



Cairo University
Faculty of Computers and Information
Department of Computer Sciences

A Better E-com

Supervised by

Dr. Basheer

TA. Hassan Mourad

Implemented by

20170191	فارس سيد حسن السيد
20170301	منير صبرى يوسف احمد
20170307	مينا بطرس فرحتات فلتس
20170174	عمر حسن حسني عبد العال
20170198	كريم محمد فتحي مسعود

Graduation Project
Academic Year 2020-2021
Midyear Short Documentation

Abstract

The old E-com features are not working properly and there are missing ones. The new E-com fixed some of the not working features but still missing other features. The missing features were found on other websites like Acadox, Blackboard and Google Classroom, but there were still some features that were not on them. We propose developing a Better E-com will combine the features existing in the old and new E-com and other platforms like Acadox and blackboard in one place, while removing unnecessary (not be helpful in the new E-com) features and adding new features and have a good look (design). By doing this, we will help new FCAI generations to have a better learning experience. In order to build our application, we will use .net with C# and JavaScript with Angular.

Table of Contents

1. Background.....	4
2. Problem Definition.....	4
3. Proposed System	4
4. Related Work.....	4
5. Project Specifications.....	6
5.1. System Architecture	6
5.2. Stakeholders.....	6
5.3. Functional Requirements.....	7
5.4. Non-Functional Requirements.....	8
5.5. Use Case Diagrams	8
5.6. Class Diagram	10
5.7. Sequence Diagrams	11
5.8. ERD	13
6. Work Plan	13
References.....	14

1. Background

The world wide web (Internet) has made sharing and communicating with other people a lot more convenient. You can communicate with someone in the other side of the globe while being at your home. Thus, Web development (the process of making a web site) has huge demand on it, especially at the current time of the pandemic. In the area of E-learning, many websites have been developed to use the power of the Internet in this field like Google Classroom, Blackboard, etc. Probably all universities have websites. Our college website, E-com, is lacking in both the look and the features, with missing features and non-working ones. There was an attempt in the new E-com to overcome those issues but there are still some issues to solve.

What we are doing is fixing these issues and blend other features from E-learning sites like Google Classroom in it, so the E-com becomes as self-sufficient as possible and adding new features. This will improve the experience that the new generation of students will have while using our E-com. We will build a restful web API with C# and .net and make the frontend framework Angular use it.

2. Problem Definition

The old E-com features are not working properly and there are missing features that they are better to be in the old E-com to make the E-learning easier for the students, instructors, and student affairs. The old E-com is not user friendly for the stakeholders that use that E-learning system.

3. Proposed System

We build the better E-com to improve the general look of the E-com, improve existing features, add features from other E-learning website like Blackboard and Google Classroom to make it self-sufficient while adding new features and removing existing unnecessary ones.

4. Related Work

1. **Old E-com:** A system built by FCAI for E-com services.
2. **New E-com:** A modified system to replace the old E-com.
3. **Acadox:** An E-learning system built to organize courses and their materials

4. **Blackboard:** The official E-learning system used by Cairo University.

5. **Google classroom:** An E-learning system built by Google.

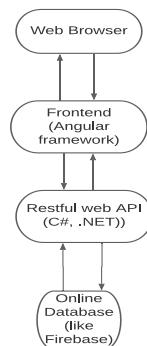
The main difference is that our better E-com combines E-com features with E-learning website features (e.g., Acadox, Blackboard).

Feature	Better E-com	Old-bylaw E-com	New bylaw E-com	Acadox	Google Classroom	Blackboard
Courses	+	+	+	+	+	+
View his registered courses	+	+	+	+	+	+
Register Course	+ In limited time.	+ In limited time.	+ In limited time.	+	+	+
Late Register Course with Fee	+	+ Only at specific time of the year	+	-	-	-
Add or Download non-related course materials	+	+	+	-	-	-
Message	+	+	-	+	-	+
Online Quiz	+	-	-	+	+	+
Top 50	+	+	+	-	-	-
Assignments (view, deliver, view deadline)	+	-	-	+	+	+
Calendar	+	-	-	+	+	+
Exam Locations	+	+	+	-	-	- But doable
General Feed (posts on your homepage, limited to the staff)	+	+	+	-	-	-
Course Feed	+	-	-	+	+	+
Course Feedback	+	+	+	-	-	+
Check Instructors for The Course	+	-	-	+	+	+
Check Instructor Profile	+	+ On the faculty staff website	+ On the faculty staff website	+	-	-
Check Instructor Mail	+ In the profile	+	+	-	-	-

		On the faculty staff website	On the faculty staff website			
Check Office Hours	+	-	-	-	-	- But doable
Check Office Location	+	-	-	-	-	-
Check Course Grade	+	For each assignment, quiz, etc.	Only Final and Class's Work.	Only Final and Class's Work.	Not used often.	+
Complain About a Grade	+	-	-	-	-	-
Check Course Absence	+	+	+	-	-	-
Submit attendance excuse	+	-	+	-	-	-

5. Project Specifications

5.1. System Architecture



5.2. Stakeholders

1. Student.
2. Instructor.
3. Dean.
4. System Admin.
5. College Vice Dean for Student Affairs.
6. Student Affairs Staff Member.

5.3. Functional Requirements

Here is the functional requirement list, they are sorted from the highest to the lowest priority.

1. Student/Instructor Registration.
2. Account Management.
3. Department Management.
4. Courses (Registration, Deletion).
5. Course Student/Instructor Registration.
6. General Feeds.
7. Course Feeds.
8. Course Attendance.
9. Course Attendance Complaints.
10. Course Attendance Excuses.
11. Course Grades.
12. Course Grades Complaints.
13. General Materials.
14. Course Materials.
15. Message System.
16. Calendar Management.
17. Time Schedule Management.
18. Search for Student/Instructor.
19. Student/Instructor contact (view profile, mail, office hours, office location).
20. Top 50 (auto generated by the system).
21. Course Assignments.
22. Send a confirmation mail after submitting an assignment.
23. Course Read-only and Archive.
24. Support Files (file, audio, video, and link).
25. Exams (offline, online, exam locations).
26. Course Discussion.
27. Poll System (Course Poll, General Poll).
28. Activity Stream.
29. Event Registration (register to attend a meeting).
30. Grouping System (Course Groups).

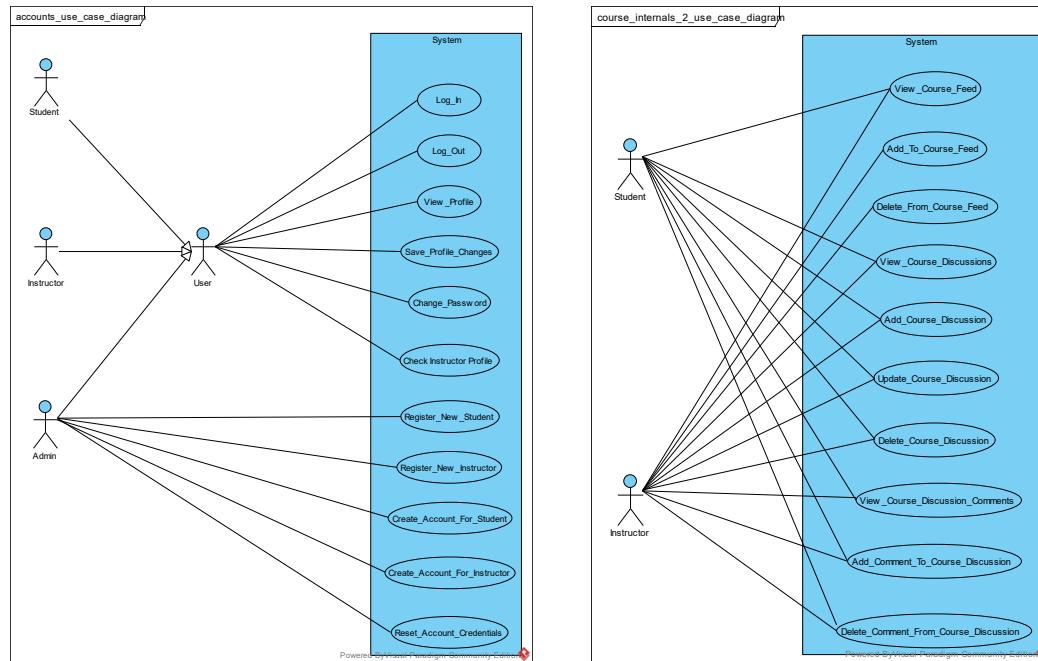
- 31. Arabic Support.
- 32. Help Guide.
- 33. Notification System.
- 34. Folder Sync (like git).

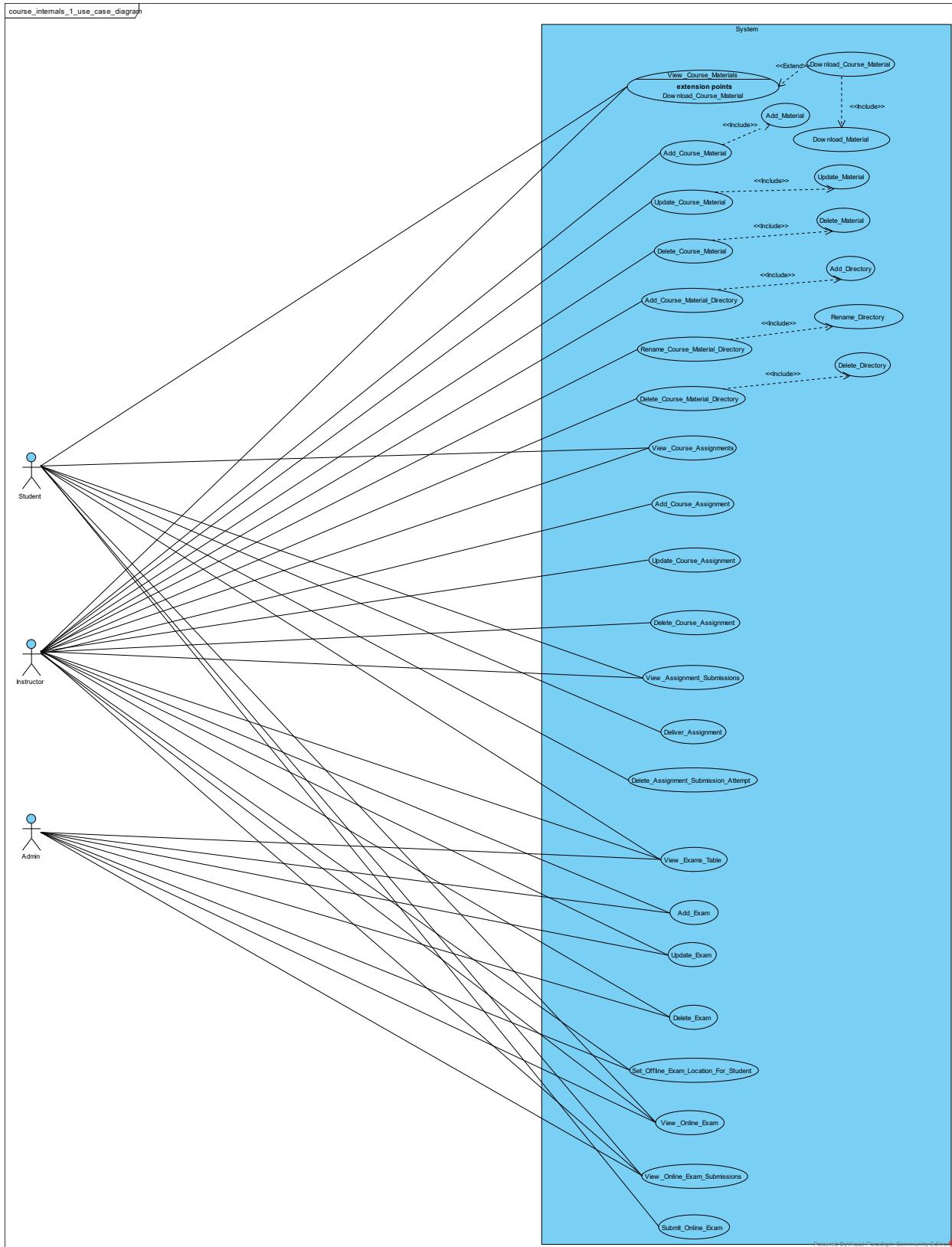
The first Agile cycle will include the first two functional requirements.

5.4. Non-Functional Requirements

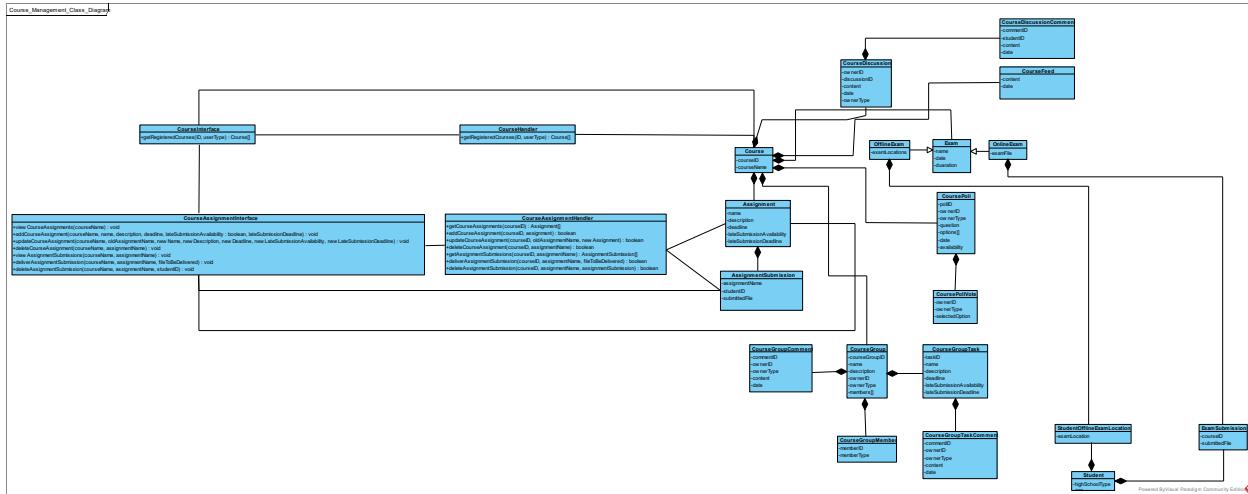
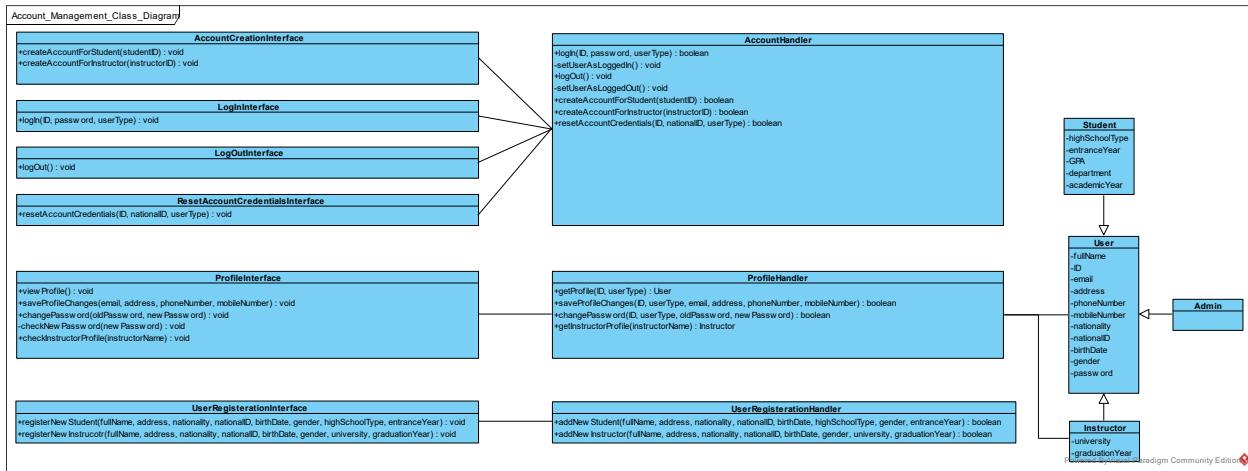
1. **Usability:** System should be easy even for new students that are not tech savvy.
2. **Performance:** The system should handle assignment turn-in in workload without a problem.
3. **Consistency:** Functions that have similar steps should have similar interfaces
4. **User Friendly:** The design and colors of the interface should be suitable for all ages and for the user's eyes.
5. **Robustness:** The system should be able to handle invalid user input.

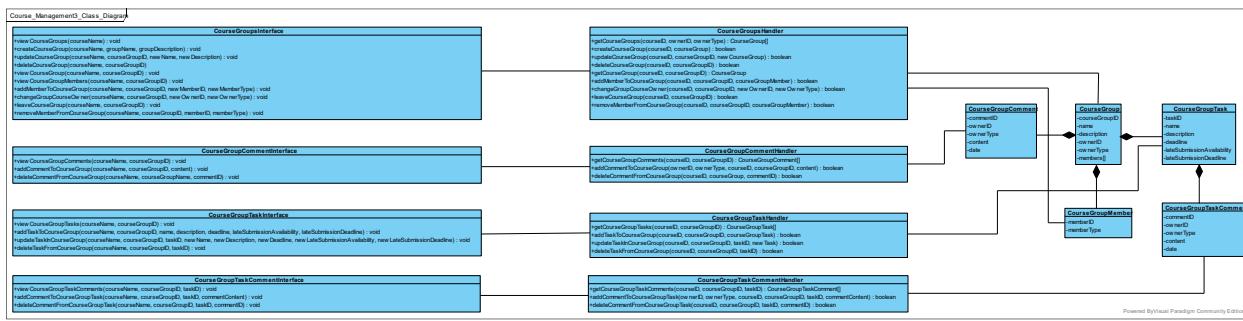
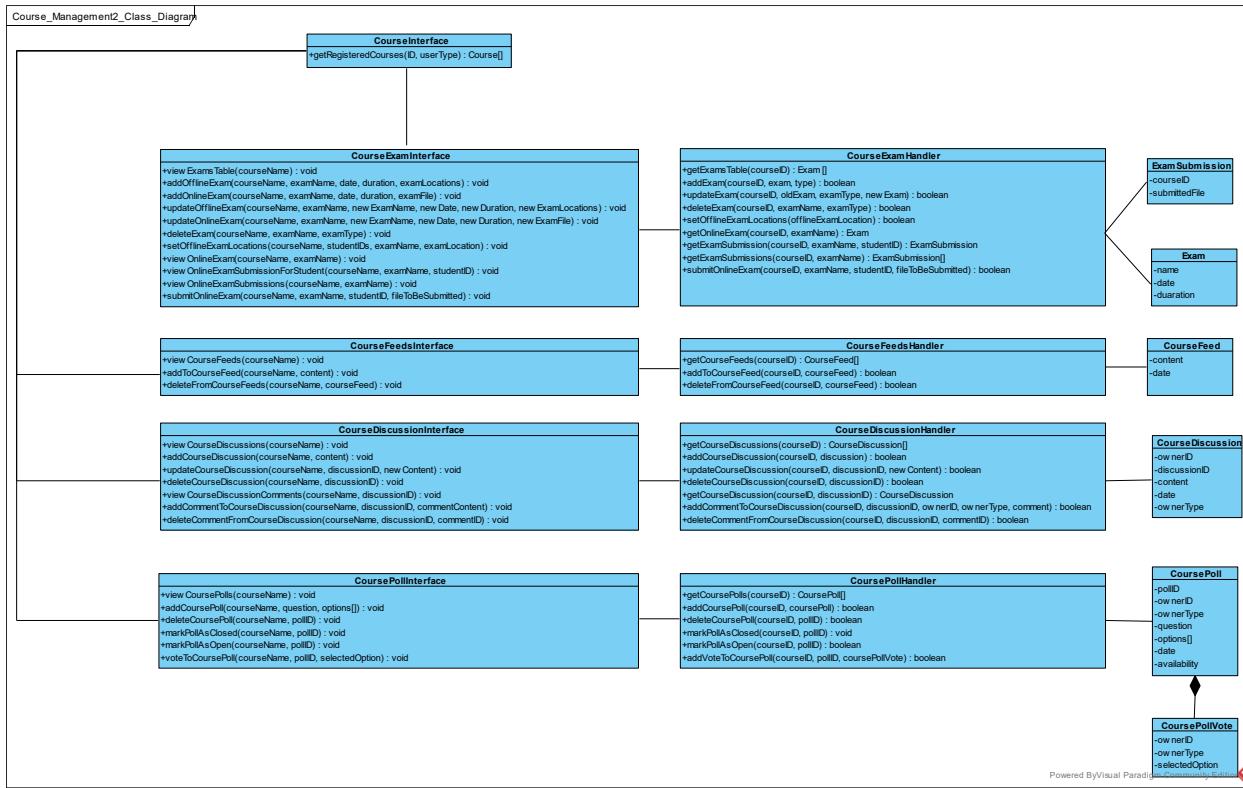
5.5. Use Case Diagrams



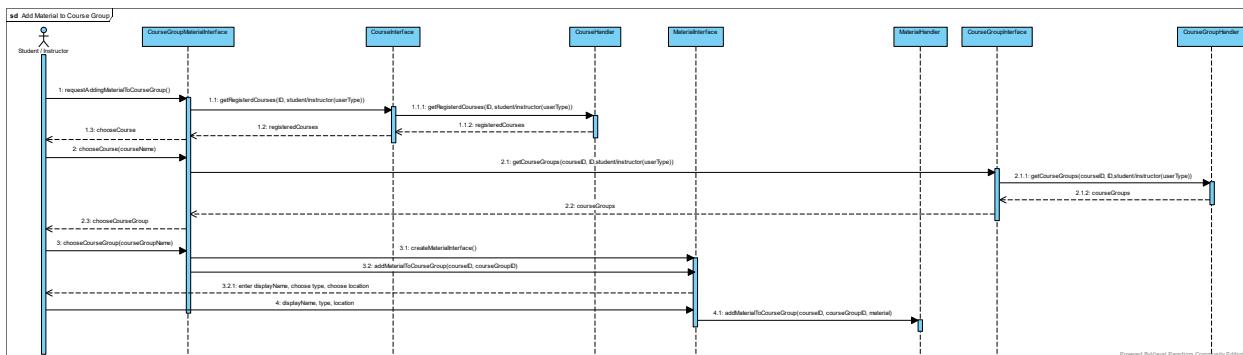


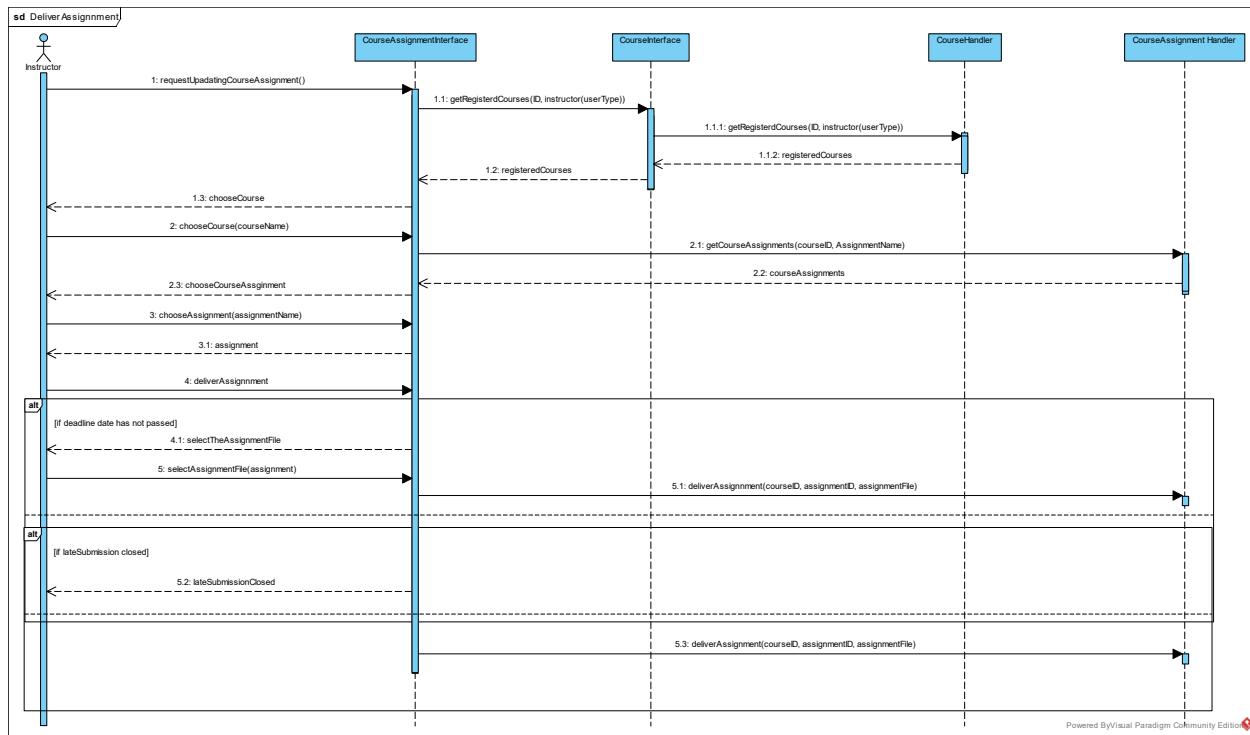
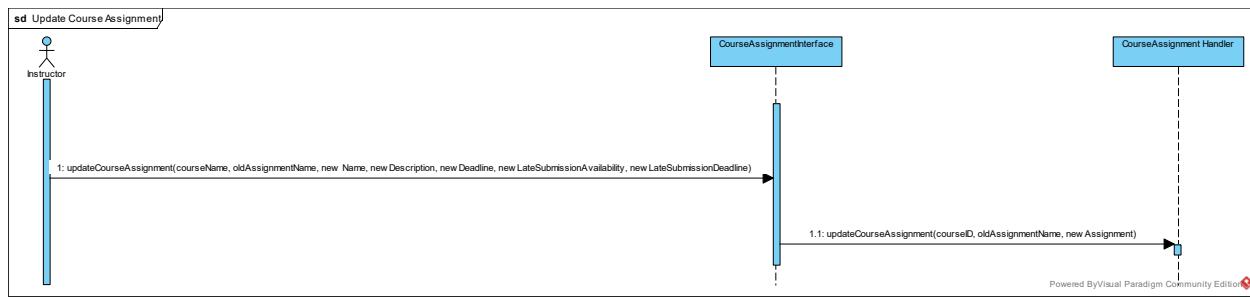
5.6. Class Diagram



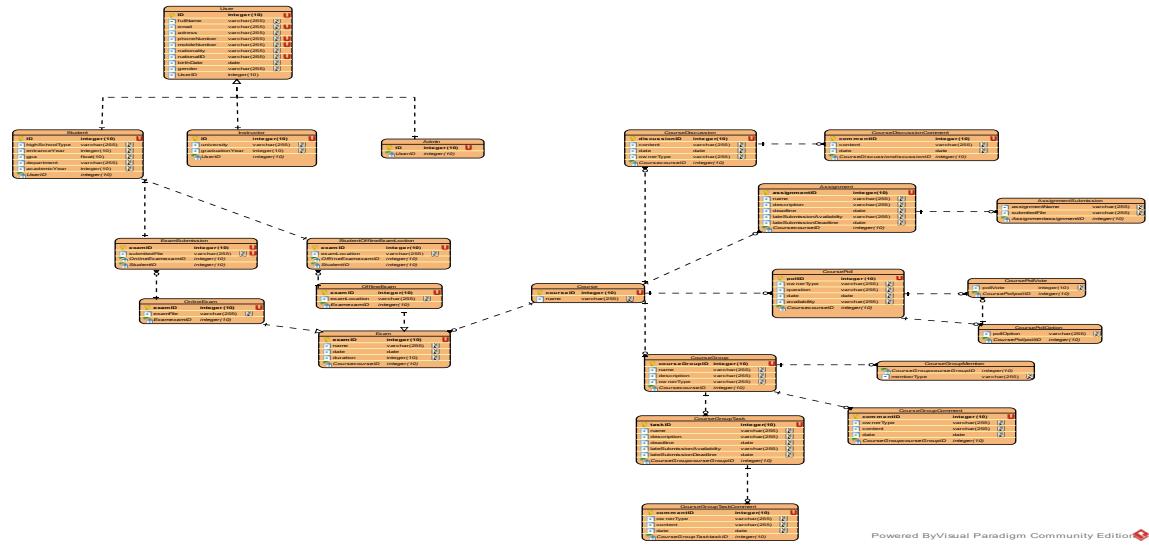


5.7. Sequence Diagrams





5.8. ERD



6. Work Plan

Task	Task Title	Description	Task Status
1	Analyze websites that give similar service to ours	<p>Analyze websites with similar service to E-com and compile list of features to be added to our project.</p> <p>Analyzed websites: Blackboard. Google Classroom. E-com (old and new). Acadox.</p>	Completed
2	Construct use case diagram.	Construct the diagrams linking each actor to the corresponding use cases.	Completed
3	Construct use case tables	Constructor tables that give more details about each use case.	Completed

4	Construct Sequence Diagrams	Make some sequence diagrams for some use cases	Partially Completed
5	Construct class diagram.	Build the known UML class diagram for our system.	Partially Completed
6	Prioritize compiled features.	Put priority of each feature to be implemented in the project.	Completed
7	Construct ERD diagram	Construct the entity relationship diagram for the database.	Partially Completed
8	Design some Website pages	Make a mockup for some of our website pages	Will be completed within month.
9	Learn Angular frontend framework	Self-explanatory	Will be completed within month.
10	Learn .NET Framework and C#.	Self-explanatory.	Will be completed within month.
11	Implementing features	Start implements features from highest to lowest priority	Will be done over cycles within the rest of the academic year.

References

1. **Old E-com:** <http://ecom.fci-cu.edu.eg/>
2. **New E-com:** <http://newecom.fci.cu.edu.eg/#/>
3. **Acadox:** <https://www.acadox.com/>
4. **Google Classroom:** <https://classroom.google.com/>
5. **Blackboard:** <https://cu.blackboard.com/>