DESIGN OF THE ONLINE COURSE EVALUATION SYSTEM OF “LUARASI UNIVERSITY”, ALBANIA

ANISA GJINI
Student number: 41376

Master’s thesis in Software Engineering
Academic Supervisors: Dragos Truscan, Sébastien Lafond
Faculty of Science and Engineering
Department of Information Technologies
Åbo Akademi University
May 2019
ABSTRACT

Web development is a topic of broad and current interest in the field of information technologies. The ability to meet people's needs, the practicality and the variety of solutions being offered, makes web development an ideal choice in solving everyday problems.

This is the case with the Online Course Evaluation System for Luarasi University, in Tirana. This system is created based on the principles used when creating web services. The components comprising this system are some of the best techniques and tools applied in the creation of web services. The combination of technologies like PHP, MariaDB, JavaScript, and Ajax, grants the ability to design some of the best online systems. When these components are combined with planning techniques, management skills, and project management tools, the implementation of the Online Course Evaluation System is made possible.

The Online Course Evaluation System is a system built on the principles of basic security. This system has integrated the best techniques of database design and data manipulation. In the creation of such a system, we consulted the best Western practices of similar systems in existence in different universities. The creation of the system went through several stages, from the conception of the design idea of the database to the coding process in PHP, Javascript, HTML or CSS. At the same time, a plan was drafted explaining in detail the implementation of the system. For the team management, Scrum methodologies and Agile techniques were used. This Online Course Evaluation System gives to Luarasi University the ability to measure the performance of its lecturers in the subjects they teach.

Keywords: Web Development, Web Service, Database Design, Online Course Evaluation System, MariaDB, Scrum.
CONTENTS

ABSTRACT 1
CONTENTS II
LIST OF ABBREVIATIONS AND TERMS IV
I INTRODUCTION 1
   1.1 Background 1
      1.1.1 Online Evaluation Systems 3
      1.1.2 Our System 4
   1.2 Web Components 6
      1.2.1 Ajax 7
      1.2.2 Advantages and Disadvantages of AJAX Technique 8
      1.2.3 XAMPP 9
   1.3 Our Project 11
      1.3.1 Objectives of the Thesis 11
   1.4 Thesis Structure 12
2. SYSTEM DESIGN 13
   2.1 Overview of the Methodology 13
   2.2 System Requirements 14
   2.3 User Interface Design 18
      Figure 2.1: Login page of the system. 20
      Figure 2.2: Password Generation 21
      Figure 2.3: Courses to evaluate 21
      Figure 2.4: Evaluation page. 23
      Figure 2.5: Likert scale for Evaluation 24
      Figure 2.6: Comment Box 25
      Figure 2.7: List of reports by courses 26
      Figure 2.8: Report of a course evaluation 26
   2.4 Database Design 27
      2.4.1 Building the Conceptual Data Model 28
         2.4.1.1 Identify Entity Types 28
         Figure 2.10: Database Design 29
         2.4.1.2: Identify Relationship Types 29
2.4.1.3 Identify and Associate Attributes with Entity or Relationship Types 30
2.4.1.4 Determine Attribute Domains 31
   Figure 2.9: Attribute Domains of program’s entities. 31
2.4.1.5: Determine candidate, primary, and alternate key attributes 33
2.4.1.6 Check Model for Redundancy 34
2.4.1.7 Validate Conceptual Model against User Transactions 35
2.4.1.8: Review Conceptual Data Model with User 35
2.4.2 Logical Database Design 35
   2.4.2.1 Steps taken to design the logical database, as described by [15] in page.462: 35
2.4.3 Physical database design 35
   2.4.3.1 Steps taken to design the physical database 36
2.5 Design Decisions 40
3. SYSTEM IMPLEMENTATION 42
   3.1 Using of Scrum for the group management 44
   3.2 Tool-set for Project Management 46
4. SYSTEM VALIDATION 48
5. CONCLUSIONS 49
6. REFERENCES 50
# LIST OF ABBREVIATIONS AND TERMS

<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Full Form</th>
</tr>
</thead>
<tbody>
<tr>
<td>GSB</td>
<td>Stanford Graduate School of Business</td>
</tr>
<tr>
<td>R.I.T</td>
<td>Rochester Institute of Technology</td>
</tr>
<tr>
<td>WMU</td>
<td>Western Michigan University</td>
</tr>
<tr>
<td>Ajax</td>
<td>Asynchronous JavaScript and XML.</td>
</tr>
<tr>
<td>Xampp</td>
<td>Cross-Platform (X), Apache (A), MariaDB (M), PHP (P) and Perl (P).</td>
</tr>
<tr>
<td>PHP</td>
<td>Hypertext Preprocessor</td>
</tr>
<tr>
<td>DBMS</td>
<td>Database Management System</td>
</tr>
<tr>
<td>UIUC</td>
<td>University of Illinois at Urbana-Champaign</td>
</tr>
<tr>
<td>HP</td>
<td>Hewlett-Packard</td>
</tr>
<tr>
<td>AJAX</td>
<td>Asynchronous JavaScript and XML</td>
</tr>
<tr>
<td>DOM</td>
<td>Document Object Model</td>
</tr>
<tr>
<td>HTML</td>
<td>Hyper Text Markup Language</td>
</tr>
<tr>
<td>XHTML</td>
<td>EXtensible HyperText Markup Language</td>
</tr>
<tr>
<td>CSS</td>
<td>Cascading Style Sheets</td>
</tr>
<tr>
<td>XML</td>
<td>eXtensible Markup Language</td>
</tr>
<tr>
<td>XSLT</td>
<td>eXtensible Stylesheet Language Transformations</td>
</tr>
<tr>
<td>ASP</td>
<td>Active Server Pages</td>
</tr>
<tr>
<td>WND</td>
<td>Windows Navigation Diagram</td>
</tr>
<tr>
<td>DDL</td>
<td>Data Definition Language</td>
</tr>
<tr>
<td>DML</td>
<td>Data Manipulation Language</td>
</tr>
<tr>
<td>CRC</td>
<td>Cyclic Redundancy Checks</td>
</tr>
<tr>
<td>DDoS</td>
<td>Distributed Denial of Service</td>
</tr>
<tr>
<td>WBS</td>
<td>Work Breakdown Structure</td>
</tr>
<tr>
<td>ER</td>
<td>Entity Relation</td>
</tr>
<tr>
<td>GUI</td>
<td>Graphical User Interface</td>
</tr>
</tbody>
</table>
1 INTRODUCTION

Online course evaluation systems are systems used in most universities all over the world. They serve as measuring standards of pedagogical activity of professors, in order to analyse and improve the didactic activities.

The online Course Evaluation System gives to educational institutions the ability to measure the performance of lecturers in the subjects they teach. Through a questionnaire form, students give their evaluations on different topics such as teaching, curriculums, and lecturers’ teaching skills. These evaluations are one of the only means in which lecturers and the head of departments can obtain qualitative feedback from the students regarding their work. This way, from statistical analysis of the data gathered, they can create different graphs and reports, making it easier for them to decide where they need to make changes to improve their work and performance. The system is also a tool for the head of the department to inquire on the quality of the academic process ascertaining if there are any substantial problems present.

It needs to be said, however, that the statistical results from the data gathered, are not to be taken as 100% accurate and precise due to many factors, including: number of students participating in the evaluation process, honesty of the students while filling in the questionnaire and students’ participation during the academic term which can impact their judgment on the respective subject and lecturer.

This evaluation is more of an indicator and shows the perception of the students, and it is a good base to work on. The information generated by the answers given by the students can be used to perform appropriate changes in the curriculum and teaching method.

In the system we created, we have integrated a reporting module that allows the respective departments to read the evaluation results and make decisions based on the results of each course and lecturer. In this thesis will be listed all the elements, methods and tools that were used to build this system. The following chapters will emphasize the importance of the system being online, and the phases that the team went through to build the system.

The implementation of such a system at Luarasi University was needed based on the students' own need to carry out the performance evaluations of their lecturers, and also the interest of the staff to have qualitative and reliable feedback from the students.

1.1 Background

Luarasi University is a private University operating in Tirana, Albania that was founded in 2003 and was one of the first private higher education institutions in the country. Its main profile since their foundation has been in the legal field. A few years
ago, they also opened the Faculty of Economy and since September 2018, they have a new faculty, the Faculty of Information Technology and Innovation.

Since they used to be a quite small university, with close to 200 students per year, the evaluation of courses and lecturers used to be performed by hand, as a survey with pen and paper.

The procedure of the survey was as follows:

The Administration created tables in Microsoft Excel Format with a questionnaire for each lecturer and would then proceed with the process by going into each class to deliver the questionnaire to students. By following such a procedure, they would interrupt the lecture that was taking place. They would wait for the students to finish their evaluation and collect the filled-in questionnaires. After the papers were collected, they would deliver the questionnaires to the secretaries, who would proceed to enter the data manually in the Excel tables in order to obtain some results.

There are many obvious drawbacks in this way of operating the process;

To begin with the manual process itself, it took much time to perform the process explained above. Besides the time needed to go to all the classes to deliver the questionnaires to students, it would take much time to enter all the data, and the process generally took days. After finishing with the questionnaires phase and the data entry by the secretaries, they needed to filter the data in order to perform some statistical analysis, a phase which took some more days to accomplish.

The human error margin, meaning that there was a probability to enter the data incorrectly in the Excel tables and by doing so, there was an increased chance to obtain different results at the end of the process.

The reasons that there would be mistakes during the data entry process include staff’s fatigue, misreading of the data and lack of quality control on the work. In this way of processing, some of the forms would also be lost in between passing from one process to the other, and the physical paper results also meant that free physical storage would be needed to put the filled forms in order to save them for archiving purposes.

Furthermore, it meant interrupting courses and waiting for the students to finish filling in the questionnaires.

This part usually was very problematic for the lecturers, since it almost always meant that the course for the day would be impaired and they did not have enough time left for their own lecture. Despite the many benefits that the lecturers themselves would obtain from this feedback from students, the amount of work required, and the vague data analysis performed on them made the university think of creating a new method of evaluating the lecturers.

In some cases, students would fill in more than one paper for a course and, in this way, they would harm the whole process of statistical analysis and make the data not fully reliable.

Performing different analyse of the data was limited due to the type and content of the data gathered.
All these things mentioned, the expansion of Luarasi University in three different faculties and the number of students and lecturers having multiplied, such a method was not reliable and efficient anymore.

1.1.1 Online Evaluation Systems

Online Evaluation Systems are systems that are being used in most renowned universities. In our research, for example, we tried to find the ones that were as close to our scenario, so we could compare them with each other and have the best parts of each implemented in our project.

The examples that will be illustrated below are the best practices on which we are based. A good online course evaluation system for lecturers has been developed by Stanford University in the United States.

As can be seen in their webpage [1], students, lecturers, and system administrators can obtain access codes to the portals dedicated to each of them. If it is a student, it logs into the student section with the username and password provided by the university and in the "Student Tab" they have placed the link leading to the evaluation. This is the case for the two other types of users, too. Student access codes have been given to a third party that manages them. This way they maintain anonymity.

To encourage student participation in the evaluation process, they have thought about setting up different incentives like if the students evaluate all the courses they have in that semester, they can have access to checking their grades in advance, before the official publishing date. This type of rewarding method is, in fact, very positive for encouraging student participation. They have also placed in the evaluation list only the courses that are frequented by more than three students in the respective academic year. The students evaluate only the subjects that appear on their list. The application is for the entire Stanford University, but in the meantime, the bonus system we mentioned above is applied only by GSB, that is the Stanford Graduate School of Business, one of the best business schools in the world. Accessing the online system to perform the evaluations, they are required to complete it and are not allowed to leave blanks. In the case of a normal evaluation, the student who is not going to perform the evaluation will have the chance to see a window that appears on the evaluation page and by pressing the decline button he can send his blank evaluation. At the same time, the students receive notification to their university’s email address reminding them to carry out the evaluations by explaining the way in which they could unsubscribe from these reminders. The evaluation that has been submitted cannot be revoked, therefore it is unchanged. GSB students also have the right to review their reward by means of a link provided only to members of the GSB community.

The next case that will be presented is the Rochester Institute of Technology or known as R.I.T, also located in the United States [2]. Until 2010, this educational institution carried out an evaluation system with questionnaires on paper, similar to the one that existed at Luarasi University. In 2010, they decided to set up a team that would build an online evaluation system for lecturers. They thought that the form would have five to seven main questions, ranging from 4-5 questions per each main question, where
the answers would be either in the form of multiple choices or with an assessment scale. They would give students at least three days to finish the evaluation. At the end of the evaluation form, they left an open question so that the student could express his/her opinion for anything related to the evaluation. Their evaluation system was compatible with all operating systems (Windows, Linux, Unix, Mac), and with most of the browsers (Google Chrome, Safari, Mozilla Firefox, etc.). The results could be exported to Excel and SPSS depending on the format the user would choose. Something interesting was the opportunity to leave an evaluation incomplete and continue at a later moment. Regarding the log in process, the credentials used were provided by the university, specifically a unique code that the student would obtain at the moment his/her enrolment in the university. During the testing phase, two of the faculties were chosen to measure system performance and student response to this approach they were offering.

The other case studied was the one of the WMU (Western Michigan University) [3].

In this university, the online evaluation system was implemented in 2010. The system was created by (UIUC) the University of Illinois, Urbana-Champaign which was outsourced for this project. In 2007, UIUC proposed the implementation of the system they had at their disposal to WMU. At the time, the project was still in pilot phase and the system was not yet fully tested, this led WMU to postpone the implementation until UIUC obtained the results of the Evaluation System. In the summer of 2009, UIUC started implementing the Evaluation System for WMU. At first, the system was only available in certain faculties and in the remaining faculties the system was implemented gradually. The deadline set for the implementation of the system in all the faculties was the Spring of 2010.

In the Spring of 2010, the pilot phase of the system commenced and lasted until Autumn 2010. Before the system was active and operational, WMU launched an extensive informative campaign. An informative web site was built, PowerPoint tutorials on how to use the system were created, meetings with the university students and faculty staff were held, a user manual was drafted, accompanied by more general questions and answers, newsletters and billboards. This informative campaign aimed at advertising the system, which was implemented for the first time in this university. To the WMU's existing infrastructure were added four HP servers for the system. A feedback button was created, which gave users the ability to write and leave their comments on the system or write about the issues encountered. The evaluation period was defined and limited. The students were notified about the evaluation via email. The student's evaluation could be re-opened and modified. The evaluation form could be saved for later editing. These two features were actually added after being mentioned in the users’ feedback. In the form completed by the students, an optional "no answer" evaluation was added, required by the students who would rather not respond to some questions.

1.1.2 Our System

In all the three cases described above, there is a common feature: in none of them, it is reflected the programming language or the technologies used to create and implement the online evaluation system of lecturers. The reason why the educational institutions
do not provide this information, is to protect themselves from the possibility of their work being copied by others. The source code is by all means closed, and we were not able to consult with the universities regarding it. We made some requests for information concerning their evaluation systems, however all responses were negative.

Nonetheless, each of the cases has assisted us in parts of our implementation. In our system, students that start to complete the evaluation are free to close the web browser at any time without losing the evaluation already given, unlike what happens in the case of Stanford University’s system.

Second, in the evaluation lists are included courses attended even by one or two students in difference of Stanford’s system. Our idea of the system is that everyone who studies at Luarasi University should have the right to evaluate the lecturer teaching them, even if only one student is participating the course. The conception that Stanford University had on the system, with the process of sending reminders to students via email seemed interesting to us. At university, this section was assigned to the coordinators of various departments while still in the pilot stage of the system and, when fully operational, it will include a mechanism for sending an email with a "reminder to carry out the course evaluation" to students.

This email will be sent to the email address that was provided by the university to each student when enrolling. Another idea that we have implemented is the non-modifiable evaluation. This was included in order to increase transparency and security. When the student completes the evaluation a pop-up message appears asking him/her whether to submit the evaluation. If he/she accepts and presses the Submit button, the result of the evaluation is stored in the database and the process is non-revocable. In the case of R.I.T [2], there are structural similarities between the form used in our evaluation system and their form.

In our system, as well as in theirs, we have used the Likert scale for the users to answer. The Likert scale is a good method to present the end results in a report form. The student has the opportunity to choose from a series of written alternatives and in the database are stored in numeric (integer) form. As soon as the report is done, it is easy to perform simpler actions, such as carrying out mathematical actions with numerical variables rather than text. At the same time, the structuring used has four core questions and one open question at the end where the student can comment on the subject or lecturer. The R.I.T case was also a good practice for the assessment system to adapt to different operating systems and with different browsers. This feature made us reflect on the platform to be used to structure the evaluation system. The best selection was PHP & MySQL, which have the ability to adapt to any environment.

Throughout this thesis, it will be explained why we used MariaDB instead of MySQL.
Reports generated at the end of the evaluation process can be exported to Excel, PDF, CSV, and other platforms. In the case of R.I.T, it was available to exported in Excel and SPSS.

Another good idea obtained from the R.I.T’s system was to pilot first with two faculties. The pilot system in Luarasi University was performed with one faculty initially to test the results and afterwards proceeded with the other two faculties. The last aspect that R.I.T's positive practice taught us was the use of university’s credentials to access the system. The student code is unique and given to the student at the time he enrols at the university. We have made this code a necessary condition to access the
system. From the third case of WMU, we have taken mostly procedural aspects rather than the functionality. The information and advertising campaign of the system seemed valuable. We built a user manual, which was attached to the e-mail that was sent to all students, announcing the creation of an Online Course Evaluation System, which would be functional at a specific time period and where all students were invited to take part.

Also, the creation of the system was announced on the website of Luarasi University, in the news section, stating also the period in which the evaluation process was to be carried out. One of the issues with WMU was the ability of students to resubmit a modified version of course evaluation. This ability was considered a violation of the transparency principle of the whole system and its integrity, given that the students could evaluate the first time in accordance to their own beliefs and his assessment in a second moment could be modified by under the influence of external factors, deforming the result.

The best practices listed above have helped us at the conceptual level, which is a key stage in the realization of such a project.

1.2 Web Components

Being that the work we are describing was a Web Application Project, explaining the Web Components is very important.

A web application is an application that uses the web browser as a client. The main advantage of web applications is the lack of necessity for the developer to write a program for a specific operating system and it can be executed in any web browser (though some applications require a specific browser). Web applications generally use a combination of scripting languages—server side (PHP, ASP, etc.) and client side (HTML, JavaScript, etc.) to develop an application. User scripts are responsible for the way the information is presented, while the server scripts handle the retention, finding, updating, and adding information.

To develop this application, we used the Ajax technique, JavaScript, CSS, PHP, and XAMPP.

The reason that these technologies were chosen is the familiarity with all of them, and their ability to cooperate together.

The biggest advantage of AJAX is the communication between the client (Web browser) and the server. Earlier, the communication was limited to those who developed a clean code for the server using languages like Perl and C. New technologies like ASP.NET, PHP and JSP encourage a mix of client-server techniques for software engineers. Server-side developers require to understand more on how server-side technology works in order to build solutions on AJAX.
1.2.1 Ajax

Among the most important features of the web is the ability to execute client codes (generally via JavaScript). Combining the ability to access and modify the client's DOM and the ability to create asynchronous web applications, commonly refer to AJAX technology.

There are many books written about Ajax, but the one we appreciated the most for his simplicity and will use further on to describe the AJAX is the book written by Jeremy Keith. [4] The extraordinary power of AJAX is the communication between the client (Web browser) and the server. The earlier concept of communication was limited to those who developed a clean code of the server using languages like Perl and C. New technologies like ASP.NET, PHP and JSP encourage a mix of client-server techniques for software engineers. Programmers working with the server need to understand more how server-side technology functions in order to create solutions on AJAX.

The AJAX technique combines different elements, as Jeremy also describes in the first chapter of his book [4]:

- HTML (Hyper Text Markup Language) is the language to describe the content inside a web page.
- XHTML to guarantee the correctness and the coherence of documents.
- Cascading Style Sheets (CSS) provides a set of visual style for elements on web pages. Provides a simple and powerful way to define and apply consistent visual styles.
- The Document Object Model (DOM) presents the structure of a web page as a set of programmed objects that can be manipulated with JavaScript. Allows an application in AJAX to immediately modify the user’s interface, retrieving efficiently parts of the page. It guarantees the already saved documents, in order to manipulate structures and content without the need to reopen the entire site.
- XSLT guarantees the structuring of data exchanged between the opened page and the server as well as the transfer of this data if necessary in their visual presentation.
- The XMLHttpRequest object serves to realize asynchronous data exchange between user browsers and web servers. Used in JavaScript, represents the heart of AJAX.
- JavaScript is a dynamic, object-oriented language that has a large number of advanced features. Without JavaScript, there would be no pages, no WEB 2.0 no AJAX.

Indeed, all of these are valuable technologies used in AJAX, but only four of these are necessarily required:

"HTML / XHTML - primary language
"DOM- Dynamic refresh of a WEB page
"CSS- sets the style
"JavaScript- coding language used to program an AJAX engine
1.2.2 Advantages and Disadvantages of AJAX Technique

AJAX is a relatively new technique that is being widely used to implement web applications, and as is usually the case, it is difficult to immediately understand the positive and negative effects of a developing technology. However, many web programmers have embraced this new development and have been using it every day more. The advantages and disadvantages presented below come as a result of using the technique in our project and the research done. [4].

Advantages

User experience: By using AJAX, the user experience in navigation is exponentially improved: it is no longer necessary to reopen the site to follow an action, but only targeted areas appear, allowing the user to simultaneously perform other actions.

Consuming resources: By using AJAX, it is possible to reduce the load on the server and consequently the data to be transmitted. Only the most needed data is processed from time to time by the web server.

Separate Memories: By using AJAX it is possible to split presentation logic from the operation one. The page level and graphics can be independent of server processing. When an application is implemented using a server-side language such as Active Server Pages (ASP) it is necessary to combine the HTML content with those extracted from the database to show the data. It can be contributed to AJAX the possibility to completely split the two parts: the user interface will be completely separated from the engine which makes the questions and returns the data.

Minimal traffic: Applications in AJAX can send and receive the smallest amount of information on the server. This way, they minimize the traffic created between the client and the server, ensuring that the application does not send or receive unnecessary information.

No surprises: Normally AJAX applications represent interaction models of different users from traditional web applications. In contrast to the click-and-wait standard, some AJAX applications use other user interface models, such as drag-and-drop or double-click. Regardless of the model to be chosen, it is important for the user to know how to act afterwards.

Disadvantages

Browser compatibility: Each browser has a different way of processing information and syntax. Unfortunately, not all browsers support AJAX and its components. The biggest disadvantage is the need to replicate the same page to make sure that all users will be able to see these web sites.

Charging Time: When developing on AJAX it is important to consider the network's distance or the interval between user request and server response. If the extension is too large, using constantly HTTP request to make any update of the site, will worsen the user experience.

Search Engine Optimization: Dynamic data import is appealing to users, but it is not the case for search engines that read Web pages that are then indexed. All dynamic components obtained for HTTP requests are not indexed by the engine.
1.2.3 XAMPP

XAMPP is a web package, consisting of Apache HTTP server, MariaDB database and script interpreters, written in PHP and PERL [5]. It is the most used cross-platform, especially among PHP developers. The name stands for Cross Platform (thus, the X), Apache (A), MariaDB (M), PHP, (P), and Perl (P). It is free and open – source, making it one of the most popular PHP development environments for any kind of operating systems; (Windows, OsX, Linux)

It is very convenient while creating dynamic pages to use programming languages like PHP, JSP, Servlets. Due to it being open-source, it is constantly updated to incorporate the latest versions Apache, MariaDB, PHP, and PERL. It also comes with a number of other modules that include OpenSSL and phpMyadmin. One of the main reasons developers and Web Designers use it is because it allows to operate it even without having to be connected in the web. It also provides support for creating and manipulating databases in MariaDB and SQLite among others. Until 2015, before switching to MariaDB it used to also create MySQL databases. The XAMPP version used in the realization of this project is PHP to 7.2.1 / 7.1.13 / 7.0.27 / 5.6.33 MariaDB to 10.1.30 phpMyAdmin 4.7.7. Below we describe briefly each of its constituent elements.

- Apache

Apache Http server is an open source web server dating since 1995. Because of it being open source, it has gained a lot of popularity because anyone can customize it according to its requirements. It has been maintained by a developer community and according to its official webpage [6], is the most used web server since 1996.

- PHP

PHP is a general-purpose scripting language embedded within HTML, mostly used for web development. Its syntax consists of C, Java, and Perl languages and the main use is found in creating dynamic web pages. [7] The use of PHP today continues to grow to an enormous extent where 79% of web pages are written in PHP [8]. The main reason for this usage could be found in the ability this programming language has to be simple, fast and very dynamic. PHP is a relatively new web programming language and like anything else, in its early stages was surrounded by general scepticism in the ability that this language would have to produce concrete results. The language was made available in 1994 when Rasmus Lerdorf first released it and resembled a Common Gateway Interface written in C language rather than a proper programming language [7]. Lewis still worked on and added a set of tools to enrich the language and made it possible to link to databases by creating dynamic web pages. The sequences that followed would be PHP2, PHP3, PHP4, and the most recent and most useful PHP5. The grade of usage of PHP language over the years, lets us understand how useful and important this language is today. In 1996, only 1% of websites were built with PHP, while today this number is 79%. [9]
- PHP Uses on Web Applications

PHP is one of the most coveted languages in the development of web applications. With the elementary knowledge of this programming language, a programmer is still able to create a functioning website. It contains numerous tools that help programmers to reach their objectives. There is a wide variety of uses for PHP. This language can be used for:

a) Server-side scripting

This is the most general use of PHP, since it involves the creation of dynamic web content. It is also used for generating XML, PDF files and graphics among other things.

b) Command-line scripting

Scripts can be run from the command line as in Unix shell.

c) Client-side GUI applications

Using PHP-GTK can write and create full-fledged, cross-platform GUI applications. [10]

The web pages created with PHP are generally HTML pages with integrated PHP functions. The Web server processes PHP commands and sends their output at the browser.

PHP is a language that is object-oriented and as such, takes on special importance as this is an attribute of high-level languages. Being an object-oriented language, PHP uses the concepts of classes, methods, constructor and inheritance. [11]

To understand the reasons why all these uses of PHP are so important, we need to understand first, the object-oriented concept.

“Object-oriented programming is a method of implementation in which programs are organized as cooperative collections of objects, each of which represents an instance of some class, and whose classes are all members of a hierarchy of classes united via inheritance relationships.” [12]

The use of object-oriented method gives developers the ability to re-use previously written code, minimizing the time and costs that would be spent on rewriting the code. The inheritance relations allow class-generated objects to inherit class attributes by eliminating unnecessary repetition and multiple rows of code. Object-Oriented programming is fast and efficient. Two of the most used PHP methods in web development are encapsulation and polymorphism.

Encapsulation is the concealment of data and functions against the client. This is accomplished through the declaration of variables as private or protected. [11]

This is important in case there would be classes with certain attributes, the content of which is better kept hidden as it can create a problem for the user who can access it and may not understand it.
Another important attribute of PHP is polymorphism, which is important because it makes possible to maintain a general interface even where there are different implementations [11].

- MariaDB

Maria DB is a database server, which had its first release on 29.10.2009 after Sun Microsystems bought MySQL. [13]

MariaDB is a diverge from MySQL, which intended to be free and open source from one of the lead developers of MySQL.

Since that day, Maria DB has undergone through many releases and is now one of the most used database servers, growing apart from MySQL every day more, and is now considered to be even faster. [14]

Since XAMPP has already integrated MariaDB, we will be using the interface of PhpMyAdmin to create and navigate in our database.

1.3 Our Project

This project, as mentioned in the Introduction part, started with the desire of Luarasi University to have an online system to evaluate courses and lecturers.

The working team for this project consisted of 5 persons, 3 working on the back and front-end programming, I was working on the Database Design and User Interface, and a Scrum Master. As it can be understood from the separation of work, we adapted Scrum Framework for our project. Throughout the thesis, it will be discussed in detail how we implemented the Scrum Framework into our project and what tools we used to adapt to the Scrum Framework processes.

1.3.1 Objectives of the Thesis

The main objective of this thesis is the development and implementation of the Online Course Evaluation System at Luarasi University. Our goal in this project was to build a comprehensive and usable system for the students so they could be able to use it by the end of the term and for the staff to be able to generate graphs and other statistical data.

This system is going to help the staff of “Luarasi University” by digitizing the process of evaluation, that until now was conducted manually, as explained in the Introduction chapter, and by having better results due to the changes introduced in the process. It is also going to help lecturers understand their shortcomings in methodology or curriculum and is going to help students by providing them with the opportunity to express their opinions on their lecturers and the courses they take throughout the year.
1.4 Thesis Structure

In this thesis will be presented how this system was designed and implemented, which web components and programming languages were used, how the university is managing the new system, and how it affected university life, with the attributes and drawbacks that this new online system carries.

We will go through the analysis of the System Components, the Design Decisions, and will explain each feature of the system. Further on will also be presented the challenges of the evaluation system while designing it.
2. SYSTEM DESIGN

I was responsible to design the Online Course Evaluation System. This work can be divided into two main parts, respectively the Database Design and the User Interface Design. In the chapter below will be explained in detail all the work that was done based on the research performed before and during the work on the project.

2.1. Overview of the Methodology

There are two available research approaches through which research work can be carried out, namely qualitative and quantitative approaches. In many cases, a combination of both approaches can be used to provide better results while conducting research.

As the objective of the research is to examine, interpret, and provide a deeper understanding of a particular problem, in this case, qualitative methods are more appropriate to use. The research strategy chosen for this work is a practical approach to the realization of such a system, based on the experience with other systems we have evaluated and with an illustration of how the theoretical part, obtained during the research, can be implemented in the realization of this system.

The database design methodology I used, is by Connolly and Begg [15]. They introduced three main phases of database design methodology, respectively: conceptual, logical and physical database design.

Throughout this subchapter, we will go through the methodology used for designing the database and give an overview of the three main phases, as Connolly and Begg described them.

The design methodology is a concept used while designing, which acts as guidance on how to proceed. The design methodology should be separated into parts stating the most convenient tools to be used for each. The main role of a design methodology should be to assist the work of designers while going through the phases of database development. One of the most fundamental features of the design methodology is validation, which helps the designers make certain that they have complied in the best way with user requirement specifications.

As described above, we have implemented the database design methodology by Connolly and Begg, to design the database for this project. Their methodology consists of three main phases, starting with conceptual, then logical and lastly the physical database design phase.

The conceptual phase aspires to create a conceptual portrayal of the database required by the client. This phase is the one where entities, relationships, and attributes of the ER modelling are established. Creating the ER model is the main activity of this phase.
On the logical design phase, normalization of the database is the main activity, verifying the ER model for data redundancy, and any anomalies in creating the relations between entities.

During the physical design phase, the primary keys defined, entities are transformed into tables and nature of attributes is established in a table according to the queries to be performed.

The key components of a successful database design summarized by Connolly and Begg [15] are:

• Work interactively with users as much as possible.
• Follow a structured methodology throughout the data modelling process.
• Employ a data-driven approach.
• Incorporate structural and integrity considerations into the data models.
• Combine conceptualization, normalization, and transaction validation techniques into the data modelling methodology.
• Use diagrams to represent as much of the data models as possible.
• Use a Database Design Language (DBDL) to represent additional data semantics.
• Build a data dictionary to supplement the data model diagrams.
• Be willing to repeat steps.

2.2 System Requirements

A requirement is simply a statement of what the system must do or what characteristics that need to be present.

To come up with these requirements we have followed the Business Process Reengineering.

“BPR means changing the fundamental way in which the organization operates, “obliterating” the current way of doing business and making major changes to take advantage of new ideas and new technology.” [16]

The stakeholders of this project are:

- Students
- Administrative staff
- Professors

We have interviewed five representatives, one from each stakeholder, and followed the five basic steps to the interview process: selecting interviewees, designing
interview questions, preparing for the interview, conducting the interview, and post-interview follow-up as prescribed by Brian James.

- Selecting Interviewees
We created a schedule for interviewing everyone and left a fifteen minutes time slot for everyone. The interviews were conducted all in one day.

- Designing Interview Questions
The questions used during the interviews belong to three different typologies: open-ended questions, probing questions, and closed-ended questions.

Open-ended questions are questions that allow the interviewed person to express himself freely and share his/her opinion on the topic.

Probing questions are used in the end, after the questions have been finished and the interviewer wants to learn more regarding one specific question or he wants the interviewee to elaborate his/her answers because it hasn’t been clear enough in the beginning.

Close-ended questions are the type of questions that usually require a simple, short and firm answer; which very often is just YES or NO.

We focused more on the open-ended questions since we needed feedback from our interviewees.

We prepared a Test Questionnaire, involving three tasks for everyone to complete. The tasks included:

- Login in the system using the student number/ university credentials
  We gave them some test student numbers just to see if they understood the login process and if they would have any sort of difficulties to perform the task.

- Select one course you want to review/evaluate
  This task was to verify if the system would recognize the student codes and to see what courses would be shown to them, in the same time understanding how they would navigate through the system.

- Complete the course evaluation
  We wanted to see if they understood the questions in the system correctly and if the evaluation system scale was easy to understand.
Afterwards, we proceeded with open-ended questions, to see if the instructions had been clear, and if they had any kind of difficulties performing the tasks. We asked them and if they wanted to make any modifications in the system so far.

These interviews were held simultaneously to collect feedback for the user interface and the system itself. From the feedback of the interviews we made changes mainly to the interface of the system, making it more polished, by leaving just the name of the course to be evaluated and the name of the lecturer, removing excess data like the department he belonged, name of the head of the department and other courses that this lecturer was teaching.

We thought in the beginning that the latter might help students that had more than one course with the same lecturer, but the students found it to be of no added value.

During the first surveys, we drafted a table of User Requirements and we consulted it with the project stakeholders. After their feedback, as it may be seen in the table below, we have added the user requirement to build the Online Course Evaluation System.

Table 2.1 User Requirements for the Online Course Evaluation System.

<table>
<thead>
<tr>
<th>Requirement</th>
<th>Description</th>
<th>Importance</th>
</tr>
</thead>
<tbody>
<tr>
<td>It should be a web application</td>
<td>The system to be built should be a web application so that the students can access it everywhere, not only in the University Building.</td>
<td>High</td>
</tr>
<tr>
<td>The system should interact with a database.</td>
<td>This system needs to interact with a database since there will be new students, lecturers, and courses every semester.</td>
<td>High</td>
</tr>
<tr>
<td>The system should have a login system for students</td>
<td>Students will need to log in since they will have access to review only the courses in which they are enrolled.</td>
<td>High</td>
</tr>
<tr>
<td>Requirement</td>
<td>Description</td>
<td>Priority</td>
</tr>
<tr>
<td>-------------</td>
<td>-------------</td>
<td>----------</td>
</tr>
<tr>
<td>The system should have a login system for Administrative Staff</td>
<td>The Administrative staff is the one who will be in charge of importing the student names and their courses, and they will be the ones checking the results from the reports that the system will generate.</td>
<td>Low</td>
</tr>
<tr>
<td>The system should have different layers of Hierarchy</td>
<td>The Administrative staff will have different levels of hierarchy. For example, the Rector will be able to see all the results from all the faculties; the Dean will be able to see only the results of its own faculty, the Head of Department will only see the courses of that department and a lecturer will only see the results of courses they teach.</td>
<td>Medium</td>
</tr>
<tr>
<td>The Database with all the courses and lecturers will be created</td>
<td>Since the former process was manual, there is no database, with the exception of the Microsoft Excel files with the students enrolled in each faculty. We will create a database containing the lecturers’ list, courses list and students’ list connected via queries to their respective questions in the questionnaire.</td>
<td>High</td>
</tr>
<tr>
<td>The system should be easy to use</td>
<td>Since the staff and the students are not used to work with similar systems, it is important that the system is easy to use and understandable. This will help more students use it and will generate better results.</td>
<td>Medium</td>
</tr>
<tr>
<td>Requirement</td>
<td>Description</td>
<td>Priority</td>
</tr>
<tr>
<td>----------------------------------------------------------------------------</td>
<td>------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>----------</td>
</tr>
<tr>
<td>The system should be fast</td>
<td>Since the system is planned to be used by many students mainly from their home, having delays in its responsiveness and loading of pages will make the students skip the evaluation. That is why we decided that any page should not take more than 3 seconds to load.</td>
<td>Medium</td>
</tr>
<tr>
<td>Everyone (student, administrative staff) connected to the internet should be able to have access to the system</td>
<td>The system should be usable even when users access it from their phone, tablet or laptop. The system should be working with any Operating System and Web browser.</td>
<td>High</td>
</tr>
<tr>
<td>DB should be secured</td>
<td>Only the administrator has full access to the DB. Everyone else has a restricted view depending on the hierarchy.</td>
<td>High</td>
</tr>
<tr>
<td>Platform consideration</td>
<td>People shall be able to access our application using at least one web browser that is running under Windows, Mac, Linux, Android, and iOS</td>
<td>High</td>
</tr>
<tr>
<td>A sufficient number of requests per second and the number of parallel connections shall be possible without saturating the server.</td>
<td>It should be considered that there will be many students trying to evaluate courses at the same time, and the system should be able to handle at least 100 persons working at the same time.</td>
<td>Medium</td>
</tr>
</tbody>
</table>

### 2.3 User Interface Design
The user interface, as it may be understood in the name, is that part of the application, that the user interacts with. It includes all the forms and questionnaires that the user sees, all the actions the user needs to take to navigate through the system and all the interactions that the user has with the system.

User interface design is described as a five-step process that is iterative, in which analysts often move back and forth between steps, rather than proceeding sequentially from step 1 to step 5. [16]

1. Examine the use cases and sequence diagrams. Interview users to develop use scenarios.
2. Developing the WND.
3. Design interface standards.
4. Interface design prototype.
5. Individual Interface Evaluation.

We created different versions of user interface and as mentioned above, in the section 2.2 System Requirements, we conducted some surveys within the university, taking students, professors, and administrative staff to receive feedback on the interface.

The interface had to be very simple, minimalistic and easy for everyone to use. For these reasons, we decided to involve all the stakeholders as much as possible, by conducting surveys and interviews when needed so that we could have their feedback and validation.

A survey was conducted in this case to decide the User Interface of the system.

We showed to the stakeholders three different versions, and they choose for the one who is now functional. It needs to be said that even this version went through some modifications during the work.

There were two main changes taking place during the work on this project. The first changes were made after the feedback we got from stakeholders after using the demo version of the project.

The other big modification came after we delivered the system and we ran it in the first Live Evaluation.

Because of the feedback, and the concern the students had regarding their anonymity in the system, we changed the login system, as described in chapter 2.5 Design Decisions. Even though it had no major impact on the user interface, we went through a thorough research in order to have as little changes as possible in the system.

The interface for the Login page has remained the same, what changes is the way that the database links the student numbers with the courses. When the system was made available in the beginning, we would use the real student numbers to log them in and evaluate their courses, while in the second semester, we created a system in which the student numbers were made up, according to their own faculty and class and we linked the courses with the class instead of the student numbers. This process will be explained in chapter 2.5 Design Decisions.
In order to Log in, the students need to first enter their Student ID and their email on the page, where the system generates a password based on their credentials. This is done because we didn’t have a way to authenticate the users since their student ID-s were not used in any kind of systems until the time speaking. Below can be seen the screenshot of the password generation page.
After the students receive their password, they are redirected to the Courses page where they need to Log in with the student number and the password generated for them by the system.

After they have logged in, they see the list of courses that they are taking on that particular academic term. As seen in the figure below, the courses appear in the form of a list where each course is represented as a button with the name of the course and the lecturer teaching the course. It was important to have the name of the lecturer present here because the same course is very often conducted by different lecturers and the evaluations could be mixed otherwise. This way, also the students are sure which lecturer they are evaluating.

![Courses to evaluate](image)

After logging in, this is the page that the student sees. In the left corner of the page are to be found all the courses that the student is enrolled in.

The students can choose the course they want to evaluate, and they are redirected immediately in the Evaluations page.
The evaluations page is the most crucial page in this project determining the outcome. If a student would provide a false evaluation because they misunderstood the question or the evaluation system, the results would be impacted, and it would not be very reliable anymore. So, it was very important for us that the questions were clear and that the evaluation system was easy to understand.

Below can be found the screenshot of the Evaluations page.

<table>
<thead>
<tr>
<th>Mësimdhënia</th>
<th>Shume mire - 5</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Pedagogu paraqet qartë programin e lëndës në orën e parë të leksionit.</td>
<td></td>
</tr>
<tr>
<td>2. Pedagogu zotëron, shpjeqon dhe e bën lëndën interesante.</td>
<td></td>
</tr>
<tr>
<td>3. Pedagogu inkurajon pjesëmarrjen aktive në diskutimet e studentëve në klasë.</td>
<td></td>
</tr>
<tr>
<td>4. Pedagogu shpjeqon dhe organizon mirë orët leksioneve/seminarëve.</td>
<td></td>
</tr>
<tr>
<td>5. Pedagogu përcorr në mënyrë efektive dhe të alteruar mjетet audio-vizive.</td>
<td></td>
</tr>
<tr>
<td>6. Lënda nxit interesin dhe mendimin kritik.</td>
<td></td>
</tr>
<tr>
<td>7. Lënda është e organizuar mirë dhe temat ndërthuen natyrshëm.</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Vlerësimi dhe sëqarimi</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>8. Pedagogu bën të qarte mënyrën e vlerësimit të dijeve dhe detyrimat e studentit për lëndën.</td>
<td></td>
</tr>
<tr>
<td>9. Pedagogu este objektiv në respektimin dhe vlerësimin e mendimeve të studentëve.</td>
<td></td>
</tr>
<tr>
<td>10. Kriteret e vlerësimit janë bërë të qarta që në filimin dhe vlerësimi i pedagogut ka qenë i drejtë.</td>
<td></td>
</tr>
</tbody>
</table>
In the leftmost part of the page, it shows the academic year, the semester and the date in which the evaluation is being made.

Underneath it shows the course and the lecturer which are selected by the student to evaluate at the time. The questions are the same for each course and lecturer.

Before proceeding to the questions, there is a statement for the students to read:

“Please answer the questions with honesty and take the evaluation seriously. The answers will not provide your name and will be treated confidentially.”

Even though we had this statement integrated, there was still a lot of scepticism from the students’ part, which made us change the login system as described in chapter 2.5 Design Decisions.

Since the project took place in Albania, the system is mainly written in Albanian. The evaluations page and the questions can be seen in Figure 2.4. Below you can find the translation of the questions in English.

The questions are separated into 4 different groups:

**Teaching**

- The lecturer introduces very clearly the course’s syllabus and program during the first day of lecture.
- The lecturer owns and explains very well the course and makes it more interesting
- The lecturer encourages active participation of students in class.
- The lecturer organizes and explains in a decent and acceptable way its classes.
- The lecturer uses effectively and alternates the audio visible means.
- The subject of the course stimulates the student’s interest and critical thinking.
- The course is well organized, and the topics incorporate naturally and easily in between them.

**Evaluation and performance of the lecturer.**

- The lecturer makes clear the evaluation method and the student’s responsibilities on taking the course.
- The lecturer is objective in respecting and evaluating the opinions of students.
The evaluation criteria are made clear at the beginning of the course and the lecturer’s final evaluation was fair.

**Academic counselling**

- The lecturer offers the opportunity to communicate with the students

**Overall evaluation**

- I am satisfied with the quality of the course.
- The objectives and goals of the course were clear to me and they have been reached.
- The study program complies to my expectations.
- Communicating with the lecturer helped me clear the uncertainties regarding the evaluation and the course.

Below you will find a closer screenshot of the evaluation columns, where the student checks the evaluation that matched their personal experience with the respective course and lecturer.

![Likert scale for Evaluation](image)

**Figure 2.5: Likert scale for Evaluation**

This form of evaluation is called the Likert scale. This scale is an appropriate method to present the end results in a report. The student has the opportunity to choose from a series of answers that are written in words and in the database are stored in numeric (integer) form.

The evaluations would have a grading from 5 to 1, with 5 being Very Good, 4 Good, 3 Enough, 2 Poorly and 1 I don’t know.

At the end of the questionnaire we have also added a comments box, which states: If you desire to emphasize any other positive or negative aspect regarding the course, which you didn’t find above, please use the comment box below.
Just besides, it is the button of conclude the evaluation.

Once pressed, the results are saved in the database and the student cannot re-evaluate the course. If the student wants to continue later with the evaluations, he/she can close the page, and the next time he/she logs in, can continue where it left.
As it is mentioned in the System Requirements, while creating the system we wanted to have also different layers of access, in order to create some sort of hierarchy.

In this case, in the Figure above, this is what the Administrator of the database and the Rector of the university would see, since they are the only ones with the credentials to see all the results and reports generated by the system.

In the database, we incorporated a user ID (caps?), which distinguished the Login credentials of different users.

Student codes are all formatted in the same way, ex.; BAAB1127. When this kind of format would log in, the system knows that this is a student and it only has access in the evaluation forms.

The rector would log in with a username, which in this case was “rector” like the admin had “admin”.

We created a Lecturer ID for the lecturers, so they would log in and see the reports only on the courses they taught. Ex: PAA001

The heads of departments and deans had their own lecturer credentials and we manually added their credentials in the system to be recognized so they could see respectively the results of their department or their faculty.

In the figure above, it can be seen a report of a course from the Law University. This is how a report of a course evaluation looks and what a lecturer sees for each of his/her courses.
In the uppermost part on the left, it is the information regarding the academic year, date, the name of the course and the lecturer that has been evaluated. Also, there is information about how many students have performed the evaluation out of all the students enrolled in that particular course.

Underneath, there are the questions that were posed to the students and on the right is the report generated from their evaluations. In this case, it was a trial version, so the report is empty, but the lecturers see in percentage how the students responded to each question.

This is also what the Dean and Head of Departments would see, but for all the courses they have access.

2.4 Database Design

“DBMS or Database Management System is a software system that allows users to define, create, maintain, and control database access” [15].

DBMS has two important components: DDL and DML. DDL is the Data Definition Language that enables users to specify, according to their needs, the structure, conditions, and type of data to be stored in the database.

DML or Data Manipulation Language allows users to input data, update or extract them from databases. DBMS has an environment where it is developed. This environment has several components: hardware, software, data, procedures, people. At the moment the database was designed, the purpose was to relational data model then the second generation appeared of DBMS: RDBMS or Relational Database Management System. [15]

Databases can be imagined as a major stockpile where information is collected and accessed by many different people seeking to obtain this information. The information in a database is logically connected, this logic is important to understand the concepts of entities, attributes, and relationships. Entities are clearly defined objects (person, thing, concept, occurrence) with qualities that describe them (attributes) and have a link in between (relationships). All databases have attributes, entities and certainly are linked to each other.

If we do not assimilate the concept of Database Design, the probability of building a faulty database is very high. This error, in the long run, takes time to revert back to change and adjust. Database design should be understood as a very early structuring, very detailed stating at how many tables will be built, what attributes will be included, how many queries can be made, how many new fields can we enter in a later period, how large will we make the database and many other questions that we need to find an answer to before starting construction. From our experience in this section, we can say that if you do not answer these questions, you may fail. In the first database I started to build, I missed the calculation of how many entities I would have and created a database with 18 tables. This was a wrong approach because in a second in-depth analysis I realized that some tables could be integrated into one, reducing the time
needed for running the queries and reducing the complexity of the database. Currently, in the final version, the database contains 12 tables.

Second, is normalization. Normalization is the process of allocating your data in the table and creating the primary keys.

The main purpose is to ensure the appearance of any information in the database only once. Data duplication is very ineffective because it makes the database larger than it really should be, and this way, slowing it down. Most importantly, the presence of duplicates creates a strong risk when updating the data as only one of the duplicate data streams will be updated, creating serious problems. To avoid duplicate probabilities, the following tips should be followed:

I. There should be no columns containing the same data
II. Each column should contain only one single value
III. There must be a primary key to identify each table

2.4.1 Building the Conceptual Data Model

Among the key elements that need to be identified for a conceptual model include:
- entity types,
- relationship types,
- attributes and attribute domains,
- primary and alternate keys,
- integrity constraints.

2.4.1.1 Identify Entity Types

As it can be understood, database design is a very structured process and by being so, everything needs to be done at the right moment and it is planned through. That is why we start the first phases of creating the entities in this stage of database design. The information to do so should be available from the User Requirements.
In our final Database Design, as it can be seen from the figure above, we have 12 Entities:
User Type, Students, Students in Groups, Questions, Program, Login, Lecturers, Departments, Course, Course and Lecturer, Academic Year and Results.

2.4.1.2: Identify Relationship Types

Afterwards, the type of logical relations between entities should be determined
For our database, I have identified the following relationships:

- Between Program and Students in Groups: in the entity Studentsingroups, ProgramID is a Foreign Key and receives the information from the Primary Key ProgramID in the entity Program.

- Between Students Table and Students in Groups: in the entity Studentsingroups, StudentCode is a Foreign Key and receives the information from the Primary Key StudentID in the entity StudentsTable.

- Between Departments and Students in Groups: in the entity Studentsingroups, DepartmentID is a Foreign Key and receives the information from the Primary Key DepartmentID in the entity Departments.
- Between Course Table and Students in Groups: in the entity Studentsingroups CourseID is a Foreign Key and receives the information from the Primary Key CourseID in the entity CourseTable.

- Between Results and Students Table: in the entity Results StudentCode is a Foreign Key and receives the information from the Primary Key StudentID in the entity StudentsTable.

- Between LoginTable and Students Table: in the entity Student Table userID is a Foreign Key and receives the information from the Primary Key UserID in the entity LoginTable.

- Between Questions and Results: in the entity results QuestionID is a Foreign Key and receives the information from the Primary Key QuestionID in the entity Questions.

- Between AcademicYear and Course and Lecturer: in the entity Course and Lecturer AcademicYear is a Foreign Key and receives the information from the Primary Key academicyear in the entity academicyear.

- Between courseandlecturer and Course Table: in the entity courseandlecturer CourseID is a Foreign Key and receives the information from the Primary Key CourseID in the entity CourseTable.

- Between results and Course Table: in the entity results Course ID is a Foreign Key and receives the information from the Primary Key CourseID in the entity CourseTable.

- Between UserTypeTable and Login Table: in the entity Login Table UserType is a Foreign Key and receives the information from the Primary Key UserTypeID in the entity UserTypeTable.

- Between courseandlecturer and Course Table: in the entity courseandlecturer CourseID is a Foreign Key and receives the information from the Primary Key CourseID in the entity CourseTable.

2.4.1.3 Identify and Associate Attributes with Entity or Relationship Types

Succeeding the first two phases, the attributes and the type of the attributes should be determined.

UserType Table: User TypeID which is a Primary key and Type.

Login Table: UserID which is a Primary Key, Password and User Type, which is a foreign key.

Course Table: Course ID which is a Primary Key and Course Name.

Lecturers Table: Lecturer ID which is a Primary Key, Name, Surname.

Courses and Lecturers: Course ID which is a Primary Key, Faculty, Academic Year which is a Foreign Key, Period, Class Title, Lecturer ID which is a Foreign Key.
Results: Results. Course ID which is a Foreign Key, Student Code which is a Foreign Key, Date, Question ID which is a Foreign Key.

Academic Year: Academic Year which is a Primary Key.

Program: Program ID which is a Primary Key.

Students in Groups: Student Code, which is a Foreign Key, Department ID which is a Foreign Key, ProgramID which is a Foreign Key, Profil, Full, Course ID which is a Foreign Key, ID which is a Primary Key.

Students Table: StudentID which is a Primary Key, Name, Surname, Email, userID which is a Foreign Key.

Departments: Department Name, Department ID, which is a Primary Key.

Questions: Questions ID, which is a Primary Key, Questions.

2.4.1.4 Determine Attribute Domains

Program Entity:
- ProgramID: varchar(110)

ProgramID is selected as Varchar because it contains values that have letters in it.

Here you have an example:

- Finance-Banke
- Finance-Banke
- Finance-Banke

Figure 2.9: Attribute Domains of program’s entities.
Students in groups Entity:
- StudentCode: varchar(255). The values it contains are in the following format: BAAB1131
- DepartmentID: varchar(100). The values it contains are the names of the Faculty where it belongs, for example: Economy, Law, Information Technology.
- ProgramID: varchar(100)
- Profile: varchar(100). The value it contains is the academic level the student belongs to, Bachelor or Master.
- Full: varchar (200). This value contains the classroom in which the student belongs to.
- Course ID: varchar (10). The values it contains are similar to ECOBAA302.
- ID: int(11)

Departments Entity:
- DepartmentName: varchar(100)
- DepartmentID: varchar(110)

StudentsTable Entity:
- StudentID: varchar(255) The values it contains are in this format: BAAB1131
- Name: varchar(150)
- Surname: varchar(100)
- Email: varchar(100)
- UserID: int(11)

Questions Entity:
- QuestionID: int(11)
- Questions: varchar(255)

Courses and lecturer Entity:
- CourseID: varchar(12) The values it contains are similar to: ECOBAA302.
- Faculty: varchar(50)
- AcademicYear: varchar(50)
- Period: varchar(30)
- classTitle: varchar(100) This value contains the name of the course.
- LecturerID: int(100)
Results Entity:
- Results: int(11)
- CourseID: varchar(10)
- StudentCode: varchar(255)
- Date: datetime
- QuestionID: int(11)

Academicyear Entity:
- Academicyear: varchar(255)

User Type Tables Entity:
- UserTypeID: int(11)
- Type: varchar(110)

Login Table Entity:
- UserID: int(11)
- Password: varchar(255)
- UserType: int(11)

Course Table Entity:
- CourseID: varchar(10)
- CourseName: varchar(255)

Lecturers Table:
- Lecturer ID: int(110)
- Name: varchar(110)
- Surname: varchar(110)

2.4.1.5: Determine candidate, primary, and alternate key attributes

In order for the database to function, it needs to have relationships between the entities. These relations are created via Primary and Foreign keys.
Primary Keys:
- ProgrameID for Program Entity
- ID for studentsingroups
- DepartmentID for Departments
- StudentID for Students Table
- QuestionsID for Questions
- Academicyear for academicyear
- UserTypeID for UserTypeTable
- UserID for LoginTable
- CourseID for CourseTable
- LecturerID for LecturersTable

Foreign keys:
- DepartmentID for studentsingroups
- ProgrameID for studentsingroup
- StudentCode for studentsingroups
- CourseID for students in groups
- UserID for StudentsTable
- AcademicYear for coursesandlecturers
- LecturerID for coursesandlecturers
- UserType for LoginTable
- CourseID for courseandlecturer
- CourseID for results

2.4.1.6 Check Model for Redundancy

Just like data duplication, redundancy is a major problem when creating a database. In this database, the data is connected within tables through foreign keys as explained in the sections above. This helped a lot when creating the database, so we would not have duplicates and also not have similar fields in different tables. We created constraints through foreign keys every time that we thought some information was needed in another table too.
2.4.1.7 Validate Conceptual Model against User Transactions

As stated in the System Requirements, the system should be able to handle at least 100 persons working at once.

2.4.1.8: Review Conceptual Data Model with User

Since we have used the Scrum Framework in this project, this was part of the working process itself. We held surveys and meetings with all the stakeholders of the system.

2.4.2 Logical Database Design

The second phase of Database Design is the Logical Design. In this phase, the goal is to transform the data model created in the conceptual phase, into a logical model, in which the entity types are decided, the relation types and how are they going to be connected with each other.

The model that will be evolved in this phase, will serve as a reference in the final stage, the physical database design phase, where the database will be implemented.

In this stage occurs relationship validation via normalization.

2.4.2.1 Steps taken to design the logical database, as described by [15] in page 462:

- Derive relations for the logical data model
- Validate relations using normalization
- Validate relations against user transactions
- Check integrity constraints
- Review logical data model with user
- Merge logical data models into a global model (optional step)
- Check for future growth

2.4.3 Physical database design

The physical database design is the final process of designing a database. In this phase, the designer decides how to implement the database after being finished with the logical design part.

The main activities in this phase include creating relational tables and putting constraints on them based on the material from creating the logical data model. This is the phase in which are determined the specific storage structures and access methods so that the information manages to attain the best performance for the system. In this phase, the designer should think also about the security of the database.
2.4.3.1 Steps taken to design the physical database

Translate logical database design for target DBMS

In this step, the mapping of the logical database occurs. The goal is to create a polished database design, so it is easy and understandable. This process should be registered and documented because it will be needed in the future, in order for those who have not worked with this system to know the logic behind creating this database.

This is the phase that starts the design of relations that are defined in the logical data model. The attributes need to be defined for every relation created. For this to happen, it is needed to set the data types and default value in each of them. In our project, we started working with the main Entities like Students Table, Lecturers Table, LoginTable, Questions Table, Course Table.

In the case of our database, this phase would translate also in adding the constraint students being allowed to evaluate only once per course.

- Design File Organisations and Indexes

Since one of the key focus of the physical design phase is on the performance efficiency, determining the optimal file organization and indexes is a crucial task.

Among the steps that need to be taken, are the following:

Since we used XAMPP, it already had the MariaDB database installed. As we know Xampp is a web server, so besides the space required to Download and install Xampp it does not take much space, which is a very positive feature.

- Design User Views

This step was particularly important and relevant to our project. We had a multi-user type environment, and we had to be very careful to distinguish firstly ourselves what should each user type do.

After brainstorming with the team and the Scrum Master, we had a clear idea of the user views.

System users are of different levels. There are simple users, administrators, super administrator, deans, heads of departments and the Rector. There are four types of system users in the online evaluation system: simple user, head of department/dean, administrator (like the rector) and super administrator. The simple user accesses the system with username and passcode and performs the evaluation by filling the evaluation form. It is authorized to access the system when the username and password that has is compatible with username and password in the database. This type of user has no privilege in the system except accessing and completing the forms.
The administrator, on the other hand, has almost complete privileges in the system. He/she can generate final evaluation reports, modify form questions. The Rector has the same privileges as an administrator, even though he/she logs in with his credentials.

The super administrator is a person who, in addition to the administrator privileges, can access and manipulate the database, have access and modify the source code.

Deans and head of departments have the same privileges in the system and fall under the same category. They log in with their credentials and are eligible to see the reports on their lecturers.

- Design security mechanism

The two basic aspects of security are physical security and technological security. [17] Physical security means putting the hardware infrastructure under control such as locking servers in a controlled room where access is limited, setting up monitoring cameras, accessing special identifiers such as the retina scanner or fingerprint. In regard to technological security, there are three components: security of applications, security of operating systems and network security. Attacks that can be made to the system are from the most diverse.

The basic principles that need to be considered in terms of security, according to [17] are:

1. Authentication
2. Authorization
3. Confidentiality
4. Integrity of data
5. Accountability
6. Availability
7. Not neglecting

Authentication is the process of verifying a person's identity. The realization of this process can be done in many forms, but the most practical is the username and password. Implementing a security system based on passwords has its advantages and disadvantages. The main advantage is that the password scheme is simple to implement. The biggest disadvantage is that people tend to set short passwords and with very low levels of difficulty to guess. Each user in our system is provided with a password stored in a database, and at the time that the password is entered, a database query is retrieved, in the case of a match, gives access or otherwise refuses access. In the online evaluation system, the students are provided with a username and code to enter the system.
The user must access the system using a username and passcode. These users have only one function in the system, that of completing the evaluation form. They don’t have access to anything else in the system besides the questionnaire. By this, we illustrate the second principle of security: authorization.

The authorization process is very important in system security. In the section above, we talked about the System Users and as mentioned, each type of user had their own level of authorization depending on which category they belong.

The third feature that a system has to have in the security perspective is confidentiality. The purpose of confidentiality is the secret storage of data and communications. This principle has become a basis for developing the evaluation system. In the discussions before launching the project for the development of the online evaluation system of lecturers, as we discussed the possibility of developing such a system we were told that if we were to take this initiative, the whole system would have to maintain the anonymity of the students who evaluated the lecturers.

This was done so that the student as a user would have the full confidence that his evaluation would remain confidential.

The fifth aspect of security is the integrity of the data. Messages and data entering or exiting a system must be protected and secured. Malicious modification or disclosure of third-party data which should not have these data violates the normal functioning of a system. Protocols are needed to avoid this risk. For this purpose, there are cyclic redundancy checks (CRCs) that can be used to achieve data integrity and detect bits that have been lost or modified due to unintentional communication failures. These techniques insert some short codes along with the message, so that when the recipient receives the message with these small parts, he understands that there were interferences.

In the evaluation system, in fact, from the risk analysis that we made in the case of the administrator intervention, we took measures and structured the system in such a way that the administrator has the privilege of just modifying questions and printing reports. When the student finishes the evaluation, he/she clicks a button that says submit the evaluation. When the button is clicked, all the data is stored in the database. The administrator does not have access to the database so there is no way to modify this data. The report it prints is predefined and simply executes a query in the database to display the results without the possibility of modification. Thus, full transparency is achieved, and the evaluation of lecturers made by the students is fair and reliable.

The only possibility that this order would break, is if the threat was to come by a super-administrator, who has full privileges including access to the database. We thought that the mandate of the super-administrator should be in the hands of the university. In this way, the system remains transparent and in case of interference and distortion of the result, will be easier to track down.

Accountability is another characteristic that a system must have in terms of security. Accountability in a system is linked to the process of identifying and assuming responsibilities when information leaks occur, an action that damages the system, or simply a human error. This concept is in fact linked with what we just said above,
when the super-administrator might interfere and damage the results produced by student evaluation.

First of all, the real possibilities that system users themselves can cause damage to the system are small. Students are simple users who have no way of modifying or misusing the system. The administrator as mentioned above cannot modify anything out of the system. His ability to act maliciously in this regard is to print out the results and to modify them with other programs. This action is easily verifiable because the supervisor of the system administrator or the person that this administrator reports to, may require that the report is generated again in his presence. In the case of the super-administrator, which presents the greatest problem, it may, with a statistical analysis be carried out on the data obtained from the database, an analysis of the trend of the grades and whether it is consistent, for example, with the results generated over the years. However, super-administrator is the main suspect in case of misuse in the system.

An available system is a system that responds requests from users at any particular time unit. Availability of the system is very important for organizational units that are in demand for full-time operation of the system. Computer systems may be the victim of various computer attacks. The most popular and most controversial cases of attacks are DDoS attacks or distributed denial of attack attacks. Authors attack maliciously protected or poorly-crafted computing devices by installing malware on these systems. At this moment, these computers are placed under the control of the attacker, which artificially increases traffic on the web site and makes it impossible to use. The servers where the site is located receive these requests, which are actually fictive, and try to respond. As the requirements exceed the normal capacity of the server activity, as the latter attempts to respond to everything, it goes down by not allowing any user to use the service [17]. To eliminate these risks, might be of use a backup server so the company will not be left out of service. Our evaluation system has been developed on a web host, which has physical servers in a country that is not known. In the service we paid we are guaranteed that, in the event of a server, site hosting or web development going down, a different server will be provided in a short period of time. This solution is really a bit costly, but it is efficient. Regarding the availability of the system is specified the date of its use too, as the evaluation of the lecturers takes place at the end of each of the two academic semesters. The system is available for two weeks during which the students conduct the evaluation process.

The last feature a security system must enforce is non-negligence or non-repudiation. For those performing activities in the system, there should be records of their activity. This increases the transparency and accountability of users. If a user is to be suspected for unauthorised activity he may show use evidence to counter the assumed allegation. If the system has the means to present this activity to the person in a documented manner, then the user cannot deny his actions [17].

In the online course evaluation system, simple users cannot hide anything from their activity because none of the students is obliged to make a compulsory evaluation. Regarding the super administrator, when accessing in the cpanel of the web host, it can be caught from the access through IP of the tool used to enter.
2.5 Design Decisions

- Changing the log-in system for the students

After releasing the software and going through the first evaluation phase during February 2018, we received feedback from students that even though we had made clear the Log in process to students, they were yet sceptical about the anonymity of their evaluations, given that they would still Log in with their Student ID and their email address.

This issue had never come up during the interviews and surveys that we conducted with students. Since the process had already started, we decided that it would be better if we proceeded with the current version also to see if this feedback would continue also in the upcoming days. We continued to have low percentages of participation in some of the student classes, and we decided to make some changes in the Login System.

We wanted to try to keep the existing format, because radical changes in the Login process, would mean big changes also in the program and in the database.

After a few brainstorms, we came up with a system that did not change the way that the students would Log in, but in the meantime, it would solve the issue of their scepticism.

It needs to be said however, that this solution would not be suitable for all universities and countries. In Luarasi University the students enrolled in the same Subject will have 90% the same curriculum. There will only be 1-2 courses that the students can select outside the mandatory study plan.

This made us think that we could try to still create a system in which we would create new StudentID-s based on their groups.

We created new lists with different StudentIDs for each subject and academic year, and we already associated them with a password. We would then print them out by always leaving the StudentID and the respective Password together. Since no one knew which StudentID each student had, this was still secure for the system and for the students it meant a more secure feeling that their evaluations were completely anonymous.

The new version of Login for the Lecturer Evaluations was used in June, and we noticed a rise by almost 20% in student participation compared to the one in February.

Even though statistically the new method was more successful, personally I feel like it was a small failure of the system itself. Performing the Evaluations like we did the second time, it brought back some parts of the manual process and the goal of the project was to create a fully online system. This way, we would need to always print made out StudentID-s and students would need to be present at university to pick up their credentials.
- Having different “layers of access”

Having to create different levels of access was a big challenge because we had to keep in mind not only the user view, but also the security of the database.

We had to have a very clear idea of what would each user do and what privileges would each have, in order to build the model securely.

In the beginning, we did not think about the different levels of access since we were not given that as a requirement.

During work, we understood that the system would eventually need different users and that they would have different requirements between them regarding the way they would use the System. That is why we created different user types and different layers of access.

- Database Entities

As mentioned in the Database Design introduction part, in the beginning, while creating the database, it was hard to conceptualise how the system would work. Without having a clear idea on how the users were going to navigate and how the system would function, we made a few mistakes in the first trials of the database. We created too many tables and the database was very complex.

We studied afterwards the data that we were going to upload in the system and understood that the database should be much simpler than what we had created. We merged some tables and created foreign keys to link tables in between them.

Nonetheless, we have still used many bridge tables but that was to make sure that the information was filtered in the correct way and that we would not make any mistakes in linking tables. The amount of bridge tables in the database does not slow down the system and does not complicate the database. It is a way for us to visualise easier the system and for it to be clearer to the other persons who will make changes in the future.
3. SYSTEM IMPLEMENTATION

The online Evaluation system of lecturers is a product obtained from detailed and serious planning based on the most familiar project management techniques. Our selection against this approach came as a result of many factors. At first, the idea was to develop a platform or a system that would be beneficial to the university in general. The discussion was narrowed from the general to realistic opportunities we had to realize them or the real need that the university had, all focused on the idea: Developing an online evaluation system of lecturers at Luarasi University.

The need to develop it properly prompted us to browse the literature in search of the best possible methodology, in order not only to develop the practical aspect of doing things (code writing, database building, etc.) but also to become organized and to conclude this project that we started. To better understand what is to be said, let’s explain what a project is.

The project is a one-time temporary job, with a final output, a start and a finish, and above all with a unique result. [18]

Each project has a life cycle. The life cycle of a project represents the linear progression of a project, from designing the project by creating a plan, to executing the work and closing the project.

The project has four initial stages:

- **Initiation.** This stage begins when a project and a project manager are written in a project chart, and it is completed when project rules are approved. Written approval means that all stakeholders agree on the project's goals, approach, and the balanced time-cost-quality triangle.
- **Planning.** Once the rules are approved, the project manager starts creating the project plan. The way the project is executed may change during the project development period. At the end of the planning phase, all parties not only have to approve the plan, but also any changes needed in the project rules.
- **Execution.** This is the stage of work after the approval of the plan. This stage takes 90 percent or more of the effort in the project. The execution phase will terminate when the project goal is achieved.
- **Closing the project.** This is the shortest phase of the project, but no less important than the others. In this phase are provided the final deliverables of the project, is determined the level of success that the project had and the evaluation of the project is done.

First, we have implemented the project design phase. After deciding what to do we selected the project manager and all the members of the working group. We set our objectives for the project, where the most important was the full realization of the
evaluation system within the pre-established time limits. We established working rules mostly related to the presence and reporting method.

All members of the working group agreed with the pre-established conditions. It should be said that the agreement with the terms was made out of good will and trust, without signing in paper the acceptance of the project rules. At the planning stage through certain tools, we cast a Gant Chart across our activity with deadlines and often with milestones to measure achievements. During this phase, we did not have any changes from the initial part of which we agreed in advance. At the execution stage, the entire evaluation system was developed by writing code in PHP, MariaDB, Java, CSS and so on. This was the most difficult and challenging phase because we were forced to make changes to the reporting system due to the issues with some of the information needed for development, postponing the delivery time by about two weeks. One of the concepts without which we could not have realized the project is the Work Breakdown Structure. WBS identifies all tasks in a project, turning a large, unique project into small, manageable tasks.

The WBS uses project outcomes, risk management and identifies tasks that are the foundation of the all planning made later. WBS can be represented graphically as in the case of using the Gant Chart. [18]

The WBS chart makes it easier to understand all parts of a project. Building a WBS helps in:

• Providing a detailed illustration of the scope of the project. Although the job statement defines the conceptual scope, a more complete understanding of the project is only realized with a WBS.
• Monitoring progress. Tasks in the WBS become the basis for monitoring progress, because each one is a measurable working unit.
• Creating accurate expectations on the schedule and costs. WBS details the costs for equipment, work, and materials in each task.
• Building project teams. Each team member wants clear job assignments and a sense that his/her work is adapted to the overall efforts. A good WBS does both.

In order to have a successful WBS, there should be followed some principles. WBS should be separated starting from the head. This is a top-down decomposition.

Below are listed some activities that need to be carried out to make a WBS successful
- It must be ensured that work packages are part of the project summary tasks.
- Project management standards should be used through specific software.
- The information present in the project should be meaningful with summarized tasks.
- Work packages must be added to the summary tasks. One of the most frustrating planning problems is the omission to introduce the necessary tasks. Avoiding this problem is accomplished by taking care when adding the products of all work packages below each summary.
- Paragraph 8/80. Each of the tasks should not be less than 8 working hours or greater than 80. This translates into keeping work packages between 1 and 10 days. Thus, control over the realization of tasks is greater and the opportunity to improve non-realization increases.
- **Rule of Reporting Period.** No task should be more than the distance between the two ancillary tasks. In other words, if weekly meetings are held, then each assignment should not be longer than one week. This rule is particularly useful when reporting the time to report the status of the schedule. In this way, the status of the assigned weekly tasks will be 0%, 50% or 100% completed.

- The rule "If it is useful". This rule is used for division of tasks into mini-tasks. The reasons why this should be done are:
  a. The task is easier to evaluate. Smaller tasks tend to have less uncertainty, leading to more accurate estimates.
  b. The task is easier to assign to someone. In cases where large tasks are assigned to many people, accountability is lost.

  By dividing the big task into smaller tasks, it can help clarifying the responsibility for each of the persons who take over the task. Another potential benefit is that having smaller tasks assigned to fewer people can give greater flexibility in scheduling the task and resources.

  c. The task is easier to follow. Small tasks are more tangible by increasing reporting accuracy.

With all that we listed above, we actually made a WBS.

It seemed necessary to have a WBS because the development of the online system itself has different programming elements that at each stage we produce a concrete result. Structuring the work was done so that it had a specific weekly assignment for each of the members of the development team. At the end of the week, a meeting was held in which everyone reported on the task he/she was assigned. Each of the tasks itself was part of a larger assignment and any advancement or final product in these mini-tasks, filled parts of the great puzzle of the main task. To illustrate, an example will be presented. The person responsible for the design (me) was charged with the task of creating the design of the entire system. This is one of the most important tasks of the whole project. I was in charge of creating the design and interface of the system and this also had to be done in small pieces, and I was also co-depending on the other people’s work. In the case when the PHP programmer ended a form, I took it and performed its graphical layout. So, in itself, I was performing a small task as representing a PHP form, but on the other hand, it was an advancement towards fulfilling the great task of designing the system.

### 3.1 Using of Scrum for the group management

Scrum is a framework, used for developing, delivering, and sustaining complex products, in which various processes and techniques can be used. [19] The essence of Scrum's use lies in the skills it has to simplify the work of persons, members of a development group in a project. Scrum is a very efficient framework if you follow the guidelines and if everyone knows what they are doing. We needed the system to be ready as soon as possible and we knew that what we were building, was most likely going to change during its development.
For these reasons, we evaluated that following the Scrum Framework it was the best way to work on this project.

After we made the decision, the scrum team was set up, with the main parts being: the project owner (the university), the development team and the Scrum Master. In organizational terms, the scrum team is extremely useful. Scrum teams are self-organized by selecting themselves how to do the job, without being influenced by other people outside the team. This was something that suited best our team. At no point did we have any third-party’s intervention for modification, repair, or anything else in the system. The whole structure was centred on the relationship between team members and the Scrum Master. Another factor that led us to choosing Scrum is that the elements that comprise the development team, at the moment they accept to be part of a scrum team are all equal, everyone has the title of developer and the job evaluation is made for the whole project, not individually. In this way, all personal egos for individual achievements are eliminated, getting rid of the divisions between a person of a higher rank and experienced and the rest of the employees. The most important thing was to increase the team-work to achieve the objectives set.

In Scrum, the main focus is on an activity known as sprint. Sprint is a one-month or less activity in which the scrum team has to deliver a functional part of the final product.

Each of the sprints can be considered a mini project having a definite target and at the end of which, a product should be delivered. The reason why sprints are maximally one-month, is to avoid changes occurring in longer periods of time, which deviate from the original plan and a remodelling of the project has to be done. For the realization of sprints, a "sprint plan" is carried out. All Scrum team members decide on the plan of the next sprint. Generally, in a 1-month sprint are 8 hours available to report, analyse, develop parts of the product that will be delivered at the end of the sprint. Scrum Master strictly controls the compliance of directives and schedules. Using sprints was helpful, for the very fact that the system we were creating would provide visualization of progress on the project. Each of the group members had the opportunity, at the end of every sprint, to demonstrate his/her work and at what stage of the development of the entire project had come. Upon completion of the sprint, a sprint review is realized, where the development team together with the Scrum Master and the university’s representatives in our case analyse the accomplishments made during the sprint. These meetings last for almost four hours, in an ordinary environment, as it is an informal meeting, and discusses what has been achieved, but also what will be done in the next sprint. The Scrum Master presents all the issues that have emerged during the sprint, which have been solved and how they have been solved, whether there has been overtime, budget overspending, and other issues. After discussing in the sprint review, the Scrum team compiles a retrospective report on the things that went well, the things that went wrong, and compiles a plan on how to improve in the next sprint. This part is very important to apply when it is such team. [19]

Using Scrum was a good find for the very simple fact that the system we were creating would provide visualization of progress at the moment of the project. Each of the group members had the opportunity, at the end of every sprint, to demonstrate his/her work and at what stage of the development of the entire project had come.
Scrum teams are self-organized by selecting themselves how to do the job without being influenced by other people outside the team. This part was applied precisely to our team.

Often, especially at the initial stage of the project, we had to go to the after-sprint meeting and understand that we had done some things in the wrong way. The Scrum Master always instructed us to create these personal and group reports later, in order to identify our mistakes, fix them, and have a comparative basis in continuity. In this way, the repetition of errors was eliminated and the mistakes that were made, were documented. If the same mistakes were to occur again, we had them written down in these reports and we could fix them quickly and efficiently.

In addition to the Scrum's one-month-long sprints, there are so-called "Daily Scrums" that are nothing but daily 15-minute meetings aimed at synchronizing the work of members of the development team and predicting what needs to be done and what will be done at the next meeting. We used this technique mostly at the closing stage of the project, in order to identify all deficiencies, problems and find a quick solution to them. Frequent end-of-session meetings occurred as we had to deliver the final product within predefined time limits.

Scrum has three main pillars: transparency, inspection, and adaptation. Transparency means finding a common language between the workgroup and the client or among the members of the team itself. This is very important because if there is the use of different terminology, then the relationship between the system seeker and the developer group can encounter problems and may not allow the client to accept the product. To minimize the possibility of occurrence of any misunderstanding, the Scrum Master itself realized communication with the university, so that the language of the dialogue was the same and we ended up accomplishing exactly what the university wanted. For the inspection part, we were also supervised by the Scrum Master, who with his experience made very professional expertise of our work.

The third stage of adaptation concerns deciding whether to incorporate or not certain parts of development in the final product. It may happen that during the development, some product components are no longer suitable and need adaptation or, in the worst case, they need to be eliminated. In our case, the changes came after the implementation of the project. When we started the Evaluation Process, the students were sceptical about the Login system and even though they Logged in with their credentials, they were not trustful that the evaluations were still anonymous. We continued the first trial as we had planned, and we started working on the end of the project. The closing stage, when the final product is delivered, a meeting with the Scrum team, Scrum Master and the university was realized. In this meeting, it is needed for all the parties to be present, in order to sign an agreement that the process is now finished and the required job is done.

3.2 Tool-set for Project Management

Projects carry many unpredictables. There are many variables that need to be defined and working methods are needed to measure the progress or achievement of objectives.
in a project. For these reasons, tools or work practices that enable these measurements to be made and to determine these variables, have been created. In the online evaluation system of lecturers at Luarasi University, we needed to use these tools to carry out our project for this evaluation system. The first handicap to appear when designing a project, is managing time and tasks.

Time management is very important because, as mentioned above, each project has a starting and finishing date and generally the time available for project development is defined and must be respected. To manage time and to distribute the timeline, all the activity that will take place is a concept known as the "Schedule". The schedule of a project specifies in the unit of time all the activity that will be performed from the starting date to the end of the project. This document must be flexible in order to adapt to the changes that may occur in the project [20]

The Schedule is an integrated part of any planning. All the schedules, whether to plan a party or update a webpage, serve three goals.

The first is to obtain the people's commitment when things are to be done. The schedule provides a contract between each person involved, confirming what each person will deliver over a specified period of time. In general, when people think about project schedules, this is the first thought that comes to mind.

The second goal of a schedule is to encourage everyone seeing their own efforts as part of a whole. Until there is a schedule for a project that suggests specific dates and moments when things should be ready, it is unlikely that links and dependencies will be verified. Without a schedule, everyone will focus on their duties and no one will think about how personal work will affect the work of others. Only when the data is written down, with the names of people besides, calculations can be made, and assumptions can be considered. This is true even for small teams or for individuals working alone. There is psychological power on a schedule because it publishes the commitments made. It's not that easy to forget or ignore something when it's written somewhere on a board, on a sheet that reminds the team of what needs to be done.

By a schedule, questions that can be raised about how realistic some things are and how comparisons can be made, are answered using the analysis of what was requested and what was accomplished. Schedules are binding functions in the projects. If properly used, they force everyone to think carefully about the work they need to do. This binding function is a critical step towards the realization of the project. Even if the schedule slips, is doubled or is halved, the dedication and relations that everyone has made as a result of drafting the plan, can be maintained. So, this second goal of a schedule can be achieved and can be completely valid even if the file itself can become inaccurate. For example, if the project comes in too late, the existence of a schedule will enable completion of the project. The third purpose of the schedule is to provide a tool for tracking the progress made and sharing work in mini-tasks. Separating tasks in one-day or two-day tasks, helps people to understand what they need to do.

Tools used to manage time and organize practical activities are numerous, but our selection would be a well-known and internationally recognized system like Asana. Asana is an online software project management program, which is designed to assist a project manager in developing a plan, assignment of tasks, tracking progress, budget management, and analysis of workloads.
4. SYSTEM VALIDATION

Validation is one of the key aspects of the design methodology as it helps to ensure that the produced models accurately represent the user requirement specifications.

We have validated the system through 3 different methods:

- Normalization
  As mentioned in the Chapter 2.4 Database Design, we used the normalization process to validate relations on the database. This way we know that the database has been created efficiently.

- Use cases
  Use cases are a small illustration of the system’s attributes from the perspective of users. Use-case diagrams are functional diagrams which describe the basic functionalities of a system like: What can the users do in the system, what actions can they perform and how the system is supposed to perform to their actions.

  To create a Use Case Diagram, it is needed to go through 2 different phases. The first one, is getting together with the users and the project team and discuss. In most cases, in this phase we will have some written use cases. The second phase is the translation of these use cases from text to diagrams from the project team.

  We utilized use case diagrams to validate the system design. We had three meetings with stakeholders regarding the use case diagrams.

  In the first one, we shared concepts and ideas on what were the basic functions of the system and how would they be displayed.

  In the second one, we showed them the use cases we created according to the first meeting and gathered feedback.

  In the final meeting, we had a summary meeting, where we showed the final version of the system and gathered feedback.

- Surveys
  Surveys held an important role in this project since it was the principal way in which we validated our work and design decision. Through surveys, we understood firstly what the stakeholders needed, and afterwards we got the feedback on the work.

  Since this was a real-life project, they are the ones who are now using the system, so it was important that we got to understand each other. We also used a “dummy computer” Processor - dual core @ 2.4 GHz and RAM - 2 GB to validate that the system would not crash and that any student could use it from any computer.
5. CONCLUSIONS

The main purpose of this thesis was to demonstrate the implementation of theories and best practices in the designing of the project of the Online Evaluation System of Lecturers. Throughout the paper, I have tried to answer the main research question: How to implement an online system to evaluate the lecturers at Luarasi University.

Today the product is concrete and tangible, but the way and the steps to arrive at such a conclusion were very important. The realization of an online evaluation system of lecturers was made possible thanks to an intensive study of the best Western contemporary literature and practices. Thorough research has been made to learn more from the best practices in this area. Currently, the system contains all the components described in the chapters of this thesis, from PHP to MariaDB, CSS to Ajax, HTML, and XML. The system has implemented the principles of security, databases structuring, and data management.

For us, it had an increased significance that this was a concrete project, and we worked as a team as in any other project of this magnitude. In the evaluation system, we applied the Scrum methodology and tools like Asana, in order to be effective and accurate in the project realization. The online evaluation system of lecturers at Luarasi University is a concrete example of the application of web development principles.

The system has helped end users generate reports with just the click of a button. Earlier, when the evaluation was done through hard copies, the university consumed considerable time to prepare the Excel files. When the system was implemented at first, the trials were conducted at the university’s lab facilities and it was still difficult to generate reports, since the access had to be obtained from the IT department. Today with this system, the administrator will be logged on to a separate page using the access codes and after the completion of the evaluation phase, the reports can be generated instantly.

Thus, this system has helped end users with their organizational work. This system increases the participation of students in the evaluation process and indirectly the performance of those who are tasked to carry out the evaluation. This benefits the students who can do the evaluation at their convenience within the period they have been assigned. For lecturers, this system minimizes the interruptions during their courses by the university administration distributing the paper-based evaluations. At the same time, the professors can receive the results of their evaluation through a more transparent approach. The system is inclusive and serves not only end-users, but also other people outside who are indirectly benefiting from the activity of the system.

However, it needs to be said that due to the students’ scepticism, the process is not fully online, because of the changes made in the login system. Until than the student needs to come and pick up the login credentials at the university.
6. REFERENCES


