



---

## تقرير مشروع MINI SCHOOL MANAGEMENT SYSTEM

---



تقديم الطالبات : رؤى الحلبي - شيماء مصهور - لين صطوف - اية عبد اللطيف

بإشراف : أ. محمود البيوش

مادة : إدارة المشاريع البرمجية

## المقدمة:

حضره الأستاذ الدكتور المحترم،

يسـ فريق العمل أن يقدم لسيادتكم هذا التقرير الخاص بمشروع **Mini School Management System (Mini SMS)** وهو نظام إدارة مدرسي مصغر لكنه متكامل، تم تطويره بهدف تطبيق المفاهيم النظرية في قواعد البيانات، البرمجة، وإدارة المشاريع البرمجية ضمن بيئـ عملية تحاكي الأنظمة الواقعية.

يركـ المشروع على بناء نظام قادر على إدارة الطلاب، المعلمين، المواد، الصحفـ، الحضور، والدرجات، مع تقسيـ واضح للمهام بين أعضـ الفريق، واعتماد أدوات احترافية مثل Git و Jira للتنظيم سير العمل وضمان التكامل بين الوحدـ المختلفة.

---

## الفصل الأول: أهداف المشروع:

يهدف المشروع إلى:

1. تصميم نظام إدارة مدرسي مصغر لكنه واقعي.
2. تطبيق عمليـ CRUD على مختلف الكـيانـات.
3. تصميم قاعدة بيانات علائقـة متـرابـطة.
4. تقسيـ العمل إلى وحدـات مستـقلـة (Modules).
5. استخدام Git لإدارة الإصدـارات.
6. استخدام Jira لإدارة المـهام.
7. دمج جميع الوحدـات في نظام واحد مـتكـامل.

---

## الفصل الثاني: الوحدـات البرمجـية (System Modules)

### أولاً : Student Management Module

تحـتصـ هذه الوحدـة بإدارة بيانات الطـلاب داخل النـظام.

المـهام المنفذـة:

- إضـافة طـالـب.
- تعـديل بـيانـات طـالـب.
- حـذـف طـالـب.
- عـرض قائـمة الطـلـاب.

- ربط الطالب بصف محمد.

قاعدة البيانات:

- جدول students

- علاقة classes مع جدول One-to-Many

إدارة الكود:

- Branch: feature/student-module

- Commits مستقلة

- توثيق المهام في Jira

---

### ثانياً : Teacher & Subject Module

تعنى هذه الوحدة بإدارة المعلمين والمواد والعلاقة بينهما.

المهام المنفذة:

- إضافة معلم.

- إضافة مادة.

- ربط المعلم بالمادة.

- عرض قائمة المعلمين والمواد.

قاعدة البيانات:

- teachers

- subjects

- جدول teacher\_subjects ( وسيطteacherSubjects )

العلاقة:

Many-to-Many بين المعلمين والمواد.

---

### ثالثاً : Class, Attendance & Grades Module

تغطي هذه الوحدة إدارة الصفوف والحضور والدرجات.

إدارة الصفوف:

إنشاء صف •

تعديل •

حذف •

عرض •

**ادارة الحضور:**

تسجيل حضور •

تعديل •

حذف •

عرض •

**ادارة الدرجات:**

إدخال درجة •

تعديل •

حذف •

عرض •

**التقارير:**

تقرير الحضور •

تقرير الدرجات •

## رابعاً : Integration, UI & Documentation

تتضمن:

دمج جميع الوحدات. •

إعداد الواجهة. •

كتابة README النهائي. •

إعداد ER Diagram. •

توثيق Git و Jira. •

## **الفصل الثالث: تصميم وتوثيق قاعدة البيانات:**

### **3.1 نظرة عامة:**

تم تصميم قاعدة البيانات وفق النموذج العلائقي (Relational Model) ، مع تطبيق مبادئ التطبيع (Normalization) لتنقيل التكرار وضمان تكامل البيانات.

---

### **3.2 الجداول الرئيسية:**

#### **1. Classes جدول:**

class\_id (Primary Key) •

class\_name •

#### **2. Students جدول:**

student\_id (Primary Key) •

first\_name •

last\_name •

class\_id (Foreign Key) •

العلاقة:

صف واحد يحتوي على عدة طلاب. (One-to-Many).

---

#### **3. Teachers جدول:**

teacher\_id (Primary Key) •

first\_name •

last\_name •

email •

created\_at •

---

#### **Subjects جدول4.**

subject\_id (Primary Key) •  
 subject\_name •  
 created\_at •

---

#### **Teacher\_Subjects جدول5.**

id (Primary Key) •  
 teacher\_id (Foreign Key) •  
 subject\_id (Foreign Key) •

يمثل علاقة Many-to-Many بين teachers و subjects.

#### **Attendance جدول6.**

attendance\_id (Primary Key) •  
 student\_id (Foreign Key) •  
 date •  
 status •

العلاقة:  
 طالب واحد يمكن أن يمتلك عدة سجلات حضور.

#### **Grades جدول7.**

grade\_id (Primary Key) •  
 student\_id (Foreign Key) •  
 subject\_id (Foreign Key) •  
 grade •

العلاقة:  
 الطالب يمتلك عدة درجات.  
 المادة تحتوي على عدة درجات.

---

### العلاقات بين الكيانات: 3.3

Classes → Students .1

One-to-Many

Students → Attendance .2

One-to-Many

Students → Grades .3

One-to-Many

Teachers ↔ Subjects .4

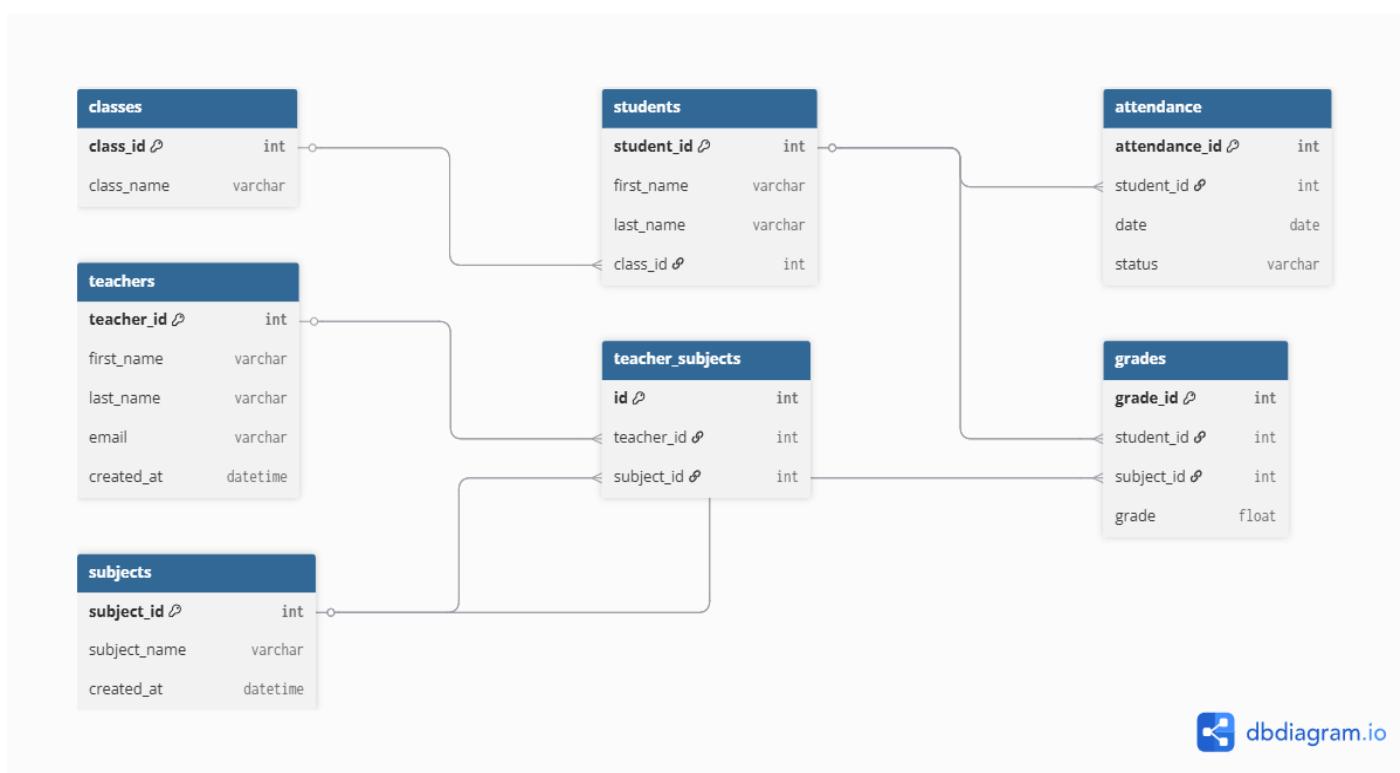
Teacher\_Subjects عبّر Many-to-Many

Subjects → Grades .5

One-to-Many

---

### الوصف التحليلي (ER Diagram) 3.4



الشكل (1): مخطط الكيانات وال العلاقات لقاعدة بيانات نظام إدارة المدرسة المصغر.

يعكس المخطط الكياني العلقي ما يلي :

- كل كيان يحتوي على مفتاح أساسي يميزه .
- استخدام المفاتيح الأجنبية لضمان تكامل العلاقات.
- فصل العلاقة متعددة القيم في جدول وسيط.
- تطبيق مبدأ التطبيع لتجنب التكرار.

يمكن تمثيل العلاقات كما يلي:

Classes (1) —— (N) Students  
Students (1) —— (N) Attendance  
Students (1) —— (N) Grades  
Teachers (N) —— (N) Subjects  
Subjects (1) —— (N) Grades

---

### 3.5 قيود التكامل : (Integrity Constraints)

- منع إدخال طالب بدون صف موجود.
- منع إدخال درجة لطالب غير موجود.
- منع ربط معلم بمادة غير معرفة.
- استخدام Primary Key لمنع التكرار.

---

### الفصل الرابع: إدارة الإصدارات باستخدام Git :

تم اعتماد منهجية تطوير تعتمد على:

1. إنشاء فرع مستقل لكل وحدة.

تم استخدام الفروع البرمجية لتقسيم العمل وتتجنب تعارض التعديلات، كما هو موضح في الشكل.(2)

The screenshot shows the GitHub 'Branches' page for the repository 'shaimaamadhour395-a11y / School-Management-System'. The page is divided into sections: 'Default', 'Your branches', and 'Active branches'. Under 'Default', there is one branch named 'main'. Under 'Your branches', there are two branches: 'feature/class-attendance-grades' and 'feature/integration-ui-docs'. Under 'Active branches', there are also two branches: 'feature/class-attendance-grades' and 'feature/integration-ui-docs'. Each branch entry includes columns for 'Updated', 'Check status', 'Behind | Ahead', and 'Pull request'.

- الشكل (2) إنشاء وإدارة الفروع البرمجية في GitHub.
2. تنفيذ التطوير داخل الفرع.
  3. رفع التعديلات عبر Commits.
  4. إنشاء Pull Requests.

بعد الانتهاء من تطوير كل وحدة برمجية ضمن فرع مستقل، تم إنشاء Pull Requests لمراجعة التعديلات قبل دمجها في الفرع الرئيسي. يوضح الشكل (3) قائمة طلبات الدمج التي تم اعتمادها وإغلاقها بنجاح.

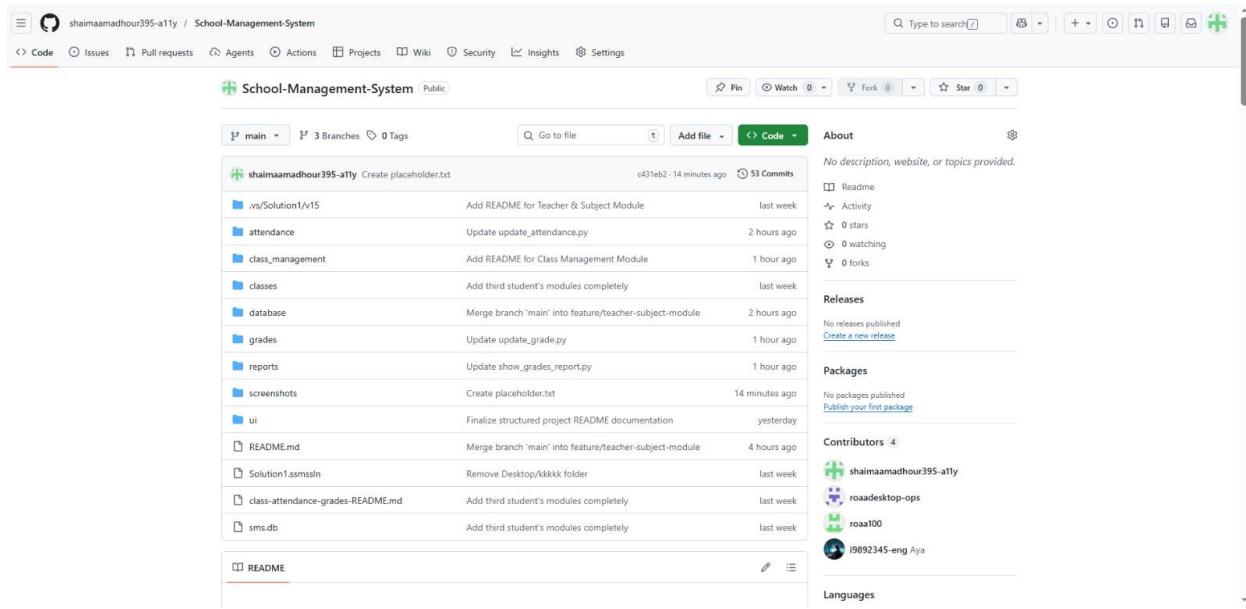
The screenshot shows the GitHub 'Pull requests' page for the repository 'shaimaamadhour395-a11y / School-Management-System'. A modal window titled 'Label issues and pull requests for new contributors' is open, providing instructions for first-time contributors. Below the modal, there is a search bar and filter options. The main list displays several merged pull requests, each with a title, author, and merge status. The pull requests listed are:

- KAN-3: Implement Class Management Module (#5 by shaimaamadhour395-a11y was merged 1 hour ago)
- KAN-2: Implement Teacher & Subject Module (#4 by shaimaamadhour395-a11y was merged 2 hours ago)
- KAN-1: Implement Student Module (#3 by shaimaamadhour395-a11y was merged 2 hours ago)
- Feature/class attendance grades (#2 by i8992345-eng was merged 4 hours ago)
- Feature/teacher subject module (#1 by leen-sat was merged 3 hours ago - Approved)

الشكل (3) عرض Pull Requests المغلقة التي تم دمجها بنجاح ضمن الفرع الرئيسي للمشروع.

## 5. دمج التعديلات في الفرع الرئيسي.

بعد تنفيذ عمليات Commit و Push بنجاح، ظهرت ملفات المشروع ضمن الفرع الرئيسي (main) في المستودع البعيد على GitHub كما هو موضح في الشكل.(4)



الشكل(4): عرض ملفات المشروع داخل الفرع الرئيسي (main) على GitHub

يعكس ذلك تطبيق منهجية تطوير احترافية قائمة على الفصل بين المهام ومنع التعارضات.

## الفصل الخامس: إدارة المشروع باستخدام Jira :

تم استخدام Kanban Board لتقسيم العمل إلى:

To Do •

In Progress •

Done •

وتم تخصيص كل مهمة لعضو محدد، مما ساهم في تنظيم العمل وضمان تتبع تقدم المشروع.

تم استخدام لوحة Kanban ضمن منصة Jira لتنظيم المهام وتحديد حالة كل مهمة خلال مراحل التنفيذ المختلفة، كما هو موضح في الشكل.(5).

Work	Assignee	Reporter	Priority	Status	Resolution	Created	Updated	Due date
KAN-4 Integration, UI & Documentation	Shaimaa Madhour	Shaimaa Madhour	Medium	TO DO	Unresolved	Feb 17, 2026, 1:13 AM	Feb 17, 2026, 1:14 AM	None
KAN-3 Class, Attendance & Grades Module	Shaimaa Madhour	Shaimaa Madhour	Medium	TO DO	Unresolved	Feb 17, 2026, 1:11 AM	Feb 17, 2026, 1:12 AM	None
KAN-2 Teacher & Subject Module	Shaimaa Madhour	Shaimaa Madhour	Medium	TO DO	Unresolved	Feb 17, 2026, 1:07 AM	Feb 17, 2026, 1:09 AM	None
KAN-1 Student Management Module	Shaimaa Madhour	Shaimaa Madhour	Medium	IN PROGRESS	Unresolved	Feb 17, 2026, 12:49 AM	Feb 17, 2026, 1:06 AM	None

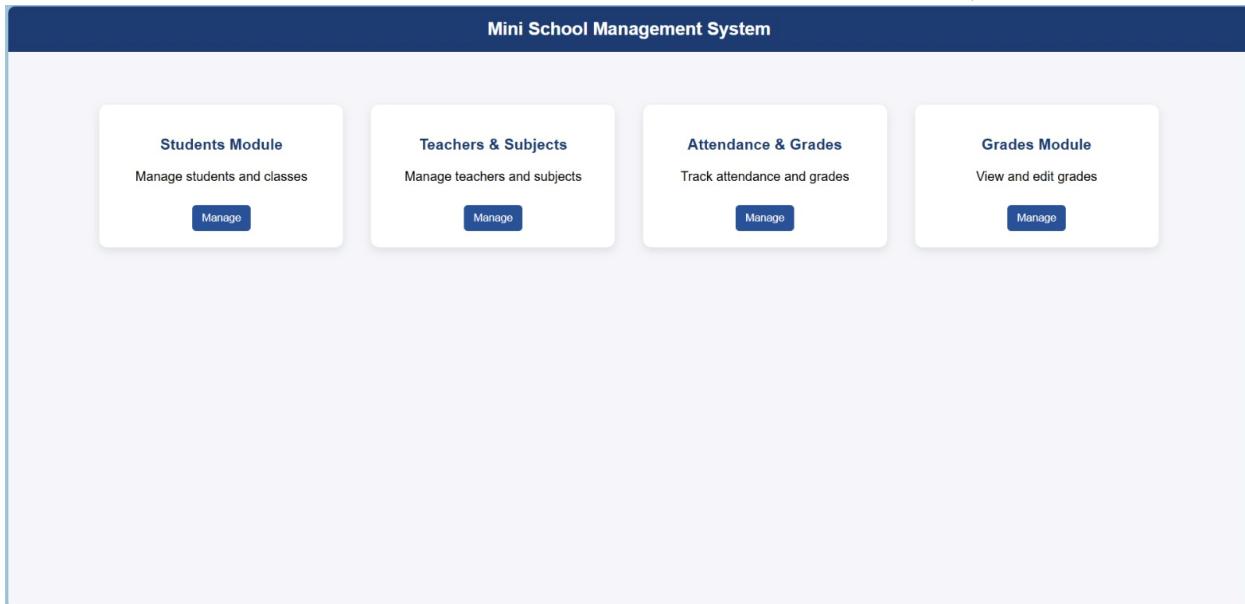
الشكل (5) لوحة متابعة سير العمل (Kanban Board) المستخدمة لتنظيم مهام المشروع وتتبع تقدم تنفيذه.

## الفصل السادس: طريقة تشغيل المشروع:

1. تشغيل نظام إدارة قواعد البيانات.
2. تنفيذ ملفات SQL بالترتيب.
3. التأكد من إنشاء الجداول.
4. تشغيل سكريبتات Python الخاصة بكل وحدة.

عرض وتنفيذ النظام:

1. الواجهة الرئيسية للنظام:



الشكل (6): الواجهة الرئيسية لنظام Mini School Management System

2. واجهة إدارة الطلاب:

Students Management			
Add Student			
ID	Name	Class	Actions
1	Ahmad	1	<button>Edit</button> <button>Delete</button>
2	Sara	1	<button>Edit</button> <button>Delete</button>

الشكل (7): واجهة إدارة الطلاب

3. واجهة إدارة المعلمين والمواد:

The screenshot shows a web-based application titled "Teachers & Subjects Management". At the top, there are two buttons: "Add Teacher" and "Add Subject". Below these are two sections: "Teachers" and "Subjects".

**Teachers**

ID	Name	Actions
1	John Doe	<a href="#">Edit</a> <a href="#">Delete</a>
2	Jane Smith	<a href="#">Edit</a> <a href="#">Delete</a>

**Subjects**

ID	Name	Actions
1	Mathematics	<a href="#">Edit</a> <a href="#">Delete</a>
2	Physics	<a href="#">Edit</a> <a href="#">Delete</a>

الشكل (8): واجهة إدارة المعلمين والمواد

4. واجهة سجلات الحضور:

The screenshot shows a web-based application titled "Attendance Records". At the top, there is a button "Add Attendance". Below it is a table with one record.

ID	Student	Status	Date	Actions
1	Ahmad	Present	2026-02-17	<a href="#">Edit</a> <a href="#">Delete</a>

الشكل (9): واجهة عرض سجلات الحضور للطلاب

5. واجهة إدارة الدرجات:

The screenshot shows a web-based application titled "Grades Management". At the top, there is a table with four columns: "Student", "Subject", "Grade", and "Actions". The table contains four rows of data. In the "Actions" column, each row has two buttons: "Edit" and "Delete". Below the table, there is a section titled "Add Grade" with three dropdown menus: "Student" (set to "Ahmad"), "Subject" (set to "Mathematics"), and "Grade" (an empty input field). To the right of the "Grade" field is a blue "Add" button. At the bottom of this section is a link "Back to Home". The browser's address bar at the top right shows the URL "127.0.0.1:5000/grades".

الشكل (10): واجهة إدارة درجات الطلاب

6. واجهة تعديل درجة:

The screenshot shows a web-based application titled "Edit Grade". It features a single input field labeled "Grade:" containing the value "88.0" and a blue "Update" button to its right. Below the input field is a link "Back to Grades". The browser's address bar at the top right shows the URL "127.0.0.1:5000/edit\_grade/4".

الشكل (11): واجهة تعديل درجة طالب

## **الفصل السابع: التقييم والاستنتاجات:**

يُظهر المشروع تكاملاً واضحاً بين:

- تصميم قاعدة البيانات
- تنفيذ العمليات البرمجية
- إدارة العلاقات بين الجداول
- استخدام أدوات إدارة المشاريع
- العمل الجماعي المنظم

ورغم أن النظام مصغر، إلا أنه يمثل نموذجاً عملياً يمكن تطويره ليصبح نظاماً متاماً على مستوى أوسع.

---

## **الخاتمة:**

حضرة الأستاذ الدكتور،

يمثل هذا المشروع تجربة عملية متكاملة في بناء نظام معلومات مدرسي باستخدام أدوات وتقنيات حديثة، مع الالتزام بمبادئ التصميم السليم لقواعد البيانات، وتنظيم العمل الجماعي، وإدارة الإصدارات باحترافية.

نأمل أن يكون العمل قد حقق الأهداف التعليمية المطلوبة، وأن يكون على المستوى الأكاديمي والتقني الذي تطمحون إليه.