الجمهورية العربية السورية

وزارة التعليم العالي

جامعة تشرين

كلية الهندسة المعلوماتية

قسم : البرمجيات

**موقع الكتروني لتسيير القرارت ضمن الجامعة**

**أعداد الطلاب :**

**بإشراف الدكتور:**

2021-2020

Contents

[1-1-الهدف من المشروع : 4](#_Toc78155073)

[1-2-فصول المشروع : 5](#_Toc78155074)

[1-3-البرامج المستخدمة: 5](#_Toc78155075)

[2-1-مقدمة : 6](#_Toc78155076)

[2-2-اهمية المواقع الالكترونية : 6](#_Toc78155077)

[3-1- تقنية :php.net 8](#_Toc78155078)

[**3-1-3- ميزات: (php.net)** 8](#_Toc78155079)

[**3-3-2- اللغات المستخدمة :** 9](#_Toc78155080)

[**3-2-1-لغة : HTML** 9](#_Toc78155081)

[**3-2-1-1-مميزات لغة HTML :** 9](#_Toc78155082)

[**3-2-2- لغة CSS :** 9](#_Toc78155083)

[**3-2-2-1-تعريف لغة css :** 10](#_Toc78155084)

[**3-2-2-2-وظيفة لغة css :** 10](#_Toc78155085)

[3-3- framework Bootsrap : 10](#_Toc78155086)

[4-1- مقدمة : 14](#_Toc78155087)

[4-2- خصائص قاعدة البيانات : 14](#_Toc78155088)

[4-3- تعريف قاعدة البيانات : 15](#_Toc78155089)

[ا*لشكل (4-1) نظام قواعد* البيانات 16](#_Toc78155090)

[4-4-مزايا أسلوب قاعدة البيانات:Characteristics of database approach 17](#_Toc78155091)

[4-4-1- طبيعة التوصيف الذاتية لنظام قاعدة البيانات Self\_Describing Nature of a Database System : 18](#_Toc78155092)

[4-4-2-الفصل بين البرامج وتجريد البيانات Insulation Between Programs And Data, And Data Abstraction : 18](#_Toc78155093)

[4-4-3- دعم مشاهد متعددة للبيانات Support of Multiple Views of the Data : 19](#_Toc78155094)

[4-4-4- مشاركة البيانات ومعالجة المناقلات متعددة المستخدمين Sharing of Data And Multiuser Transaction Processing : 20](#_Toc78155095)

[4-5- العاملون في قاعدة البيانات Actors on the DataBase : 20](#_Toc78155096)

[4-6- مدراء قاعدة البيانات Database Administrators : 21](#_Toc78155097)

[4-7- مصممو قاعدة البيانات : Database Designers 21](#_Toc78155098)

[4-8- البنية ثلاثية المخططات واستقلال قاعدة البيانات Three-Schema Architecture and Data Independence : 21](#_Toc78155099)

[4-8-1- البنية ثلاثية المخططات The Three-Schema Architecture: 22](#_Toc78155100)

[4-8-2- استقلال البيانات Data Independence : 23](#_Toc78155101)

[4-9- نمذجة البيانات باستخام ترابط الكينونات ( ( ER : 24](#_Toc78155102)

[4-9-1- مقدمة : 24](#_Toc78155103)

[5-1- المقدمة : 26](#_Toc78155104)

[5-2-حالات الاستخدام الأساسية: 26](#_Toc78155105)

[**5-2-1-حالات استخدام الخاصة بالمدير :** 26](#_Toc78155106)

[**5-2-2-حالات استخدام المسؤول عن قسم :** 26](#_Toc78155107)

[5-2-3 مخطط حالة الدكتور : 27](#_Toc78155108)

[6-1 واجهات التطبيق : 29](#_Toc78155109)

[ا*لشكل (6-8) واجهة ادارة الكليات* 34](#_Toc78155110)

[7-1-المشاكل التي واجهت المشروع: 39](#_Toc78155111)

[7-2-إمكانية تطوير المشروع: 39](#_Toc78155112)

[7-3-الخاتمة: 40](#_Toc78155113)

[المراجع : 40](#_Toc78155114)

فهرس الاشكال :

[3-4-1طرق طلب البيانات وفق النظام : 12](#_Toc78155115)

[الشكل(4-2) البنية ثلاثية المخططات 23](#_Toc78155116)

[الشكل (6-1) الواجهة الرئيسية 30](#_Toc78155117)

[الشكل (2-6) تسجيل دخول 30](#_Toc78155118)

[الشكل (6-3) واجهة المدير الرئيسية 31](#_Toc78155119)

[الشكل (6-4) واجهة اضافة كلية جديد 31](#_Toc78155120)

[الشكل (6-5) واجهة اضافة قسم 32](#_Toc78155121)

[الشكل (6-6) واجهة اضافة دكتور 33](#_Toc78155122)

[الشكل (6-7) واجهة ادارة المستخدمين 33](#_Toc78155123)

[الشكل (6-9) واجه اضافة اعلان 34](#_Toc78155124)

[الشكل (6-10) واجهة اضافة قرار 35](#_Toc78155125)

[الشكل (6-11) واجهة عرض اصوات قرار ما 35](#_Toc78155126)

[الشكل (6-12) واجهة عرض كل القرارت 36](#_Toc78155127)

**الفصل الاول**

# 1-1-الهدف من المشروع :

الهدف من مشروعنا هو تقديم موقع الكتروني لتنسيق الية انشاء القرارت ضمن الكلية وكيفية عرضها للتصويت والموافقة عليها او الرفض ويقسم الى ثلاث اقسام

1. المدير :

-يحق له اضافة كلية

-اضافة مجلس

-اضافة رئيس قسم

- اضافة دكتور ضمن قسم

-ادارة المستخدمين

-ادارة الكليات والاقسام

-اضافة منشورات وادارتها

2- رئيس القسم :

يمكنه اضافة قرار

* تقديم القرار للتصويت
* الموافقة على القرار
* رؤية الاصوات
* عرض الاعلانات والمنشورات

-3 الدكتور :

-رؤية القرارات ضمن القسم الذي يتبع له

- التصويت على قرار او المناقشة عليه

عرض الاعلانات

# 1-2-فصول المشروع :

الفصل الاول : مقدمة عامة ونتحدث فيها عن استخدام الموقع الالكتروني في الشبكة الالكترونيه , وايضا عن هدف المشروع وفصول المشروع والبرامج المستخمة في المشروع

الفصل الثاني : نعرف في هذا الفصل المواقع الالكترونيه والهدف منها .

الفصل الثالث : نتحدث عن التقنيات المستخدمة وهي تقنية php.net عملها واهدافها , واللغات المستخدمة في عمل المشروع ونوضح مفهوم ال framework bootstrapt

الفصل الرابع : لمحة عن قواعد البيانات

الفصل الخامس : نتحدث في هذا الفصل عن مراحل تحليل وتوصيف النظام باستخدام حالات الاستخدام

الفصل السادس : نوضح في هذا الفصل مراحل بناء النظام , الواجهات وكيفية عمل المشروع

الفصل السابع : نناقش في هذا الفصل الخاتمة وامكانية تطوير المشروع والمشاكل التي واجهتنا .

# 1-3-البرامج المستخدمة:

* كما تم استعمال "xammp Server" في بناء قاعدة بيانات التي سوف تخزن عليها المعلومات.
* وسيتم استعمال " "sublime textفي وضع واجهات يمكن التحكم والوصول من خلالها الى البيانات المخزنة ضمن قاعدة البيانات.
* تم استعمال برنامج ""rational rose&"vision" في رسم مجموعة المخططات التي تعبر عن تحليل ودراسة النظام والعلاقات الموجودة فيه.

**الفصل الثاني**

المواقع الالكترونية

# 2-1-مقدمة :

الموقع الإلكترونيّ يعرف الموقع الإلكترونيّ بأنّه عدد من الصفحات التي ترتبط فيما بينها بروابط مختلفة، والتي يتمّ تخزينها على الخادم ذاته، حيث يمكن الولوج إلى هذا الموقع المُنشأ من خلال استعمال جهاز حاسوب مُحمَّل عليه متصفح الإنترنت، ومشبوك مع الشبكة العنكبوتية، كما ويمكن تصفح الموقع الإلكتروني باستعمال الأجهزة الذكية المحمولة، والأجهزة اللوحية أيضاً. تُعتبر المواقع الإلكترونيّة من الأدوات العصريّة التي تساعد على ربط الناس ببعضهم البعض، وتناقل المعلومات وتداولها، فلها أهميّة عظمى وكبيرة بشكل لا يوصف، لهذا فإننا نجد إقبالاً كبيراً من مختلف أصناف الناس على إنشاء المواقع الإلكترونيّة سواء العامّة أو المتخصّصة بما يحقّق الفائدة العامة أو الخاصة على حدّ سواء، وفيما يلي بيان لأهمّيّة المواقع الإلكترونيّة القصوى في حياة الإنسان.

# 2-2-اهمية المواقع الالكترونية :

يشهد العصر الذي نعيشه تضخم هائل للمعلومات وثورة معلوماتية بكل متطلباتها وقد أخذت المكتبات الجامعية على عاتقها مواكبة هذا التطور والمشي ركبه المتعايش مع هذا الفجر المعلوماتي. فالمكتبة الجامعية تعد القلب النابض للجامعة ومرفق حضاري لاستطيع هذه الأخيرة الاستغناء عنه لأنها تلعب دورا هاما في تطوير البحث العلمي .وفضلا عن كونها مركز للبحث والاطلاع أصبحت تمثل عنصرا أساسيا في تقويم الجامعات العصرية والحكم على مدى نجاحها . ولهذا فقد قامت المكتبات الجامعية بتوفير العديد من الخدمات لجمهورها وعملت على تقديمها بشكل سهل وسريع إلا أن تلك الخدمات افتقرت للجودة في الأداء فأصبح المستفيد يرغب في الحصول على معلومات أكثر فعالية وكفاءة، فإدارة الجودة لخدمات المكتبة هي عبارة عن نظام يعمل على تحسين مستوى الأداء للخدمة التي تقدمها من خلال موقعها على الانترنت . ولا يخفى كما سبق الذكر المكتبة الجامعية وجودها مبني أساسا على الجامعة وأهداف تلك المكتبات تكون أهداف الجامعة نفسها، لأنها تعمل على تشغيل خدماتها واتاحتها للمستفيدين من طلبة وأعضاء هيئة التدريس وأفراد المجتمع ككل ولابد أن تكون تلك الخدمات تقتصد عليهم الجهد والوقت والتكلفة. ولهذا الغرض كان لابد للمكتبة الجامعية أن يكون لها موقع الكتروني يبث مختلف خدماتها المذكورة آنفا بشكل أسهل وأسرع لتدعيم البحث العلمي والنهوض به إلى أرقى المستويات، حيث أصبحت البحوث العلمية تتطلب الدقة والسرعة والحداثة في المعلومات، وهذا ما حتم على المكتبة الجامعية تقديم مثل هذه الخدمات الجديدة للتماشي مع تطورات العصر كيالمقدمة ‌ ب تجعل الباحث يواظب على التعرف على مساهمة المواقع الالكترونية للمكتبة الجامعية في تجسيد وتطوير البحث العلمي وكذا معرفة أهمية هذه المواقع بالنسبة للباحثين ومدى اعتمادهم عليها وعلى مصادرها في إعداد بحوثهم . وحاولنا الإجابة عن إشكالية الدراسة من خلال مجموعة من الفرضيات والتي قمنا بدراستها من خلال مجموعة من الفرضيات والتي قمنا بدراستها من خلال تقسيم البحث إلى فصلين، الفصل الأول معنون ب"مكانة المواقع الإلكترونية في الفضاء الرقمي " نشأة وتعريف المواقع الالكترونية، أجيال الويب، معايير تقييم مواقع المكتبات الجامعية ،أهمية تقييم مواقع المكتبات الجامعية. و الفصل الثاني موسوم ب"البحث العلمي و نماذج عن المواقع الإلكترونية للمكتبات الجامعية" تطرقنا فيه مفهوم البحث العلمي، الأهداف والأهمية ،خصائص و خطوات البحث العلمي و نماذج عن مواقع المكتبات الجامعية العربية و أخرى أجنبية. والفصل الثالث المعنون ب "دراسة واقع المواقع الإلكترونية للمكتبات الجامعية في الجزائر: جامعة مستغانم نموذجا ". و هذا هو الشق الميداني للدراسة، وقد طبقت هذه الدراسة على طلبة الماستر تخصص إدارة واقتصاد المؤسسة. ودراسة وصفية لموقع جامعة عبد الحميد بن باديس الذي يضم بدوره موقع مكتبة الجامعة، حيث حاولنا من خلال ها التعرف على آراء الطلبة حول موقع المكتبة الجامعية وكيف ينظرون إلى ما يقدمه من تدعيم لبحثوهم العلمية، كما أردنا معرفة حالة موقع مكتبة الجامعة. كما تضمن هذا الفصل التحليل و الإستنتاج بناءا على نتائج الاستبيان وفي الأخير الوصول إلى نتائج على ضوء الفرضيات و النتائج العامة والاقتراحات

**الفصل الثالث**

**التقنيات واللغات المستخدمة**

# 3-1- تقنية :php.net

تعتبر لغة ال php من أقوى اللغات الحالية وأسرعها وتعتبر المفضلة عند كثير من المبرمجين

لسهولتها وبساطة التعامل معها وتمتاز هذه اللغة بأنها مفتوحة المصدر كما انها مجانية كما انها تتصدر لغات برمجة الويب في سهولة تعلمها وتشبه لغة c ولغة ال c++ في طريقة كتابة الكود ولكنها لا تحوي كل التعقيدات الموجودة في تلك اللغتين كما تتصدر لغات برمجة الويب

من حيث زهد الاسعار .

php عبارة عن لغة تسمى (server side scripting language) ونعني بذلك أنها تترجم

من جهة الخادم وليس من جهاز المستخدم لنضرب مثالا واقعيا على ذلك

عند الدخول الى الدردشة يجب عليك ان تقوم بتحميل (java vm) على جهازك لتتمكن من الدخول ولكن اذا قمت بزيارة اي موقع يستخدم لغة php فلن يكون هناك حاجة لتنصيب أو تحميل أي برامج مساعدة, صممت هذة اللغة خصيصا للمواقع ومن ضمن صفحات html يمكن

ادراج لغة php بداخلها لكي تنقذ في كل مرة قمت بزيارة الصفحة .بدأت لغة php بمجهود شخص واحد يسمى Rasmus Lerdorf في سنة 1994 ومرت بثلاث مراحل جذرية حتى انه في سنة ال2002 كان عدد المواقع التي تستخدمها اكثر من 9000000

لغة ال php لغة برمجية متاحة للكل لتطويرها او العمل عليها وهذا ما يسمى open source قديما كان اختصار php هو (personal home page) ولكن تغيرت االى (hypertext processor) .

## **3-1-3- ميزات: (php.net)**

* الأداء العالي: تعتبر اللغة جدا مميزة وفعالةحتى مع دعمها لشركة zend أصبحت أقوى بحيث يمكن أن تتحمل ملايين الأوامر في اليوم الواحد
* الدعم الكبير لعديد من قواعد البيانات: تدعم لغة php عددا كبيرا من قواعد البيانات منها dbm,mysql,oracle وغيرها.
* القيمة: اللغة مجانية ولا يوجد أي مبلغ تقوم بدفعه لتحميلها ويمكنك تحميل اللغة من *موقعها الرسمي.*

## **3-3-2- اللغات المستخدمة :**

## **3-2-1-لغة : HTML**

باعتبارها اللغة التي تسمح لنا ببناء الهيكل الأساسي لمواقع وتطبيقات الويب، ومن دونها لا نستطيع إنشاء الصفحات، إذ هي التي تسمح لنا بالقيام بأول خطوة وهي عرض المحتوى وبناء الهيكل الرئيسي للصفحات. إذ تعدّ من ضمن بروتوكولا الإنترنت المهمة، وتنتهج هذه اللغة عدداً من الأساليب المشهورة في مجال النشر لتشكل حلقة وصل بين المؤلف والمحرر

لغة HyperText Markup Language ، ويشار إليها اختصاراً بHTML، وتُعرف بلغة رميز النص التشعبي، هي عبارة عن لغة ترميز خاصة تستخدم لغايات تصميم وإنشاء صفحات الويب، كما يمكننا القول بإنّها عبارة عن الهيكل الرئيسي لصفحات الويب والبنية التحتيّة لها؛ إذ تقدّم وصفاً مفصلاً حول الكيفيّة التي ستكون عليها آلية عرض محتويات الموقع الإلكتروني ويكون ذلك بتقسيمه إلى عنوان وفقرات، ويتم ذلك كله بالاعتماد على ما يُعرف بالوسوم Tags.

### **3-2-1-1-مميزات لغة HTML :**

* لا نحتاج لاي برامج خاصه للتصميم بل يكفي برنامج المفكره
* سعلة التعلم حيث انه يمكن بعد درسين ان تقوم بتصميم صفحتك مع هذه اللغه
* ان تعلم هذه اللغه تمكن من اجراء التعديلات التي تريدها على الموقع بكل يسر وسهولة
* امكانية اللكتابة باللغه العربيه بسهوله
* ان معرفة لوسم لغة ال html تمكنك من اضافة اي نصنص برمجيه على صفحتك , مثل نصوص ال java البرمجية او java script او نصوص dhtml الديناميكية
* معرفة وسوم هذه اللغه من ترجمة الصوص البرمجية الى اللغه العربية
* معرفة هذه اللغه تعطي تحكما دقيقا في خصائص وجزئيات اللصفحة اكثر من اي برنامج تصميم صفحه اخر
* جميع برامج التصميم الجاهزه مبنيه على اساس لغة ال html وهذه اللغه هي القاعدع لمثل هذه البرامج , مثل برنامج front page والذي مبني كليا على لغة ال html

## **3-2-2- لغة CSS :**

قد قامت منظمة [W3C](http://www.w3.org/) بابتكار هذه التقنية و جعلتها كمعيار قياسي لتصميم المواقع ، وقد بدأت الكثير من المواقع في الانتقال من الأسلوب التقليدي للتصميم باستخدام الجداول إلى الأسلوب الجديد باستخدام تقنية CSS .,  فالجداول لها وظيفة أخرى وهي عرض وتنسيق البيانات التي تحتاج إلى أن تكون ضمن جدول , و بمعنى آخر يستطيع أي شخص باستخدام CSS أن يصمم موقعه بالكامل دون استخدام الجداول كما هو متبع في الطرق التقليدية لتصميم الصفحات، و هذا لا يعني أن الجداول لن يتم استخدامها مع CSS,  تسهل عملية التصميم والتحكم في شكل الصفحات وتحديد مواقع و خصائص مكونات الصفحة .

### **3-2-2-1-تعريف لغة css :**

CSS هي اختصار لـ Cascading Style Sheet و هي تعني صفحات الأنماط الانسيابية, وال CSS هي لغة تصميم تحدد شكل وثيقة HTML، فهي تهتم بالخطوط، الألوان، الهامش، والعرض والارتفاع، صورة خلفية الموقع، وكيفية توزيع المساحات وأشياء أخرى كثيرة CSS, توفر المزيد من الخيارات وهي أكثر دقة وعملية، وهي مدعومة من قبل المتصفحات الرئيسية اليوم.

### **3-2-2-2-وظيفة لغة css :**

وظيفة هذه التقنية هي التحكم بكيفية عرض صفحة الويب دون التدخل في المحتوى، مما يسهل من عملية إدارة الموقع، فتعديل ملف CSS واحد سيؤثر على تصميم كافة صفحات الموقع . فملفات CSS تعمل على تبسيط عملية تصميم و إدارة المواقع، وتحل الكثير من المشاكل التي قد يواجهها أي مطور للمواقع .

# 3-3- framework Bootsrap :

الـ Bootsrap هو إطار عمل يستعمل في تصميم المواقع يعتمد أساسا على أكواد Html ,Css و Javascript قام بتطويره فريق عمل موقع تويتر وكان الهدف منه هو مساعدة االمبرمجين في تطوير تطوير تطبيقات ويب الفردية وكان ذلك مطلع سنة 2011 إلا أنه سرعان ما انتشر بين المصممين حتى صار الإعتماد عليه من طرف المصمم شبه كلي ، ولا يكاد يخلو تصميم الآن من الـ bootsrap., ويعتبر framework أداة تتوفر على عدة مكونات تساعد على وضع أساس التطبيق الذي ترغب في إنشائه, كما يوفر مجموعة من الاكواد المكتوبة مسبقا لجزء من التطبيق, ويتواجد هذا المفهوم في عدة لغات برمجية نذكر منها java و php مما يوضح أنه يساعد على إنشاء تطبيقات الويب أو لمكونات البرامج وغيرها من الاستعمالات سنتطرق إليها لاحقا.

وقد تم اللجوء إلى هذه الطريقة لتوفير الوقت على المبرمج لانه يجد عدة أمور مسبقا ولايضطر إلى إعادة برمجتها, وذلك راجع لانه يتوفر على عدة مكتبات كل واحدة متخصصة في مجال ما, وعلى الشخص الذي يستعمل framework أن يستعمل هذه المكتبات وكذلك أن يحترم نمطها, بمعنى اخر عليك إن أردت استعمال framework عليك أن تدرسه جيدا وتعرف كيف تمبرمج به وماهي المكتبات المتوفرة به حتى تستعملها جيدا, أمر اخر يجب الاشارة إليه هو أن framework يتطلب عملا من مطور التطبيقات فما هو إلا أداة مساعدة.

ويستعمل أغلب framework مفهوم MVC , ومن مميزاته كذلك أنه يسهل عملية صيانة التطبيق, وكمثال على إفادته, فهناك من framework ما يبسط طريقة التعامل مع database, وهناك منها ما يتيح إمكانية التحكم في استمارات صفحات الويب, التحكم في تأمين ما يسمى SQL injections وهي نوع من الاختراقات التي تم ل database المواقع (رغم أنه لا أمان على الانترنت ), توفر كذلك بعض framework لغات البرمجة إمكانية استعمال API تطبيقات أخرى (Application Programming Interface)

إذن كخلاصة framework وجد لمساعدة المطورين على تسهيل عملهم باستخدام أدوات ومكتبات برمجت مسبقا, وذلك لترك المجال للمطور للابداع في أمور أخرى غير الموجودة مسبقا,

3-4-بروتوكول ال HTTP :

بروتوكول نقل النص الفائق أو بروتوكول نقل النص التشعبي ([بالإنجليزية](https://ar.wikipedia.org/wiki/%D8%A7%D9%84%D9%84%D8%BA%D8%A9_%D8%A7%D9%84%D8%A5%D9%86%D8%AC%D9%84%D9%8A%D8%B2%D9%8A%D8%A9): HyperText Transfer Protocol HTTP)‏ هو [بروتوكول عديم الحالة](https://ar.wikipedia.org/wiki/%D8%A8%D8%B1%D9%88%D8%AA%D9%88%D9%83%D9%88%D9%84_%D8%B9%D8%AF%D9%8A%D9%85_%D8%A7%D9%84%D8%AD%D8%A7%D9%84%D8%A9) لنقل الموارد في [الإنترنت](https://ar.wikipedia.org/wiki/%D8%A5%D9%86%D8%AA%D8%B1%D9%86%D8%AA)، وهو الطريقة الرئيسة والأكثر انتشاراً لنقل البيانات في [الشبكة العنكبوتية العالمية](https://ar.wikipedia.org/wiki/%D8%B4%D8%A8%D9%83%D8%A9_%D8%B9%D9%86%D9%83%D8%A8%D9%88%D8%AA%D9%8A%D8%A9_%D8%B9%D8%A7%D9%84%D9%85%D9%8A%D8%A9). الهدف الأساسي من بنائه كان إيجاد طريقة لنشر واستقبال صفحات [HTML](https://ar.wikipedia.org/wiki/%D9%84%D8%BA%D8%A9_%D8%AA%D8%B1%D9%85%D9%8A%D8%B2_%D8%A7%D9%84%D9%86%D8%B5_%D8%A7%D9%84%D9%81%D8%A7%D8%A6%D9%82).

بدأ تطوير البروتوكول من قبل [تيم بيرنرز لي](https://ar.wikipedia.org/wiki/%D8%AA%D9%8A%D9%85_%D8%A8%D9%8A%D8%B1%D9%86%D8%B1%D8%B2_%D9%84%D9%8A) في معهد [سيرن](https://ar.wikipedia.org/wiki/%D8%B3%D9%8A%D8%B1%D9%86) سنة 1989. تم تطوير وثائق [طلب التعليقات](https://ar.wikipedia.org/wiki/%D8%B7%D9%84%D8%A8_%D8%AA%D8%B9%D9%84%D9%8A%D9%82%D8%A7%D8%AA) بالتنسيق مع [مجموعة مهندسي شبكة الإنترنت](https://ar.wikipedia.org/wiki/%D9%85%D8%AC%D9%85%D9%88%D8%B9%D8%A9_%D9%85%D9%87%D9%86%D8%AF%D8%B3%D9%8A_%D8%B4%D8%A8%D9%83%D8%A9_%D8%A7%D9%84%D8%A5%D9%86%D8%AA%D8%B1%D9%86%D8%AA) [ورابطة الشبكة العالمية](https://ar.wikipedia.org/wiki/%D8%B1%D8%A7%D8%A8%D8%B7%D8%A9_%D8%A7%D9%84%D8%B4%D8%A8%D9%83%D8%A9_%D8%A7%D9%84%D8%B9%D8%A7%D9%84%D9%85%D9%8A%D8%A9).

وهو من [الطبقة السابعة](https://ar.wikipedia.org/wiki/%D8%B7%D8%A8%D9%82%D8%A9_%D8%A7%D9%84%D8%AA%D8%B7%D8%A8%D9%8A%D9%82%D8%A7%D8%AA) لنظام [حزمة بروتوكولات الإنترنت](https://ar.wikipedia.org/wiki/%D8%AD%D8%B2%D9%85%D8%A9_%D8%A8%D8%B1%D9%88%D8%AA%D9%88%D9%83%D9%88%D9%84%D8%A7%D8%AA_%D8%A7%D9%84%D8%A5%D9%86%D8%AA%D8%B1%D9%86%D8%AA) وهي [طبقة التطبيقات](https://ar.wikipedia.org/wiki/%D8%B7%D8%A8%D9%82%D8%A9_%D8%A7%D9%84%D8%AA%D8%B7%D8%A8%D9%8A%D9%82%D8%A7%D8%AA) ويستخدم من قبل [متصفحات الويب](https://ar.wikipedia.org/wiki/%D9%85%D8%AA%D8%B5%D9%81%D8%AD_%D9%88%D9%8A%D8%A8) والتي تسمى عميل المستخدم user-agent ويستخدم [المنفذ](https://ar.wikipedia.org/wiki/%D9%85%D9%86%D9%81%D8%B0_(%D8%B4%D8%A8%D9%83%D8%A7%D8%AA)) رقم 80 على الخادم غالبًا بالتعاون مع [الطبقة الرابعة](https://ar.wikipedia.org/w/index.php?title=%D8%A7%D9%84%D8%B7%D8%A8%D9%82%D8%A9_%D8%A7%D9%84%D8%B1%D8%A7%D8%A8%D8%B9%D8%A9&action=edit&redlink=1) وبالتحديد مع [بروتوكول التحكم بالنقل](https://ar.wikipedia.org/wiki/%D8%A8%D8%B1%D9%88%D8%AA%D9%88%D9%83%D9%88%D9%84_%D8%A7%D9%84%D8%AA%D8%AD%D9%83%D9%85_%D8%A8%D8%A7%D9%84%D9%86%D9%82%D9%84) للحصول على الصفحات المطلوبة وبعد ذلك تبدا مهمة بروتوكول TCP لتولى المهمة من هنا والبدأ في عمله.

وتعتبر الصفحة كأي وثيقة عادية نصية ولكنها تحتوى على بعض الإضافات الأخرى [الفائقة](https://ar.wikipedia.org/wiki/%D8%B1%D8%A7%D8%A8%D8%B7_%D8%AA%D8%B4%D8%B9%D8%A8%D9%8A) مثل روابط لأماكن أو مواقع أو صفحات أخرى بالإضافة إلى بعض الإضافات الحديثة مثل الصور والوسائط المتعددة والتي يتم الحصول عليها في طلب منفصل آخر ولكنها تتواجد داخل تلك الصفحة بعد تحميلها.

يتميز هذا النظام بمعياريته السهلة التي تجعل من استضافة أي موقع أمرا سهلًا وبسيطاً، وحسب ما توضح صفحات وصف هذا النظام في ملفات [RFC](https://ar.wikipedia.org/wiki/%D8%B7%D9%84%D8%A8_%D8%AA%D8%B9%D9%84%D9%8A%D9%82%D8%A7%D8%AA) فإن استضافة هذا النظام على [المسيرات](https://ar.wikipedia.org/wiki/%D8%B1%D8%A7%D9%88%D8%AA%D8%B1_(%D8%AD%D9%88%D8%B3%D8%A8%D8%A9)) Routers تجعل من عملية نقل صفحات المواقع أمراً سريعًا جداً بحسب سرعة استضافة موقعك.

وحيث أن الاستضافة الخاصة بأي موقع لا بد أن تتعامل مع هذا النظام، فإن مواقع الويب جميعها تعمل على أساس بنية هذا النظام وطريقته بغض النظر عن جهة الاستضافة

#### 3-4-1طرق طلب البيانات وفق النظام :

تقسم لثمانية حالات

HEAD

تطلب رداً مطابقاً لذلك الذي يرجعه طلب GET، لكن بدون قسم الـ (جسم body). ويفيد هذا في الحصول على معلومات عن المطلوب دون نقل على كامل المحتوى.

GET

طلب عرض؛ وهو أكثر وسيلة مستخدمة اليوم على الشبكة.

POST

إرسال معلومات من المستخدم الزبون (من form في صفحة html مثلا) للهدف المرفق. توضع البيانات ضمن قسم الـ (جسم body) من الطلب.

PUT

تحميل إلى الهدف.

DELETE

مسح الهدف (و نادراً ما تستخدم).

TRACE

ترد بذات الطلب الذي وصل، وذلك ليستطيع الزبون أي تعديلات تقوم بها خوادم في الطريق.

OPTIONS

ترد الوسائل التي يدعمها خادم الويب، يستخدم لفحص كيف يعمل خادم الويب.

CONNECT

للاستخدام مع خادم وكيل قد يحول إلى نفق [بروتوكول طبقة المنافذ الآمنة](https://ar.wikipedia.org/wiki/%D8%A8%D8%B1%D9%88%D8%AA%D9%88%D9%83%D9%88%D9%84_%D8%B7%D8%A8%D9%82%D8%A9_%D8%A7%D9%84%D9%85%D9%86%D8%A7%D9%81%D8%B0_%D8%A7%D9%84%D8%A2%D9%85%D9%86%D8%A9).

تدعم خوادم الويب GET و HEAD على أقل تقدير، وعادة ما تدعم OPTIONS أيضاً.

3-4-2اصدارات البروتوكول :

صدر من الميثاق أربع إصدارات، ويعتمد كل إصدار على قصور في الأداء في الإصدار السابق له:

**الإصدار 0.9**: هذا الإصدار تم استخدامه في [النموذج مبدئي](https://ar.wikipedia.org/wiki/%D9%86%D9%85%D9%88%D8%B0%D8%AC_%D9%85%D8%A8%D8%AF%D8%A6%D9%8A) [للشبكة العنكبوتية العالمية](https://ar.wikipedia.org/wiki/%D8%B4%D8%A8%D9%83%D8%A9_%D8%B9%D9%86%D9%83%D8%A8%D9%88%D8%AA%D9%8A%D8%A9_%D8%B9%D8%A7%D9%84%D9%85%D9%8A%D8%A9).

**الإصدار 1.0:** وهو الإصدار الأكثر شهرة وما زال يستخدم على نطاق واسع جداً

**الإصدار 1.1:** وهو يستخدم على نطاق متوسط

**الإصدار 2:** تم اعتماد هذا الإصدار في فبراير 2015. بحسب احصائيات W3Techs في يناير 2018 كان يدعم 20.5% من أهم 10 ملايين موقع هذا الإصدار.

**الإصدار 3:** تم اعتماد هذا الإصدار في نوفمبر 2018. يقوم هذا الإصدار باستبدال بروتوكول [TCP](https://ar.wikipedia.org/wiki/%D8%A8%D8%B1%D9%88%D8%AA%D9%88%D9%83%D9%88%D9%84_%D8%A7%D9%84%D8%AA%D8%AD%D9%83%D9%85_%D8%A8%D8%A7%D9%84%D9%86%D9%82%D9%84) لنقل المعلومات ببروتوكول QUIC المبني على بروتوكول [UDP](https://ar.wikipedia.org/wiki/%D8%A8%D8%B1%D9%88%D8%AA%D9%88%D9%83%D9%88%D9%84_%D8%AD%D8%B2%D9%85_%D8%A8%D9%8A%D8%A7%D9%86%D8%A7%D8%AA_%D8%A7%D9%84%D9%85%D8%B3%D8%AA%D8%AE%D8%AF%D9%85).

**الفصل الرابع**

**قواعد البيانات**

# 4-1- مقدمة :

يمكن تعريف قاعدة البيانات database على أنها مجموعة من البيانات المرتبطة ببعضها. ويقصد بالبيانات data حقائق معروفة يمكن تسجيلها وهي تملك القليل من المعنى.

على سبيل المثال: القيمة 47 لاتعبر إلا عن أربعة في مرتبة العشرات وسبعة في مرتبة الآحاد وعندما نقرنها بالحرارة c47 تملك معنى أكثر ولكن عندما نربطها بموقع مثل المدرج أو الفرن تصبح أكثر وضوحاً، يمكن تسجيل اسماء، أرقام هواتف وعناوين الأشخاص الذين تعرفهم في دفتر عناوين مفهرس، أو قد تقوم بتخزينها على قرص صلب باستخدام حاسوب شخصي و أحد البرامج مثل Microsoft Access. .

إن هذه من البيانات عندما تسند لعلاقة مترابطة تصبح ذات معنى أكثر، وهي عبارة عن قاعدة بيانات.

# 4-2- خصائص قاعدة البيانات :

تملك قاعدة البيانات الخصائص التالية:

* تملك قاعدة البيانات صفة العالم الحقيقي، تُدعى أحياناً العالم المصّغر miniworld وبالتالي فإنّ التغيرات على العالم المصغّر تنعكس أو نظهر في قاعدة البيانات.
* قاعدة البيانات هي مجموعة متماسكة منطقياً من البيانات وبالتالي فإن تشكيلة عشوائية من البيانات لا تصح تسميتها قاعدة بيانات.
* قاعدة البيانات مصممة ومبنية ومؤهلة ببيانات لغرض محدد. ولها مجموعة محددة من المستخدمين وبعض التطبيقات التي يكون هؤلاء المستخدمون معنيون بها.

بمعنى آخر، تملك قاعدة البيانات مصدراً معيناً يتم الحصول على البيانات منه، وفق درجة معينة من التفاعل مع أحداث العالم الحقيقي، وأشخاص معنيين بمحتوياتها. لكي تكون قاعدة البيانات دقيقة وموثوقة طوال الوقت يجب أن تعكس فعلياً العالم المصغّر الذي تمثله، وبالتالي يجب أن تظهر التغييرات في قاعدة البيانات بأقصى سرعة ممكنة.

يمكن لقاعدة البيانات أن تكون من أي حجم size أو مستوى تعقيد complexity. , على سبيل المثال، يمكن أن تتألف قائمة الأسماء والعناوين من بضعة مئات من السجلات ولكل سجل بينة بسيطة. من جهة أخرى، يمكن أن يتضمن ارشيف حاسوبي لمكتبة ضخمة نصف مليون سجل

فئات مختلفة ( حسب تسم المؤلف، حسب الموضوع، حسب عنوان الكتاب ) حيثُ كل فئة مرتبة أبجدياً.

يمكن إنشاء قاعدة بيانات حاسوبية إما باستخدام مجموعة من البرامج الخاصة بهذا الأمر وإما باستخدام نظام إدارة قواعد بيانات.

# 4-3- تعريف قاعدة البيانات :

يمكن تعريف نظام إدارة قواعد البيانات ( DBMS ) DataBase Management System على أنه مجموعة من البرامج التي تمكن المستخدمين من إنشاء والتعامل مع قاعدة البيانات. إنّ نظام إدارة قواعد البيانات هو عبارة عن نظام برمجيات للأغراض العامة يُتيح عمليات تعريف، بناء، معالجة، ومشاركة قواعد البيانات بين العديد من المستخدمين والتطيقات.

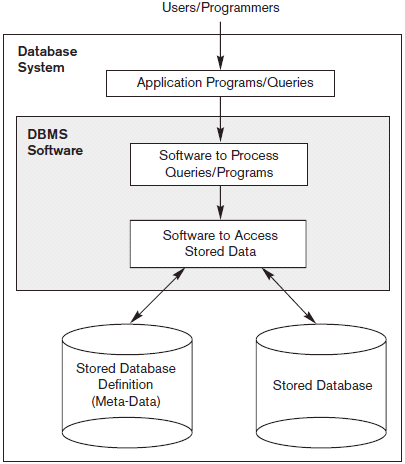
تتضمن عملية تعريف قاعدة بيانات تحديد: أنواع، بنى، والقيود على البيانات التي ستخزن في قاعدة البيانات، يتم أيضاً تخزين تعريف قاعدة البيانات أو معلومات التوصيف في قاعدة البيانات على شكل دليل أو قاموس قاعدة البيانات database dictionary.

تتضمن عملية معالجة قاعدة البيانات وظائف مثل الاستعلام من قاعدة البيانات لجلب معلومات محددة، تعديل قاعدة البيانات لإظهار التغيرات في العالم المصغّر، وتوليد التقارير عن البيانات.

يقوم برنامج تطبيقي بالوصول إلى قاعدة البيانات من خلال إرسال استفسارات queries أو طلبات requests عن البيانات إلى نظام إدارة قاعدة البيانات. يُسبب الاستعلام أو الاستفسار query جلب بعض البيانات. أما المناقلة transaction فقد تؤدي إلى قراءة أو كتابة بعض البيانات من \ إلى قاعدة البيانات.

من الوظائف المهمة الأخرى المتاحة في نظام إدارة قاعدة البيانات همليات حماية وحفظ قاعدة البيانات لفترة طويلة من الزمن. تتضمن عملية الحماية protection حماية النظام من سوء عمل، أو أذى البرمجيات والعتاد الصلب بالإذافة إلى الحماية الأمنية من الوصول غير المخول أو الخبيث. وبما أن قواعد البيانات الضحمة النموذجية تكون ذات عمر طويل بالتالي يجب أن يكون نظام إدارة قاعدة البيانات قادراً على حفظ نظام قاعدة البيانات من خلال جعل النظام قابلاً للتطور بحسب متطلبات التغيير بمرور الزمن.

نسمي قاعدة البيانات وبرمجيات نظام إدارة قاعدة البيانات بالاسم نظام قاعدة البيانات database system، يبين الشكل (4-1)هذا النظام:



### الشكل (4-1) نظام قواعد البيانات

تتضمن أغلب قواعد البيانات متوسطة الحجم والضخمة أنواعاً عديدة من السجلات وتملك العديد من العلاقات relationships بين السجلات انّ عمليات التحديث والاستعلام هذه يجب أن تُعرّف بدقة باستخدام لغة الاستعلام التي يتعامل معها نظام إدارة قاعدة البيانات قبل أن يتم تنفيذها أو معالجتها.

من الملائم في هذه المرحلة أن يتم وصف قاعدة البيانات على أنها جزء من نظام المعلومات Information System الخاص بالمؤسسة

# 4-4-مزايا أسلوب قاعدة البيانات:Characteristics of database approach

هناك العديد من الخصائص التي تميز أسلوب قواعد البيانات عن الأسلوب التقليدي للبرمجة باستخدام الملفات.

ففي الأسلوب التقليدي لمعالجة الملفات، يقوم كل مستخدم بتعريف وبناء الملفات اللازمة لتطبيق برمجي ما كجزء من برمجة هذا التطبيق. اما في اسلوب قواعد البيانات، يتم استخدام مخزن واحد للبيانات يتم تعريف لمرة واحدة ومن ثم الوصول إليه من قبل عدة مستخدمين.

وفي حين أنه في النظام القائم على استخدام الملفات يكون لدى كل تطبيق الحرية في تسمية عناصر البيانات بشكل مستقل، فإنه في النظام القائم على قواعد البيانات، يتم تحديد الأسماء مرة واحدة ومن ثم يستخدم بشكل متكرر في الاستعلامات، المناقلات، والتطبيقات.

من أهم مزايا أسلوب قاعدة البيانات مقارنة بالأسلوب القائم على الملفات:

* طبيعة التوصيف الذاتية لنظام قاعدة البيانات.
* الفصل بين البرامج والتطبيقات، وتجريد البيانات.
* دعم مشاهد متعددة للبيانات.
* مشاركة البيانات ومعالجة المناقلات متعددة المستخدمين.

نبين فيما يلي هذه المزايا بشيء من التفصيل.

## 4-4-1- طبيعة التوصيف الذاتية لنظام قاعدة البيانات Self\_Describing Nature of a Database System :

الميزة الأساسية لأسلوب قواعد البيانات هي أن نظام قاعدة البيانات لا يتضمن قاعدة البيانات فقط، وإنما يتضمن التعريف والتوصيف الكامل لبنية وقيود قاعدة البيانات. يتم تخزين هذا التعريف في دليل catalog نظام إدارة قاعدة البيانات، والذي يتضمن معلومات مثل، بنية كل ملف، نوع وشكل التخزين لكل عنصر بيانات، والقيود الأخرى على البيانات، تدعى المعلومات المخزنة في الدليل باسم بيانات التوصيف meta-data وهي تصف بنية قاعدة البيانات.

يتم استخدام هذا الدليل من قبل برمجيات إدارة قاعدة البيانات ومن قبل مستخدمي قاعدة البيانات الذين يحتاجون معلومات عن بنية قاعدة البيانات. نظراً لعدم وجود حزمة عامة، مما يتطلب معرفة ونوع وشكل البيانات التي يمكن الوصول إليها.

يجب أن تعمل برمجيات نظام إدارة قاعدة البيانات مع أي عدد لتطبيقات قاعدة البيانات، على سبيل المثال، قاعدة بيانات الجامعة، قاعدة بيانات بنك، قاعدة بيانات شركة، طالما أن تعريف قاعدة البيانات مخزن في الدليل. في الأسلوب التقليدي القائم على الملفات، يكون تعريف البيانات جزءاً من برامج التطبيقات بحد ذاتها. لذلك، تكون البرامج مقيدة بالعمل مع قاعدة بيانات واحدة محددة، وهي قاعدة البيانات التي تم التصريح عن بنيتها في برامج التطبيق.

## 4-4-2-الفصل بين البرامج وتجريد البيانات Insulation Between Programs And Data, And Data Abstraction :

في أسلوب معالجة الملفات التقليدي، تكون بنية ملفات البيانات مضمنة في برامج التطبيقات وبالتالي فأي تغييرات على بنية الملف قد تتطلب تغيير جميع البرامج التي تقوم بالوصول إلى ذلك الملف. بالمقابل، يقوم نظام إدارة قاعدة البيانات بالوصول إلى البرامج بغض النظر عن هذه التغيرات في أغلب الأحيان. يتم تخزين بنية ملفات البيانات في دليل نظام إدارة قاعدة البيانات بشكل منفصل عن برامج الوصول. ندعو هذه الخاصية استقلال البرامج عن البيانات program-data independence.

في بعض أنظمة قواعد البيانات، يمكن للمستخدمين القيام بتعريف العمليات على البيانات كجزء من تعريف قاعدة البيانات. يتم تحديد العملية operation وتدعى أيضاً التابع function أو الطريقة Method من خلال جزئين:

* واجهة العملية interfaceأو التوقيع signatureوتتضمن أسماء وأنواع بيانات وسطائها أو بارامتراتها prototype.
* تحقيق العملية implementationأو الطريقة method ) وتحدد بشكل مستقل ويمكن أن تعدل بشكل مستقل دون التأثير على الواجهة.

تُدعى الخاصية التي تتيح استقلال البرنامج عن البيانات واستقلال البرنامج عن العمليات باسم تجريد البيانات data abstraction.

يقدم نظام إدارة قاعدة البيانات للمستخدمين تمثيلاً مفاهيمياً للبيانات لايتضمن العديد من التفاصيل حول كيفية تخزين البيانات وتحقيق العمليات.

إنّ نموذج البيانات data model هو نوع من تجريد البيانات الذي يستخدم لتحقيق هذا التمثيل المفاهيمي. يستخدم نموذج البيانات مفاهيم منطقية، مثل الأغراض Objects، خصائصها، والعلاقات بينها، الأمر الذي يُسهل فهمها مقارنةً بمفاهيم التخزين الحاسوبي. يقوم نموذج البيانات بإخفاء التفاصيل المتعلق بالتخزين والتحقيق التي لا تهم أغلب مستخدمي قاعدة البيانات.

تخزن بنية وتنظيم كل ملف في الدليل( القاموس ) ويقوم نظام إدارة قاعدة البيانات باقتطاع تفاصيل تخزين الملف من الدليل عند الحاجة إليها من قبل الوحدات البرمجية التي تقوم بالوصول إلى الملفات في نظام إدارة قاعدة البيانات.

## 4-4-3- دعم مشاهد متعددة للبيانات Support of Multiple Views of the Data :

يكون عادةً لقاعدة البيانات عدة مستخدمين، وقد يكون لكل منهم منظوراً أو مشهداً view مختلفاً إلى البيانات. قد يكون المشهد مجموعة جزئية من قاعدة البيانات أو قد يحتوي على بيانات افتراضية virtual data مشتقة من ملفات قاعدة البيانات ولكنها غير مخزنة بشكل صريح. وبالتالي على نظام غدارة قاعدة البيانات متعدد المستخدمين أن يُؤمن ذلك.

## 4-4-4- مشاركة البيانات ومعالجة المناقلات متعددة المستخدمين Sharing of Data And Multiuser Transaction Processing :

يجب(3) أن يُتيح نظام إدارة قاعدة البيانات متعدد المستخدمين لعدد كبير من المستخدمين بالوصول إلى قاعدة البيانات في نفس الوقت، ويجب أن يتضمن برمجيات تحكم بالتزامن concurrency control لضمان أنه عند قيام عدة مستخدمين بمحاولة تعديل نفس البيانات فإن ذلك يتم بطريقة متحكم بها بحيث تتم عملية التعديل بنجاح. على سبيل المثال، عندما يقوم عدة موظفي حجز بمحاولة حجز مفعد على متن إحدى الرحلات، يجب أن يضمن نظام إدارة قاعدة البيانات أن المقعد يمكن الوصول إليه من قبل موظف واحد في نفس الوقت ولمدة زمنية معينة. يُدعى هذا النوع من التطبيقات عادةً باسم تطبيقات معالجة المناقلات عبر الأنترنت ( OLTP ) Online Transaction Processing. , الدور الأساسي لنظام إدارة قاعدة البيانات متعدد المستخدمين هو ضمان أن المناقلات المتزامنة تعمل بشكل صريح وفعال.

إنّ المناقلة هي برنامج تنفيذي أو عملية تتضمن وصولاً أو أكثر إلى قاعدة البيانات، كقراءة أو تعديل سجلات قاعدة البيانات.

يجب أن يضمن العديد من الخصائص للمناقلة. خاصية الفصل insulation تضمن أن كل مناقلة تُنفذ بشكل منفصل عن باقي المناقلات حتى لو وجد مئات المناقلات التي تُنفذ بنفس الوقت. أما خاصية الـــــ atomicity تضمن أن جميع عمليات قاعدة البيانات في مناقلة ما قد نُفذت أم لا.

# 4-5- العاملون في قاعدة البيانات Actors on the DataBase :

في قاعدة البيانات الشخصية البسيطة هناك شخص يبني ويستخدم قاعدة البيانات العناوين في حين أنه، في المنظمات الضخمة، يكون هناك عدة أشخاص مشتركين في تصميم استخدام، وصيانة قاعدة بيانات ضخمة ذات مئات مستخدمين.

# 4-6- مدراء قاعدة البيانات Database Administrators :

في بيئة قواعد قاعدة البيانات، المورد الرئيسي هو قاعدة البيانات بحد ذاتها، والمورد الثاني هو نظام إدارة قاعدة البيانات والبرمجيات المرتبطة به. إنّ إدارة هذه الموارد هي من مسؤولية مدير قاعدة البيانات ( DBA ) DataBase Administrators.

إنّ مدير قاعدة البيانات DBA مسؤول عن الإشراف وإدارة الموارد مثل: تخويل الوصول إلى قاعدة البيانات، تنسيق ومراقبة استخدامها، والحصول على البرمجيات والعتاد الصلب اللازمة. إنّ مدير قاعدة البيانات مسؤول عن المشاكل مثل خرق الأمن أو ضعف زمن استجابة النظام , يتم معاونة مدير قاعدة البيانات في المنظومات الضخمة من قبل فريق عمل يقوم بتنفيذ هذه المهام.

# 4-7- مصممو قاعدة البيانات : Database Designers

يكون مصمو قاعدة البيانات مسؤولون عن تعريف البيانات التي ستخزن في قاعدة البيانات وعن اختيار البنية الملائمة لتمثيل وتخزين هذه البيانات. وتنفيذ المهمة بغالبها قبل تزويدها بالبيانات.

إنّ مصممو قاعدة البيانات مسؤولون عن التواصل مع جميع مستخدمي قاعدة البيانات المحتملين من أجل فهم متطلباتهم وبناء تصميم يحقق هذه المتطلبات ويبنون مشاهد views خاصة بقاعدة البيانات تحقق متطلبات البيانات ومعالجتها من قبل المجموعة المحتملة يجب على التصميم النهائي قادراً على دهم متطلبات جميع المستخدمين. وهم في الغالب ضمن فريق عمل مدير قاعدة البيانات.

# 4-8- البنية ثلاثية المخططات واستقلال قاعدة البيانات Three-Schema Architecture and Data Independence :

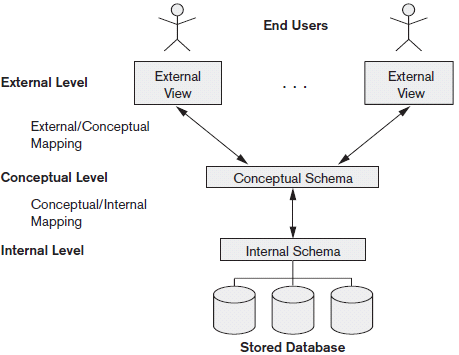
إنّ البنية ثلاثية المخططات اقترحت للمساعدة في تحقيق وتصور هذه المزايا.

## 4-8-1- البنية ثلاثية المخططات The Three-Schema Architecture:

تهدف(3) البنية ثلاثية المخططات المبينة في الشكل (4-2 ) إلى الفصل بين تطبيقات المستخدم وقاعدة البيانات الفيزيائية. في هذه البنية، تُعرّف المخططات عند المستويات الثلاثة التالية:

* يملك المستوى الداخلي internal level مخططاً داخلياً internal schema، وهو يصف بنية التخزين الفيزيائية لقاعدة البيانات. يصف تفاصيل التخزين ومسارات الوصول إلى قاعدة البيانات.
* يملك المستوى المفاهيمي conceptual level مخططاً مفاهيمياً conceptual schema يصف البنية الكاملة لقاعدة البيانات لمجموعة من المستخدمين. يقوم المخطط المفاهيمي بإخفاء تفاصيل بنى التخزين الفيزيائي ويركز على الكينونات الواصفة، أنواع البيانات، العلاقات، عمليات المستخدم، والقيود.
* يتضمن المستوى الخارجي external level أو مستوى المشهد view level عدداً من المخططات الخارجية external schema أو مشاهد المستخدم user views. يصف كل مخطط خارجي الجزء من قاعدة البيانات الخاص بمجموعة مستخدمين خاصة ويخفي باقي قاعدة البيانات عن هذه المجموعة. وكما في الحالة السابقة.

تحتل البنية ثلاثية المخططات مكاناً هاماً في تكنولوجيا تطوير قواعد البيانات لأنها تفصل بشكل واضح مستوى المستخدمين الخارجي، مستوى النظام المفاهيمي، ومستوى التخزين الداخلي لتصميم قاعدة بيانات. حالياً في أغلب نظم إدارة قواعد البيانات التي تدعم مشاهد المستخدمين، تُعرّف المخططات الخارجية في نفس نموذج البيانات الذي يصف المستوى المفاهيمي يستخدم نظام إدارة قاعدة بيانات علائقي مثل ORACLE لغة SQL لتحقيق ذلك.



#### الشكل(4-2) البنية ثلاثية المخططات

[*DB-Engines Ranking"*](https://web.archive.org/web/20180731161212/https:/db-engines.com/en/ranking). January 2013. مؤرشف من [*الأصل*](https://db-engines.com/en/ranking) في 31 يوليو 2018. اطلع عليه بتاريخ 22 يناير 2013

## 4-8-2- استقلال البيانات Data Independence :

يمكن أنّ تستخدم البنية ثلاثية المخططات لتوضيح مفهوم استقلال البيانات بشكل أكبر.

يمكن تعريف نوعين من استقلال البيانات:

* استقلال البيانات المنطقي logical data independence وهو القدرة على تعديل المخطط المفاهيمي بدون تعديل المخططات الخارجية أو برامج التطبيقات. قد يكون بغية توسيع قاعدة بيانات بإضافة نوع سجل أو عنصر بيانات ، لتعديل القيود، أو لاختصار قاعدة البيانات بحذف نوع سجل أو عنصر بيانات.
* استقلال البيانات الفيزيائي physical data independence هو القدرة على تعديل المخطط الداخلي بدون تعديل المخطط المفاهيمي. وبالتالي، فإن المخطط الخارجي لا يحتاج لأن يتغير. ولا تتأثر بطرق وصول جديدة.

عندما يكون لدينا نظام إدارة قاعدة البيانات متعدد المستويات، فإن دليلها يجب أن يتم توسيعه ليتضمن معلومات عن كيفية تخطيط الطلبات والبيانات بين المستويات المختلفة. يستخدم نظام إدارة قاعدة البيانات برمجيات إضافية لتحقيق هذه التخطيطات من خلال الإشارة إلى معلومات التخطيط ضمن الدليل.

يحدث استقلال البيانات لأنه عندما يتم تغيير المخطط schema عند مستوى ما، فإنّ المخطط عن المستوى التالي الأعلى يبقى بدون تغيير، وإنما فقط التخطيط mapping بين هذين المستويين يتغير. لذا فإنّ برامج التطبيقات التي تُشير إلى المخطط ذو المستوى الأعلى لا تحتاج لأن تتغير.

إنّ البنية ثلاثية المخططات يمكن أن تجعل من الأسهل الحصول على استقلال حقيقي للبيانات سواء كان فيزيائياً أو منطقياً.

في حين أن التخطيطات ثنائية المستوى تولد عبئاً إضافياً خلال عملية ترجمة وتنفيذ استعلام أو برنامج، مما يؤدي إلى عدم كفاءة نظام إدارة قاعدة البيانات DBMS.

# 4-9- نمذجة البيانات باستخام ترابط الكينونات ( ( ER :

## 4-9-1- مقدمة :

نعتبر النمذجة المفاهيمية conceptual modeling مرحلة هامة جداً من مراحل تصميم وتطبيق قواعد بيانات ناجحة. عموماً، يُشير مصطلح تطبيق قواعد بيانات DataBase Application إلى قاعدة بيانات محددة والبرامج المرافقة التي تُحقق استعلامات وتحديثات قاعدة البيانات.

على سبيل المثال، تطبيق قاعدة بيانات BANK يحفظ بيانات حسابات الزبائن قد يتضمن برامج تحقق عمليات تحديث قاعدة البيانات المتعلقة بإيداعات الزبون وسحوباته. تُتيح هذه البرامج واجهات مستخدم رسومية سهلة الاستخدام تقوم باستخدام نماذج وقوائم من أجل المستخدمين النهائيين للتطبيق على سبيل المثال، أمناء الصناديق. وبالتالي فإنّ جزءاً من تطبيق قاعدة البيانات سيتطلب تصميم، تحقيق، واختبار برامج التطبيق هذه. تقليدياً، اعتبر تصميم واختبار برامج التطبيقات أمراً ينتمي إلى هندسة البرمجيات وليس إلى مجال قواعد البيانات.

بما أن طرق تصميم قواعد البيانات تتضمن أكثر من مجرد مفاهيم تحديد العمليات على أغراض قاعدة البيانات، وبما أن طرق هندسة البرمجيات تحدد بنية قاعدة البيانات خلال مرحلة تصميم قاعدة البيانات التي ستقوم البرامج باستخدامها والوصول إليها بتفصيل أكبر، وبالتالي فإنه من الواضح أن هذه العمليات مترابطة بقوة.

خلال مرحلة تصميم قاعدة البيانات. يتم التركيز على بنى وقيود قاعدة البيانات. حيثُ تعرض في هذا الفصل مفاهيم النمذجة الخاصة بنموذج ترابط الكينونات ER model، وهو عبارة عن نموذج مفاهيمي عالي المستوى.

لقد استخدم هذا النموذج وأشكاله المختلفة من أجل التصميم المفاهيمي لتطبيقات قواعد البيانات، حيثُ تُستخدم العديد من أدوات تصميم قواعد البيانات مفاهيمه.

**الفصل الخامس**

**تحليل النظام**

# 5-1- المقدمة :

انطلاقا من متطلبات العمل نقدم في هذا الفصل للتعرّف على منصة العمل التي من شأنها تقديم طيف واسع من الخدمات إلى العملاء وتقديم تكنولوجيا المعلومات ..حيث نوضح في هذا الفصل أدوات تخطيطية )رسومية( متقدمة في تحليل وتوصيف النظام مثل لغة النمذجة الموحدة UML ومخططات نذمجة البيانات: مخططات ترابط الكينونات ERD .

# 5-2-حالات الاستخدام الأساسية:

## **5-2-1-حالات استخدام الخاصة بالمدير :**

حالات الاستخدام الاساسية لمستخدم النظام ويتألف الحالات المبينة في المخطط التالي :

## **5-2-2-حالات استخدام المسؤول عن قسم :**

نوضح في المخطط التالي الحالات الاساسية التي يتكون منها مسؤول النظام :

# 5-2-3 مخطط حالة الدكتور :

**الفصل السادس**

**القسم العملي**

# 6-1 واجهات التطبيق :

**الواجهة الرئيسية :**

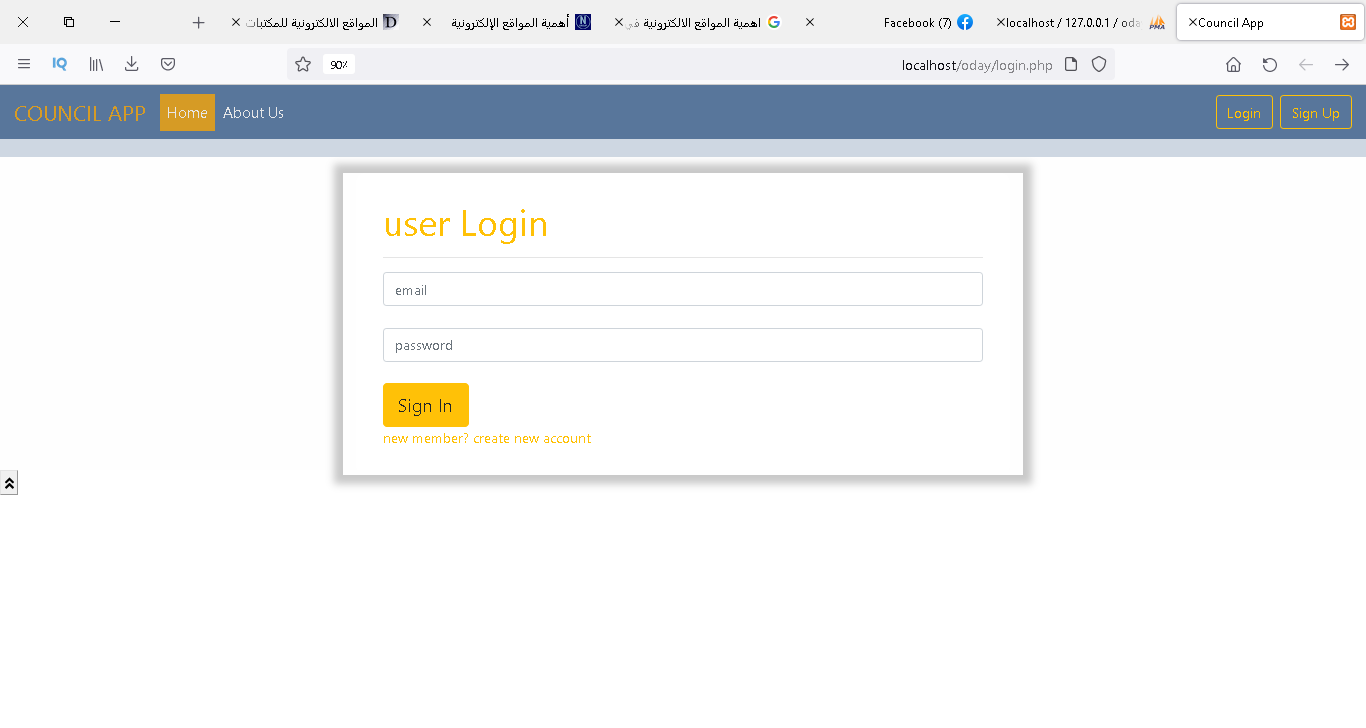
صفحة البداية للمشروع كما في الشكل ( 6-1 ) التالي :



#### الشكل (6-1) الواجهة الرئيسية

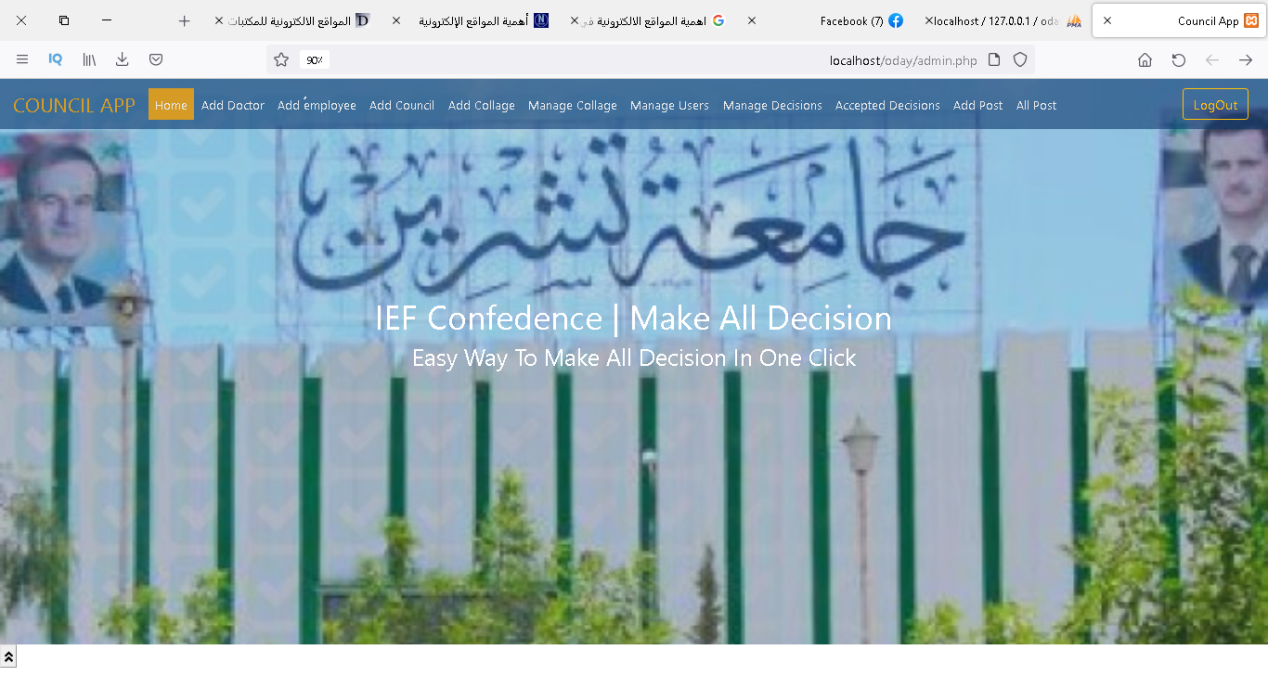
وهي الواجهة التي يتم من خلالها الدخول الى الموقع ,والتوجه لانشاءحساب او تسجيل الدخول

فاذا تم الضغط على تسجيل الدخول سينتقل الى الصفحة التاليه في الشكل (6-2) :



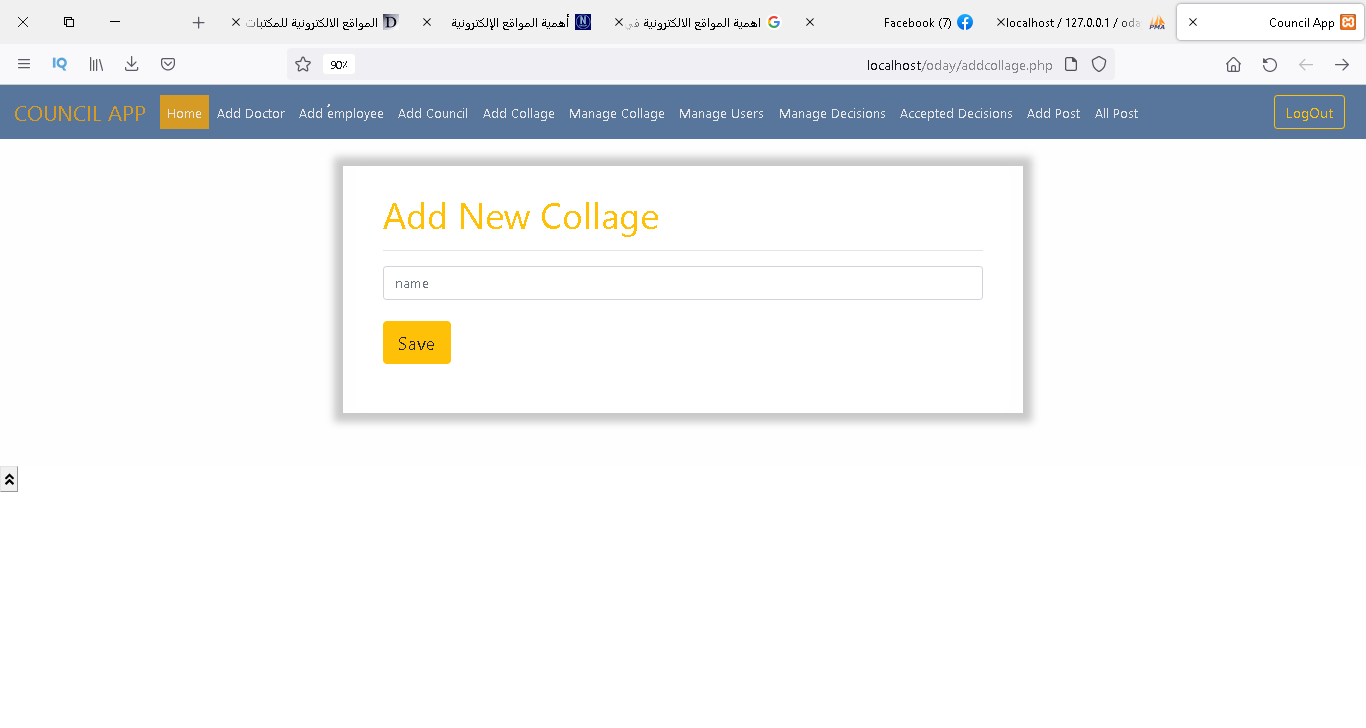
#### الشكل (2-6) تسجيل دخول

بعد اختيار طبيعة الدخول عبر واجهة ال ( log in ) , فاذا كان الاختيار مسؤول( admain) عندها يستطيع المسؤول اضافة او حذف او تعديل رئيس قسم او دكتور ما او كلية او قسم ضمن كلية حيث ستنتقل بنا الى الواجهة في الشكل (6-4) التالية :



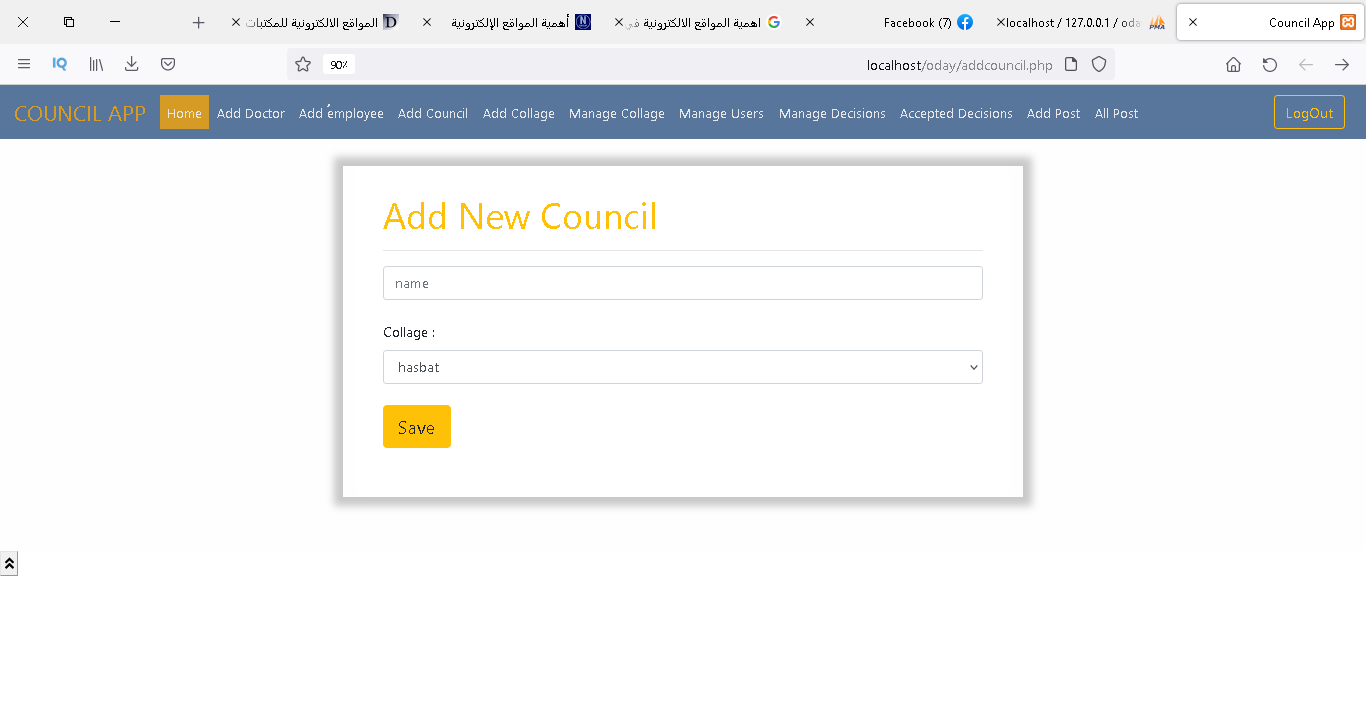
#### الشكل (6-3) واجهة المدير الرئيسية

وعند الضغط على زر ( add collage) سوف تظهر لنا صفحة يتم من خلالها اضافة بيانات كلية جديدة الى قائمة الكليات ضمن الموقع وذلك بعد تعبئة البيانات بحيث يتوجب ملىء بيانات جميع الحقول :



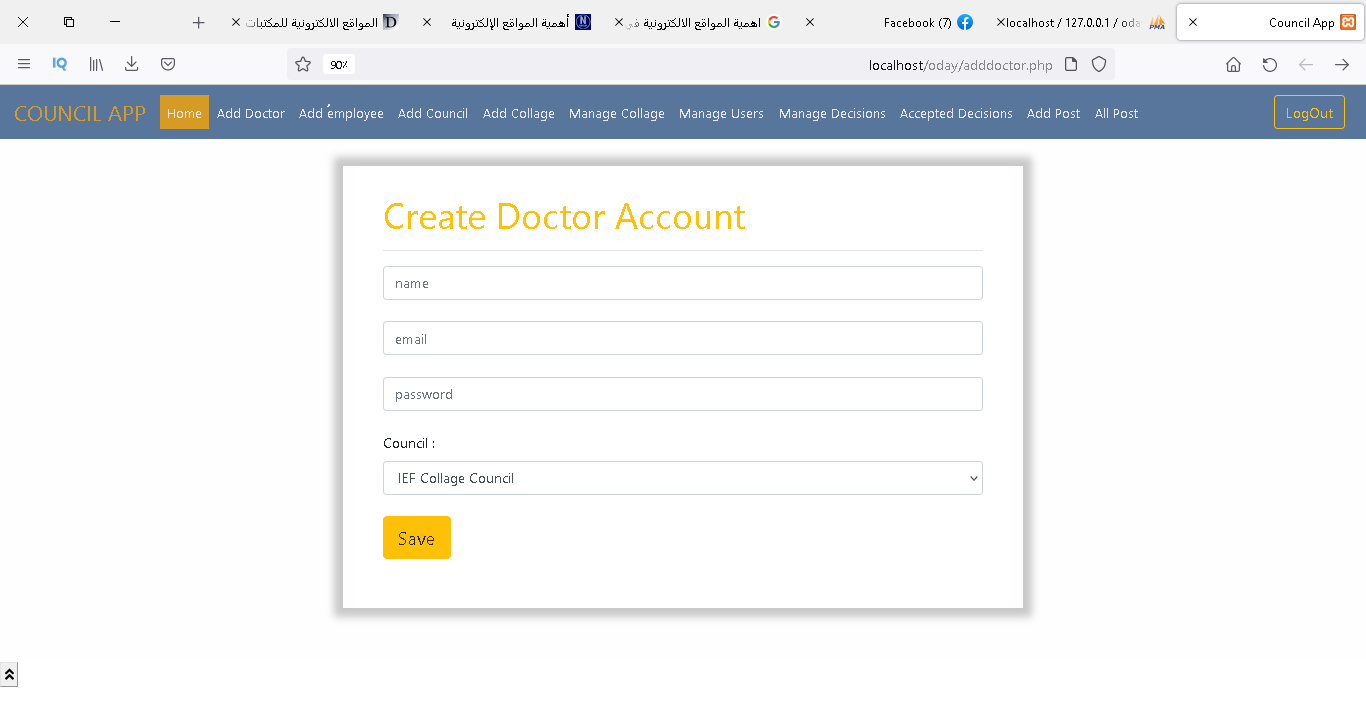
#### الشكل (6-4) واجهة اضافة كلية جديد

كما يمكن اضافة قسم ضمن الكلية من خلال الواجهة التالية



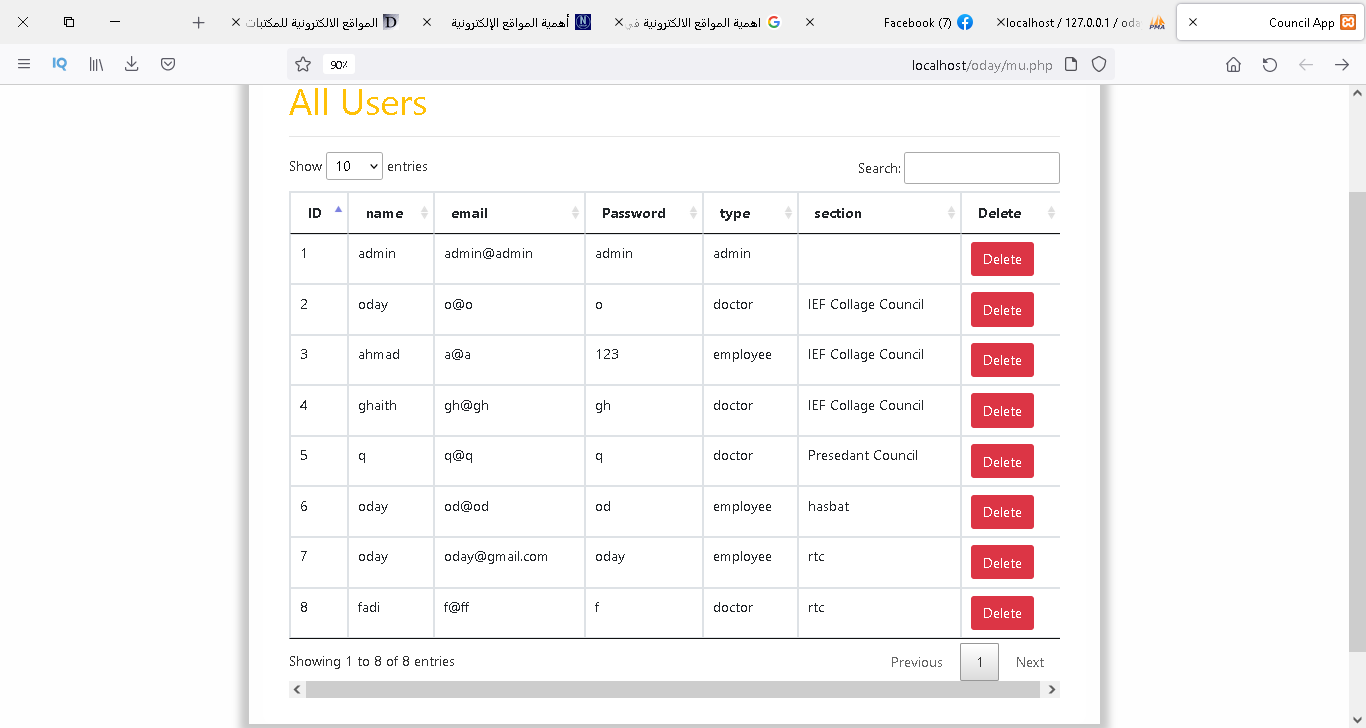
#### الشكل (6-5) واجهة اضافة قسم

ويمكن اضافة دكتور بعد اضافة كافة البيانات اللازمة



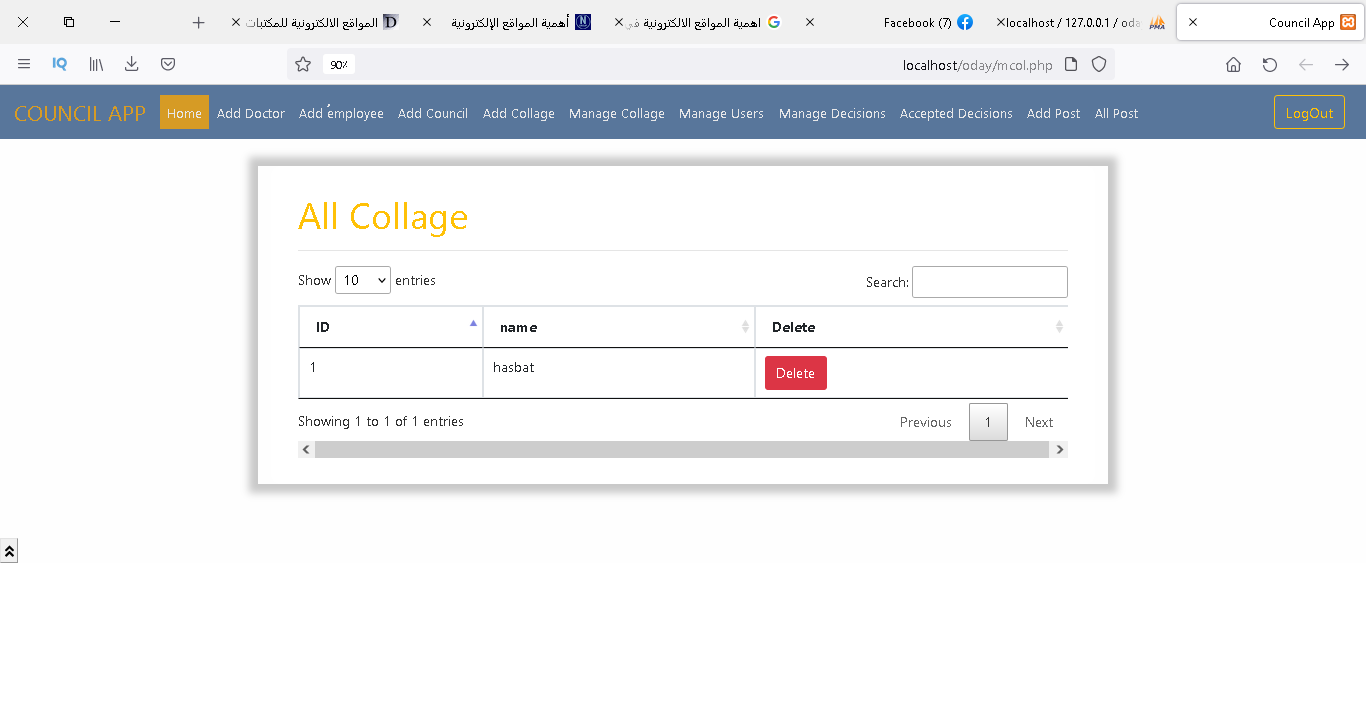
#### الشكل (6-6) واجهة اضافة دكتور

أيضا لدينا واجهة ادارة كافة المستخدمين كما في الشكل (6-7) :



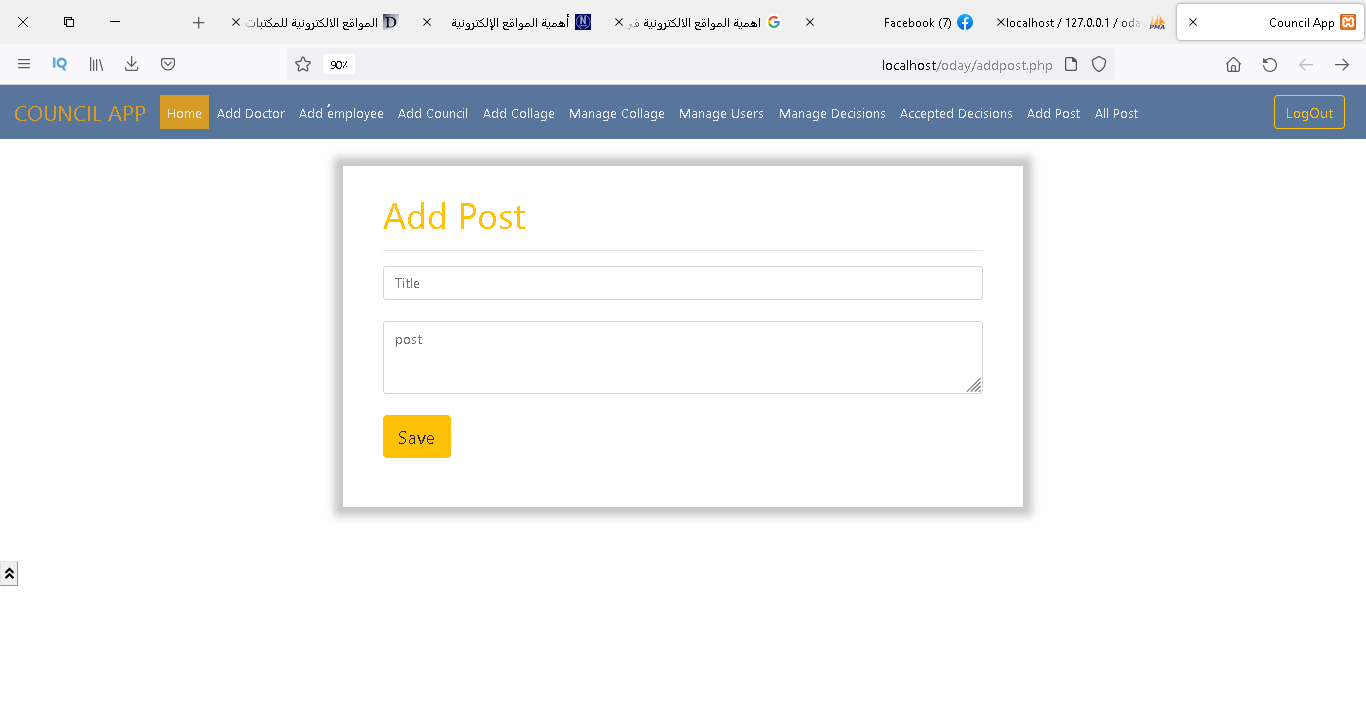
#### الشكل (6-7) واجهة ادارة المستخدمين

يمكننا الموقع من ادارة الكليات ايضا :



### الشكل (6-8) واجهة ادارة الكليات

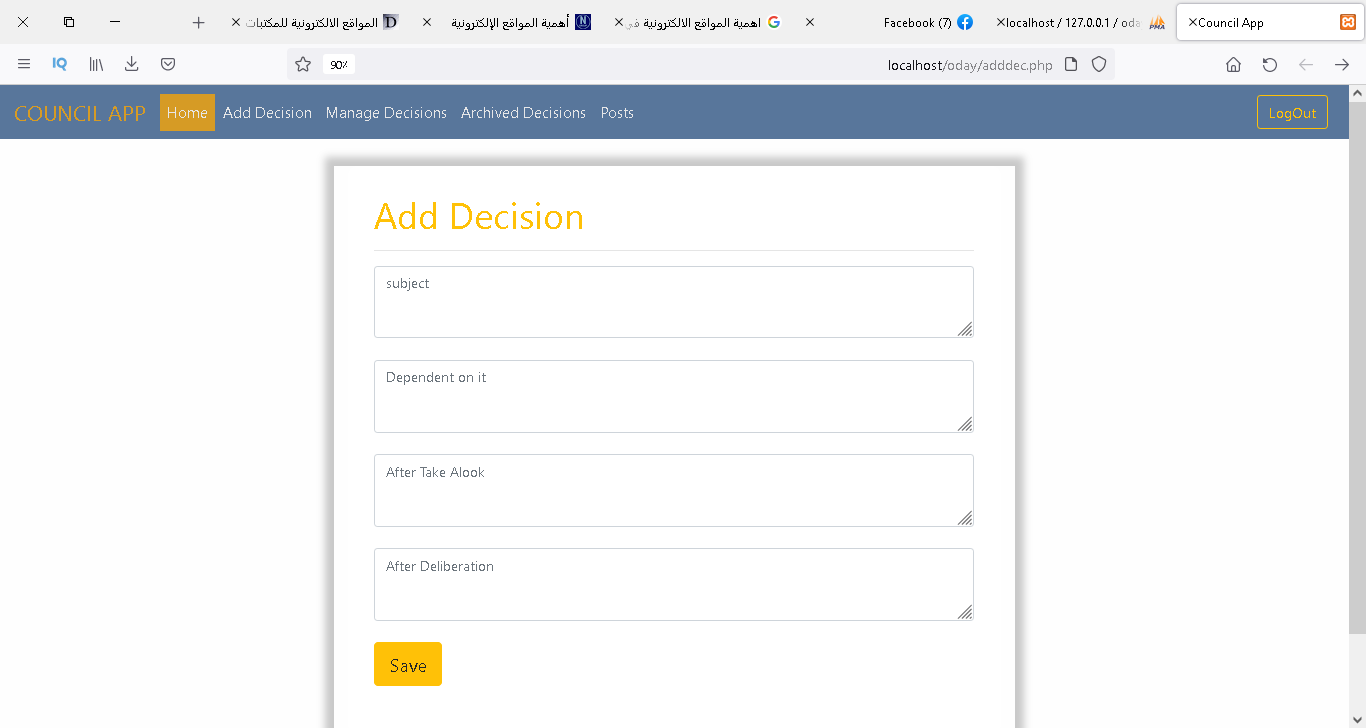
حيث يمكن ايضا اضافة اعلان جديد :



#### الشكل (6-9) واجه اضافة اعلان

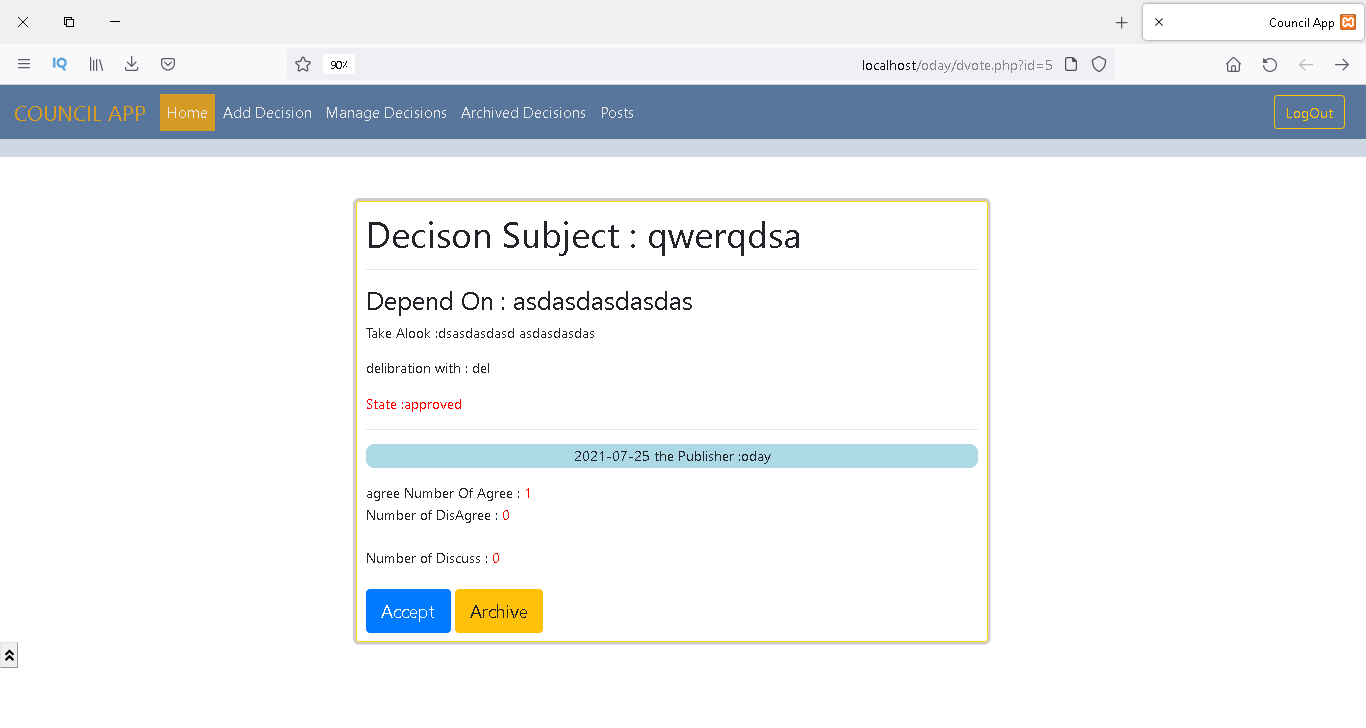
في حال تم تسجيل الدخول كرئيس لقسم ما تظهر الواجهة التالية :

التي تمكنه من طرح قرار جديد



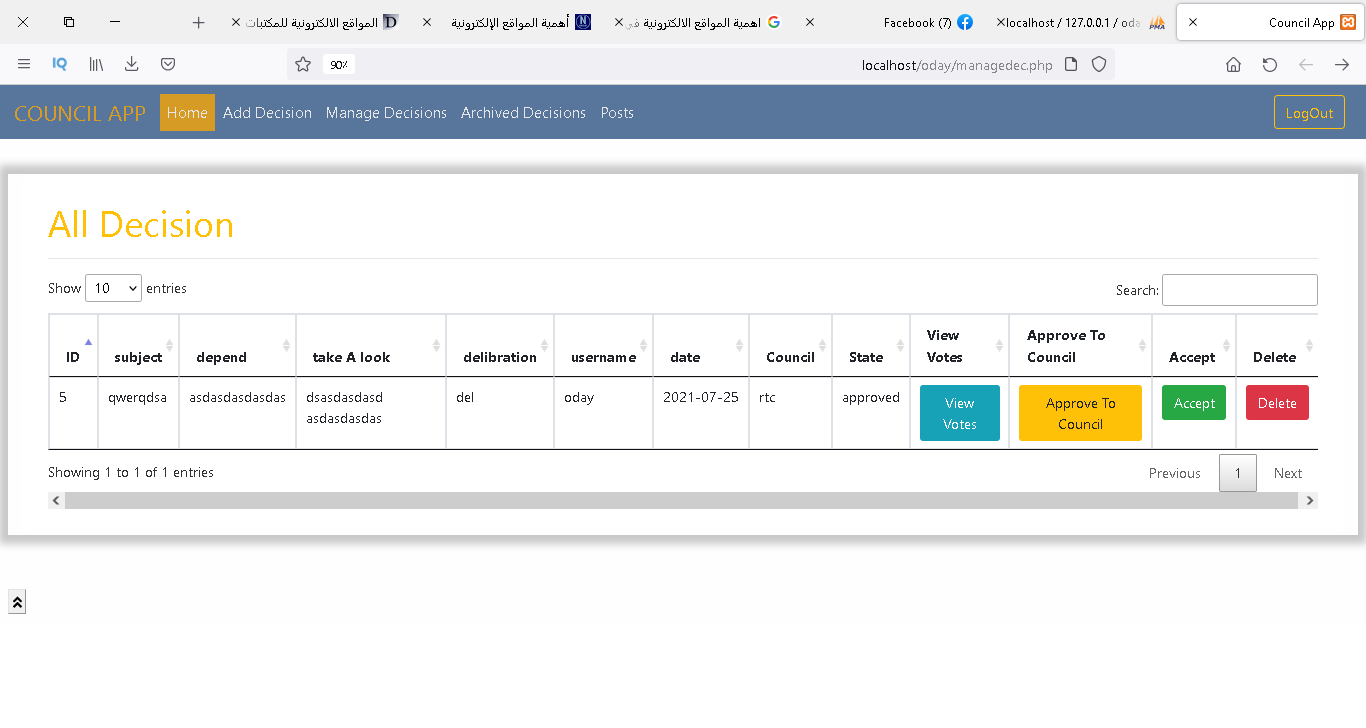
#### الشكل (6-10) واجهة اضافة قرار

كما يمكن لرئيس اللقسم ان يعرض الاصوات لقرار ما :



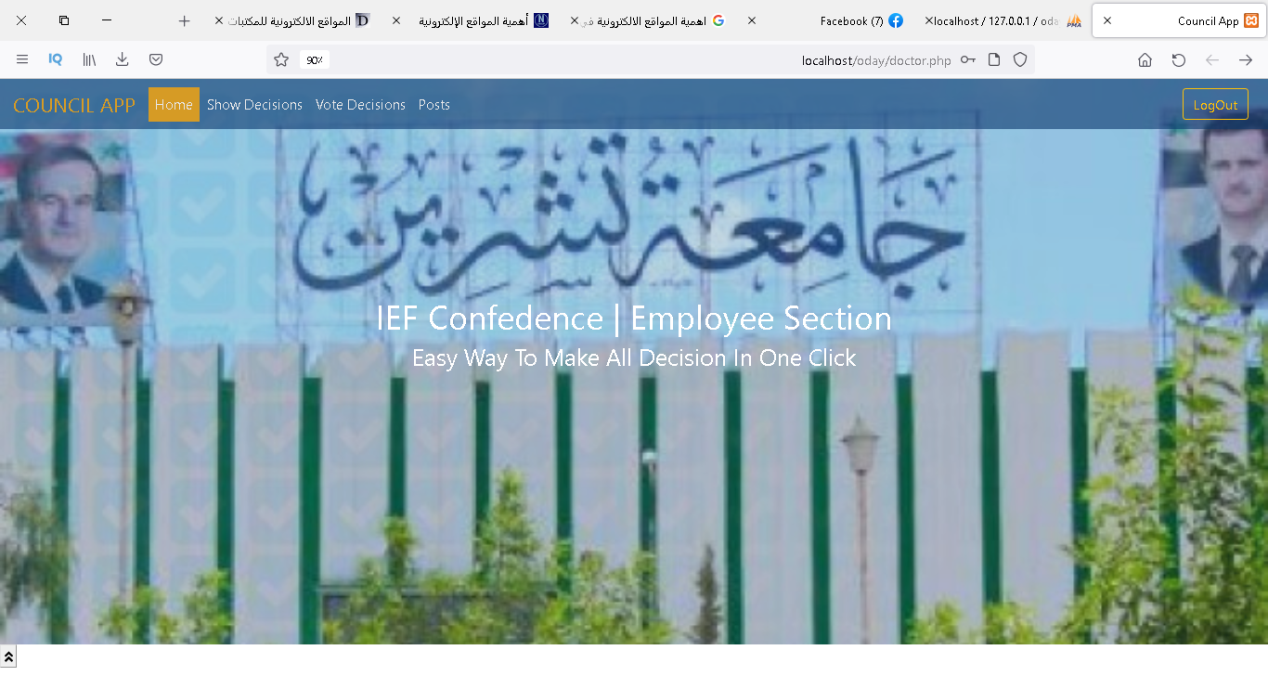
#### الشكل (6-11) واجهة عرض اصوات قرار ما

ويمكنه ايضا عرض كل قرارت القسم وادارتها :

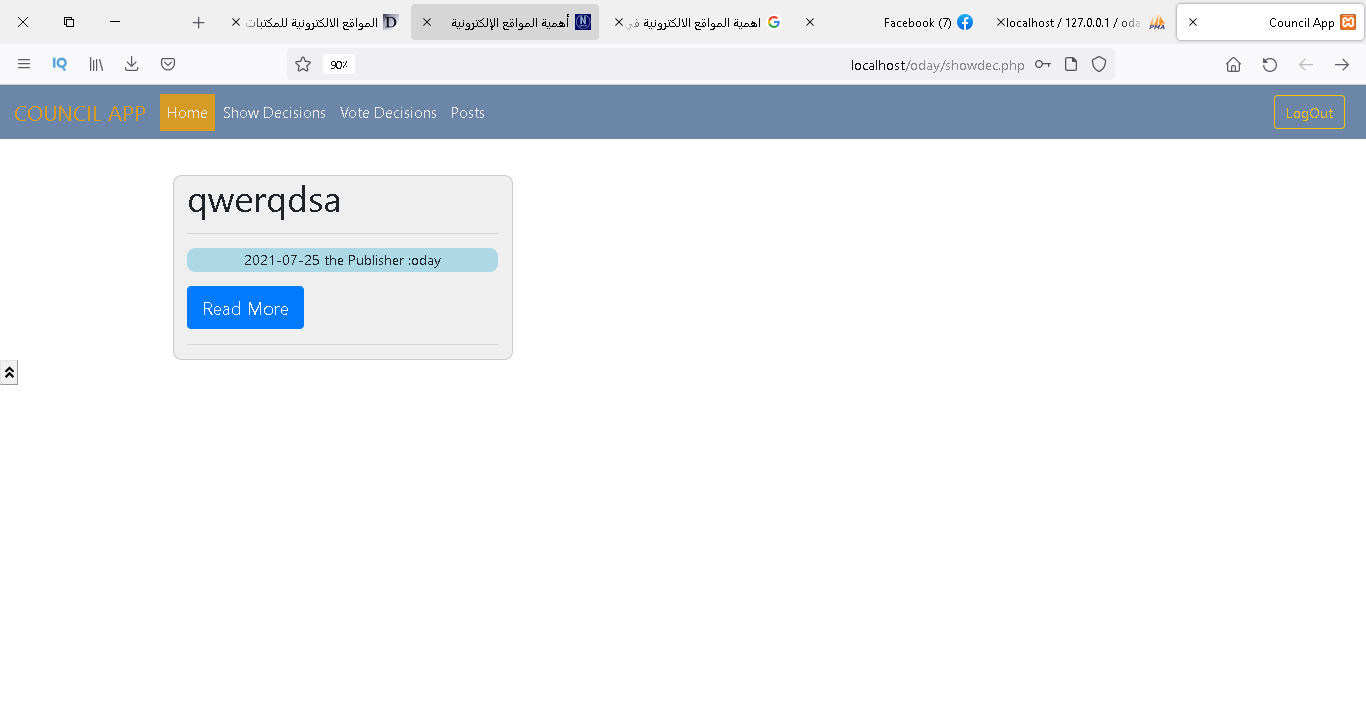


#### الشكل (6-12) واجهة عرض كل القرارت

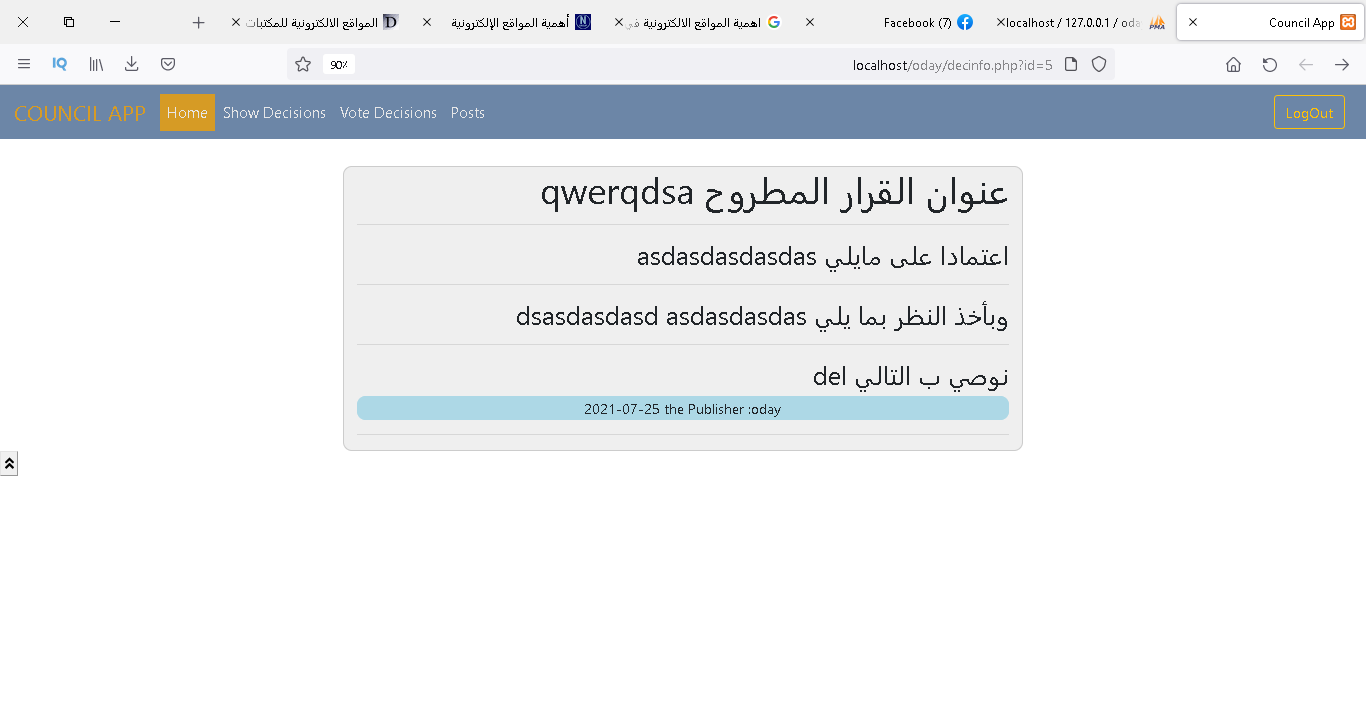
اما عند تسجيل الدخول كدكتور ضمن قسم ما تظهر الواجهة الرئيسية التالية :



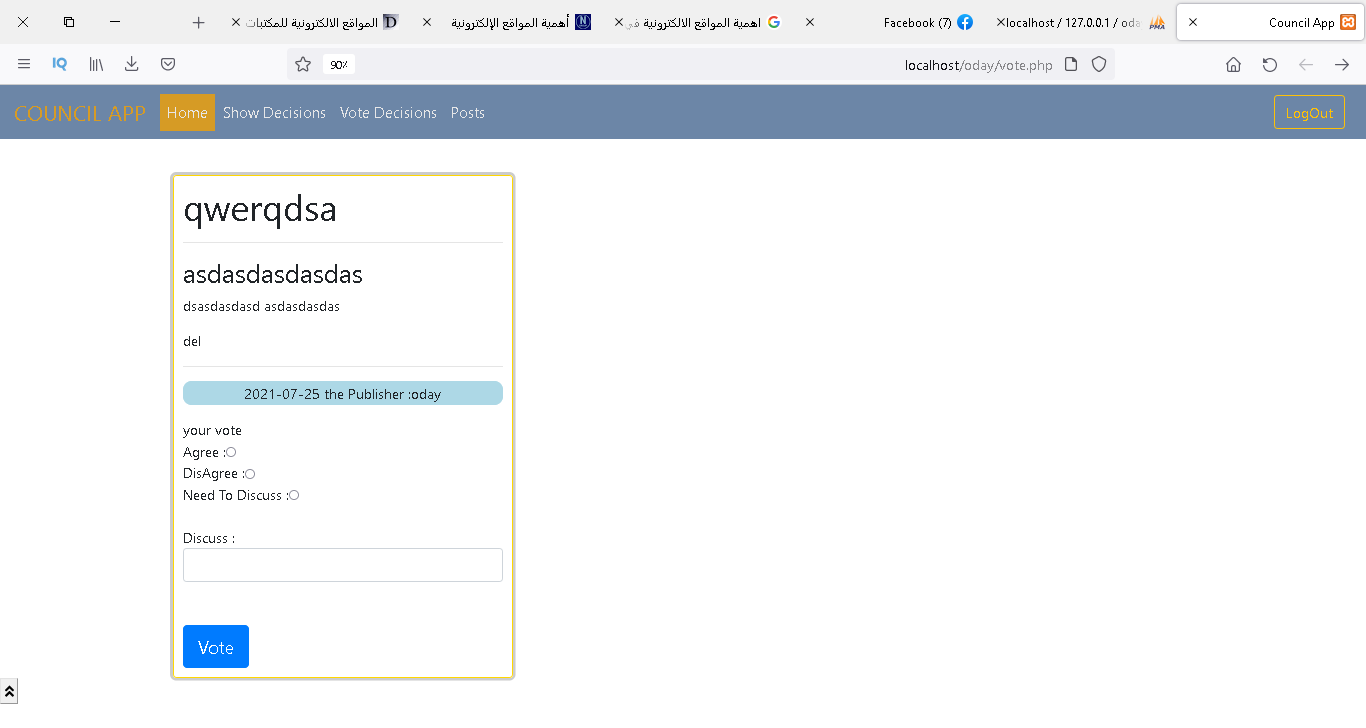
التي تمكنه من عرض القرارات كافة :



وعند الضغط على زر قراءة المزيد يتم عرض واجهة القرار الرسمية :



كما يمكنه التصويت على القرارت الموجودة ضمن القسم الذي يعمل ضمنه :



**الفصل السابع**

**الخاتمة**

# 7-1-المشاكل التي واجهت المشروع:

* واجهتنا مشكلة تعلم برامج جديدة خلال وقت قصير.
* مشكلة ربط البرمجيات المستخدمة مع أنظمة الحواسيب المتاحة.

# 7-2-إمكانية تطوير المشروع:

* إمكانية إرسال رسائل قصيرة للمستخدمين بأهم المشاريع والبحوث المتوفرة
* إنشاء تطبيق موبايل يحاكي عمل الموقع الالكتروني وتسهيل العمليات بالنسبة للمستخدمين والمدير .
* اضافة خوارزميات ذكاء صنعي للبحث وتصنيف القرارات وفقا لاهميتها
* اضافة الية تواصل سريع بين المستخدمين
* التعامل مع التخزين السحابي لتخزين صور عن كافة القرارات

# 7-3-الخاتمة:

* لقد تطور قطاع تكنولوجيا المدعومات وقواعد البيانات وأتمته النظم الصناعية تطورا كبيرا وأصبح من أكثر انواع العلوم انتشارا فقد أصبح العامل الاساسي في تقدم سائر العلوم والمجالات.
* مما تقدم يمكننا القول إن هذا المشروع الذي هو أحد تطبيقات هندسة المعلومات وقواعد البيانات يوضح وبشكل مفصل خطوات بناء نظام مؤتمت وربطه بقاعدة بيانات وواجهات سهلة وسلسة في التعامل مع المستخدم حيث بدئنا بتحليل النظام وتحديد متطلباته ثم تحديد البرمجيات اللازمة ثم بنينا قاعدة البيانات وانتهينا بالواجهات.

# المراجع :

**1-د. علي سليمان، مدخل إلى قواعد البيانات، جامعة تشرين 2015**

**2-الشامل في إدارة قواعد البيانات ترجمة واعداد احسان الرفاعي \_نهى حنقون.**

**3-** **cs.tsu.edu/ghemri/CS681/ClassNotes/Intro\_Databases.pdf**

**بعض المواقع التي استفدنا منها في حل بعض المشكلات البرمجية:**

**4-http://stackoverflow.com**

**5-http://www.codeproject.com/KB/cs**

المراع العلمية " :

Raggett, Dave (1998). [*Raggett on HTML 4*](https://web.archive.org/web/20191118082454/http:/www.w3.org/People/Raggett/book4/ch02.html). مؤرشف من [*الأصل*](https://www.w3.org/People/Raggett/book4/ch02.html) في 18 نوفمبر 2020. اطلع عليه بتاريخ *July 9,* 2007.

Arnoud Engelfriet. [*"Introduction to Wilbur"*](https://web.archive.org/web/20171014054700/http:/htmlhelp.com/reference/wilbur/intro.html). Web Design Group. مؤرشف من [*الأصل*](http://htmlhelp.com/reference/wilbur/intro.html) في 14 مايو 2021. اطلع عليه بتاريخ 16 يونيو 2007

[*W3C Html"*](https://web.archive.org/web/20190810202506/https:/www.w3.org/html/). مؤرشف من [*الأصل*](https://www.w3.org/html/) في 10 أغسطس 2019.

Tim Berners-Lee, "Information Management: A Proposal." CERN (March 1989, May 1990). [W3.org](https://www.w3.org/History/1989/proposal.html) [نسخة محفوظة](https://web.archive.org/web/20180626035335/https:/www.w3.org/History/1989/proposal.html) 26 يونيو 2021 على موقع [واي باك مشين](https://ar.wikipedia.org/wiki/%D9%88%D8%A7%D9%8A_%D8%A8%D8%A7%D9%83_%D9%85%D8%B4%D9%8A%D9%86).

[*DB-Engines Ranking"*](https://web.archive.org/web/20180731161212/https:/db-engines.com/en/ranking). January 2013. مؤرشف من [*الأصل*](https://db-engines.com/en/ranking) في 31 يوليو 2021. اطلع عليه بتاريخ 22 يناير 2013

[*ACID Transactions, MarkLogic"*](https://web.archive.org/web/20160602151816/http:/www.marklogic.com/what-is-marklogic/features/acid-transactions/). مؤرشف من [*الأصل*](https://www.marklogic.com/product/marklogic-database-overview/database-features/) في 02 يونيو 2016. اطلع عليه بتاريخ 10 مارس 2015.