SOFTWARE DESIGN DOCUMENT
for
E-SYLLABUS

Yunus Tufan Bozkurt

Hazal Öykü Şen

Merve Nur Telli

Özge Tula

Table Of Contents

Table Of Figure

1.INTRODUCTION

1.1 PURPOSE:

This Software Design Document (SDD) describes the detailed structure of the components of the e-syllabus. This includes the architectural features of the system down through details of what operations each code module will perform and the database layout.

1.2 SCOPE:

Our program is a computer application which helps lecturers to create, export, import and edit syllabus with both English and Turkish language support. The program based on a relational database with school web site and application. The data which given by user held in a database for every use of that data.

2.SYSTEM OVERVIEW

2.1 FLOWCHART:

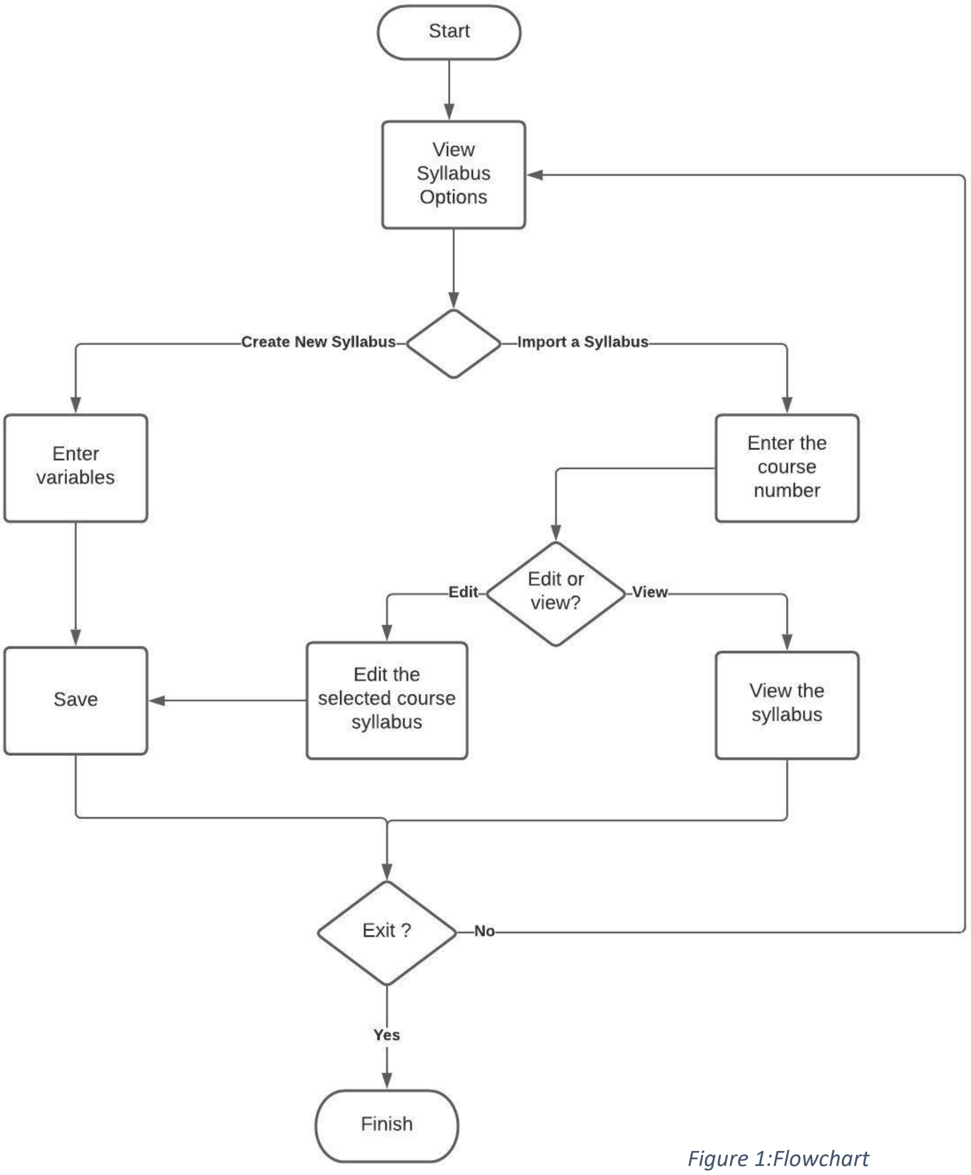


Figure 1:Flowchart

3.DESIGN VIEWPOINTS

3.1.Context Viewpoint

There is only one kind of user in our system. The system supplies four services to them, namely view, import, export and edit syllabus.

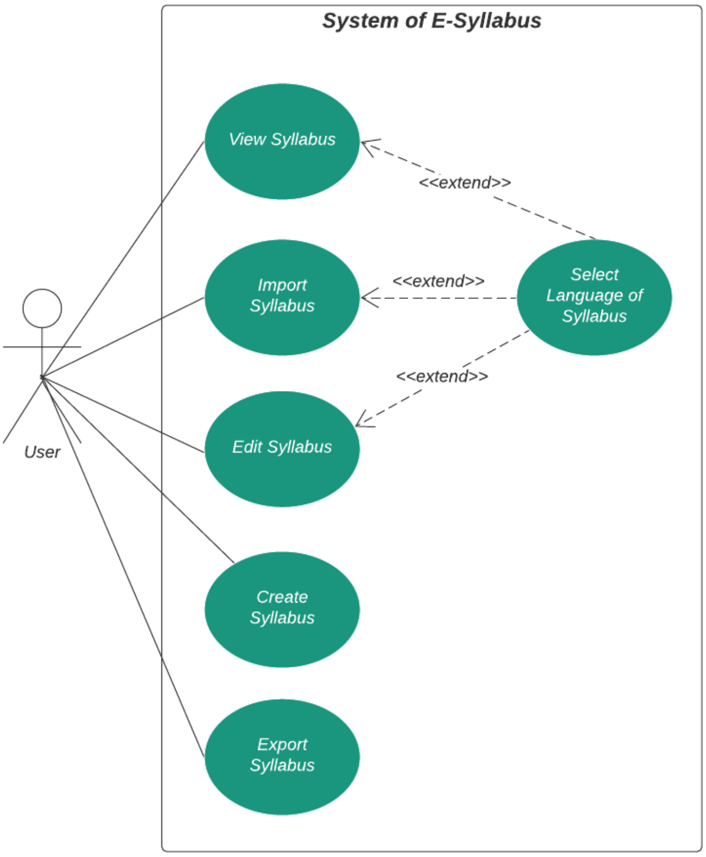


Figure 2: Use Case Diagram

3.1.1. View Syllabus

The user can view the curriculum of the course they want by selecting their language.

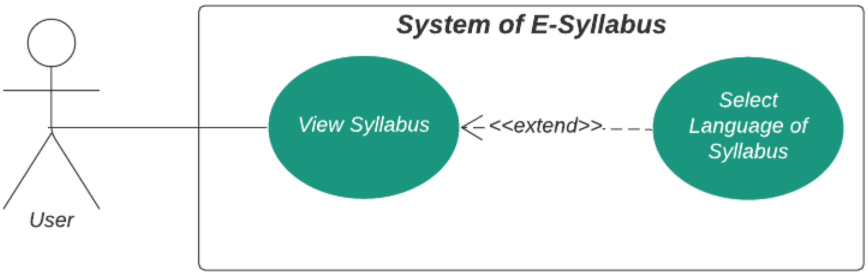


Figure 3: View Syllabus Use Case

3.1.2. Import Syllabus

The users can import the curriculum of the course them wants to the application by choosing language.

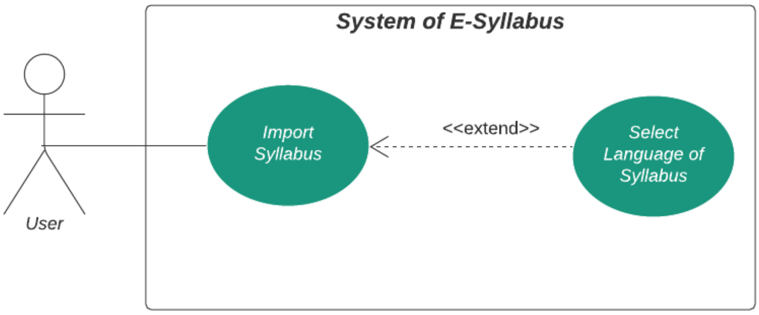


Figure 4: Import Syllabus Use Case

3.1.3.Edit Syllabus

Users can make changes in the curriculum table on the course they transfer by choosing the language.

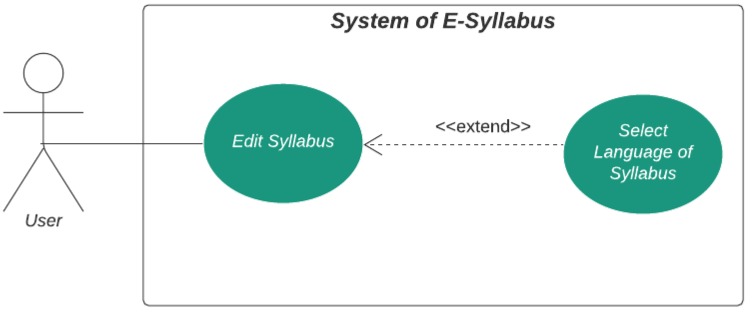


Figure 5: Edit Syllabus Use Case

3.1.4.Create Syllabus

Users can create a new syllabus from scratch.

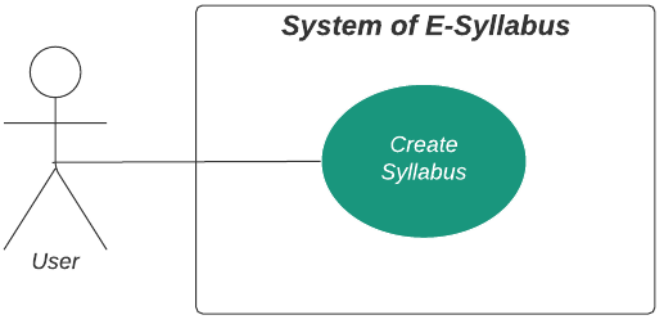


Figure 6: Create Syllabus Use Case

3.1.5.Export Syllabus

Users can export syllabus created from scratch or edited.

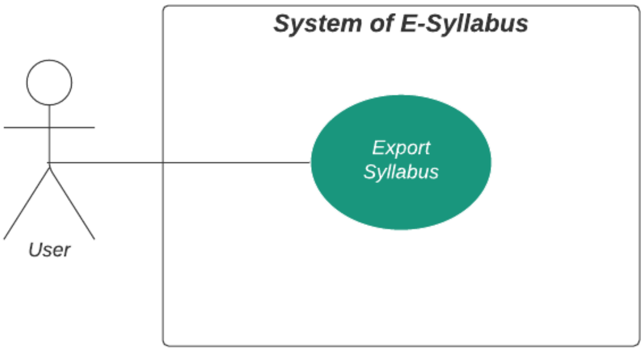


Figure 7: Export Syllabus Use Case

3.2.Logical Viewpoint

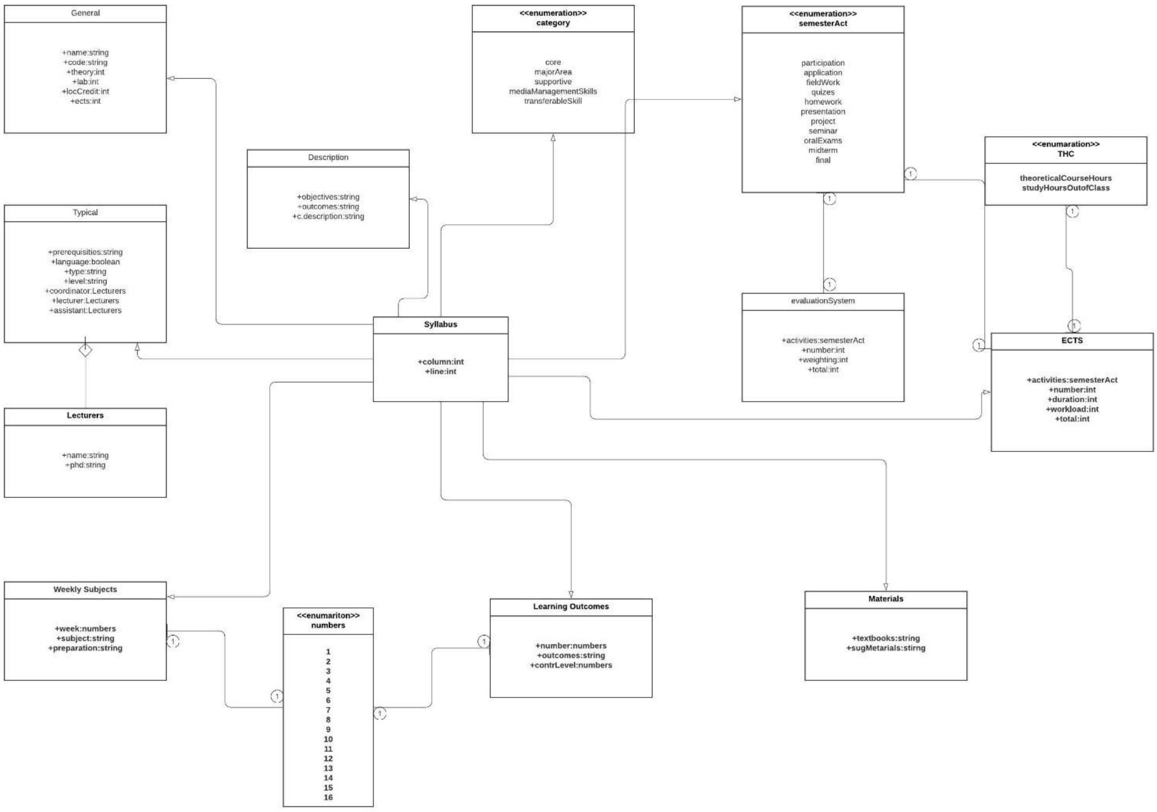


Figure 8:Class Diagram

3.3.Interface ViewPoint

3.3.1.User Interface

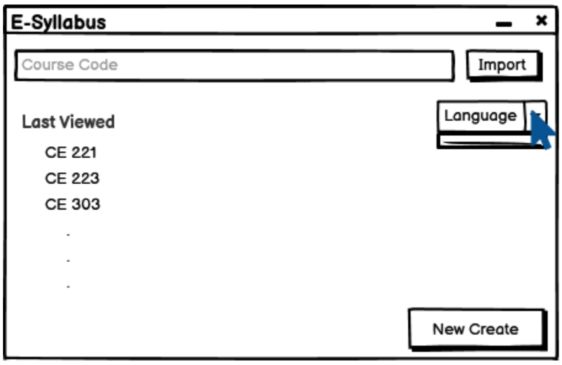


Figure 9: Login page

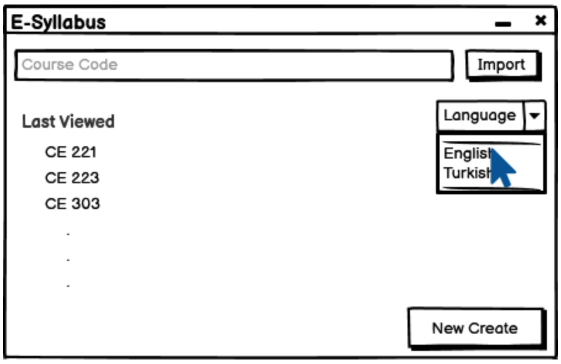


Figure 10: Language Choosing

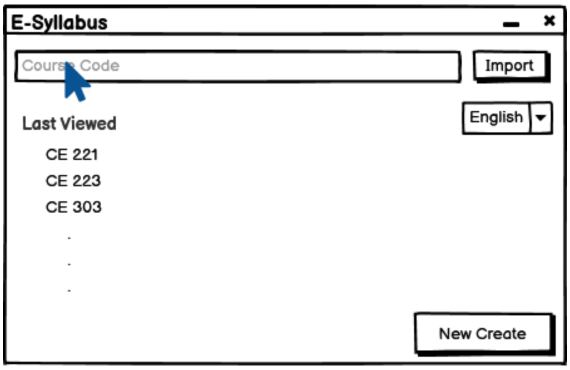


Figure 11: Entering the course code

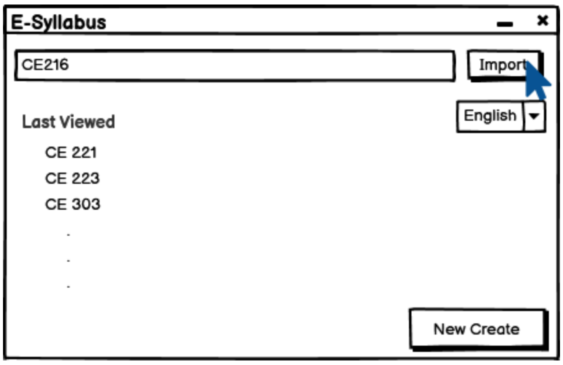
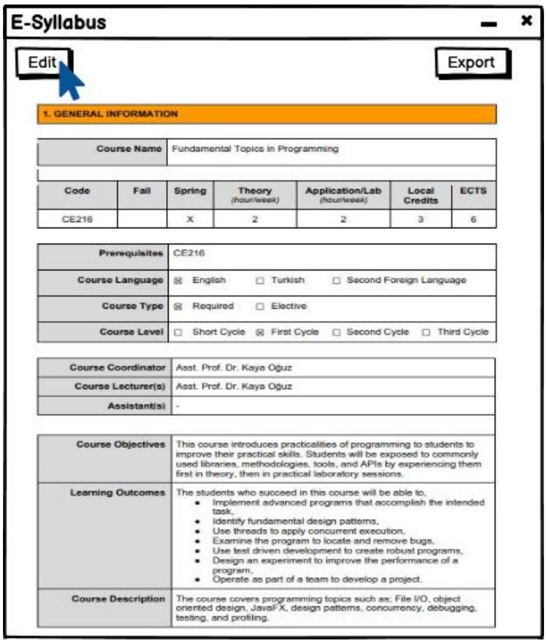


Figure 12: Importing syllabus



1. GENERAL INFORMATION

Course Name	Fundamental Topics in Programming

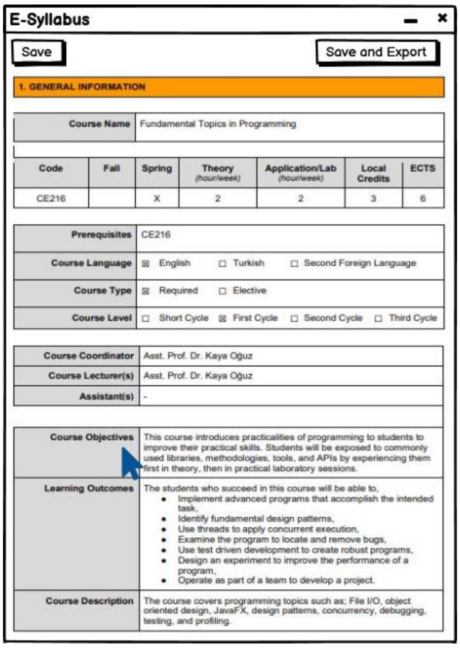
Code	Fall	Spring	Theory (hour/week)	Application/Lab (hour/week)	Local Credits	ECTS
CE216		X	2	2	3	6

Prerequisites	CE216
Course Language	☒ English ☐ Turkish ☐ Second Foreign Language
Course Type	☒ Required ☐ Elective
Course Level	☐ Short Cycle ☒ First Cycle ☐ Second Cycle ☐ Third Cycle

Course Coordinator	Asst. Prof. Dr. Kaya Oğuz
Course Lecturer(s)	Asst. Prof. Dr. Kaya Oğuz
Assistant(s)	-

Course Objectives	This course introduces practicalities of programming to students to improve their practical skills. Students will be exposed to commonly used libraries, methodologies, tools, and APIs by experiencing them first in theory, then in practical laboratory sessions.
Learning Outcomes	The students who succeed in this course will be able to, • Implement advanced programs that accomplish the intended task, • Identify fundamental design patterns, • Use threads to apply concurrent execution, • Examine the program to locate and remove bugs, • Use test driven development to create robust programs, • Design an experiment to improve the performance of a program, • Operate as part of a team to develop a project.
Course Description	The course covers programming topics such as; File I/O, object oriented design, JavaFX, design patterns, concurrency, debugging, testing, and profiling.

Figure 13: Editing syllabus



1. GENERAL INFORMATION

Course Name	Fundamental Topics in Programming

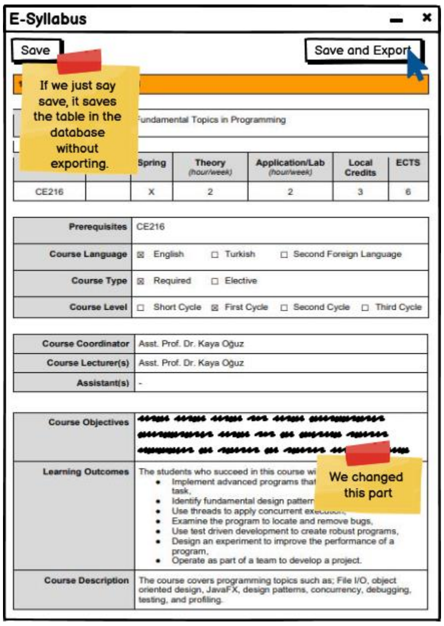
Code	Fall	Spring	Theory (hour/week)	Application/Lab (hour/week)	Local Credits	ECTS
CE216		X	2	2	3	6

Prerequisites	CE216
Course Language	☒ English ☐ Turkish ☐ Second Foreign Language
Course Type	☒ Required ☐ Elective
Course Level	☐ Short Cycle ☒ First Cycle ☐ Second Cycle ☐ Third Cycle

Course Coordinator	Asst. Prof. Dr. Kaya Oğuz
Course Lecturer(s)	Asst. Prof. Dr. Kaya Oğuz
Assistant(s)	-

Course Objectives	This course introduces practicalities of programming to students to improve their practical skills. Students will be exposed to commonly used libraries, methodologies, tools, and APIs by experiencing them first in theory, then in practical laboratory sessions.
Learning Outcomes	The students who succeed in this course will be able to, • Implement advanced programs that accomplish the intended task, • Identify fundamental design patterns, • Use threads to apply concurrent execution, • Examine the program to locate and remove bugs, • Use test driven development to create robust programs, • Design an experiment to improve the performance of a program, • Operate as part of a team to develop a project.
Course Description	The course covers programming topics such as; File I/O, object oriented design, JavaFX, design patterns, concurrency, debugging, testing, and profiling.

Figure 14: Editing Course Objectives



E-Syllabus

If we just say save, it saves the table in the database without exporting.

	Fundamental Topics in Programming

		Spring	Theory (hour/week)	Application/Lab (hour/week)	Local Credits	ECTS
CE216		X	2	2	3	6

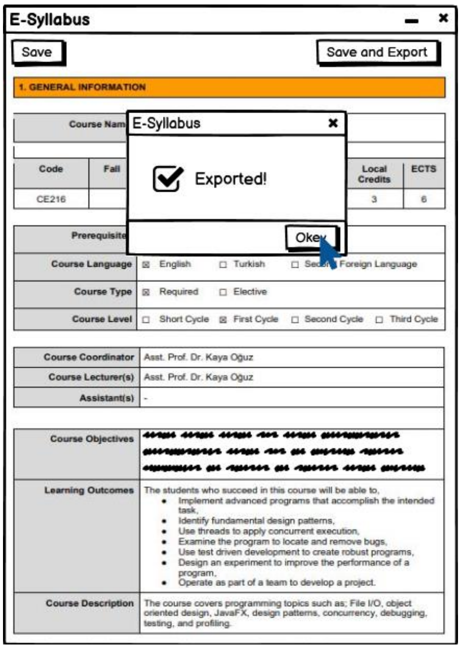
Prerequisites	CE216
Course Language	☒ English ☐ Turkish ☐ Second Foreign Language
Course Type	☒ Required ☐ Elective
Course Level	☐ Short Cycle ☒ First Cycle ☐ Second Cycle ☐ Third Cycle

Course Coordinator	Asst. Prof. Dr. Kaya Oğuz
Course Lecturer(s)	Asst. Prof. Dr. Kaya Oğuz
Assistant(s)	-

Course Objectives	
Learning Outcomes	The students who succeed in this course wi • Implement advanced programs that task, • Identify fundamental design pattern • Use threads to apply concurrent execution, • Examine the program to locate and remove bugs, • Use test driven development to create robust programs, • Design an experiment to improve the performance of a program, • Operate as part of a team to develop a project.
Course Description	The course covers programming topics such as; File I/O, object oriented design, JavaFX, design patterns, concurrency, debugging, testing, and profiling.

We changed this part

Figure 15: Exporting Syllabus



E-Syllabus

1. GENERAL INFORMATION

E-Syllabus

Exported!

Course Nam	

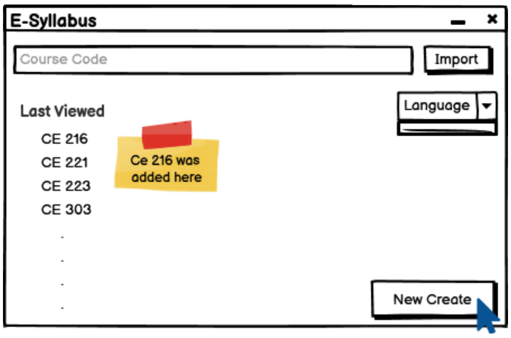
Code	Fall		Local Credits	ECTS
CE216			3	6

Prerequisite	
Course Language	☒ English ☐ Turkish ☐ Sec Foreign Language
Course Type	☒ Required ☐ Elective
Course Level	☐ Short Cycle ☒ First Cycle ☐ Second Cycle ☐ Third Cycle

Course Coordinator	Asst. Prof. Dr. Kaya Oğuz
Course Lecturer(s)	Asst. Prof. Dr. Kaya Oğuz
Assistant(s)	-

Course Objectives	
Learning Outcomes	The students who succeed in this course will be able to, • Implement advanced programs that accomplish the intended task, • Identify fundamental design patterns, • Use threads to apply concurrent execution, • Examine the program to locate and remove bugs, • Use test driven development to create robust programs, • Design an experiment to improve the performance of a program, • Operate as part of a team to develop a project.
Course Description	The course covers programming topics such as; File I/O, object oriented design, JavaFX, design patterns, concurrency, debugging, testing, and profiling.

Figure 16: Confirmation window



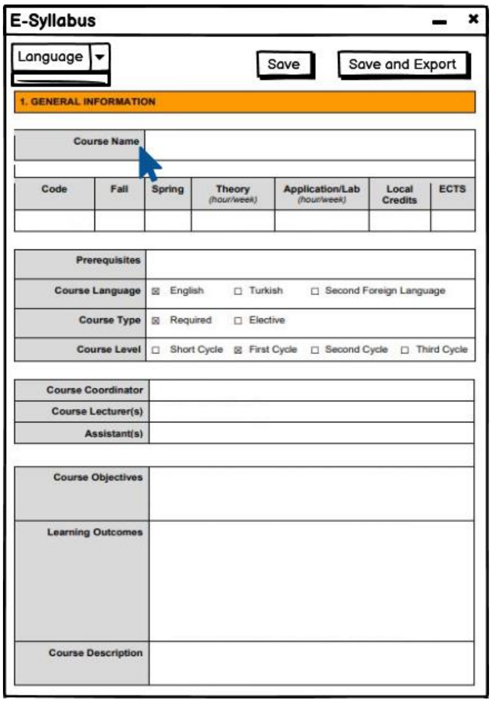
E-Syllabus

Last Viewed

CE 216
CE 221
CE 223
CE 303
.
.
.
.

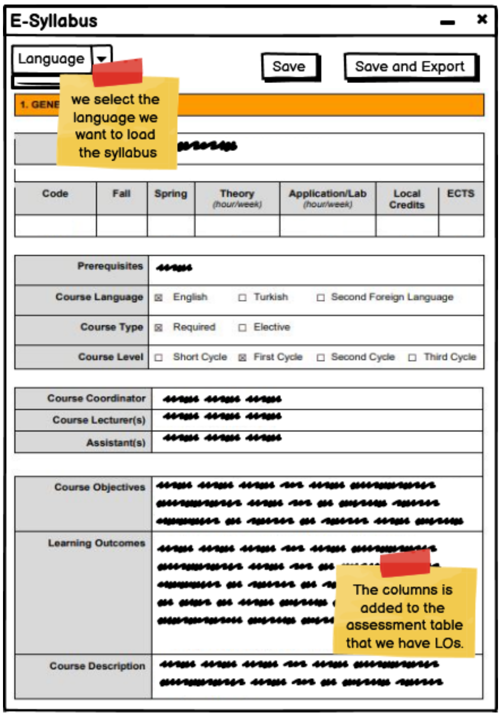
Ce 216 was added here

Figure 17: Creating new syllabus



E-Syllabus

Figure 18: Filling empty sections



E-Syllabus

we select the language we want to load the syllabus

The columns is added to the assessment table that we have LOs.

Figure 19: After filling in the blank sections, choosing a language and save or export.

3.4.Interaction Viewpoints

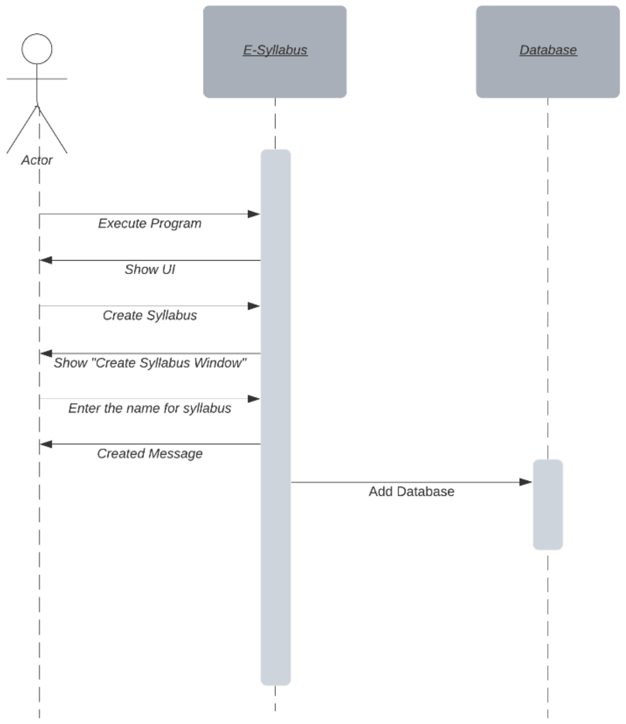


Figure 20: Create Sequence Diagram

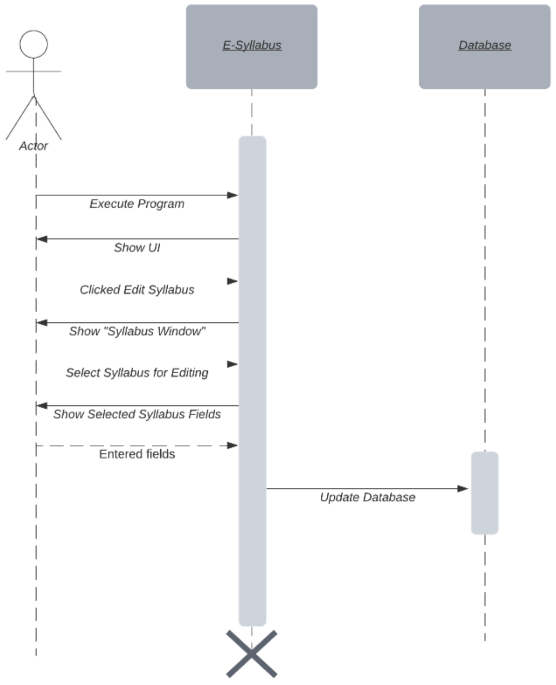


Figure 21: Edit Sequence Diagram

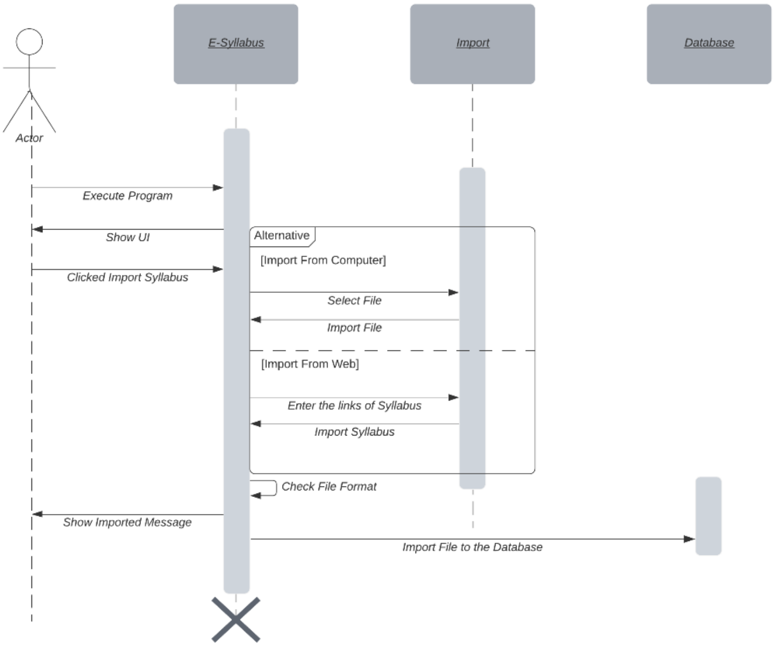


Figure 22: Import Sequence Diagram

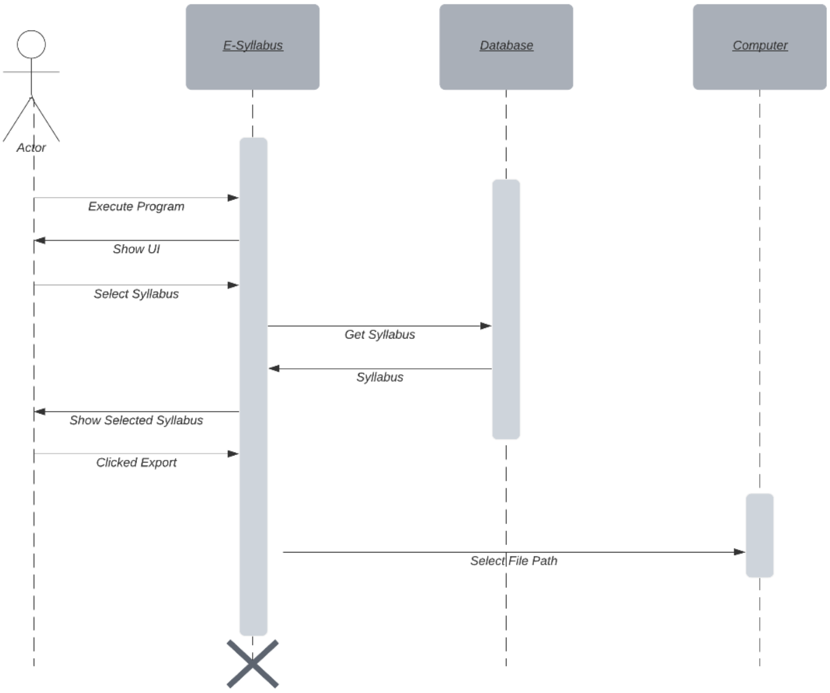


Figure 23: Export Sequence Diagram

3.5.Database Viewpoint

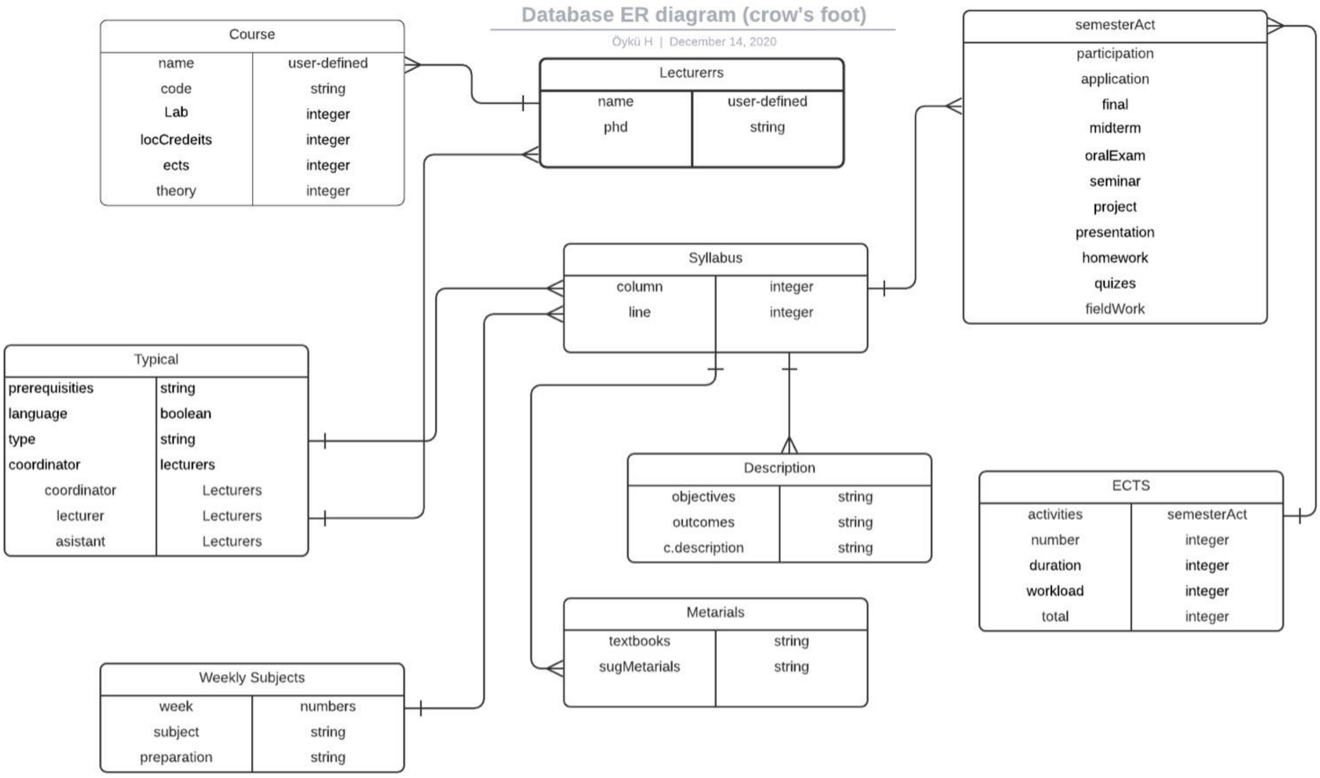


Figure 24: Database Diagram