

**Faculty of Engineering Department of  
Informatics Engineering-Software and  
Information System Engineering**



# **University software project management and follow-up system**

**Senior 1 project**

**Prepared by :**

**Baraa Abdallah   Souhaib Sabboura   Obada Kahwajy**

**Supervised by :**

**Eng.Anas Abdulaziz**



الجامعة السورية الخاصة  
SYRIAN PRIVATE UNIVERSITY

كلية الهندسة  
قسم الهندسة المعلوماتية  
هندسة البرمجيات ونظم المعلومات

## نظام إدارة ومتابعة المشاريع البرمجية الجامعية

### مشروع تخرج 1

إعداد

عبادة قهوجي      محمد صهيب صبورة      براء عبدالله

إشراف

المهندس أنس عبدالعزيز

# Chapter 1

## Introduction

**الهدف العام من المشروع :** يهدف المشروع إلى إنشاء نظام رقمي يساعد في عملية التنظيم ,متابعة وتقييم المشاريع البرمجية التي ينفذها طلاب الجامعات مع تسهيل التواصل بين المشرفين والطلاب وضمان سير العمل وفق جدول زمني محدد .

## **نطاق المشروع :**

يشمل نطاق المشروع تصميم وتطوير منصة ادارة المشاريع البرمجية حيث سيتم التركيز على خصائص متعددة منها إدارة المهام حيث سيتم تعيين المهام مع أدوار محددة للطلاب والمشرفين الإداريين لتلبية احتياجات البيئة الأكاديمية. توجها المنصة أساساً للاستخدام في الجامعات والمؤسسات التعليمية التي تدير مشاريع برمجية, مع إمكانية التوسع مستقبلاً لتشمل سياقاً أخرى .يمكن تقسيم المشروع إلى مراحل :

### **1. المرحلة الأولى :**

المنتج الأدنى القابل للتطبيق MVP- Minimum Viable Product حيث سيتم التركيز على الوظائف الأساسية :

- إدارة المستخدمين والأدوار :
- السماح للطلاب والمشرفين والإداريين بإنشاء حسابات جديدة ( اسم , بريد إلكتروني , كلمة المرور )
- إدارة المشاريع الأساسية :
- نموذج لإنشاء مشروع جديد(عنوان , وصف , الطلاب المشاركين , الأستاذ المشرف )
- صفحة لعرض قائمة جميع المشاريع
- صفحة تفاصيل لكل مشروع ( عرض المعلومات الأساسية )
- إدارة المهام الأساسية :
- إضافة مهام بسيطة لكل مشروع ( عنوان المهمة ,موعد التسليم ).
- تحديد حالة المهمة (قيد الانتظار, قيد التنفيذ , مكتملة ) .
- التواصل الأساسي :
- قسم التعليقات ( Comments ) داخل صفحة كل مشروع للمناقشات بين الطلاب والمشرف .

### **2. المرحلة الثانية :**

- الهدف في هذه المرحلة تحسين تجربة المستخدم , إضافة ميزات التعاون المتقدمة :
- نظام الإشعارات ( Notifications ) داخل المنصة ( عند تعليق جديد او تغيير في مهمة )

- رفع ومشاركة الملفات (مستندات , عروض تقديمية , أكواد )مرتبطة بكل مشروع أو مهمة
- لوحة المشروع ( Project Dashboard ) تعرض نظرة عامة وسريعة على التقدم
- البحث والفلتر : شريط بحث للعثور على المشاريع , فلاتر متقدمة ( حسب القسم , السنة )

### 3. المرحلة الثالثة :

- إضافة ميزات متقدمة مثل نظام الاقتراحات والعروض ( proposals ) يمكن للطلاب من خلاله طرح أفكار مشاريع والعثور على زملاء أو مشرفين .
- إضافة تقارير عن أداء الطلاب والمشرفين .
- تحسين الأداء وقابلية التوسع .

### □ النطاق التقني للمشروع :

النطاق التقني المحدد لهذا المشروع يتضمن جميع العناصر اللازمة لبناء منصة إدارة ومتابعة المشاريع البرمجية الطلابية بما يضمن سهولة التواصل بين المشرفين والطلاب .

#### 1. الواجهة الأمامية **FrontEnd** :

سيتم تصميم وتطوير واجهات النظام باستخدام **HTML, CSS , MUI Library** ,**React**

#### 2. الواجهة الخلفية **BackEnd** :

سيتم باستخدام **Php Laravel** كإطار عمل لتطوير الجزء الخلفي للموقع مما يمكن من إدارة الحسابات والمشاريع الطلابية

#### 3. قاعدة البيانات **Database** :

استخدام قاعدة بيانات علائقية **MySQL** لتخزين معلومات المستخدمين من طلاب ومشرفين , المشاريع البرمجية المسجلة في النظام و التقييمات عليها .

#### 4. التكامل مع نظام التحكم بالاصدارات :

حيث سيتم ربط المشروع بمستودع خارجي وهذه ميزة حيوية للمنصة البرمجية حيث ستتفاعل مع **GitHub Apis**

# Chapter 2

## Project Management :

## 1. مقدمة

في هذا الفصل سنتعمق بمرحلة إدارة المشروع , والتي تعد جانباً حاسماً لضمان نجاح المشروع .

سنقوم بفحص ميثاق المشروع Project charter , وخطة المشروع , ووثيقة بيان العمل ( SOW -Statement of Work ) , ومخطط ( WBS-Work Breakdown Structure ) و تحليل أصحاب المصلحة , واستراتيجيات إدارة المخاطر لتوجيه المشروع والتحكم فيه بشكل فعال من البداية إلى اكتمال المشروع .

## 2. ميثاق المشروع Project charter

ميثاق المشروع هو وثيقة رسمية بمثابة تفويض رسمي لبدء المشروع , وبمثابة نقطة مرجعية في جميع أنحاء المشروع , مما يوفر فهماً واضحاً لغرض المشروع ويضع أساساً لصنع القرار وإدارة المشروع .

عنوان المشروع	نظام إدارة ومتابعة المشاريع البرمجية الجامعية
تاريخ بدء المشروع	18/October/2025
تاريخ انتهاء المشروع	
مشرف المشروع	المهندس أنس عبد العزيز
منهجية المشروع	Scrum

### ● أهداف المشروع :

- تأسيس نظام إدارة ومتابعة المشاريع البرمجية الجامعية يضمن التواصل بين الأساتذة المشرفين والطلاب بكفاءة .
- توفير بيئة آمنة تُتيح للمشرفين إدارة ومتابعة مشاريع طلابهم وتمكن الطلاب من تحديث بيانات مشاريعهم .
- تنظيم مراحل المشروع بشكل واضح مما يضمن نجاح سير العملية .
- تسهيل التواصل بين الطلاب والمشرفين داخل النظام .

## • نهج المشروع - Approach :

- تم تطوير المشروع تدريجياً على مراحل وفقاً للأولوية كما يلي :

1. إدارة المستخدمين والأدوار : تطوير نظام تسجيل دخول للمستخدمين ( مسؤولين , طلاب , مشرفين ) مع صلاحيات وصول مخصصة لكل منهم , تم تنفيذه ضمن أول Sprint لضمان الوصول الآمن والتحكم في الحسابات .
2. إدارة المشاريع : تمكين الطلاب من إضافة مشروع جديد مع تحديد الاسم , الوصف , والمدة وتمكين المشرفين من قراءة بيانات المشروع .
3. إدارة المهام : إتاحة إنشاء مهام فرعية مع وصف وتاريخ تسليم , تحديثها وإزالتها .
4. تتبع التقدم : سيتم في هذه المرحلة إنشاء لوحة التحكم لتتبع تقدم المشروع و قراءة حالة التقدم في المهام , نفذت ضمن Sprint مستقل لضمان اكتمال العمليات الأساسية .
5. التحكم في الإصدارات : تمكين الطلاب من ربط مشاريعهم مع مستودعات برمجية خاصة وعرض التغييرات في التعليمات البرمجية لكل مهمة .
6. الإشعارات الآلية : السماح بالنظام بإرسال إشعارات عند تعيين المهام أو اقتراب موعد التسليم .
7. إدارة التعليقات : السماح للمشرفين بالتعليق على مشاريع الطلاب وتقييم الأداء .



● أصحاب المصلحة Stakeholders :

Stakeholders	Role	Responsibility
Eng.Anas abdualaziz	Project Manager & Supervisor	Guides the project and provides feedback .
Souhaib sabboura	Software Engineer	Design and implement user interfaces , connect frontend with backend
Obada kahwajy	Software Engineer	Develop backend functionalities, implement authentication, handle database integrations .
Baraa abdallah	Software Engineer	Frontend development ,ensure a seamless user experience .

## ● وثيقة بيان العمل (SOW-Statement of Work) :

بيان العمل هو وثيقة شاملة تحدد نطاق العمل للمشروع من المهام المحددة والمخرجات والجدول الزمني والمسؤوليات، وتوفّر فهماً واضحاً لما يجب إنجازه وأهداف المشروع ومعايير النجاح .

## ● وصف المشروع :

بيان العمل هو وثيقة شاملة تحدد نطاق العمل للمشروع من المهام المحددة والمخرجات والجدول الزمني والمسؤوليات، وتوفّر فهماً واضحاً لما يجب إنجازه وأهداف المشروع ومعايير النجاح .

## ● نطاق المشروع : يشمل نطاق المشروع تصميم وتطوير منصة ويب مخصصة

لإدارة المشاريع الأكاديمية، تتيح لكل من الطلاب والمشرفين والإداريين إدارة المشاريع وفق أدوار محددة.

يُركز النظام على:

- تسجيل المستخدمين ( طلاب - مشرفين - إداريين )

- إدارة المشاريع والمهام المرتبطة بها .

- تسهيل التواصل بين أعضاء الفريق والمشرفين .

- رفع ومشاركة الملفات الأكاديمية

- عرض تقارير الأداء وتقديم العمل .

يستهدف المشروع المؤسسات التعليمية والجامعات، مع قابلية التوسع مستقبلاً لتشمل منصات أخرى لإدارة مشاريع تعليمية أو بحثية.

### ● أهداف المشروع :

- تطوير منصة ويب فعالة لإدارة ومتابعة المشاريع البرمجية الجامعية.
- تسهيل عملية التواصل بين الطلاب والمشرفين.
- توفير لوحة تحكم لمتابعة التقدم في المشاريع
- تحسين جودة الإشراف الأكاديمي من خلال الأتمتة والتوثيق الرقمي.
- تعزيز الشفافية وسرعة الوصول إلى بيانات المشاريع وتقارير الأداء.

### ● المخرجات ( Deliverables ) :

- خطة المشروع
- وثيقة المتطلبات البرمجية ( SRS-Software Requirements Specification)
- النظام البرمجي نفسه (Project Management System)، ويشمل الواجهة الأمامية والخلفية .
- التقرير النهائي للمشروع .

## ● متطلبات المشروع :

التكنولوجيا والأدوات:

- لغات البرمجة : PHP,HTML,CSS,JavaScript
- أطر العمل : React + MUI(Frontend) ,Laravel ( Backend)
- قاعدة البيانات : MySQL
- نظام التحكم بالإصدارات : GitHub API integration

## ● الافتراضات (Assumptions):

- توفر أعضاء الفريق والمشرفين بشكل مستمر لتقديم الملاحظات والمراجعات.
- إشراف ومتابعة دورية من مشرف المشروع لضمان التقدم المنتظم
- استقرار بيئة التطوير وقاعدة البيانات أثناء فترة التنفيذ.
- تعاون مستمر مع أصحاب المصلحة لتحديث متطلبات النظام إن لزم الأمر.

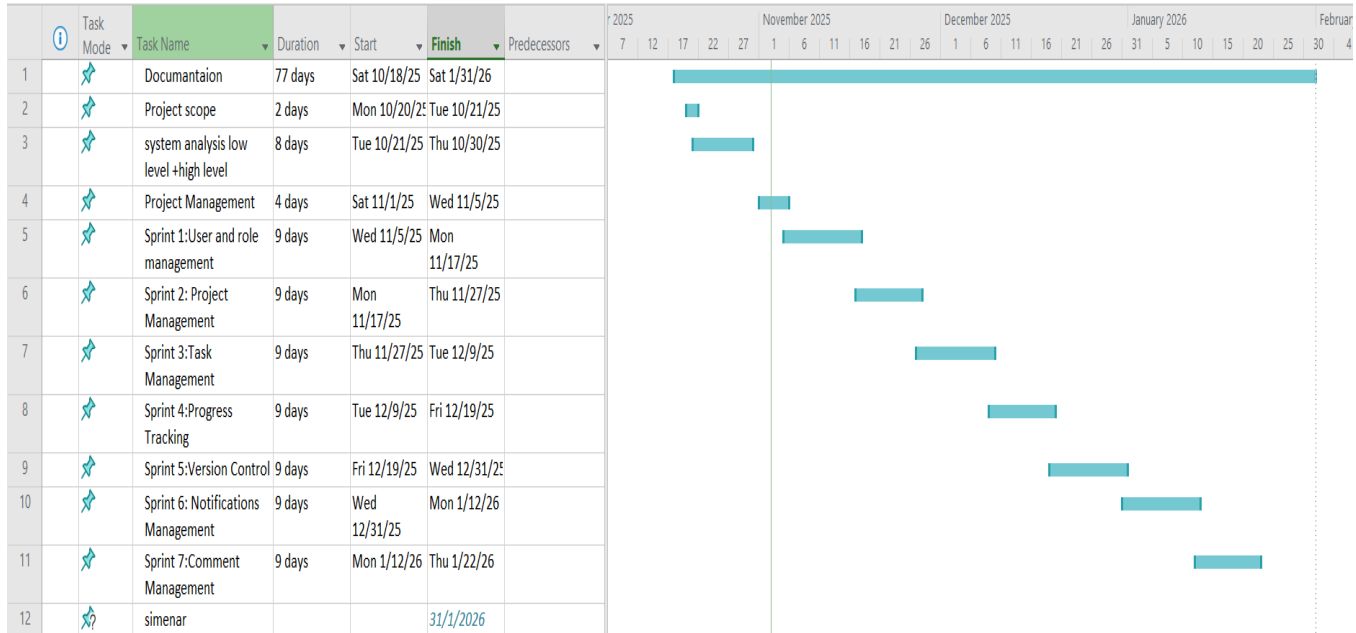
## ● الموارد البشرية (Project Resources):

- م. أنس عبد العزيز: مشرف المشروع (Project Manager)
- محمد صهيب صبورة : مطور واجهة أمامية (Frontend Developer)
- عبادة قهوجي : مطور واجهة خلفية (Backend Developer)
- براء عبدالله : مطور واجهة أمامية (Frontend Developer)

## ● الجدول الزمني للمشروع (Schedule):

Project Management System		
المرحلة	التاريخ	الوصف
بداية المشروع	18/Oct/2025	بدء التحليل وجمع المتطلبات
السيمنار الأول	16/Nov/2025	عرض التقدم الأولي والتحليل
اللجنة الفنية	10/Jan/2026	مناقشة النسخة شبه النهائية
الانتهاء من المشروع		تسليم النظام والتقارير النهائي
المناقشة النهائية		عرض المشروع أمام اللجنة

## ● خطة المشروع \_مخطط غانت Gantt Chart :



## 5. إدارة المخاطر Risk Management :

Risk Title	Risk Description	Raised Date	Tracking Frequency	State	Impact	Mitigation Plan
تأخير في الجدول الزمني	عدم الالتزام بالجدول الزمني بسبب تعقيدات في التطوير أو نقص الموارد	18/10/2025	أسبوعي	Active	High	تقسيم المشروع إلى Sprints, استخدام أدوات مثل Gantt Chart, مع تحديد أولويات المهام .
تضارب في توزيع المهام	تراكم المهام على عضو واحد في الفريق	18/10/2025	أسبوعي	Active	Medium	استخدام Jira لتوزيع المهام واجتماعات يومية .
صعوبة تعلم تقنيات جديدة	تأخر الفريق في إتقان Laravel أو React	5/11/2025	أسبوعي	Active	High	تخصيص جلسات تعليمية أسبوعية .
مشاكل في تكامل الواجهات	تعارض بين واجهات Frontend و React Backend (laravel)	10/11/2025	يومي	Active	High	اختبار التكامل بشكل متكرر , استخدام وثائق API منفصلة , وتنسيق مستمر بين أعضاء الفريق .
نقص خبراء الاختبار	ظهور أخطاء غير مكتشفة بسبب عدم وجود خبير متخصص في الفريق	20/11/2025	أسبوعي	Active	High	تطبيق اختبارات يدوية لكل ميزة وتخصيص عضو لفهم أساسيات الاختبار

نقص مشاركة المشرفين والطلاب	قلة إقبال المشرفين والطلاب على استخدام النظام	1/12/2025	شهري	Potential	High	إجراء حملات توعية مسبقة وإعلانات مستهدفة على وسائل التواصل الاجتماعي .
مشاكل في نظام التعليقات	تعطل نظام تقييم المشاريع بعد التحديثات .	15/12/2025	أسبوعي	Potential	Medium	عمل نسخة احتياطية من قاعدة البيانات قبل أي تحديث .

## 6. مخطط : WBS (Work Breakdown Structure)

		Task Mode	Task Name	Duration	Start	Finish	Predecessors	Resource Names
1			University software pr	77 days	Sat 10/18/25	Thu 1/29/26		
2			project start up and	14 days	Sun 10/19/25	Tue 11/4/25		
3			▲ Scope Definition	9 days	Sun 10/19/25	Tue 10/28/25		
4			Reference Study	2 days	Sun 10/19/25	Mon 10/20/25		sohayb
5			Project Scope Defin	2 days	Sat 10/11/25	Sun 10/12/25		sohayb
6			High-Level Requirer	2 days	Mon 10/13/25	Tue 10/14/25		sohayb
7			Technical Scope Def	2 days	Wed 10/15/25	Thu 10/16/25		sohayb
8			▲ Project Management	7 days?	Fri 10/17/25	Sat 10/25/25		
9			Project Charter	2 days	Sun 10/19/25	Mon 10/20/25		sohayb
10			Statement of Work	3 days	Tue 10/21/25	Thu 10/23/25		baraa
11			Risk Management P	2 days	Sun 10/19/25	Mon 10/20/25		sohayb
12			Work Breakdown St	3 days	Mon 10/20/25	Wed 10/22/25		obada
13			Effort Estimation	2 days	Thu 10/23/25	Fri 10/24/25		sohayb
14			Project Plan Docum	1 day	Sat 10/25/25	Sat 10/25/25		sohayb
15			Sprint 1: User and Rol	13 days	Sun 10/26/25	Mon 11/10/25		
16			▲ Analysis	2 days	Sun 10/26/25	Mon 10/27/25		
17			Use Case Diagram (	1 day	Sun 10/26/25	Sun 10/26/25		sohayb
18			Use Case Specification	1 day	Tue 10/28/25	Tue 10/28/25		baraa
19			Sequence Diagram (Us	1 day	Wed 10/29/25	Wed 10/29/25		sohayb
20			Activity Diagram (User	1 day	Wed 10/29/25	Wed 10/29/25		sohayb
21			▲ Design	3 days	Tue 10/28/25	Thu 10/30/25		
22			Class Diagram (user	2 days	Tue 10/28/25	Wed 10/29/25		sohayb

24	✈	Implementation	7 days	Fri 10/31/25	Sat 11/8/25		
25	✈	Frontend Development	3 days	Wed 11/5/25	Fri 11/7/25		baraa
26	✈	Backend Development	7 days	Sun 11/2/25	Sun 11/9/25		obada
27	✈	Integration Testing	2 days	Mon 11/10/25	Tue 11/11/25		sohayb
28	✈	Testing	1 day	Sun 11/9/25	Sun 11/9/25		
29	✈	System Testing	1 day	Sun 11/9/25	Sun 11/9/25		baraa
30	✈	Acceptance Testing	1 day	Sun 11/9/25	Sun 11/9/25		sohayb
31	✈	Documentation	1 day?	Mon 11/10/25	Mon 11/10/25		
32	✈	Updating Project Doc	1 day	Mon 11/10/25	Mon 11/10/25		sohayb
33	✈	Sprint 2: project Management	15 days	Tue 11/11/25	Sat 11/29/25		
34	✈	Analysis	9 days	Wed 11/19/25	Sat 11/29/25		
35	✈	Use Case Diagram (project)	2 days	Sun 10/19/25	Mon 10/20/25		obada
36	✈	Use Case Specification	2 days	Fri 11/21/25	Sat 11/22/25		obada
37	✈	Sequence Diagram (Project)	3 days	Sun 11/23/25	Tue 11/25/25		sohayb
38	✈	Activity Diagram (Project)	2 days	Thu 11/27/25	Fri 11/28/25		
39	✈	Design	4 days	Tue 11/18/25	Fri 11/21/25		
40	✈	Class Diagram (Project)	2 days	Tue 11/18/25	Wed 11/19/25		sohayb
41	✈	ERD Diagram (Project)	3 days	Wed 11/19/25	Fri 11/21/25		baraa
42	✈	Implementation	3 days	Sat 11/22/25	Tue 11/25/25		
43	✈	Frontend Development	2 days	Sat 11/22/25	Sun 11/23/25		baraa
44	✈	Backend Development	2 days	Sun 11/23/25	Mon 11/24/25		obada

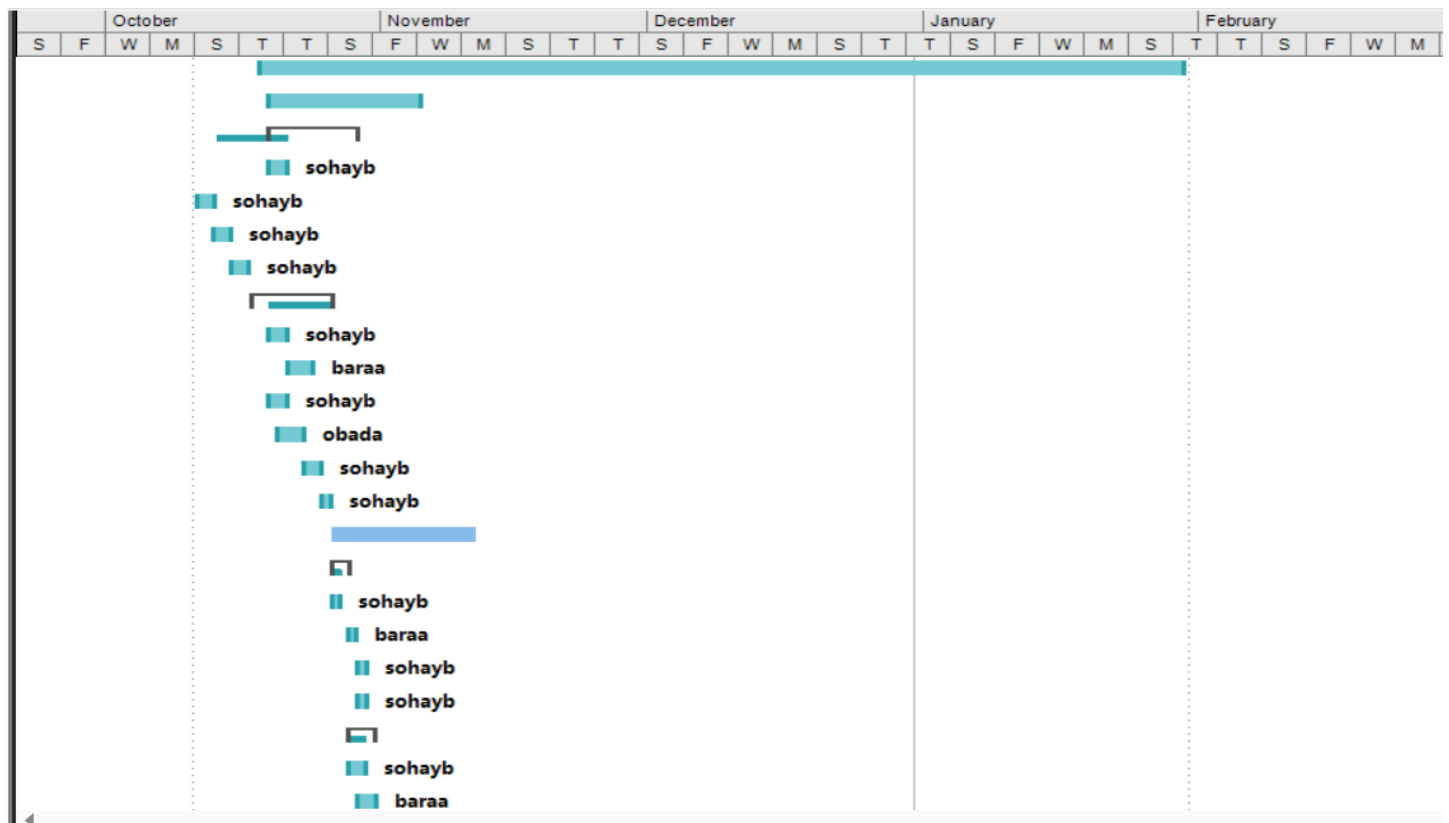
	Task Mode	Task Name	Duration	Start	Finish	Predecessors	Resource Names
45	✈	Integration Testing	1 day	Tue 11/25/25	Tue 11/25/25		sohayb
46	✈	Testing	2 days	Wed 11/26/25	Thu 11/27/25		
47	✈	System Testing	1 day	Wed 11/26/25	Wed 11/26/25		sohayb
48	✈	Acceptance Testing	1 day	Thu 11/27/25	Thu 11/27/25		baraa
49	✈	Documentation	1 day	Fri 11/28/25	Fri 11/28/25		
50	✈	Updating Project Doc	1 day	Fri 11/28/25	Fri 11/28/25		sohayb
51	✈	Sprint 3: Task Management	12 days	Sat 11/29/25	Mon 12/15/25		
52	✈	Analysis	8 days	Sun 11/30/25	Tue 12/9/25		
53	✈	Use Case Diagram (Task)	2 days	Sun 11/30/25	Mon 12/1/25		sohayb
54	✈	Use Case Specification	2 days	Tue 12/2/25	Wed 12/3/25		sohayb
55	✈	Sequence Diagram (Task)	2 days	Thu 12/4/25	Fri 12/5/25		baraa
56	✈	Activity Diagram (Task)	2 days	Sat 12/6/25	Mon 12/8/25		
57	✈	Design	3 days	Sat 12/6/25	Tue 12/9/25		
58	✈	Class Diagram (Task)	2 days	Sat 12/6/25	Sun 12/7/25		sohayb
59	✈	ERD Diagram (Task Management)	2 days	Mon 12/8/25	Tue 12/9/25		baraa
60	✈	Implementation	3 days	Wed 12/10/25	Fri 12/12/25		
61	✈	Frontend Development	1 day	Wed 12/10/25	Wed 12/10/25		sohayb
62	✈	Backend Development	1 day	Thu 12/11/25	Thu 12/11/25		obada
63	✈	Integration Testing	1 day	Fri 12/12/25	Fri 12/12/25		baraa
64	✈	Testing	2 days	Sat 12/13/25	Sun 12/14/25		
65	✈	System Testing	1 day	Sat 12/13/25	Sat 12/13/25		sohayb
66	✈	Acceptance Testing	1 day	Sun 12/14/25	Sun 12/14/25		baraa

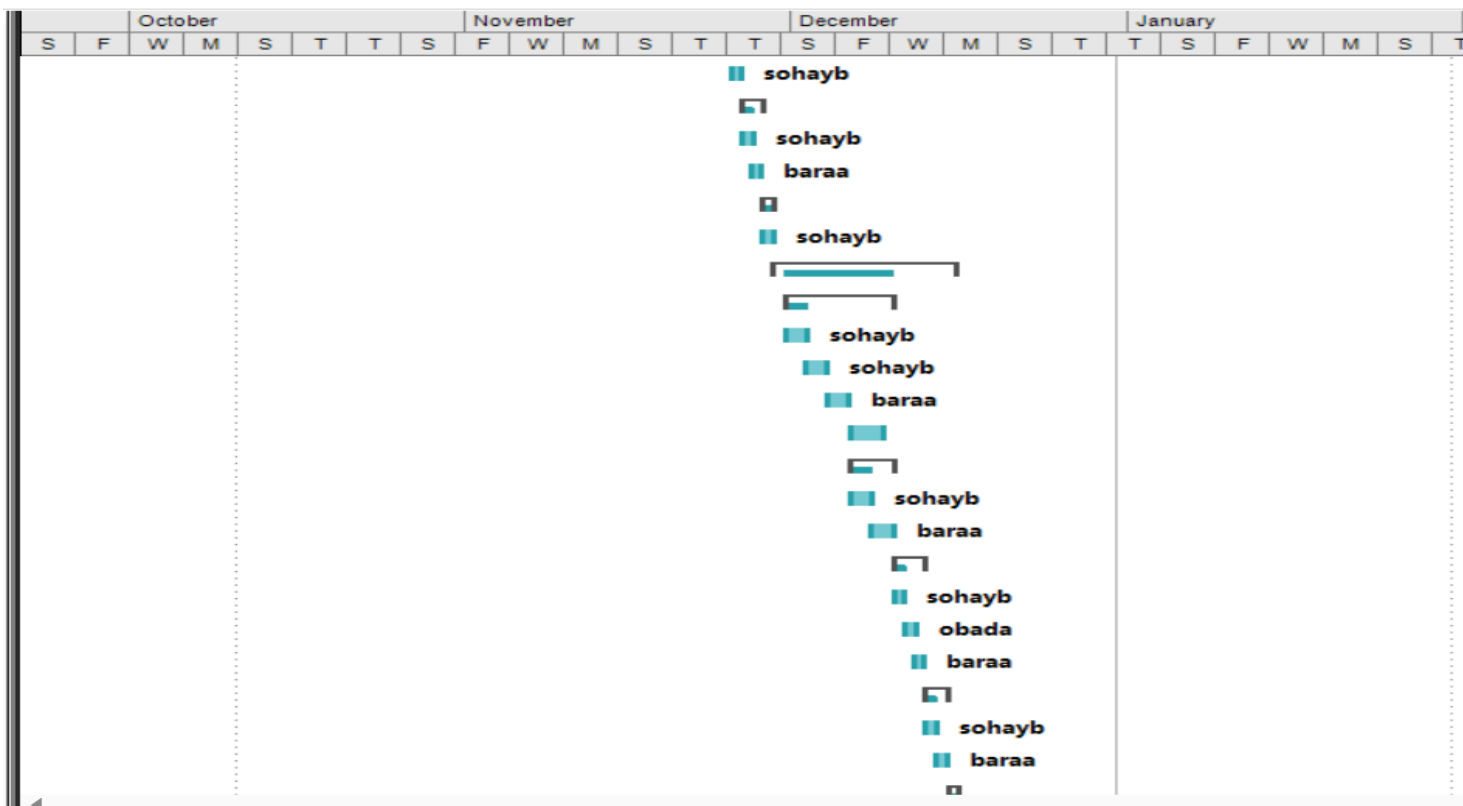
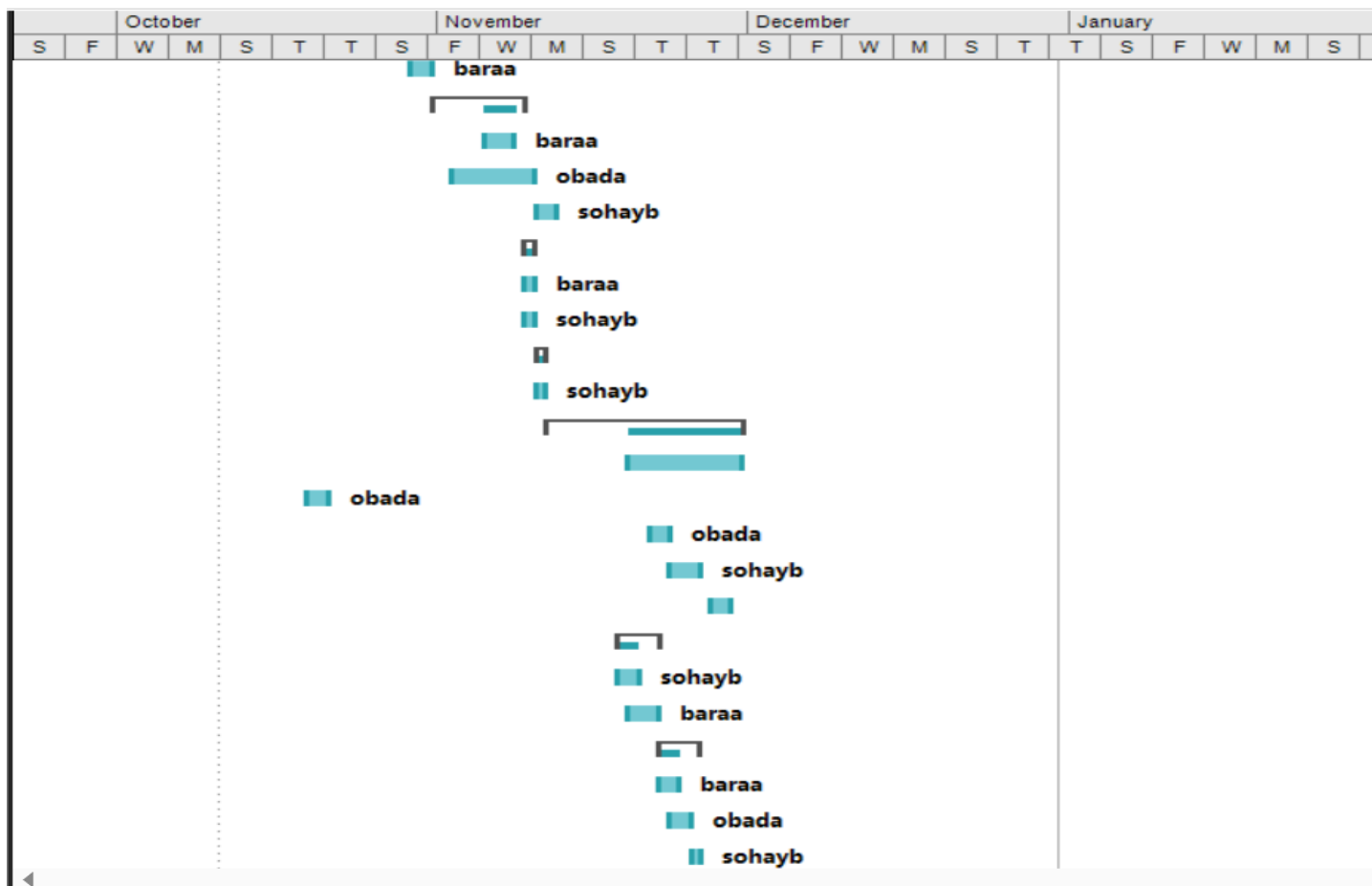


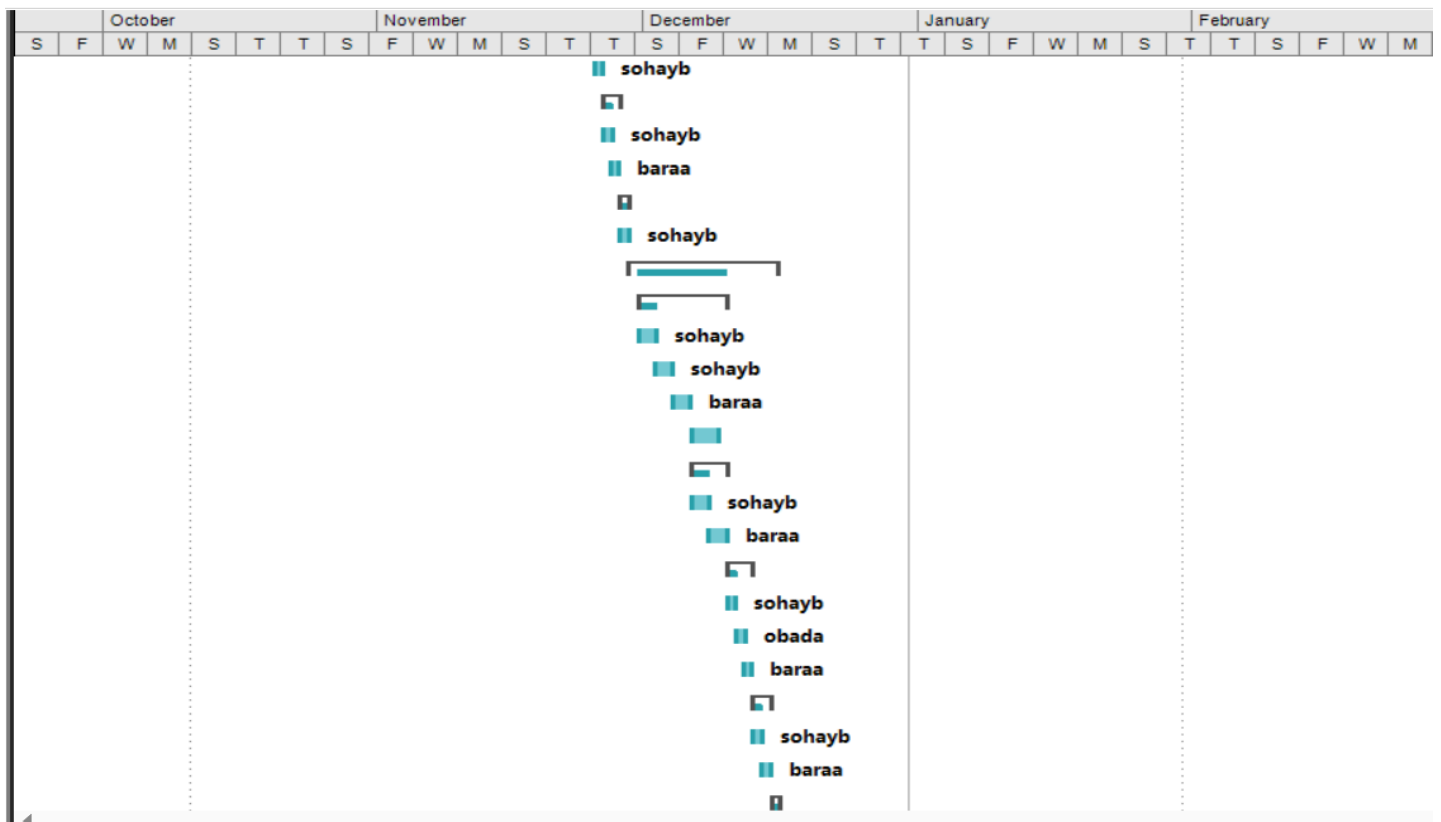
		Task Mode	Task Name	Duration	Start	Finish	Predecessors	Resource Names
45			Integration Testing	1 day	Tue 11/25/25	Tue 11/25/25		sohayb
46			▀ Testing	2 days	Wed 11/26/25	Thu 11/27/25		
47			System Testing	1 day	Wed 11/26/25	Wed 11/26/25		sohayb
48			Acceptance Testing	1 day	Thu 11/27/25	Thu 11/27/25		baraa
49			▀ Documentation	1 day	Fri 11/28/25	Fri 11/28/25		
50			Updating Project D	1 day	Fri 11/28/25	Fri 11/28/25		sohayb
51			▀ Sprint 3: Task Manage	12 days	Sat 11/29/25	Mon 12/15/25		
52			▀ Analysis	8 days	Sun 11/30/25	Tue 12/9/25		
53			Use Case Diagram	2 days	Sun 11/30/25	Mon 12/1/25		sohayb
54			Use Case Specification	2 days	Tue 12/2/25	Wed 12/3/25		sohayb
55			Sequence Diagram (Ta	2 days	Thu 12/4/25	Fri 12/5/25		baraa
56			Activity Diagram (Task	2 days	Sat 12/6/25	Mon 12/8/25		
57			▀ Design	3 days	Sat 12/6/25	Tue 12/9/25		
58			Class Diagram (Task	2 days	Sat 12/6/25	Sun 12/7/25		sohayb
59			ERD Diagram (Task Ma	2 days	Mon 12/8/25	Tue 12/9/25		baraa
60			▀ Implementation	3 days	Wed 12/10/25	Fri 12/12/25		
61			Frontend Developm	1 day	Wed 12/10/25	Wed 12/10/25		sohayb
62			Backend Development	1 day	Thu 12/11/25	Thu 12/11/25		obada
63			Integration Testing	1 day	Fri 12/12/25	Fri 12/12/25		baraa
64			▀ Testing	2 days	Sat 12/13/25	Sun 12/14/25		
65			System Testing	1 day	Sat 12/13/25	Sat 12/13/25		sohayb
66			Acceptance Testing	1 day	Sun 12/14/25	Sun 12/14/25		baraa

		Task Mode	Task Name	Duration	Start	Finish	Predecessors	Resource Names
67			▀ Documentation	1 day	Mon 12/15/25	Mon 12/15/25		
68			Updating Project D	1 day	Mon 12/15/25	Mon 12/15/25		sohayb
69			▀ Sprint 4: Progress Trac	16 days?	Tue 12/16/25	Tue 1/6/26		
70			▀ Analysis	7 days	Tue 12/16/25	Wed 12/24/25		
71			Use Case Diagram	2 days	Tue 12/16/25	Wed 12/17/25		sohayb
72			Use Case Specification	2 days	Thu 12/18/25	Fri 12/19/25		sohayb
73			Sequence Diagram (Pr	2 days	Sat 12/20/25	Mon 12/22/25		baraa
74			Activity Diagram (Prog	2 days	Tue 12/23/25	Wed 12/24/25		
75			▀ Design	3 days	Thu 12/25/25	Sun 12/28/25		
76			Class Diagram (Prog	2 days	Thu 12/25/25	Fri 12/26/25		sohayb
77			ERD Diagram (Progress	2 days	Mon 10/27/25	Tue 10/28/25		baraa
78			▀ Implementation	3 days	Mon 12/29/25	Wed 12/31/25		
79			Frontend Developm	1 day	Mon 12/29/25	Mon 12/29/25		sohayb
80			Backend Development	1 day	Tue 12/30/25	Tue 12/30/25		obada
81			Integration Testing	1 day	Wed 12/31/25	Wed 12/31/25		baraa
82			▀ Testing	2 days	Thu 1/1/26	Fri 1/2/26		
83			System Testing	1 day	Thu 1/1/26	Thu 1/1/26		sohayb
84			Acceptance Testing	1 day	Fri 1/2/26	Fri 1/2/26		baraa
85			▀ Documentation	1 day	Sat 1/3/26	Sat 1/3/26		
86			Updating Project D	1 day	Sat 1/3/26	Sat 1/3/26		sohayb
87			▀ Sprint 5: Notification	19 days?	Wed 1/7/26	Sat 1/31/26		
88			▀ Analysis	8 days	Wed 1/7/26	Fri 1/16/26		

		Task Mode ▾	Task Name ▾	Duration ▾	Start ▾	Finish ▾	Predecessors ▾	Resource Names ▾
84			Acceptance Testing	1 day	Fri 1/2/26	Fri 1/2/26		baraa
85			▸ Documentation	1 day	Sat 1/3/26	Sat 1/3/26		
86			Updating Project Doc	1 day	Sat 1/3/26	Sat 1/3/26		sohayb
87			▸ Sprint 5: Notification	19 days?	Wed 1/7/26	Sat 1/31/26		
88			▸ Analysis	8 days	Wed 1/7/26	Fri 1/16/26		
89			Use Case Diagram	2 days	Wed 1/7/26	Thu 1/8/26		sohayb
90			Use Case Specification	2 days	Fri 1/9/26	Mon 1/12/26		obada
91			Sequence Diagram (No	2 days	Tue 1/13/26	Wed 1/14/26		obada
92			Activity Diagram (Notif	2 days	Thu 1/15/26	Fri 1/16/26		baraa
93			▸ Design	3 days	Sat 1/17/26	Tue 1/20/26		
94			Class Diagram (Notif	2 days	Sat 1/17/26	Mon 1/19/26		sohayb
95			ERD Diagram (Notificat	2 days	Mon 1/19/26	Tue 1/20/26		baraa
96			▸ Implementation	6 days	Wed 1/21/26	Wed 1/28/26		
97			Frontend Developm	2 days	Wed 1/21/26	Thu 1/22/26		sohayb
98			Backend Development	2 days	Fri 1/23/26	Mon 1/26/26		obada
99			Integration Testing	2 days	Mon 1/26/26	Tue 1/27/26		baraa
100			▸ Testing	2 days	Thu 1/29/26	Fri 1/30/26		
101			System Testing	1 day	Thu 1/29/26	Thu 1/29/26		sohayb
102			Acceptance Testing	1 day	Fri 1/30/26	Fri 1/30/26		baraa
103			▸ Documentation	1 day	Sat 1/31/26	Sat 1/31/26		
104			Updating Project Doc	1 day	Sat 1/31/26	Sat 1/31/26		sohayb
105								







# Chapter 3

## Fundamental Concepts And Literature Review

## 1. مقدمة :

في هذا الفصل سنقوم بدراسة مرجعية تهدف إلى تحليل الأنظمة المشابهة لمشروعنا " نظام إدارة ومتابعة المشاريع البرمجية الجامعية " . سنتناول مجموعة من الأنظمة التي تقدم خدمات قريبة أو جزئية من أهداف مشروعنا , وذلك بهدف التعرف على أبرز الميزات والتقنيات المستخدمة فيها , واكتشاف نقاط القوة والضعف للاستفادة منها في تحسين وتطوير نظامنا .

## 2. المفاهيم الأساسية :

يشمل هذا القسم شرحاً للمفاهيم والمصطلحات الأساسية ذات الصلة بمشروعنا :

### ● إدارة المشروع - Project Management :

هو استخدام النظام لإدارة المشروع وتتبع التقدم بدءاً من مرحلة التخطيط وصولاً إلى مرحلة التنفيذ ومن ثم مرحلة النشر

### ● المنتج الأدنى القابل للتطبيق ( MVP ) :

هو إصدار أولي من المنتج يحتوي فقط على الوظائف الأساسية , يستخدم لاختبار الفكرة وجمع ملاحظات المستخدمين قبل التوسع في التطوير

### ● توفير وقت المشرفين ورفع جودة المخرجات التعليمية :

يقوم بتوفير وقت المشرفين من خلال أتمتة المتابعة وتقييم مشاريع الطلاب وتوثيق الأعمال الطلابية وأرشيف للمشاريع المتميزة أيضاً .

## 3. مراجعة الأدبيات :

الغاية من هذا القسم هو تحليل وتقييم الأنظمة التي تتعلق بنظامنا الحالي , حيث تهدف هذه الدراسة إلى تقديم نظرة عامة شاملة على الأنظمة العاملة في مجال إدارة ومتابعة المشاريع البرمجية , والاستفادة من التجارب السابقة في تطوير نظامنا الخاص .

## JIRA :

هو نظام احترافي لإدارة المشاريع البرمجية، خاصة تلك التي تعتمد على منهجيات Agile وScrum. يُستخدم على نطاق واسع في الشركات التقنية لتتبع الأخطاء، تنظيم المهام، وتخطيط الإصدارات. يتميز بمرونته العالية وتكامله مع أدوات التطوير مثل Git وBitbucket.



### الايجابيات :

قوية جداً في تتبع الأخطاء وادارة المشاريع البرمجية  
تدعم منهجيات Agile و Scrum بشكل ممتاز  
مناسب للفرق الكبيرة التي تعمل على مشاريع برمجية معقدة وتحتاج إلى تتبع دقيق.

### السلبيات :

واجهة معقدة نسبياً للمستخدمين الجدد  
مكلفة  
تحتاج إلى تدريب للشركات الصغيرة لفهم جميع الميزات

### : TRELLO

نظام بسيط وسهل الاستخدام يعتمد على مفهوم لوحات Kanban، حيث يتم تنظيم المهام في بطاقات يمكن نقلها بين الأعمدة حسب حالة التنفيذ. مناسب جداً للفرق الصغيرة والمشاريع غير المعقدة،



### الايجابيات :

واجهة مستخدم سهلة وبسيطة  
تعتمد على لوحات Kanban المرئية  
مجانية في خطتها الأساسية للمشاريع الصغيرة او الفرق غير التقنية

### السلبيات :

1. لا تدعم إدارة المشاريع المعقدة بشكل جيد
2. محدودة في التقارير
3. لا يحتوي على نظام تتبع أخطاء متقدم

### : Redmine

نظام مفتوح المصدر لإدارة المشاريع، يُستخدم بشكل خاص من قبل الفرق التقنية التي تفضل الاستضافة الذاتية. يدعم تتبع القضايا، إدارة الوقت، المهام الفرعية، والمنتديات الداخلية. يتميز بإمكانية التخصيص العالية .



### الإيجابيات :

مفتوح المصدر ومجاني وقابل للتخصيص  
يدعم ادارة مشاريع متعددة ومهام بدرجة عالية  
مناسب للفرق التقنية التي تفضل الاستضافة

### السلبات :

واجهة قديمة وغير جذابة  
يحتاج إعداد وتكوين يدوي  
لا يحتوي دعم رسمي او تحديثات مستمرة

### : GitHub Classroom

منصة تربط بين المعلم والطلاب عبر GitHub لتوزيع ومتابعة وتصحيح الواجبات  
البرمجية



### الإيجابيات :

1. توفير الوقت وتنظيم أفضل
2. التصحيح الآلي
3. بيئة عمل واقعية مشابهة لبيئات التطوير
4. مجاني تماماً

### السلبات :

1. تعقيد في الإعداد يحتاج إلى إعداد مسبق
2. يركز على الجانب التقني فقط
3. غير مناسب للمبتدئين



فيما يلي ملخص للخدمات المتميزة لكل من أنظمة إدارة المشاريع البرمجية الأربعة المذكورة:

(\*) تعني الميزات التي سيتم تحقيقها في المستقبل , خارج نطاق الفصل الدراسي .

System Feature	Jira	Trello	Redmine	Git Hub Class	Our System
واجهة مستخدم بسيطة		✓		✓	✓
نظام اشعارات	✓	✓	✓	✓	✓
متابعة التقدم	✓	✓	✓	✓	✓
البحث المتقدم	✓		✓		✓
التواصل بين المشرف والطالب				✓	✓
اللغات المتعددة	✓	✓	✓	✓	✓*
مجاني بالكامل			✓	✓	✓

# Chapter 4

## System Analysis

## 1. مقدمة :

يركز هذا الفصل على التحليل التفصيلي لنظام إدارة ومتابعة المشاريع البرمجية الجامعية , يتمثل الهدف الرئيسي في تحليل احتياجات المشرفين والسياق الذي يعملون فيه , مما يساعد على تحديد المتطلبات الأساسية والوظائف الضرورية التي يجب أن يتضمنها نظام إدارة المشاريع . يتناول الفصل استكشافاً للوظائف لإدارة ومتابعة المشاريع , بالإضافة إلى تحليل شامل لحالات الاستخدام وسير العمل, ما يوفر فهماً شاملاً لكيفية تفاعل المشرفين مع النظام وكيفية تطبيق الإدارة والمتابعة من خلالها .

## 2. مواصفات متطلبات البرمجيات (SRS):

مواصفات متطلبات البرمجيات تعتبر وثيقة هامة في عملية تطوير البرمجيات . تُستخدم SRS لوصف المتطلبات الوظيفية وغير الوظيفية للنظام البرمجي الذي سيتم تطويره .

تشمل مواصفات متطلبات البرمجيات عادةً عناصر مثل :

1. وصف النظام : توضح فهماً شاملاً للنظام المراد تطويره واستخدامه .
2. متطلبات النظام : تصف الأنشطة والعمليات التي يجب أن يقوم النظام بتنفيذها .
3. متطلبات الأداء : تحدد كيفية أداء النظام في مختلف الظروف والمتطلبات .
4. متطلبات الواجهة : تحدد كيفية تفاعل النظام مع المستخدم ومع أنظمة أخرى .
5. متطلبات الأمان : تصف كيفية حماية البيانات والمعلومات وضمان سلامة النظام .
6. متطلبات الصيانة : توضح كيفية الحفاظ على النظام وتطويره في المستقبل .

وجود SRS يساعد على تحقيق التواصل الفعال بين فريق تطوير البرمجيات والعملاء أو المستخدمين ويضمن فهماً دقيقاً لمتطلبات النظام . كما يمكن استخدامه كمرجع أثناء تطوير البرمجيات للتحقق من تنفيذ كل المتطلبات بشكل صحيح .

تعتبر وثيقة SRS وثيقة أساسية تساعد على تحديد , توثيق , وتنظيم متطلبات البرمجيات بوضوح وإتقان , مما يسهل على فريق التطوير تلبية تلك المتطلبات بنجاح .

### 3. مواصفات متطلبات البرمجيات - Software Requirement Specification (SRS) : Specification(SRS

#### Version 0.1

تاريخ المراجعات :

النسخة	سبب التغيير	التاريخ
0.1		9/Nov/2025
0.2	إضافة دور المشرف	20/Nov/2025
0.3	إضافة نظام الدعوات	19/Dec/2025

## 1. المقدمة :

### الهدف

تطوير منصة متخصصة لإدارة ومتابعة مشاريع البرمجيات الجامعية . تهدف إلى تسهيل التنسيق بين الطلاب والمشرفين والإداريين في البيئات الأكاديمية .

### نطاق المشروع

سيركز النطاق الرئيسي على إنشاء بيئة تعاونية شاملة , ستمكن هذه المنصة المستخدمين من التسجيل وإنشاء حسابات امنة , حيث سيكون بمقدور كل مجموعة طلابية إنشاء صفحة مشروعها الخاص , حيث يحددون عنوانه ويضعون وصفاً واضحاً لأهدافه , ويحددون متطلباته الأساسية .

## متطلبات عالية المستوى High-level Requirement :

- **إدارة الحسابات :** تتيح هذه الوظيفة في تحديد أدوار المستخدمين وفصل صلاحياتهم , حيث يتمكن المستخدم من دخول حسابه كمشرف أو طالب , ويتمكن من تسجيل الخروج بأمان .
- **إدارة المستخدمين :** النظام يسمح للمسؤول بإدارة مستخدمي النظام ( المشرفين - الطلاب )
- **إدارة المشاريع :** النظام يُمكن الطلاب من إنشاء مشروع خاص به يحتوي على معلومات عن هذا المشروع , ويتضمن تمكن المشرف من قراءة تفاصيل المشروع وعرض قائمة الأنشطة المكتملة وغير المكتملة .
- **إدارة المهام :** حسب نوع المستخدم في النظام وصلاحياته
  - يتمكن المشرف من :
  - إضافة مهام فرعية للمشروع مع وصف وتاريخ التسليم .
  - استعراض جميع المهام المرتبطة بمشروع معين .
  - تحديث وصف المهمة وتاريخ التسليم .
  - يتمكن الطالب من :
  - عرض وقراءة تفاصيل المهمة بما يتضمن وصفها .

- **تتبع التقدم :** يوفر النظام للمشرفين تتبع تقدم العمل وذلك من خلال عرض حالة المهام ( قيد الانتظار , قيد التنفيذ , مكتملة )
- **التحكم في الإصدارات :** يزود النظام الطالب بخاصية ربط مشروعه بمستودع التعليمات البرمجية , وقراءة سجل الإصدارات مما يمنحه عرض التغييرات في التعليمات البرمجية .
- **الإشعارات الآلية :** يقوم النظام بإرسال إشعارات عند تعيين مهمة أو اقتراب موعد التسليم وعرض سجل الإشعارات لكل مستخدم .
- **التعليقات :** يزود النظام للمشرف عملية كتابة التعليقات على المهام وإضافة تقييماته الخاصة

## الفاعلين Actors :

- **المشرف (Supervisor) :** المشرف هو المسؤول عن التوجيه والإشراف على الجودة . حيث يمكن للمشرف عرض وتعديل مقترحات مشاريع تخرج للطلاب ليختاروا منها , ومراقبة تقدم الطلاب أو الفرق في المشاريع المسندة إليهم . مراجعة تسليمات الطلاب ( نماذج أولية , كود ) وتقديم ملاحظات عليها و إبداء الرأي والمشورة للطلاب بشكل مستمر . بالإضافة إلى التقييم النهائي .
- **الطالب (Student) :** هو المستفيد الأساسي الرئيسي والمنفذ الأساسي في النظام , وهدفه الأساسي هو إكمال مشروعه بنجاح . حيث يقوم بالتسجيل الرسمي للمشروع بعد الاتفاق مع المشرف , العمل على مراحل المشروع المختلفة ورفع التسليمات في مواعيدها المحددة عبر النظام , تحديث حالة المهام ونسبة الإنجاز في النظام لمشاركة تقدمه مع المشرف . بالإضافة إلى رفع التقرير النهائي .
- **المسؤول (Admin) :** يدير المستخدمين حيث يستطيع إضافة وحذف وتعديل حسابات جميع المستخدمين (المشرفين , طلاب ) .

## 2. الوصف العام

### منظور المنتج

يُعد نظام إدارة ومتابعة المشاريع البرمجية الجامعية منتجاً ذو قيمة ملموسة حيث يحمل العديد من المزايا للمستخدمين إذ يتميز النظام بتوفير بيئة إلكترونية آمنة تسهل التواصل بين المشرفين والطلاب وتجعل عملية تتبع التقدم سلسلة وسهلة أيضاً . ويحل مشاكل حقيقية بطريقة أفضل من البدائل الحالية .

### مميزات المنتج

- **إدارة المستخدمين :**  
تسجيل الدخول للحسابات والخروج منه ( مشرف - طالب ) .
- **إدارة المشاريع :**  
انشاء مشروع خاص للطلاب يحتوي جميع معلومات المشروع  
عرض تفاصيل المشروع للمشرف وحالة الأنشطة التي يتم العمل عليها .
- **إدارة المهام :**  
إضافة مهام فرعية للمشروع وتحديد تاريخ التسليم ( للمشرف )  
استعراض جميع المهام المتعلقة بمشروع معين ( للمشرف )  
عرض وقراءة تفاصيل المهمة بما يتضمن الوصف الخاص بها ( للطلاب )
- **تتبع التقدم :**  
يوفر النظام تتبع سير العمل من خلال عرض حالة المهمة ( للمشرف )
- **الإشعارات الآلية :**  
يقوم النظام بإرسال إشعارات عند تعيين مهمة أو اقتراب موعد تسليم .

## ● التحكم بالإصدارات :

تزويد خاصية ربط المشروع بمستودع التعليمات البرمجية وعرض التغييرات في تلك التعليمات .

## فئات المستخدمين وخصائصهم :

### المشرف

- المسؤوليات : الإشراف على المشاريع البرمجية , إضافة مهام فرعية للمشروع واستعراض معلومات جميع المشاريع .
- الامتيازات : إدارة المشروع الجامعي , تتبع تقدم الطلاب والتعليق على مشاريعهم لإعطاء التقييم .

### الطالب

- المسؤوليات : تسجيل مشروعه الخاص , العمل على مراحل المشروع المختلفة , رفع التسليمات .
- الامتيازات : طرح أسئلة فنية واستقبال التغذية الراجعة من خلال النظام , تحديث حالة المهام لمشاركة تقدمه مع المشرف .

### المسؤول

- المسؤوليات : إدارة المستخدمين وتحديد صلاحيات كل من يستعمل النظام .
- الامتيازات : الإشراف العام ومراقبة سير جميع المشاريع بصورة شاملة على التقدم .



### 3. ميزات النظام

#### 3.1. المتطلبات الوظيفية \_ Functional Requirements :

##### 3.1.1. إدارة الحسابات :

###### ● المتطلب 1.1 :

يتيح النظام للمستخدمين بتسجيل الدخول من خلال اسم المستخدم وكلمة المرور

###### ● المتطلب 1.2 :

يتيح النظام للمستخدمين بتسجيل الخروج .

##### 3.1.2. إدارة المشاريع :

###### ● المتطلب 2.1 :

يتيح النظام للطالب بإنشاء مشروع وإضافة المعلومات الخاصة به .

###### ● المتطلب 2.2 :

يتيح النظام للمشرف بعرض تفاصيل المشاريع الخاص بالطلاب .

###### ● المتطلب 2.3 :

يتيح النظام للمسؤول بإنشاء مشروع جديد .

###### ● المتطلب 2.4 :

يتيح النظام للمسؤول بتعديل معلومات المشروع أو حذفه .

###### ● المتطلب 2.5 :

يسمح النظام للمسؤول بعرض كافة المشاريع المسجلة على النظام .

##### 3.1.3. إدارة المهام :

###### ● المتطلب 3.1 :

يتيح النظام للمشرف بإضافة مهام فرعية للمشروع مع وصف وتحديد موعد التسليم .

###### ● المتطلب 3.2 :

يتيح النظام للمشرف بعرض جميع المهام المتعلقة بمشروع معين .

###### ● المتطلب 3.3 :

يسمح النظام للطالب عرض وقراءة تفاصيل المهمة بما يتضمن الوصف الخاص بها .

#### 3.1.4. تتبع التقدم :

- المتطلب 4.1 :

يتيح النظام للمشرف بعرض حالة المهام ( قيد الإنتظار , قيد التنفيذ , منتهية )

#### 3.1.5. التحكم في الإصدارات :

- المتطلب 5.1 :

يتيح النظام بخاصية تزويده بربط مشروعه مع مستودع التعليمات البرمجية

#### 3.1.6. الإشعارات الآلية والتعليقات :

- المتطلب 6.1 :

النظام متاح بخاصية إرسال الإشعارات بحالة إضافة مهمة جديدة أو اقتراب موعد تسليم .

- المتطلب 6.2 :

يتيح النظام للمشرف بالتعليق على المشروع لإعطاء تقييمه .

### 3.2. المتطلبات غير الوظيفية \_ Non-Functional Requirements

- سهولة الاستخدام Usability :

يجب أن تكون واجهة المستخدم بسيطة وسهلة التنقل لمستخدمين .

- التوافر Availability :

يجب أن يحافظ النظام على جاهزية التشغيل لضمان التوافر المستمر لميزات نظام إدارة المشاريع .

- قابلية التوسع Scalability :

يجب أن يكون النظام قابلاً للتوسع لدعم عدد متزايد من المستخدمين دون انخفاض في الأداء .

## • التكامل integrity :

يجب أن يكون النظام قادراً على التكامل مع أنظمة أخرى مثل مستودعات إدارة المشاريع البرمجية وغيرها .

## • الأمان والخصوصية Security & Privacy :

تنفيذ المصادقة الأساسية وتخزين كلمات المرور مشفرة .

## 4. متطلبات النظام :

Req-ID	Requirement Title	Category	Priority
REQ-1.1	يجب على النظام السماح للمستخدم بتسجيل الدخول باستخدام اسم المستخدم وكلمة المرور	إدارة الحسابات	عالي
REQ-1.2	يجب على النظام السماح للمستخدم بتسجيل الخروج من حسابه	إدارة الحسابات	متوسط
REQ-2.1	يجب على النظام السماح للطالب بإنشاء مشروعه الخاص	إدارة المشاريع	عالي
REQ-2.2	يجب على النظام السماح للطالب برؤية تفاصيل مشروعه	إدارة المشاريع	عالي
REQ-2.3	يجب على النظام السماح للمشرف برؤية تفاصيل جميع المشاريع	إدارة المشاريع	عالي
REQ-2.4	يجب على النظام السماح للمسؤول بإنشاء مشروع	إدارة المشاريع	متوسط
REQ-2.5	يجب على النظام السماح للمسؤول بتعديل بيانات مشروع	إدارة المشاريع	متوسط
REQ-3.1	يجب على النظام السماح للمسؤول بحذف مشروع	إدارة المشاريع	متوسط

REQ-3.2	يجب على النظام السماح للمشرف بإضافة مهمة معينة	إدارة المهام	عالي
REQ-3.3	يجب على النظام السماح للطالب برؤية تفاصيل المهمة المسندة إليه	إدارة المهام	متوسط
REQ-3.4	يجب على النظام السماح للمشرف بعرض جميع المهام الخاصة بمشروع معين	إدارة المهام	متوسط
REQ-4.1	يجب على النظام السماح للمشرف بمتابعة تقدم طلابه	تتبع التقدم	عالي
REQ-5.1	يجب على النظام السماح للطالب بإنشاء مستودعه الخاص	التكامل مع المستودعات البرمجية	متوسط
REQ-5.2	يجب على النظام السماح للطالب برفع الكود الخاص به على المستودع البرمجي	التكامل مع المستودعات البرمجية	متوسط
REQ-6.1	يجب على النظام السماح للمشرف بإرسال الإشعارات للطلاب	الإشعارات	متوسط
REQ-6.2	يجب على النظام السماح للطلاب بتلقي الإشعارات	الإشعارات	متوسط
REQ-7.1	يجب على النظام السماح للمشرفين بكتابة التعليقات على مشاريع الطلاب	التعليقات	عالي
REQ-7.2	يجب على النظام السماح للطلاب بقراءة تعليقات المشرفين وملاحظاتهم	التعليقات	عالي
NFR-1	يجب أن تكون واجهة المستخدم بسيطة وسهلة التنقل والاستخدام	سهولة الاستخدام	عالي
NFR-2	يجب أن يحافظ على جاهزية التشغيل لضمان التوافر المستمر لميزات النظام	التوافر	عالي
NFR-3	يجب على النظام أن يكون قابلاً للتوسع لدعم عدد متزايد من المستخدمين	قابلية التوسع	متوسط
NFR-4	يجب على النظام أن يكون قابلاً للتكامل مع أنظمة أخرى مثل المستودعات البرمجية	التكامل	عالي

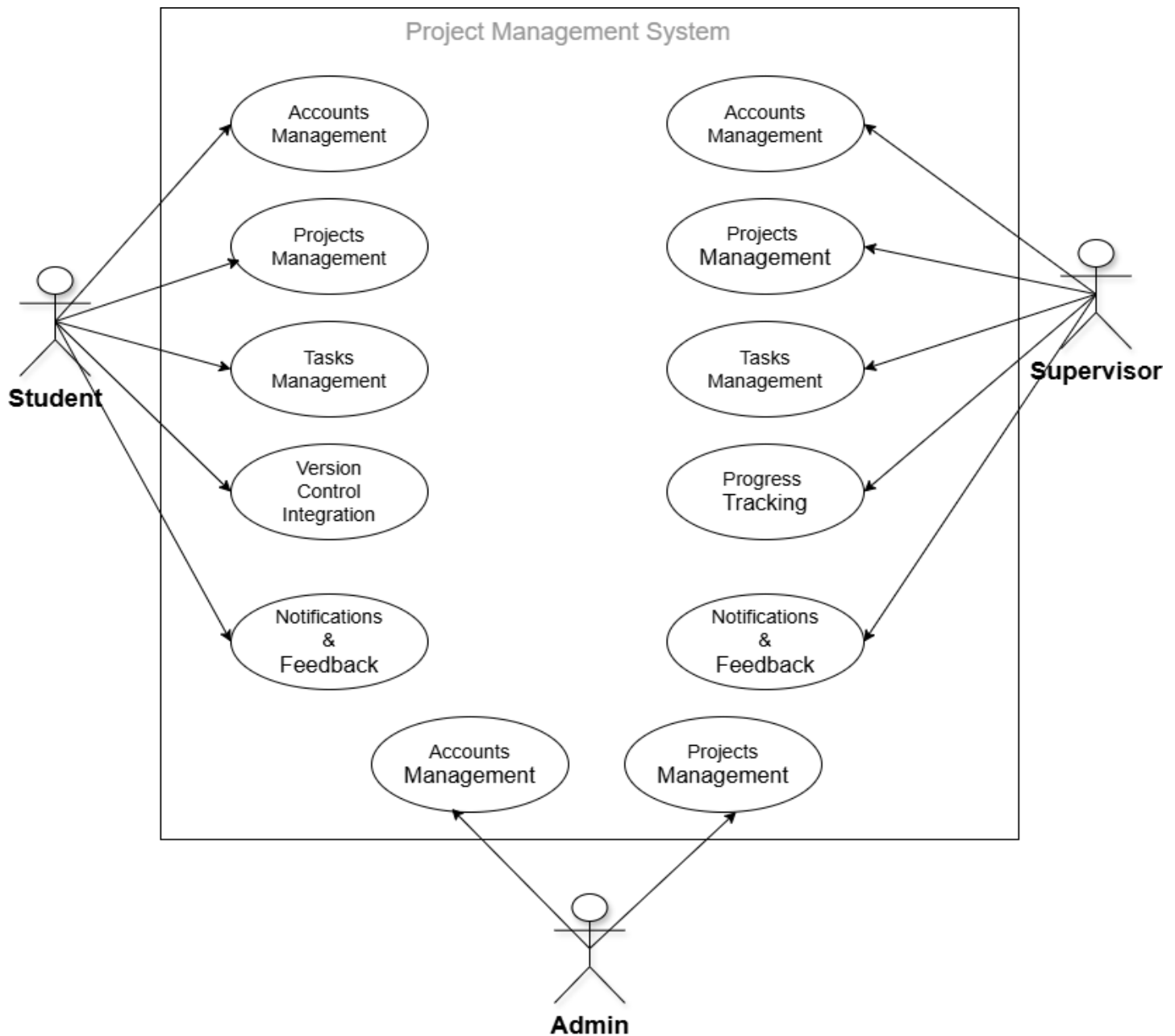
<b>NFR-5</b>	يجب على النظام تأمين الحماية والتشفير لبيانات المستخدم وكلمة المرور	الأمان والخصوصية	عالي
--------------	--	------------------	------

## 5. نمذجة متطلبات النظام :

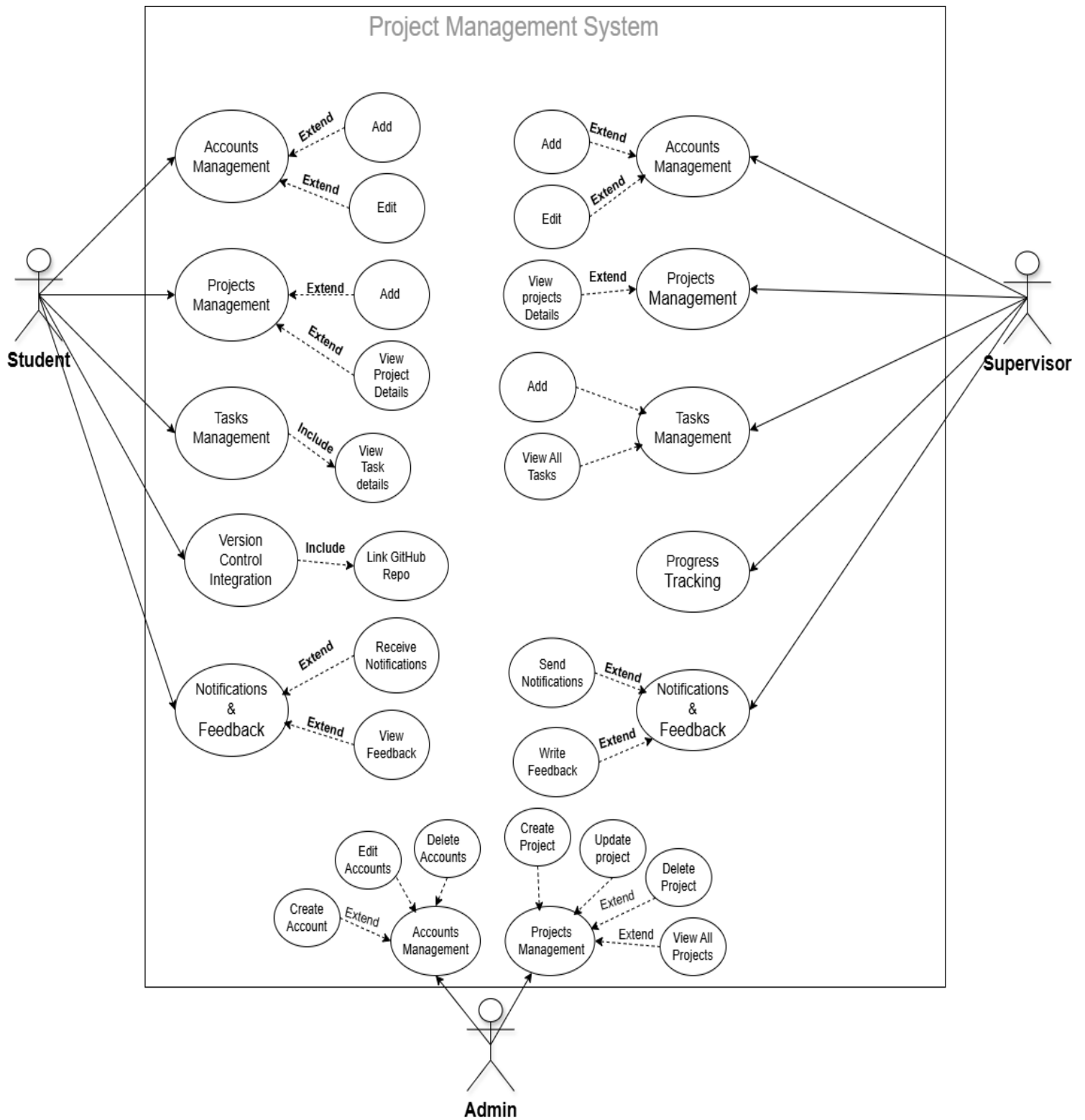
### 5.1. مخطط الحالة \_ Use Case Diagram :

تمثل مخططات Use Case سلوك النظام وتساعد في معرفة متطلبات النظام

High level - Use Case diagram :



## Low level - Use Case Diagram :



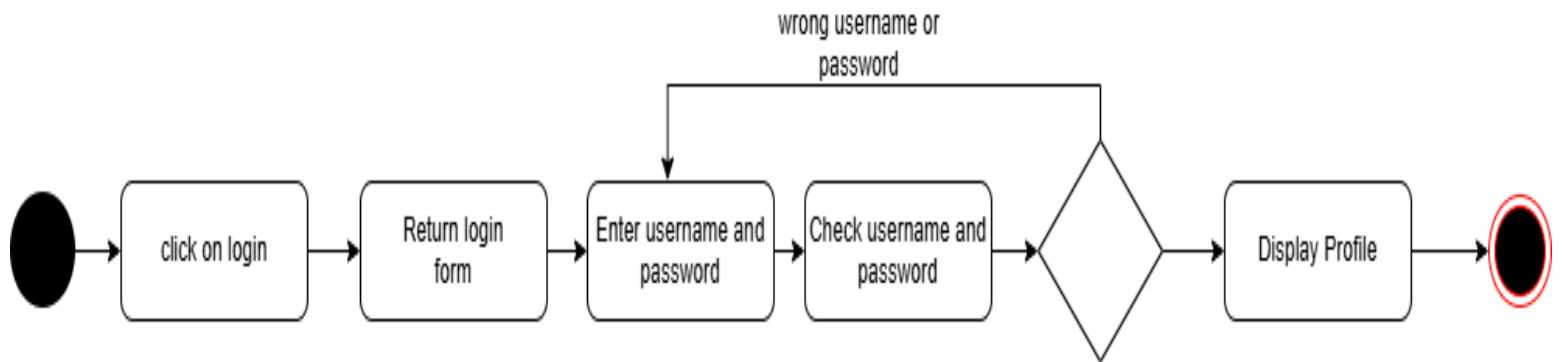
## 6. ميزات النظام - توصيف حالات الاستخدام ومخططات النشاط

### 6.1. Use Case Specifications :

#### ● تسجيل الدخول (UC-01) :

UC-01	Use Case ID
تسجيل الدخول	Use Case Name
المستخدمين	Actors
تسجيل الدخول لحساب المستخدم	Description
يجب أن يكون لديه حساب	Precondition
<p>1. يتطلب النظام معلومات تسجيل الدخول</p> <p>2. يقوم المستخدم بإدخال بياناته ( اسم المستخدم , كلمة المرور )</p> <p>3. يتحقق النظام من صحة البيانات المدخلة</p> <p>4. يتم إظهار الصفحة الرئيسية الخاصة بحساب المستخدم</p>	Main Flow
<p>في حال كانت البيانات خاطئة في المرة الأولى أو : A1 الثانية يعرض النظام رسالة خطأ</p> <p>في حالة ترك أحد الحقول فارغة يطلب النظام : A2 إدخال جميع الحقول</p>	Alternative Flow
يتم تسجيل المستخدم إلى حسابه بنجاح	Postconditions

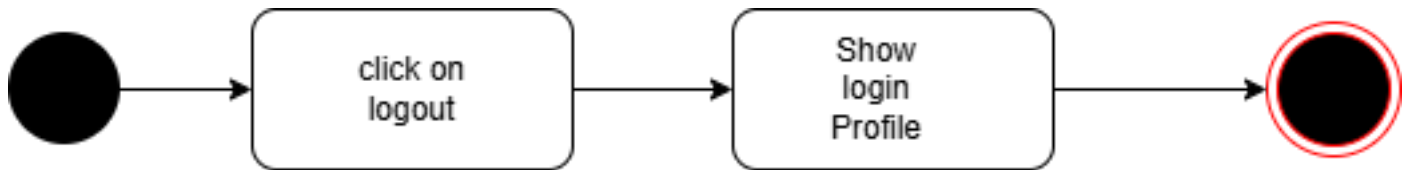




● تسجيل الخروج (UC-02) :

UC-02	Use Case ID
تسجيل الخروج	Use Case Name
المستخدمين	Actors
تسجيل الخروج من حساب المستخدم	Description
يجب أن يكون المستخدم مسجل الدخول إلى النظام	Precondition
1. يختار المستخدم خيار تسجيل الخروج 2. يقوم النظام بإنهاء الجلسة الحالية 3. يتم تحويل المستخدم إلى صفحة تسجيل الدخول	Main Flow
لا يوجد	Alternative Flow

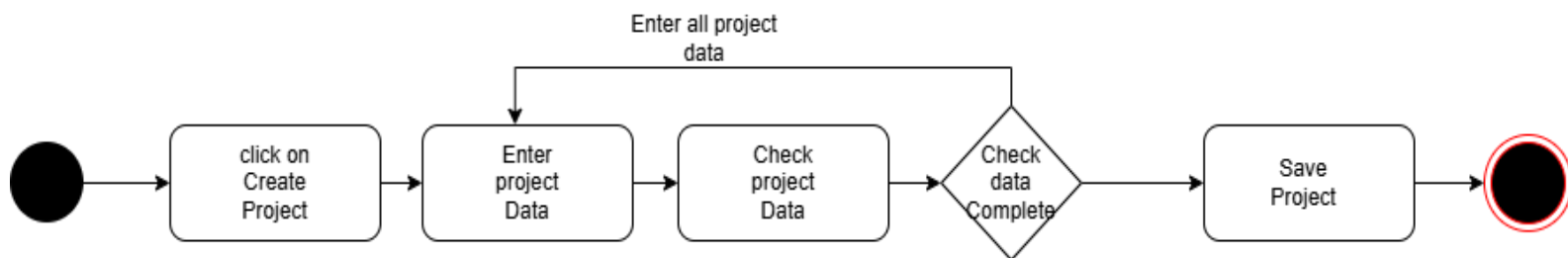
يتم تسجيل الخروج بنجاح	Postconditions
------------------------	----------------



● إنشاء مشروع (UC-03) :

UC-03	Use Case ID
إنشاء مشروع	Use Case Name
الطالب , المسؤول	Actors
يتيح النظام للطالب إنشاء مشروع جديد وإدخال بياناته الأساسية	Description
يجب أن يكون الطالب مسجل الدخول الى النظام	Precondition
1. يختار الطالب خيار إنشاء مشروع	Main Flow

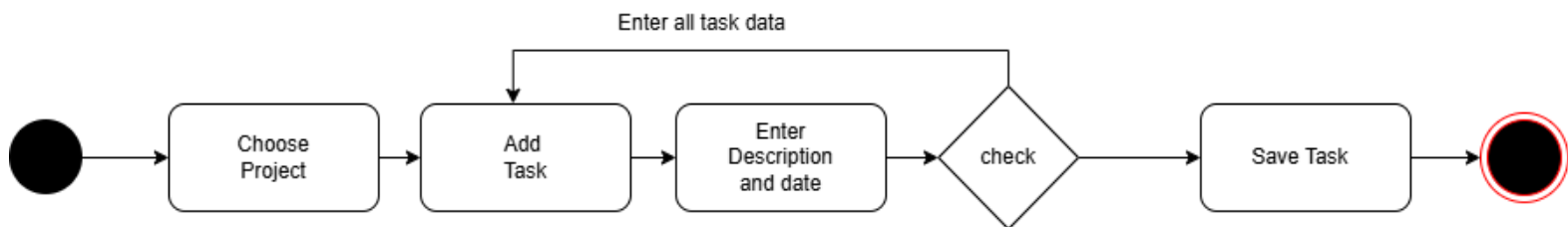
<p>2. يقوم بإدخال بيانات المشروع</p> <p>3. يتحقق النظام من صحة البيانات</p> <p>4. يتم حفظ المشروع في النظام</p>	
<p>في حال ادخال بيانات ناقصة يطلب النظام استكمال: A1 جميع الحقول اللازمة</p>	Alternative Flow
<p>يتم إنشاء المشروع بنجاح</p>	Postconditions



● إضافة مهمة (UC-04) :

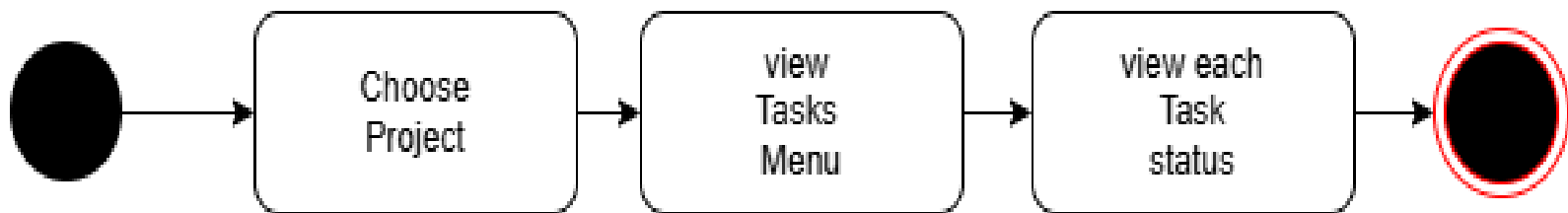
UC-04	Use Case ID
إضافة مهمة	Use Case Name
المشرف	Actors
يتيح النظام للمشرف إضافة مهمة فرعية لمشروع معين	Description

مع تحديد الوصف وموعد التسليم	
يجب أن يكون المشرف مسجل الدخول ولديه مشروع يقوم بالإشراف عليه	Precondition
1. يختار المشرف إضافة مهمة جديدة 2. يدخل وصف المهمة وموعد التسليم 3. يتحقق النظام من صحة البيانات المدخلة 4. يتم حفظ المهمة وربطها بالمشروع	Main Flow
في حالة ترك أحد الحقول فارغة يطلب النظام : A1 استكمال البيانات	Alternative Flow
تتم إضافة المهمة بنجاح	Postconditions



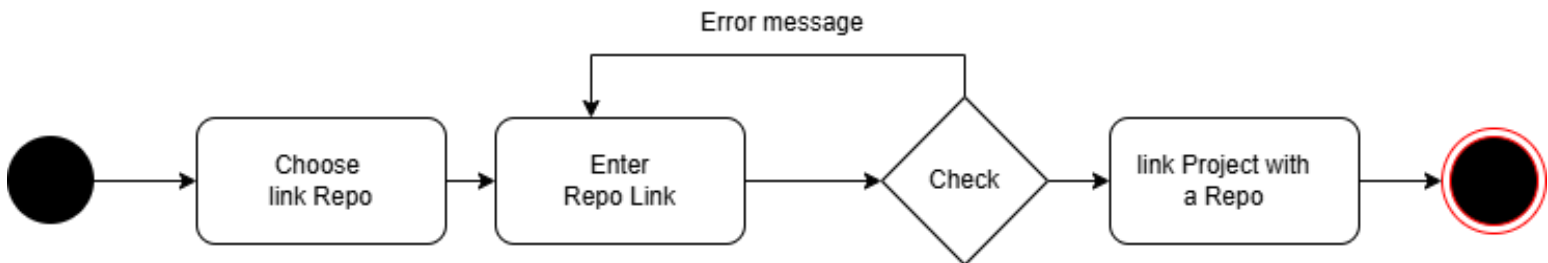
• تتبع التقدم (UC-05) :

UC-05	Use Case ID
تتبع تقدم المهام	Use Case Name
المشرف	Actors
يتيح النظام للمشرف متابعة حالة تنفيذ مهام المشروع	Description
يجب أن يكون المشرف مسجل الدخول	Precondition
1. يختار المشرف مشروعاً 2. يعرض النظام حالة كل مهمة ( قيد الانتظار , قيد التنفيذ , منتهية )	Main Flow
لا يوجد	Alternative Flow
يتم عرض حالة المهام	Postconditions



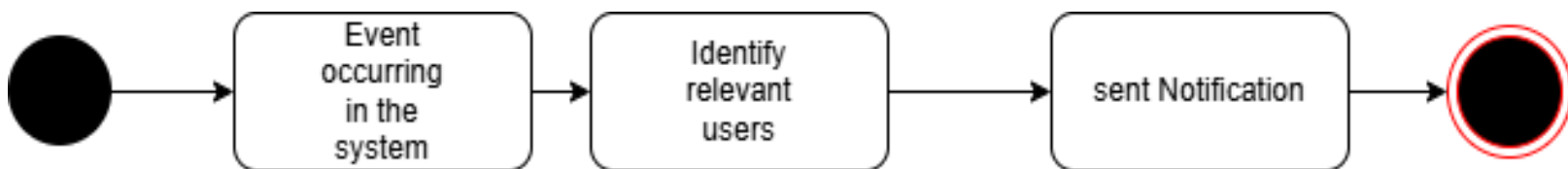
● التحكم بالإصدارات (UC-06) :

UC-06	Use Case ID
التحكم بالإصدارات	Use Case Name
الطالب	Actors
يتيح النظام للطالب ربط مشروعه بمستودع التعليمات البرمجية لمتابعة الإصدارات	Description
يجب أن يكون الطالب مسجل الدخول ولديه مشروع مسجل	Precondition
1. يختار الطالب خيار ربط المستودع البرمجي 2. يُدخل رابط المستودع 3. يتحقق النظام من صحة الرابط 4. يتم ربط المستودع بالمشروع	Main Flow
في حال إدخال رابط غير صالح , يعرض النظام : A1 رسالة خطأ , ويطلب إعادة الإدخال	Alternative Flow
يتم ربط المشروع بالمستودع البرمجي بنجاح	Postconditions



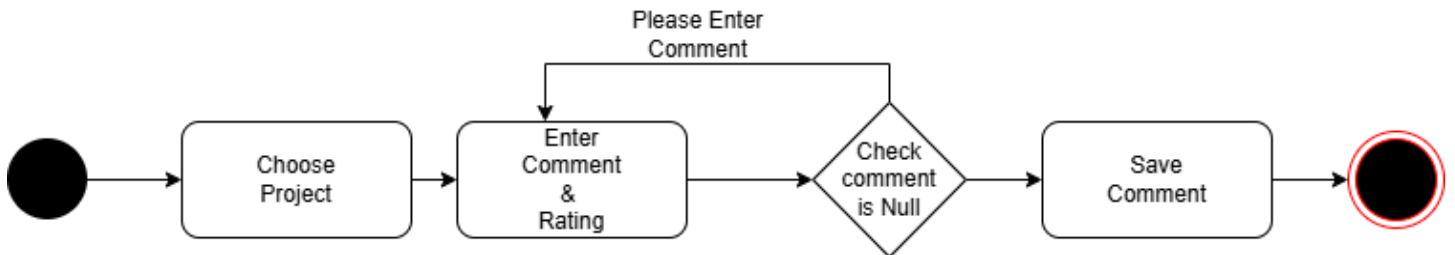
● الإشعارات الآلية (UC-07):

UC-07	Use Case ID
الإشعارات الآلية	Use Case Name
النظام	Actors
يقوم النظام بإرسال إشعارات تلقائية عند إضافة مهمة جديدة أو اقتراب موعد التسليم	Description
وجود حدث داخل النظام يستدعي إرسال إشعار	Precondition
1. يكتشف النظام الحدث 2. يحدد المستخدمين المعنيين 3. يقوم النظام بإرسال الإشعار	Main Flow
لا يوجد	Alternative Flow
يتم إرسال الإشعار بنجاح	Postconditions



● إضافة تعليق وتقييم (UC-08) :

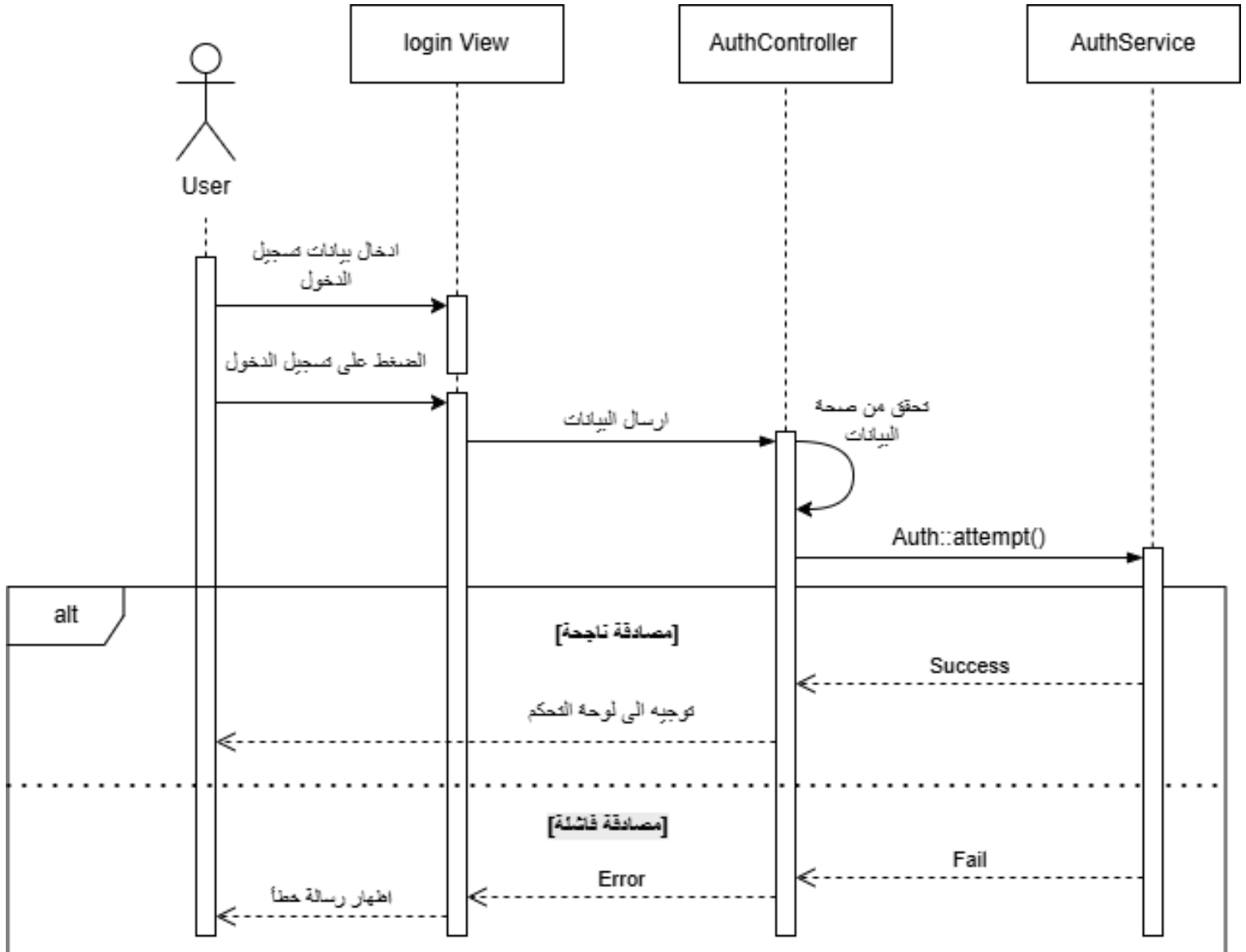
UC-08	Use Case ID
التعليقات والتقييم	Use Case Name
المشرف	Actors
يتيح النظام للمشرف إضافة تعليق وتقييم على مشروع الطالب	Description
يجب أن يكون المشرف مسجل الدخول ولديه مشروع يقوم بالإشراف عليه	Precondition
1. يختار المشرف المشروع 2. يدخل التعليق والتقييم 3. يقوم النظام بحفظ التعليق	Main Flow
في حال ترك حقل التعليق فارغاً يطلب النظام : A1 إدخاله	Alternative Flow
يتم حفظ التعليق والتقييم بنجاح	Postconditions



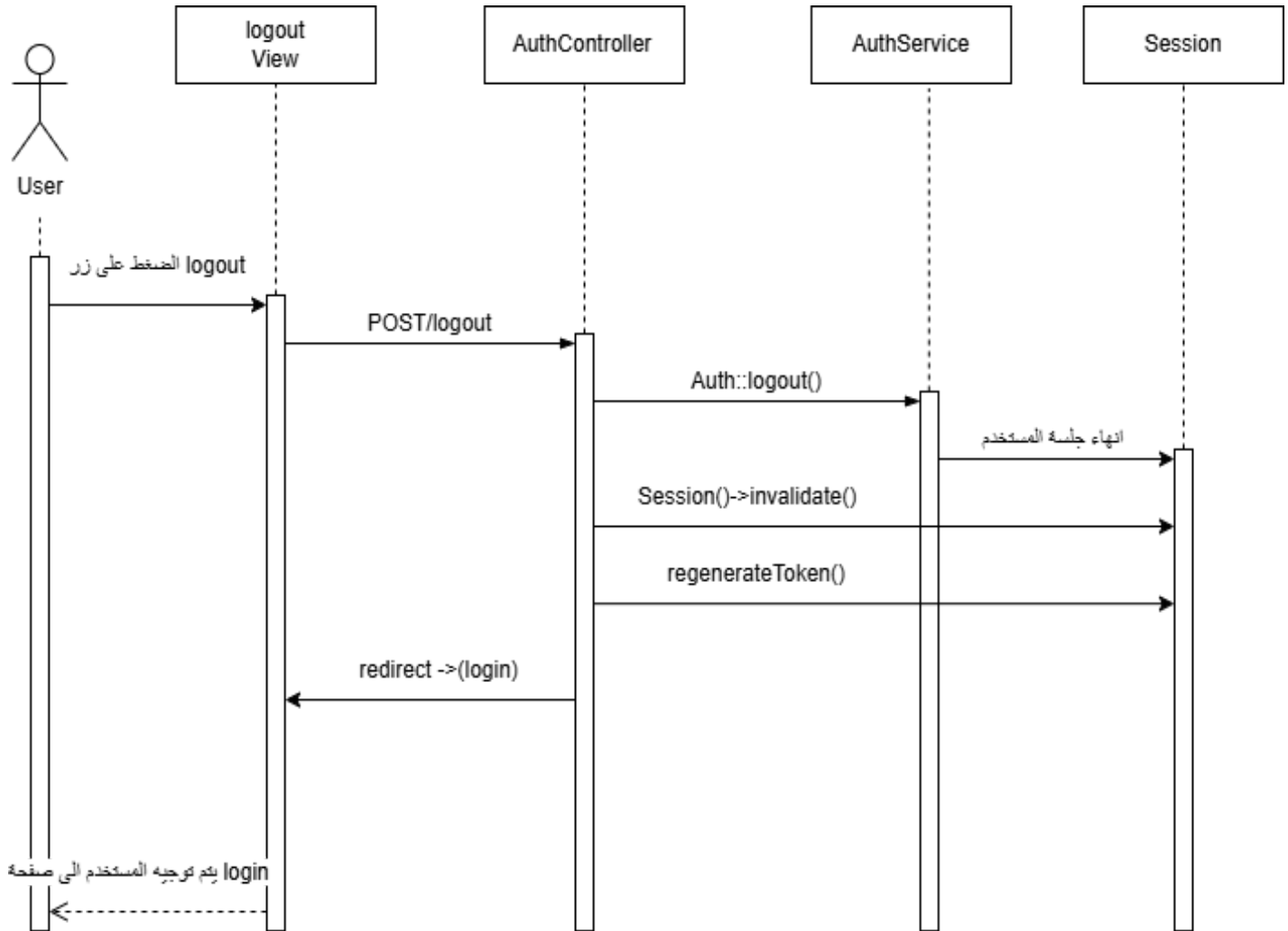


## 7. مخططات التسلسل : Sequence Diagrams

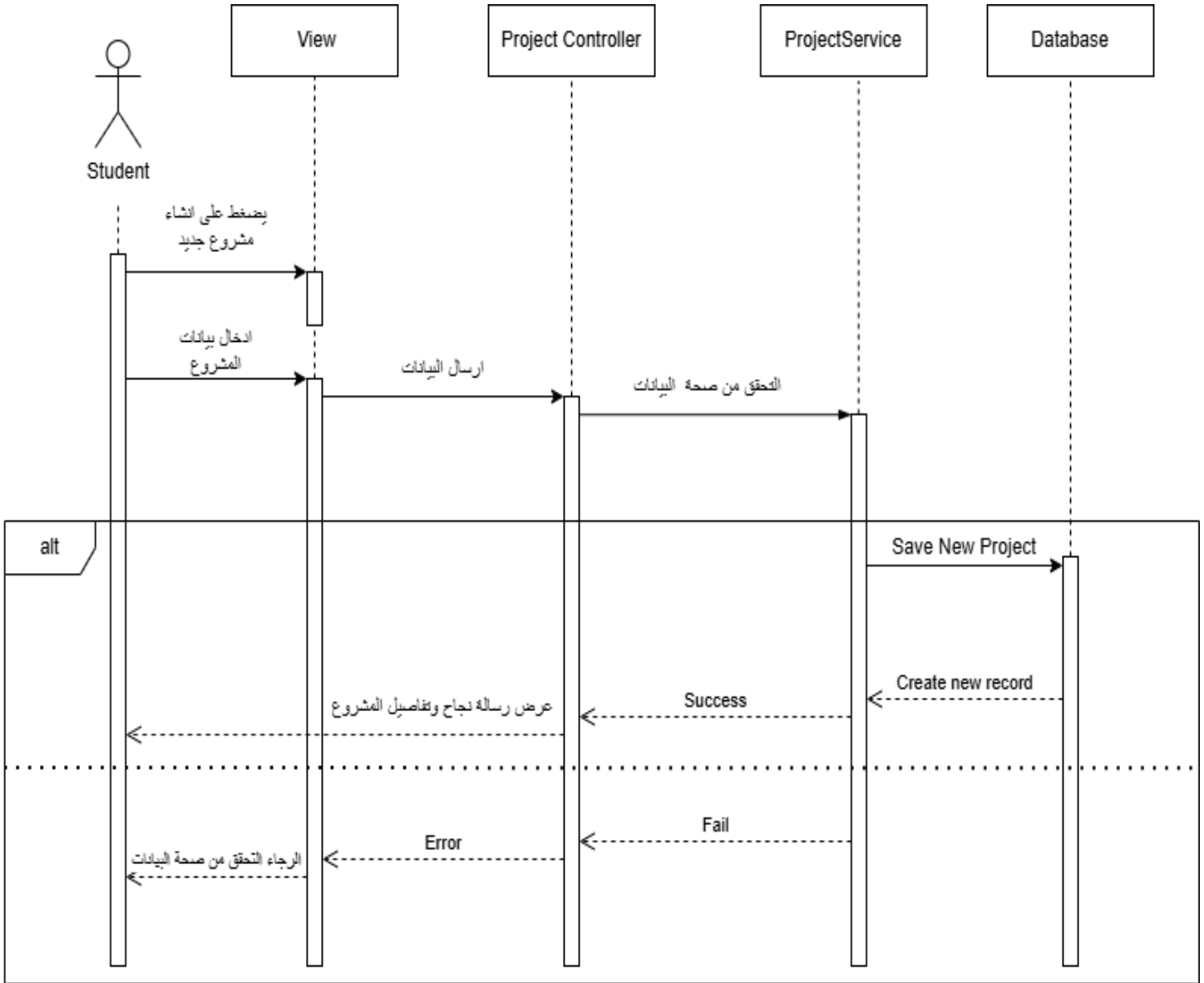
### ● تسجيل الدخول (UC-01) : مخطط التسلسل- Sequence diagram



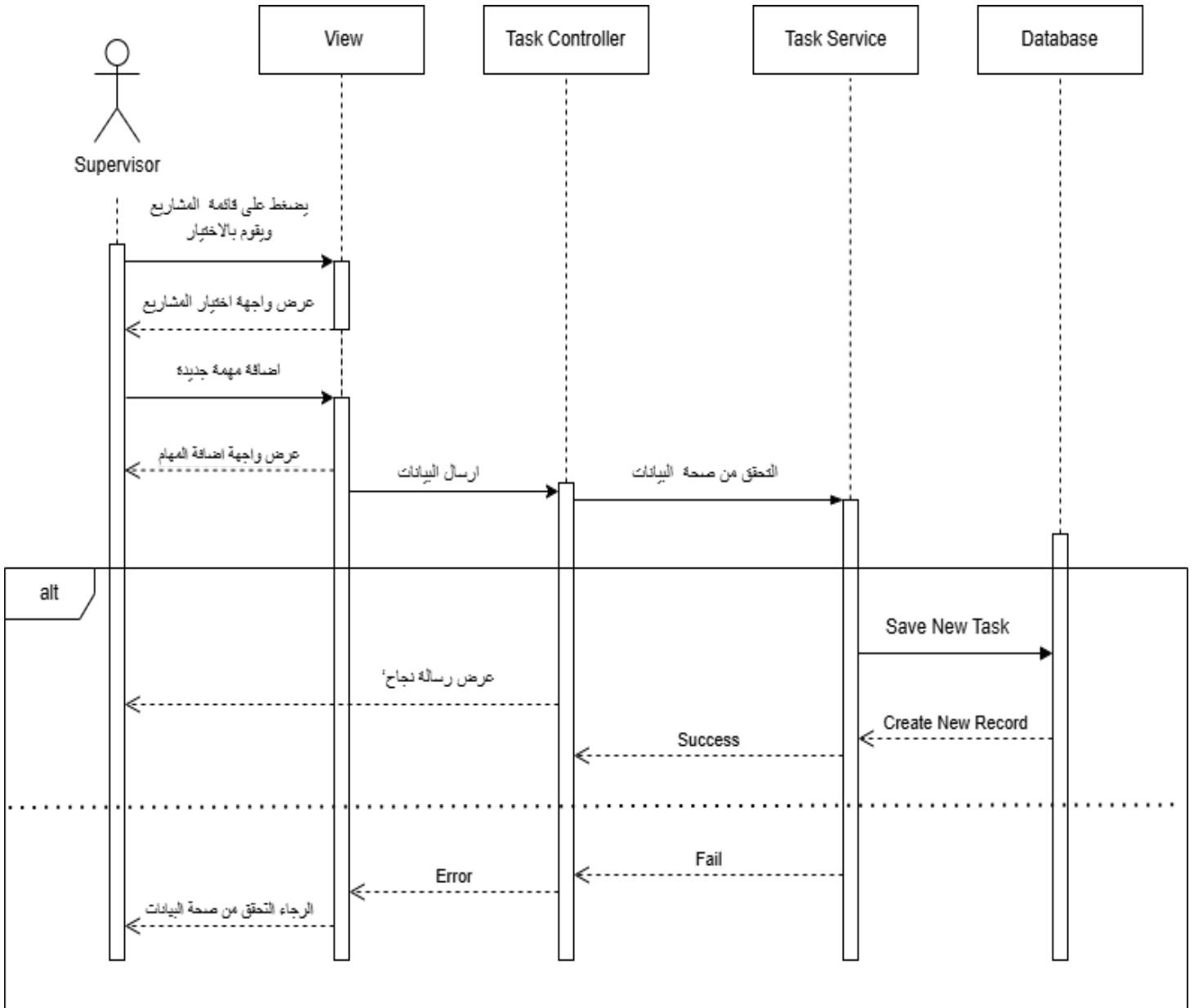
## ● تسجيل الخروج (UC-02):



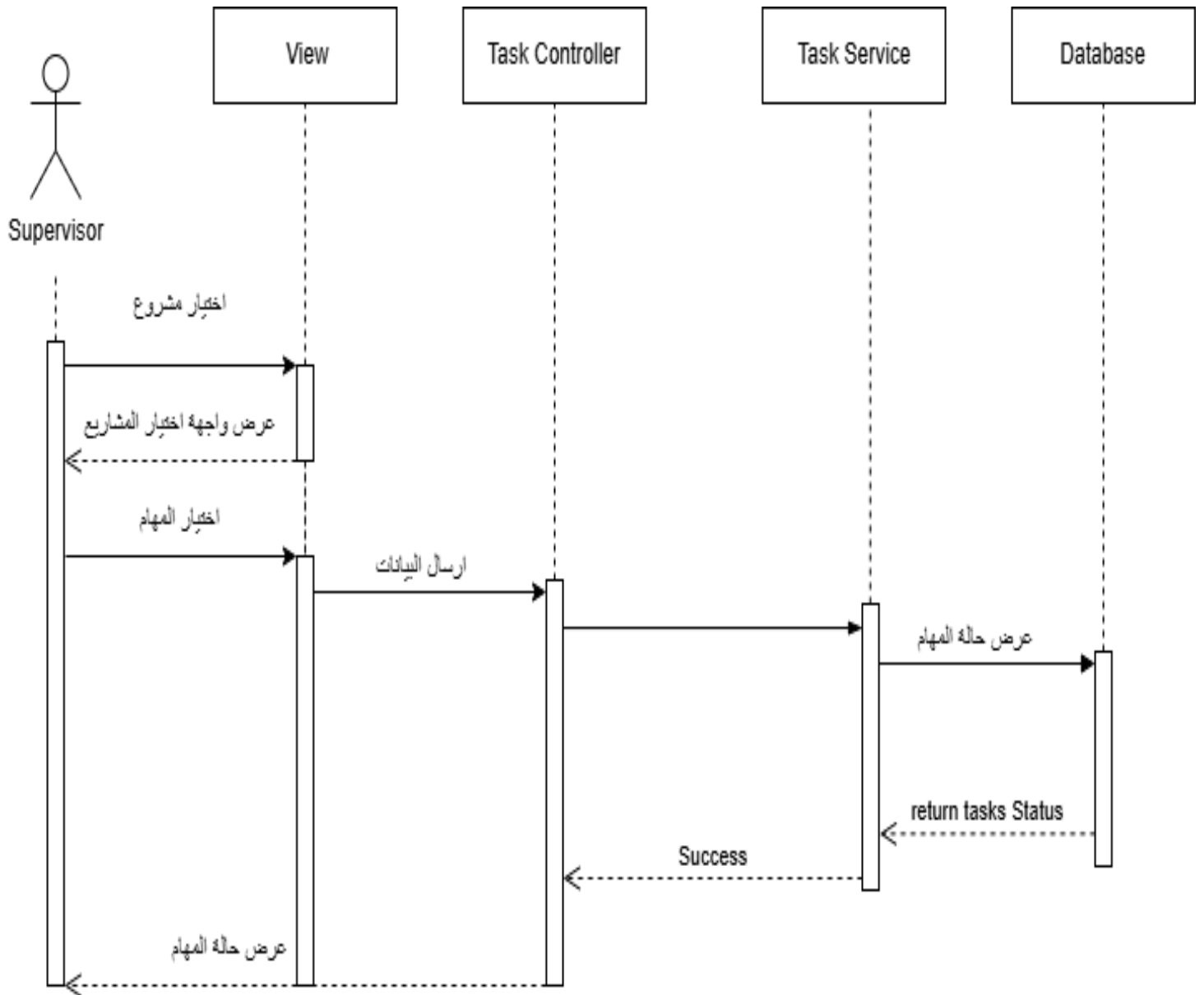
## ● إنشاء مشروع (UC-03) :



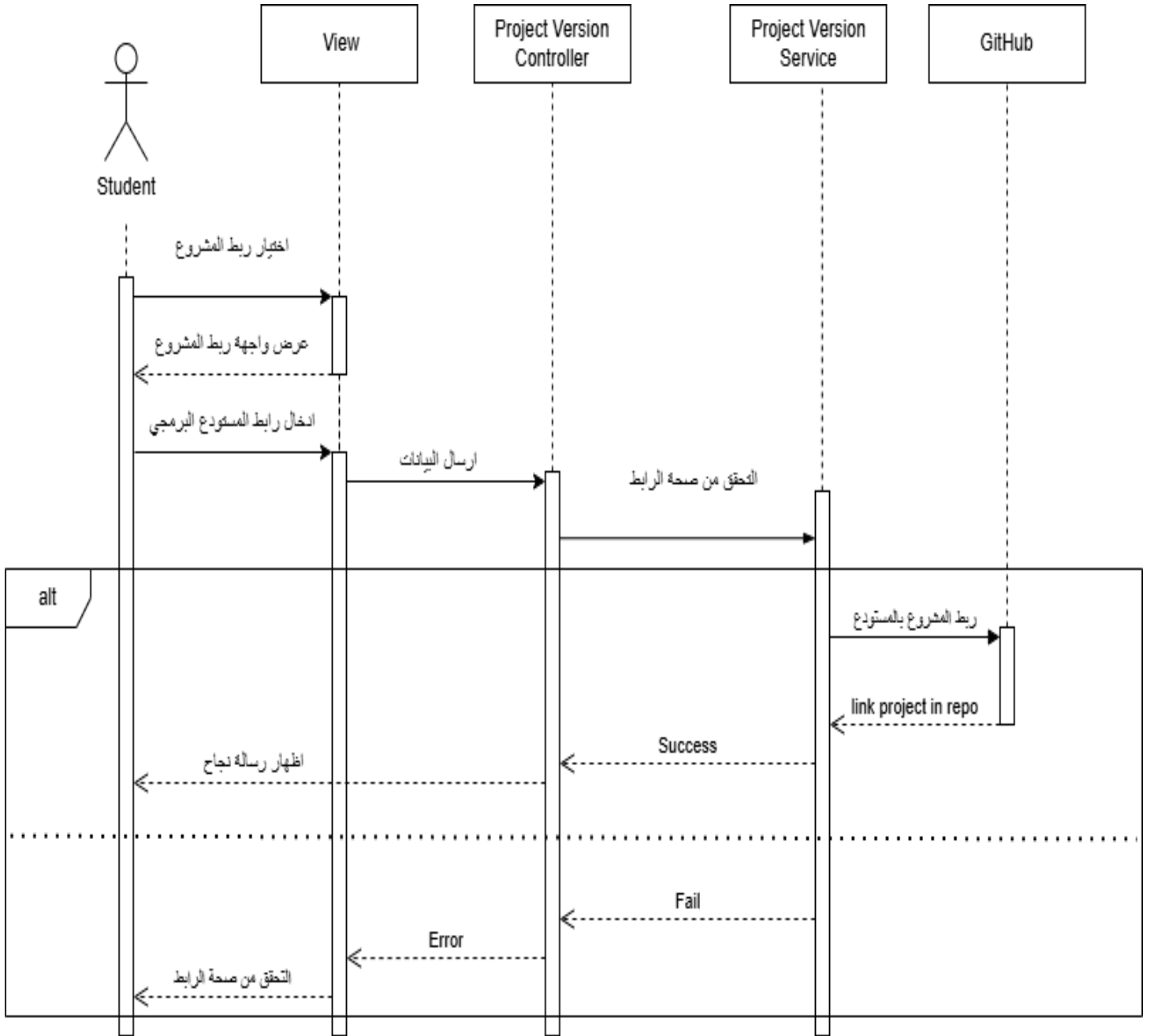
## ● إضافة مهمة (UC-04) :



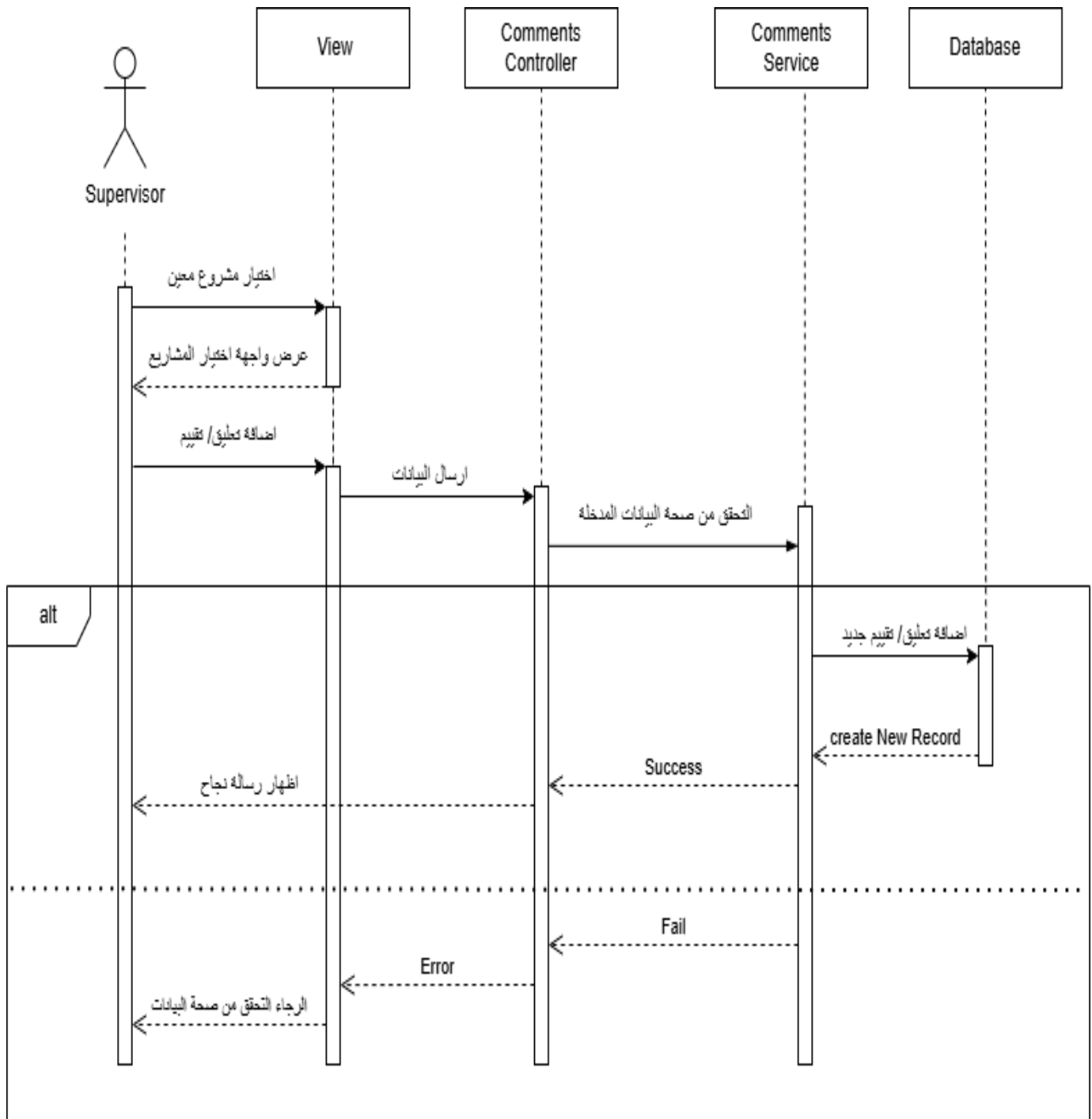
• تتبع التقدم (UC-05) :



## ● التحكم بالإصدارات (UC-06) :

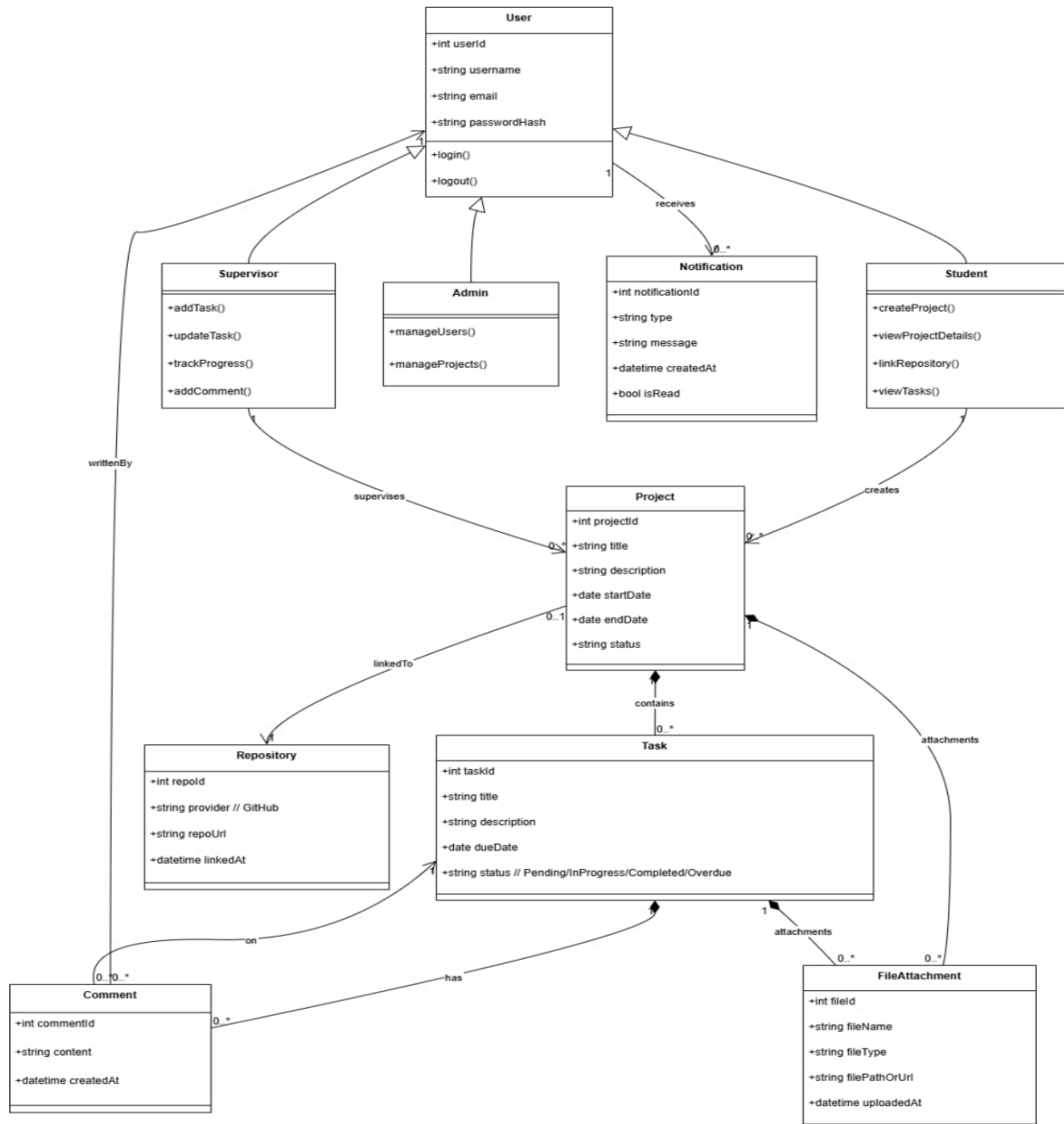


## ● التعليقات والتقييم (UC-07):



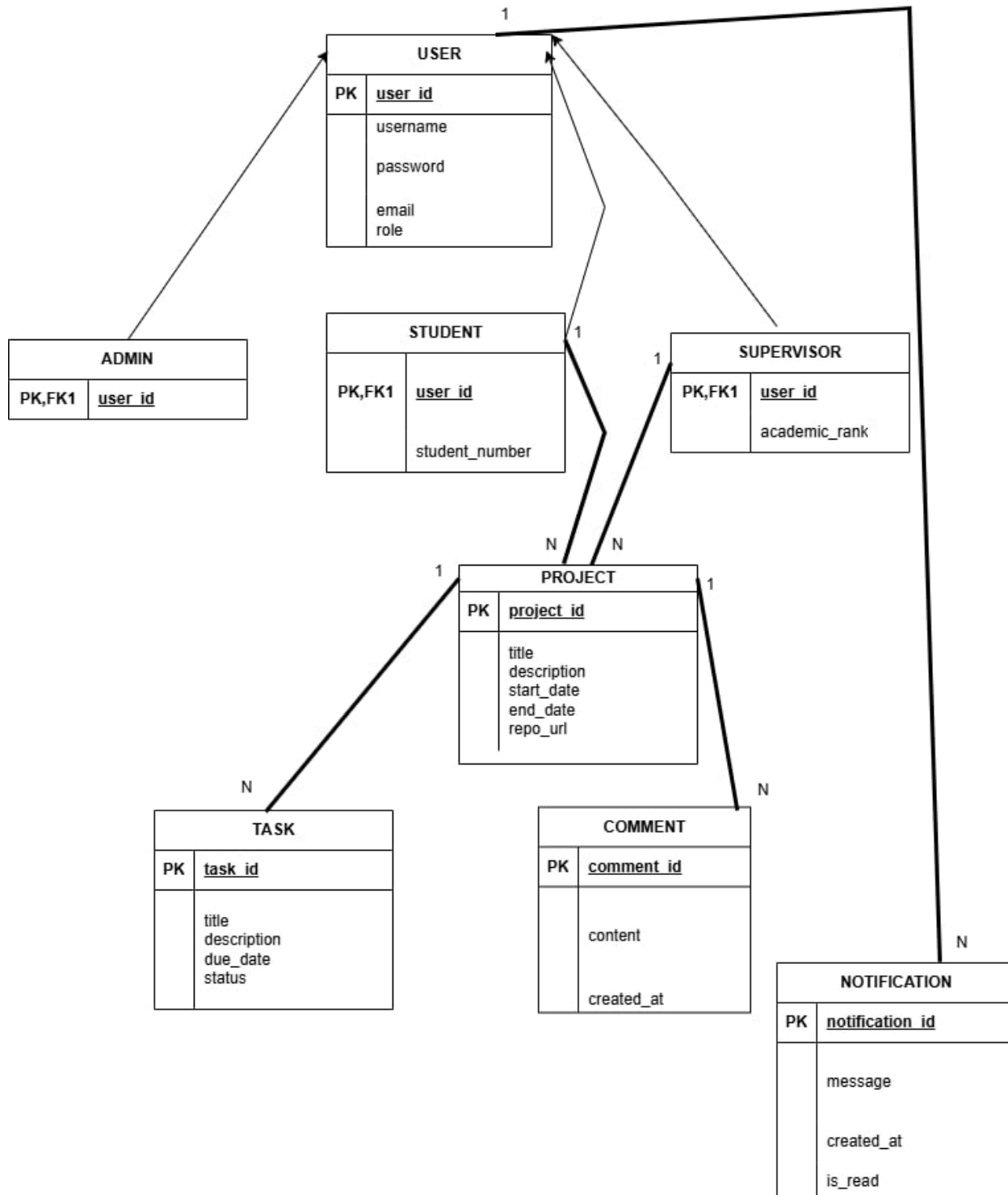
## 8. مخطط الفئات \_ Analysis Class Diagram :

### High level class Diagram





## 8.1 مخطط الكيان العلائقي ER diagram :



## 9. حالات الاختبار الأولية : Initial test cases


Test Case Scenario		Account management	
Test Case Id	Test Case Title	Test Steps	Expected result
TC-01	تسجيل الدخول للمشرف والطالب والمسؤول	1. فتح الموقع 2. الضغط على تسجيل الدخول 3. إدخال اسم المستخدم وكلمة المرور الصحيحة 4. الضغط على زر " تسجيل الدخول"	يتم الدخول للحساب الشخصي لكل منهم
TC-02	تسجيل الدخول ببريد إلكتروني خاطئ	1. فتح الموقع 2. الضغط على تسجيل الدخول 3. إدخال بريد إلكتروني خاطئ 4. الضغط على زر " تسجيل الدخول"	إظهار رسالة " البريد الإلكتروني غير صحيح "
TC-03	تسجيل الدخول بكلمة مرور غير صحيحة	1. فتح الموقع 2. الضغط على تسجيل الدخول 3. إدخال بريد إلكتروني خاطئ 4. الضغط على تسجيل الدخول	إظهار رسالة " كلمة المرور غير صحيحة "


TC-05	ترك أحد حقول التسجيل فارغة	1. فتح الموقع 2. الضغط على تسجيل الدخول 3. ترك أحد الحقول فارغاً 4. الضغط على تسجيل الدخول	تعطيل زر تسجيل الدخول لغاية تعبئة الحقول بالمطلوب
TC-06	تسجيل الخروج للمشرف والطالب	1. فتح الموقع 2. تسجيل الدخول إلى الحساب 3. الضغط على تسجيل الخروج	تسجيل الخروج بنجاح وإعادة توجيه المستخدم إلى واجهة تسجيل الدخول

Test Case Scenario		Project Management	
Test Case Id	Test Case Title	Test Steps	Expected result
TC-07	إنشاء مشروع جديد من قبل الطالب أو المسؤول	1. فتح الموقع 2. الضغط على نافذة المشاريع 3. إدخال عنوان ووصف المشروع 4. الضغط على "إنشاء"	تظهر رسالة نجاح " تم إنشاء المشروع " وعرض المشروع بقائمة المشاريع
TC-08	إنشاء مشروع مع ترك أحد الحقول فارغة	1. فتح الموقع 2. الضغط على نافذة المشاريع 3. ترك أحد الحقول فارغاً 4. الضغط على "إنشاء"	تظهر رسالة خطأ " الرجاء ملء هذا الحقل "

TC-09	تعديل بيانات المشروع	1. فتح الموقع 2. الضغط على نافذة المشاريع 3. فتح المشروع من قائمة المشاريع 4. من إعدادات المشروع 5. الضغط على "تعديل"	تعديل بيانات المشروع بنجاح
TC-10	تعديل بيانات المشروع وترك أحد الحقول فارغة	1. فتح الموقع 2. الضغط على نافذة المشاريع 3. فتح المشروع من قائمة المشاريع 4. من إعدادات المشروع الضغط على "تعديل" 5. ترك بعض الحقول فارغة	ظهور رسالة "الرجاء ملء هذا الحقل"
TC-11	حذف المشروع	1. فتح الموقع 2. الضغط على نافذة المشاريع 3. اختيار المشروع 4. من إعدادات المشروع الضغط على "حذف"	يتم حذف المشروع بنجاح

Test Case Scenario		Task Management	
Test Case Id	Test Case Title	Test Steps	Expected result
TC-12	إضافة مهمة جديدة	1. الضغط على قائمة المشاريع الذهاب 2. لواجهة المهام 3. إضافة مهمة جديدة ووصف وتاريخ تسليم 4. الضغط على " إضافة "	إضافة مهمة جديدة بنجاح
TC-13	إضافة مهمة جديدة وترك أحد الحقول فارغاً	1. الضغط على قائمة المشاريع الذهاب 2. لواجهة المهام 3. ارك أحد الحقول فارغاً 4. الضغط على " إضافة "	إظهار رسالة " الرجاء ملء الحقول المطلوبة

Test Case Scenario		Invitations Management	
Test Case Id	Test Case Title	Test Steps	Expected result
TC-15	إضافة دعوة لأحد المشرفين	1. تسجيل الدخول كطالب 2. الضغط على قائمة المشاريع 3. الذهاب لواجهة الدعوات 4. اختيار المشرف أو الطالب 5. الضغط على دعوة/ إرسال	إرسال الدعوة بنجاح وظهور رسالة تأكيد
TC-16	قبول دعوة أحد الطلاب من قبل المشرف	1. تسجيل الدخول كمشرف 2. الضغط على قائمة الدعوات 3. الضغط على إشارة  لقبول الدعوة	قبول الدعوة بنجاح وإظهار رسالة "أصبحت المشرف لهذا المشروع"

Test Case Scenario		Comments Management	
Test Case Id	Test Case Title	Test Steps	Expected result
TC-17	إضافة تعليق	1. الضغط على قائمة المشاريع 2. الذهاب لواجهة التعليقات 3. إضافة تعليق 4. الضغط على زر " إضافة "	اضافة التعليق بنجاح وظهوره في قائمة التعليقات
TC-18	حذف تعليق	1. الضغط على قائمة المشاريع 2. الذهاب لواجهة التعليقات 3. الضغط على  بجانب التعليق	حذف التعليق بنجاح

## 10. مصفوفة تتبع المتطلبات الأولية \_ Initial Requirement :Traceability Matrix ( RTM

ID	Title	Associated Dependenc y	SRS Section	Design Section	Code	Integration Test	Accept Test
RE-Fr-AM 1.1	يجب أن يسمح النظام للمستخدم بتسجيل الدخول من خلال اسم المستخدم وكلمة المرور	يجب أن يكون اسم المستخدم وكلمة المرور متطابقين مع البيانات الموجودة في قاعدة البيانات	إدارة الحسابات 1.1				
RE-Fr-AM 1.2	يجب أن يسمح النظام للمستخدم بتسجيل الخروج من حسابه	يجب أن يكون المستخدم مسجل الدخول إلى حسابه	إدارة الحسابات 1.2				
RE-Fr-PM 2.1	يجب أن يسمح النظام للطالب بإضافة مشروع جديد	يجب أن تكون جميع الحقول (عنوان المشروع , ووصف المشروع )	إدارة المشاريع 2.1				
RE-Fr-PM 2.2	يجب أن يسمح النظام	يجب أن يكون هناك ترتيب معين للمشاريع	إدارة المشاريع 2.2				



	للمشرف بعرض تفاصيل المشاريع الخاص بالطلاب						
RE-Fr- PM 2.3	يجب للنظام أن يسمح للمسؤول بإنشاء مشروع جديد للطالب	يجب أن يكون الطالب مستوفياً لشروط إنشاء مشروع جديد ) أن تكون بياناته مكتملة (	إدارة المشاريع 2.3				
RE-Fr- PM 2.4	يجب للنظام أن يسمح للمسؤول بتعديل معلومات المشروع	يجب أن تكون التعديلات متوافقة مع سياسات النظام	إدارة المشاريع 2.4				
RE-Fr- PM 2.4	يجب للنظام أن يسمح للمسؤول بحذف مشروع معين	يجب إرسال إشعار للطالب قبل الحذف بفترة كافية	إدارة المشاريع 2.4				
RE-Fr- PM 2.5	السماح للمسؤول بعرض كافة المشاريع	يجب أن يكون هناك نظام فلتره وبحث	إدارة المشاريع 2.5				
RE-Fr- TM 3.1	يجب أن يسمح النظام	يجب أن تكون جميع معلومات المهمة مكتملة	إدارة المهام 3.1				

	للمسؤول بإضافة مهمة معينة وتاريخ تسليم						
RE-Fr- TM 3.2	يتيح النظام للمشرف بعرض جميع المهام المتعلقة بمشروع معين	يجب أن يكون هناك نظام تصفية وطريقة عرض معينة	إدارة المهام 3.2				
RE-Fr- TM 3.3	يجب أن يسمح النظام للطالب بعرض تفاصيل المهمة المسندة إليه	يجب أن تكون تفاصيل المهمة متاحة للطلاب	إدارة المهام 3.3				
RE-Fr- PT 4.1	يجب أن يسمح النظام للمشرف برؤية حالة المهام	يجب أن تكون حالة المهام متاحة لكي يراها المشرف	تتبع التقدم 4.1				
RE-Fr- VC 5.1	يجب أن يسمح النظام للطالب بوضع رابط المستودع الخاص به	يجب أن يكون رابط المستودع صحيح ومطابق للمستودع الخاص به	التحكم بالإصدارات 5.1				

RE-Fr-CM 6.1	يجب على النظام أن يسمح للطالب والمشرف بإضافة تعليقات	يجب أن تكون التعليقات ضمن السياسات المطلوبة	التعليقات 6.1				
RE-NF-US 1.1	يجب أن يوفر النظام واجهة سهلة وبسيطة للمستخدم	يجب أن تكون أزرار وواجهات الصفحة واضحة للمستخدم	سهولة الاستخدام 1.1				
RE-NF-AV 1.2	يجب أن يحافظ النظام على جاهزية التشغيل	يجب توافر نسخ احتياطية تلقائية لقواعد البيانات	التوافر 1.2				
RE-NF-SEC 1.3	يجب أن يؤمن النظام المصادقة الأساسية وتشفير كلمات المرور	يجب استخدام خوارزمية تشفير قوية	الأمان 1.3				
RE-NF-Sca 1.4	يجب أن يكون النظام قابلاً لإعادة التوسع لدعم عدد متزايد من المستخدمين	يجب استخدام تقنيات load balancing عند زيادة عدد المستخدمين	قابلية التوسع 1.4				

# Chapter 5

## System Design

