

ANKARA YILDIRIM BEYAZIT UNIVERSITY
FACULTY OF
NATURAL SCIENCES AND ENGINEERING



CENG 206 - PROGRAMMING LANGUAGES
PROJECT REPORT (GROUP 4)

14/05/2024

ANKARA

Group Members:

Efe Baştuğ - 22050151039

Mehmet Ali Yılmaz - 21050111057

Bülent Durusoy - 21050111029

USAMA Al-khadzh - 22050141014

Project Overview

The goal of this project is to develop a web-based course scheduler application using React.js. The application will serve as a tool for managing and organizing the course schedules for a university department.

Project Purpose

The main purpose of this course scheduler web application is to provide an efficient and user-friendly platform for both students and instructors to manage their course-related information and schedules. The key objectives of the project include:

Instructor Course Management: Instructors should have the ability to add, edit, and delete course information, such as course details, enrollment numbers, and instructor availability.

Calendar Visualization: The application should provide a calendar-based view of the entire course schedule, making it easy for users to visualize the overall course plan.

Constraint-based Scheduling: The system should automatically assign classrooms and time slots for each course while respecting various constraints, such as course intersections, classroom capacities, fixed time slots for service courses, and instructor availability.

Error Handling: If the system is unable to find a feasible schedule that satisfies all the constraints, it should display an error message and suggest ways to resolve the issue, such as increasing the number of classrooms or their capacities.

Flexible Configuration: The application should allow users to easily configure and modify the course information, classroom details, service course schedules, and instructor availability through a user-friendly interface or by uploading relevant data files.

Features and Functionalities

Course Management:

Ability for instructors to add, edit, and delete course information, including course code, name, year, credit, type (compulsory/elective), department (department/service), number of students, instructor name, and course hours preference (block or divided).

Functionality to upload course information from a CSV file.

Display of the parsed course information in a tabular format.

Classroom Management:

Ability for administrators to add, edit, and remove classroom information, including classroom ID and capacity.

Functionality to upload classroom information from a CSV file.

Display of the parsed classroom information in a tabular format.

Service Course Schedule:

Functionality to upload the fixed time slots for service courses (courses provided by other departments) from a CSV file.

Display of the parsed service course schedule information in a tabular format.

Instructor Availability:

Ability to upload the busy time slots for each instructor from a CSV file.

Display of the parsed instructor busy time slots in a tabular format.

Course Scheduling:

Automatic assignment of classrooms and time slots for each course, respecting the specified constraints (course intersections, classroom capacities, fixed time slots for service courses, and instructor availability).

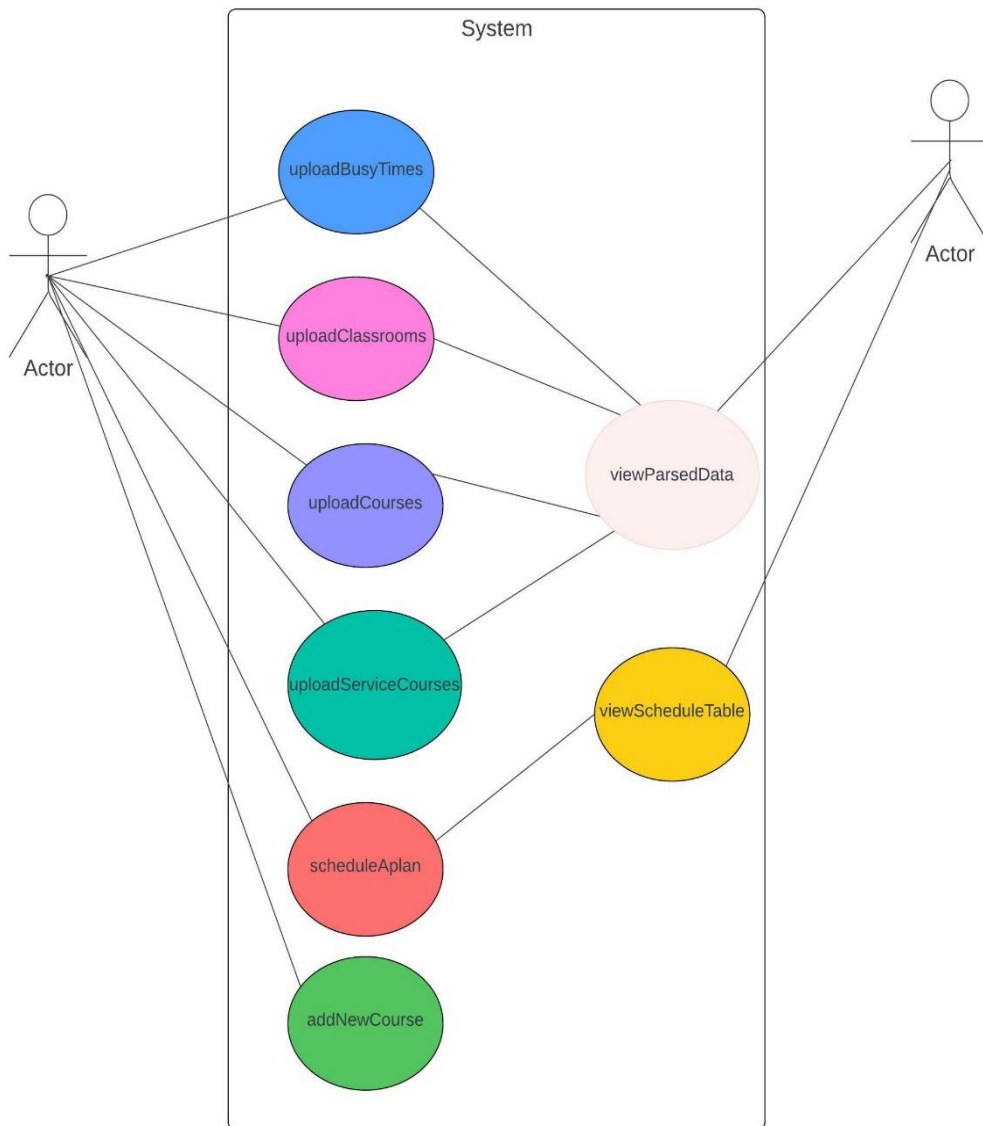
Validation to ensure that courses in the same year do not intersect.

Error Handling and Notification:

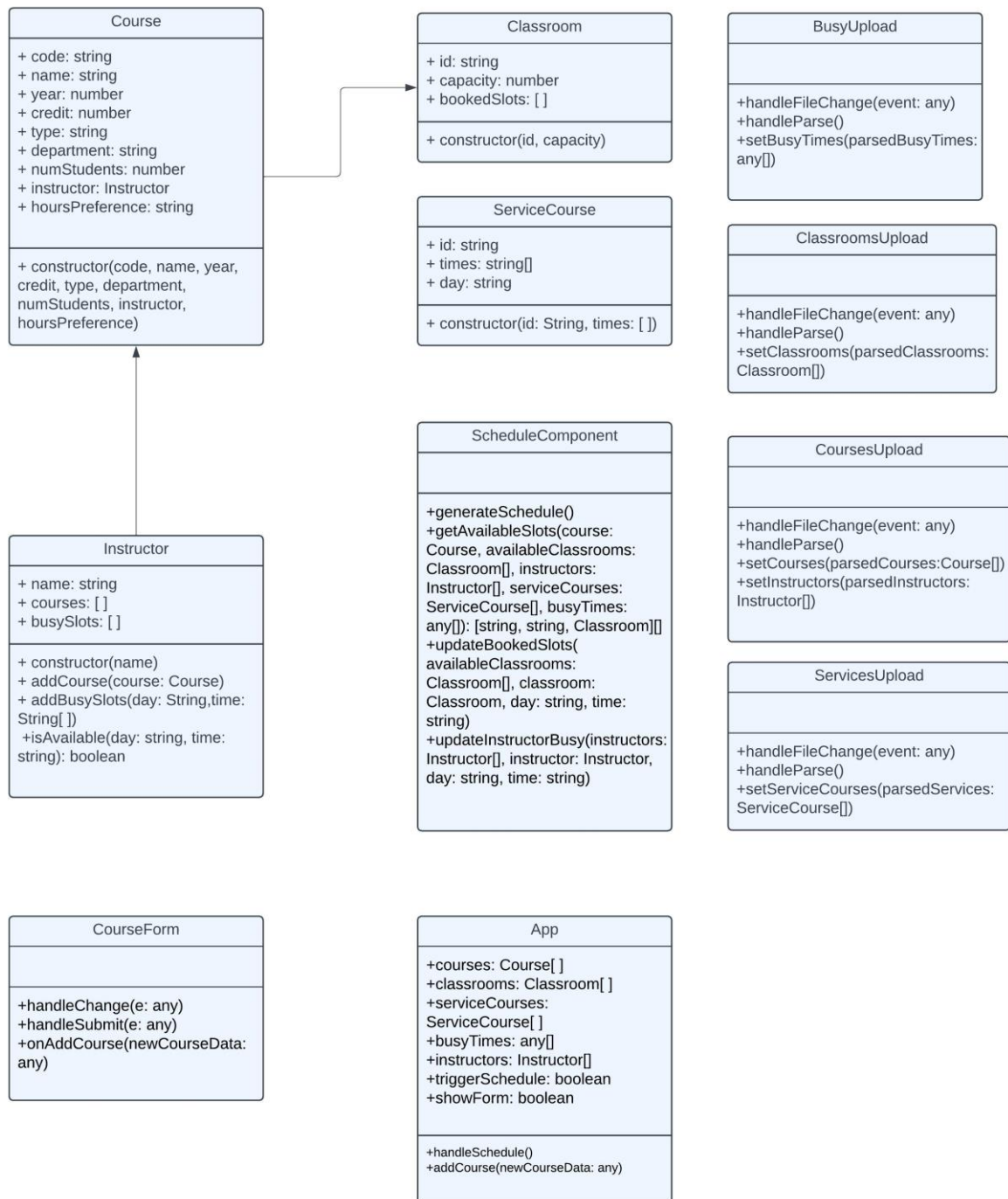
If the system cannot find a feasible schedule that satisfies all the constraints.

Architecture

Use case diagram for the project



UML diagram for the Project



Pseudocode:

ScheduleCourses(courses, classrooms, serviceCourses, busyTimes)

Initialize an empty schedule for putting courses

Push the busy times of the instructors

Schedule service courses first

Look for classroom and instructor availability for consecutive slots

IF found, schedule the course in the predetermined time slot

Start scheduling for department courses

IF course has department S, skip it

IF course has a hours preference like 2+1, parse its hours separately

First, look for the first hours before the +

IF managed to schedule, look for other hours

IF failed to schedule the last hour, give an error message

ELSE if course has a normal hours preference, call the regular scheduleCourseHelper function

Get the starting hour of the course

Get the hours currently scheduled for the course and check if it is equal to the current hours preference for the course

Check for classroom, instructor and year availability for each consecutive time slots for the course's hours preference

IF all are available, push the course into the schedule, push the time slots to the classroom's booked slots and instructor's busy slots

RETURN true

Keep going until the courses end and push the courses

RETURN schedule

User Interface Design:

Screenshots and descriptions of the key user interface elements

Before uploading files:

Upload Courses

Select Courses FileParse Courses

Parsed Courses:

Code	Name	Year	Credit	Type	Department	Capacity	Instructor
------	------	------	--------	------	------------	----------	------------

Upload Classrooms

Select Classroom FileParse Classrooms

Parsed Classrooms:

Name	Capacity
------	----------

Upload Service

Select Service FileParse Service

Parsed Service:

Name	Instructor	Time Slots
------	------------	------------

Upload Busy

Select Busy FileParse Busy

Parsed Busy:

Instructor Name	Day	Time Slots
-----------------	-----	------------

Schedule A Plan

Schedule for Year 1

Time	Monday	Tuesday	Wednesday	Thursday	Friday
8:30	-	-	-	-	-
9:30	-	-	-	-	-
10:30	-	-	-	-	-
11:30	-	-	-	-	-
12:30	-	-	-	-	-
13:30	-	-	-	-	-
14:30	-	-	-	-	-
15:30	-	-	-	-	-

After uploading files:

Upload Courses

Select Courses FileParse Courses

Parsed Courses:

Code	Name	Year	Credit	Type	Department	Capacity	Instructor
CENG114	Computer Programming II	1	5	C	D	95	OGR.GOR. YUSUF EVREN AYKAC
CHEM101	GENERAL CHEMISTRY	1	5	C	S	110	DOC.DR. NURAY CELEBI
MATH104	Applied Linear Algebra	1	5	C	D	103	DOC.DR. MUHAMMED ABDULLAH BULBUL
MATH102	CALCULUS II	1	6	C	S	116	DR. OGR. UYESI SELIM BAHADIR

Upload Service

Select Service File

Parse Service

Parsed Service:

Name	Instructor	Time Slots
CHEM101	Tuesday	8:30,09:30,10:30
MATH102	Monday	13:30,14:30,15:30
MATH106	Thursday	13:30,14:30,15:30
PHYS102	Friday	8:30,09:30,10:30
TDL102	Wednesday	13:30,14:30,15:30
TIT101	Wednesday	8:30,09:30,10:30
ENGR202	Tuesday	8:30,09:30,10:30
ENGR254	Tuesday	13:30,14:30,15:30

Schedule for Year 1

Time	Monday	Tuesday	Wednesday	Thursday	Friday
8:30	Computer Programming II (B403)	GENERAL CHEMISTRY (B503)	TURK INKILAP TARIHI I (B503)		PHYSICS II (B503)
9:30	Computer Programming II (B403)				
10:30	Computer Programming II (B403)	GENERAL CHEMISTRY (B503)	TURK INKILAP TARIHI I (B503)		PHYSICS II (B503)
11:30		Applied Linear Algebra (B503)			
12:30		Applied Linear Algebra (B503)			
13:30	CALCULUS II (B503)	Applied Linear Algebra (B503)	TURK DILI II (B403)	PROBABILITY AND STATISTICS (B403)	
14:30	CALCULUS II (B503)		TURK DILI II (B403)	PROBABILITY AND STATISTICS (B403)	
15:30	CALCULUS II (B503)		TURK DILI II (B403)	PROBABILITY AND STATISTICS (B403)	

Conclusion:

Unfortunately, our team was unable to complete the course scheduler web application project as planned. Despite our best efforts, we faced several challenges that hindered our progress. Primarily, there was a lack of effective communication and coordination within the group. Team members were not always responsive or available, which led to delays in decision-making and implementation. Additionally, we underestimated the amount of time and effort required to build a comprehensive application with all the specified features and constraints. Poor time management on our part resulted in us falling behind schedule, and we were unable to allocate sufficient time to thoroughly test and debug the application. These issues, combined with the complexity of the project, proved to be too much for us to overcome within the given timeframe. This experience has been a valuable lesson for us, and we will apply these learnings to our next group endeavor.