

جامعة الشام الخاصة كلية الهندسة المعلوماتية

**إظهار نتائج اختبار الأداء للنظم المؤسساتية الداعمة للويب**

**System**

إشراف

د. عمار جوخدار

تقديم

إيناس محمد سمير الصباغ رنيم مظهر دغوظ

14452021-

كلمة شكر

نتقدم بجزيل الشكر والامتنان إلى الدكتور عمار جوخدار لإشرافه على إنجاز هذا العمل ولمعلوماته القيّمة والدقيقة التي أفادنا بها. ويسعدنا أيضاً أن نتوجه بالشكر إلى المهندسة هدى ساغرجي. كما نتقدم بالشكر إلى الدكتور عميد كلية الهندسة المعلوماتية جورج كراز للتسهيلات التي قدمها لطلاب السنة الخامسة عموماً. ونشكر أيضاً كل من ساهم في إنجاز هذا المشروع.

# الخلاصة

يهدف المشروع إلى تصميم وبناء تقارير تبين: هل تحقق أن النظام قد نفذ مهام محددة بزم محدد على عتاديات محددة، بوجود عدد محدد من المستخدمين المتنافسين يقومون بعدد محدد من العمليات وعلى قاعدة بيانات تتضمن حجم معطيات يمثل عشر سنوات للأمام، ويؤخذ بعين الاعتبار المعطيات الفعالة وغير الفعالة في قاعدة البيانات بدون أخطاء وبحال عدم تحقق المعايير المتفق عليها سيتم تحديد نوع المشكلة (بطئ أو خطأ) واقتراح حلول لحل المشكلة.

**Abstract**

The project aims to design and build .

المحتويات

[الخلاصة ii](#_Toc50056791)

[قائمة الأشكال v](#_Toc50056792)

[قائمة الجداول vi](#_Toc50056793)

[الاختصارات vii](#_Toc50056794)

[المقدّمة viii](#_Toc50056795)

[الفصل الأوّل متطلَّبات المشروع 1](#_Toc50056796)

[1.1-الهدف من المشروع 1](#_Toc50056797)

[2.1-المتطلبات الوظيفية 1](#_Toc50056798)

[3.1-المتطلبات غير الوظيفية 3](#_Toc50056799)

[1.3.1-متطلبات البيئة التقنية 3](#_Toc50056800)

[2.3.1-المتطلبات الأمنية 3](#_Toc50056801)

[3.3.1-متطلبات التخزين وإدارة الذاكرة 3](#_Toc50056802)

[الفصل الثاني الدراسة المرجعية 4](#_Toc50056803)

[2.1-مقدّمة 4](#_Toc50056804)

[2.2-لمحة عن تطبيقات الويب 4](#_Toc50056804)

[2..21-بيئة العمل Elixir 4](#_Toc50056804)

[2.3-لمحة عن الهواتف الذكيةSmart Phone 5](#_Toc50056805)

[23..1-نظام التشغيل (flutter)Android 7](#_Toc50056807)

[2.3.2- بنية نظام تشغيل Android 8](#_Toc50056808)

[4.2-خدمات الويب Web Services 9](#_Toc50056809)

[2.4.2-Application Programming Interface) API 9](#_Toc50056810)

[2.4.1 بنية المعطيات JSON 9](#_Toc50056811)

[الفصل الثالث إدارة المشروع 11](#_Toc50056812)

[1.3-مقدّمة 11](#_Toc50056813)

[2.3-خطة إدارة المشروع 11](#_Toc50056814)

[الفصل الرابع تحليل المتطلبات 12](#_Toc50056815)

[1.4-مقدّمة 12](#_Toc50056816)

[2.4-التطبيقات المشابهة 12](#_Toc50056817)

[3.4-مخطط حالات الاستخدام"Usecase Digram" 14](#_Toc50056818)

[4.4-توصيف حالات الاستخدام"Narrative Use Case" 13](#_Toc50056819)

[5.4-مخطط النشاط"Activity Digram" 24](#_Toc50056818)

[الفصل الخامس تصميم النظام 27](#_Toc50056820)

[1.5-مقدّمة 27](#_Toc50056821)

[2.5-التّصميم الأولي للنظام 27](#_Toc50056822)

[3.5-مكونات النظام 27](#_Toc50056824)

4.5-مخطط الصفوف"Class Diagram"…………………………………………. 28

[5.5- مخطط العلاقات والارتباطات"ERD" 29](#_Toc50056825)

[6.5- واجهات النظام 30](#_Toc50056825)

[الفصل السادس التنفيذ والاختبارات 31](#_Toc50056827)

[1.6- مقدمة 31](#_Toc50056828)

[2.6-أدوات التنفيذ 31](#_Toc50056829)

[3.6-الاختبارات 31](#_Toc50056830)

[4.6-دليل التنصيب 31](#_Toc50056831)

[5.6-دليل المستخدم 31](#_Toc50056832)

[6.6- مشاكل وصعوبات 47](#_Toc50056833)

[الخاتمة والآفاق المستقبلية 49](#_Toc50056834)

[المراجع 50](#_Toc50056835)

# قائمة الأشكال

[**الشكل 1: مقارنة بين بيئات التطوير لأظمة تشغيل الهواتف الذكية 3**](#_Toc493134758)

[**الشكل 2: مقارنة بين انتشار أنظمة التشغيل للهواتف الذكية خلال السنتين الماضيتين [2] 4**](#_Toc493134759)

[**الشكل 3:** واجهة تسجيل الدخول **3**](#_Toc493134758)

[**الشكل 4:** الواجهة الرئيسية للمواقع **3**](#_Toc493134758)

[**الشكل 5:** واجهة مشاهدة الإعلانات للأستاذ **3**](#_Toc493134758)

[**الشكل 6:** واجهة مشاهدة الإعلانات للموظف الإداري **3**](#_Toc493134758)

[**الشكل 7:** واجهة مشاهدة الإعلانات للطلاب **3**](#_Toc493134758)

[**الشكل 8:** واجهة إضافة إعلان **3**](#_Toc493134758)

[**الشكل 9:** واجهة عرض المواد عند الطالب أو الأستاذ **3**](#_Toc493134758)

[**الشكل 10:** واجهة عرض المواد عند الموظف الإداري **3**](#_Toc493134758)

[**الشكل 11:** واجهة عرض العلامات عند الطالب **3**](#_Toc493134758)

[**الشكل 12:** واجهة عرض العلامات عند الموظف الإداري **3**](#_Toc493134758)

[**الشكل 13:** واجهة إضافة العلامات للطلاب **3**](#_Toc493134758)

[**الشكل 14:** عرض جميع الطلاب والموظفين والأساتذة **3**](#_Toc493134758)

[**الشكل 15:** شاشة البدء **3**](#_Toc493134758)

[**الشكل 16:** الشاشة الرئيسية **3**](#_Toc493134758)

[**الشكل 17:** الشاشة الرئيسية للأستاذ **3**](#_Toc493134758)

[**الشكل 18:** شاشة عرض الإعلانات **3**](#_Toc493134758)

[**الشكل 19:** شاشة الفرز عن طريق التاريخ **3**](#_Toc493134758)

[**الشكل 20:** شاشة إضافة الإعلانات **3**](#_Toc493134758)

[**الشكل 21:** شاشة تأكيد عملية الإضافة **3**](#_Toc493134758)

[**الشكل 22:** شاشة عرض المواد **3**](#_Toc493134758)

[**الشكل 23:** شاشة تأكيد عملية المتابعة **3**](#_Toc493134758)

[**الشكل 24:** شاشة علامات الطلاب **3**](#_Toc493134758)

[**الشكل 25:** شاشة بحث عن طالب **3**](#_Toc493134758)

[**الشكل 26:** شاشة إعلانات الطالب **3**](#_Toc493134758)

[**الشكل 27:** شاشة بحث وفرز حسب اسم إعلان **3**](#_Toc493134758)

[**الشكل 28:** شاشة عرض العلامات **3**](#_Toc493134758)

[**الشكل 29:** شاشة البحث والفرز حسب اسم مادة **3**](#_Toc493134758)

[**الشكل 30:** شاشة المواد المتوافرة **3**](#_Toc493134758)

[**الشكل 31:** شاشة إضافة متابعة **3**](#_Toc493134758)

# قائمة الجداول

[**الجدول 1: بعض شركات الهواتف الذكيّة والأتظمة التي تبنّتها 6**](#_Toc493039866)

[**الجدول 2: المخطط الزمني لمراحل العمل 11**](#_Toc493039867)

[**الجدول 3: توصيف حالة تسجيل دخول 16**](#_Toc493039867)

[**الجدول 4: توصيف حالة عرض إعلانات 17**](#_Toc493039867)

[**الجدول 5: توصيف حالة عرض علامات 18**](#_Toc493039867)

[**الجدول 6: توصيف حالة عرض مواد 19**](#_Toc493039867)

[**الجدول 7: توصيف حالةإضافة إعلان 20**](#_Toc493039867)

[**الجدول 8: توصيف حالة إضافة علامة 21**](#_Toc493039867)

[**الجدول 9: توصيف حالة اهتمام بمادة 22**](#_Toc493039867)

[**الجدول 9: توصيف حالة إلغاء اهتمام بمادة 23**](#_Toc493039867)

# الاختصارات

iOS iPhone Operating System

GUI Graphical User Interface

JSON JavaScript Object Notation

MDA Model Driven Architecture

ORM Object Relational Mapping

# المقدّمة

قد دخل عالم الانترنت حياتنا من أوسع أبوابه ليشمل جميع ومختلف أنشطتنا وأعمالنا اليومية، فجعل العالم قرية كونية صغيرة تختصر فيه المسافات والأوقات، وتنتقل فيه البيانات والمعلومات كلمح البصر، فتزايد الإقبال على الانترنت بشكل سريع جداً نظراً لفوائده المتعددة في كثير من المجالات.

كما أنّ الانتشار الهائل للأجهزة المحمولة العاملة بنظام Android في السنين الأخيرة حتّم على جميع شركات العالم إطلاق تطبيقات تعمل على هذا النظام، لتسهيل تقديم خدماتها والتواصل مع عملائها. إضافةً لذلك، نجحت غوغل بجذب آلاف المطوّرين للعمل على إنتاج تطبيقات Android لجميع فئات المستخدمين ترضي مختلف التوجهات.

وتحاول شركات تطوير التطبيقات دمج التقنيّات الحديثة مع الهواتف الذكيّة للتكيّف مع احتياجات المستخدمين والسّماح لهم بالنّفاذ إلى الخدمات والميّزات التي تؤمّنها الهواتف الذكيّة في أيّ زمان ومكان.

رغم كل هذه التقنيات عانت المؤسسات وخصوصاً الأكاديمية منها من مشكلات نشر الإعلانات والتنبيهات يدوياً على لوحة إعلانات، وإضافة إلى ذلك قلة عدد الأشخاص الذين تلفتهم هذه التنبيهات، فاللوحة الجدارية تلزم الشخص بمكان وزمان محددين للاطلاع عليها.

وفي محاولة حل هذه المشكلات، تم تجريب العديد من تقنيات نشر الإعلانات في العالم، واستخدامها في التواصل الجامعي بين الهيئة الجامعية والطلاب، ومن ضمن هذه التجارب أثبتت الوسائل الإلكترونية أنها الأكثر فاعلية، فهي تمثل حلاً فعالاً لمشكلة صعوبة النشر اليدوية، وأيضاً لسهولة الوصول الى الإعلانات في الجامعات، وكان أحد هذه الحلول جعل لوحة إعلانات متاحة على الانترنت.

وفي مشروعنا هذا سنقوم بالاستفادة من هذه الخدمات لبناء تطبيق يساعد على توفير أهمّ احتياجات الطلاب الجامعيين في حياتهم اليوميّة.

# الفصل الأوّل متطلَّبات المشروع

نعرض في هذا الفصل المتطلبات الوظيفية للمشروع أي الوظائف الأساسية التي يجب أن يحققها النظام كما نعرض المتطلبات غير الوظيفية من متطلبات الأداء والواجهات وغيرها.

**متطلّبات الأداء**

* الوثوقية: reliability يجب أن ينجز التطبيق الوظائف المطلوبة منه بدقة عالية. ويُفضّل ضمان عدم التعطّل خلال وقت تنفيذ عملية كاملة (بِشرط توافر الاتصال بالأنترنيت أثناء القيام ببعض العمليات).
* زمن الاستجابة Response time: يجب أن تكون سرعة الأداء عالية وزمن الاستجابة قليل.
* سهولة الاستخدام usability: يجب أن يوفّر التطبيق واجهات واضحة بالنسبة للمستخدم.
* ظروف الاتصال: قابليّة العمل ضمن ظروف اتصال ضعيف بالأنترنيت (2G).

## 1.1-الهدف من المشروع

يهدف المشروع الى بناء وتصميم موقع وتطبيق اندرويد يقوم بعمل منظومة تواصل جامعية تسمح للطالب بالاطلاع على الإعلانات حسب حاجته والعلامات حسب خصوصيته بدون الذهاب إلى الحرم الجامعي وبدون الحاجة إلى اللوحات الجدارية، وبالطبع هذا يختصر الكلفة الورقية مع الزمن والجهد، من حيث الذهاب إلى الجامعة من جهة، والبحث عن الإعلانات من جهة أخرى، وأيضاً موثوقيتها العالية، بحيث توسع نطاق الأشخاص المتوقع وصول الإعلانات إليهم.

كما تسمح المنظومة بالتواصل بين الهيئة الجامعية والطلاب بحيث يستطيع كل من الأستاذ والموظف الإداري بإضافة الإعلانات بشكل أسرع وأسهل.

## 2.1-المتطلبات الوظيفية

يجب أن يحقق النظام الوظائف التالية:

1. **تسجيل الدخول**

يجب أن يسمح النظام للمستخدم بإضافة اسم المستخدم وكلمة المرور (تم انشائهم مسبقا).

1. **إضافة إعلان**

إضافة إعلان للمواد أو إعلان للكلية أو إعلان للجامعة.

1. **إضافة علامة**

إضافة علامات للطلاب سواء كانت علامة مذاكرة أو درجات أعمال أو محصلة نهائية.

1. **عرض الإعلانات وإمكانية البحث ضمنها**

عرض إعلانات المواد التي قام الطالب او الأستاذ بمتابعتها بالإضافة إلى إعلانات الكلية والجامعة، ويوفر النظام إمكانية البحث عن إعلان عن طريق إدخال اسم الإعلان أو محتواه في مربع البحث، وإمكانية فرز الإعلانات حسب التاريخ.

1. **عرض العلامات وإمكانية البحث ضمنها**

عرض علامات مواد الطالب فقط الذي قام في تسجيل الدخول الى النظام وإمكانية البحث عن علامة مادة معينة عن طريق إدخال اسم المادة في مريع البحث، أما إذا كان أستاذ سيعرض النظام علامات الطلاب كلها وعندها سيكون البحث عن اسم المادة او الطالب.

1. **عرض المواد ومتابعتها**

عرض المواد المتوفرة في الفصل الدراسي، وإمكانية إضافة المتابعة من قبل الطلاب والأساتذة.

1. **عرض الطلاب**

يقوم النظام بعرض معلومات الطلاب ويوفر إمكانية البحث عن طالب معين.

1. **استيراد علامات الطلاب ومعلوماتهم الشخصية**

يقوم النظام بعملية استيراد علامات الطلاب ومعلوماتهم الشخصية من ملف Excel.

## 3.1-المتطلبات غير الوظيفية

### 1.3.1-متطلبات **البيئة التقنية**

يعمل على هاتف محمول ذكي ضمن بيئة Android، يجب توافر الاتصال بشبكة الانترنت عند أول استخدام للتطبيق.

### 2.3.1-المتطلبات الأمنية والأداء

* آمن: ستكون جميع كلمات السر مشفرة.
* مرن وسهل الاستخدام
* سرعة الأداء.
* سيكون الموقع والتطبيق متاحاً 24 ساعة.

### 3.3.1-متطلبات التخزين وإدارة الذاكرة

تطبيق يقوم بتخزين المعلومات على ذاكرة الهاتف المحمول وبالتالي يعرض المعلومات المخزنة في حالة عدم الاتصال بالإنترنت.

# الفصل الثاني الدراسة المرجعية

## **1-مقدّمة**

في هذا الفصل سنقوم بتعريف بيئة العمل Elixir، وسنعرض لمحة عن API وعن بنية المعطيات JSON. وبما أنَّ المشروع يتضمن اختبار أداء النظم المؤسساتية، سنقوم باستعراض بعض أهم مقاييس الأداء.

**2.2- لمحة عن تطبيقات الويب**

تخدم تطبيقات الويب عددًا كبيرًا من المستخدمين مع قدرة الأجهزة على خادم واحد مما يزيد من المشكلات التي يواجهها مطورو الويب كل يوم مثل إدارة الذاكرة، واستخدام قواعد بيانات مختلفة، وإدارة الوصول المتزامن، وأمن البيانات والأعمال، وقابلية الصيانة، وامتلاك شبكة ويب واجهة المستخدم التي تعمل عبر متصفحات الويب المتعددة (على سبيل المثال لا الحصر)، لهذا السبب يحتاج المطورون إلى بيئات عمل لتغطية كل هذا التعقيد بالنسبة لهم.

**2.2.1- بيئة العمل Elixir**

تقوم بيئات عمل مثل Spring و Laravel بعمل جيد في تغطية بعض هذه المشكلات ولكنك ستقاتل دائمًا على ثلاث جبهات (العرض والتحكم والتخزين)، بينما يستخدم Elixir من ناحية أخرى نموذج الهندسة المعمارية (MDA) مما يعني أنك تحتاج فقط إلى التركيز وفقًا لمنطق ونموذج سير العمل التجاري وسيهتم Elixir بالباقي، فأنت لست بحاجة إلى تعلم HTML أو كتابة أي بناء جملة لقاعدة البيانات أو حتى القلق بشأن معاملات قاعدة البيانات لأن Elixir يتيح لك التحكم في جميع المستويات الثلاثة من مكان واحد، ويولد المستخدم واجهة لك ويشكل استفساراتك.

الميزات الرئيسية للإكسير:

1. سهولة في البناء والتطوير من خلال مبدأ run then specify.

.2 إنشاء تلقائي لواجهة المستخدم وقاعدة البيانات وإجراءات العمل.

3. إدارة آلية للموارد من ذاكرة وCBU.

4. نموذج أمان متين وسهل للمستخدم.

## 4.2**-خدمات الويب Web Services**

خدمة الويب بشكل أساسي هي برامج مكتوبة بلغات مثل java، asp، php تطلُب برامج تعمل على مخدّم آخر (خدمة الويب) واستخدام استجابة ذلك البرنامج في تطبيق وِب أو أي تطبيق آخر.

تتميز خدمة الويب عن تطبيق الويب بشكل أساسي بطريقة طلب الخدمة، حيث يتم طلب الخدمة عبر واجهتها البرمجية   
API وليس الرسومية GUI (Graphical User Interface)، ويتم إعادة نتيجة الطلب على هيئة XML أو JSON.

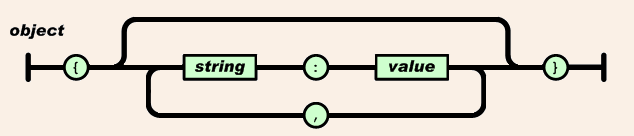
### 1.4.2**-Application Programming Interface) API)**

API هي اختصار ل Application Programming Interface، وهي مجموعة **صفوف** **وتوابع** مكتوبة بلغة برمجة معينة، توفر ال API مجموعة من **الواجهات** المجرّدة التي تخفي خلفها تعقيدات النص البرمجي بحيث يمكن للبرامج الأخرى (التي من الممكن أن تكون مكتوبة بلغات برمجة مختلفة) أن تتفاعل مع هذه الواجهات بكل سهولة، يتم تنظيم هذه الصّفوف والتوابع في مكتبة برمجيّات أو خدمة وِب، تصاحبها شروحات تمكن المبرمج من استيعاب طريقة استعمال تِلك الخدمات أو محتويات المكتبة.

### 2.4.2- بنية المعطيات JSON

(JSON) Java Script Object Notation هي بنية معطيات سلسة لنقل البيانات، سهلة القراءة والاستيعاب من قبل المبرمجين بشكل عام، وسهلة التوليد والتحليل للأجهزة والحواسيب، وهي ظهرت بدايةً للاستخدام مع لغة جافا سكريبت (Java Script)، ولكنها مستقلة تماماً عن اللغة، حيث يُمكن استخدامها من قِبل عائلة لغات ال C والتي تشمل ++C، C# ، Java، والعديد من اللغات الأُخرى لاحتواء اللغات آنفة الذكر على مكتبات خاصّة بهذه البنية، تلك المميّزات تجعل من JSON الأفضل في نقل البيانات data)) من لغة لأُخرى.

تتكوّن بنية JSON من مجموعة من الأزواج: الأسماء ((Names والقيم Values)) حيث يمكن للقيمة أن تكون غرض (Object) من النمط JSON أيضاً، ويمكنها أن تكون مصفوفة من أغراض JSON.



الشكل 5: بنية JSON

مثال على بنية JSON:

{"**menu**": {

"**header**": "SVG Viewer",

"**items**": [

{"**id":** "Open"},

{"**id**": "Open New", "label": "Open New"},

null,

{"**id**": "Zooming", "label": "Zoom In"},

{"**id**": "Zoom Out", "label": "Zoom Out"},

{"**id**": "Original View", "label": "Original View"},

null,

{"**id**": "Quality"},

{"**id**": "Pause"}

]

}}

أنواع اختبار البرمجيات:

1. Unit Testing
2. Integration Testing
3. Regression Testing
4. Smoke Testing
5. Alpha Testing
6. Beta Testing
7. System Testing
8. Stress Testing
9. Performance Testing  
   اختبار الأداء هو عملية اختبار برمجية تستخدم لاختبار السرعة ووقت الاستجابة والاستقرار و الموثوقية وقابلية التوسع واستخدام الموارد لتطبيق برمجي تحت عبء عمل معين. الغرض الرئيسي من اختبار الأداء هو تحديد وإزالة الاختناقات في الأداء في تطبيق البرنامج. يمكن استخدام اختبار الأداء كشكل من أشكال اختبار البرامج للمساعدة في تحديد طبيعة أو موقع مشكلة الأداء المتعلقة بالبرمجيات من خلال تمييز المكان الذي قد يفشل فيه التطبيق أو يتأخر فيه.
10. Object-Oriented Testing

أنواع اختبار الأداء:

1. Load Testing: عبء متوقع (لتحديد الاختناقات)
2. Stress Testing: عبء عمل شاق (لتحديد نقطة انهيار البرنامج)
3. Endurance Testing: التعامل مع الحمل على فترة طويلة (التحمل)
4. Spike Testing: يختبر رد فعل البرنامج على الارتفاعات الكبيرة في الحمل
5. Volume Testing: التحقق من الأداء في احجام قاعدة بيانات مختلفة
6. Scalability Testing: تحديد قابلية التوسع لدعم زيادة حمل المستخدم

# الفصل الثالث إدارة المشروع

*في هذا الفصل يتم تقديم خطة عمل المشروع، وتحديد المنهجية المتبعة لتطوير النظام.*

## 1.3-مقدّمة

من الأمور المهمة في أي مشروع وجود خطة، تسهل العمل لبناء هذا المشروع، وتحقق ضمان تسليم النظام المطلوب في الوقت المناسب من خلال جدولة العمل، وتضمن سير العمل بما يتوافق مع متطلبات النظام.

## 2.3-خطة إدارة المشروع

في الواقع من الصعب اختيار منهجية مناسبة للتطوير، خصوصاً في حالتنا فنحن لا نعلم العقبات التي يمكن أن تعترضنا، لكن يمكن التنبؤ مسبقاً بمدى صعوبة مرحلة معينة، وتحديد وقت إضافي احتياطاً، وهذا ما تمّ العمل عليه، إذ قسمنا النظام المطلوب إلى عدة أجزاء حسب الوظائف التي يتعين عليه القيام بها، وقدرنا مدى صعوبة كل مرحلة، وتم تحديد الفترة الزمنية المناسبة وفقاً لهذه التقديرات، امتدّ العمل نحو 32 أسبوع بمعدل 6 ساعات يومياً، على عدة مراحل كما يوضح الجدول التالي:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| رقم المرحلة | اسم المرحلة | المدة (يوم) | تاريخ البدء | تاريخ الانتهاء |
| 1 | دراسة نظرية | 56 | 1/11/2020 | 1/1/2021 |
| 2 | تحديد متطلبات المشروع | 10 | 25/1/2021 | 5/2/2021 |
| 3 | تحليل النظام | 15 | 10/2/2021 | 25/2/2021 |
| 4 | التصميم | 20 | 1/3/2021 | 20/3/2021 |
| 5 | التنجيز العملي | 40 | 1/4/2021 | 10/5/2021 |
| 6 | الاختبارات | 20 | 15/5/2021 | 5/6/2021 |
| 7 | التوثيق النهائي | 14 | 1/7/2021 | 14/7/2021 |

الجدول 2: المخطط الزمني لمراحل العمل

استمرّت كتابة أجزاء التقرير طيلة فترة المشروع وتمّ تجميع الأجزاء والتوثيق النهائي في نهاية هذه الفترة.

# الفصل الرابع تحليل المتطلبات

*يعرض هذا الفصل ما يجب أن يقوم به النظام من خلال تحليل للمتطلَّبات المذكورة في الفصل الأول.*

## 1.4-مقدّمة

عرضنا في الفصل الأول متطلَّبات النظام، حيث تعرِّف هذهِ المتطلبات ما يجب على النظام فعله. سنقدم في هذا الفصل عرضاً تحليلياً للنظام.

## 2.4-التطبيقات المشابهة

**JMeter**

تطبيق **Apache JMeter™** هو برنامج مفتوح المصدر، وهو تطبيق Java خالص 100٪ مصمم لتحميل اختبار السلوك الوظيفي وقياس الأداء. تم تصميمه في الأصل لاختبار تطبيقات الويب ولكنه امتد منذ ذلك الحين إلى وظائف الاختبار الأخرى.

**من السلبيات التي نجدها في الموقع:**

- استهلاك الذاكرة: يمكن لـ JMeter محاكاة الحمل الثقيل وتصور تقرير الاختبار. قد يستهلك هذا الكثير من الذاكرة ويمكن أن يؤدي إلى نفاد الذاكرة تحت الحمل الثقيل.

- تطبيق الويب فقط: يعد JMeter أداة جيدة لاختبار تطبيق الويب ولكنه ليس أداة مناسبة لاختبار تطبيقات سطح المكتب.

- عدم وجود دعم لـ JavaScript: JMeter ليس مستعرضًا، لذلك لا يمكنه تشغيل JavaScript في تطبيق الويب. لديها دعم محدود لمعالجة JavaScript أو Ajax، وهذا قد يؤثر على دقة المحاكاة.

**من أهم ميزات الموقع:**

- مجاني ومفتوح المصدر.

- يستخدم على أنظمة تشغيل مختلفة **(Linux، Windows، Mac).**

**Web Runner**

LoadRunner هي أداة اختبار برمجية من Micro Focus. يتم استخدامه لاختبار التطبيقات وقياس سلوك النظام وأدائه تحت الحمل. يمكن لـ LoadRunnerمحاكاة آلاف المستخدمين في نفس الوقت باستخدام برنامج تطبيقي، وتسجيل أداء المكونات الرئيسية للتطبيق وتحليلها لاحقًا.

**من سلبيات الموقع:**

* يحدد السبب الجذري لمشاكل أداء التطبيق.
* يقلل تكاليف البرامج والأجهزة من خلال التنبؤ بسعة البرامج وقابليتها للتوسع.

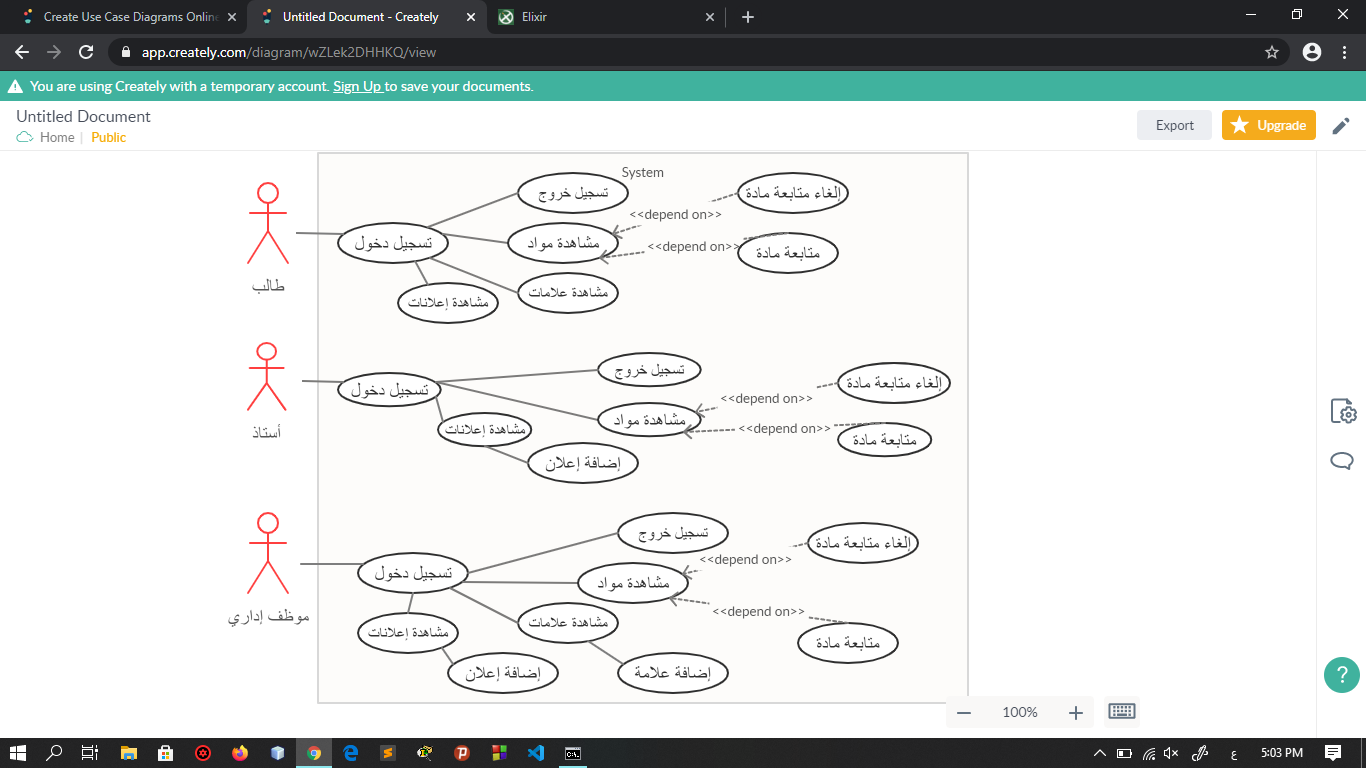
**من أهم ميزات الموقع:**

* يستخدم قدرًا كبيرًا من الذاكرة ويؤدي إلى حدوث أعطال إذا لم يتمكن النظام من تلبية متطلبات الحوسبة الخاصة به.
* القدرة على التصحيح يحتاج إلى تحسين غالي جدا.

## 3.4-مخطط حالات الاستخدام

نعرُض هنا مخطّط حالات الاستخدام الذي يوضح كيفية تفاعل النظام مع البيئة التي يعمل ضمنها، وكيفية تفاعل مستخدمي النظام معه.

يوضح الشكل 10 مخطط حالات الاستخدام، حيث تمّ استخدام الأسهم الموجهة بين الفاعل الأساسي وحالة الاستخدام. ونلاحظ في الشّكل الوظائف الأساسية التي يجب أن يؤمّنها النظام.



الشكل 10: مُخطّط حالات الاستخدام

## 4.4-توصيف حالات الاستخدام"Narrative Use Case"

نكتب لكل حالة استخدام توصيف يشرح تدفق الأحداث ومختلف السيناريوهات الممكنة.

|  |  |
| --- | --- |
| حالة الاستخدام | **تسجيل الدخول** |
| رقم الحالة | 1 |
| شرح مختصر | يقوم التطبيق بعرض واجهة الطالب أو الأستاذ حسب اسم المستخدم |
| شريحة المستخدمين | طالب أو أستاذ أو عميد كلية. |
| الشروط السابقة | توافر اتصال بالأنترنيت. |
| خطوات حالة الاستخدام | 1. تبدأ حالة الاستخدام يقوم المستخدم بفتح التطبيق. 2. يتحقّق النظام من توافر الاتصال بالأنترنيت 3. يتصل النظام ب Elixir API ويقوم بالتحقق من اسم المستخدم وكلمة السر   4. يقوم النظام بِعرض الواجهات مباشرةً. |
| الحالة اللاحقة | تمّ تسجيل الدخول. |
| التدفق البديل | في حال عدم توفّر اتّصال بالأنترنيت يقوم التطبيق بِعرض رسالة تُعلم المستخدم بالحاجة لتوافر الاتصال بالأنترنيت ويبقى التطبيق مفتوحاً في انتظار توفّر الاتّصال بالأنترنيت وفي حالة إدخال اسم مستخدم او كلمة مرور خاطئة يقوم التطبيق بِعرض رسالة تُعلم المستخدم بالحاجة إلى إعادة التأكد من اسم المستخدم وكلمة المرور. |

الجدول 3: توصيف حالة تسجيل دخول

|  |  |
| --- | --- |
| حالة الاستخدام | **عرض الإعلانات** |
| رقم الحالة | 2 |
| شرح مختصر | قوم التطبيق بِعرض الإعلانات |
| شريحة المستخدمين | الطالب، الأستاذ وعميد الكلية. |
| الشروط السابقة | تسجيل الدخول. |
| خطوات حالة الاستخدام | 1. تبدأ حالة الاستخدام عندما يقوم المستخدم بتسجيل الدخول. 2. يُدخل المستخدم اسم الإعلان او محتواه للبحث عنه. 3. يتحقّق النظام من توافر الاتصال بالأنترنيت. 4. يقوم النظام بإرسال طلب عرض الإعلانات إلى Elixir API. 5. يرسل المخدّم المعلومات المطلوبة إلى النظام. 6. يقوم النظام بعرض إعلانات المواد التي قام بمتابعتها الطالب أو الأستاذ وإعلانات الكلية والجامعة القادمة من المخدم. 7. يقوم النظام بعرض نتائج عملية البحث. |
| الحالة اللاحقة | تمّ عرض الإعلانات. |
| التدفق البديل | في حال عدم توافر الاتصال بالأنترنيت يظهر آخر 15 إعلان تم عرضهم في السابق. |

الجدول 4: توصيف حالة عرض الإعلانات

|  |  |
| --- | --- |
| حالة الاستخدام | **عرض العلامات** |
| رقم الحالة | 3 |
| شرح مختصر | يقوم التطبيق بعرض العلامات. |
| شريحة المستخدمين | الطالب، الأستاذ. |
| الشروط السابقة | توافر الاتصال بالإنترنيت وتسجيل الدخول. |
| خطوات حالة الاستخدام | 1. تبدأ حالة الاستخدام عندما يقوم المستخدم بتسجيل الدخول. 2. يُدخل المستخدم اسم المادة للبحث عن علامتها أذا كان المستخدم أستاذ يمكنه بالإضافة البحث عن طريق اسم الطالب. 3. يقوم النظام بالتحقّق من الاتصال بشبكة الانترنت. 4. يقوم النظام بإرسال طلب عرض المواد إلى Elixir API. 5. يرسل المخدّم المعلومات المطلوبة إلى النظام. 6. يقوم النظام بعرض علامات المواد التي قام الطالب بمتابعتها وبالنسبة إلى الأستاذ يعرض النظام له علامات كل الطلاب. 7. يقوم النظام بعرض نتائج عملية البحث. |
| الحالة اللاحقة | تمّ عرض العلامات. |
| التدفق البديل | في حال عدم توافر اتصال بالإنترنيت تظهر رسالة للمستخدم تطلب منه توفير الاتصال بالإنترنيت. |

الجدول 5: توصيف حالة عرض العلامات

|  |  |
| --- | --- |
| حالة الاستخدام | **عرض المواد** |
| رقم الحالة | 4 |
| شرح مختصر | يقوم النظام بعرض المواد المتوفرة في الفصل الحالي. |
| شريحة المستخدمين | الطالب، الأستاذ. |
| الشروط السابقة | توافر الاتصال بالإنترنيت وتسجيل الدخول. |
| خطوات حالة الاستخدام | 1. يقوم المستخدم بالضّغط على الزّر الخاصّ لعرض المواد. 2. يقوم المستخدم بإضافة متابعة للمواد المتوفرة. 3. يقوم النظام بإرسال طلب المتابعة إلى Elixir API. 4. يرسل المخدّم المعلومات المطلوبة إلى النظام. 5. يقوم النظام بعرض المواد المتوفرة في الفصل الحالي. |
| الحالة اللاحقة | تمّ عرض المواد. |
| التدفق البديل | في حال عدم توافر اتصال بالإنترنيت تظهر رسالة للمستخدم تطلب منه توفير الاتصال بالإنترنيت. |

الجدول 6: توصيف حالة عرض المواد

|  |  |
| --- | --- |
| حالة الاستخدام | **إضافة إعلان** |
| رقم الحالة | 5 |
| شرح مختصر | سيقوم المستخدم بإنشاء إعلان عن طريق تعبئة الحقول المطلوبة. |
| شريحة المستخدمين | الأستاذ والموظف الإداري وعميد الكلبة. |
| الشروط السابقة | توافر الاتصال بالإنترنيت وتسجيل الدخول. |
| خطوات حالة الاستخدام | 1. تبدأ حالة الاستخدام عندما يقوم المستخدم باختيار إعلان جديد. 2. سيظهر النظام للمستخدم عدة حقول من عنوان الإعلان وتاريخه و... 3. سيقوم المستخدم بملء الحقول 4. سيقوم المستخدم بالضغط على زر تنفيذ أو نشر. 5. ستظهر نافذة بإتمام عملية النشر. |
| الحالة اللاحقة | تمّ رفع المنشور على النظام. |
| التدفق البديل | إذا لم تنجح العملية، سيعود إلى الخطوة 1. |

الجدول 7: توصيف حالة إضافة إعلان

|  |  |
| --- | --- |
| حالة الاستخدام | **إضافة علامة** |
| رقم الحالة | 6 |
| شرح مختصر | سيقوم المستخدم بإضافة علامة للطالب عن طريق تعبئة الحقول المطلوبة. |
| شريحة المستخدمين | الموظف الإداري |
| الشروط السابقة | توافر الاتصال بالإنترنيت وتسجيل الدخول. |
| خطوات حالة الاستخدام | 1. تبدأ حالة الاستخدام عندما يقوم المستخدم باختيار إضافة علامة. 2. سيظهر النظام عدة حقول من اسم طالب واسم المادة. 3. سيقوم المستخدم بملء الحقول. 4. سيقوم المستخدم بالضغط على زر تنفيذ. 5. ستظهر نافذة بإتمام العملية. |
| الحالة اللاحقة | تم إضافة علامة للطالب على النظام |
| التدفق البديل | إذا لم تنجح العملية، سيعود إلى الخطوة 1. |

الجدول 8: توصيف حالة إضافة علامة

|  |  |
| --- | --- |
| حالة الاستخدام | **اهتمام بمادة** |
| رقم الحالة | 7 |
| شرح مختصر | سيقوم المستخدم باختيار المواد المهتم بها ثم تفعيل زر مهتم. |
| شريحة المستخدمين | الطالب والعميد والأستاذ. |
| الشروط السابقة | توافر الاتصال بالإنترنيت وتسجيل الدخول.  غير مهتم فيها سابقاً. |
| خطوات حالة الاستخدام | 1. تبدأ حالة الاستخدام عندما يقوم المستخدم بالضغط على زر متهتم للمادة التي يريدها. 2. ستظهر نافذة بإتمام العملية بجاح. |
| الحالة اللاحقة | تم الاهتمام بالمادة وسيتم عرض إعلاناتها في صفحة الإعلانات. |
| التدفق البديل | إذا لم تنجح العملية، سيعود إلى الخطوة 1. |

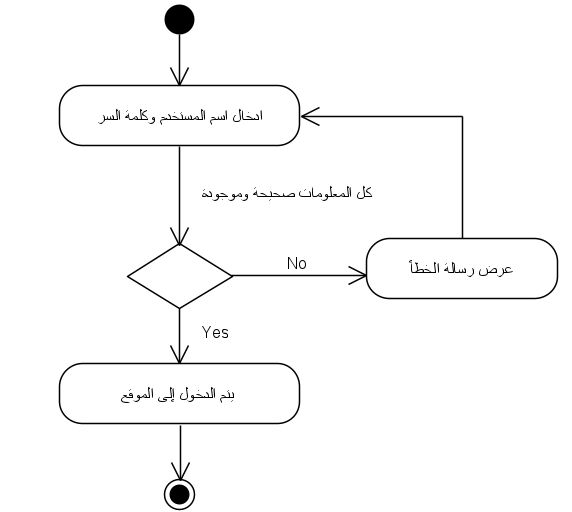
الجدول 9: توصيف حالة اهتمام بمادة

|  |  |
| --- | --- |
| حالة الاستخدام | **إلغاء اهتمام بالمادة** |
| رقم الحالة | 8 |
| شرح مختصر | سيقوم النظام بإلغاء اهتمامه بالمادة التي يريدها |
| شريحة المستخدمين | الطالب والعميد والأستاذ. |
| الشروط السابقة | توافر الاتصال بالإنترنيت وتسجيل الدخول.  مهتم فيها سابقاً. |
| خطوات حالة الاستخدام | 1. تبدأ حالة الاستخدام عندما يقوم المستخدم بالضغط على زر غير متهتم للمادة التي يريدها. 2. ستظهر نافذة بإتمام العملية بجاح. |
| الحالة اللاحقة | سيتم إلغاء الاهتمام بالمادة وسيتم ازاله إعلانات هذه المادة من صفحة الإعلانات ولن يتم إضافة إعلانات جديد لهذه المادة. |
| التدفق البديل | إذا لم تنجح العملية، سيعود إلى الخطوة 1. |

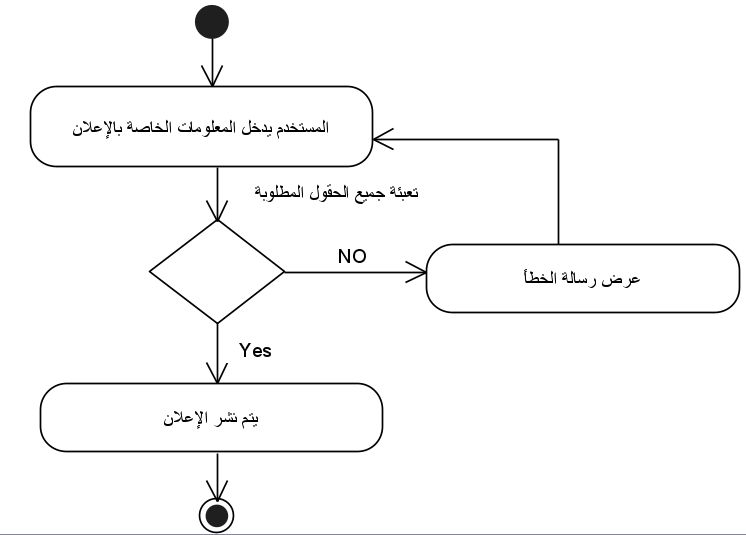
الجدول 10: توصيف حالة إلغاء اهتمام بمادة

## 4.5-مخطط النشاطات"Activity Diagram"

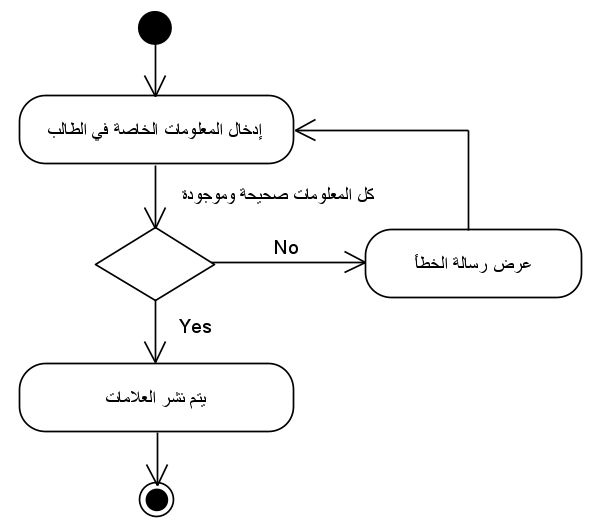
حالة تسجيل دخول:



حالة إضافة إعلان:



حالة إضافة علامة:



# الفصل الخامس تصميم النظام

*في هذا الفصل نعرض توصيفاً للنظام وكيفية عمله، حيث نعرض شرح مفصل للتصميم المقترح*

## 1.5-مقدّمة

تعرض مرحلة التصميم البنى البرمجية التي سيتم استخدامها، نقدم فيها أيضاً واجهات التخاطب مع المستخدم، بالإضافة الى المكاتب التي يستخدمها النظام لإنجاز مهامه، نتكلم أيضاً عن دخل النظام وخرجه.

## 2.5-التّصميم **الأولي للنظام**

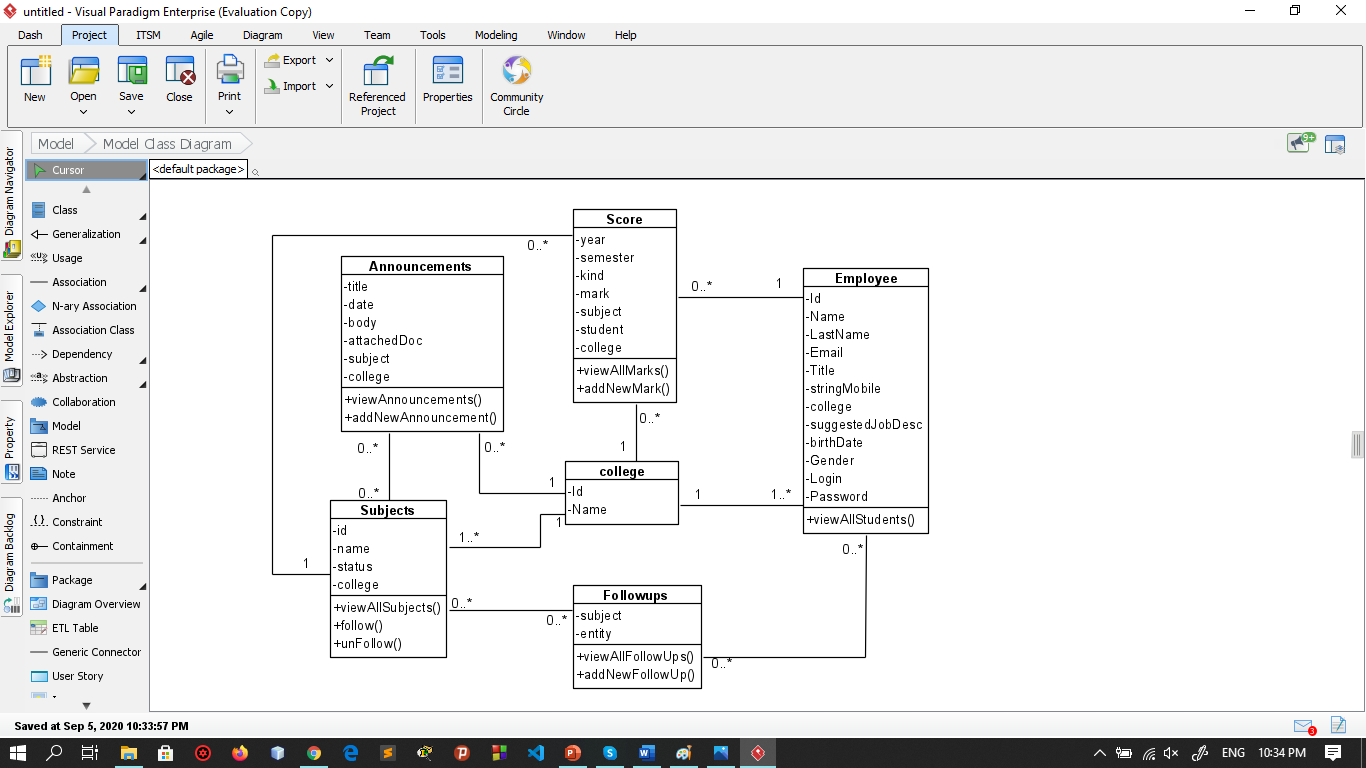
## 

## 3.5-مكونات النظام

هي عبارة عن الصّفوف المستخدمة والمكتبات الخاصّة ببناء هذه الصّفوف، بالإضافة إلى واجهات البرنامج، لتشكّل جميعها بُنية التطبيق، الذي تؤمّن المهام المطلوبة منَ التطبيق التي سُبق وأن ذكرت في الفصل الأول ضمن المتطلبات الوظيفية، سنتكلّم عن هذه المكونات لاحقاً بالتفصيل.

**4.5-مخطط الصفوف"Class Diagram"**

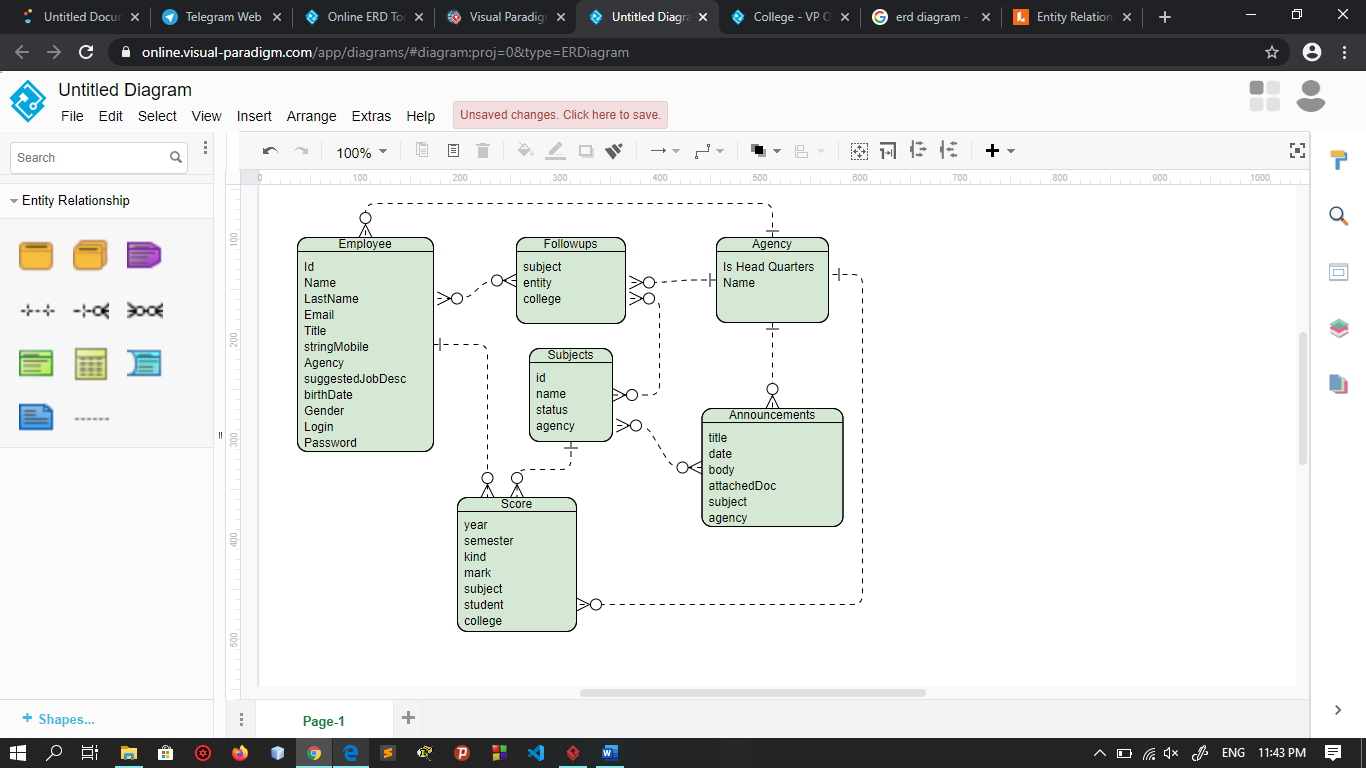
هو نوع من الرسم التخطيطي للهيكل الثابت الذي يصف بينة النظام، من خلال إظهار فئات النظام وخصائصه، وعملياته، والعلاقات بين الكائنات، وقد تم استخراج الصفوف من الهيكلية العامة للنظام والمتطلبات المعطاة.



## 

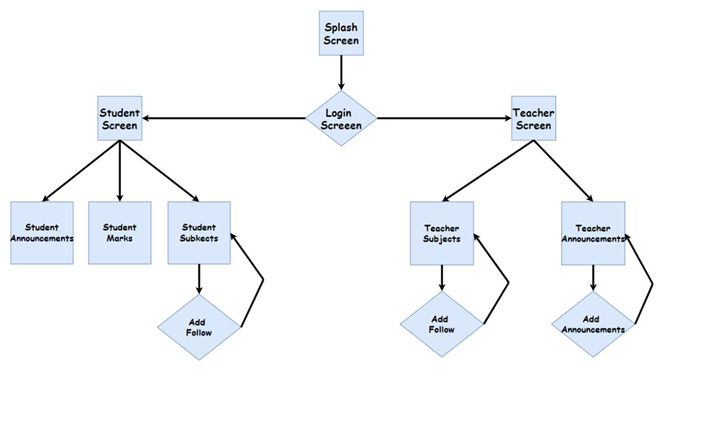
## 5.5-مخطط العلاقات والارتباطات"ERD"

يظهر المخطط قاعدة بيانات الموقع وارتباط جداولها معاً وما يحتويه كل جدول من كائنات وأنواعها.



### 6.5-واجهات النظام

تتميز واجهات النظام بالبساطة والوضوح وهي تتبع المخطّط التالي:



# الفصل السادس التنفيذ والاختبارات

*يعرض هذا الفصل بناء النظام واختباره وتدريب المستخدمين على النظام.*

## 1.6- مقدمة

نتحدث في هذا الفصل عن الأدوات المستخدمة في تنفيذ المشروع، نتطرّق أيضاً للحديث عن المشاكل البرمجية التي واجهتنا أثناء التنفيذ العملي، نعرض بعض الاختبارات للتطبيق، كما نزوّد المستخدم بدليل مساعد على استخدام التطبيق، ونختم هذا الفصل بالدروس المستفادة والخلاصة والمقترحات.

في هذه المرحلة يجري بناء النظام واختباره للتأكد من أدائه للعمل كما جرى تصميمه. نتحدث في هذا الفصل عن الأدوات المستخدمة في تنفيذ المشروع، ثم الحديث عن العوائق البرمجية التي واجهتنا ثم ننتقل إلى خطة الاختبارات المطبقة بعدها نتطرّق للحديث دليل استخدام التطبيق.

## 2.6-أدوات التنفيذ

* بيئة العمل Elixir
* Visual Studio code لتصميم وتنفيذ النظام.

## 3.6-الاختبارات

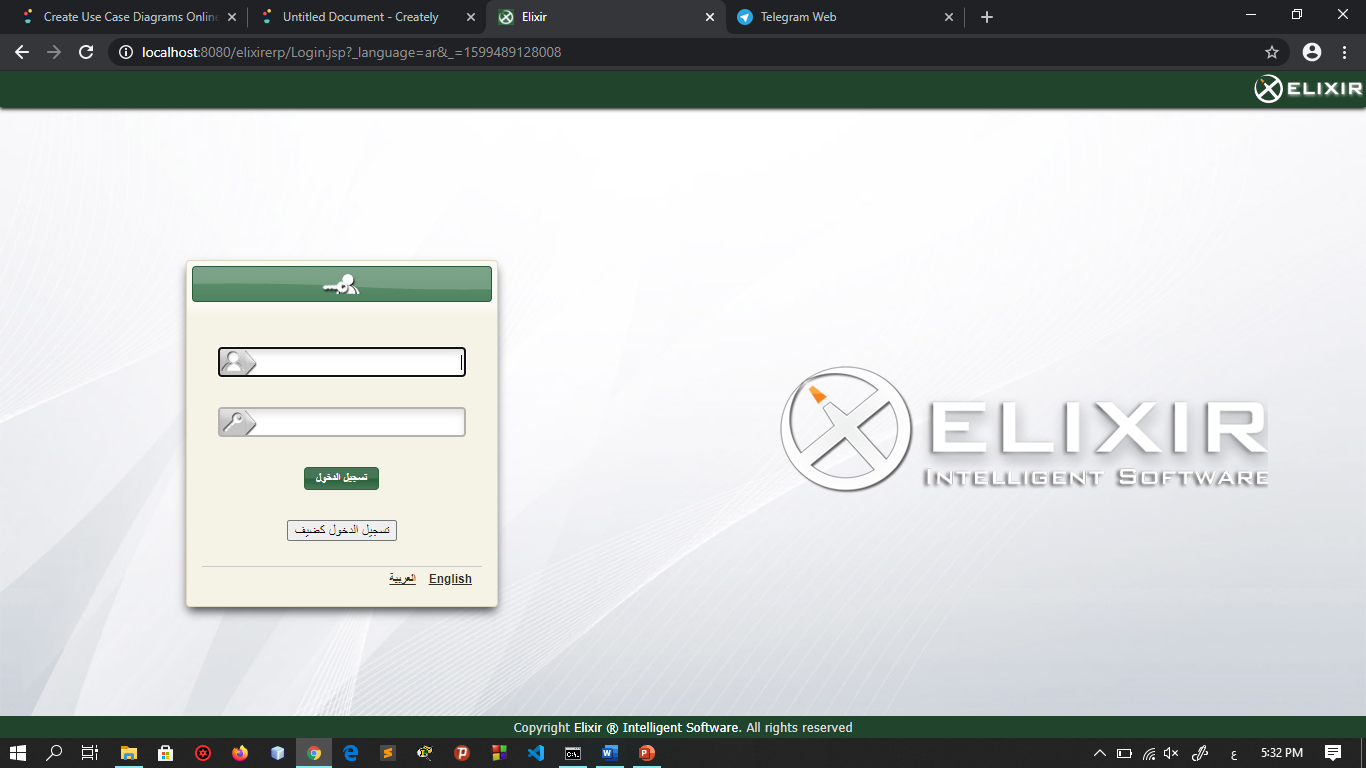
* تمّ اختبار التطبيق في جميع المهام التي يقوم بها كما وثّقنا هذه الاختبارات بفيديو يوضِّح ذلك، سنعرضهُ أثناء مقابلة الدفاع.

## 4.6-دليل التنصيب

## 5.6-دليل المستخدم

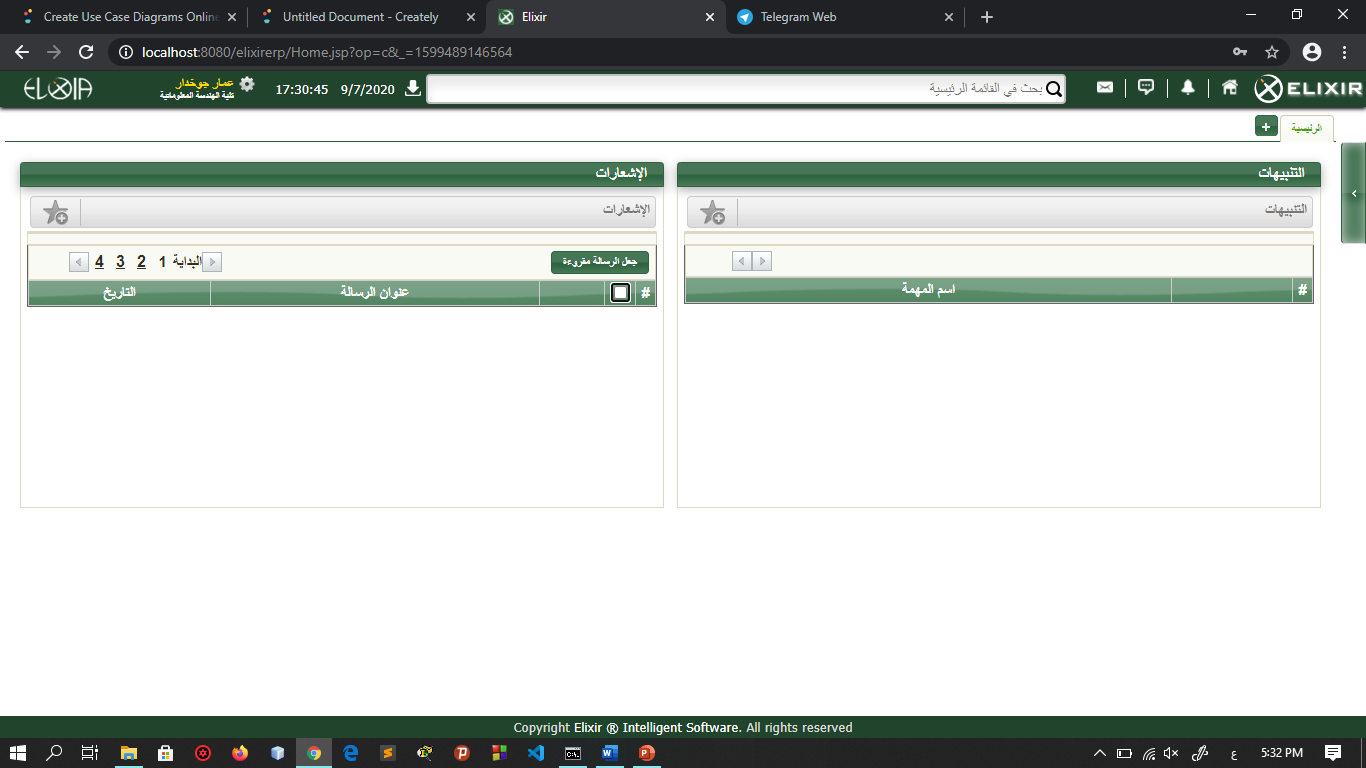
**واجهات التفاعل مع النظام في الموقع:**

واجهة تسجيل الدخول



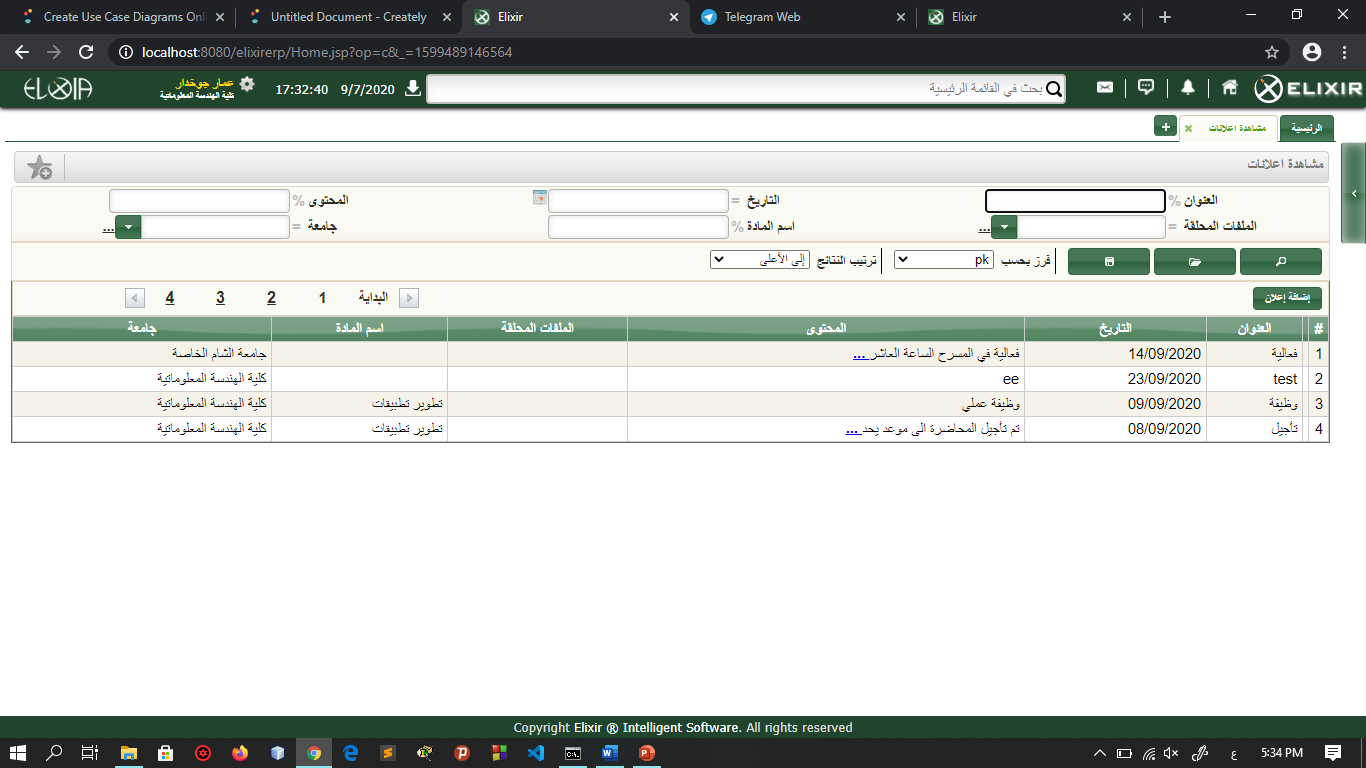
الشكل3: واجهة تسجيل الدخول

الواجهة الرئيسية للموقع

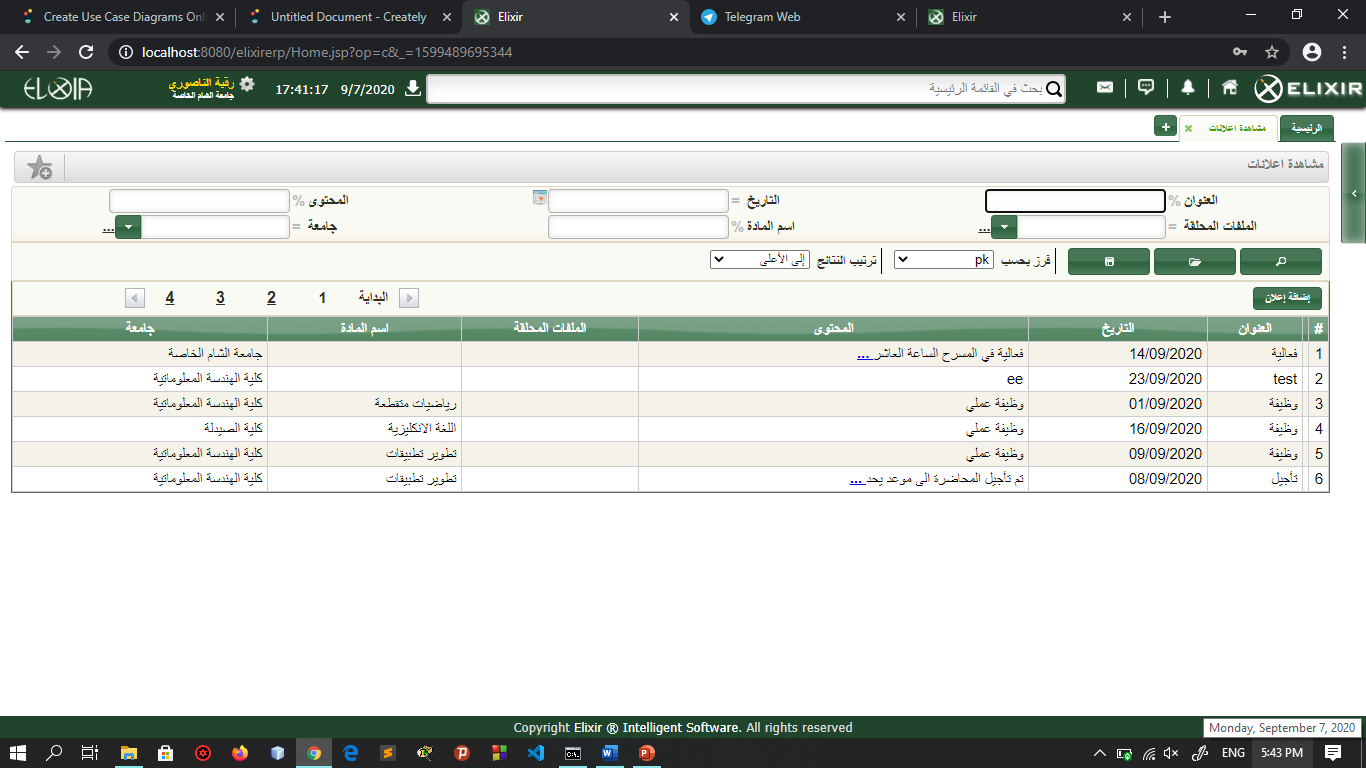


الشكل 4: الواجهة الرئيسية للمواقع

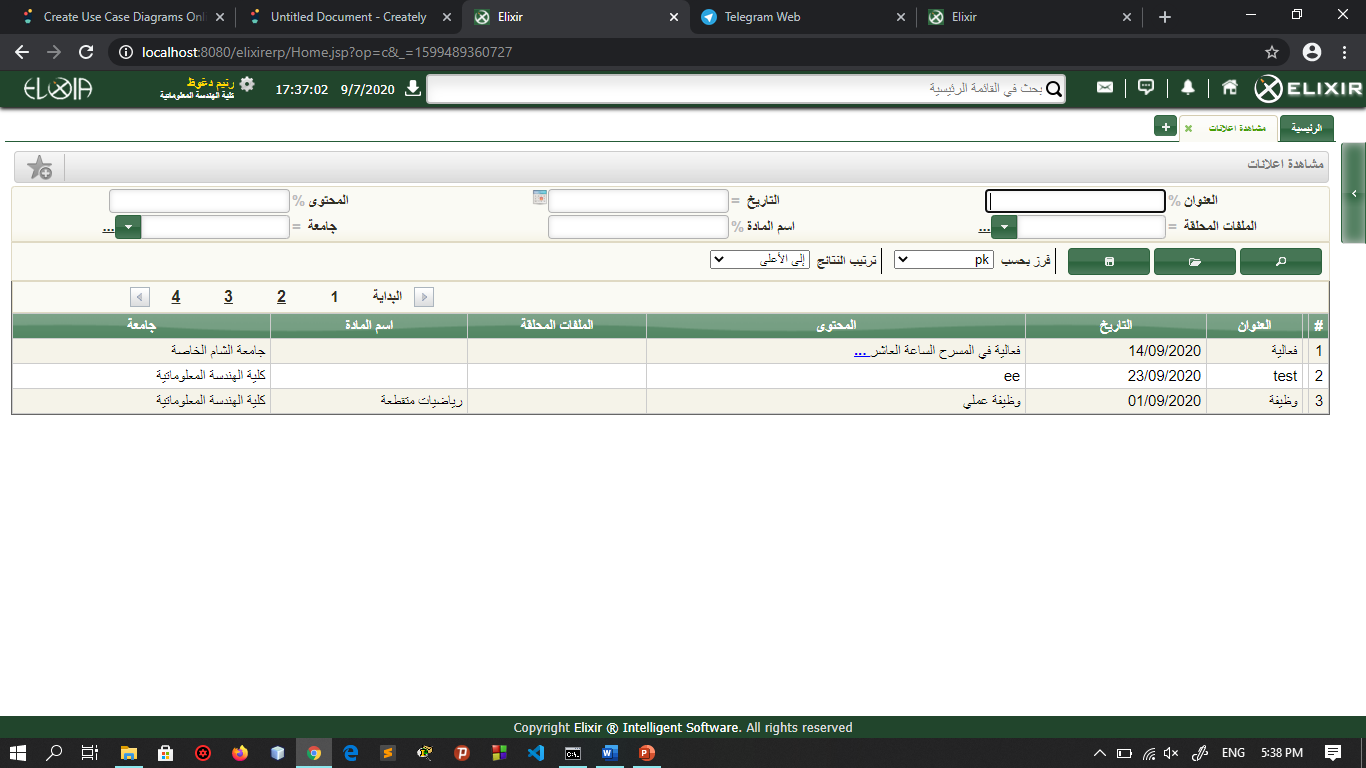
واجهة مشاهدة الإعلانات: للطالب و الأستاذ يتم عرض الإعلانات على مستوى جامعته و كليته و المواد التي قام بمتابعتها، أما الموظف الإداري فيستطيع مشاهدة جميع الإعلانات التي تم نشرها. في حالة كان أستاذ أو موظف إداري سيكون لديه صلاحية إضافة إعلان أما إذا كان طالب فلن يظهر له هذا الخيار.



الشكل 5: واجهة مشاهدة الإعلانات للأستاذ

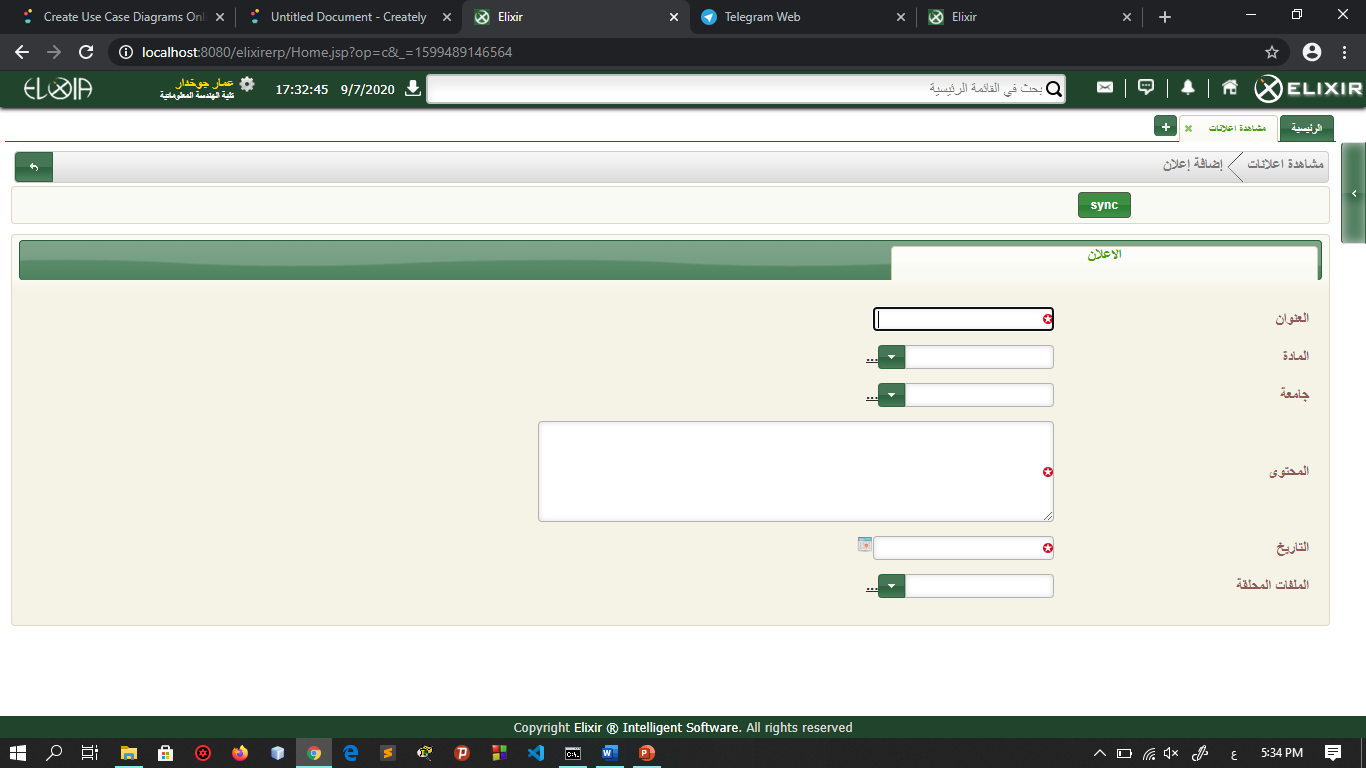


الشكل 6: واجهة مشاهدة الإعلانات للموظف الإداري



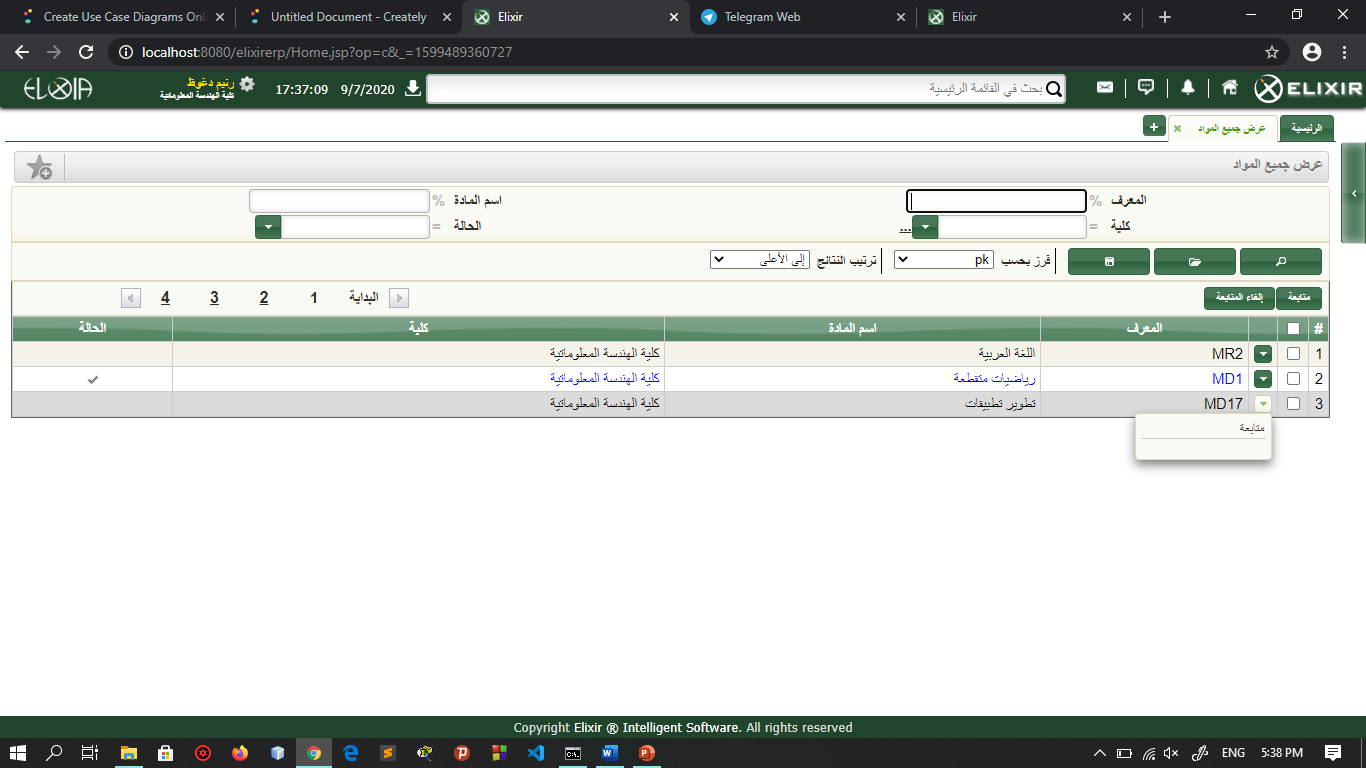
الشكل7: واجهة مشاهدة الإعلانات للطلاب

واجهة إضافة إعلان

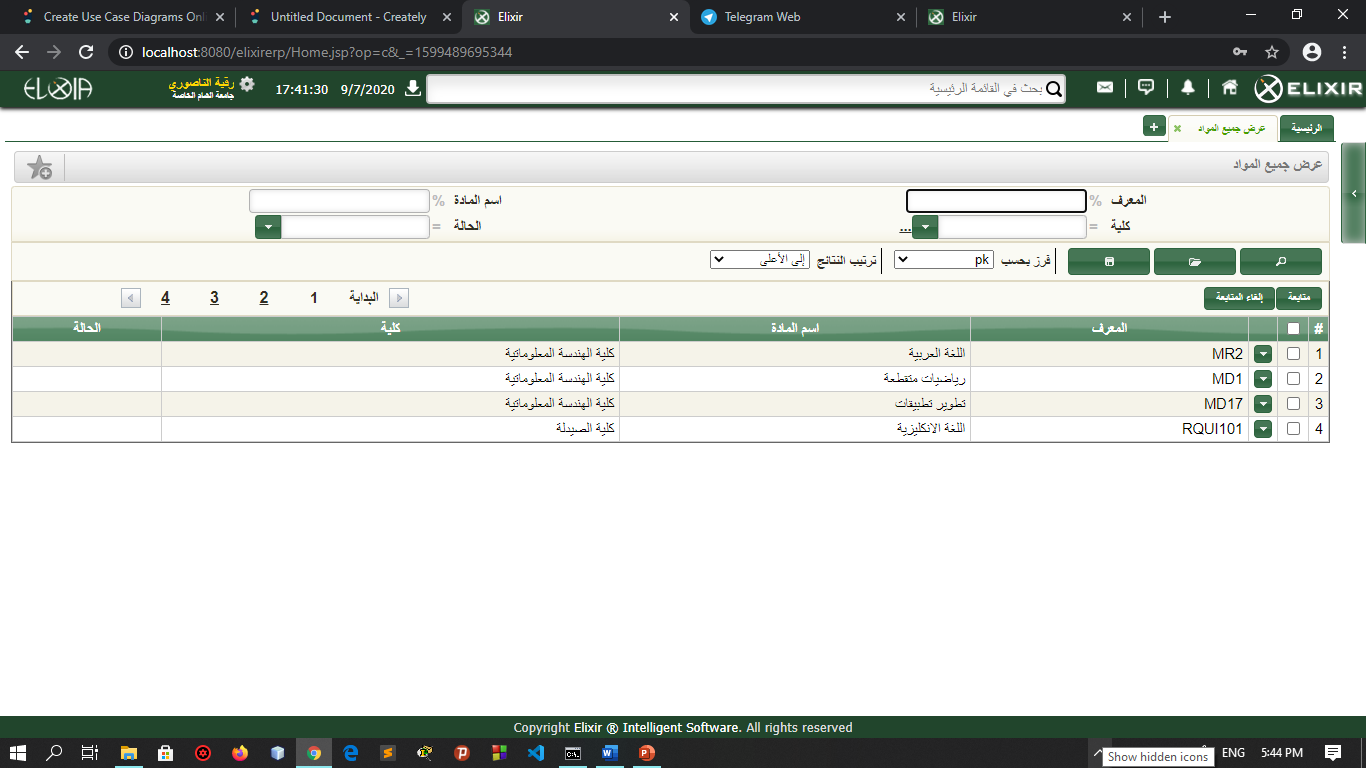


الشكل 8: واجهة إضافة إعلان

واجهة عرض المواد: تظهر عند الأستاذ والطالب مواد كليتهم فقط أما عند الموظف الإداري فتظهر مواد الجامعة كلها، وفي حال كان المادة متابعة فتظهر باللون الأزرق وأما المادة الغير متابعة فتظهر باللون الأسود

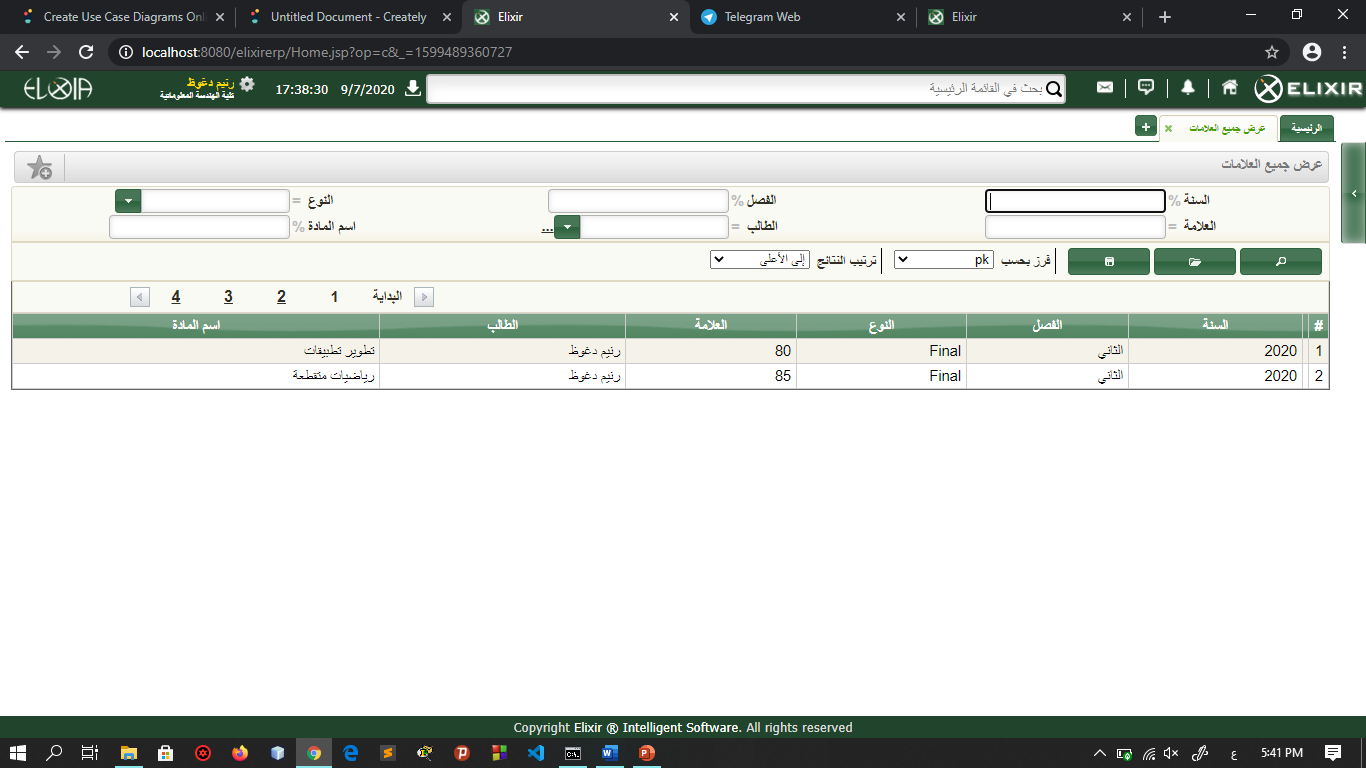


الشكل9: واجهة عرض المواد عند الطالب أو الأستاذ

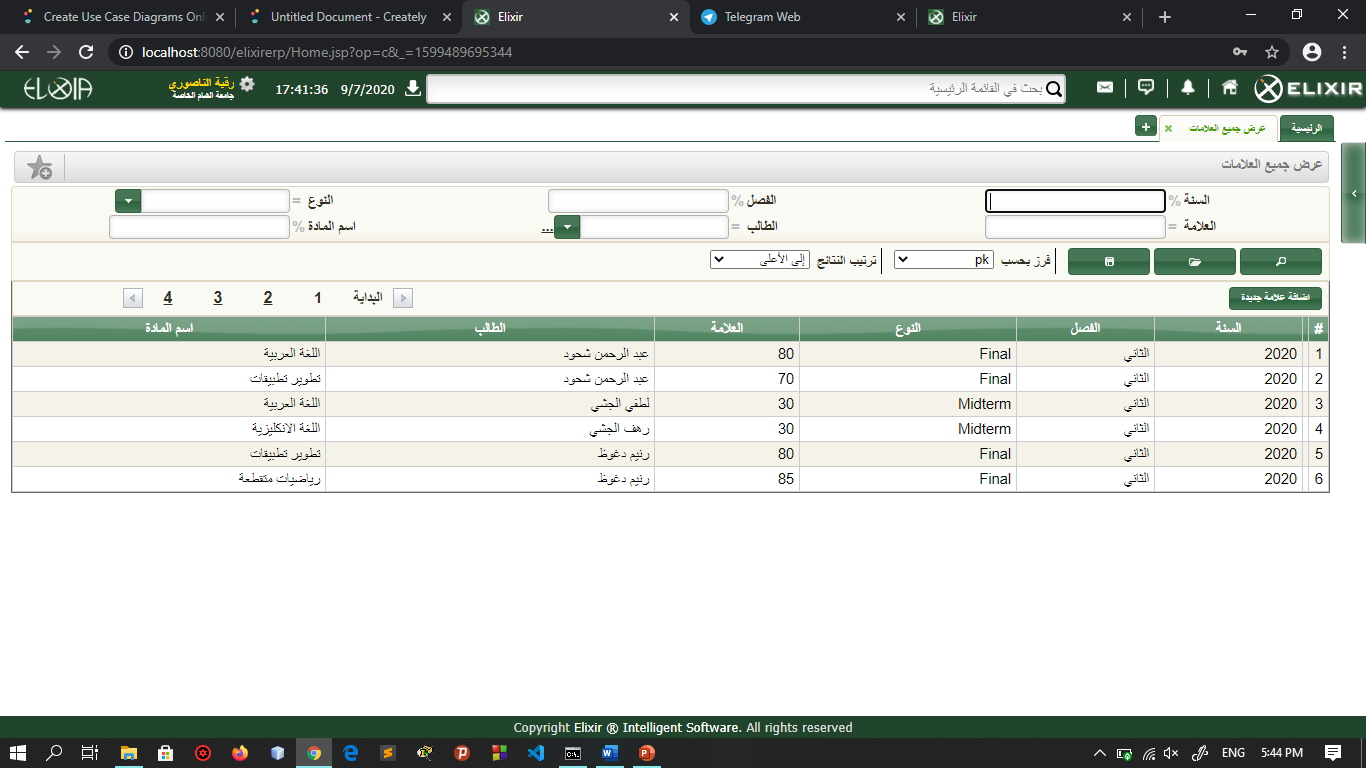


الشكل10: واجهة عرض المواد عند الموظف الإداري

واجهة عرض العلامات: يظهر عند كل طالب علاماته هو فقط ويظهر عن الموظف الإداري علامات جميع الطلاب مع صلاحيته لإضافة علامات للطلاب.

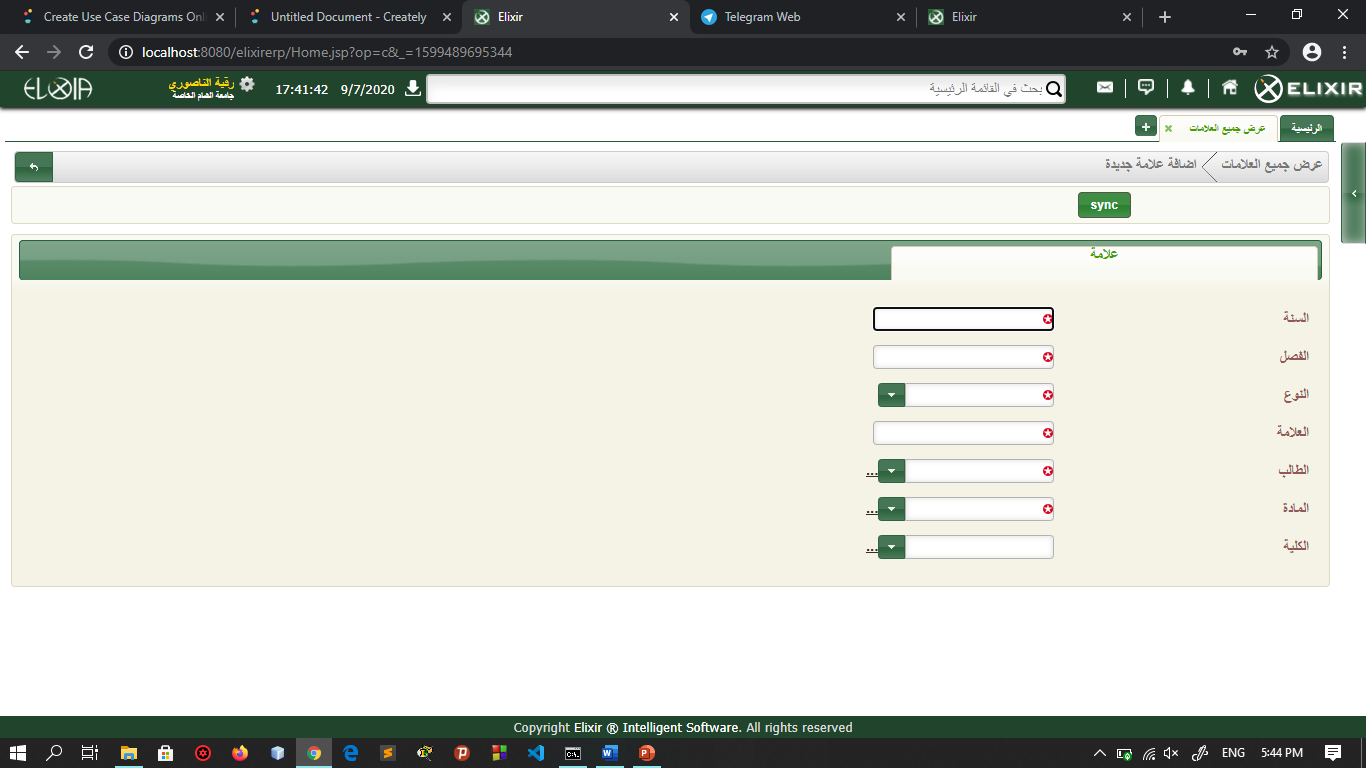


الشكل11: واجهة عرض العلامات عند الطالب



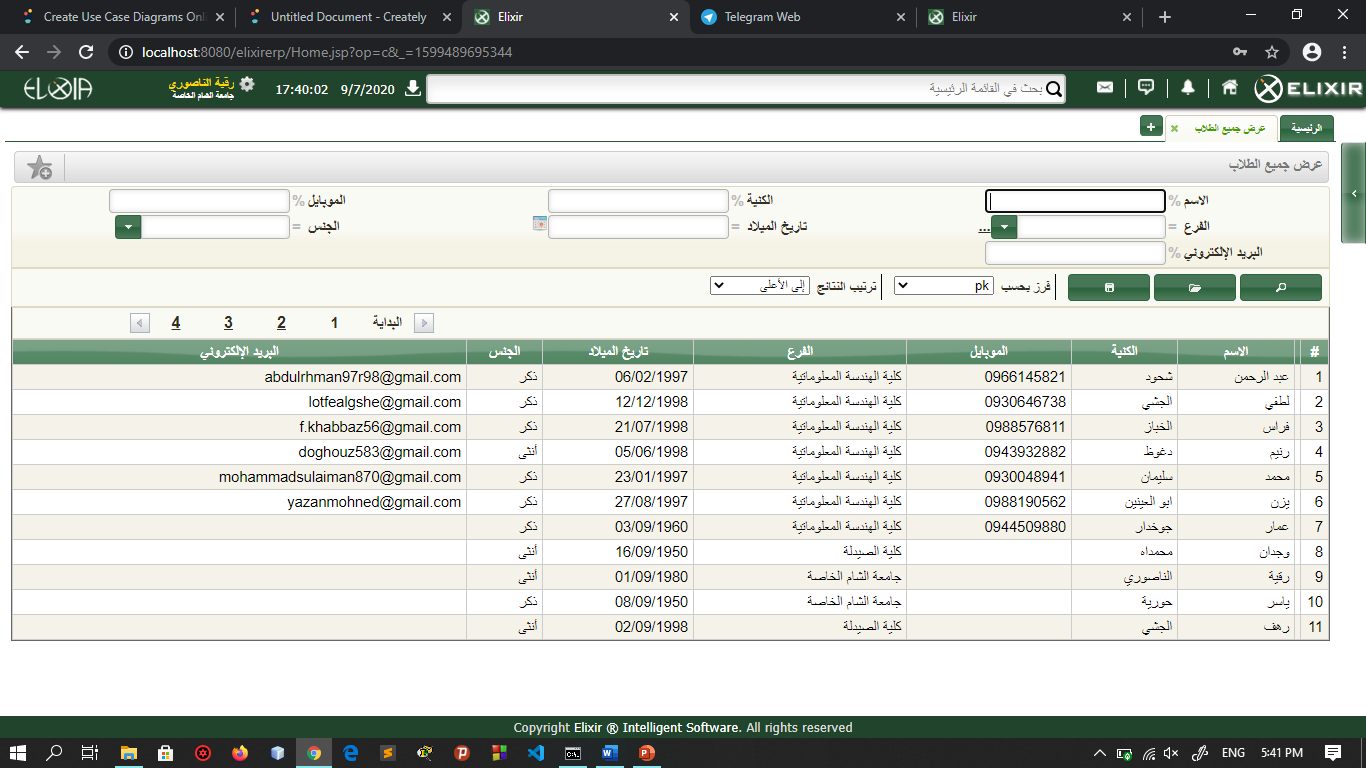
الشكل12: واجهة عرض العلامات عند الموظف الإداري

واجهة إضافة علامة: تظهر عند الموظف الإداري فقط



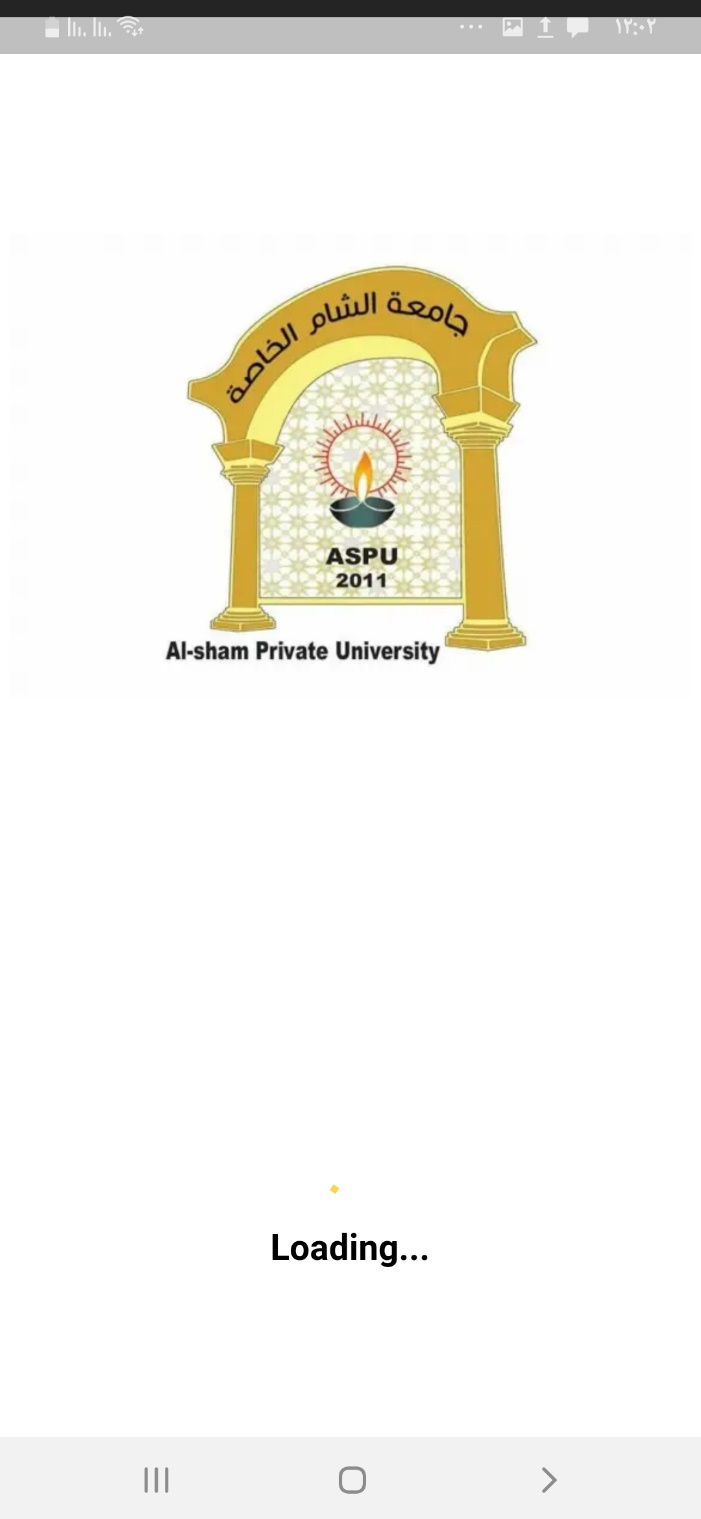
الشكل13: واجهة إضافة العلامات للطلاب

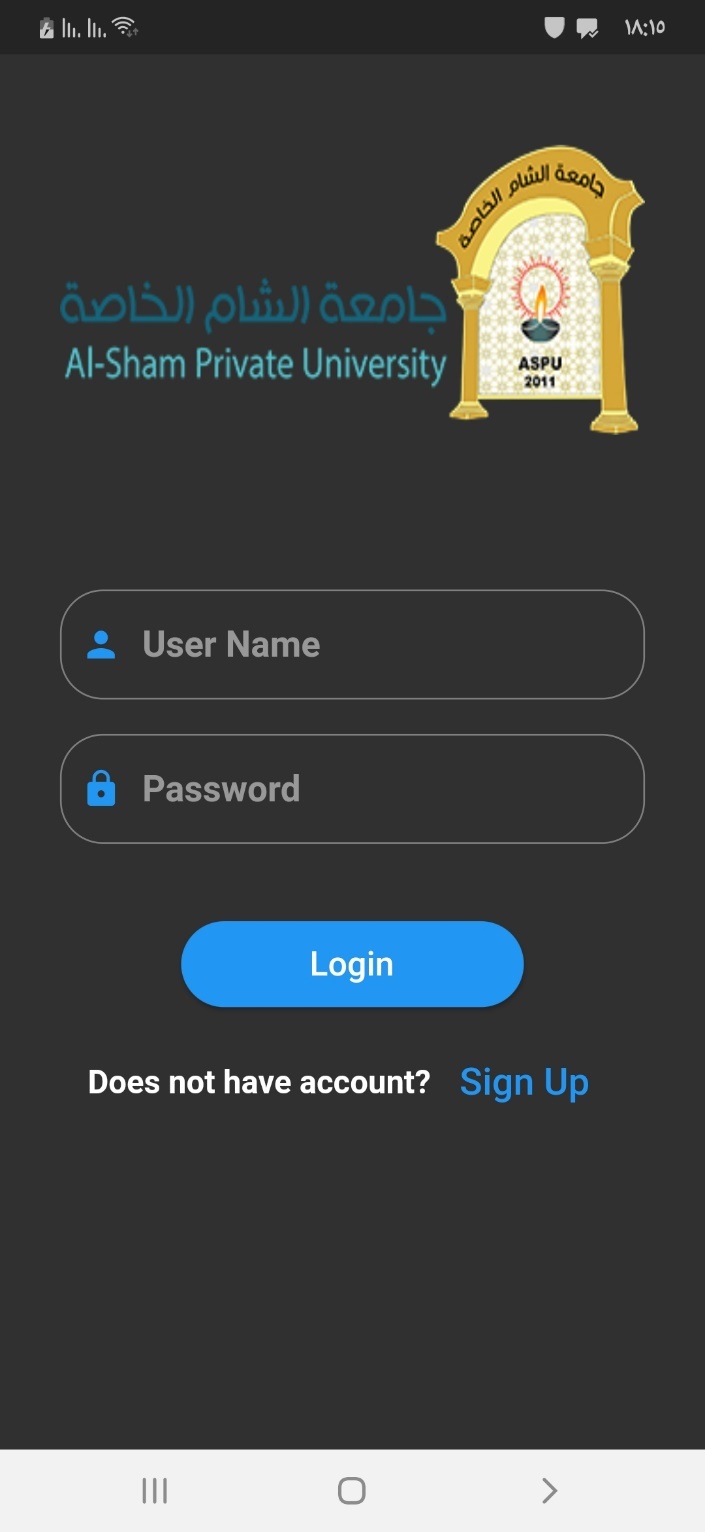
واجهة عرض جميع الطلاب والموظفين والأساتذة: وتكون صالحة للموظف الإداري فقط



الشكل14: عرض جميع الطلاب والموظفين والأساتذة

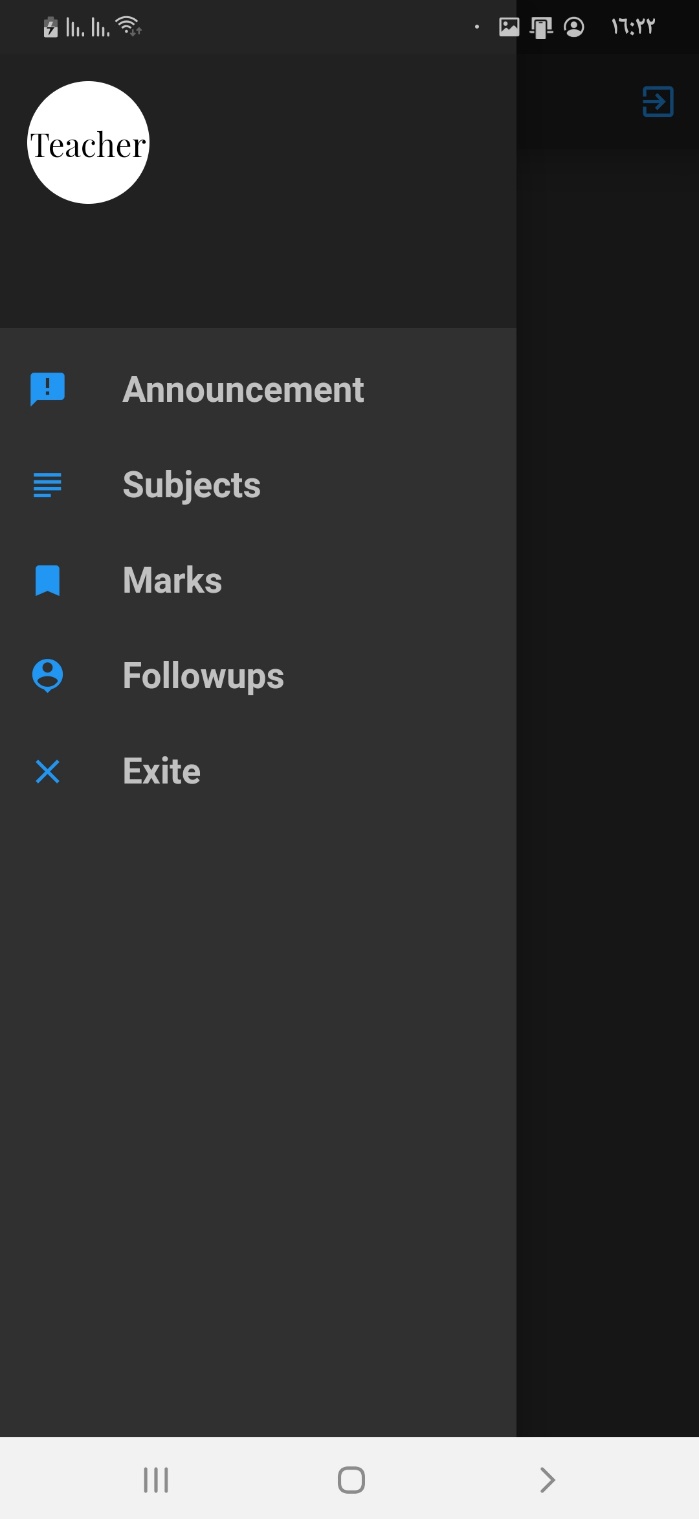
**واجهات التفاعل مع النظام في تطبيق الاندرويد:**

يبدأ التطبيق عندما يضغط المستخدم على الأيقونة الخاصة به في هاتفه، عندما يبدأ البرنامج يتم عرض شاشة البدء وبعد مرور خمس ثوان يتم عرض الواجهة الرئيسية والتي تكون واجهة تسجيل الدخول (الشكل 4: الشاشة الرئيسية):



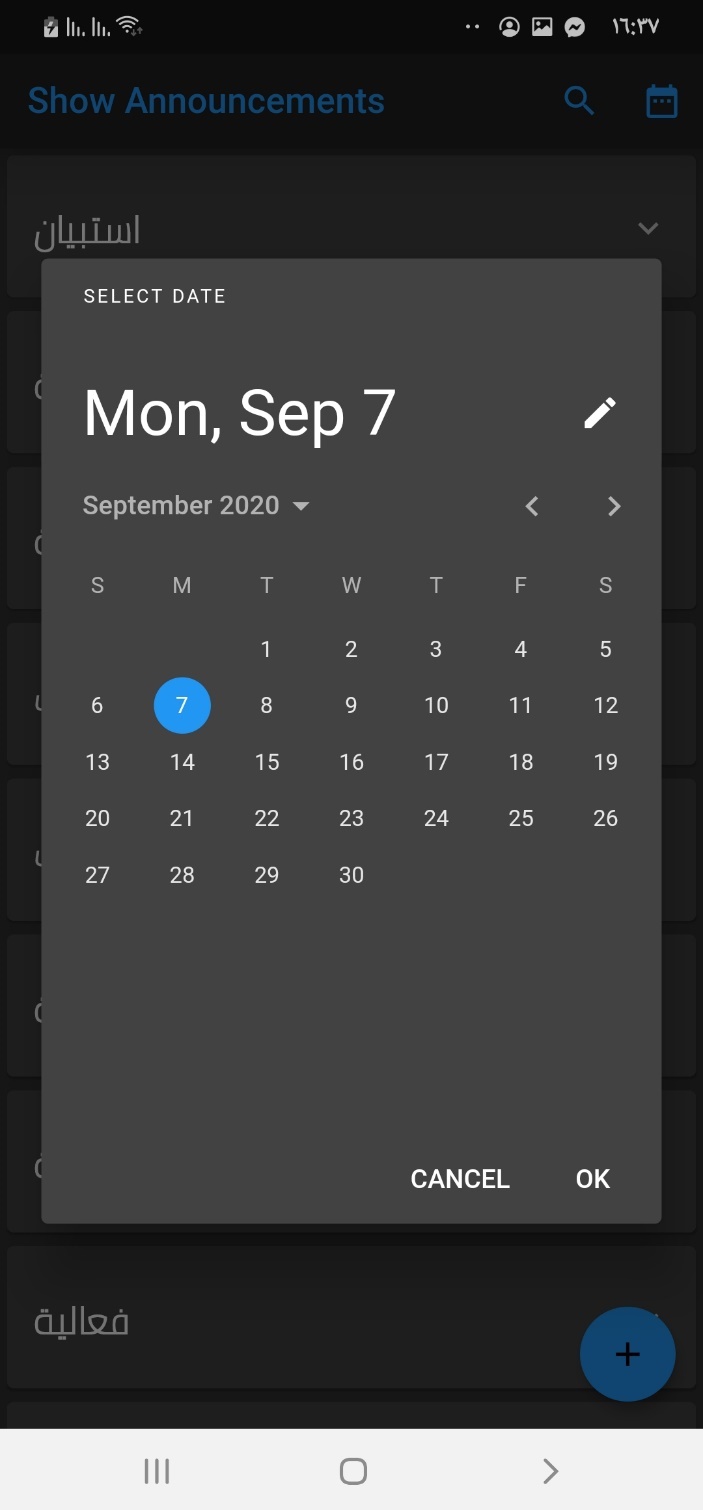
الشكل 16: الشاشة الرئيسية الشكل 15: شاشة البدء

بعد التحقق من اسم المستخدم وكلمة المرور إذا كان المستخدم أستاذ تعرض الشاشة الرئيسية للأستاذ وعندها تظهر له الشاشة التالية بالشكل 5



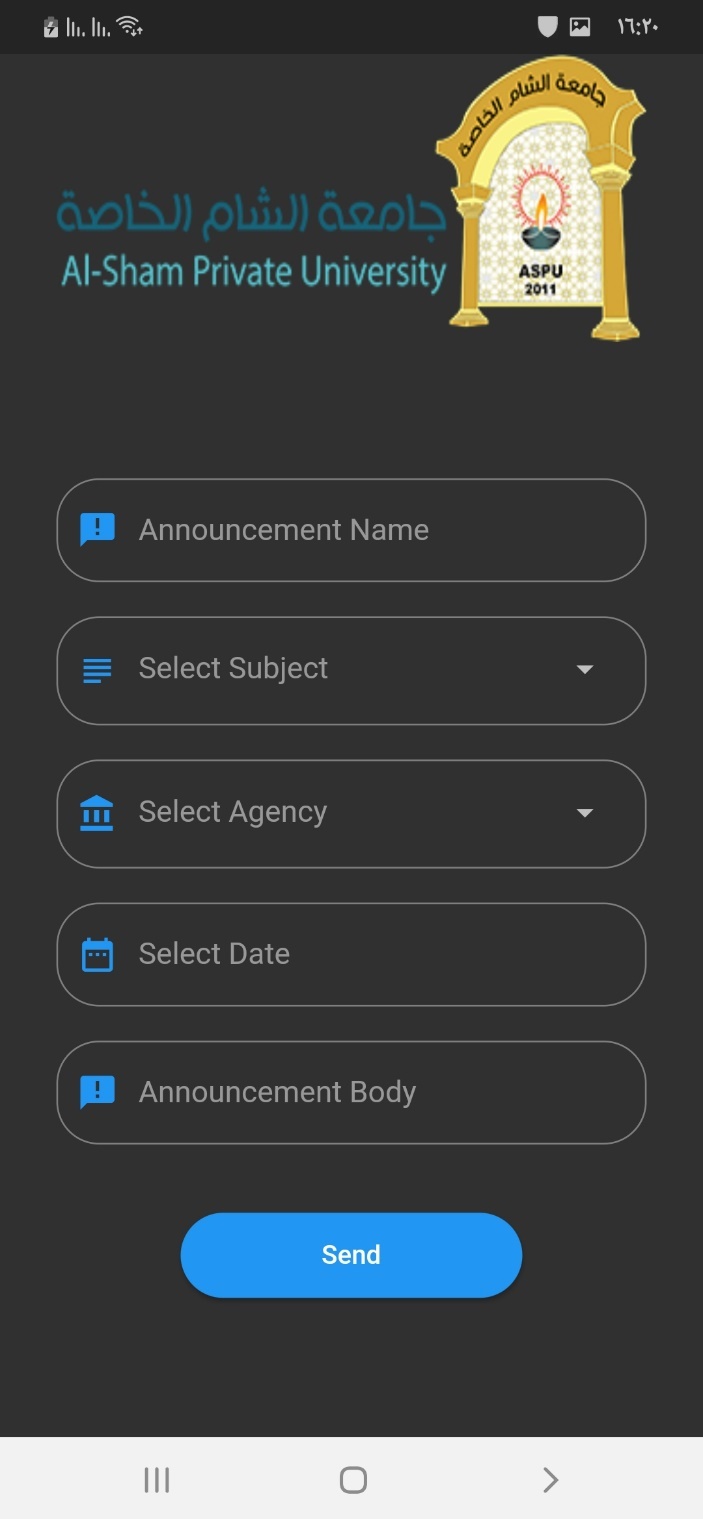
الشكل 17: الشاشة الرئيسية للأستاذ

* + وعندها يستطيع الأستاذ التنقل بين الواجهات المبينة في القائمة الرئيسية كما في الشكل 5
  + بعد اختيار الأستاذ لواجهة Announcements تظهر له الإعلانات بالإضافة لإمكانية البحث عن طريق إدخال اسم إعلان معين او تاريخ للإظهار الإعلانات التي أصدرت في التاريخ المختار كما هو مبين بالشكل 6



الشكل 18: شاشة عرض الإعلانات الشكل 19: شاشة الفرز عن طريق التاريخ

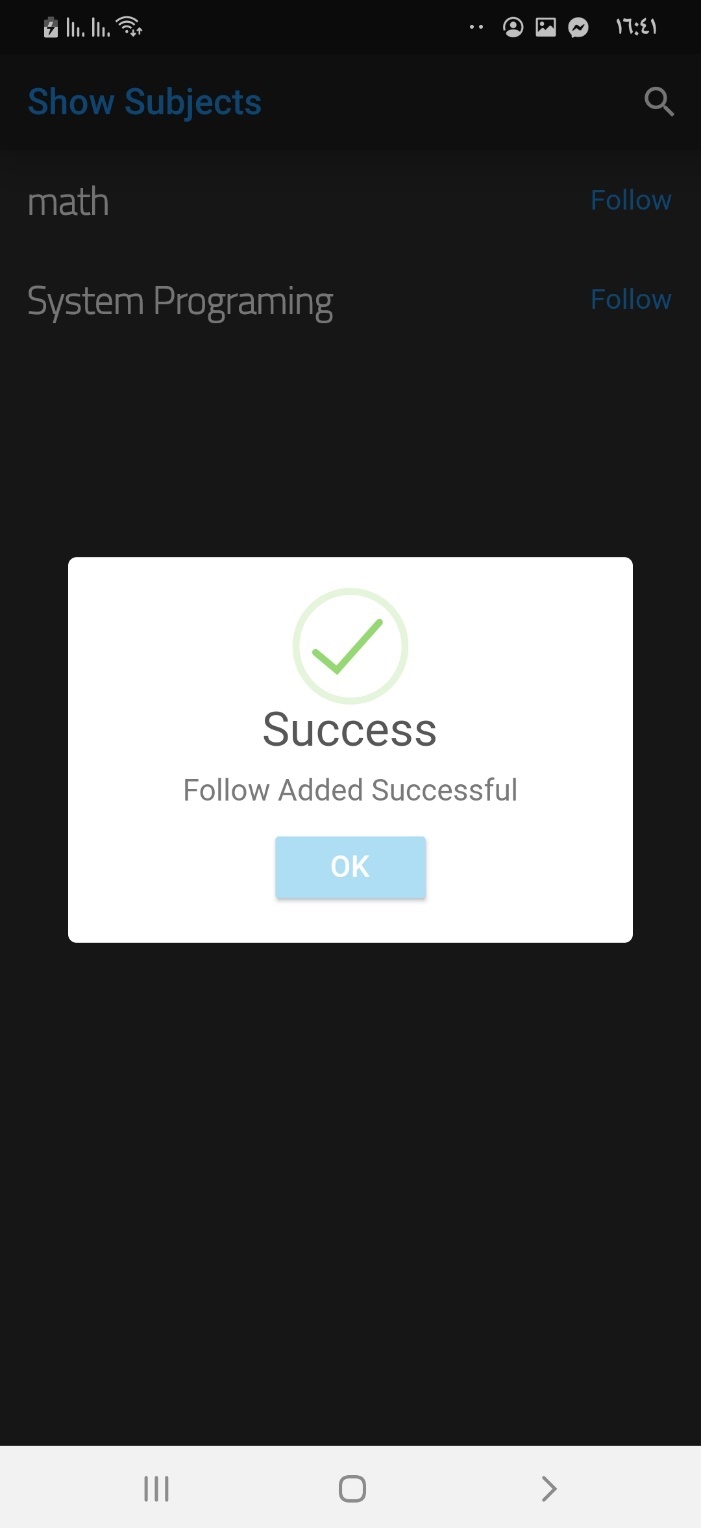
* عندما يقوم الأستاذ بالضغط على زر إضافة تظهر له واجهة إضافة إعلان كما هو مبين بالشكل 8 وعند نجاح عملية إضافة إعلان تظهر Toast يؤكد نجاح الطلب كما بالشكل 9





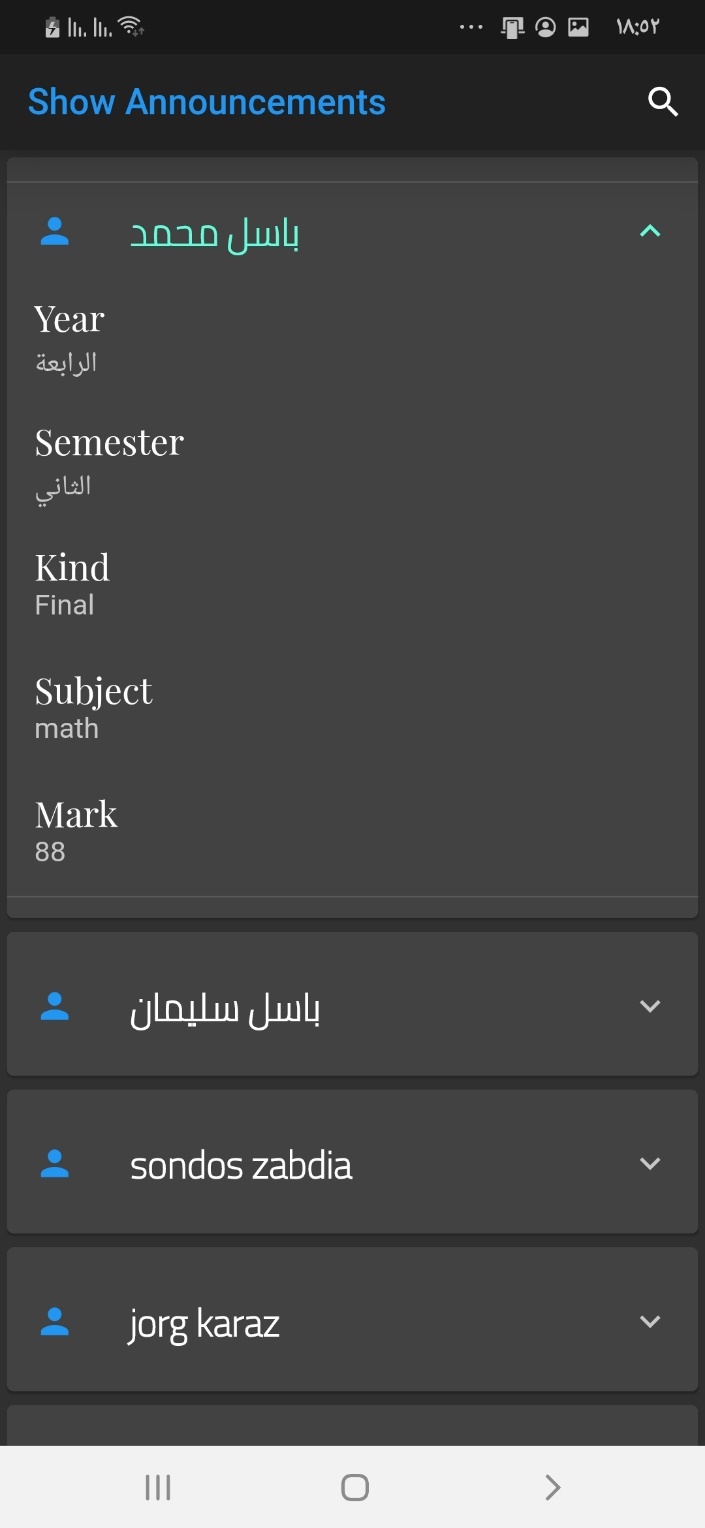
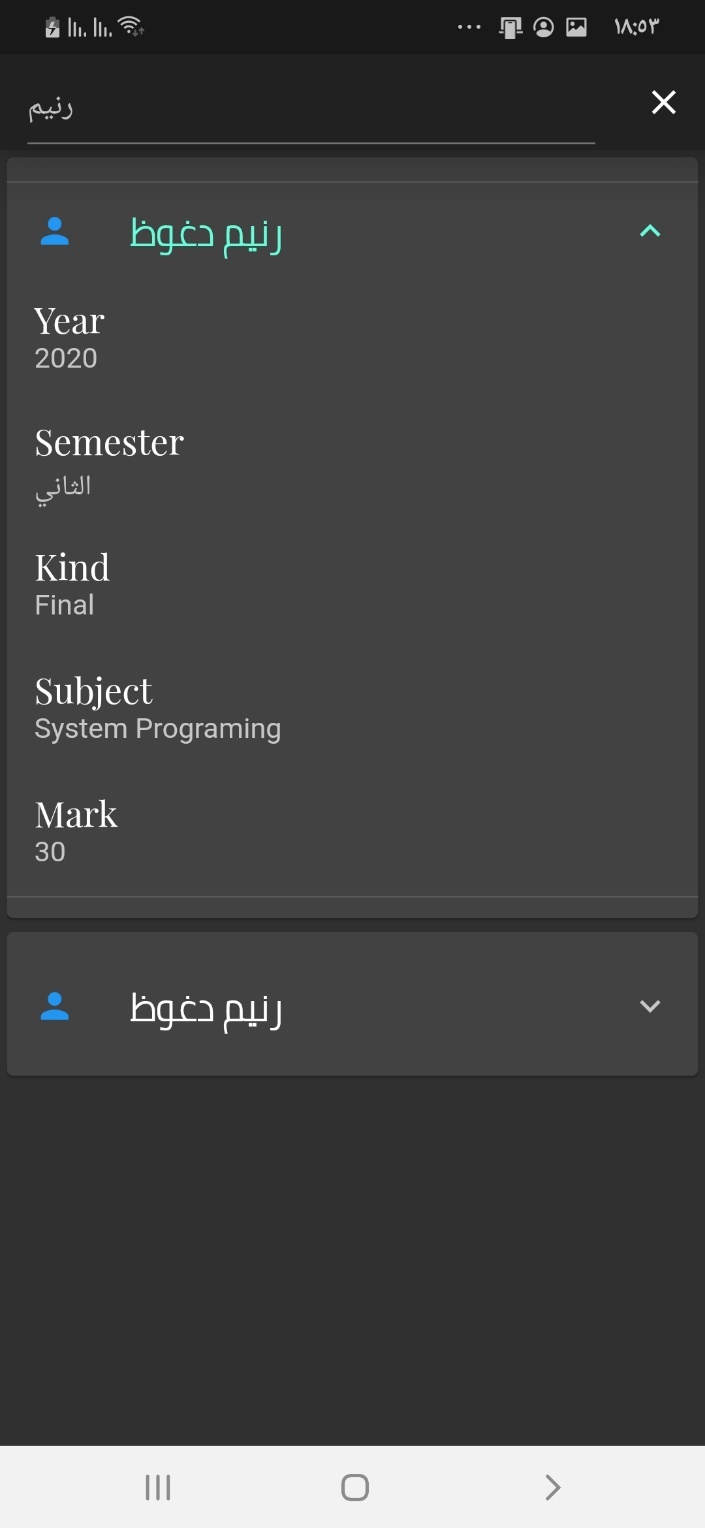
الشكل 20: شاشة إضافة الإعلانات الشكل 21: شاشة تأكيد عملية الإضافة

* + بعد اختيار الأستاذ لواجهة Subjects تظهر له المواد المتوفرة كما هو موضح بالشكل 9 بالإضافة لإمكانية إضافة متابعة عن طريق الضغط على زر Follow ولإلغاء المتابعة الضغط مطولا عل زر Follow كما بالشكل 10



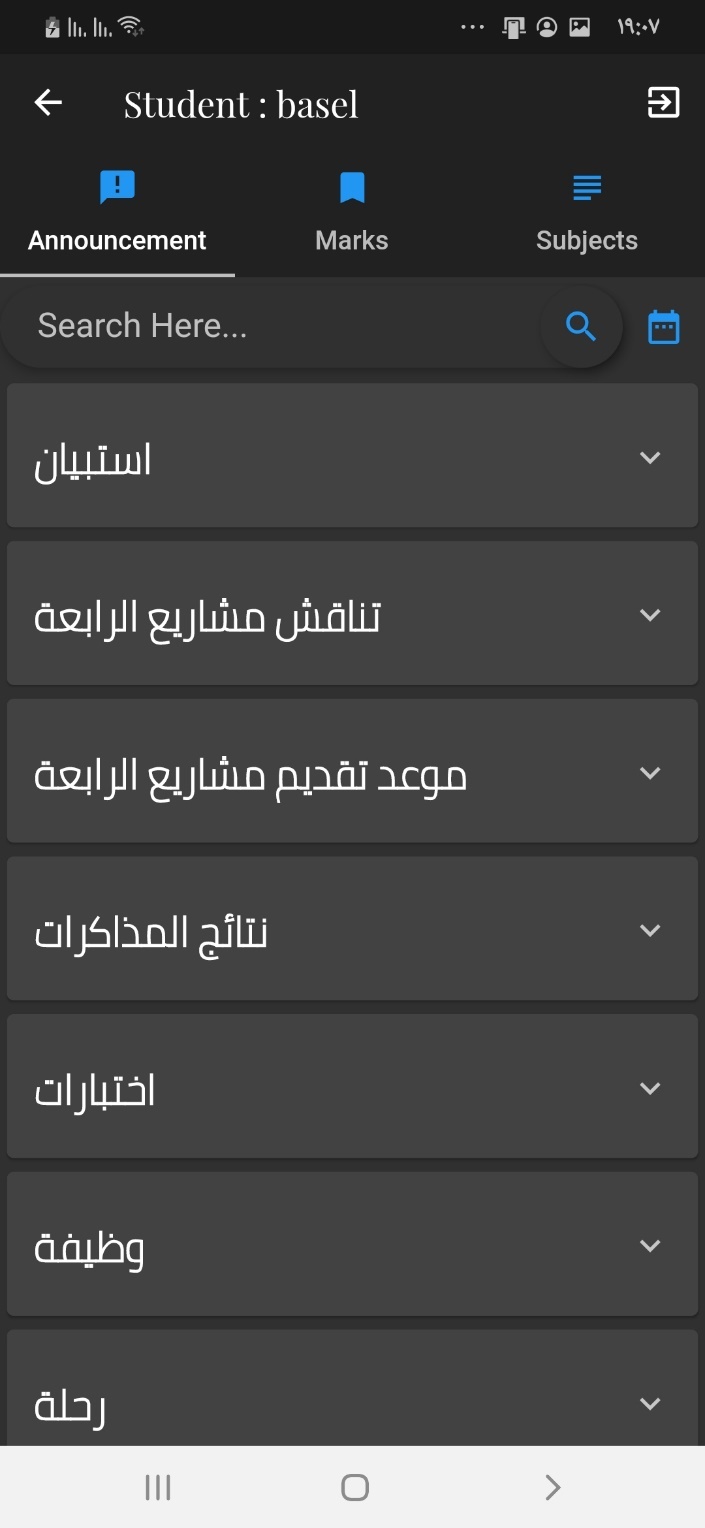
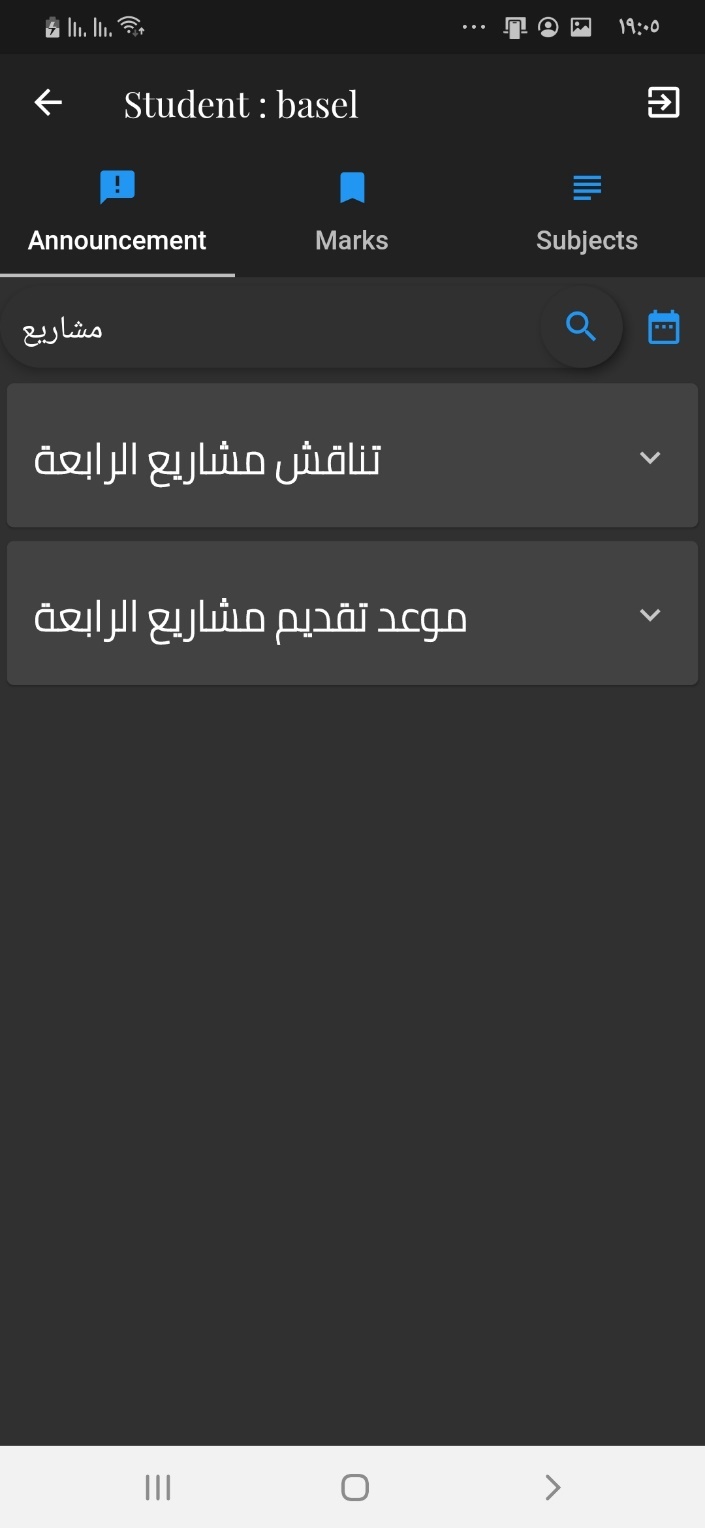
الشكل 22: شاشة عرض المواد الشكل 23: شاشة تأكيد عملية المتابعة

* + بعد اختيار الأستاذ لواجهة Marks تظهر له علامات الطلاب كما في الشكل 11 بالإضافة لإمكانية البحث والفرز عن طريق إدخال اسم الطالب او اسم المادة كما في الشكل 12

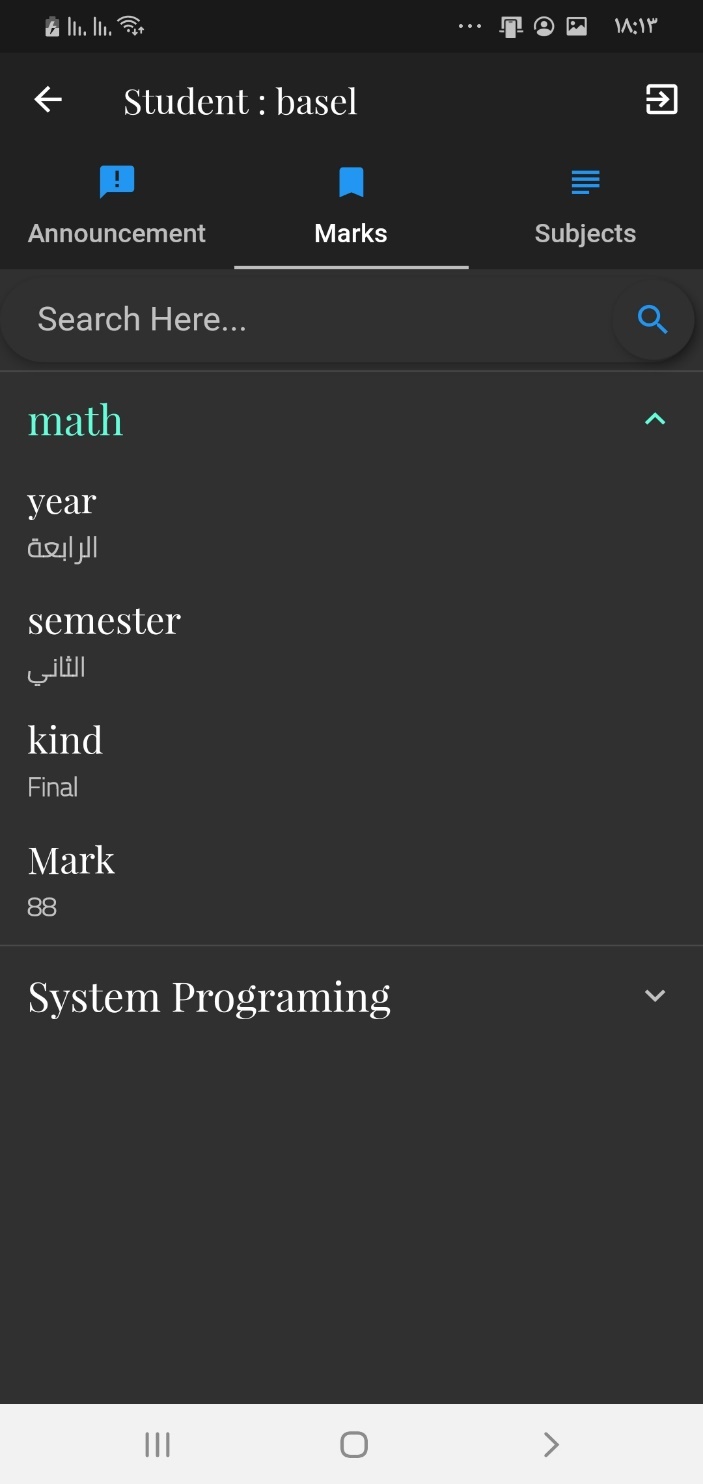
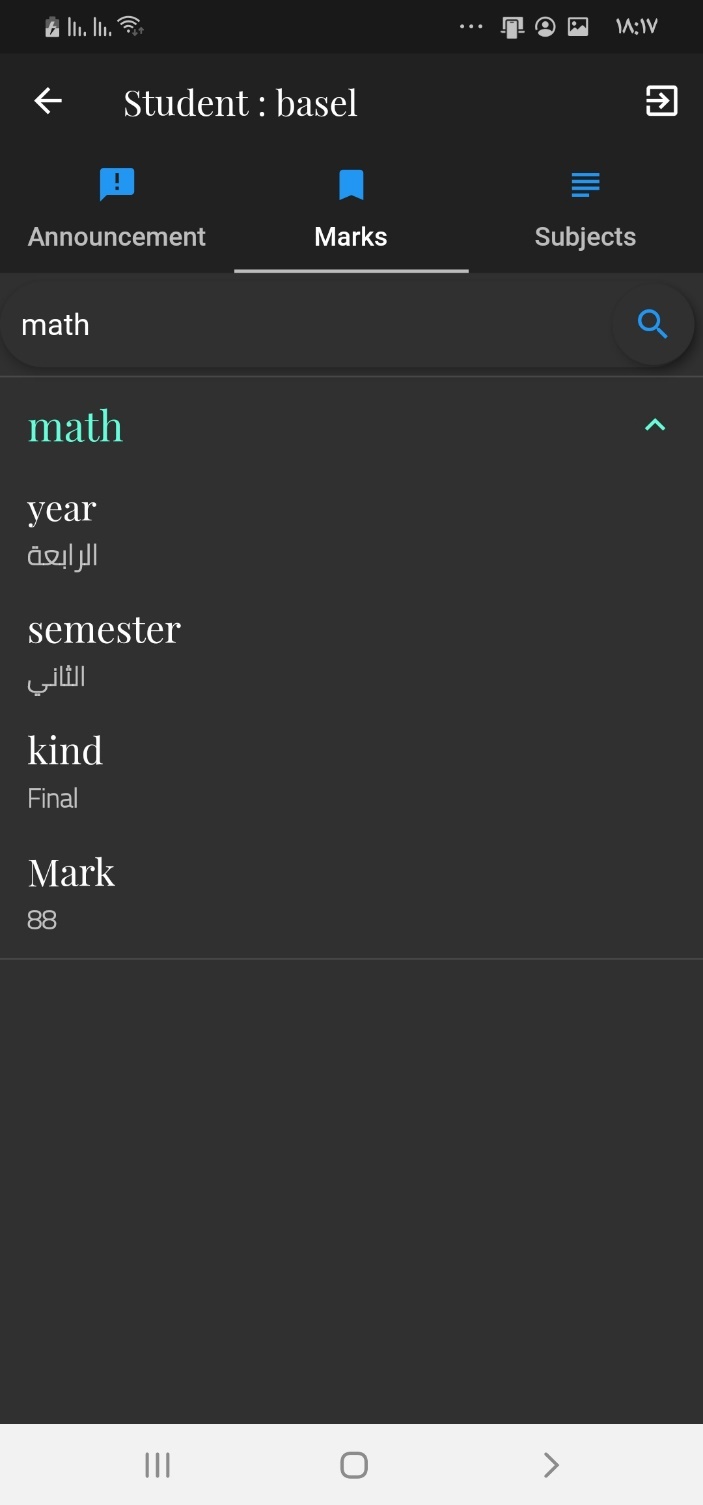


الشكل 24: شاشة علامات الطلاب الشكل 25: شاشة بحث عن طالب

* + إذا كان المستخدم طالب تعرض الشاشة الرئيسية للطالب Announcements كما في الشكل 14 وإمكانية البحث وفرز الإعلانات عن طريق اسم الإعلان او محتوى الإعلان والتاريخ كما في الشكل 15

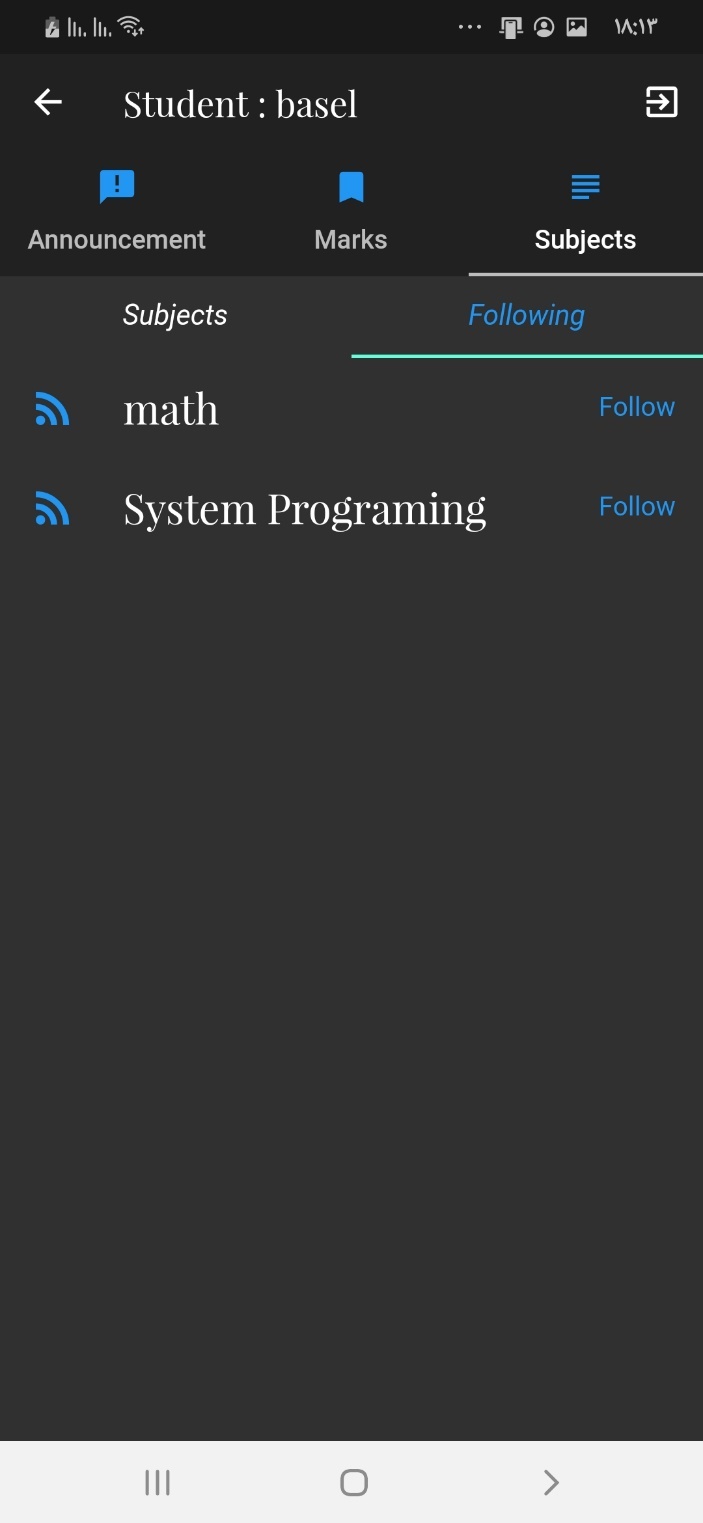
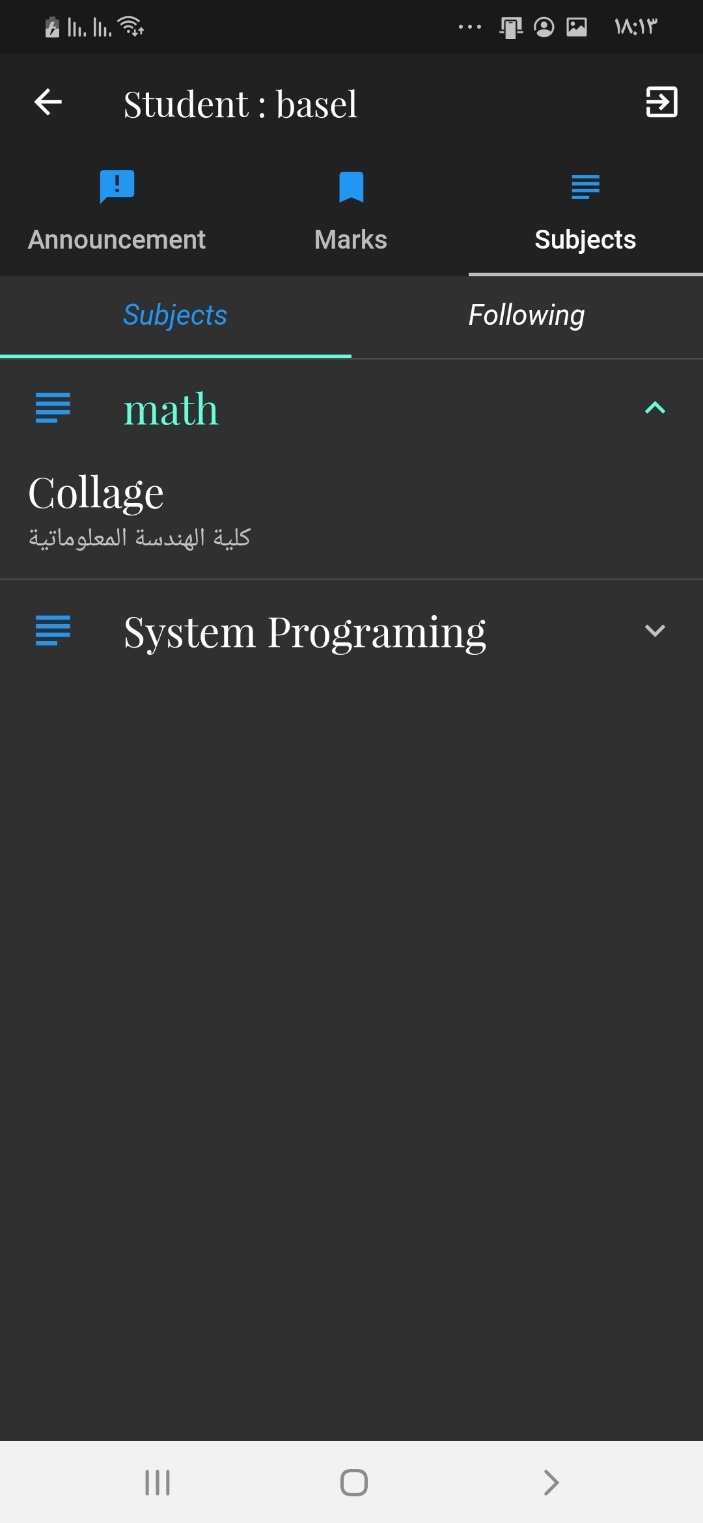


الشكل 26: شاشة إعلانات الطالب الشكل 27: شاشة بحث وفرز حسب اسم إعلان

* + بعد اختيار الطالب واجهة Marks تظهر له علاماته بالمواد التي قام هو بمتابعتها كما في الشكل 16 بالإضافة لإمكانية البحث والفرز عن طريق إدخال اسم المادة كما في الشكل 17
* 

الشكل 28: شاشة عرض العلامات الشكل 29: شاشة البحث والفرز حسب اسم مادة

* + وعند اختياره واجهة Subjects تظهر له واجهتان هما Subjects وهي الواجهة التي تحتوي على المواد المتاحة كما في الشكل 18 وواجهة Following وهي الواجهة التي يستطيع من خلالها إضافة متابعة كما في الشكل 19



الشكل 30: شاشة المواد المتوافرة الشكل 31: شاشة إضافة متابعة

## 6.6- مشاكل وصعوبات

خلال مراحل العمل بالمشروع صادفتنا العديد من المشاكل، لكن كانت تقليدية بالنسبة لنا وحلها يتطلّب بذل المزيد من الجهد والوقت.

* **العائق الأول:**

تعقيدات البيئة المستخدمة وخاصة أنها بيئة جديدة وتطلب مني الوقت الطويل للتغلب على مشاكل التنصيب والتعامل معها، بالإضافة إلى مشكلة تحميل المكتبات بِسبب صعوبة تعامل برنامج Android Studio مع شبكة المعهد ووجود البروكسي الذي يمنع تحميلها وقد تمّ حلّ المشكلة باستخدام برنامج كسر بروكسي (your-freedom).

* **العائق الثاني:**

1. التعرف على بيئات العمل المؤسساتية عموما والإكسير خصوصا
2. التعرف على بيئات تطوير الموبايل وخاصة flutter
3. التعرف العمل online offline
4. دعم بيئة متعددة الفروع
5. تحقيق الأمن الوظيفي

* **العائق الثالث:**

تعقيدات التعامل مع قاعدة البيانات SQLITE ولا سيما في عمليات حفظ والاسترجاع من قاعدة البيانات

المخزنة في الهاتف المحمول أثناء انقطاع الاتصال بالأنترنيت

# الخاتمة والآفاق المستقبلية

قمنا في هذا المشروع بِبناء موقع وتطبيق يعمل على الهواتِف الذكيّة، حيث يسمح للطالب والأستاذ بمشاهدة مواد كليته وتحديد المواد التي يريد متابعتها، ويستطيع مشاهدة إعلانات الجامعة، وإعلانات كليته، وإعلانات المواد المهتم بها.

إضافة إلى ذلك فإن هذا التطبيق يسمح للأستاذ والموظف الإداري بإضافة إعلانات يريد نشرها.

أيضا جعلنا للموقع إمكانية إدخال علامات الطلاب ومعلوماتهم الشخصية واستيرادها وتصديرها من قبل الموظفين الإداريين.

كما أعطينا للطالب خصوصية تامة للاطلاع على علاماته بشكل شخصي.

كما يتيح (تطبيق الاندرويد) مشاهدة الإعلانات التي تم تحميلها مسبقاً في حال عدم توافر اتصال بالانترنت، وإمكانية البحث عن إعلان ما أو علامة لمادة معينة.

يمكن توسيع المشروع عن طريق جعل النظام يشمل أعمال الجامعة الإدارية كافة.

أيضاً جعل الأستاذ يضيف معلومات مفيدة مختصرة للمادة بحيث تسهل وتساعد الطلاب في دراستهم.

ويمكننا إضافة بنك أسئلة لكل مادة، من خلاله يستطيع النظام توليد ورقة امتحان بشكل أتوماتيكي.

# المراجع

|  |  |
| --- | --- |
| [1] | 30 12 2015. [Online]. Available: https://mawdoo3.com/الهواتف\_الذكية. |
| [2] | 1 7 2016. [Online]. Available: https://www.statista.com/statistics/385001/smartphone-worldwide-installed-base-operating-systems/. |
| [3] | [متصل]. Available: https://www.wallaapps.com/2019/11/flutter.html. |
| [4] | [Online]. Available: http://www.json.org. |
| [5] | [Online]. Available: play.google.com. |
| [6] | [Online]. Available: https://en.wikipedia.org/wiki/Google\_Maps. |
| [7] | [Online]. Available: https://geojson.org/geojson-spec.html. |
| [8] | [Online]. Available: https://developers.google.com/maps/terms. |
| [9] | [Online]. Available: http://www.microimages.com/documentation/TechGuides/78googleMapsStruc.pdf. |
| [10] | [Online]. Available: https://www.codeproject.com/Articles/14793/How-Google-Map-Works. |
| [11] | [Online]. Available: https://developers.google.com/maps/documentation/utilities/polylinealgorithm. |
| [12] | [متصل]. Available: https://www.wallaapps.com/2019/11/flutter.html. |
| [13] | [متصل]. Available: https://www.wallaapps.com/2019/11/flutter.html. |