



تقرير مشروع MINI SCHOOL MANAGEMENT SYSTEM



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

بإشراف : أ. محمود البيوش

مادة : إدارة المشاريع البرمجية

المقدمة:

حضرة الأستاذ الدكتور المحترم،

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

يركز المشروع على بناء نظام قادر على إدارة الطلاب، المعلمين، المواد، الصفوف، الحضور، والدرجات، مع تقسيم واضح للمهام بين أعضاء الفريق، واعتماد أدوات احترافية مثل Git و Jira للتنظيم سير العمل وضمان التكامل بين الوحدات المختلفة.

الفصل الأول: أهداف المشروع:

يهدف المشروع إلى:

1. تصميم نظام إدارة مدرسي مصغر لكنه واقعي.
2. تطبيق عمليات CRUD على مختلف الكيانات.
3. تصميم قاعدة بيانات علائقية مترابطة.
4. تقسيم العمل إلى وحدات مستقلة (Modules).
5. استخدام Git لإدارة الإصدارات.
6. استخدام Jira لإدارة المهام.
7. دمج جميع الوحدات في نظام واحد متكامل.

الفصل الثاني: الوحدات البرمجية (System Modules)

أولاً Student Management Module :

تختص هذه الوحدة بإدارة بيانات الطلاب داخل النظام.

المهام المنفذة:

- إضافة طالب.
- تعديل بيانات طالب.
- حذف طالب.
- عرض قائمة الطلاب.

- ربط الطالب بصف محدد.

قاعدة البيانات:

- جدول students
- علاقة One-to-Many مع جدول classes

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
- (teacher_subjects جدول وسيط)

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

إدارة الحضور:

- تسجيل حضور

- تعديل

- حذف

- عرض

إدارة 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) لتقليل التكرار وضمان تكامل البيانات.

3.2 الجداول الرئيسية:

1. جدول Classes

- class_id (Primary Key)
- class_name

2. جدول Students

- student_id (Primary Key)
- first_name
- last_name
- class_id (Foreign Key)

العلاقة:

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

3. جدول Teachers

- teacher_id (Primary Key)
 - first_name
 - last_name
 - email
 - created_at
-

4. جدول Subjects

- subject_id (Primary Key)
 - subject_name
 - created_at
-

5. جدول Teacher_Subjects

- id (Primary Key)
- teacher_id (Foreign Key)
- subject_id (Foreign Key)

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

6. جدول Attendance

- attendance_id (Primary Key)
- student_id (Foreign Key)
- date
- status

العلاقة:

طالب واحد يمكن أن يمتلك عدة سجلات حضور.

7. جدول Grades

- grade_id (Primary Key)
- student_id (Foreign Key)
- subject_id (Foreign Key)
- grade

العلاقة:

الطالب يمتلك عدة درجات.

المادة تحتوي على عدة درجات.

3.3 العلاقات بين الكيانات:

1. Classes → Students
One-to-Many
2. Students → Attendance
One-to-Many
3. Students → Grades
One-to-Many
4. Teachers ↔ Subjects
Teacher_Subjects Many-to-Many
5. Subjects → Grades
One-to-Many

3.4 الوصف التحليلي (ER Diagram):



الشكل (1): مخطط الكيانات والعلاقات لقاعدة بيانات نظام إدارة المدرسة المصغّر.

يعكس المخطط الكياني العلاقي ما يلي :

- كل كيان يحتوي على مفتاح أساسي يميّزه .
- 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) :

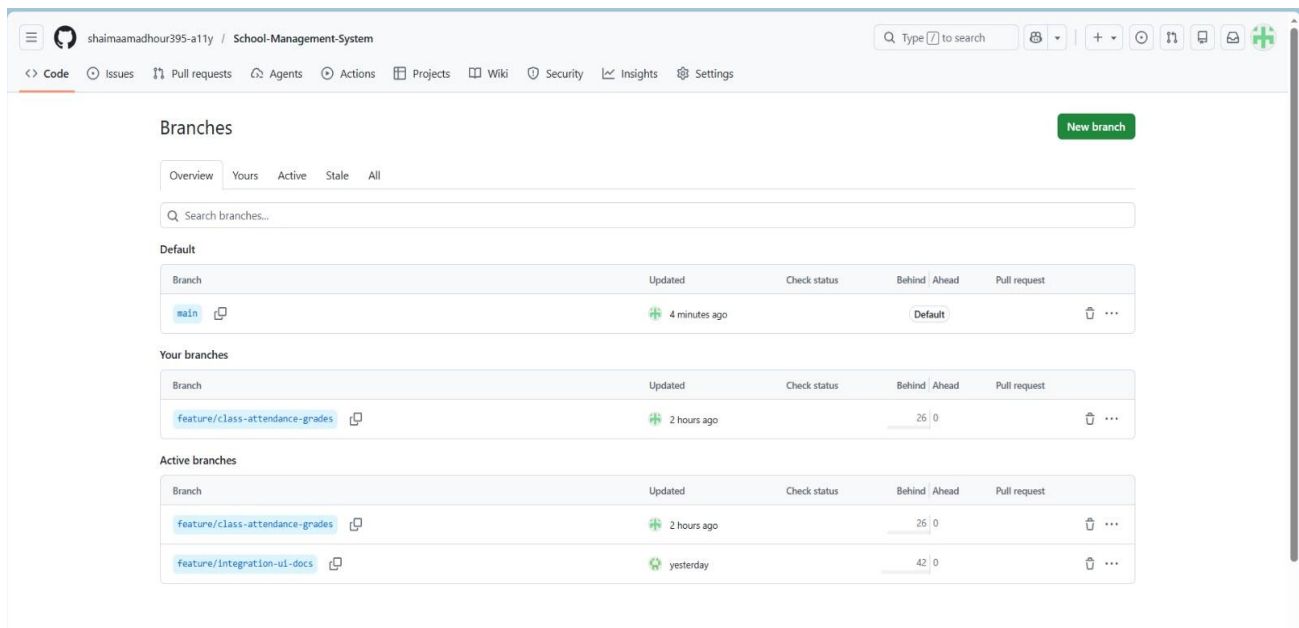
- منع إدخال طالب بدون صف موجود.
- منع إدخال درجة لطالب غير موجود.
- منع ربط معلم بمادة غير معرفة.
- استخدام Primary Key لمنع التكرار.

الفصل الرابع: إدارة الإصدارات باستخدام Git :

تم اعتماد منهجية تطوير تعتمد على:

1. إنشاء فرع مستقل لكل وحدة.

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



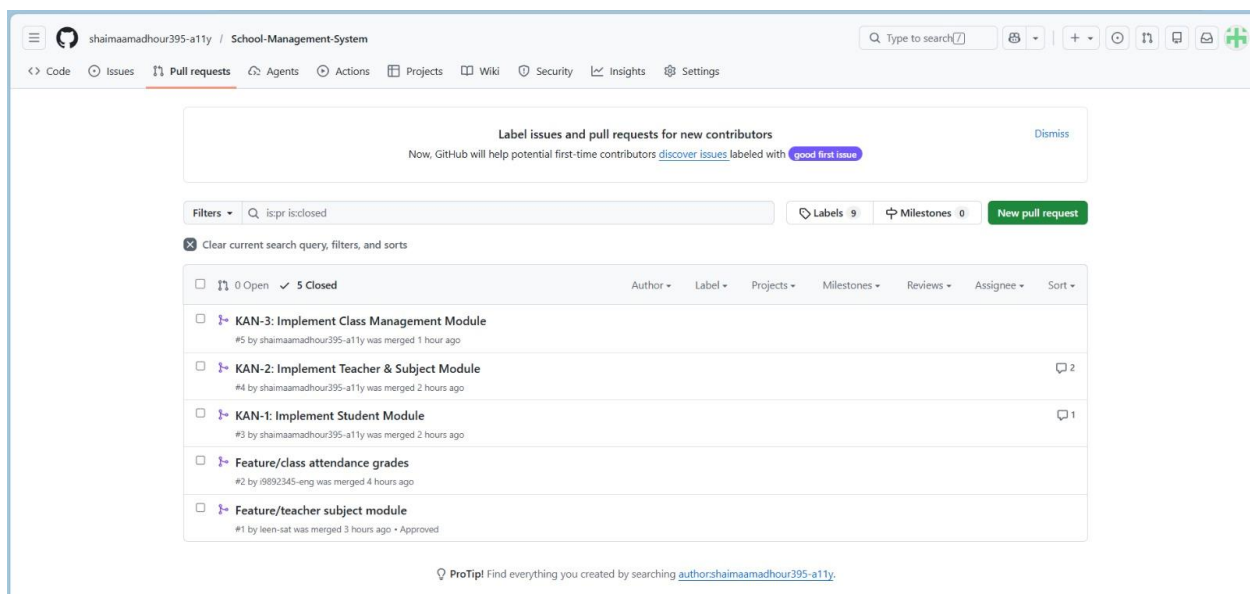
الشكل (2): إنشاء وإدارة الفروع البرمجية في GitHub.

2. تنفيذ التطوير داخل الفرع.

3. رفع التعديلات عبر Commits.

4. إنشاء Pull Requests.

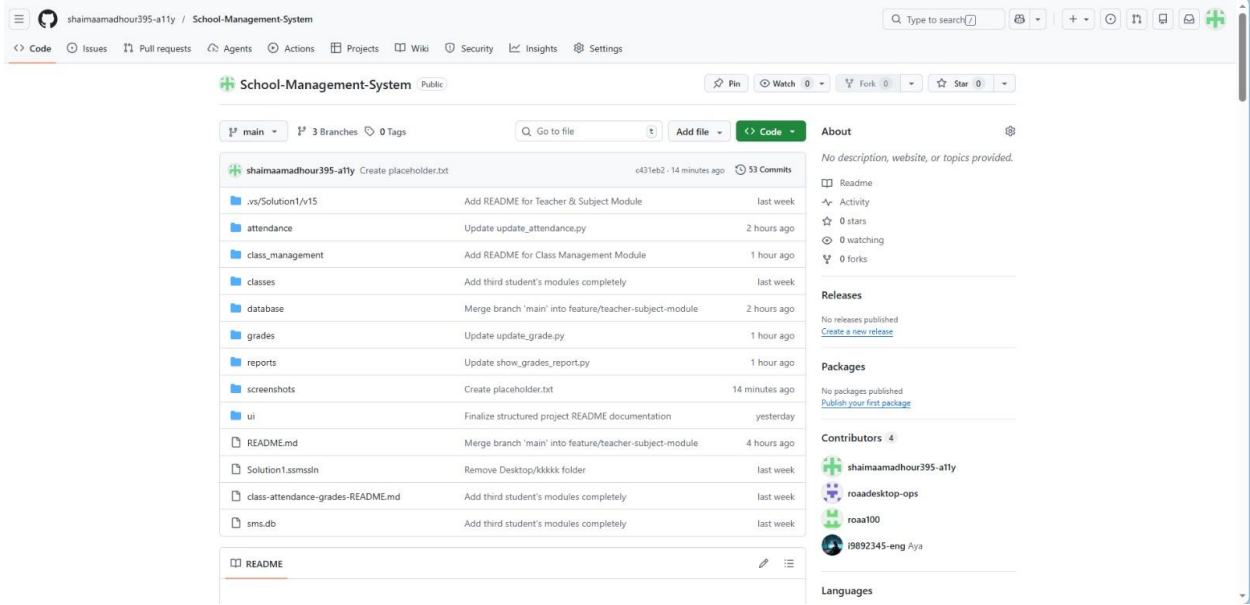
بعد الانتهاء من تطوير كل وحدة برمجية ضمن فرع مستقل، تم إنشاء Pull Requests لمراجعة التعديلات قبل دمجها في الفرع الرئيسي. يوضح الشكل (3) قائمة طلبات الدمج التي تم اعتمادها وإغلاقها بنجاح.



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

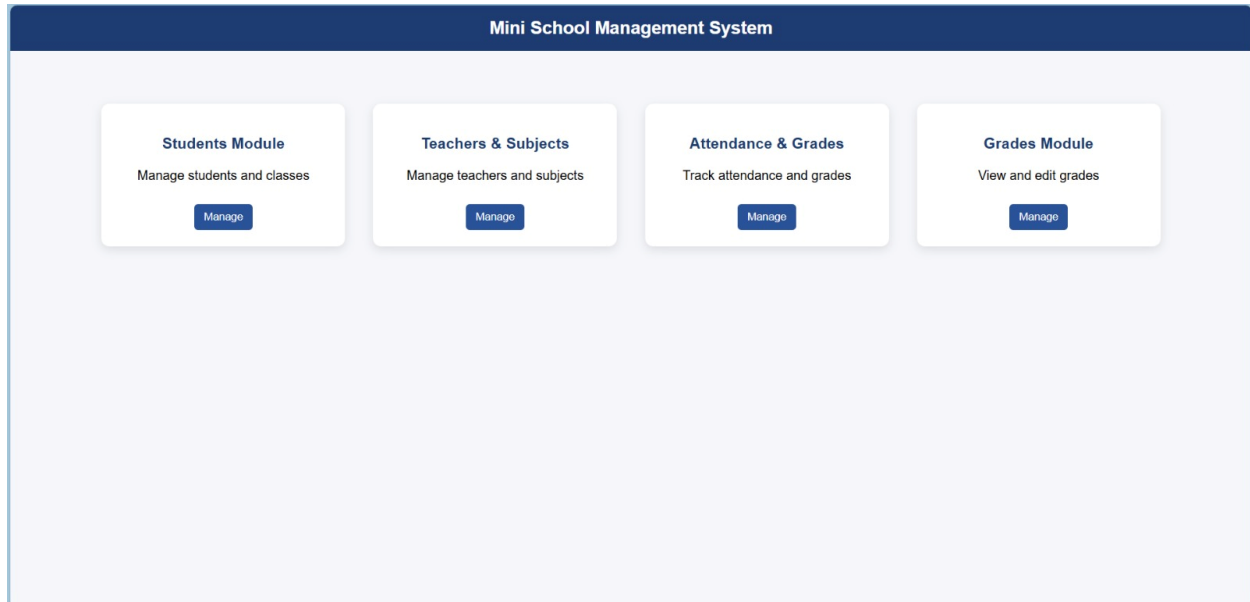
الشكل: (5) لوحة متابعة سير العمل (Kanban Board) المستخدمة لتنظيم مهام المشروع وتتبع تقدم تنفيذها.

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.

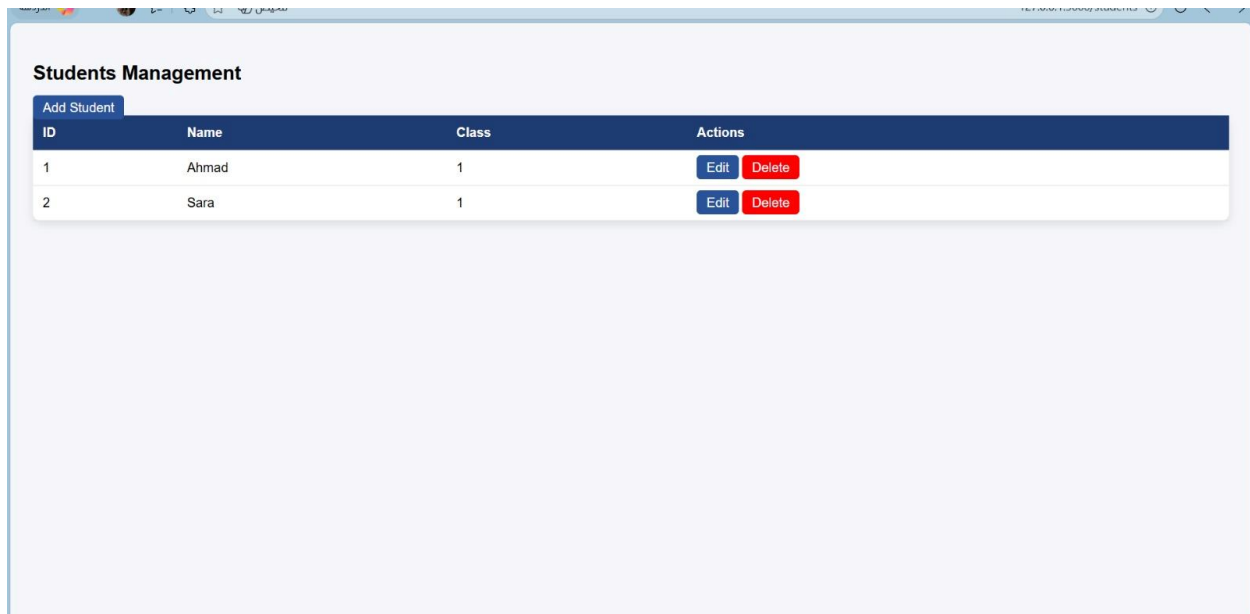
عرض وتنفيذ النظام:

1. الواجهة الرئيسية للنظام:



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

2. واجهة إدارة الطلاب:



الشكل (7): واجهة إدارة الطلاب

3. واجهة إدارة المعلمين والمواد:

Teachers & Subjects Management

[Add Teacher](#) [Add Subject](#)

Teachers

ID	Name	Actions
----	------	---------

Subjects

ID	Name	Actions
1	Mathematics	Edit Delete
2	Physics	Edit Delete

127.0.0.1:5000/add_subject

الشكل (8): واجهة إدارة المعلمين والمواد

4. واجهة سجلات الحضور:

Attendance Records

[Add Attendance](#)

ID	Student	Status	Date	Actions
1	Ahmad	Present	2026-02-17	Edit Delete

127.0.0.1:5000/attendance

الشكل (9): واجهة عرض سجلات الحضور للطلاب

5. واجهة إدارة الدرجات:

The screenshot shows a web browser window with the address bar displaying '127.0.0.1:5000/grades'. The page title is 'Grades Management'. It features a table with four columns: 'Student', 'Subject', 'Grade', and 'Actions'. The table contains four rows of data. Below the table, there is an 'Add Grade' section with a form containing three dropdown menus for 'Student' (selected 'Ahmad'), 'Subject' (selected 'Mathematics'), and 'Grade' (empty), followed by an 'Add' button. A 'Back to Home' link is located below the form.

Student	Subject	Grade	Actions
Ahmad	Mathematics	95.0	Edit Delete
Sara	Physics	88.0	Edit Delete
Sara	Physics	88.0	Edit Delete
Sara	Physics	88.0	Edit Delete

Add Grade

Student: Subject: Grade: [Add](#)

[Back to Home](#)

الشكل (10): واجهة إدارة درجات الطلاب

6. واجهة تعديل درجة:

The screenshot shows a web browser window with the address bar displaying '127.0.0.1:5000/edit_grade/4'. The page title is 'Edit Grade'. It features a form with a 'Grade' input field containing the value '88.0' and an 'Update' button. A 'Back to Grades' link is located below the form.

Edit Grade

Grade: [Update](#)

[Back to Grades](#)

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 مصغّر، إلا أنه يمثل نموذجاً عملياً يمكن تطويره ليصبح نظاماً متكاملًا 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.