بسم الله الرحمن الرحيم

جامعة بوليتكنك فلسطين

كلية تكنولوجيا المعلومات وهندسة الحاسوب

دائرة علم الحاسوب وتكنولوجيا المعلومات

مشروع تخرج بعنوان:

PPU Students Clubs

أعضاء المشروع:

أحمد محمد عصافرة

محمد جهاد عقيلي

بشار إسماعيل الهرش

مشرف المشروع:

د. ضياء أبو زينة

Certification and Anti-Plagiarism Declaration

This is to declare that the graduation project produced under the supervision of **Dr. Dia Eddin AbuZeina** having the title **PPU Students Clubs** was prepared by student/s below in partial fulfilment of the requirements for the degree of Bachelor in Computer Science and no part hereof has been reproduced illegally (in particular: cut and paste) which can be considered as Plagiarism.

All referenced parts have been used to support and argue the idea and have been cited properly. We certify that we have not committed any plagiarism, cheating, or any other academic integrity violation. We will be responsible and liable for any consequence if violation of this declaration is proven.

Any use of Artificial Intelligence (AI) tools in the preparation of this project has been clearly declared and documented in the related appendix at the end of the report, in accordance with the PPU guidelines.

Date: 01 / 06 / 2025

Graduation project group's students:

Ahmed Mohammed Asafrah

Mohammad Jihad Oqaili

Bashar Ismail Al Hersh

الإهداء

قال تعالى: {يَرْفَعِ اللَّهُ الَّذِينَ آمَنُوا مِنكُمْ وَالَّذِينَ أُوتُوا الْعِلْمَ دَرَجَاتٍ}

إلى من سهروا الليالي ودعمونا بحبهم وتضحياتهم حتى وصلنا إلى هذه اللحظة... والدينا الأعزاء، الذين زرعوا فينا قوة العزيمة والإيمان.

إلى إخوتي، نبض القلب ورفاق الدرب، الذين كانوا دومًا السند والدعم في كل محنة وفرحة.

إلى أصدقائي الأوفياء، شركاء الذكريات الجميلة والتحديات الصعبة، الذين جعلوا رحلة الحياة أسهل وأكثر إشراقًا.

إلى أساتذتنا الأجلاء، المنارات التي أضاءت لنا الدرب، ووهبتنا المعرفة والإلهام لنواصل المسير بعزيمة وثبات.

إلى كل لحظة عشناها بين جدران جامعة بوليتكنك فلسطين، حيث تعلّمنا أن الطموح لا يعرف حدودًا.

إلى الوطن الحبيب، فلسطين، أرض الصمود والتضحيات، الذي نبض في عروقنا وألهمنا حب العمل والاجتهاد، وأرواح الشهداء الذين منحونا العزة والإباء.

إلى هذا الوطن الطاهر وكل ما هو مقدس ونبيل فيه، نهدي هذا المشروع المتواضع، حصيلة جهد وإصرار لا ينضب.

الملخص:

تعد الأنشطة الطلابية في الجامعات من أهم ركائز تعزيز الحياة الجامعية لدى الطلاب فهي تسهم في بناء شخصية الطالب وتنمية مهاراته المختلفة خارج القاعات الدراسية حيث تتيح للطلاب فرصة تطوير مهارات التواصل القيادة العمل الجماعي وحل المشكلات مما يعدهم بشكل أفضل للحياة العملية بعد التخرج لذا فإن وجود منصة تجمع الأنشطة والفعاليات في مكان واحد يضمن تقديم بيئة تعليمية متكاملة ومتنوعة تعزز من تجربة الطالب الجامعية وتجعلها أكثر تفاعلا.

يهدف هذا المشروع إلى تطوير موقع إلكتروني لإدارة الأنشطة الطلابية داخل أندية الجامعة تحت إشراف عمادة شؤون الطلبة ومسؤولين يقوم الموقع بجمع كافة أندية الجامعة في منصة واحدة مما يسهل على الطلاب الاطلاع على الأندية المتاحة والأنشطة المختلفة التي تنظمها عمادة شؤون الطلبة يتيح الموقع للطلاب التسجيل بسهولة في الأندية والفعاليات بالإضافة إلى متابعة التحديثات والأنشطة الجديدة كما يسهل عملية تنظيم الفعاليات وإدارة المشاركات بشكل يتناسب مع متطلبات كل فعالية.

يشمل المشروع أتمتة عمليات تسجيل الأندية والطلاب بالإضافة إلى تنظيم الجداول الزمنية للفعاليات وإدارة المشاركات كما يساهم في توثيق كافة الأنشطة المنظمة وتوفير قاعدة بيانات يسهل الوصول إليها في أي وقت عبر هذا التحول الإلكتروني يتم الانتقال من الأساليب التقليدية إلى حلول تقنية حديثة مما يحسن من كفاءة التخطيط والتنفيذ ويوفر تجربة سلسة لكل من الطلاب والقائمين على الأنشطة هذا النهج يساهم بشكل كبير في تعزيز مشاركة الطلاب وإثراء تجربتهم الجامعية.

Abstract

Student activities in universities are one of the most important pillars of enhancing university life for students. They contribute to building the student's personality and developing his various skills outside the classroom, as they provide students with the opportunity to develop communication, leadership, teamwork and problem-solving skills, which better prepares them for practical life after graduation. Therefore, the presence of a platform that brings together activities and events in one place ensures the provision of an integrated and diverse educational environment that enhances the student's university experience and makes it more interactive.

This project aims to develop an integrated website to manage student activities within the university under the supervision of the Deanship of Student Affairs. The site collects all university clubs on one platform, which makes it easier for students to view the available clubs and various activities organized by the Deanship of Student Affairs. The site allows students to easily register for clubs and events in addition to following up on updates and new activities. It also facilitates the process of organizing events and managing participation in a manner that suits the requirements of each event.

The project includes automating club and student registration processes, as well as organizing event schedules and managing participation. It also contributes to documenting all organized activities and providing a database that is easy to access at any time. Through this digital transformation, we are moving from traditional methods to modern technical solutions, which improves the efficiency of planning and implementation and provides a smooth experience for both students and those in charge of activities. This approach contributes greatly to enhancing student participation and enriching their university experience.

فهرس المحتويات

Certification and Anti-Plagiarism Declaration 2

الإهداء 3

الفصل الأول: المقدمة 8

1.1 المقدمة: 9

1.2 الدوافع والمحفزات: 9

1.3 نطاق المشروع: 10

1.4 الأهداف: 10

1.5 الإجراءات لتحقيق الأهداف: 11

1.6 وصف مختصر للنظام: 11

1.7 توزيع الأدوار: 12

1.8 تصور عام عن النظام (Context Diagram) 13

الفصل الثاني: متطلبات النظام 14

2.1 المقدمة 14

2.2 المتطلبات الوظيفية للنظام:(Functional Requirements) 14

2.3 المتطلبات غير الوظيفية للنظام (*Non-Functional Requirements*) 17

2.4 Use Case Diagram 18

2.5 وصف المتطلبات الوظيفية للنظام 19

2.6 ملخص الفصل 32

الفصل الثالث: تصميم النظام 33

3.1 المقدمة 33

3.2 الهيكلية العامة للنظام 33

3.3 هيكلية بناء النظام: 34

3.4نظام قواعد البيانات المعتمد 35

3.5 وصف لقاعدة البيانات 37

..... 38

3.6 وصف جداول قاعدة البيانات 39

الفصل الرابع : التطبيق 47

4.1 نظرة عامة..... 47

4.2 واجهات المستخدم 50

الفصل الخامس: فحص النظام 54

5.1التحقق من صحة البيانات(Validation) 54

5.2اختبار الجودة اليدوي (Testing Manual QA)..... 55

5.3 اختبار نقاط النهاية..... 57

الخاتمة 62

الأعمال المستقبلية(Future Work) 62

المراجع(References) 63

نموذج إقرار وتوثيق استخدام أدوات الذكاء الاصطناعي في مشروع التخرج 64

فهرس الأشكال

شكل 1 Context Diagram..... 13

شكل 2 Use Case Diagram..... 18

شكل 3Model-View-Controller..... 35

شكل 4 : شكل الجداول في قاعدة البيانات 38

شكل 5 : توضيح للجداول في قاعدة البيانات 38

فهرس الجداول

جدول 1: وصف عملية إدارة المستخدمين.....19
جدول 2 :وصف عملية إدارة جميع الأندية.....20
جدول 3 : وصف عملية إدارة جميع الفعاليات.....21
جدول 4 :وصف عملية إدارة النادي المحدد.....22
جدول 5: وصف عملية إدارة العضوية.....23
جدول 6 : وصف عملية إدارة الفعالية الخاصة بالنادي.....24
جدول 7: وصف عملية التقديم للعضوية.....25
جدول 8: وصف عملية متابعة الفعاليات.....26
جدول 9 : وصف عملية لوحة المتصدرين.....27
جدول 10: وصف عملية كتابة تعليق.....28
جدول 11:وصف عملية عمل إعجاب.....29
جدول 12:وصف عملية إنشاء حساب مستخدم.....31
جدول 13: وصف عملية تسجيل الدخول.....32
جدول 14: الفرق بين قواعد البيانات العلائقية وغير العلائقية.....36
جدول 15:وصف لقاعدة البيانات.....38
جدول 16: وصف حقول جدول المستخدم.....40
جدول 17 :وصف حقول جدول الأندية.....41
جدول 18:وصف حقول جدول الفعاليات.....41
جدول 19:وصف حقول جدول المنشورات.....42
جدول 20: وصف حقول جدول الأحداث.....43
جدول 21 :وصف حقول جدول الإعجابات.....44
جدول 22:وصف حقول جدول المتصدرين.....45
جدول 23 : وصف جدول التسجيل في الأندية.....46
جدول 24 : جدول استخدام أدوات الذكاء الاصطناعي.....66

الفصل الأول: المقدمة

1.1 المقدمة:

في زمن تتسارع فيه عجلة التطور التكنولوجي بشكل غير مسبوق، اصبحت التكنولوجيا عنصرا جوهريا يؤثر بشكل مباشر على مختلف نواحي الحياة اليومية. فقد تمكنت من اعادة تشكيل الاساليب التقليدية التي كنا نعتمد عليها في التواصل، الادارة، والتعليم، مما ساعد على تقديم حلول مبتكرة وفعالة تلبي احتياجات الافراد والمؤسسات بطريقة أكثر دقة وكفاءة.

في عالم التعليم، أصبحت الأنظمة التقنية الحديثة ضرورة لا غنى عنها لتحقيق التحول الرقمي، حيث تتيح للجامعات والمؤسسات التعليمية امكانية تحسين بيئتها التنظيمية والادارية بشكل جذري. من خلال تبسيط العمليات وتعزيز التعاون بين الطلاب واعضاء الهيئات الاكاديمية، تستطيع المؤسسات التعليمية ان تقدم تجربة متكاملة تزيد من التفاعل والارتباط داخل المجتمع الجامعي.

مشروع "PPU Students Clubs" يعد موقع الكتروني شامل يهدف الى توفير بيئة تنظيمية متكاملة تخدم جميع طلاب الجامعة، تحت إشراف عمادة شؤون الطلبة. يقدم المشروع خدمات متنوعة تشمل تسجيل الطلاب في الأندية، تنظيم الانشطة المختلفة، بالإضافة الى تزويد المسؤولين بحسابات ادارية توفر لهم صلاحيات مخصصة لإدارة النظام بكفاءة.

تم تطوير هذا المشروع انطلاقا من الحاجة الملحة الى وجود منصة إلكترونية متكاملة لتحسين تنظيم الانشطة الطلابية في الجامعة، والتخلص من الطرق التقليدية التي قد تكون بطيئة وغير فعالة. بفضل الحلول التقنية الحديثة التي يعتمدها المشروع، سيتمكن الطلاب من التفاعل بشكل أفضل مع بيئتهم الجامعية، مما يعزز من مشاركتهم ويثري تجربتهم الاكاديمية والعملية، ويخلق مجتمعا جامعيا متطورا.

1.2 الدوافع والمحفزات:

1. عدم وجود منصة موحدة تجمع طلاب الجامعة في مكان واحد لتسهيل الوصول إلى الأندية والفعاليات والمعلومات بشكل منظم وفعال
2. تعزيز وعي الطلاب بجميع الأنشطة والفعاليات التي تقام داخل الجامعة مما يساعدهم على المشاركة الفعالة وتنمية مهاراتهم
3. التخلي عن الأساليب التقليدية التي تعتمد على الإدارة اليدوية للأندية والفعاليات واستبدالها بحلول رقمية توفر الوقت والجهد وتقلل الأخطاء
4. توفير آلية شاملة لإدارة الفعاليات بما يشمل التسجيل المتابعة وتنظيم المشاركات بطريقة مرنة وسلسة
5. تعزيز روح التفاعل والمنافسة بين الطلاب من خلال نظام لوحات الصدارة والمسابقات مما يشجعهم على تقديم أفضل ما لديهم
6. تقديم حلول تقنية متطورة تواكب التحول الرقمي وتعزز الكفاءة الإدارية لعمادة شؤون الطلبة والأندية الجامعية

1.3 نطاق المشروع:

المشروع يركز بشكل أساسي على طلاب جامعة بوليتكنك فلسطين من خلال توفير منصة إلكترونية تساعدهم في التفاعل والمشاركة في الأنشطة والفعاليات الجامعية. المنصة ستكون وسيلة لتسهيل انضمام الطلاب للأندية والفعاليات المختلفة مما يعزز تجربتهم الجامعية بشكل شامل. بالإضافة إلى ذلك يستهدف المشروع العاملين داخل الجامعة مثل موظفي عمادة شؤون الطلبة وأعضاء الهيئة التدريسية حيث سيتمكنون من إدارة الفعاليات وتنظيم الأنشطة عبر حسابات إدارية خاصة بهم.

يسعى المشروع إلى تحسين التواصل بين الطلاب والإدارة وتوفير بيئة تفاعلية تسهم في تسهيل تنظيم الأنشطة. كما ويعمل على أتمتة العديد من العمليات التي كانت تتم يدويًا مثل التسجيل في الأندية والفعاليات مما يوفر الوقت والجهد.

إن المشروع يشمل جميع الأنشطة المتعلقة بالحياة الجامعية من مشاركة الطلاب في الفعاليات إلى تنظيم وإدارة الأنشطة من قبل الإدارة ليقدم بيئة أكاديمية أكثر تفاعلية وسلاسة لجميع الأطراف.

1.4 الأهداف:

يهدف المشروع الى توفير:

- منصة إلكترونية تفاعلية تمكن الطلاب من التسجيل والانضمام إلى الأندية والمشاركة في الفعاليات.
- منصة موحدة تجمع أندية الجامعة في مكان واحد، لتسهيل الوصول إليها من قبل الطلاب والإدارة.
- تشجيع الطلاب الجدد على الانضمام للأندية والفعاليات من خلال عرض شامل للأندية والأنشطة المتاحة.
- تمكين عمادة شؤون الطلبة من إدارة الأنشطة والفعاليات بكفاءة عالية من خلال أدوات رقمية متطورة.
- تعزيز روح المنافسة بين الطلاب من خلال نظام المسابقات ولوحة المتصدرين التي تشجع على المشاركة الفعالة.
- تسهيل عملية البحث عن الأندية والفعاليات بما يتناسب مع اهتمامات الطلاب واحتياجاتهم.
- تحسين عملية التواصل بين الطلاب والإدارة عبر منصات تفاعلية تدعم تبادل المعلومات والاقتراحات.
- أتمتة عملية التسجيل في الأنشطة والفعاليات لتوفير الوقت والجهد على الطلاب والإدارة.
- دعم تفاعل الطلاب مع أعضاء الأندية وأعضاء الهيئة التدريسية من خلال أدوات تواصل مباشرة.
- تحقيق تجربة جامعية متميزة من خلال دمج جميع الأنشطة الاجتماعية والثقافية في بيئة رقمية واحدة
- توثيق الأنشطة والفعاليات في قاعدة بيانات مركزية لتسهيل الوصول إليها ومراجعتها لاحقًا

1.5 الإجراءات لتحقيق الأهداف:

من أجل تحقيق هذه الأهداف سيتم اجراء ما يلي:

- تصميم وتطوير واجهة مستخدم سهلة الاستخدام لكل من الطلاب والمسؤولين بحيث تكون بديهية وتتناسب مع احتياجات جميع الفئات المستهدفة.
- تطوير نظام إدارة الأندية والفعاليات بما يضمن سلاسة التعامل مع الطلبات والتأكد من أن النظام يدعم العمليات بشكل مرن وفعال.
- إنشاء لوحة النتائج لتعزيز المشاركة الفعالة بين الطلاب وتشجيع التنافسية الإيجابية من خلال تتبع النتائج والتقييمات في الأنشطة والمسابقات.
- بناء عملية تسجيل الطلاب في الأندية والفعاليات لتسهيل الإجراءات وتقليل الوقت المستغرق في التعاملات اليدوية.
- تحسين الأمان وحماية البيانات لضمان سرية معلومات الطلاب والأنشطة وعدم تعرضها لأي اختراقات.
- توفير ميزة التعليقات من خلال السماح للطلاب بتقديم ملاحظات وتقييمات حول الأنشطة والفعاليات، مما يساعد على تحسين الجودة بشكل مستمر.

1.6 وصف مختصر للنظام:

يعمل النظام على تمكين الطلاب من التسجيل باستخدام حساب "طالب" خاص بهم، حيث يمكنهم الانضمام بسهولة إلى الأندية والمشاركة في الفعاليات المختلفة التي تنظمها الجامعة. بالإضافة إلى ذلك، يوفر النظام للمسؤولين حسابات خاصة تحتوي على صلاحيات إدارية تمكنهم من إدارة وتنظيم الأنشطة والفعاليات بكفاءة عالية. يتم عرض جميع الفعاليات والمسابقات المستقبلية في صفحة مخصصة، تسهل على الطلاب الاطلاع على الأنشطة المتاحة ومواعيدها. كما يتضمن النظام نظام إشعارات فعال يضمن وصول التنبيهات بشكل فوري إلى الطلاب، مما يبقيهم على اطلاع دائم بكل جديد يتعلق بالفعاليات والأنشطة التي تهمهم. هذا النظام يعزز من التفاعل بين الطلاب وإدارات الأندية، مما يسهل عملية التواصل ويزيد من فرص المشاركة الفعالة في الأنشطة المختلفة. كما يساهم في تنظيم الفعاليات بشكل مرن وسريع ويوفر بيئة تعليمية واجتماعية متكاملة.

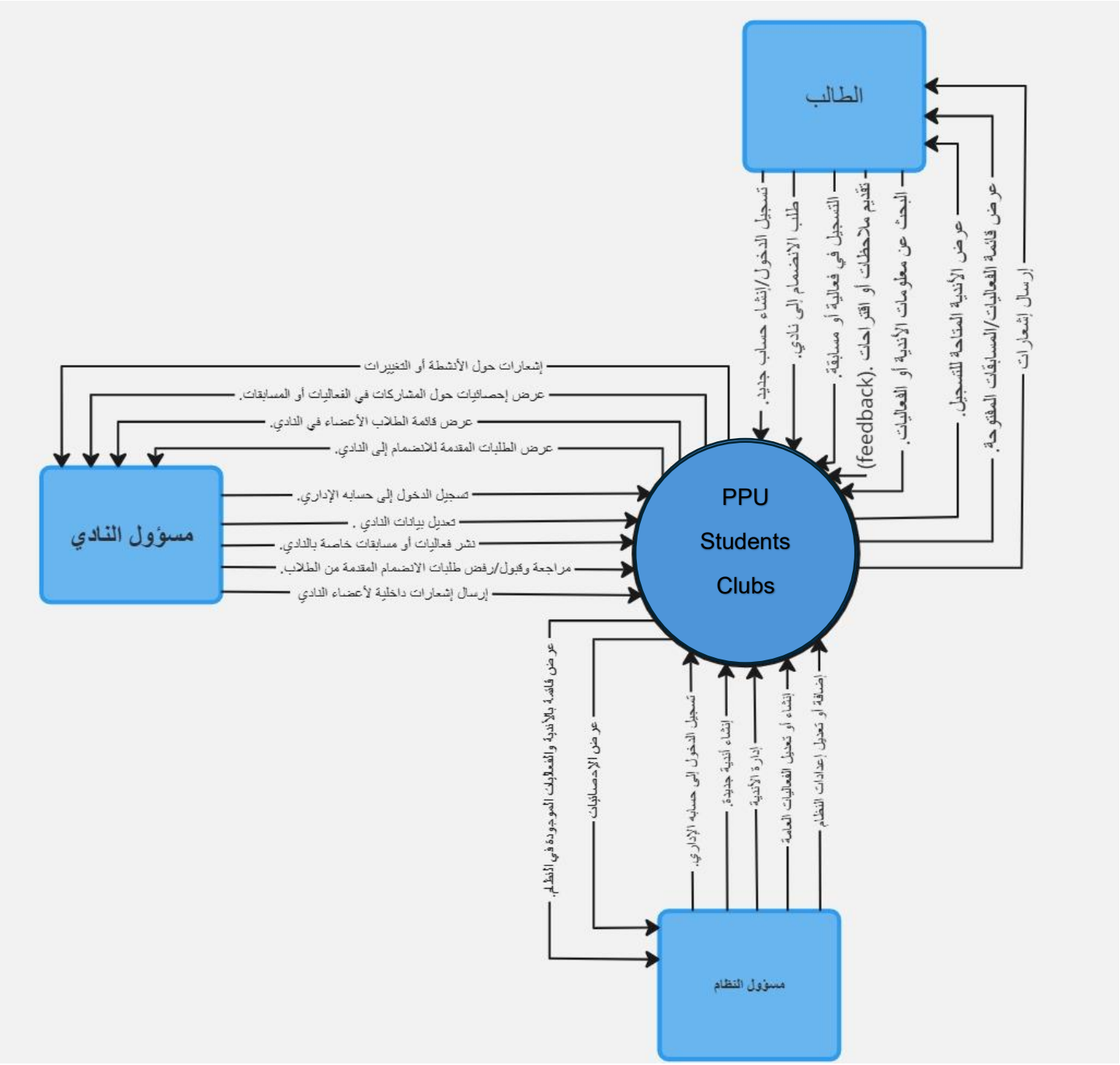
1.7 توزيع الأدوار:

يتكون فريق العمل من ثلاثة أعضاء، حيث يتم تبادل الأدوار بيننا طوال مراحل تطوير النظام، بما في ذلك إدارة النظام، البرمجة، وتحميل البيانات، وغيرها من المهام. يتم هذا التناوب لأن كل فرد من أفراد الفريق يمتلك مهارات متميزة في مجالات معينة أكثر من غيره.

يهدف هذا التناوب إلى إعطاء ديناميكية للعمل والاستفادة قدر الإمكان من المشروع من خلال زيادة القدرات والمهارات لجميع أفراد الفريق في جميع مجالات المشروع.

- **جمع البيانات:** من خلال المقابلات مع الطلاب والعاملين في إدارة النظام، ومن ثم تحميل هذه البيانات وتحليلها للحصول على المعلومات والإحصائيات اللازمة.
- **تحديد متطلبات النظام:** تحديد جميع المتطلبات الضرورية لإنشاء النظام وضمان تلبية احتياجات الطلاب والإدارة.
- **تصميم النظام:** تصميم النظام بشكل كامل مع وضع خطة واضحة لكافة مكوناته ووظائفه.
- **برمجة الموقع الإلكتروني:** البدء في تطوير الموقع الإلكتروني وربطه بقاعدة البيانات الخاصة بالجامعة لتسهيل إدارة الأنشطة والتسجيل.
- **فحص النظام:** تطبيق النظام في الجامعة لفترة تجريبية محددة لضمان فعاليته في التعامل مع إجراءات التسجيل وتنظيم الأنشطة.
- **توثيق النظام:** توثيق جميع مراحل تطوير النظام، بما في ذلك التفاصيل الفنية والتقارير المتعلقة بالاختبارات والنتائج لضمان سهولة الصيانة والتحديثات المستقبلية.

1.8 تصور عام عن النظام (Context Diagram)



شكل 1 Context Diagram

الفصل الثاني: متطلبات النظام

2.1 المقدمة

في هذا الفصل، سيتم تقديم شرح تفصيلي حول متطلبات النظام الوظيفية (Functional Requirements)ومتطلبات النظام غير الوظيفية (Non-Functional Requirements)، حيث تشكل هذه المتطلبات الأساس الذي يعتمد عليه تطوير النظام وفهم آلية عمله بشكل دقيق. سيتم التطرق إلى الوظائف التي يجب أن يقدمها النظام للمستخدمين لتحقيق الأهداف المرجوة، كما سيتم تحديد الخصائص التي ينبغي أن يمتاز بها النظام لضمان جودته وكفاءته، مثل السرعة، الأمان، القابلية للتوسع، والموثوقية.

بالإضافة إلى ذلك، سيتم عرض رسمة حالة الاستخدام (Use Case Diagram)بشكل مفصل لتوضيح كيفية تفاعل المستخدمين مع النظام في سيناريوهات مختلفة، مع تسليط الضوء على العمليات المختلفة التي يمكن أن يقوم بها كل نوع من المستخدمين، سواء كانوا من الطلاب، الإداريين، أو أي أطراف أخرى لها علاقة بالنظام. ستساعد هذه الرسمة في تقديم تصور مرئي ومنظم عن كيفية ارتباط المستخدمين بالنظام ومدى تداخلهم مع وظائفه، بالإضافة إلى كيفية تفاعل النظام مع قواعد البيانات لضمان تنفيذ العمليات بسلاسة وكفاءة.

من خلال هذا العرض المفصل للمتطلبات وحالة الاستخدام، سيتسنى فهم كيف يعمل النظام بشكل متكامل، وكيف تساهم هذه المكونات في تحقيق الهدف النهائي للنظام بطريقة مرنة وفعالة.

2.2 المتطلبات الوظيفية للنظام:(Functional Requirements)

المتطلبات الوظيفية هي تلك الوظائف والمهام التي يجب أن يؤديها النظام لتحقيق الأهداف التي تم تصميمه من أجلها. هذه المتطلبات توضح ماذا يجب أن يقوم به النظام عند التفاعل مع المستخدمين أو مع البيئة المحيطة به

أ) المتطلبات الوظيفية للطالب:

1. **تسجيل الدخول:**
   - إنشاء حساب جديد باستخدام بياناته الجامعية.
   - تسجيل الدخول إلى النظام باستخدام بيانات الاعتماد.
2. **إدارة العضوية:**
   - تصفح قائمة الأندية المتاحة.
   - إرسال طلب انضمام إلى أحد الأندية.

3. **إدارة الفعاليات:**

- عرض قائمة الفعاليات والمسابقات المتاحة.
- متابعة التحديثات أو التغييرات في مواعيد الفعاليات.

4. **نظام الإشعارات:**

- استلام إشعارات حول الفعاليات أو حالة طلبات الانضمام.
- استلام تنبيهات بخصوص الفعاليات الجديدة أو المواعيد.

5. **لوحة النتائج:**

- عرض ترتيب الطالب بين الطلاب في لوحة النتائج بناءً على مشاركاته.

6. **كتابة تعليق:**

- التعليق على المنشورات الخاصة بالأندية والفعاليات .

7. **عمل إعجاب:**

- عمل اعجاب على المنشورات والفعاليات.

ب) المتطلبات الوظيفية لمسؤول النظام (عميد شؤون الطلبة):

1. **إدارة المستخدمين:**

- تسجيل أو حذف أو تعديل حسابات المستخدمين (طلاب، مسؤولي أندية).
- ضبط صلاحيات مسؤولي الأندية وإدارتها.

2. **إدارة الأندية:**

- إنشاء أندية جديدة داخل النظام.
- تعديل أو حذف أندية موجودة.
- تعيين مسؤولي أندية أو تحديث بياناتهم.

3. **إدارة الفعاليات:**

- إنشاء فعاليات ومسابقات عامة للجامعة.
- مراجعة الفعاليات التي ينشئها مسؤولو الأندية.

4. **كتابة تعليق:**

- التعليق على المنشورات الخاصة بالأندية والفعاليات .

5. **عمل إعجاب:**

- عمل اعجاب على المنشورات والفعاليات.

6. **إنشاء منشور:**

- إنشاء منشور خاص به
- تعديل وحذف المنشورات

7. **إنشاء لوحة نتائج:**

- القدرة على إنشاء وتعديل وحذف لوحة النتائج

ج) المتطلبات الوظيفية لمسؤول النادي:

1. **إدارة النادي:**

- تعديل بيانات النادي (الوصف، الصورة، الشروط).
- عرض قائمة الأعضاء الحاليين.

2. **إدارة العضوية:**

- مراجعة طلبات انضمام الطلاب إلى النادي.
- قبول أو رفض الطلبات بناءً على الشروط المحددة.

3. **إدارة الفعاليات:**

- إنشاء فعاليات ومسابقات خاصة بالنادي.
- تعديل أو حذف الفعاليات قبل تاريخها.
- متابعة تسجيل الطلاب في الفعاليات.

4. **كتابة تعليق:**

- التعليق على المنشورات الخاصة بالأندية والفعاليات .

5. **عمل إعجاب:**

- عمل اعجاب على المنشورات والفعاليات

6. **إنشاء منشور:**

- إنشاء منشور خاص بالنادي الذي يديره
- تعديل المنشور الخاص بالنادي وحذفه

7. **إنشاء لوحة نتائج:**

- القدرة على إنشاء وتعديل لوحة نتائج لمسابقات النادي

2.3 المتطلبات غير الوظيفية للنظام (*Non-Functional Requirements*)

المتطلبات غير الوظيفية تهدف إلى تحديد معايير الأداء والجودة التي يجب أن يحققها النظام لضمان توفير تجربة استخدام سلسة ومريحة.

فيما يلي المتطلبات غير الوظيفية للنظام:

1) الأداء العالي (Performance):

- يجب أن يتم تحميل الصفحات عند الانتقال بين الواجهات خلال 1 إلى 2 ثانية كحد أقصى، لضمان عدم شعور المستخدم بالتأخير النظام بسرعة وكفاءة عالية لضمان تجربة مستخدم سلسة.
- يتم تحميل الصفحات عند الانتقال بين الواجهات خلال 1 إلى 2 ثانية كحد أقصى، لضمان عدم شعور المستخدم بالتأخير
- يتم تنفيذ العمليات البسيطة (مثل "أعجبني" أو "انضمام إلى فعالية") خلال 100 إلى 300 مللي ثانية لضمان استجابة فورية للتفاعل .

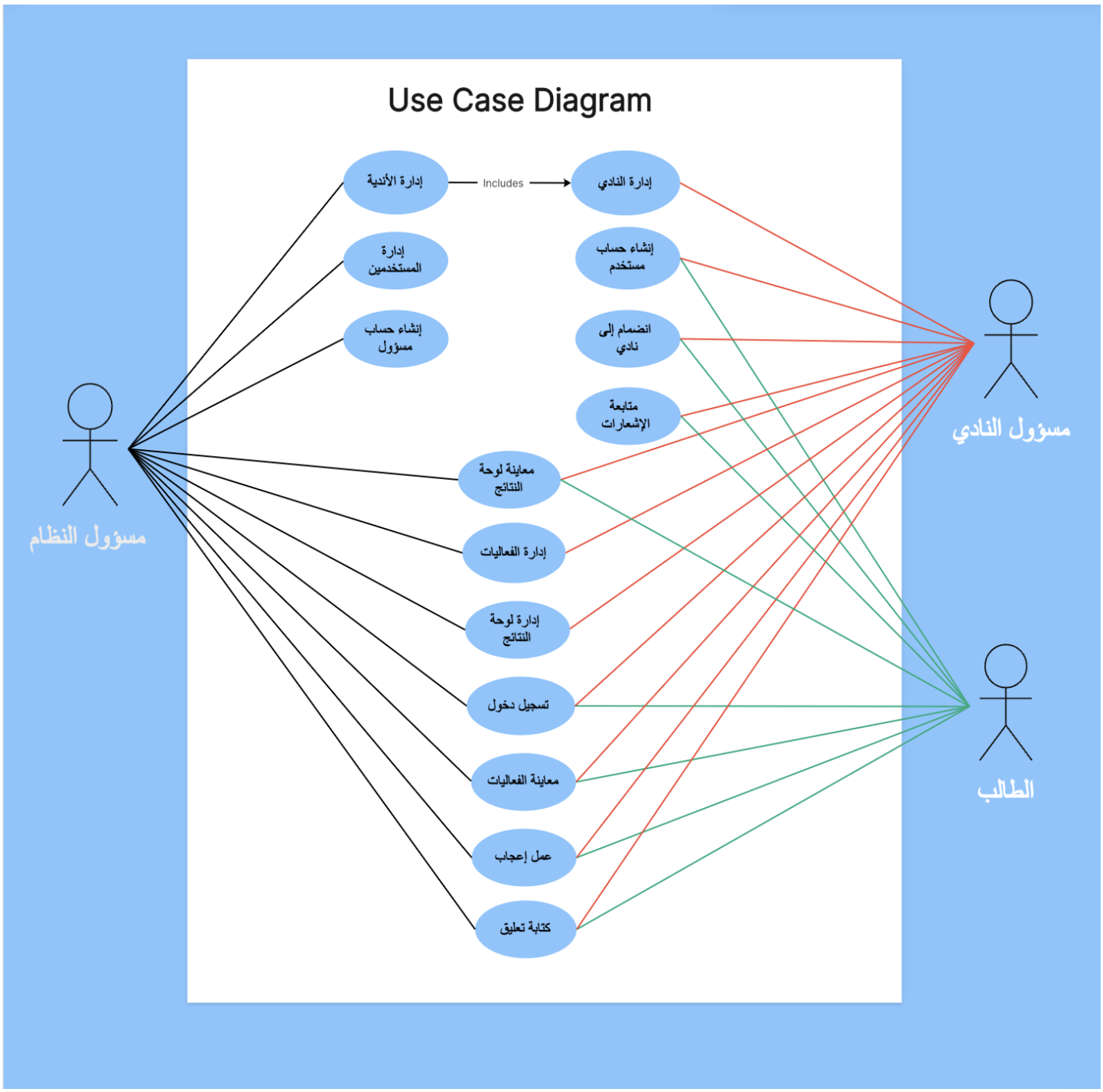
2) الأمن والحماية (Security) :

- تم حماية بيانات المستخدمين من الوصول غير المصرح به من خلال تطبيق تشفير قوي لكلمات المرور باستخدام مكتبة bcrypt، مما يضمن عدم تخزين كلمات المرور بنص صريح في قاعدة البيانات.
- تم تصميم واجهات النظام بحيث :
  - يفصل الواجهات الخاصة بالمدير (Admin) عن باقي المستخدمين على الواجهة الأمامية (Frontend)، مما يمنع ظهور أو تنفيذ المهام الإدارية لغير المخولين بها .
  - تُعرض المهام والصلاحيات وفقًا لنوع الحساب، ولا يتم إرسال أي بيانات أو صلاحيات إدارية إلى المستخدمين العاديين .

3) سهولة الاستخدام (Usability) :

- تصميم واجهة مستخدم بسيطة وسهلة الفهم لجميع الفئات المستهدفة.
- توفير تجربة مستخدم مرنة تدعم التفاعل السريع والواضح مع النظام.

Use Case Diagram 2.4



Use Case Diagram 2 شكل

2.5 وصف المتطلبات الوظيفية للنظام

وصف عملية إدارة المستخدمين

إدارة حسابات المستخدمين تشمل إضافة، تعديل، وحذف حسابات الطلاب ومسؤولي الأندية. كما يتم ضبط صلاحيات الأعضاء بناءً على الأدوار المعينة لهم.

اسم الوظيفة	إدارة المستخدمين
الممثل	**مسؤول شؤون الطلبة**
الوصف	**إدارة حسابات المستخدمين (طلاب، مسؤولي أندية)**
المدخلات	**بيانات المستخدم (اسم، كلمة مرور)**
المخرجات	**تحديث حالة الحساب**
الإجراءات	**1) إضافة أو تعديل أو حذف حسابات المستخدمين** **2) ضبط صلاحيات المستخدمين** **3) إدارة الأدوار والصلاحيات**
الاستثناءات	**حدوث خطأ في بيانات المستخدم**

جدول 1: وصف عملية إدارة المستخدمين

وو

وصف عملية إدارة الأندية

مسؤول النظام يتولى إدارة الأندية عبر إضافة أندية جديدة، تعديل بياناتها، أو حذفها. يتم إدارة تفاصيل الأندية مثل الاسم، الوصف، والصورة بشكل مرن.

اسم الوظيفة	إدارة الأندية
الممثل	**مسؤول شؤون الطلبة.**
الوصف	**إدارة الأندية (إنشاء، تعديل، حذف)**
المدخلات	**تفاصيل النادي (اسم، وصف، صورة)**
المخرجات	**إضافة أو تعديل أو حذف النادي**
الإجراءات	**1) إضافة نادي جديد** **2) تعديل بيانات النادي** **3) حذف نادي**
الاستثناءات	**إذا فشلت عملية إضافة أو تعديل النادي، تظهر رسالة خطأ**

جدول 2 :وصف عملية إدارة جميع الأندية

وصف عملية إدارة الفعاليات

تتمثل مهام مسؤولي النظام والنادي في إدارة الفعاليات، من إنشاء فعالية جديدة مع نشر المعلومات المتعلقة بالفعاليات.

اسم الوظيفة	إدارة الفعاليات
الممثل	**مسؤول النادي ، مسؤول شؤون الطلبة .**
الوصف	**إدارة الفعاليات (إنشاء، تعديل، حذف)**
المدخلات	**تفاصيل الفعالية (اسم، تاريخ، صورة)**
المخرجات	**منشور خاص بالفعالية**
الإجراءات	**1) إضافة فعالية جديدة** **2) وصف الفعالية** **3) اظهار منشور خاص بالفعالية** **4) الغاء فعالية**
الاستثناءات	**عدم وجود فعالية**

جدول 3 : وصف عملية إدارة جميع الفعاليات

وصف عملية إدارة النادي

مسؤول النادي يتولى تعديل بيانات النادي وعرض الأعضاء الحاليين، مما يساعد في ضمان تحديث المعلومات باستمرار وتحسين التفاعل بين الأعضاء.

اسم الوظيفة	إدارة النادي
الممثل	**مسؤول النادي**
الوصف	**إدارة بيانات النادي (تعديل، عرض الأعضاء)**
المدخلات	**تفاصيل النادي (اسم، وصف)**
المخرجات	**تحديث بيانات النادي**
الإجراءات	**1) تعديل بيانات النادي** **2) عرض قائمة الأعضاء الحاليين**
الاستثناءات	**فشل التعديل أو التحديث**

جدول 4 :وصف عملية إدارة النادي المحدد

وصف عملية إدارة العضوية

مسؤول النادي يدير طلبات انضمام الطلاب إلى النادي من خلال مراجعة ، ويقوم بقبول أو رفض الطلبات.

اسم الوظيفة	إدارة العضوية
الممثل	**مسؤول النادي**
الوصف	**إدارة طلبات انضمام الطلاب إلى النادي**
المدخلات	**طلبات الانضمام من الطلاب**
المخرجات	**حالة الطلب (مقبول/مرفوض)**
الإجراءات	**1) مراجعة الطلبات** **2) قبول أو رفض الطلبات**
الاستثناءات	**فشل قبول أو رفض الطلب**

جدول 5: وصف عملية إدارة العضوية

وصف عملية إدارة الفعاليات

مسؤول النادي يتعامل مع الفعاليات الخاصة بالنادي عبر إضافة، تعديل الفعاليات.

اسم الوظيفة	إدارة الفعاليات
الممثل	**مسؤول النادي**
الوصف	**إدارة الفعاليات الخاصة بالنادي**
المدخلات	**تفاصيل الفعالية (اسم، تاريخ، الموقع، الوقت)**
المخرجات	**منشور خاص بالفعالية**
الإجراءات	**1) إضافة فعالية جديدة** **2) تعديل الفعاليات قبل تاريخها** **3) متابعة تسجيل الطلاب**
الاستثناءات	**فشل إضافة أو تعديل الفعالية**

جدول 6 : وصف عملية إدارة الفعالية الخاصة بالنادي

وصف عملية التقديم للعضوية

الطلاب يقدمون طلبات الانضمام إلى الأندية من خلال اختيار الأندية المتاحة، إرسال الطلبات، ومتابعة حالة قبول أو رفض الطلب.

اسم الوظيفة	التقديم للعضوية
الممثل	**الطالب، مسؤول النادي**
الوصف	**التقديم للانضمام إلى أحد الأندية**
المدخلات	**اختيار النادي وتقديم طلب العضوية**
المخرجات	**حالة الطلب (مقبول/مرفوض)**
الإجراءات	**1) تصفح الأندية المتاحة** **2) إرسال طلب الانضمام**
الاستثناءات	**رفض الطلب**

جدول 7: وصف عملية التقديم للعضوية

وصف عملية معاينة الفعاليات

الطلاب يتابعون الفعاليات المتاحة عن طريق عرض تفاصيل الفعاليات والتسجيل فيها.

اسم الوظيفة	معاينة الفعاليات
الممثل	**الطالب**
الوصف	**متابعة الفعاليات والمسابقات المتاحة**
المدخلات	**قائمة الفعاليات المتاحة**
المخرجات	**تفاصيل الفعالية والتغييرات في المواعيد**
الإجراءات	**1) عرض الفعاليات المتاحة** **2) التسجيل في الفعاليات**
الاستثناءات	**اكتمال العدد المطلوب**

جدول 8: وصف عملية متابعة الفعاليات

وصف عملية إدارة لوحة النتائج

يتم عرض ترتيب المراكز الثلاثة الأولى من كل مسابقة، مع إمكانية تعديل أو حذف محتوى لوحة النتائج من قبل مدير النادي أو مدير التظام .

اسم الوظيفة	لوحة المتصدرين
الممثل	**مسؤول النادي ، مسؤول شؤون الطلبة .**
الوصف	**عرض ترتيب المراكز الثلاث الأولى من كل مسابقة يتم عملها .**
المدخلات	**أسماء طلاب المراكز الثلاث الأولى** **اسم المسابقة** **وصف المسابقة** **نتيجة الطالب**
المخرجات	**ترتيب الطالب في لوحة المتصدرين**
الإجراءات	**1) الانتقال الى لوحة النتائج وملئ أسماء اول ثلاثة مراكز** **2) تحديث بيانات لوحة النتائج أو حذف أحد المسابقات**
الاستثناءات	**حدوث مشكلة في الشبكة**

جدول 9 : وصف عملية لوحة المتصدرين

وصف عملية كتابة تعليق

عملية كتابة تعليق تتيح للمستخدم التفاعل مع المنشورات من خلال إدخال نص في خانة التعليقات وإرساله، ليتم عرضه ضمن التعليقات أسفل المنشور.

اسم الوظيفة	كتابة تعليق
الممثل	**بلاطلا مسؤول النادي ، مسؤول شؤون الطلبة .**
الوصف	**ةدامعلا وأ ٍداذلبق نم ةروشنم ةيلاعف ىلع قيلعت ةفاضإ.**
المدخلات	**قيلعتلا صن.**
المخرجات	**ةيلاعفلا ةحفصي ف قيلعتلا ضرع**
الإجراءات	**1) الذهاب للمنشور أو الفعالية الخاصة بنادي أو شؤون الطلبة** **2) كتابة نص التعليق** **3) تحديث التعليقات ليتم إضافة التعليق لها**
الاستثناءات	**إرسال تعليق فارغ ، حذف المنشور اثناء التعليق**

جدول 10: وصف عملية كتابة تعليق

وصف عملية عمل إعجاب

عملية الإعجاب تُمكّن المستخدم من التفاعل مع منشور معين عبر الضغط على زر "إعجاب"، حيث يتم ت وتحديث عدد الإعجابات المعروض للمستخدمين.

اسم الوظيفة	عمل إعجاب
الممثل	**طالب، مسؤول النادي ، مسؤول شؤون الطلبة .**
الوصف	**يمكن للمستخدم الضغط للتفاعل مع منشورات الأنشطة أو الفعاليات.**
المدخلات	**الضغط على زر "إعجاب" بجانب منشور معين.**
المخرجات	**1) الذهاب الى لوحة النتائج.** **2) عرض تغيير في واجهة المستخدم (زر الإعجاب يتغير ليتم إبداء الإعجاب).**
الإجراءات	**3) الذهاب للمنشور أو الفعالية الخاصة بنادي أو شؤون الطلبة** **4) الضغط على زر إعجاب** **5) النشر على صفحة لوحة النتائج** **6) تحديث حالة الإعجاب وعدد الإعجابات**
الاستثناءات	**إذا سبق للطالب أن أعجب بالمنشور: يتم إزالة الإعجاب بدلاً من إضافته**

جدول 11:وصف عملية عمل إعجاب

وصف عملية إنشاء حساب مستخدم

عملية إنشاء حساب مستخدم تتيح للطالب تسجيل حساب جديد على المنصة باستخدام بريده الإلكتروني الجامعي، حيث يُدخل بياناته في نموذج التسجيل ليتمكن من تسجيل الدخول لاحقًا.

اسم الوظيفة	إنشاء حساب مستخدم
الممثل	**الطالب، مسؤول النادي، مسؤول شؤون الطلبة.**
الوصف	**يقوم الطالب بإنشاء حساب شخصي على المنصة باستخدام البريد الإلكتروني الجامعي، ليتمكن من استخدام خدمات المنصة.**
المدخلات	➢ **الاسم الكامل** ➢ **البريد الإلكتروني الجامعي** ➢ **اسم الكلية التابع لها** ➢ **كلمة المرور** ➢ **تأكيد كلمة المرور**
المخرجات	1) **إنشاء حساب جديد وتخزينه في قاعدة البيانات.** 2) **إعادة توجيه المستخدم إلى صفحة تسجيل الدخول أو صفحته الشخصية.** 3) **رسالة "تم إنشاء الحساب بنجاح".**
الإجراءات	1) **الطالب يفتح واجهة "إنشاء حساب".** 2) **يملأ النموذج بالبيانات المطلوبة.** 3) **يضغط على زر "إنشاء حساب".** 4) **يتم التحقق من صحة البيانات (صيغة البريد، تطابق كلمتي المرور...).** 5) **يتم إرسال البيانات إلى الخادم.**

	6) يقوم الخادم بإنشاء الحساب وتخزينه في قاعدة البيانات. 7) يتم عرض رسالة النجاح وتوجيه المستخدم
الاستثناءات	**إذا كان البريد غير جامعي.** **إذا كانت كلمة المرور غير متطابقة.** **إذا كان البريد مستخدم مسبقًا.**

جدول 12:وصف عملية إنشاء حساب مستخدم

وصف عملية تسجيل الدخول

عملية تسجيل الدخول تتيح للمستخدم الوصول إلى حسابه على المنصة من خلال إدخال بريده الإلكتروني الجامعي وكلمة المرور.

اسم الوظيفة	تسجيل الدخول
الممثل	**الطالب ، مسؤول النادي ، مسؤول النظام .**
الوصف	**يتيح هذا الإجراء للمستخدمين المسجلين الدخول إلى حساباتهم لاستخدام وظائف النظام المختلفة حسب الصلاحيات.**
المدخلات	➢ **البريد الإلكتروني الجامعي** ➢ **كلمة المرور**
المخرجات	1) **الدخول إلى النظام بنجاح** 2) **الدخول للصفحة الرئيسية**
الإجراءات	1) **فتح واجهة تسجيل الدخول.** 2) **ادخال البريد الإلكتروني الجامعي وكلمة المرور.** 3) **الضغط على زر "تسجيل الدخول".** 4) **يتم توجيه المستخدم إلى صفحته اذا كانت المعلومات صحيحة**
الاستثناءات	➢ **إذا كانت البيانات غير صحيحة** ➢ **إذا لم يكن هناك حساب بهذا البريد**

جدول 13: وصف عملية تسجيل الدخول

2.6 ملخص الفصل

في هذا الفصل، تم تحديد متطلبات النظام بشكل دقيق مع توضيح العمليات الأساسية التي يقوم بها النظام باستخدام أشكال توضيحية تسهل فهم وتحليل آلية عمله. استندت هذه الأشكال إلى معلومات تم جمعها من خلال تواصل مباشر مع الجهات المستفيدة لضمان تلبية احتياجاتهم بشكل فعّال.

الفصل الثالث: تصميم النظام

3.1 المقدمة

في هذا الفصل، سيتم التركيز على تصميم وتنفيذ قواعد البيانات التي تعد الأساس لأي نظام معلوماتي. قواعد البيانات تساهم في تنظيم البيانات بشكل منهجي وسهل الوصول إليها، مما يتيح إمكانية تخزين واسترجاع البيانات بكفاءة عالية. سيتم تسليط الضوء على هيكلية الجداول والعلاقات بينها، بالإضافة إلى كيفية تطبيق مفاهيم الفهرسة والسلامة لضمان حماية البيانات وضمان جودتها

3.2 الهيكلية العامة للنظام

يتكون النظام من مجموعة متكاملة من الأجزاء التي تعمل بتناسق لتقديم وظائفه بشكل فعّال. هذه الأجزاء تم تصميمها لضمان سهولة التفاعل بين المستخدمين والنظام مع الحفاظ على الأداء العالي والكفاءة. يمكن تقسيم الهيكلية إلى جزأين رئيسيين:

1) تطبيق الويب:(Web Application)

يُعد تطبيق الويب الركيزة الأساسية للنظام، حيث يوفر واجهة تفاعلية تُسهل على جميع الأطراف (مثل الطلاب، مسؤولي الأندية، وإدارة النظام) التفاعل مع وظائف الموقع بطريقة مرنة وسهلة الاستخدام. تم تصميم التطبيق باستخدام تقنيات حديثة لضمان تجربة مستخدم مميزة وسريعة الاستجابة، بما يعكس احتياجات المستفيدين بشكل فعّال.

2) قاعدة البيانات:(Database)

تُعتبر قاعدة البيانات الجزء الخلفي الذي يدعم تشغيل النظام، حيث تحتوي على مجموعة من الجداول التي تخزن كافة البيانات المتعلقة بالمستخدمين، الأندية، الفعاليات، العضويات، والملاحظات. تم تصميم قاعدة البيانات لتكون منظمة ومتناسقة، مما يسهل عمليات البحث والاسترجاع ومعالجة البيانات بشكل دقيق وسريع. كما أن قاعدة البيانات تتسم بالمرونة لتلبية أي احتياجات مستقبلية أو توسيع للنظام.

3.3 هيكلية بناء النظام:

يعتمد النظام على نمط التصميم الشائع Model View Controller (MVC)، وهو نمط تصميم يهدف إلى فصل المكونات المنطقية للنظام عن بعضها البعض. يُعد هذا النمط من أفضل الأساليب لتطوير الأنظمة التي تحتاج إلى قابلية الصيانة والتوسع.

مكونات نمط:MVC

1) Model

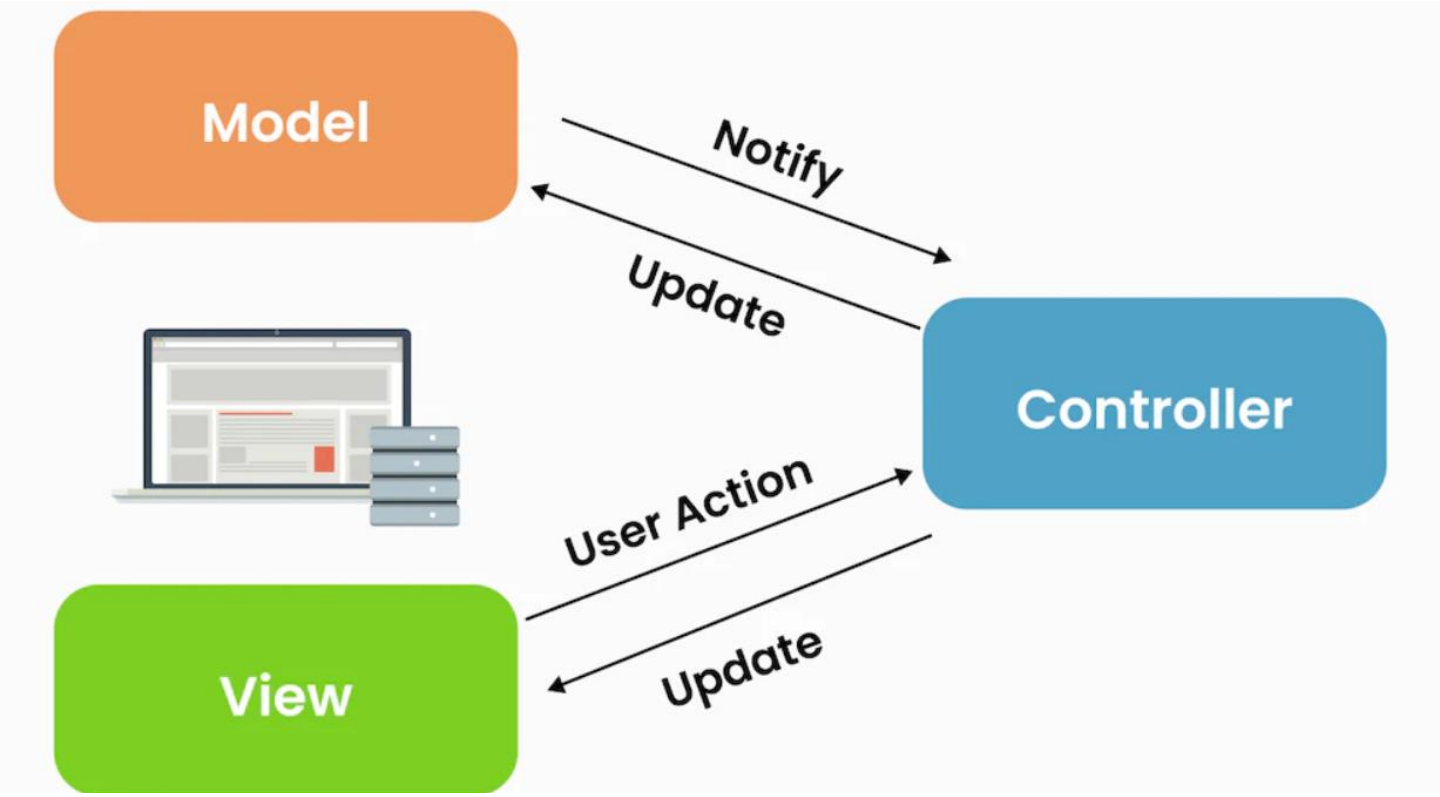
يمثل منطق البيانات والعمليات التي تتعلق بها، ويُعتبر الجزء المسؤول عن التعامل مع قاعدة البيانات والبيانات الأساسية للنظام.

2) View

يمثل الواجهة الأمامية للنظام أو واجهة المستخدم الرسومية(GUI) ، وهو المسؤول عن عرض البيانات للمستخدم بطريقة سهلة وبصرية.

3) Controller

يعمل كوسيط بين الـ Modelوالـ View يقوم بمعالجة الطلبات القادمة من المستخدم، والتعامل مع البيانات المطلوبة من قاعدة البيانات، ومن ثم عرض النتائج في الواجهة الأمامية.



شكل 3Model-View-Controller

3.4 نظام قواعد البيانات المعتمد

في مشروعنا، تم الاعتماد على قاعدة بيانات MongoDB، وهي قاعدة بيانات من نوع NoSQL تخزن البيانات بصيغة مستنداتJSON ، مما يمنحها مرونة كبيرة في التعامل مع البيانات غير المهيكلة والقابلة للتغيير. تلعب قاعدة البيانات دورًا رئيسيًا في تنظيم وتخزين جميع المعلومات المتعلقة بالمستخدمين، الأندية، المنشورات، طلبات الانضمام، والفعاليات المختلفة. يسمح هذا النظام بتسجيل بيانات الطلاب والمديرين، وتتبع عضويتهم في الأندية، بالإضافة إلى حفظ تفاصيل المنشورات التي تصدرها الأندية وتنظيم طلبات الانضمام وحالتها. كما تُستخدم قاعدة البيانات لتخزين معلومات المسابقات والنتائج المرتبطة بها، مما يسهل إدارة المنافسات بين الأندية. اختيار MongoDB جاء ليتناسب مع طبيعة المشروع التي تتطلب سرعة في التطوير ومرونة في تعديل هيكل البيانات مع استمرار نمو النظام وتطوره.

أيهما أفضل: قواعد البيانات العلائقية أم قواعد البيانات غير العلائقية؟

SQL	NoSQL
علاقات بيانات محددة بوضوح	لا علاقة؛ البيانات الموجودة في قاعدة بياناتنا مقترنة بشكل فضفا ض
يتم تخزين البيانات في الجداول	البيانات المخزنة في المستندات والرسوم البيانية، كما يسمى القيمة الرئيسية
مخطط محدد	مخطط ديناميكي، بيانات غير مرتبة
يفضل في حالة العمليات متعددة الخطوط	يفضل عندما تكون سرعة الحصول على البيانات مهمة
قابلة للتطوير عموديا	قابلة للتطوير أفقيا

جدول 14: الفرق بين قواعد البيانات العلائقية وغير العلائقية

مزايا قواعد بيانات NoSQL في مشروعنا:

في مشروعنا، تم اختيار قاعدة بيانات **NoSQL (MongoDB)** بدلاً من قواعد البيانات العلائقية (SQL) لعدة أسباب تتناسب مع طبيعة البيانات ومتطلبات النظام. تعتمد قواعد بيانات NoSQL على تخزين البيانات بشكل غير مرتب ومنظم في مستندات أو رسوم بيانية، مما يجعلها مناسبة للبيانات التي تكون علاقاتها فضفاضة أو غير محددة بوضوح، كما هو الحال في بيانات الأندية، المستخدمين، والمنشورات التي قد تختلف من سجل لآخر.

بالإضافة إلى ذلك، تتميز قواعد بيانات NoSQL بمخطط ديناميكي يسمح بإضافة حقول جديدة أو تعديل هيكل البيانات بسهولة دون الحاجة لتغيير قاعدة البيانات بالكامل، وهذا يتيح سرعة في التطوير والتكيف مع التغييرات المستقبلية في المشروع. كما توفر NoSQL قابلية توسع أفقية عالية، مما يعني إمكانية زيادة عدد الخوادم لتعزيز الأداء بدلاً من الاعتماد فقط على تحسين مواصفات خادم واحد كما هو الحال في قواعد البيانات العلائقية.

أيضًا، في مشروعنا كانت سرعة الحصول على البيانات مهمة جدًا خصوصًا عند تحميل المنشورات، عرض بيانات الأندية، ومعالجة طلبات الانضمام، الأمر الذي يجعل NoSQL خيارًا أفضل مقارنة بـ SQL التي تتفوق في العمليات المعقدة متعددة الخطوط والتي تتطلب علاقات قوية بين الجداول.

بالتالي، اختيار MongoDB كقاعدة بيانات NoSQL ساعد في تحقيق مرونة أكبر في تصميم البيانات، سهولة التوسع، وتحسين سرعة الاستجابة، ما ساهم في بناء نظام متكامل وفعال لإدارة الأندية الجامعية ومنشوراتها.

3.5 وصف لقاعدة البيانات

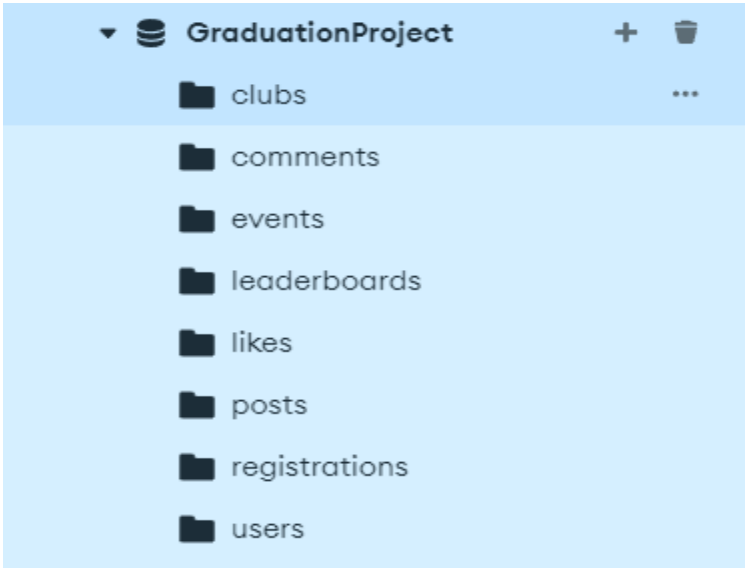
فيما يلي العديد من ال **Collections** المرتبطة بالنظام من خلال قاعدة البيانات.

اسم المجموعة	اسم المجموعة في قاعدة البيانات
مستخدم	Users
النادي	Clubs
الفعالية	Events
المنشور	Posts
التعليق	Club
الإعجاب	Likes

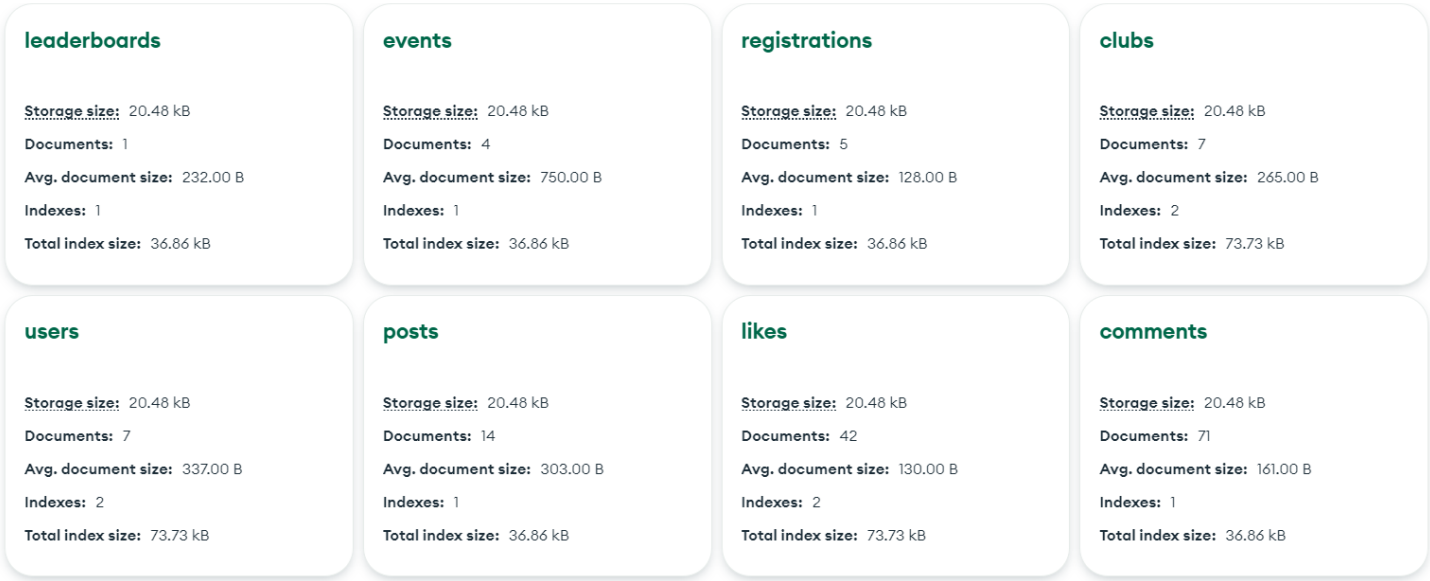
لوحة المتصدرين	Leaderboards
التسجيل	Registrations

جدول 15:وصف لقاعدة البيانات

شكل الجداول في قاعدة البيانات



شكل 4 : شكل الجداول في قاعدة البيانات



شكل 5 : توضيح للجداول في قاعدة البيانات

نظرة عامة على جداول قاعدة البيانات

تتكون قاعدة بيانات المشروع من عدة جداول تمثل العناصر الأساسية لنظام المنصة. كل جدول يحتوي على نوع مختلف من البيانات، ويؤدي دورًا مهمًا في تنظيم وتخزين معلومات المستخدمين والتفاعلات داخل المنصة.

- **جدول المستخدمين (Users):** يحتوي على معلومات الحسابات المسجلة في النظام مثل الاسم، البريد الإلكتروني، والصلاحيات.
- **جدول الأندية (Clubs):** يُستخدم لحفظ بيانات الأندية الجامعية، مثل اسم النادي، وصفه، والمسؤولين عنه.
- **جدول الفعاليات (Events):** يُسجل تفاصيل الفعاليات التي تنظمها الأندية أو الجهات داخل الجامعة.
- **جدول التسجيلات (Registrations):** يربط بين المستخدمين والفعاليات أو الأندية، حيث يُستخدم لتخزين عمليات التسجيل والمشاركة.
- **جدول المشاركات (Posts):** يحتوي على المنشورات التي يتم مشاركتها من قبل المستخدمين أو الأندية داخل المنصة.
- **جدول التعليقات (Comments):** يُستخدم لحفظ التعليقات التي يضيفها المستخدمون على المنشورات، مما يعزز التفاعل والنقاش.
- **جدول الإعجابات (Likes):** يخزن تفاعلات الإعجاب مع المشاركات أو التعليقات من قِبل المستخدمين.
- **جدول لوحات الشرف أو الترتيب (Leaderboards):** يُظهر المستخدمين أو الأندية الأكثر تميزًا أو مشاركةً، بناءً على معايير محددة مثل النقاط أو عدد المشاركات.

3.6 وصف جداول قاعدة البيانات

جدول المستخدمين(Users Collection)

هذا الجدول يحتوي على معلومات المستخدمين المسجلين في النظام، ويُستخدم لتخزين البيانات الشخصية والوظيفية لكل مستخدم.

```
_id: ObjectId('682a34694abd3410d3db1d40')                                  ObjectId
name : "Hi, Im Ahmad"                                                      String
email : "201124@ppu.edu.ps"                                                String
emailVerified : true                                                       Boolean
password : "$2b$12$.bkKG13WB7J5vNRWi/PjT.oRBMdzQ5Ry/ChEyjy/c5G0UApl8/1VO"  String
passwordResetVerified : false                                              Boolean
profilePicture : "default.jpg"                                             String
role : "student"                                                           String
college : "Engineering"                                                    String
createdAt : 2025-05-18T19:26:33.865+00:00                                  Date
updatedAt : 2025-05-18T19:28:53.378+00:00                                  Date
__v : 0                                                                    Int32
```

وهنا توضيح لكل حقل من حقول جدول المستخدم ...

الحقل	الوصف	نوع البيانات
_id	المعرف الفريد للمستخدم، يتم إنشاؤه تلقائيًا منMongoDB	ObjectId
name	اسم المستخدم المعروض في النظام.	String

email	البريد الإلكتروني للمستخدم، يُستخدم لتسجيل الدخول والتواصل.	String
emailVerified	حالة تأكيد البريد الإلكتروني (هل تم التحقق منه؟).	Boolean
password	كلمة المرور المشفرة الخاصة بالمستخدم.	String
passwordResetVerified	حالة تحقق تغيير كلمة المرور.	Boolean
profilePicture	اسم ملف الصورة الافتراضية أو المرفوعة للمستخدم.	String
role	دور المستخدم في النظام مثل الطالب أو مسؤول النظام	String
college	اسم الكلية التي ينتمي إليها المستخدم.	String
createdAt	تاريخ ووقت إنشاء الحساب.	Date
updatedAt	آخر وقت تم فيه تعديل بيانات المستخدم.	Date

جدول 16: وصف حقول جدول المستخدم

جدول الأندية (Clubs Collection)

هذا الجدول يُستخدم لتخزين معلومات الأندية الطلابية الموجودة داخل الجامعة. يحتوي على بيانات وصفية لكل نادٍ، مثل اسمه وصوره والكلية التابعة له.

```
1  _id: ObjectId('68191d5755fddb4f1887f972')                                    ObjectId
2  name : "Coding Club"                                                         String
3  description : "A of coding with some tea"                                    String
4  profilePicture : "https://res.cloudinary.com/dto9h9akz/image/upload/v1746476374/clubs/pr"   String
5  coverPicture : "https://res.cloudinary.com/dto9h9akz/image/upload/v1746476374/clubs/co"     String
6  college : "Photographers "                                                   String
7  createdAt : 2025-05-05T20:19:35.675+00:00                                    Date
8  updatedAt : 2025-05-05T20:19:35.675+00:00                                    Date
9  __v : 0                                                                      Int32
```

وهنا توضيح لكل حقل من حقول جدول الأندية ...

جدول الفعاليات(Events Collection)

هذا الجدول يُستخدم لتخزين معلومات الفعاليات التي تنظمها الأندية الطلابية، مثل الاجتماعات أو الورش أو المسابقات.

```
_id: ObjectId('68169a038cedf652f8d19cfe')                                    ObjectId
title : "Club Meeting"                                                       String
description : "Monthly club meeting"                                         String
date : 2025-06-01T14:00:00.000+00:00                                         Date
startTime : "14:00"                                                          String
endTime : "16:00"                                                            String
location : "Room 101"                                                        String
image : "https://res.cloudinary.com/dto9h9akz/image/upload/v1746476771/events/i"  String
club : 680965c238f46f548fee39db                                              ObjectId
author : 680d1de2ba9ea11d9af2027f                                            ObjectId
comments : Array (2)                                                         Array
  0: 682f3a12dc8ce84e0187268d                                                ObjectId
  1: 682f3a1cdc8ce84e01872695                                                ObjectId
likes : Array (2)                                                            Array
  0: 682f3aebdc8ce84e018726a0                                                ObjectId
  1: 682f3b36dc8ce84e018726d3                                                ObjectId
createdAt : 2025-05-03T22:34:43.466+00:00                                    Date
updatedAt : 2025-05-22T14:56:54.927+00:00                                    Date
__v : 4                                                                      Int32
```

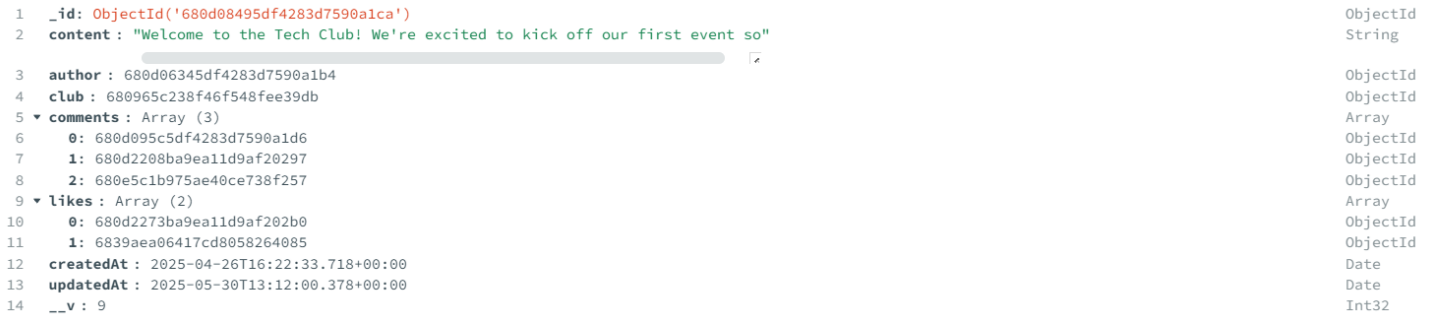
description		وصف مختصر لمحتوى الفعالية.	String
date		تاريخ انعقاد الفعالية.	Date
startTime		وقت بداية الفعالية (بنمط نصي).	String
endTime		وقت نهاية الفعالية (بنمط نصي).	String
location		مثلاً: Room 101).	String
image		رابط لصورة مرفقة بالفعالية.	String
club		معرف النادي المنظم للفعالية (يرتبط بجدول الأندية).	ObjectId
author		معرف المستخدم الذي أضاف الفعالية (يرتبط بجدول المستخدمين).	ObjectId
comments		قائمة بمعرفات التعليقات المرتبطة بهذه الفعالية.	Array<ObjectId>
likes		قائمة بمعرفات الإعجابات المرتبطة بهذه الفعالية.	Array<ObjectId>
createdAt		تاريخ ووقت إنشاء الفعالية.	Date
updatedAt		آخر تاريخ تم فيه تعديل بيانات النادي.	Date

جدول 18:وصف حقول جدول الفعاليات

وهنا توضيح لكل حقل من حقول جدول الفعاليات ...

جدول المنشورات(Posts Collection)

هذا الجدول يُستخدم لتخزين المنشورات التي ينشرها الأعضاء أو مسؤولو الأندية داخل النظام، مثل الإعلانات أو الأخبار



وهنا توضيح لكل حقل من حقول جدول المنشورات ...

الحقل	الوصف	نوع البيانات
_id	المعرف الفريد للمنشور (يتم توليده تلقائيًا من MongoDB).	ObjectId
content	محتوى المنشور النصي (مثلاً: ترحيب أو إعلان فعالية).	String
author	معرف المستخدم الذي كتب المنشور (يرتبط بجدول المستخدمين).	ObjectId
club	معرف النادي الذي نُشر فيه هذا المنشور (يرتبط بجدول الأندية).	ObjectId
comments	قائمة بمعرفات التعليقات المرتبطة بهذا المنشور.	Array<ObjectId>
likes	قائمة بمعرفات المستخدمين الذين أعجبوا بالمنشور.	Array<ObjectId>
createdAt	تاريخ ووقت إنشاء المنشور.	Date
updatedAt	تاريخ آخر تعديل للمنشور.	Date

جدول 19:وصف حقول جدول المنشوررات

تعليق على منشور(Post Comment)

تعليق يُضاف من قبل المستخدمين على منشورات الأندية بهدف التفاعل أو النقاش. يحتوي على نص التعليق، ومُعرف كاتب التعليق، ويرتبط بمنشور معين. نوع هذا التعليق يُحدد بالحقل role بـ"post".

```
_id: ObjectId('680d2208ba9ea11d9af20297')                    ObjectId
content : "Hi Every One"                                     String
author : 680d06345df4283d7590a1b4                            ObjectId
role : "post"                                                String
post : 680d08495df4283d7590a1ca                              ObjectId
event : null                                                 Null
createdAt : 2025-04-26T18:12:24.688+00:00                    Date
updatedAt : 2025-04-26T18:12:24.688+00:00                    Date
__v : 0                                                      Int32
```

تعليق على حدث(Event Comment)

تعليق يُستخدم للتفاعل مع الأحداث المنشورة داخل النظام، مثل الاجتماعات أو الأنشطة. يتضمن نصًا ومُعرف الكاتب، ويرتبط بحدث معين. نوع التعليق يُحدد بالحقل role بـ"event" .

```
_id: ObjectId('680d1c36ba9ea11d9af2026a')                    ObjectId
content : "Looking forward to the event! Thanks for sharing."  String
author : 680cfb5602b7e870084682eb                            ObjectId
role : "event"                                               String
post : null                                                  Null
event : 680d1b874af8340501d8e8aa                             ObjectId
createdAt : 2025-04-26T17:47:34.868+00:00                    Date
updatedAt : 2025-04-26T17:47:34.868+00:00                    Date
__v : 0                                                      Int32
```

الحقل	الوصف	نوع البيانات
_id	المعرف الفريد للتعليق.	ObjectId
content	محتوى التعليق النصي.	String
author	معرف المستخدم الذي كتب التعليق (يرتبط بجدول المستخدمين).	ObjectId
role	نوع التعليق.: لتحديد السياق إما"post"أو "event"	String
post	اذا كانت ال (Role) = Post , Event=Null	ObjectId or null
event	اذا كانت ال (Role) =Event , Post=Null	ObjectId or null
createdAt	تاريخ إنشاء التعليق.	Date
updatedAt	تاريخ التعديل الأخير للتعليق.	Date

وهنا توضيح لكل حقل من حقول جدول التعليق على حدث ... *جدول 20: وصف حقول جدول الأحداث*

جدول الإعجابات(Likes Collection)

يُستخدم هذا الجدول لتخزين معلومات الإعجابات التي يضعها المستخدمون على محتويات متعددة داخل النظام، مثل التعليقات أو الفعاليات. يحتوي كل إعجاب على مرجع للمستخدم الذي وضعه، ونوع الهدف (تعليق أو فعالية)، ومعرف العنصر الذي تم وضع الإعجاب عليه.

```
_id: ObjectId('680a5aadbb8ffe28b8814fc7')                    ObjectId
user : 6809652738f46f548fee39d6                              ObjectId
targetType : "event"                                         String
targetId : 6809673d38f46f548fee39f4                          ObjectId
createdAt : 2025-04-24T15:37:17.235+00:00                    Date
updatedAt : 2025-04-24T15:37:17.235+00:00                    Date
__v : 0                                                      Int32
```

```
_id: ObjectId('6803f391bf7a2dc6664188d1')                    ObjectId
user : 6803e0e4f29e66e0dd0e1a9e                              ObjectId
targetType : "post"                                          String
targetId : 6803e72831390ce29566faef                          ObjectId
createdAt : 2025-04-19T19:03:45.226+00:00                    Date
updatedAt : 2025-04-19T19:03:45.226+00:00                    Date
__v : 0                                                      Int32
```

وهنا توضيح لكل حقل من حقول جدول الإعجابات ...

الحقل	الوصف	نوع البيانات
_id	المعرف الفريد للإعجاب ، يتم إنشاؤه تلقائيًا من قبلMongoDB.	ObjectId
user	معرف المستخدم الذي قام بعمل الإعجاب .	ObjectId
targetType	نوع الهدف الذي تم وضع الإعجاب عليه ("event"أو "comment")	String
targetId	معرف العنصر الهدف (تعليق أو فعالية).	ObjectId
createdAt	تاريخ ووقت إنشاء الإعجاب.	Date
updatedAt	آخر تاريخ تم فيه تعديل الإعجاب.	Date

جدول 21 :وصف حقول جدول الإعجابات

جدول المتصدرين(Leaderboard Collection)

يُستخدم هذا الجدول لتخزين قائمة المتصدرين في فعالية أو مناسبة معينة. يحتوي على أسماء المتصدرين الثلاثة الأوائل عبر معرفاتهم، بالإضافة إلى وصف وتفاصيل متعلقة بالفعالية المرتبطة بهم.

```
_id: ObjectId('681f754b9665f91dc408ce52')                    ObjectId
name : "Monthly Leaderboard"                                 String
description : "Top performers for the month"                 String
top1 : 6814fe77d18d8f83dfaca45a                              ObjectId
top2 : 680d1de2ba9ea11d9af2027f                              ObjectId
top3 : 680d06345df4283d7590a1b4                              ObjectId
event : 68191e713727e8a81e18a488                             ObjectId
date : 2025-05-10T15:48:27.535+00:00                         Date
createdAt : 2025-05-10T15:48:27.539+00:00                    Date
updatedAt : 2025-05-10T15:48:27.539+00:00                    Date
__v : 0                                                      Int32
```

وهنا توضيح لكل حقل من حقول جدول المتصدرين ...

الحقل	الوصف	نوع البيانات
_id	المعرف الفريد لسجل المتصدرين، يتم إنشاؤه تلقائيًا من MongoDB.	ObjectId
name	اسم القائمة كما يظهر في واجهة المستخدم .	String
description	وصف مختصر للقائمة يوضح الغرض منها أو تفاصيل إضافية.	String
top1	معرف المستخدم الحاصل على المركز الأول.	ObjectId
top2	معرف المستخدم الحاصل على المركز الثاني.	ObjectId
top3	معرف المستخدم الحاصل على المركز الثالث.	ObjectId
event	معرف الفعالية المرتبطة بهذه القائمة.	ObjectId
date	تاريخ إصدار القائمة أو وقت الاحتساب.	Date
createdAt	تاريخ ووقت إنشاء السجل.	Date
updatedAt	آخر تاريخ تم فيه تعديل السجل.	Date

جدول 22:وصف حقول جدول المتصدرين

جدول التسجيل في الأندية(Club Registrations Collection)

يُستخدم هذا الجدول لتخزين معلومات تسجيل الطلاب في الأندية الجامعية. يحتوي على بيانات الطالب، النادي، وحالة التسجيل.

```
_id: ObjectId('680d1f6dba9ea11d9af20288')                ObjectId
student : 680d1de2ba9ea11d9af2027f                       ObjectId
club : 680965c238f46f548fee39db                          ObjectId
status : "approved"                                      String
createdAt : 2025-04-26T18:01:17.420+00:00                Date
updatedAt : 2025-04-26T18:11:37.840+00:00                Date
__v : 0                                                  Int32
```

```
_id: ObjectId('68150356d18d8f83dfaca46d')                ObjectId
student : 6814fe77d18d8f83dfaca45a                       ObjectId
club : 680965c238f46f548fee39db                          ObjectId
status : "pending"                                       String
createdAt : 2025-05-02T17:39:34.691+00:00                Date
updatedAt : 2025-05-02T17:39:34.691+00:00                Date
__v : 0                                                  Int32
```

وهنا توضيح لكل حقل من حقول جدول التسجيل في الاندية ...

الحقل	الوصف	نوع البيانات
_id	المعرف الفريد للتسجيل، يتم إنشاؤه تلقائيًا من MongoDB.	ObjectId
student	معرف الطالب الذي قام بطلب التسجيل في النادي.	ObjectId
club	معرف النادي الذي يرغب الطالب بالانضمام إليه.	ObjectId
status	["pending" ،"approved" ،"rejected"]	String
createdAt	تاريخ ووقت إنشاء طلب التسجيل.	Date

جدول 23 : وصف جدول التسجيل في الأندية

الفصل الرابع : التطبيق

4.1 نظرة عامة

في هذا الفصل، سنناقش التقنيات المستخدمة في بناء الواجهة الخلفية(Back-end) ، والواجهة الأمامية(Front-end) ، وعمليات النشر(Deployment) ، بالإضافة إلى أدوات أخرى

Backend

Node.js و Express.js:

Node.js هو بيئة تشغيل تتيح استخدام JavaScript في بناء الخوادم، أما Express.js فهو إطار عمل يُستخدم لتنظيم البنية الخلفية للتطبيق، وتسهيل إدارة الطلبات وتوجيهها.

MongoDB:

قاعدة بيانات غير علائقية (NoSQL) تُستخدم لتخزين البيانات على شكل مستندات JSON، وتتميز بالمرونة والسرعة، مما يجعلها مناسبة لتطبيقات الويب الحديثة.

Cloudinary:

خدمة سحابية متخصصة في رفع وإدارة الصور والفيديوهات، وتُستخدم لتخزين الوسائط خارج الخادم وتوفير روابط جاهزة للاستخدام داخل التطبيق.

JWT (JSON Web Token):

طريقة تُستخدم للتحقق من هوية المستخدم عبر إرسال رمز مشفّر بين العميل والخادم، وتُستخدم غالبًا في أنظمة تسجيل الدخول لحماية البيانات ومنع الوصول غير المصرّح به.

Bcrypt:

مكتبة تُستخدم لتشفير كلمات المرور قبل تخزينها في قاعدة البيانات، مما يزيد من أمان النظام ويمنع كشف كلمات السر في حال تسريب البيانات.

express-validator:

أداة تُستخدم للتحقق من صحة البيانات التي يُدخلها المستخدم، مثل التأكد من تنسيق البريد الإلكتروني أو طول كلمة المرور، وذلك قبل معالجتها في الخادم.

Multer:

أداة وسيطة (middleware) تُستخدم لمعالجة الملفات المُرسلة من خلال النماذج، وتُعد ضرورية عند رفع صور أو مستندات من المستخدم إلى الخادم.

Nodemailer:

مكتبة تُستخدم لإرسال رسائل البريد الإلكتروني من الخادم، وتُفيد في إرسال رسائل التفعيل أو استرجاع كلمة المرور وغيرها من الإشعارات.

Morgan:

أداة تُستخدم لتسجيل جميع الطلبات التي تصل إلى الخادم أثناء التطوير، مما يساعد في تتبع الأخطاء وتحسين الأداء.

CORS:

مكتبة تُستخدم للسماح لتطبيقات الويب بالتواصل مع السيرفر حتى لو كان كل منهما على نطاق (domain) مختلف، وتُعد ضرورية لتجاوز سياسات المتصفح الأمنية.

Frontend

Figma:

أداة تصميم تُستخدم لإنشاء واجهات المستخدم بطريقة احترافية وتفاعلية، وتُفيد في تخطيط شكل الموقع أو التطبيق قبل البدء في البرمجة.

React.js:

مكتبة JavaScript خاصة بالواجهة الأمامية، تُستخدم لبناء واجهات مستخدم تعتمد على المكونات (Components)، وتُسهّل إدارة الواجهة وتحديثها بشكل سريع وفعّال.

Tailwind CSS:

إطار عمل لتنسيق وتصميم الواجهة الأمامية باستخدام أصناف جاهزة، يُسهّل التحكم بالمظهر ويوفر مرونة كبيرة في تخصيص الستايلات دون الحاجة لكتابة CSS من الصفر

Axios:

مكتبة تُستخدم لإرسال واستقبال الطلبات من وإلى واجهات برمجة التطبيقات (APIs)، وتُساعد في التواصل بين الواجهة الأمامية والخلفية.

Deploy & Tools

GitHub:

منصة لإدارة الأكواد والمشاريع البرمجية باستخدام نظام التحكم في الإصدارات(Git) ، وتُستخدم لحفظ الكود، تتبع التعديلات، والعمل الجماعي بين المطورين.

Postman:

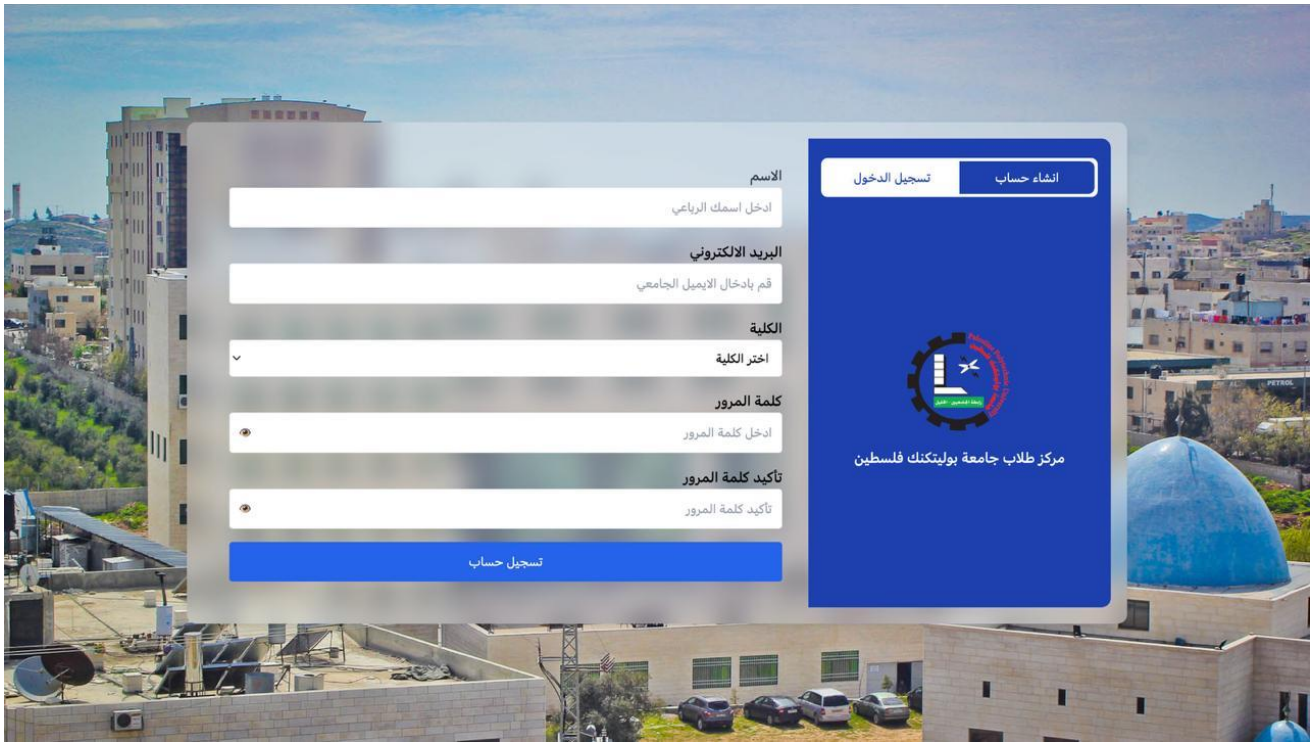
أداة تُستخدم لاختبار واجهات برمجة التطبيقات(APIs) ، تساعد في إرسال الطلبات ومعاينة الردود من السيرفر بشكل واضح ومنظم، مما يسهل عملية التحقق من صحة عمل الخوادم.

Render:

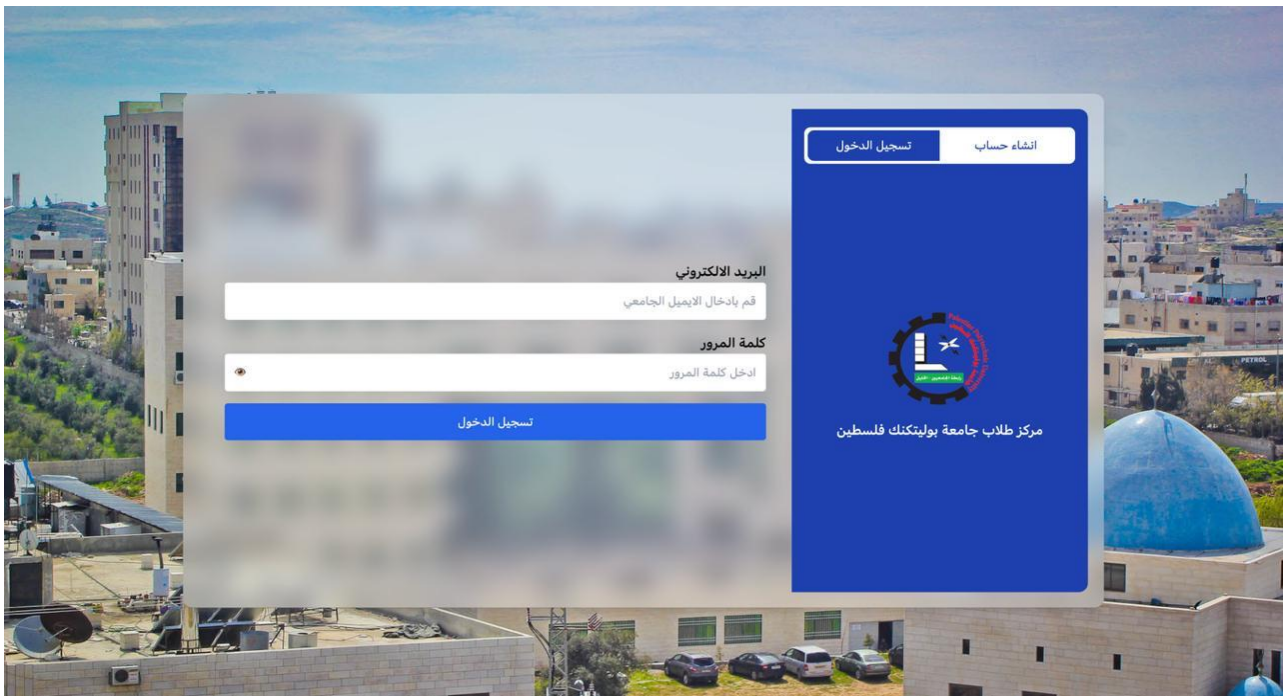
منصة سحابية تُستخدم لنشر (Deploy) التطبيقات والمواقع، وتُوفر بيئة جاهزة لتشغيل الخوادم وقواعد البيانات دون الحاجة لإعدادات معقدة.

4.2 واجهات المستخدم

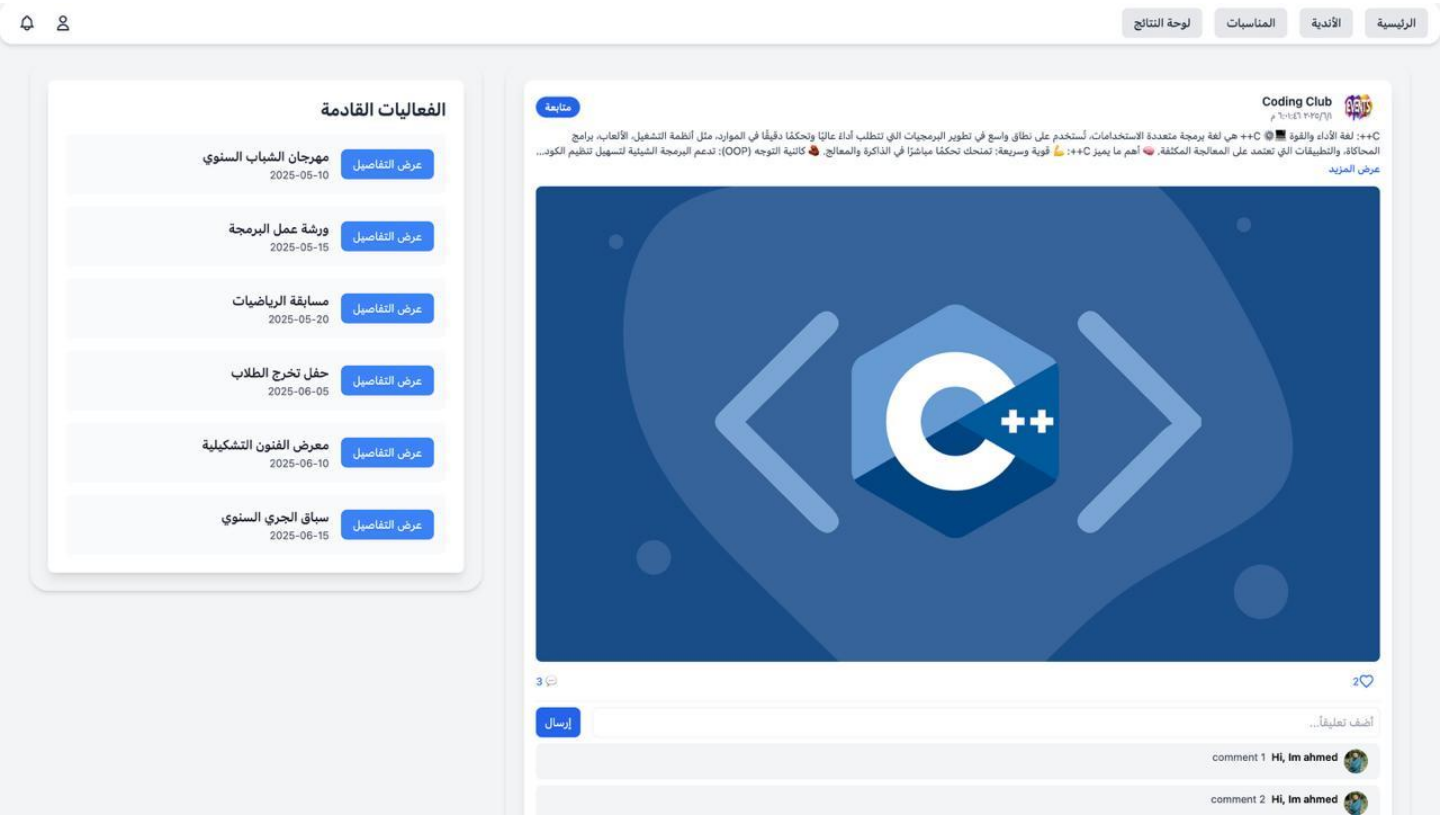
صفحة إنشاء حساب للموقع:



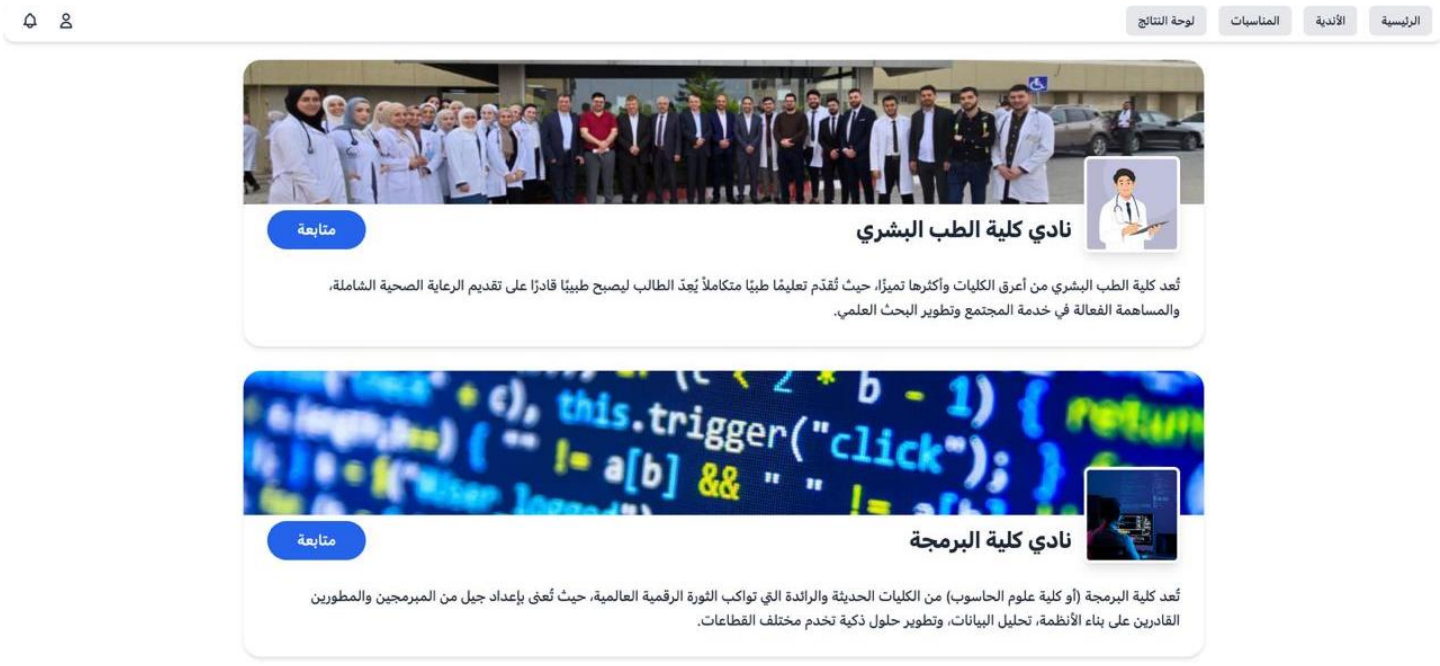
صفحة تسجيل الدخول:



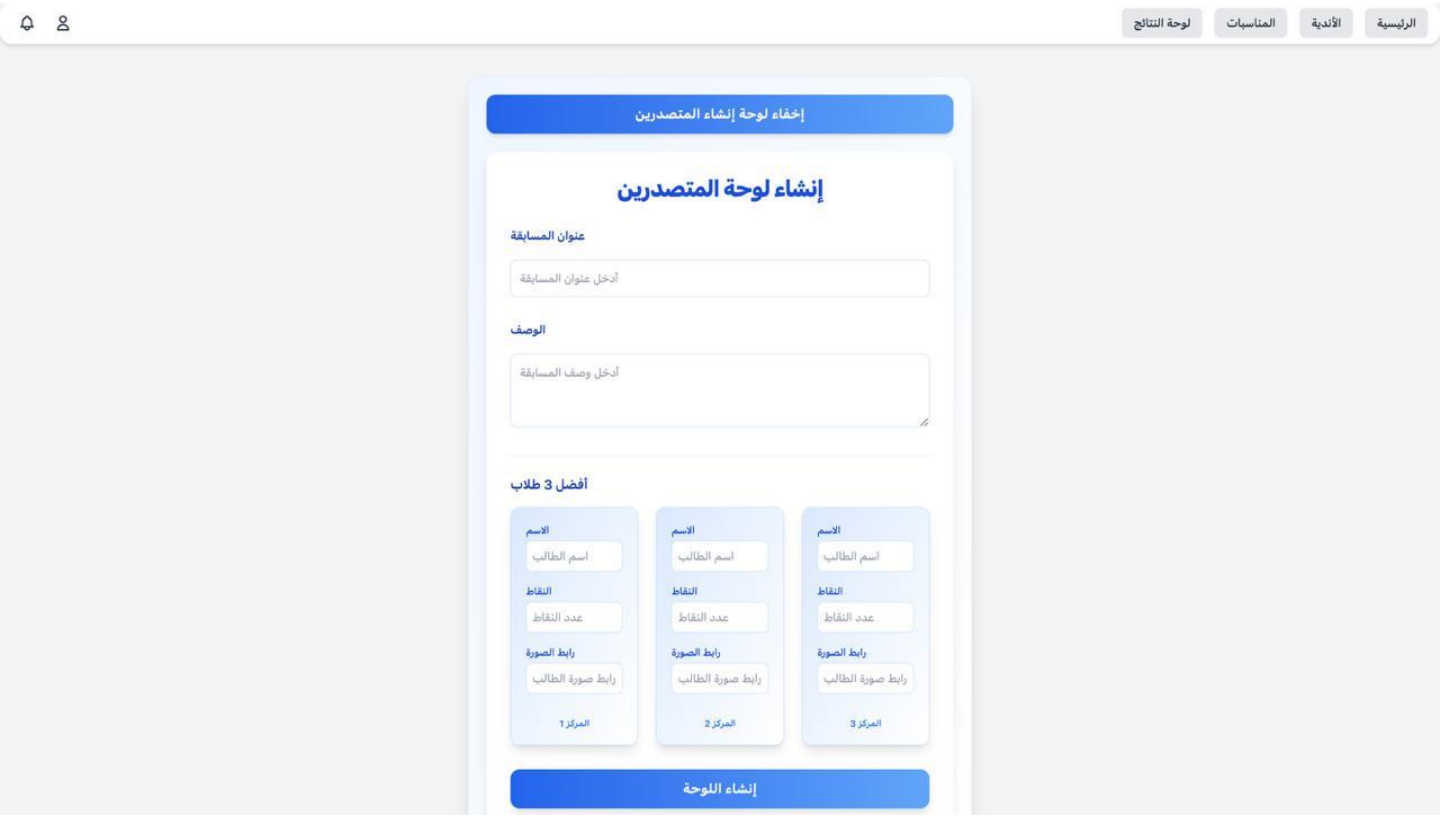
الصفحة الرئيسية:



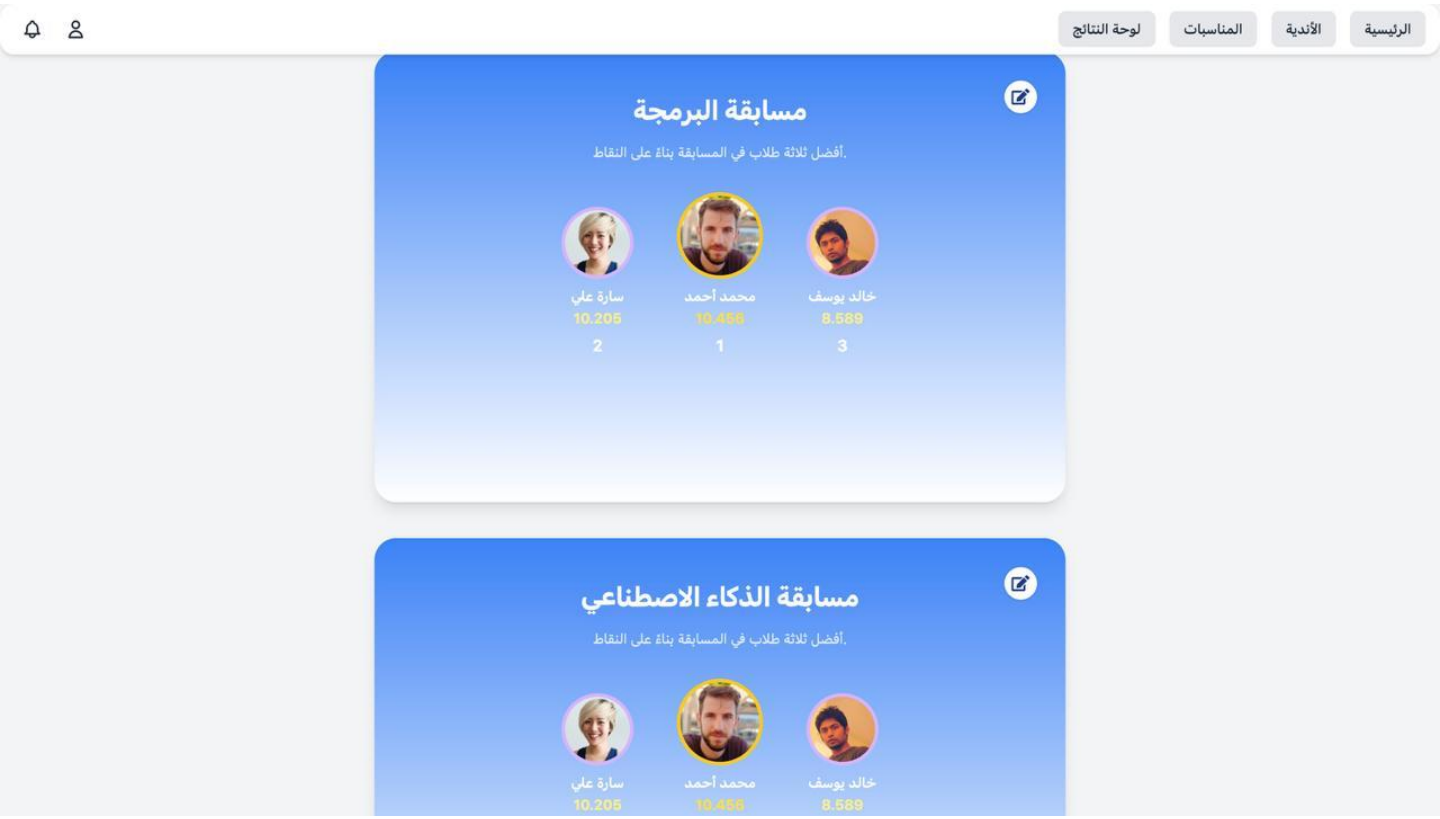
صفحة الأندية الموجودة على الموقع:



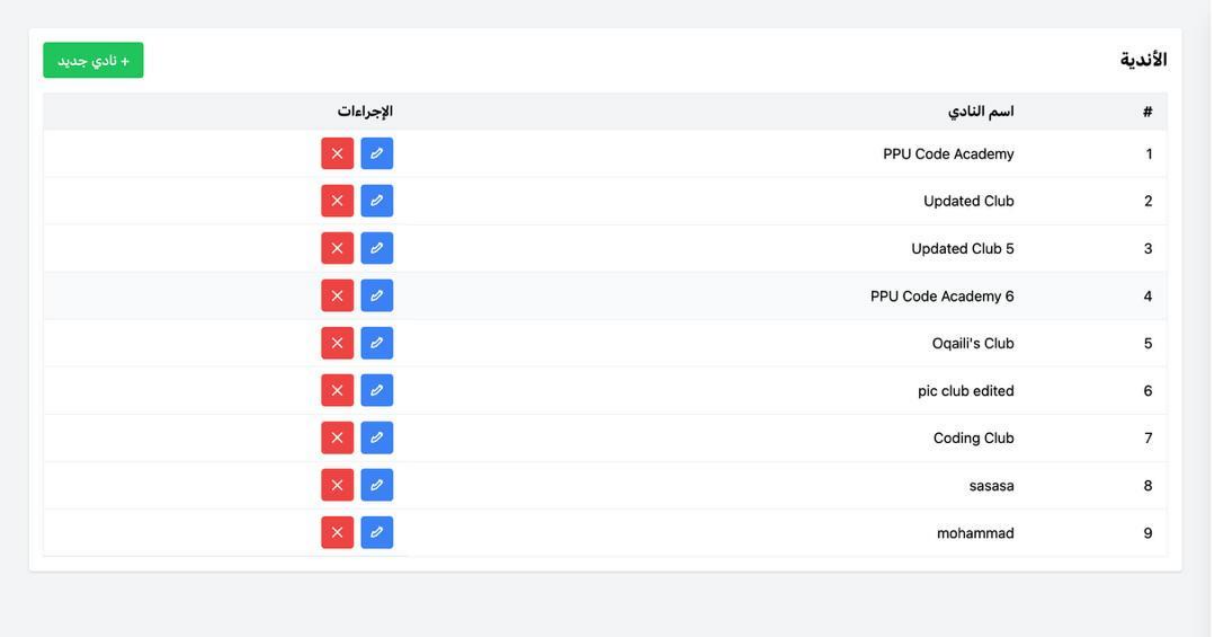
صفحة لإنشاء لوحة متصدرين للطلبة المتفوقين في مسابقة معينة:



صفحة لعرض جميع المتصدرين في المسابقات:



جزء من صفحات الآدمن :



الفصل الخامس: فحص النظام

في هذا الفصل قمنا بتوضيح كيفية التأكد من أن الموقع يعمل بالشكل المطلوب، وذلك من خلال تجربة جميع الصفحات والوظائف يدويًا، ومراقبة استجابة النظام في الحالات المختلفة. وقد قمنا بتوثيق بعض هذه التجارب بصور توضيحية.

5.1 التحقق من صحة البيانات(Validation)

من الأمور الأساسية التي تم التركيز عليها أثناء تطوير الموقع هي التأكد من صحة البيانات المدخلة من قِبل المستخدم. حيث تم تطبيق نظام تحقق (Validation)دقيق على جميع الحقول، لضمان إدخال بيانات صحيحة ومتوافقة مع الشروط المطلوبة.

وقد قمنا بتعريف شروط التحقق داخل **Schema** لكل نموذج، بحيث:

- يتم تخصيص كل حقل ليتناسب مع نوع البيانات المناسب (مثل النصوص، الأرقام، البريد الإلكتروني... إلخ).
- لا تُنفّذ العمليات إذا كانت البيانات غير صحيحة أو غير مكتملة.
- يتم منع إرسال النموذج في حال ترك الحقول المطلوبة فارغة.
- استخدمنا أدوات التحقق المضمنة في مكتبة Mongoose لتعريف خصائص كل حقل، مثل required, minlength, maxlength, match,وغيرها.

5.2 اختبار الجودة اليدوي (Testing Manual QA)

محاولة إنشاء حساب باستخدام بريد إلكتروني غير تابع لجامعةPPU

المتوقع:

أن لا يستطيع التسجيل في النظام

النتيجة الفعلية:

يجب أن يكون البريد الجامعي على شكل بريد جامعة البوليتكنك

محاولة إنشاء حساب وتفعيله بكود غير صحيح:

المتوقع:

أن لا يستطيع تفعيل البريد

النتيجة الفعلية: رمز التحقق غير صحيح أو منتهى الصلاحية

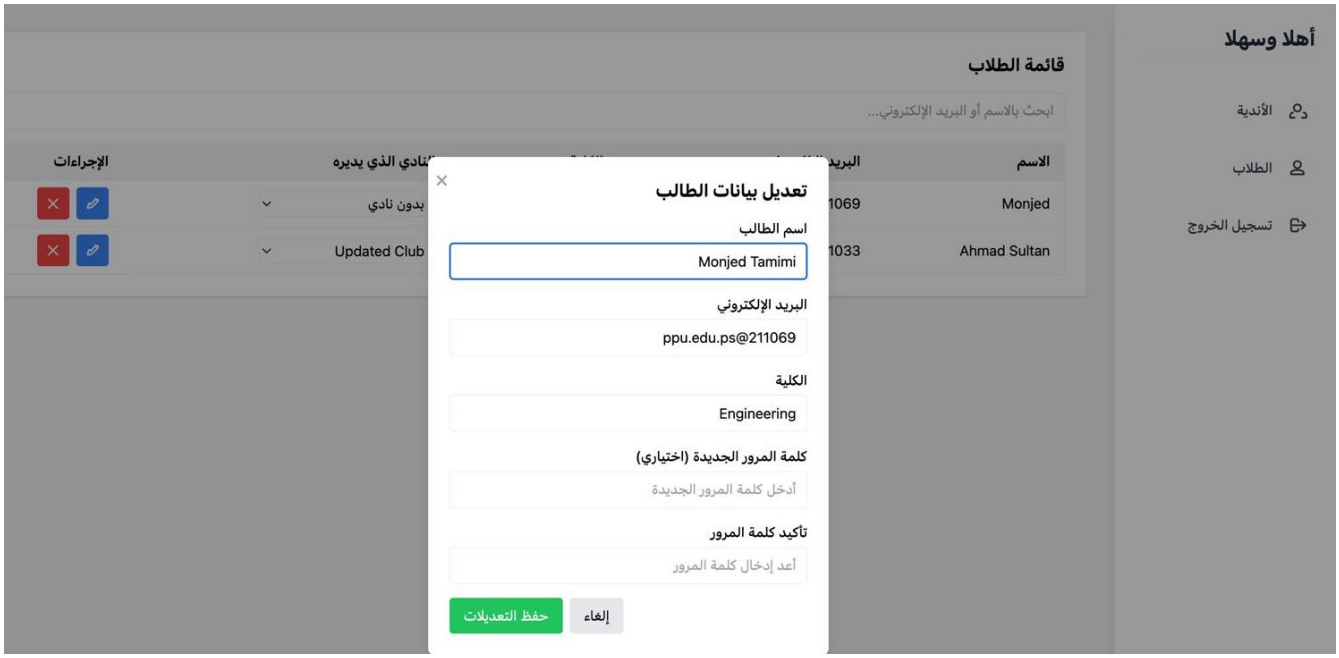
محاولة المسؤول لتغيير بيانات الطلاب

المتوقع:

أن يتم تحديث البيانات بنجاح

النتيجة الفعلية:

تم تحديث البيانات بنجاح



محاولة إنشاء حساب بكلمات مرور غير متطابقة

المتوقع:

أن لا يستطيع المستخدم تسجيل الدخول

النتيجة الفعلية:

كلمتا المرور غير متطابقات مع عدم القدرة على إنشاء حساب

5.3 اختبار نقاط النهاية

هو عملية اختبار تُستخدم للتحقق من سلامة ووظائف نقاط النهاية في واجهة برمجة التطبيقات .(API) تُمثل هذه النقاط الأماكن التي يتفاعل من خلالها المستخدم أو النظام مع الخادم، مثل إرسال طلب تسجيل دخول أو جلب بيانات من قاعدة البيانات. يتم خلال هذا النوع من الاختبار إرسال طلبات مختلفة مثل GET ، POST، PUT، DELETE إلى هذه النقاط وتحليل الاستجابة للتأكد من أنها صحيحة من حيث البيانات، الكود البرمجي للحالة(status code) ، والسرعة، وكذلك التعامل مع الحالات الخاطئة أو المدخلات غير المتوقعة. يساهم هذا الاختبار في ضمان تكامل النظام وتواصله بشكل فعال وآمن بين الواجهة الأمامية والخلفية.

اختبار نقطة النهاية: التسجيل(Sign Up)

الاختبار:

محاولة إنشاء حساب باستخدام بريد إلكتروني غير تابع لجامعة PPU.

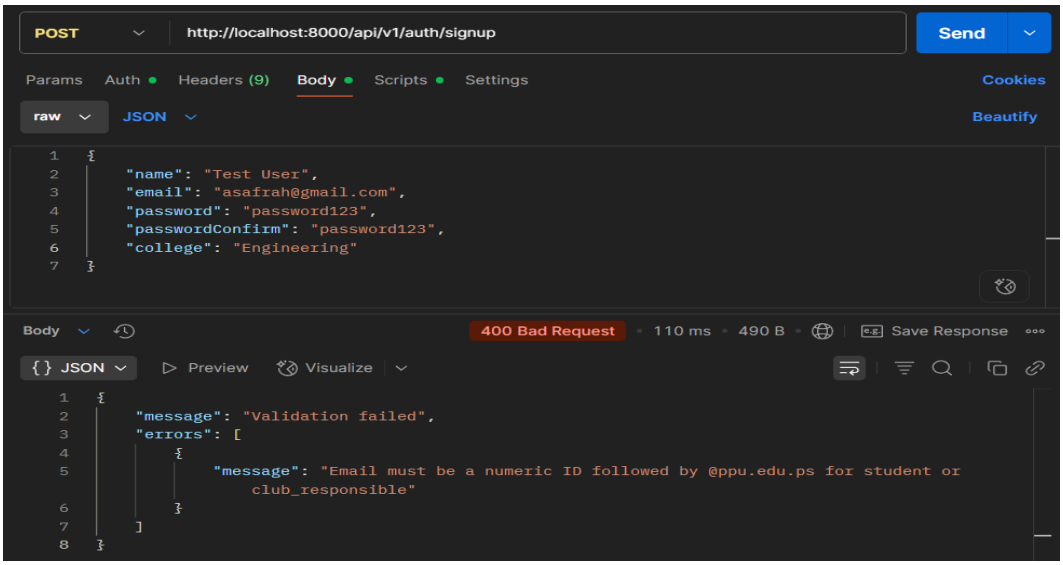
المتوقع:

رفض الطلب وإظهار رسالة توضح أن البريد يجب أن يكون بصيغة الرقم الجامعي متبوعًا بـ .@ppu.edu.ps

النتيجة الفعلية:

تم رفض الطلب بنجاح وظهرت رسالة:

Email must be a numeric ID followed by @ppu.edu.ps for student or club_responsible



اختبار نقطة النهاية: إنشاء نادي جديد (POST /api/v1/clubs)

الاختبار:

محاولة إنشاء نادي باستخدام حساب طالب أو مسؤول نادي (ليس لديه صلاحيات إدارية).

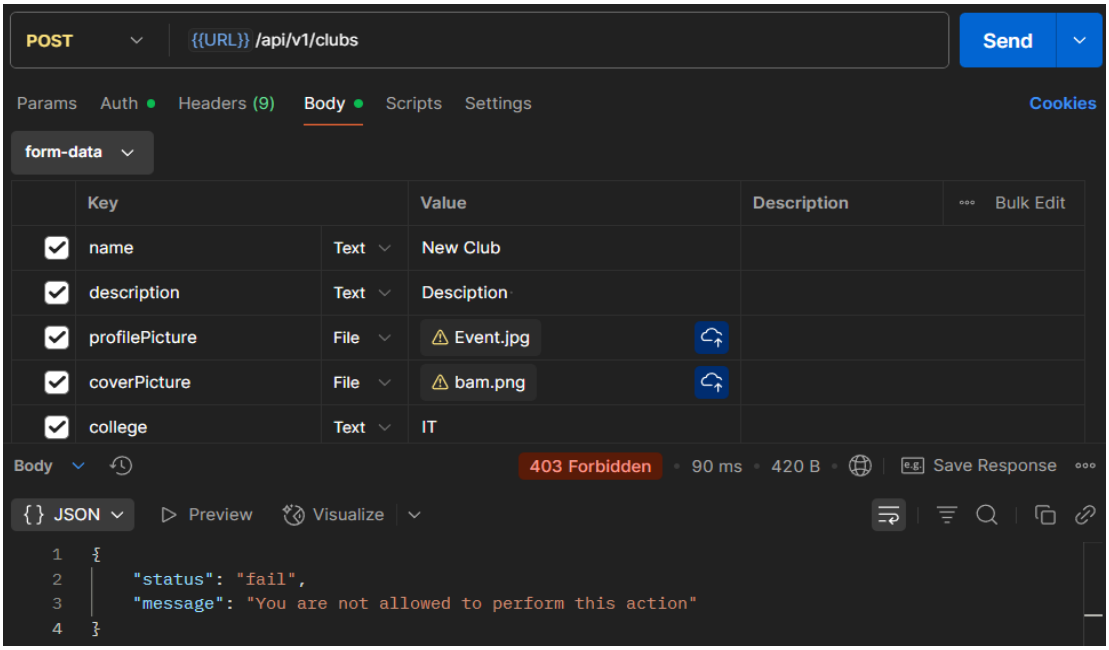
المتوقع:

رفض العملية برسالة توضح أن هذا المستخدم لا يمتلك صلاحية إنشاء نادي.

النتيجة الفعلية:

تم الرفض برسالة:

You are not allowed to perform this action



اختبار نقطة النهاية: إنشاء منشور جديد(POST /api/v1/posts)

الاختبار:

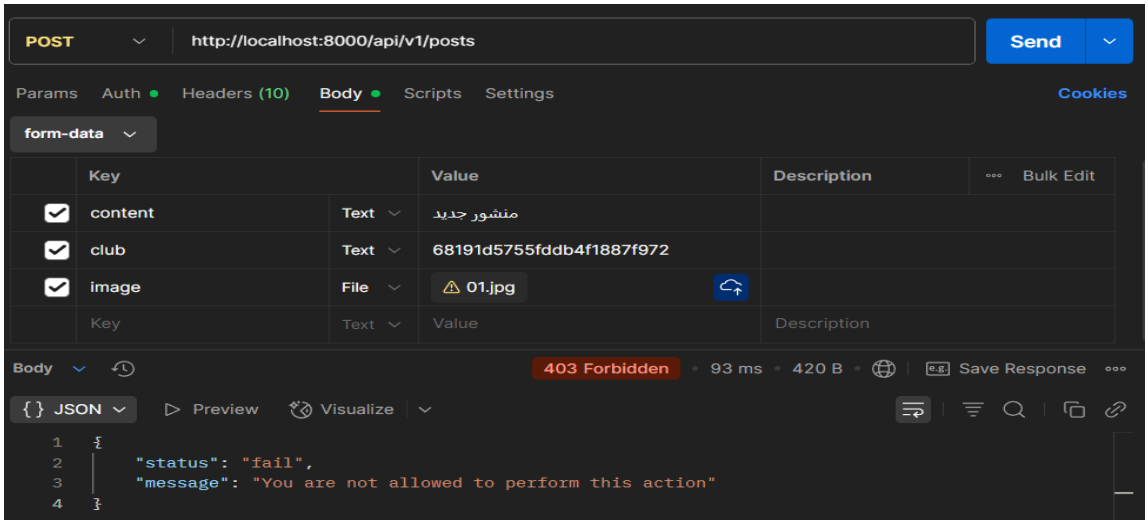
محاولة إنشاء منشور داخل نادٍ باستخدام حساب (طالب) لا يمتلك صلاحيات للنشر.

المتوقع:

يُرفض الطلب برسالة توضح أن الطالب ليس لديه صلاحية إنشاء منشور.

النتيجة الفعلية:

You are not allowed to perform this action



اختبار نقطة النهاية: إضافة منشور لنادي (POST /api/v1/posts)

الاختبار:

محاولة إضافة منشور باستخدام حساب مسؤول نادي لنادي ليس لديه صلاحية النشر فيه.

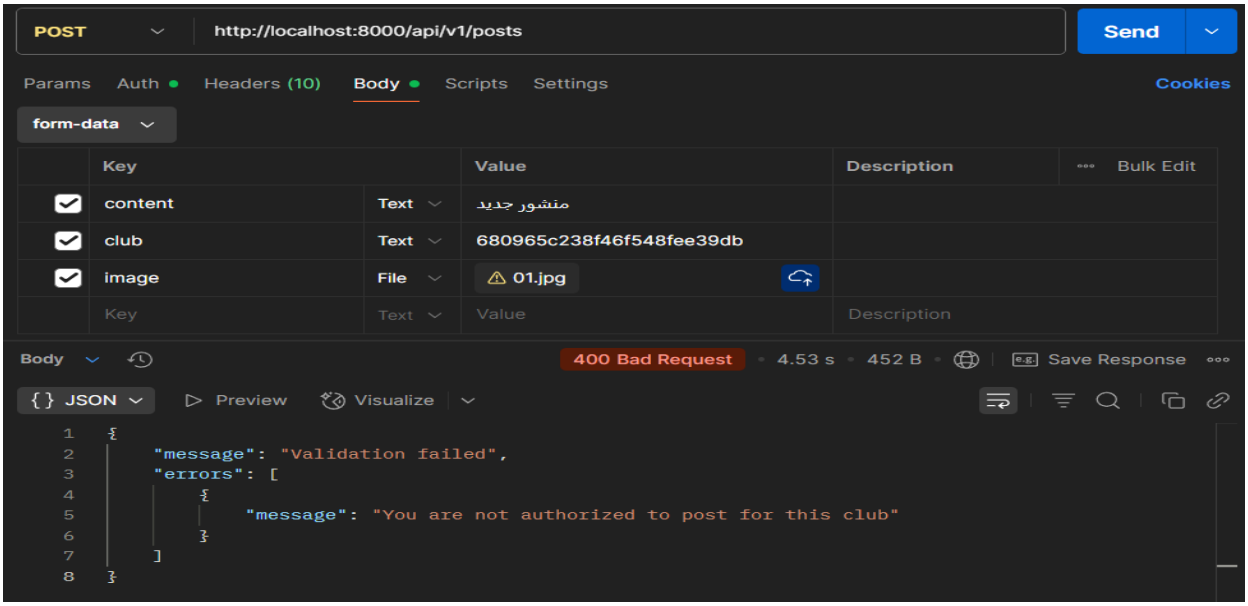
المُتوقّع:

رفض العملية برسالة توضح أن هذا المستخدم غير مخوّل للنشر في هذا النادي.

النتيجة الفعلية:

تم الرفض برسالة:

You are not authorized to post for this club



اختبار نقطة النهاية: تغيير كلمة المرور(PUT /api/v1/auth/changePassword)

الاختبار:

محاولة تغيير كلمة المرور بإدخال كلمة المرور الحالية بشكل خاطئ.

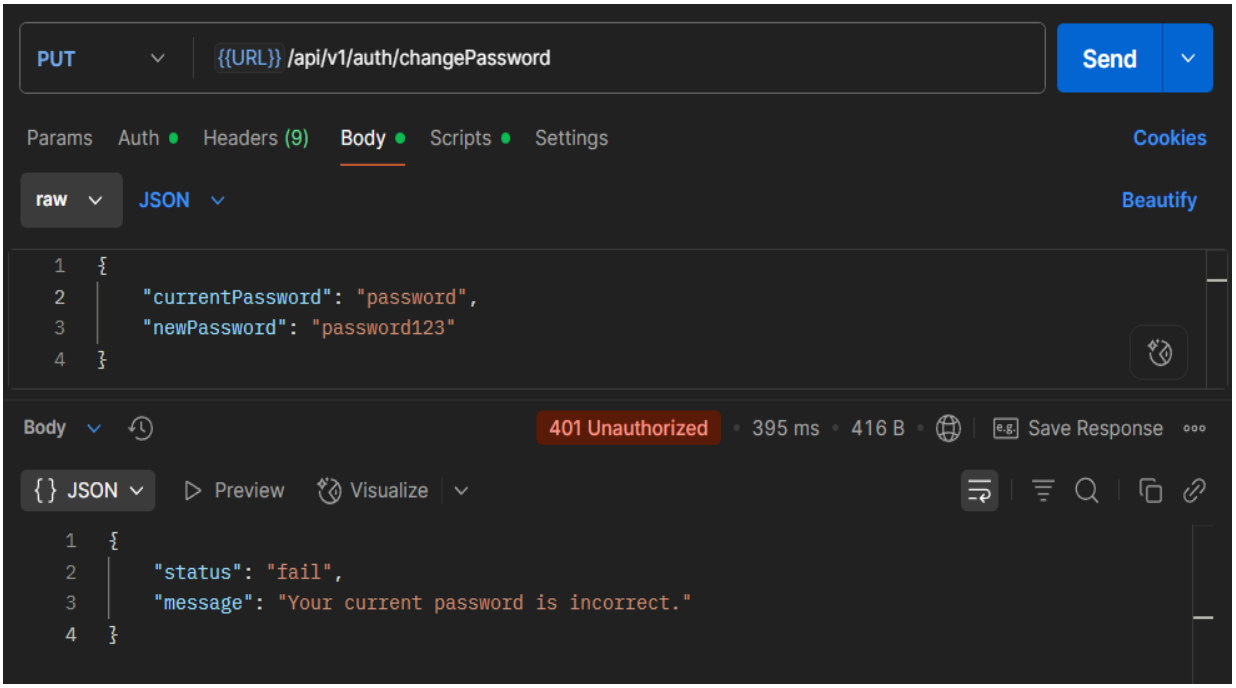
المُتوقّع:

رفض العملية برسالة توضح أن كلمة المرور الحالية غير صحيحة.

النتيجة الفعلية:

تم الرفض برسالة:

Your current password is incorrect



الخاتمة

في ختام هذا المشروع، نرى أننا لم نقدم مجرد نظام إلكتروني فحسب، بل وضعنا حجر الأساس لتعزيز التواصل والتفاعل بين طلاب جامعة بوليتكنك فلسطين. سعينا جاهدين لخلق بيئة إلكترونية تسهّل على الطلاب الانخراط والمشاركة في الأنشطة، وتدعم بناء مجتمع جامعي نابض بالحياة وروح التعاون والعمل الجماعي. نؤمن أن التجربة الجامعية تتعدى حدود الدراسة لتشمل الانتماء والتفاعل والمشاركة الفعالة. نأمل أن يكون هذا المشروع قد أسهم ولو بالقليل في إثراء هذه التجربة، وأن يربط الطلاب ببعضهم البعض وبإدارة الجامعة بطريقة أكثر سهولة وفعالية. ونسأل الله أن يستمر التطور والتحديث لمواكبة طموحات الطلاب وتحقيق بيئة جامعية محفزة ومُلهمة.

الأعمال المستقبلية (Future Work)

لتحقيق أقصى استفادة من المشروع وتطويره بما يلبي احتياجات الطلاب بشكل أفضل، يمكن إضافة مجموعة من التحسينات والوظائف الجديدة التي تسهم في رفع كفاءة المنصة وتعزيز تجربة المستخدم. تهدف هذه التحسينات إلى تسهيل التواصل، تحسين التنظيم، وتوفير أدوات تفاعلية متطورة تلبي تطلعات المجتمع الجامعي في المستقبل، ومنها:

- تطوير تطبيق جوال متكامل يتيح الوصول السهل والسريع إلى المنصة عبر الهواتف الذكية، مع دعم الإشعارات الفورية لتحديثات الأندية والفعاليات.
- دمج نظام تقييم وتغذية راجعة متقدم يمكّن الطلاب من تقييم الأنشطة وتقديم مقترحات لتحسين جودة الفعاليات.
- إضافة خاصية البث المباشر للفعاليات لتمكين الطلاب من متابعة الأنشطة عن بُعد والمشاركة الافتراضية.
- ربط الفعاليات مع التقويم (Callender) الشخصي للطلاب لتسهيل تنظيم الجداول ومتابعة المواعيد الهامة.
- توفير نظام تشات وتواصل داخلي بين الطلاب لتعزيز التفاعل الاجتماعي وتبادل المعلومات داخل المنصة.

(References) المراجع

1) **Node.js**
Node.js is a JavaScript runtime built on Chrome's V8 JavaScript engine.
Available at: https://nodejs.org/docs/latest/api/

2) **Express.js**
Express is a minimal and flexible Node.js web application framework.
Available at: https://expressjs.com/

3) **React**
A JavaScript library for building user interfaces, maintained by Facebook.
Available at: https://react.dev/learn

4) **MongoDB**
MongoDB is a document-oriented NoSQL database for high volume data storage.
Available at: https://docs.mongodb.com/manual/

5) **Cloudinary**
Cloudinary provides cloud-based image and video management services.
Available at: https://cloudinary.com/documentation

6) **Tailwind CSS**
Utility-first CSS framework for rapid UI development.
Available at: https://tailwindcss.com/docs

7) **JWT.io**
JSON Web Tokens (JWT) for secure transmission of information between parties as a JSON object.
Available at: https://jwt.io/introduction

8) **Render**
Render is a unified cloud platform to build and run all your apps and websites with free SSL, a global CDN, private networks, and auto deploys from Git.
Available at: https://render.com/docs

9) **Axios**
Promise based HTTP client for the browser and node.js, used for API communication.
Available at: https://axios-http.com/docs/intro

10) **Multer**
Middleware for handling multipart/form-data, used for uploading files in Node.js.
Available at: https://github.com/expressjs/multer

11) **Nodemailer**

Module for Node.js to send emails easily. Available at: https://nodemailer.com/about/

نموذج إقرار وتوثيق استخدام أدوات الذكاء الاصطناعي في مشروع التخرج

أسماء الطلبة: • أحمد محمد عصافرة • محمد جهاد عقيلي • بشار إسماعيل الهرش	عنوان المشروع: PPU Students Clubs
الأرقام الجامعية: • 211033 • 211057 • 211134	اسم المشرف الأكاديمي: الدكتور ضياء أبو زينة
الكلية: كلية تكنولوجيا المعلومات وهندسة الحاسوب / دائرة علم الحاسوب	السنة/الفصل الدراسي: فصل ثاني/ 2025

الغرض من الاستخدام	اسم الأداة	أماكن الاستخدام في التقرير (فصول / صفحات)	الأمر الأساسي المستخدم (Prompt)	هل تم تعديل الناتج؟	نسبة الاستخدام التقديرية	ملاحظات
توليد نصوص	ChatGPT	**الفصل الأول**: وصف مختصر للنظام	أعطيني شرح بسيط وواضح عن النظام اللي اشتغلنا عليه، شو الفكرة الرئيسية شو الهدف منه وكيف بساعد المستخدمين	نعم	50%	
		الفصل الثالث: الهيكلية العامة للنظام ... هيكلية بناء النظام	فسرلي بشكل مختصر كيف النظام مبني شو الأجزاء الأساسية فيه وكيف بتشتغل مع بعض عشان يشتغل النظام تمام	نعم	60%	
تلخيص محتوى	ChatGPT	**الفصل الأول:** المقدمة	اكتب لي مقدمة بسيطة وواضحة للفصل الأول تعطي فكرة عامة عن موضوع الفصل وأهميته	نعم	30%	
		الفصل الثاني: المقدمة	اكتب لي مقدمة مختصرة للفصل الثاني تشرح الهدف منه ونبذة عن المحتوى اللي رح يغطيه	نعم	40%	
تدقيق لغوي	Word Editor AI	**كامل المشروع**	صحح الأخطاء اللغوية في كل فقرة أقوم بإعطائها لك	نعم	15%	
كتابة أكواد برمجية	GitHub Copilot	تصحيح أكواد JS & JSX	تصحيح الأكواد التي نقوم بها أثناء الكتابة	نعم	20%	
Backend	Grok (AI assistant)	في بناء وإدارة التخزين للصور ضمن النظام	Build Cloudinary services for storing and managing images integrating them smoothly and efficiently with the backend	نعم	40%	تم التعريف عن طريقة الاستخدام من قبله
		في تصميم نمط Factory لإنشاء الحقول وتعديلها	Create a Factory that makes different system objects in a simple organized way so everything works smoothly together	نعم	10%	لم يتم التعمق في بنائه لأننا قمنا ببنائه في أكثر من مشروع
		بناء بعض ال middlewares المستخدمة في ال	Help me build middlewares that manage requests like checking user authorization and handling errors in one place	نعم	25%	

	30%	نعم	Fix my error when testing endpoints	في اختبار وتصحيح أخطاء API وEndpointsEnd Points		
نسيان أو خطأ في إضافة توكن التوثيق (Bearer token)	30%	نعم	Handle missing or incorrect headers like Authorization in frontend requests	إرسال (Headers)	GitHub Copilot	Frontend
	20%	نعم	Fix CORS errors when frontend tries to access backend resources	مشاكل مشاركة البيانات (CORS)		
	20%	نعم	Handle timeout or slow responses from backend in frontend UI	إدارة التأخيرات في الردود		
قمنا بأخذ روابط المراجع المستخدمة في المشروع			أعطيني جميع المراجع التي تستخدم في بناء تطبيقات الويب بشكل كامل	المراجع (References)	ChatGPT	توثيق ومراجع

جدول 24 : جدول استخدام أدوات الذكاء الاصطناعي

إقرار فريق مشروع التخرج :نقر بأن استخدام أدوات الذكاء الاصطناعي تم بشكل مسؤول وبما يتوافق مع السياسات الجامعية، وقد راجعنا وحررنا كافة المخرجات بما يعكس فهمنا الشخصي.

التاريخ: 01/06/2025

توقيع الطلبة:

أحمد محمد عصافرة محمد جهاد عقيلي بشار إسماعيل الهرش