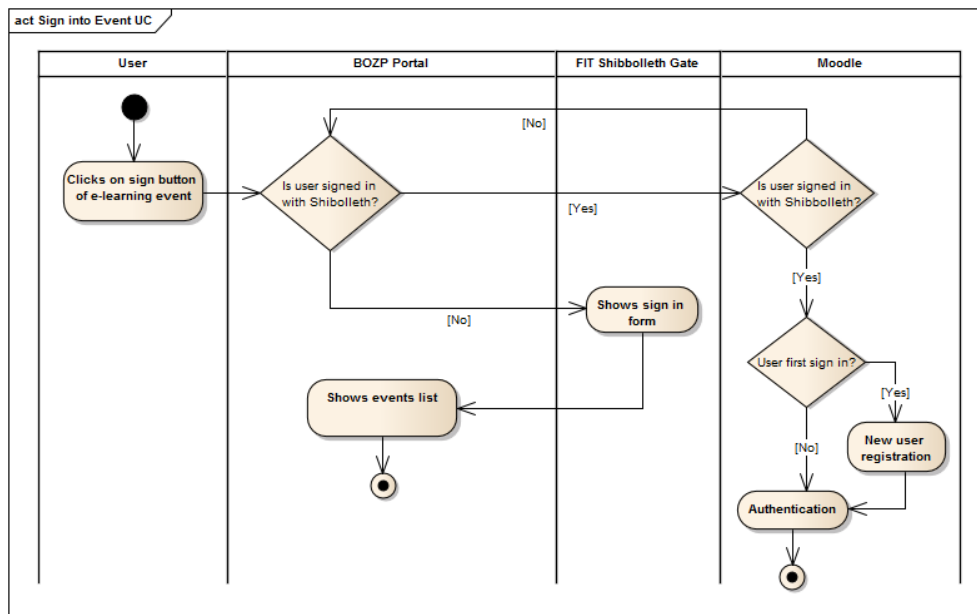


Figure 3.3: Sign in to event use case diagram

3.2.4 Acquire results data use case

This use case involves three roles that are web server, BOZP portal and Moodle. The web server runs BOZP portal that can have scheduled jobs. The activity diagram of the use case is shown in Figure 3.4. The preconditions of the use case expect working cron on the web server that calls the predefined script. The use case steps show the workflow of getting grades data off Moodle and following the store procedure in BOZP portal.

The object of this use case is to acquire data involving student grades. As described in the “Create event” use case when a new event of type eLearning is created, the new event is queued in the database, which represents a task. The task carries information about quiz start time that is used for launching a retrieval script. When the task is started, new grades are requested and stored into the database. Also the admin is notified by the system of the final retrieval process.

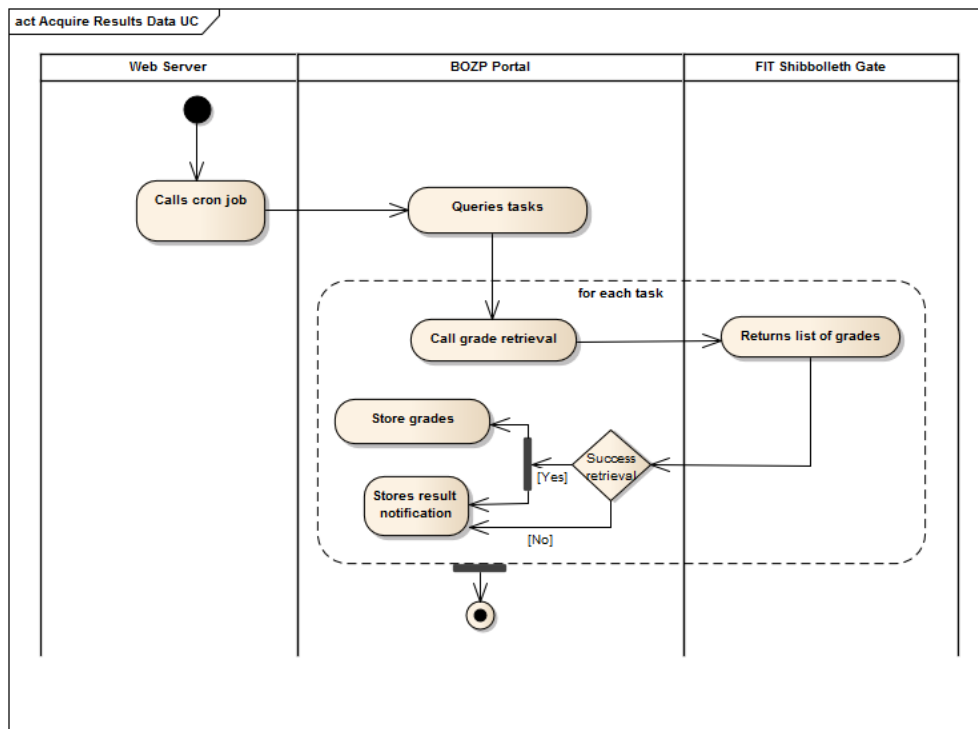


Figure 3.4: Acquire results data use case diagram

3.2.5 Receive notifications use case

This use case is related to the “Acquire results data” use case, having use in the step when the new data are stored into the database. The role system in

this use case sends notifications about success or fail of the data acquisition. The user then receives these information and is able to view them on the BOZP portal's homepage.

Other use cases

The following use cases cover simple actions or, they are derived from the previously mentioned use case. Each of them is briefly described, pointing at the main purpose.

- **Set up Moodle:** Admin will be able to change the Moodle connection settings in BOZP portal that involves target URL, security token, default course, section for quiz templates and section for live quizzes.
- **Grant training:** As soon as a student finishes a test and signs on the attendance list, the admin is enabled to grant the training to the student. System will provide a list of those students who have passed, allowing the admin to confirm students' training. The preconditions are laid on the student training results that has to be in the status "passed".
- **Show event detail:** Any user is enabled to view the details of all announced events. The detail includes attendance list with grades, if available. The precondition counts with signed in user in BOZP portal.
- **Generate attendance list:** This use case provides a PDF format output of a list with users who passed the assigned test. The list is in table form including a column for signatures. System enables this options when any successful test result was processed. The use case is a part of the "Show event detail" use case, but available only to the users with the admin privileges.
- **Edit event:** Similarly to the "Create event" use case, this use case operates with the event settings and its related quiz on the Moodle's behalf. Any change that is affecting the quiz configuration will trigger an edit request to Moodle.
- **Delete event:** Admin is permitted to delete an event. Calling this action will initiate the sent request to Moodle that deletes the related quiz. The precondition assumes the event is type of eLearning.
- **Disable event:** This use case allows user to disable an active event. Not only in BOZP portal becomes the event unavailable to students, but also in Moodle, the action will hide the quiz, but still will be visible to the administrator.

- **Filter event participants:** This use case is a part of the “Show detail event” use case, allowing the user to filter each field of the attendance list that involves: name, username, grade and user training state.

3.3 Moodle integration

BOZP portal will serve as Moodle interface to manage quizzes. The interconnection between both system is assured by a web service that allows to send a request to Moodle and receive a respond at the portal’s side. For this project purposes is necessary to develop a custom web service with features involving quiz management, grades export and course content export.

The quiz management will be solving a categorization of quizzes into course topics. In order to be able create a copy of an existing quiz in Moodle, there will be introduced two types of quizzes. First, quizzes that serve as templates. These will be included in a course topic, hidden from students, but available to administrators. Second, regular quizzes paired with events from BOZP portal. Unlike template quizzes, the regular quizzes will be placed in their own topic, visible to all users in Moodle.

3.3.1 Custom web service

To be able to manage quizzes in Moodle, there is need to provide new functionalities into the custom web service. The goal is to partition functions into the single units and have an opportunity for their use in the future cases. The following is a list of the necessary functionalities:

Duplicate quiz

Creating new quizzes assures duplication method that copies an existing template quiz and adds a new quiz record into the system. This functionality consists also of calling “Edit quiz” method, for changing settings parameters.

Edit quiz

This method allows to edit quiz parameters, regarding name, opening and closing time, password, section and visibility. The opening time matches to the event beginning time and the closing time is calculated from the limit for the testing. The section parameter stands for course topic in Moodle and visibility enables hiding of inactive regular or template quizzes. As mentioned previously, the “Edit quiz” method is used in the duplication process, but besides that, also when editing an event of eLearning type.

Retrieve quizzes

The list of the available quizzes from Moodle is used when deciding on the template option while creating a training event in BOZP portal. Hence, the output list can be narrowed down by choosing only items from specific course topic.

Retrieve grades

By a given parameter, specifying concrete quiz, this function returns grades of users who passed a quiz. The output is narrowed down by given time parameter that determines the minimal date for the submitted quiz. On the behalf of BOZP portal, there is subsequently scheduled a cron job that periodically grabs the quiz results.

Retrieve course sections

Following the analysis, the template quizzes are stored in a specific course topic. BOZP portal has to know where to look up the templates. This method will serve in determining, which course topic is the right one.

Deactive quiz

This method is specifically made for disabling quizzes. In Moodle, it will result in hiding a quiz, which become available only to the course administrators.

Retrieve courses

Similarly to the “Retrieve course sections” function, the main course is needed to be selected in BOZP portal to determine the source of the template quizzes, as well as targeting the duplicated quizzes.

Delete quiz

Fully manageable system includes also deletion. This function will delete a quiz with all associated modules.

3.3.2 Acquiring new data

Considering the process when uncommon structured data such as user grades arrive to the other system, there is need to be resolved a field matching. The web service function “Retrieve data” is responsible for extracting quiz grades, pack them and deliver to BOZP portal. Both systems has different database structure, thus users are stored in a variant way. In the school environment, there exists couple unique person identifiers. The most common is e-mail

address, username and digital number. Linking users based on the e-mail addresses, is for both systems very convenient distinct identifier of users.

3.4 Mockups

Set of Mockups were created to present user interface of the newly designed functionalities in the web application.

In Figure B.1 is shown a layout of the homepage. There are three boxes, each containing important information to the signed in user. First, the notification box, stores received alerts from the system. This involves status of scheduled import jobs in eLearning module. Also, each notification is linked to the event detail's page where are shown errors, in case of some unsuccessful import occurs. However, only users with the admin privileges can see the notification box. Next to the notifications lays the "My trainings" box. This box lists finished trainings, still within a valid date. Expired trainings change into the red colored font, indicating the need for extension. Last box, "My events", displays the list of events that belong to the currently signed in user.

Mockup in Figure 3.5 shows, how the event list will look like when user navigates to the "Training" menu item. The list includes the action buttons that allow user to manage an individual event. However, only to admin is presented the full set of the managing tools as shown in the mockup. Whereas, common user can view the event detail (show button) and sign in to eLearning events. The sign in button leads to Moodle that will next present a quiz to complete. When user wants to remove an event, the delete button pops up a window with the need to approve or cancel the action.

Figure 3.6 shows the web form for creating new events of type eLearning. Depending on the selected event type, the number of fields changes. If user selects the eLearning event type, additional four fields appear under the select box: quiz template, quiz type, parallel, password and activation. The quiz type field has also an effect on the appearance of the parallel select box. The presented mock shows the form when the regular term is chosen. The form fields are described with more details in Table 3.1. The processing of this form is related to the triggering the new quiz creation in Moodle. Beside the eLearning additional fields, the system passes to Moodle also other values that are start and end time. Another essential point, there is a different process when user wants to create an event for resit. If this option is selected as the quiz type, then clicking on the create button follows Figure B.3.

3. DESIGN

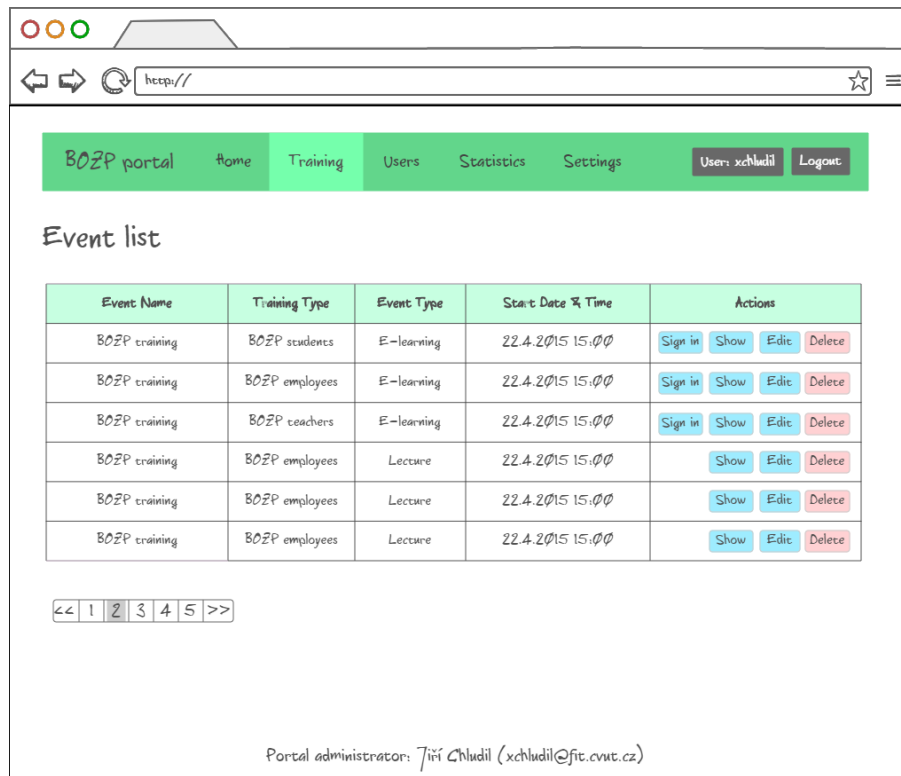


Figure 3.5: Event list mockup

In Figure 3.7 is shown an event detail from the admin point of view. At the top of the page, there is an outline describing the event, among others it involves a button to generate an attendance list. Clicking on the button triggers a pop-up window with the PDF file and list of names on it. The bottom of the page includes user attendance list containing students who passed the attached quiz. Next to each listed user, there is an action button to grant the training to him or her, under condition the attendance list was signed before. These functionalities are part of privileges of admin user. Other users can only see the details of training without any managing actions.

Figure B.3 shows the next step after the “Create event” mockup that is a form for selecting e-mail addresses to be notified about the upcoming event. This page occurs only if the resit test was selected before. The list contains users who are still missing the specific training type. At the top left corner of

Name	Description	Options
Training type	Determines period of training validity and its target user group. It joins resit quiz type to the regular ones.	BOZP pro vedoucí pracovníky PO pro vedoucí pracovníky BOZP pro pracovníky PO pro pracovníky BOZP a PO studenty Vyhláška 50/78 Sb, 4
Event type	Distinguishes two cases: when eLearning then quiz options are showed, otherwise creates regular event	Lecture eLearning Credit
Quiz template	A template has defined settings such as questions and feedbacks. New quiz will inherit these settings.	Source of options are pre-defined quizzes from Moodle
Quiz type	Type can be regular for a specific parallel or resit that is common for all parallels.	Regular term Resit
Parallel	A specific class that is defined with beginning time and a room.	Source of options are watched classes in BOZP portal
Password	A password phrase that will allow access to Moodle quiz	-
Active for users	Allows to show/hide an event from users without admin role	True, False
Date and time	The start date of event	Calendar is displayed to select date and time
Duration	Time in minutes that will be provided for training	-
Place	Specifies event location	-
Trainer	Responsible person for training process	-

Table 3.1: Create event form fields

3. DESIGN

BOZP portal Home Training Users Statistics Settings User: xchludil Logout

Create Event

General

Name

Training type

Event type

Quiz template

Quiz type

Parallel

Password

Active for users

Timing

Deadline

Date and time

Duration

Additional info

Capacity

Place

Trainer

Create Event

Portal administrator: Jiří Chludil (xchludil@fit.cvut.cz)

Figure 3.6: Create event mockup

the list, there is a checkbox that allows to select all displayed users.

The main page for changing the settings of Moodle connection is shown in Figure B.2. There are several options regarding not only connection parameters, but also possibility to select a course which will affect other options that are section for live and template quizzes. The bottom part of the page is composed of table with quiz templates. The table provide a quick preview of offered quizzes, directly exported from Moodle.

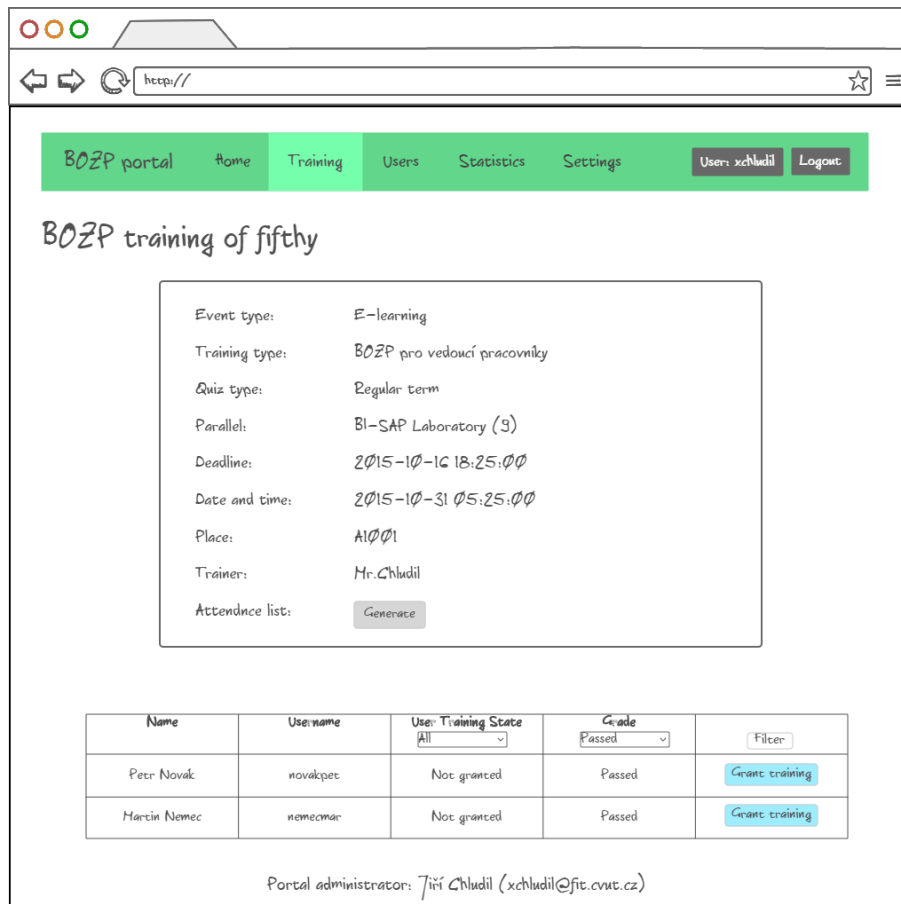


Figure 3.7: Event detail mockup

3.5 KOSapi integration

KOSapi plays a critical role in getting user data into BOZP portal. It is a REST-ful web service providing an access to the study information portal (KOS). The key data that are needed specifically for the scope of this project involves class parallels and teachers that are assigned to them. These information will use the eLearning module of BOZP portal at the point when creating a new event type. The teacher is set based on the previously chosen parallel class. Subsequently, the system sends a notification e-mail to the teacher, which is again, exported through KOSapi.

3.6 Architecture

Currently, BOZP portal is running on Nette Framework that divides application into three components, each very loosely tied together. These components involves presentation and application layer and database. The following sections describe the key points of each layer.

3.6.1 Presentation layer

In BOZP portal this layer represents an intermediary between the application layer and its views. The views defines the visual output and can get the content just from the presentation layer. While the application layer provides communication with the database, giving the application results or vice versa modifying the database tables.

The layer consists of many classes all sharing one main base class as their parent. Although, the eLearning module will affect only couple of the application classes, mainly the one related to the web parts: events, settings, dashboard and importing. The following list is a summary of all the classes that will be modified:

- **EventPresenter:** Any process and data manipulation regarding events is managed in this presenter. Event creation is related to quiz handling on the Moodle behalf which requires enrichment of the form by additional fields. Next, there will be modified the attendance list generator and event list and added a new e-mail notification process for the resit tests.
- **SettingPresenter:** This class deals with application settings that involves communication to other systems. There will be added the fields for the Moodle connection settings.
- **DashboardPresenter:** Dashboard provides system information to signed user such as notifications, trainings and upcoming events. The notifications will newly report results from the exported grades that are produced by Moodle.
- **CliPresenter:** This class allows communication with cron to call the scheduled jobs. As addition to the other functionalities, the presenter will be responsible for calling the export process directed to Moodle.

3.6.2 Application layer

In the Nette way, this layer is represented by model that is common class. The model assures communication with the database and also works as a gate to the outside world such as KOS system for getting necessary data. The main tables that will be used for the eLearning module are *quiz*, *quiz_grades*,

cron_job and *notification*. There are more other tables but not significantly affecting the developed module. In Figure 3.8 is shown the database scheme that illustrates relations between tables.

The quiz table stores an information of Moodle's quizzes that will be received as a result of calling the web service and its functionality for the quiz creation. Moodle returns a *quiz id* that is linked in the database to the column *moodle_quiz_id*. Later, operations such as update or delete, will use this parameter to remotely manage Moodle quizzes. The quiz table relates to the event table (*disposable_event*) that clarifies mainly the beginning and ending date.

When creating a new event, there will be also added a new task, which represents scheduled job for exporting grades. This task will be stored in the *cron_job* table. The table includes only active and incomplete tasks.

Finally, the notification table, where the application will keep alert notifications, involving the export process information. It does not include any joining column to other tables, notifications are general and able to carry only a link that leads to any application artifact.

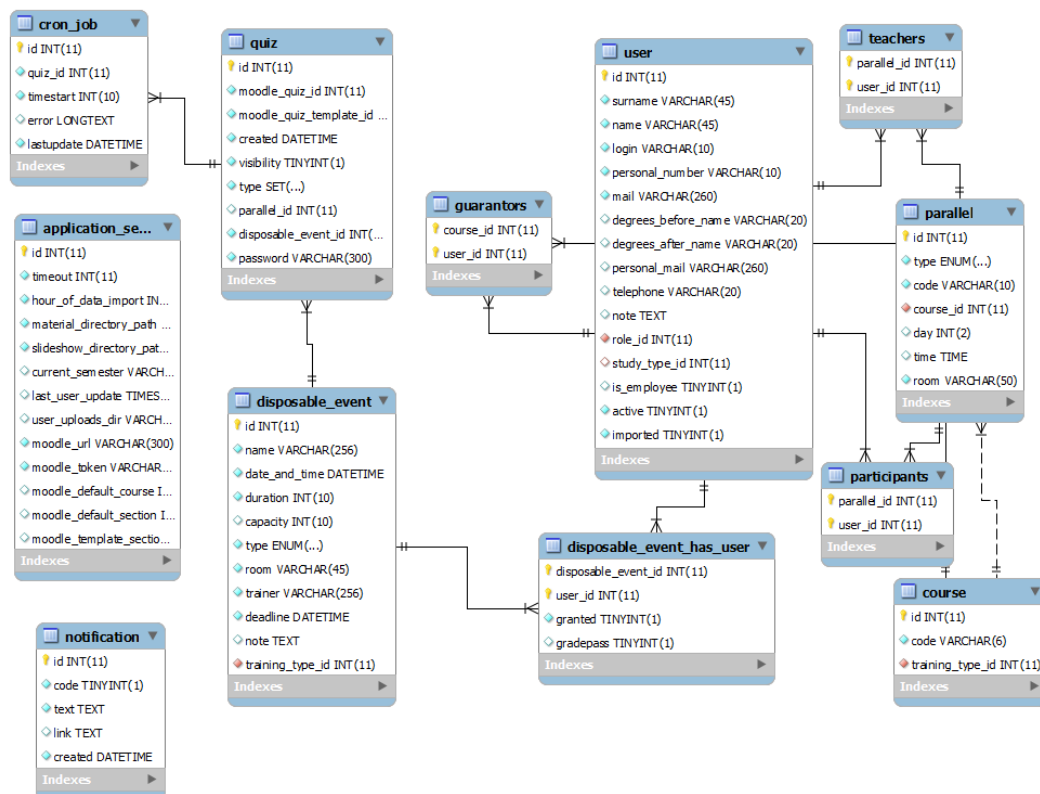


Figure 3.8: Database scheme

Implementation

This section covers the installation and configuration process of both BOZP portal and Moodle. Moreover, there are explained some implementation details which were most significant as for designing as well for development.

4.1 BOZP portal

This project was built on the copy from the previous BOZP portal version, only with the difference, the old functions has been migrated onto the new Nette Framework. Also the new design has been delivered by Kamil Falta who has worked on the training management module. The implementation of the eLearning module was performed independently on the other projects, mainly locally on the LAMP server with the support of versioning system - GitLab.

4.1.1 Installation

To run properly BOZP portal, it is required to have installed following software:

- Web server with Apache 2.0
- PHP version 5.4 and higher
- MySQL database version 5.5.4 and higher
- Allowed mod_rewrite module, *.htaccess* file protection
- Composer

Then when all above points are fulfilled, the source files of BOZP portal can be copied to the web root directory. Next, it is necessary to download the third-party libraries such as Nette and its related components. This can done through Composer, using command `php composer.phar install`, composer

will start downloading dependencies from its repositories that were defined in the file *composer.json*. Another essential point is to import the database scheme into the MySQL. There is performed only import of tables without any further setup.

4.1.2 Configuration

After the installation there is need to realize the configuration of the web application and also set up a cron job to turn on the task launcher. The application can be configured in the file *config.neon*, which is part of the directory *appconfig*. There are two important parameters to modify. First, the configuration allows to change the default language of the application. This parameter is called *translation*. Second, the database connection parameters. The following parameters are responsible for successful setup of the connection with BOZP portal:

- **host** - server hostname / ip address
- **username** - username to access MySQL client
- **scheme** - database name
- **password** - password to access MySQL client

On the web server side it is necessary to configure cron. The cron service runs in the background and constantly checks the */etc/crontab* file, */etc/cron.** directories. It also checks the */var/spool/cron/* directory [8]. BOZP portal requires to automatically perform exportation from Moodle, the crontab command shown below will activate the cron task automatically every ten minutes:

```
*/10 * * * * /usr/bin/php /home/www/app/test.php
```

In the above code snippet, the **/10 * * * ** represents when the task should happen. The first figure represents minutes in this case, on every “ten” minute. The other figures represent, respectively, hour, day, month and day of the week.

4.2 Moodle configuration

Moodle has a very simple installation process through the web application using step by step forms. After setting up the connection to the database and creating the admin user, the Moodle application is ready to use. The detailed approach can be find on the official Moodle website [9].

The activities of BOZP portal will take part in one single course. The course has to be created with the topics format. Topics will serve as training

sections for quizzes. One of the section will be devoted to the quiz templates and will be hidden from public users.

Communication with BOZP portal and Moodle is mediated via web services. The web service exposes to the outside world its functions that can be, for example, managing user accounts or quizzes. And also, previously mentioned quizzes are the main focus in this project. Moodle has not implemented any quiz web service. Therefore it was necessary to create a new custom web service. The following steps are needed to be performed to make the new web service working in Moodle.

- Copy source files to directory *moodledata/local*
- Create a web service in Moodle administration
- Enable web services
- Create an user account to use web services
- Create a new system role
- Assign capabilities to the role
- Assign user to the role
- Create a token

The token listed in the last point above is then set up in BOZP portal, allowing secured communication with Moodle.

The next step is to fill the database with predefined questions that will be part of quizzes. In the administration, there is a possibility to import any file format of data. I have prepared an export from the older system including questions for trainings that are currently used at school. The file can be found in the source files of this thesis.

Moodle itself does not allow automated enrollment to a course. This functionality is required to work as newly registered users will access Moodle through the Shibboleth service without interaction of the administrator. Users will use Moodle only for taking quizzes. In this case, the main course have to be automatically assigned to everybody. It can be achieved by the plugin AutoEnrol Enrolment Method [10], which fulfills the stated requirements. Installation process is following:

1. Copy plugin's directory *autoenrol* into *moodle/enrol*
2. Check admin notifications to install the plugin

3. Visit the “Site Administration →Plugins →Enrolments” page
4. Click the eye symbol next to “Auto Enrol” to enable the plugin

To complete the configuration and have ready system for usage, we have to create quizzes which will serve as templates for BOZP portal. In the following list are defined standard parameters common for required template quiz:

- Grade to pass: 0.85
- Attempts allowed: 1
- Review options: check mark all
- Grade boundary 100%: success
- Grade boundary 85%: fail

4.3 Moodle integration

In the following code snippet is presented a request call to a web server with Moodle. The *functionName* parameter represents the web service’s exposed function such as duplication or editing. The request is transported by XML-RPC protocol.

```
$request = xmlrpc_encode_request($functionName, $params,
                                array('encoding' => 'UTF-8'));

$content = stream_context_create(array('http' => array(
    'method' => "POST",
    'header' => "Content-Type: text/xml",
    'content' => $request
)));

$path = $this->url .
        "/webservice/xmlrpc/server.php?wstoken=" . $this->token;
$file = file_get_contents($path, false, $content);
$response = xmlrpc_decode($file, 'utf8');
```

4.4 Web service

The web service plugin for Moodle consist of files that define functions, current plugin version and translation variables. In the following list is described role of each file:

- **externallib.php** - implementation of core functions with a description of method parameters and return values
- **version.php** - defines current version that determines plugin's update in Moodle
- **db/services.php** - contains an array of definitions of the web service functions to install
- **lang/en/local_moodle_quiz_ws.php** - stores english localization of the plugin's name
- **client/client.php, curl.php** - files for remote testing of the plugin

For example, the code Snippet 4.1 shows definition of the duplication function in *services.php* file. The *classname* parameter represents implementation class in the *externallib.php* file, *methodname* is the name of a function, *classpath* stores path to the implementation, *type* can be set to “write”, whether the web service modifies something, or “read” when only reads data.

Listing 4.1: Example of function definition in the Moodle web service

```
$functions = array(
    'local_duplicate_quiz' => array(
        'classname' => 'local_quiz_external',
        'methodname' => 'duplicate_quiz',
        'classpath' => 'local/moodle_quiz_ws/externallib.php',
        'description' => 'Duplicates_quiz',
        'type' => 'write',
    );
```

4.5 Shibboleth

Shibboleth has its own plugin as a part of the Moodle package. The plugin has to be configured in the settings. As Moodle admin, go to “Administrations →Users →Authentication” and click on the Shibboleth settings. There is necessary to fill in the form fields. The fields *Username*, *First name*, *Surname*, etc. should contain the name of the environment variables of the Shibboleth attributes to map onto the corresponding Moodle [11]. These values for CTU's Shibboleth are following:

- **Username:** eppn
- **First name:** givenName
- **Surname:** sn

- **E-mail address:** mail

Next, Shibboleth should be set up as the only authentication method. To make this happen, we need to set the “Alternate Login URL” in the “Common settings” in “Administrations →Users →Authentication Options” to the URL of the file *moodle/auth/shibboleth/index.php*. This will enforce Shibboleth login [11]. Also the URL has to be protected by Shibboleth which causes the Shibboleth login procedure to start when accessing it. For Apache has to be defined a rule like the following:

```
<Directory /path/to/moodle/auth/shibboleth/index.php>
    AuthType shibboleth
    ShibRequireSession On
    require valid-user
</Directory>
```

Testing

In this section are outlined the key points from testing of the newly implemented eLearning module. BOZP portal involves wide range of technologies and services, from the Shibboleth authentication to integration with Moodle. Therefore, it is necessary to assure, the application works as expected, acquires correct data, and is user applicable.

5.1 User testing

This type of testing targets at finding application pitfalls from user perspective. Specific group of users are selected with the focus at their role in the system. They are given a set of tasks which cover the implemented functionalities. During the testing, it is being recorded user behavior, as well any problem, that the tested subject encounters. At the end, results are processed and put together to summarize the common issues and consider the other less often drawbacks.

5.1.1 Purpose

Tests are focused on user interaction with the application while measuring time that is need to perform requested tasks. Thus, it is possible to discover any problems which may be not only related to application functionality but also a lack of user friendly environment.

5.1.2 Scope and goal

The web application was extended and additional functionalities caused various design changes. Specifically, the event creation form was expanded by new fields, and the overall layout took a fresh look. For example when selecting the event type, the form changes interactively, which could possibly cost confusion. Next closeup look will be focused on the action buttons and

their correct using by users who choose and click on them as demanded. This involves monitoring sufficiently informed data tables, buttons and other web components.

5.1.3 User profile

The primary users of the application are students and the OHS worker. Both are familiar with the school environment and have the essential knowledge of the other school services such as the Shibboleth login or Moodle. The OSH worker tests the application from the view of the administration. That includes event managing and settings of the application. User student will be conducting the testing on the use cases such as signing in to courses, viewing event details and signing out.

Student

This can be any student from the school. No system knowledge is required. Chosen student need to have valid CTU credentials and be able to login to the school systems via Shibboleth.

OSH worker

A suitable candidate is a person who has experiences with the older version of BOZP portal. It is not only good to have a comparison but this person is the one, who knows, what to expect from the system, and is able to provide more needed information, such as possibly missing functions.

5.1.4 Testing process

The testing process will be held in the user experience (UX) laboratory that is located at the CTU's faculty of civil engineering. About three student participants and one OSH worker will be invited to be tested under real testing conditions which involve: a separate room with no disruptive elements, camera recordings from various angles, and audio recordings. The user will be seated in front of a personal computer that is able to access the portal website. A person, who supervises the testing, is responsible for fulfilling the testing conditions and also giving the user tasks to perform.

5.1.5 Tasks

In Table 5.1 are listed all the tasks that will be presented to the tested subjects. Each task describes the presumption before the task beginning, the conditions that need to be met, the state after finishing the task and time limit. The tasks has to be performed in the stated order to be able to test every implemented part of the system.

Number	Role	Task description	Details
1	Admin	Create an event of eLearning type - regular term	<p>Presumption: User has the “OSH worker” role in the system</p> <p>Conditions: User is signed out, starting at homepage</p> <p>Success: Event created</p> <p>Max. time: 10min</p>
2	Student	Sign in to the event and finish a quiz in Moodle	<p>Presumption: The event “My training” was successfully created in previous task by admin</p> <p>Conditions: User is signed out, starting at homepage</p> <p>Success: Moodle website opened</p> <p>Max. time: 5min</p>
3	Admin	Review received grades and create an attendance list	<p>Presumption: At least one student finished the quiz in Moodle, the deadline for finishing the quiz has passed</p> <p>Conditions: User is signed in, starting at homepage</p> <p>Success: A PDF output on screen with the attendance list</p> <p>Max. time: 5min</p>
4	Admin	Grant training to the students who has signed on the attendance list	<p>Presumption: Students on the attendance list have personally signed the printed paper</p> <p>Conditions: User is signed in, starting at homepage</p> <p>Success: Event created</p> <p>Max. time: 15min</p>
5	Student	Review your result of the finished training	<p>Presumption: User finished the quiz in task 2</p> <p>Conditions: User is signed in, starting at homepage</p> <p>Success: Training result is displayed</p> <p>Max. time: 5min</p>
6	Admin	Create an event of eLearning type - resit	<p>Presumption: There are students who have not passed the previous training</p> <p>Conditions: User is logged in, starting at homepage</p> <p>Success: Event created and notifications sent to users</p> <p>Max. time: 15min</p>

Table 5.1: Description of tasks for user the testing

5.1.6 Tester tasks

In the following points is briefly described each task. The number of a task corresponds to the one in Table 5.1. Detailed scenarios can be found in appendix C.

1. As an OSH worker, you are the organizer of the OSH trainings for course BI-SAP. For today at 12:00, create an eLearning event, taking place in the room THA1452. Set this term as regular term for the parallel “BI-SAP Lecture (1)”, select “Vzor test 50” as the quiz template. Students will have 10min time limit to finish the quiz. To create an event, open BOZP portal website, sign in and navigate to “Event management” under “Training” section. Fill in the form according to the previous information. Create the event.
2. As a student, you are attending the course of BI-SAP every Tuesday at midday with Ing. Jiří Chludil. Assuming now is the date and time of the class and also there has been scheduled the OSH training. Open BOZP portal and sign in with your CTU’s credentials. As logged in user, navigate to “Available events”. From the list of events, select the parallel “BI-SAP Lecture (1)”. Click on the sign in button to be redirected to Moodle. In order to start the quiz, ask the trainer for access password. Finish the quiz and submit answers.
3. Students have finished the quiz and system has exported the results from Moodle. Now, you as the OSH worker, have to print out the attendance list and present the copy to the students for their signatures. Navigate to the event detail page and generate the attendance list by clicking on the button “attendance list”. The list will pop up, ready for printing out.
4. After students sign on the attendance list, the confirmation of their training has to be done in BOZP portal. Navigate to the same page where the event detail is located, find relevant users and click on the button “Grant” to confirm the training.
5. As the student who has finished the quiz in Moodle, you want to find out information about the received grade. In BOZP portal navigate to your profile. Then see the training list in the right part of the page and find out what grade you got.
6. Some students did not pass the regular term training and some were missing due to various reasons. For this time comes at the scene resit terms. Create an event for the next week Thursday at 11:00 in the room THA1452. The time limit is 10min. When you finish filling out all the fields, click on the “Create” button. Next, select all the shown students

to notify them about the new term. Click on the button “Send” to finish the action.

5.1.7 Final questionnaire

After testing is finished, all tested subjects will be presented the final questionnaire. This will provide a review of the overall using.

1. Did you find the website understandable?
 - a) No
 - b) Rather not
 - c) Yes
 - d) Rather yes

2. Did you find the arrangement of the controls elements clear and transparent?
 - a) No
 - b) Rather not
 - c) Yes
 - d) Rather yes

3. Do you consider the application as a contribution?
 - a) No
 - b) Rather not
 - c) Yes
 - d) Rather yes

4. Would you add any other functionalities?
 - a) No
 - b) Yes Which one?.....

5. Are you satisfied with the training testing in Moodle?
 - a) No
 - b) Rather not
 - c) Yes
 - d) Rather yes

5.1.8 Results

After an agreement with the supervisor of this project, we have decided to postpone the user testing due to unfinished construction of the UX laboratory. It is necessary to conduct the testing under good quality conditions such as will be in the upcoming laboratory. The eLearning module will have a big impact on the changes in using of the BOZP portal training form. As a part of this project I have prepared all the materials needed to conduct the user testing. In the appendix C are shown detailed scenarios of each task that are mentioned in the section above.

5.2 Automated testing

This type of tests is able to automatically verify the key functionalities, ensure software quality and help programmers work more time effectively. The testing process is conducted by software tools which execute pre-scripted tests. Automated tests can easily test software when code changes occur to determine properly working application. Once automated tests are created, they can be easily be repeated as well as extended to perform tasks impossible with manual testing [12]. The automated testing in the eLearning module involves unit tests and integration tests. As the software tool was used Nette Tester that provides various options for assertion tests, easy usage with annotations and most importantly, it is able to test every Nette Framework's component such as presenters, latte templates and data models.

The testing process was focused on several aspects:

- Verify the application can handle any kind of received value
- Monitor communication with Moodle
- Create a good code documentation for future extensions

5.2.1 Unit tests

This testing method was used to verify that the application functions work as expected. Any given value can be determined and evaluated as correct result, as well as failure states are handle in the proper way. The actual tests consist of single classes for each functionality that are events, quizzes and notifications. The testing process was conducted every time when a new function was added.

Conclusion

Although, there was needed more time to create these tests, they had helped speed up the development. Mainly, because of the high coherence of the

application functions, some unexpected error were coming up very often. Also, any future code change will be put through this testing and help stabilize the application without any risk of damaging the original functionality.

5.2.2 Integration tests

This project involves integration of Moodle platform, which requires communication between applications. The test plan verifies correct return values of called functions that Moodle exposes through its web service. The methods use the same request sender, but they vary in the structure and the returned value. However, if Moodle fails to a process request, it always returns the same response including an error code and additional message about the problem. The testing covers verification of these aspects:

- connection to a Moodle server,
- Moodle side errors,
- and verifying expected values for BOZP portal.

Conclusion

These tests have a big asset in the process when migrating to a newer version which is very frequent concern in case of Moodle. Besides, Moodle and BOZP portal can show a different behavior on a local machine than running on the online web server.

Conclusion

This project presented BOZP portal - eLearning module, a web based application for OSH management in the school environment. It has explored the vision and conception as well as requirements and design. The module has been implemented successfully using Moodle platform as supportive tool for creation of training tests. I am confident that the next person who will participate on the development of this part of application will benefit by building on good basics given into the design and the implementation. The future plan in Section 5.2.2 outlines other possible module extensions. After completing this project, I have learned how to use Moodle, its installation process as well as configuration, changing the default behavior to customize as is required and also how to develop a custom plugin. All gained knowledge is explained and discussed in this work. I am also more familiar with Shibboleth and Apache technologies and their using.

The final product delivered the features as defined in the tasks of this thesis. It also demonstrated the technology and techniques discussed in the analysis part and further used in the design. The following captures the key items achieved from the work on this project:

- Implemented the communication interface using XML-RPC protocol to expose functions that can manage quizzes in Moodle. Also successfully installed the plugin into Moodle and configured its integration.
- Configured Moodle and its quiz module in order to suit the eLearning requirements of OSH at the university. That involve course setup, questions import and template quiz settings.
- Working Shibboleth authentication method across both applications to allow continuous transition between them.
- Implemented BOZP portal - eLearning module by integrating with existing parts and their extension.

The remodeled eLearning module has a significant impact on the whole system of training within the OSH. It facilitates the administration of preparing tests for students, as well as the process for granting trainings. Starting from the announcement of a training event, the OSH worker has a complete view on students' attendances and results of quizzes. While students are obligated to complete tests, the process is lightened by using well known Moodle environment.

Evaluation of completion performance

In the following section I will discuss the quality of meeting task requirements that has been stated at the beginning.

Analyze the existing solution of the eLearning form of training, identify its drawbacks and get the latest requirements from the OSH workers

The existing form of the eLearning module has been analyzed from usability view as well as functional quality. The key drawbacks were acquired from the end users and the OSH worker (Ing. Jiří Chludil), then clearly summarized in a list in Section 2.1.1. Next, I have identified the functional requirements based on the business model 2.3 which served as a range of necessary application extensions. The main source of application changes and new implantations requests was the OSH worker. In addition, I have summarized the used technologies and pointed out the value to BOZP portal.

Analyze Moodle platform and verify its usability to the needs of BOZP portal

I have performed a complete analysis of Moodle platform, covering its possible benefits for BOZP portal. In Section 2.3 are discussed various form of quiz creation, user registration and exportation of data. These gained knowledge have been helpful when working on the following design and implementation of the eLearning module as well as the Moodle web service.

Design an integration of Moodle into BOZP portal for training tests, add the following functionalities

On the basis of the previous analysis I have designed an optimal solution for the integration of Moodle by comparison the proposed options. The first part of design consists of use cases which define a functional scope of the project within an user competence. Some use cases include an activity diagram to better understand the described functionality. The design further includes description of the architecture and its changes regarding the new module.

Implement changes into BOZP portal

I have fulfilled all requirements put on the early concept. On the side of BOZP portal, there were implemented functions from the scheduled jobs to event management using XML-RPC protocol. Moodle included implementation of the additional plugin for communication with BOZP portal. Every use case from the design is fully functional and ready to use. In the implementation I have described the installation and configuration process of BOZP portal as well as Moodle. Furthermore, there are also mentioned some interesting implementation cases.

Perform automated and user testing of implementations

The newly implemented parts were closely verified throughout written test cases such as unit and integration testing. Also, I have prepared materials for user testing that involves questionnaires, scenarios, and recommended testing method. However, due to unfinished construction of the UX laboratory, user testing schedule was impacted to the point where these tests have to be done after the submission of this work. The expected testing process and outputs will be presented at the defense.

Future work

This project covers only basic usage of Moodle to realize the connection to BOZP portal. More work can still be done to fully utilize the features that Moodle provides. In addition to this project scope, other improvement can be considered. In the following are discussed some possible extensions:

Question bank manager

BOZP portal now allows to use the predefined quizzes (as templates for new one) from Moodle, which causes, that also questions are inherited. Although, there is an option to have questions selected one out of many, it is not always the wanted option. The improvement would consist of ability to choose any group of questions that question bank in Moodle offers. Also, it would good to consider an implementation of question import function.

Support for more courses

In the future, BOZP portal could be used by the other CTU faculties. For this purpose, each faculty would have to be separated into the distinct courses. The scope of this work covers only possibility to cooperate with one single course in Moodle.

Notifying user via e-mail

People are nowadays overwhelmed with information that comes from countless sources. It is hard to keep up with the news that every source provides. E-mail notifications are the way to stay in touch with users and inform them of various events. BOZP portal should be also enhanced of this feature more in depth. There only exist notifications when user is invited to a resit test or a new password is given to teacher. But it is missing some important event notifications. For example, when an user is granted a training or a training is about to expiry. These features can very be helpful for BOZP portal's users and help even more facilitate job of the OSH worker.

Bibliography

- [1] JESCHKE, Luk. *BOZP portl - E-learning* [online]. May 2012, [cit. 2012-05-18]. Available from: https://dip.felk.cvut.cz/browse/pdfcache/jeschluk_2012bach.pdf
- [2] FALTA, Kamil. *BOZP portl - Sprva kolen (zatm nepublikovan prace)*. Bakalsk prace, VUT v Praze, Fakulta informanch technologi. 2016.
- [3] CourseSites. *CourseSites - Online Learning Management System* [online]. March 2013, [cit. 2013-03-15]. Available from: <http://edtechreview.in/reviews/197-coursesites-online-learning-management-system>
- [4] eFront. *Moodle and eFront side-by-side* [online]. October 2012, [cit. 2012-10-18]. Available from: <http://www.efrontlearning.net/blog/2012/10/moodle-and-efront-side-by-side.html>
- [5] MEDVED, JP. *The Top 8 Free/Open Source LMSs* [online]. October 2015, [cit. 2015-10-01]. Available from: <http://blog.capterra.com/top-8-freeopen-source-lmss/>
- [6] Xeo Bangalore. *Moodle overview and feature lists* [online]. February 2013, [cit. 2013-02-14]. Available from: <http://xeois.com/blog/Moodle-feature-list-LMS-e-learning-2.3-2.2>
- [7] Moodle. *Events API* [online]. October 2014, [cit. 2014-10-29]. Available from: https://docs.moodle.org/dev/Events_API
- [8] Red Hat. *Chapter 34. Automated Tasks* [online]. December 2015, [cit. 2015-12-10]. Available from: https://www.centos.org/docs/5/html/Deployment_Guide-en-US/ch-autotasks.html
- [9] Moodle. *Installing Moodle* [online]. January 2013, [cit. 2013-01-18]. Available from: https://docs.moodle.org/23/en/Installing_Moodle

BIBLIOGRAPHY

- [10] WARD, Mark. *AutoEnrol Enrolment Method* [online]. February 2015, [cit. 2015-02-20]. Available from: https://github.com/markward/enrol_autoenrol
- [11] HAGMAN, Markus. *Shibboleth Authentication for Moodle* [online]. October 2009, [cit. 2009-10-02]. Available from: <https://moodle.org/auth/shibboleth/README.txt>
- [12] SmartBear. *Why Automated Testing?* [online]. December 2015, [cit. 2015-12-20]. Available from: <http://smartbear.com/learn/automated-testing/>

Acronyms

API Application Programming Interface

XML Extensible Markup Language

REST Representational State Transfer

SOAP Simple Object Access Protocol

XML-RPC XML Remote Procedure Call

BOZP Bezpečnost a ochrana zdraví při práci (OSH in English)

OSH Occupational Safety and Health

OHS Occupational Health and Safety

CTU Czech Technical University in Prague

LDAP Lightweight Directory Access Protocol

PHP Hypertext Preprocessor

PDF Portable Document Format

KOS Student Information System

LMS Learning Management System

HTML HyperText Markup Language

PO Požární ochrana (Firefighting in English)

URL Uniform Resource Locator

UX User Experience

LAMP Linux Apache, MySQL and PHP

Mockups

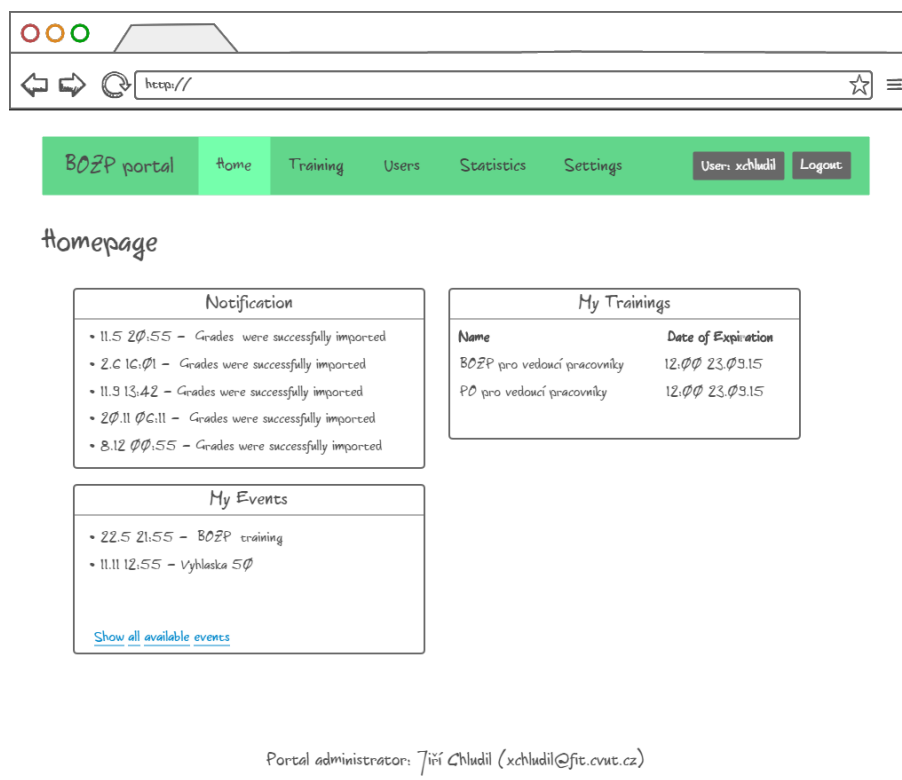


Figure B.1: Homepage mockup

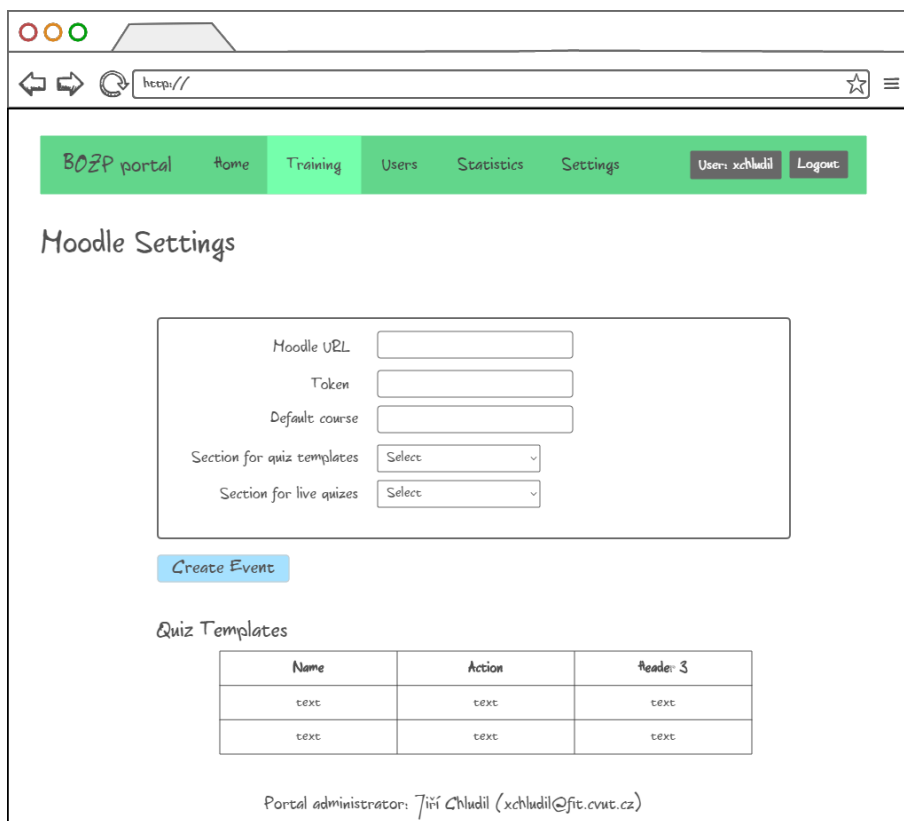


Figure B.2: Moodle settings mockup

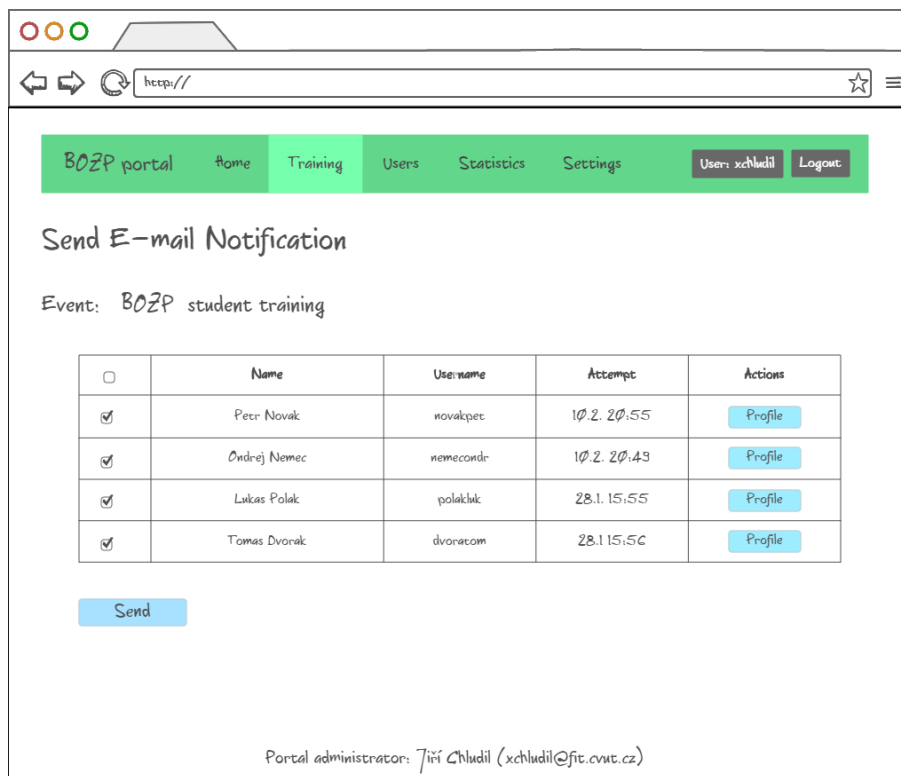


Figure B.3: Send e-mail notification mockup

Scenarios for testing

C.1 Scenario 1 - Create an event of eLearning type - regular term

Role: admin

1. Login in to BOZP portal with your CTU's credentials by clicking on the button "Login" in the upper-right-hand corner.
2. Select "Training" from the top menu and go to "Event management".
3. Click on the "Create event" button to go to the creation form.
4. Try to create an eLearning event named "My training" of the type "BOZP a PO studenty". As the quiz template select "Demo test" and for the quiz type "regular term". The training will be for the parallel "BI-SAP LECTURE (1)", today at 12:00, taking place in the room "THA1452". The password for access is "monkey". Students will have 10min time limit to finish the quiz.
5. Click the "Save" button to create the event.
6. Verify in the list of events that the new eLearning event was created.

C.2 Scenario 2 - Sign in to the event and finish a quiz in Moodle

Role: student

1. Login in to BOZP portal with your CTU's credentials by clicking on the button "Login" in the upper-right-hand corner.
2. Navigate to "Available events" in the top menu.

3. Click on the “Sign in” button next to the the event named “My training”, training type of “BOZP a PO studenty” and event type of eLearning.
4. In Moodle click on the “Attempt quiz now” button.
5. In the quiz password field enter “monkey”.
6. Fill in each presented question.
7. Click on the “Next” button to save the selected answers.
8. Review your answers and click on the “Submit all and finish” button to submit the results.
9. Confirm the pop-up window.

C.3 Scenario 3 - Review received grades and create an attendance list

Role: admin

1. Login in to BOZP portal in case you were logged out, as is described in Scenario 1 and step 1.
2. Select “Training” from the top menu and go to “Event management”.
3. In the table of events find the event “My training” and click on the “Show” button.
4. Scroll to the bottom of the of the event detail screen and review the received grades from Moodle.
5. Click on the “Generate attendance list” button above in the box with events details to view the attendance list.

C.4 Scenario 4 - Grant training to the students who has signed on the attendance list

Role: admin

1. Login in to BOZP portal in case you were logged out, as is described in Scenario 2 and step 1.
2. Select “Training” from the top menu and go to “Event management”.
3. In the table of events find the event “My training” and click on the “Show” button.

4. Scroll to the bottom of the of the event detail screen to the table with users.
5. In the table find students that are eligible to have the training according to the attendance list and click on the “Grant” in the right column.
6. Check if the training status of the selected students has value “Granted”.

C.5 Scenario 5 - Review your result of the finished training

Role: student

1. Login in to BOZP portal in case you were logged out, as is described in Scenario 2 and step 1.
2. Go to your profile page by clicking on your username in the upper-right-hand corner and select “Profile”.
3. Review the result of the “My training” training in the “Trainings” box located on the right side of the screen.

C.6 Scenario 6 - Create an event of eLearning type - resit

Role: admin

1. Login in to BOZP portal in case you were logged out, as is described in Scenario 2 and step 1.
2. Select “Training” from the top menu and go to “Event management”.
3. Click on the “Create event” button to go to the creation form.
4. Try to create an eLearning event named “My training - resit” of the type “BOZP a PO studenty”. As the quiz template select “Demo test” and for the quiz type “regular term”. The training will be for the parallel “BI-SAP LECTURE (1)”, today at 13:00, taking place in the room “THA1452”. The password for access is “monkey2”. Students will have 10min time limit to finish the quiz.
5. Click the “Save” button to create the event.
6. Verify in the list of events that the new eLearning event was successfully created.

User Manual

This guide will walk you through the basics of using BOZP portal - eLearning module including how to manage events, generating attendance lists, and setting up Moodle.

Get started by logging in to BOZP portal:

1. Visit the official website: <https://bozp.fit.cvut.cz/>.
2. In the right-upper-hand corner click on the “Login” button.
3. Login to Shibboleth with your CTU’s username and password.

D.1 Adding an eLearning event

To add an eLearning event, click on the “Training” menu item and choose “Event management”. Next, click on the “Create event” button. The system displays page with form for creating a new event D.1. Fill in the required fields labeled by star.

The default form does not show fields for creation of quiz (quiz template, quiz type, etc.). They are shown after the eLearning event type is selected.

The eLearning event can be type of regular or resit term. If you select the regular term, the form will be expanded by additional field labeled “parallel”. This type of event is dedicated for testing in chosen parallel. On the contrary, the resit term has no specific parallel and is dedicated for everybody.

After filling in all the fields, click on the “Save” button to create the event. In case the resit term was selected, the system will show page for notifying users. Otherwise system redirects to the starting page.

BOZP Dashboard Training Users Statistics FAQ Settings Martin Náhlovský cs en

Create event

General

Name *

Note

Event type

Training type *
BOZP pro vedouci pracovníky

Event type
E-learning

Quiz template
- Choose template -

Quiz type
Regular term

Parallel
BI-SAP LECTURE (1)

Password

Active for users?

Timing

Deadline *

Date and time *

Duration (min)

Additional info

Capacity

Place *

Trainer *

Save

Portal manager: Jiří Chludil (xchludil@fit.cvut.cz)

Figure D.1: Create event form

D.2 Managing events

The application allows to edit and delete previously created event. Both actions can be triggered from the same page. Get started by click on the “Trainings” menu item and choose “Event management”. The system displays list of events D.2.

D.2.1 Editing an event

Select your desired event from the list and click on the “Edit” button. The system will display editing form similar to D.1. If the event is type of eLearning, then this option can not be change because the event has related quiz in Moodle.

D.2.2 Deleting an event

To delete an event, click on the “Delete” button of the desired event in the list. The system will pop up the confirmation window. Proceed by click on the “OK” button to delete the event and related quiz in Moodle.

The screenshot shows the 'Events management' page. At the top, there is a navigation bar with 'BOZP', 'Dashboard', 'Training', 'Users', 'Statistics', 'FAQ', and 'Settings'. The user is identified as 'Martin Náhlovský' with language options 'cs' and 'en'. Below the navigation is the title 'Events management' and a 'Create event' button. The main content is a table with the following data:

Name	Training type	Event type	Free spots	Date and time	
	All	All			Filter
Muj test	Vyhláška 50/78 Sb, 4	E-learning	0/0	06.01.2016 14:00	Show Edit Delete
My test (rest)	BOZP pro pracovníky	E-learning	0/0	06.01.2016 11:55	Show Edit Delete
Muj pokus streda	BOZP pro vedouci pracovníky	E-learning	0/0	06.01.2016 11:05	Show Edit Delete
Pokus 1	BOZP pro pracovníky	E-learning	0/0	06.01.2016 09:10	Show Edit Delete
Test školení	BOZP a PO studenti	Lecture	4/5	23.09.2015 16:30	Show Edit Delete
BOZP a PO studenti	BOZP a PO studenti	Lecture	50/50	01.07.2015 13:00	Show Edit Delete
Školení pro studenty	BOZP pro vedouci pracovníky	Lecture	30/50	28.06.2015 10:00	Show Edit Delete

At the bottom of the table, there are pagination controls: « First « Previous 1 / 1 Next » Last ».

At the very bottom of the page, it says 'Portal manager: Jiří Chludil (xchludil@ft.cvut.cz)'.

Figure D.2: Event list

D.3 Generating an attendance list

To generate an attendance list of event, click on the “Training” menu item and choose “Event management”. In the list of events, select the desired event and click on the “Show” button. The system will display the page with the event details D.3. Click the “Create attendance list” button located under the box with the event details. The system generates the list and displays the PDF file.

D.4 Granting a training

To grant a training to users who have accomplished all the requirements to obtain the training, click on the “Training” menu item and choose “Event management”. Select the desired event and click the “Show” button. Scroll

The screenshot shows the BOZP portal interface. At the top, there is a navigation bar with 'BOZP', 'Dashboard', 'Training', 'Users', 'Statistics', 'FAQ', and 'Settings'. The user 'Martin Náhlavský' is logged in, with language options 'cs' and 'en'. The main heading is 'Event detail: Muj test'. Below this is a box containing event details:

- Event type: E-learning
- Training type: Vyháška 50/78 Sb. 4
- Quiz type: Regular term
- Parallel: BI-SAP LECTURE (1)
- Deadline: 2016-01-14 10:10:00
- Date and time: 2016-01-14 14:00:00
- Place: THA1452
- Trainer: Chludil

Below the details is a 'Create attendance list' button. Underneath is a table with columns: Name, Surname, Username, User training state, Grade, and Filter. The table contains four rows of user data:

Name	Surname	Username	User training state	Grade	Filter
Baranov	Nikolai	barannik	Training not granted	Passed	Grant training
Smith	John	smiijo	Training not granted	Passed	Grant training
Barkov	Sasha	barksa	Training not granted	Passed	Grant training
Kulikov	Tom	kulitto	Training not granted	Passed	Grant training

At the bottom of the table, there are navigation buttons: « First, « Previous, 1 / 0, Next », Last ».

At the very bottom, it says 'Portal manager: Jiří Chludil (xchludil@fit.cvut.cz)'.

Figure D.3: Event detail

down to the bottom of the displayed page with the event detail to get to the list of users. Click the “Grant training” button next to the user whom you want to grant the training D.3.

D.5 Signing in to an eLearning event

To sign in to an eLearning event, click on the “Available events” menu item. The system will display a table including the list of available events. Select the desired eLearning event and click on the “Sign in” button to be redirected to Moodle D.4. In Moodle you will be required to enter a password which should be communicated by trainer.

D.6 Setting up Moodle in BOZP portal

To set up configuration for the Moodle connection, click on the “Settings” menu item and choose “Moodle”. The system will display the form for Moodle settings D.5. The form fields are described in the following list:

- URL - reference to the server where Moodle resides
- Token - a security token given by Moodle
- Default course - a course where quizzes will be stored

D.7. Creating a template in Moodle

Name	Training type	Event type	Free spots	Date and time	
	All	All			Filter
Test sample 3	BOZP pro pracovníky	E-learning	0/0	31.01.2016 11:35	Sign in Show
Test sample 2	BOZP pro vedoucí pracovníky	E-learning	0/0	30.01.2016 11:30	Sign in Show
My test (resit)	BOZP pro pracovníky	E-learning	0/0	23.01.2016 19:55	Sign in Show
Muj test	Vyháška 50/78 Sb. 4	E-learning	0/0	14.01.2016 14:00	Sign in Show

« First « Previous 1 / 1 Next » Last »

Portal manager: Jiří Chludil (xchludil@fit.cvut.cz)

Figure D.4: Available events

- Section for quiz templates - a section in Moodle that is dedicated to templates for quizzes
- Section for live quizzes - a section in Moodle where new quizzes will be stored

To update modified settings, click on the “Save” button.

BOZP Dashboard Training Users Statistics FAQ Settings

Moodle settings

URL
https://bozp.fit.cvut.cz/www/moodle

Token
89d1cf3d6d4d2498a668d26236ffac00

Default course
FIT-BOZP

Section for quiz templates
Sample tests

Section for live quizzes
Testy dle vyhášky 50

Save

Portal manager: Jiří Chludil (xchludil@fit.cvut.cz)

Figure D.5: Settings from for Moodle connection

D.7 Creating a template in Moodle

Templates are necessary when creating a new eLearning event. To create a new template, go to the Moodle application at URL:



<https://bozp.fit.cvut.cz/www/moodle>.

In the left navigation, click on the course “FIT-BOZP”. Moodle will display the course dashboard with sections for activities (i.e. quizzes) D.6. Click on the “Turn editing on” button in the upper-right-hand corner to allow module editing. Scroll down to the bottom of the page and click on the “Add an activity or resource” link in the section “Sample tests”. Moodle will display new page with the form for new quiz D.7. Fill in the fields and click on the “Save and display” button at the bottom of the form. As final step, add questions to the quiz D.8. The newly added quiz will be displayed as an option for quiz templates in BOZP portal.

The screenshot shows the Moodle course dashboard for the course "FIT-BOZP Bezpečnost a ochrana zdraví při práci". The page is titled "Moodle - FIT" and the user is identified as "Martin Náhlovský". The dashboard is organized into several sections:

- NAVIGATION:** A sidebar menu on the left containing "Dashboard", "Site home", "Site pages", "Current course" (with "FIT-BOZP" expanded to show "Participants", "Badges", "General", "Testy dle vyhlášky 50", "Sample tests", and "My courses"), and "My courses".
- ADMINISTRATION:** A sidebar menu on the left containing "Course administration" (with "Turn editing off", "Activity chooser off", "Edit settings", "Users", "Unenrol me from FIT-BOZP", "Filters", "Reports", and "Grades") and "My courses".
- Main Content Area:** Displays a list of activities and resources. At the top is a "News forum" link. Below it is a section titled "Testy dle vyhlášky 50" containing "Test sample 3", "Test sample 2", "My test (resit)", and "Muj test". At the bottom is a section titled "Sample tests" containing "Vzor Test 50" and "My sample test". Each activity has an "Add" icon, a link, and an "Edit" button.
- SEARCH FORUMS:** A search box with a "Go" button and an "Advanced search" link.
- LATEST NEWS:** A section with the text "Add a new topic..." and "(No news has been posted yet)".
- UPCOMING EVENTS:** A calendar view showing events: "Muj test" on Thursday, 14 January, 2:00 PM; "My test (resit)" on Saturday, 23 January, 7:55 PM; and "Test sample 2" on Saturday, 30 January, 11:30 AM. A "Go to calendar..." link is provided.


Figure D.6: Course’s dashboard in Moodle


 **Adding a new Quiz to Sample tests** 

[Expand all](#)

General

Name*

Description 


Display description on course page 

Timing

Grade


Layout

Question behaviour

Review options 

Appearance

Extra restrictions on attempts

Overall feedback 

Common module settings

Figure D.7: Adding quiz in Moodle

Moodle - FIT English (en) Martin Náhlovský

FIT-BOZP Bezpečnost a ochrana zdraví při práci

Dashboard > FIT-BOZP > Sample tests > Vzor Test 50 > Edit quiz

NAVIGATION

- Dashboard
- Site home
- Site pages
- Current course
 - FIT-BOZP
 - Participants
 - Badges
 - General
 - Testy dle vyhlášky 50
 - Sample tests
 - Vzor Test 50
 - My sample test
 - My courses

Editing quiz: Vzor Test 50

Questions: 14 | This quiz is closed Maximum grade: 10.00

Total of marks: 15.00

Shuffle

Page 1 Add -

1	1. pomoc Kdy můžete ukončit laickou první pomoc?	2.00
2	2. Odmítnutí pomoci Kdy mohou odmítnout poskytnout první pomoc?	1.00
3	3. číslo Jaké číslo má záchranná služba?	1.00
4	4. Random (Barvy vodičů and subcategories) (See questions)	1.00
5	5. Random (Bezpečné napětí) (See questions)	1.00
6	6. Random (Druhy sítí) (See questions)	1.00
7	7. Random (MN a NN) (See questions)	1.00
8	8. Random (Ochrana živých a neživých částí) (See questions)	1.00
9	9. Random (Ochrana živých částí (podrobně)) (See questions)	1.00
10	10. Random (Ochrany) (See questions)	1.00
11	11. Random (Paragraf) (See questions)	1.00
12	12. Random (Pojmy) (See questions)	1.00
13	13. Random (Symboly) (See questions)	1.00
14	14. Random (Zákazy) (See questions)	1.00

Add -

ADMINISTRATION

- Quiz administration
 - Edit settings
 - Group overrides
 - User overrides
 - Edit quiz**
 - Preview
 - Results
 - Locally assigned roles

Figure D.8: Question manager of quiz in Moodle

Contents of enclosed CD

	readme.txt	the file with CD contents description
	src	the directory of source codes
	impl.....	implementation sources
	bozp.....	BOZP portal sources
	moodle.....	Moodle web service sources
	thesis.....	the directory of L ^A T _E X source codes of the thesis
	scripts	database scripts
	text	the thesis text directory
	thesis.pdf.....	the thesis text in PDF format
	thesis.ps.....	the thesis text in PS format