**Outline:**

**1** Introduction…………………………………………………………………….2

**1.1.**  Problem………………………………………………………………….2

**1.2.** Solution…………………………………………………………………..2

**2** Objectives………………………………………………………………………3

**3** High-Level Functionality………………………………………………………4

**3.1.** Application Overview……………………………………………………4

**3.2.** Functional Requirements…………………………………………………4

**3.3.** Non-Functional Requirements……………………………………………5

**4** Stakeholders……………………………………………………………………5

**4.1.** Students…………………………………………………………………..6

**4.2.** Lecturers………………………………………………………………….6

**4.3.** Scrum Team………………………………………………………………6

**4.4.** Student Affairs……………………………………………………………6

**4.5.** University Administration………………………………………...……...6

**4.6.** Publishers…………………………………………………………………6

**4.7.** Scholarships……………………………………………………………....6

**4.8.** Career Center……………………………………………………………..6

**5** Process model…………………………………………………………………..7

**6** Project Risks……………………………………………………………………8

**6.1.** Likelihood Risk List……………………………………………………...8

**6.2.** Impact Risk List………………………………………………………….10

**6.3.** Combined Risk List………………………………………………………11

**7** Project Needs…………………………………………………………………12

**7.1.** Hardware Needs………………………………………………………...12

**7.2.** Software Needs…………………………………………………………12

**7.2.** Support Needs…………………………………………………………..12

**8** Measurements…………………………………………………………………13

**9** Tools……………………………………………………………………………13

**9.1.** Database Tool Cost/Training/Functionality Data………………………..13

**9.2.** Normalized Database Tool Cost/Training/Functionality Data…………...13

**9.3.** Normalized Database Tool Graph………………………………………..13

**9.4.** Additional Tools………………………………………………………….14

**10** Project Schedule……………………………………………………………....14

**11** Project pay-offs……………………………………………………………….16

**12** User Interfaces………………………………………………………………..16

**13** Conclusion……………………………………………………………………21

**1 - INTRODUCTION**

**1.1. Problem**

Currently, universities use web-site systems to support students and faculty in their academic activities. These systems facilitate tasks such as course enrollment, messaging, and viewing announcements for students, while enabling instructors to upload materials and manage their courses. However, these functionalities often come from different applications and software, which are not unified into a single system. In addition, different universities use different applications for their systems.

Moreover, nearly all of these systems lack social features, and while, more than 1 application for 1 university makes it challenging for students and faculty to navigate between multiple platforms.

**1.2 Solution**

To address the navigation issues faced by students and faculty, we propose unifying multiple universities and their functionalities into a single, unified system. This system will be accessible via the internet and through a mobile application. Our primary motivation for this solution is to make people not struggle to use multiple systems, which can indirectly lead to contractual lock-ins and high maintenance costs. By offering a cost-effective, all-in-one functional system, we aim to streamline university operations and reduce expenses.

Additionally, to enhance the social environment, we will upgrade existing profile systems by adding personal background sections, profile icons, and photos. We will also introduce student communities, which will be created and maintained by students based on specific interests or contexts. These communities will enable students to seek help, interact with other people, and improve their social skills.

After the development phase, universities can sign up by contacting us, and we will ensure that only verified students from registered universities have access to the system.

**2 - OBJECTIVES**

Here are some of the objectives on this Project that we aimed to achieve both after launch and in development process.

1- Designing a simple GUI so that everyone can understand the system easily. For example, without any knowledge, users can find a way to access the help section about the platform. This can be measured by surveys or customer feedback.

2- Implementing everything according to plan with the most efficiency and safety. The system must be implemented so that usage of resources is minimalized without loss of functionality, and in any type of big issue (system slowdown due to overusage, DBMS crash), the system must ensure users receive minimal harm.

3- Average response time for course enrollments below 60 milliseconds.

4- Existing communities should immediately transferred from old system to our new system after launch.

5- Security of application should be maintained continuously by us.

6- After launching the project, we expect to get at least 2 universities working with us in the first year.

**3 - HIGH-LEVEL FUNCTIONALITY**

**3.1. Application Overview**

Our developed software is mobile application which has mainly 5 parts, more shown in gui part of report. And explained briefly in here.

1- Mails:

Users can recieve or send mails to any people here.

2- Announcements:

Students or teachers will get announcements and can read from here.

3- Community:

Students can create or join communities for diverse events.

4- Courses

Students or teachers can download-upload materials here, and students can enroll courses or any kind of course actions done here, from online-asynchronous courses to document requests.

5- Profile

People can write public personal informations here and upload photo, icons and this section can be displayed in the community section.

**3.2. Functional Requirements**

Here are some of the functional requirements that we specified and explained briefly here. Some of them explained in the overview.

1- System should provide managing course materials to teachers.

2- Asynchronous lectures.

3- Announcements.

4- Providing profile section.

5- System activity report:

System must save activities of communities end of every month in database, these informations will stay at least 6 months. And stay accessible from authorized peoples.

6- Mailing system for communicating with both students and teachers.

**3.3. Non-Functional Requirements**

Here are 6 non-functional requirements specified for Project.

1- Easy to navigate user interface, even for the first-time users.

2- System should be available for users 24/7 with at most 5 second downtime.

3- System must give response not longer than 0.1 seconds whenever user sent to email anyone, or in course enrollments.

4- When accessed to courses section, system must ask additional pin code for more security.

5- System must maintain the performance while getting larger by new universities added to system.

6- System should support both iOS and Android on mobile application.

**4 - STAKEHOLDERS**

There are eight types of stakeholders in this project;

* Students
* Lecturers
* Scrum Team
* Student Affairs
* University Administration
* Publishers
* Scholarships
* Career Center

**4.1. Students**

Students have benefit from the software throughout their courses. Since our software is based on academic purposes, they are the most important internal stakeholder group. They are going to benefit from the richness of content and information, easiness of accessing materials visually. They also can give feedback to development team about design of the program.

**4.2. Lecturers**

Lecturers are one of the financial stakeholders of our software. They have benefits from the popularity of their university and if they are independent lecturers, they directly have benefits from the popularity of the software. They are going to find the possibility of spread their fame and advancing in their career.

**4.3. Scrum Team**

As they have benefit in the development, scrum team is also the direct creator of the project, with the suggestions of other stakeholders such as students and lecturers. If the program is successful, they are going to earn a lot of respect in software community, and they are going to be more preferable for upcoming projects in the future.

**4.4. Student Affairs**

The student affairs office is an important stakeholder of the project in real life, as it is the primary communication center that provides services such as accommodation, scholarships and academic consultancy for students who have registered or will enroll at the university.

**4.5. University Administration**

University administration is the main stakeholders responsible for the general operation and management of the institution. These may include positions such as the university rector, department heads and the board of directors. Management group establishes the policy of the university by overseeing the budget and academic programs. They evaluate the general effectiveness of the student information system and make strategic decisions.

**4.6. Publishers**

Publishers are stakeholders who produce and distribute textbooks, supplementary resources, and academic materials. They develop textbooks and related resources by collaborating with universities. They also provide valuable data on updating materials in the system.

**4.7. Scholarships**

Scholarship offices are units that provide financial support to students at the university and manage scholarships. They can use the system to monitor the academic performance of students and determine whether they meet the scholarship requirements.

**4.8. Career Center**

Career centers help a university on achieving career development, employment processes and professional goals. They provide internship and job opportunities and career counseling to students. This center is an important stakeholder in terms of student connections, career training and development.

**5 - PROCESS MODEL**

We selected scrum for this Project which is simple framework and theory for general value creation with group of people.

Reasons to choose this model listed below;

1- Flexibility, since our Project related with education and education linked with too many regulations, and practices. We need to update our application continously and maintain the validity of the Project.

2- Team collaboration, scrum makes teams more effective in agile aproaches. In this Project we can only maintain these team effectively by using scrum.

3- Risk management, scrum enables us to see and identify risks earlier and take action accordingly.

4- Overall scrum enables team members to take responsibility and increase their motivations during Project development and sprints are makes team members stay focused on goals.

We have done sprint planning as follows;

Sprint 1: Course section

Course section’s interface and pin section implementation with material management, document requests, asynchronous lecture support.

Sprint 2: Profile section

Profile section gui with public personal background information section, photograph taking functionalities with course completion badges, and achievement indicators.

Sprint 3: Community section

Community section implementation in gui, which students can join communities they want and they can have discussios here, exchange ideas, joining events etc.

Sprint 4: Communication

Communication will be provided mostly by mails section, in this sprint we planned implementation of mails, how people will use it and gui design.

Sprint 5: Reports and analysis

Reporting course completion rates and students progress systems will be developed.

Sprint 6: Security

All necessary systems for log in and registeration to system. Seperate authorization levels will be determined like administrator, student, teacher etc. User passwords will be stored with necessary encryption methods.

**6 - PROJECT RISKS**

**6.1. Likelihood Risk List**

**metin, ekran görüntüsü, yazı tipi, doküman, belge içeren bir resim

Açıklama otomatik olarak oluşturuldumetin, ekran görüntüsü, yazı tipi, sayı, numara içeren bir resim

Açıklama otomatik olarak oluşturuldu**

**6.2. Impact Risk List**

**metin, ekran görüntüsü, sayı, numara, yazı tipi içeren bir resim

Açıklama otomatik olarak oluşturuldu**

**6.3. Combined Risk List**

**metin, ekran görüntüsü, sayı, numara, paralel içeren bir resim

Açıklama otomatik olarak oluşturuldu**

**7 - PROJECT NEEDS**

In this project, process is going to require some hardware, software and support needs.

**7.1. Hardware Needs**

* Memory
* Processor
* Graphics card
* Monitor
* Input devices

**7.2. Software Needs**

* Integrated development environment (IDE)
* Version control system
* Collaboration and communication tools
* Unit testing frameworks
* Database management systems
* Documentation tools
* Design tools

**7.3. Support Needs**

* Technical support for users
* Update and maintenance support
* Technical support for development team
* Database management
* Tool learning sources

All need’s feature may change as the project advances, because new stakeholder requirements may occur, some unexpected delays may happen and our current system may not provide stability and required sufficiency.

In order to avoid imbalances, we are going to consider features/resources/time triangle.

çizgi, metin, üçgen, yazı tipi içeren bir resim

Açıklama otomatik olarak oluşturuldu

**8 - MEASUREMENTS**

There are different software measurements needed in this project:

* **Features:** Total number of features added in the current development cycle
* **Effort on training:** Total amount of hours to spent on each tool that used in the project.
* **Requirements in backlog:** Total number of new user stories added to the backlog.
* **Amount of API:** Total number of APIs integrated into the project.
* **Optimization effectiveness:** Impact of each optimization on system performance
* **Time spent on performance testing:** Total number of hours spent on performance testing throughout the project
* **Amount of fixed bugs:** Total bug that are fixed
* **Communication effectiveness:** Number of missunderstanding found in meetings
* **Security:** Authentication system assessment

**9 - TOOLS**

Software tools are chosen by using a comparison graph:

**9.1. Database Tool Cost/Training/Functionality Data**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Tool | Oracle Database | IBM  DB2 | Amazon RDS | Microsoft SQL Server |
| Cost | $3,454 | $969 | $6,240 | $1,260 |
| Training Days | 14 | 11 | 13 | 17 |
| Functionality | 85 | 40 | 90 | 70 |

**9.2. Normalized Database Tool Cost/Training/Functionality Data**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Tool | Oracle Database | IBM  DB2 | Amazon RDS | Microsoft SQL Server |
| Cost | 55.14 | 15.58 | 100 | 20.16 |
| Training Days | 82.35 | 64.71 | 76.47 | 100 |
| Functionality | 94.44 | 44.44 | 100 | 77.78 |

**9.3. Normalized Database Tool Graph**

We decided to use Amazon RDS, because its managed service and cost- effectiveness. Its higher functionality rate to lower training time ratio is an important reason why we chose it.

**9.4. Additional Tools**

* **Microsoft SQL Server:** Microsoft SQL Server is a relational database management system (RDBMS). Applications and tools connect to a SQL Server instance or database and communicate using Transact-SQL (T-SQL).
* **IBM DB2:** IBM® Db2® is the cloud-native database built to power low-latency transactions and real-time analytics at scale. Built on decades of innovation in data security, scalability and availability, you can use Db2 to keep your applications and analytics protected, highly performant and resilient, anywhere.

**Oracle Database:** Oracle makes software, called database management systems, to create and manage databases.

**10 - PROJECT SCHEDULE**

We created gantt chart for our Project and Scheduled the Project tasks. Here is the explanation of Schedule tasks shown below.

1. Project management. (41 days)

In this task we firstly define goals in 5 days and conduct meetings with stakeholders, we will review requirements and plan the Project in detail.

1. Training. (37 days)

This part consist of training of selected tools fort his Project.

1. Design. (17 days)

Designing ui models, database schema and defining system architecture.

1. Test case preperation (12 days)

Specifying test plans and writing test cases.

1. Implementation(131 days)

Longest task in our Project, consist of implementation of all gui parts, and functionality.

Gantt Chart is shown below;

A screenshot of a computer

Description automatically generated

Fig.1 Project-management to implementation.

A screenshot of a computer

Description automatically generated

Fig.2 implementation.

A screenshot of a computer

Description automatically generated

Fig.3 Implementation. (cont.)

**11 - PROJECT PAY-OFFS**

If the project is successfully implemented;

• All universities will be collected in a single application

• University lecturers will have the opportunity to make themselves known on a global scale

• Students will have the opportunity to socialise on a global scale

• Team members are going to obtain priceless professional experience

• These payoffs will maximize the production quality and performance of team members.

**12 - USER INTERFACES**

**metin, elektronik donanım, ekran görüntüsü, mobil telefon içeren bir resim

Açıklama otomatik olarak oluşturuldu**

**metin, ekran görüntüsü, sayı, numara, paralel içeren bir resim

Açıklama otomatik olarak oluşturuldu**

**metin, ekran görüntüsü, mobil telefon, mobil cihaz içeren bir resim

Açıklama otomatik olarak oluşturuldu**

**metin, ekran görüntüsü, mobil telefon, mobil cihaz içeren bir resim

Açıklama otomatik olarak oluşturuldu**

**metin, ekran görüntüsü, mobil telefon, multimedya içeren bir resim

Açıklama otomatik olarak oluşturuldu**

**13 - CONCLUSION**

In this report, we discussed all of the work done for our Software Project UniLearn. We have discussed problem definition, Project objectives, requirements and overview of the Project, scheduling, risks, measurements, etc. clearly.