

نظام اتمتة مكتبة الخدمات الطلابية في الجامعة السورية الخاصة

مشروع السنة الرابعة في كلية هندسة الحاسوب و المعلوماتية
قسم : هندسة البرمجيات.

تقديم الطلاب :

مهند سميح حاتم - حسان فارس الباروكي - مرح خليل الحلبي - مارلين وائل شحادة

بإشراف :

الدكتور المهندس : غسان النمر.

المهندس : حسام بوفاعور.

جميع الحقوق محفوظة للجامعة السورية الخاصة – 2013 ©

بكل محبة واحترام ...

نتوجه بالشكر الى الكادر التدريسي في الجامعة السورية الخاصة من دكاترة ومهندسين على جهودهم المبذولة وعطائهم الكبير ، الذي أوصلنا الى هذه المرحلة من المعرفة ، كما نشكر السادة الاداريين على حسن تعاملهم . ونخص بالشكر الدكتور المهندس غسان النمر المشرف على المشروع على اتعابه معنا ومتابعتنا الدائمة ودعمه الكامل بكل المعلومات اللازمة .

كما نشكر الاستاذ المهندس حسام بوفاعور الذي كان عوناً لنا طوال فترة العمل.

كما نتقدم بالشكر للصديق المهندس عامر جودية على نصائحه وتوجيهاته القيمة طوال فترة العمل في المشروع وفي كافة المجالات .

الخلاصة

تعد أنظمة الأتمتة من أكثر التطبيقات شيوعاً و انتشاراً في عالم المعلوماتية نظراً لما تقدمه من تسهيلات في اختصار الوقت و الموارد البشرية، إضافة الى موثوقيتها العالية.

مجالات هذه الأنظمة غير محدودة ، و أثبتت فعالية كبيرة و إقبال منقطع النظير في كافة القطاعات التي دخلتها.

ومن أقسامها الهامة أنظمة أتمتة المكتبات ، التي تعنى في تنظيم كافة العمليات الخاصة بالمكتبات من طباعة

و أرشفة، و تخزين، و نشر، و بيع، و اعارة ... الخ.

في هذا التقرير نعرض عملنا في إطار مقرر المشروع الفصلي في كلية هندسة الحاسوب و المعلوماتية في الجامعة السورية الخاصة ، و الذي يتناول تطوير نظام أتمتة خاص بمكتبة الخدمات الطلابية في الجامعة (مركز النسخ و الطباعة) ، ليكون الحل الأمثل والأسهل لتنظيم أعمال المكتبة بشكل سريع و دقيق يختصر الوقت على الطلاب ، و يقدم مساعدة كبيرة للعاملين في المكتبة.

وهذا النظام مبني على قواعد بيانات SQL Server ، و مدعوم بواجهات C#.

جدول المحتويات	
الموضوع	رقم الصفحة
المقدمة	1
الفصل الأول : تلخيص الأفكار والاحتياجات	2
1.1- المشكلة المراد حلها	2
1.2- الجهود المطبقة في هذا المجال	2
1.3- أهمية استخدام هذا النظام	4
1.4- الاستبيان	4
1.5- صعوبة تطوير مثل هذا النظام	6
1.6- النتيجة المقترضة تحقيقها	6
1.7- متطلبات نظام أتمتة أعمال مكتبة الخدمات الطلابية في الجامعة	6
الفصل الثاني : الأدوات المستخدمة	7
2.1- المفاهيم التي يمكن الاستفادة منها	7
2.2- لغة البرمجة المستخدمة	7
2.3- بيئة قواعد البيانات المستخدمة	7
2.4- برنامج إعداد الدراسة التحليلية	7
الفصل الثالث : الدراسة التحليلية	8
3.1- مخطط حالات الاستخدام (Use Case Diagram)	8
3.2- توصيف حالات الاستخدام (Use Cases Description)	9
الفصل الرابع : تصميم قاعدة البيانات	20
4.1- مخطط الكيانات والعلاقات	20
4.2- بناء قاعدة البيانات	21
الفصل الخامس : فحص النظام	22
5.1- الطريقة المتبعة	22
5.2- الدخول الى البرنامج	22
5.3- اضافة / تعديل طالب	23
5.4- واجهة ادارة الموجودات	28
5.5- البحث عن المحاضرات	32
5.6- خيارات النظام	37
الفصل السادس : المقترحات المطروحة لتطوير النظام	43
المراجع	44
الملحق (الكود)	45

قائمة الرسوم التوضيحية

رقم الصفحة	العنوان	الرسم التوضيحي
2 تطبيق Copy Print Room	رسم توضيحي 1
4 ورقة الاستبيان	رسم توضيحي 2
8 مخطط حالات الاستخدام	رسم توضيحي 3
20 مخطط ال ERD	رسم توضيحي 4
21 مخطط قاعدة البيانات	رسم توضيحي 5
22 واجهة الدخول الى النظام	رسم توضيحي 6
22 الدخول الخاطئ	رسم توضيحي 7
23 الواجهة الرئيسية للبرنامج	رسم توضيحي 8
23 واجهة اضافة وتعديل طالب	رسم توضيحي 9
24 عملية اضافة طالب	رسم توضيحي 10
24 عملية تعديل بيانات الطالب	رسم توضيحي 11
25 عملية تنزيل المواد	رسم توضيحي 12
26 اتمام عملية تنزيل المواد	رسم توضيحي 13
26 عملية حذف الطالب	رسم توضيحي 14
27 البحث عن طالب عن طريق اسمه الاول	رسم توضيحي 15
27 معالجة تسجيل مادة موجودة مسبقا	رسم توضيحي 16
28 واجهة ادارة الموجودات	رسم توضيحي 17
28 عملية اضافة كلية 1	رسم توضيحي 18
29 عملية اضافة كلية 2	رسم توضيحي 19
29 عملية اضافة كلية 3	رسم توضيحي 20
29 عملية اضافة مادة 1	رسم توضيحي 21

30	عملية اضافة مادة 2	رسم توضيحي 22
30	عملية اضافة مادة 3	رسم توضيحي 23
31	عدد الطلاب في كل مادة	رسم توضيحي 24
32	عملية البحث عن المحاضرات 1	رسم توضيحي 25
32	عملية البحث عن المحاضرات 2	رسم توضيحي 26
33	تعديل المحاضرات المحجوزة	رسم توضيحي 27
33	حجز محاضرة مأخوذة	رسم توضيحي 28
34	عرض عدد المحاضرات المحجوزة	رسم توضيحي 29
34	طلب محاضرة نفذت نسخها	رسم توضيحي 30
35	طلب عدد نسخ غير متوفر	رسم توضيحي 31
35	عرض المحاضرات المحجوزة	رسم توضيحي 32
36	معاينة الوصل	رسم توضيحي 33
37	واجهة خيارات النظام	رسم توضيحي 34
37	تسجيل الدخول الى خيارات النظام	رسم توضيحي 35
38	الجرد اليومي	رسم توضيحي 36
39	جرد يوم معين	رسم توضيحي 37
39	استعراض فاتورة مبيعة	رسم توضيحي 38
40	عملية اضافة مستخدم 1	رسم توضيحي 39
40	عملية اضافة مستخدم 2	رسم توضيحي 40
41	عملية تعديل المستخدمين	رسم توضيحي 41
41	عملية حذف مستخدم 1	رسم توضيحي 42
41	عملية حذف مستخدم 2	رسم توضيحي 43
42	اعادة ضبط النظام	رسم توضيحي 44

قائمة الجداول

رقم الصفحة	العنوان	الرسم التوضيحي
9	حالة الاستخدام : تسجيل الدخول الى البرنامج	جدول 1
10	حالة الاستخدام : البحث عن المحاضرات	جدول 2
10	حالة الاستخدام : حجز المحاضرة	جدول 3
11	حالة الاستخدام : معاينة الوصل	جدول 4
11	حالة الاستخدام : اضافة طالب	جدول 5
12	حالة الاستخدام : تعديل بيانات طالب	جدول 6
12	حالة الاستخدام : حذف طالب	جدول 7
13	حالة الاستخدام : اضافة / حذف محاضرة	جدول 8
13	حالة الاستخدام : اضافة نسخ محاضرة	جدول 9
14	حالة الاستخدام : اضافة / حذف مادة	جدول 10
14	حالة الاستخدام : اضافة/حذف كلية	جدول 11
15	حالة الاستخدام : تأكيد تسجيل الدخول كمدير	جدول 12
15	حالة الاستخدام : جرد يومي	جدول 13
16	حالة الاستخدام : تعديل الاسعار	جدول 14
16	حالة الاستخدام : اعادة ضبط النظام	جدول 15
17	حالة الاستخدام : اضافة مستخدم	جدول 16
17	حالة الاستخدام : تعديل بيانات مستخدم	جدول 17
18	حالة الاستخدام : تغيير كلمة مرور المدير	جدول 18
18	حالة الاستخدام : ملاحظات النظام	جدول 19
19	حالة الاستخدام : تنبيهات النظام	جدول 20

المقدمة

من خلال دراستنا في جامعتنا السورية الخاصة , تركزت الدراسة فيها باعتماد السادة المحاضرين على محاضرات ورقية وهنا يجدر الإشارة أنه في بعض المواد تكون المحاضرات حصرا ورقية نظرا لطبيعة المادة ، أو تكون المحاضرة الكترونية باختلاف موادها وكلياتها.

يقوم المحاضرون بوضع المحاضرة في المكتبة ليتم نسخها و بيعها للطلاب بحيث تكون المرجع لهم في دراستهم ، أما النسخ الالكترونية فتوزع بينهم بمعرفتهم ، أو عن طريق رفعها الى شبكة الانترنت على شبكات التواصل الاجتماعي .

وحيث أن الدوام في الجامعة هو دوام مجتزأ ومحدود ضمن ساعات قليلة ، هنا كانت تعترضنا بعض الصعوبات التي عانينا منها ورأيناها لدى أغلب الطلاب .

فقد لاحظنا عدم التنسيق فيما بينهم نظرا لعدم معرفتهم الشخصية ببعضهم باعتبار ان المادة يمكن ان تكون لطلاب مختلفي المراحل او عدم توفر الانترنت لدى بعض الطلاب مما كان يصعب الحصول على النسخة الالكترونية للمحاضرة .

كما أن الانقطاع المتكرر للتيار الكهربائي كان سبباً رئيسياً لتوجه الطلاب الى طباعة كافة محاضراتهم تحسبا لهذه المشكلة وخاصة في اوقات الامتحانات لعدم قدرتهم على الدراسة على حواسيبهم الشخصية.

وقد قامت الجامعة السورية الخاصة مشكورة باحداث مركز لخدمات الطباعة في موقعها المؤقت الذي يجمع كافة الكليات .

وبعد مرور أكثر من عام على احداث هذا المركز ، بدأت تظهر بعض المشاكل التي نذكر منها :

صعوبة تنسيق المحاضرات الورقية نظرا لكثرة الكليات وكثرة المواد .

إن عدم معرفة مركز الطباعة بأعداد الطلاب في كل مادة كان يضعهم في مشكلة طباعة كميات معينة من المحاضرات بأكثر من عدد الطلاب المسجلين في هذه المادة مما يؤدي الى خسارة مادية .

الازدحام الكبير الحاصل في اوقات الامتحانات ايضا بسبب اتجاه كافة طلاب الجامعة الى مركز التصوير والذي يقارب عددهم تقريبا 2000 طالب .

الازدحام الناتج عن وقت الدوام المجتزأ الذي يكون في أيام الاسبوع من 3 الى 4 ساعات فقط .

كما وإن كثرة المواد وكثرة المحاضرات للطلاب وقلة التواصل المباشر مع مركز الطباعة كان يعرضهم لنسيان المحاضرات المأخوذة والمحاضرات الغير المأخوذة.

الفصل الأول

1. تلخيص الأفكار و الاحتياجات

1.1- المشكلة المراد حلها:

تكمّن في صعوبة ادارة المحاضرات المطبوعة الخاصة بمقررات الكليات في الجامعة ، و الازدحام المستمر في المكتبة ، بالاضافة الى عدم المام الطلاب بشكل جيد بالمحاضرات المتوجب عليهم الحصول عليها.

1.2- الجهود المطبقة في هذا المجال:

بعد قيامنا باستعراض المنتجات البرمجية التي يمكن ان تحل المشكلة ، وجدنا العديد من البرمجيات الشبيهة نوعا ما بالنظام الذي تخيلناه ، و كان الأقرب من متطلباتنا النظام التالي :

Copitrack Print Room على الرابط التالي :

<http://www.xenith.co.uk/DocumentSolutions/Copitrak/CopitrakPrintRoom/>



رسم توضيحي 2

و هنا نرفق شرح مختصر عن البرنامج "مترجم الى العربية" كما هو وارد على الرابط في الاعلى :

"أتمتة كاملة لتتبع وإدارة العمل في مركز النسخ الخاص بك :

غرفة الطباعة هو تطبيق إدارة مركز النسخ الشامل الذي يتيح لمديري مركز النسخ لتتبع وإدارة وفوترة كل وظائف النسخ بسهولة.

يربط Copitrak مركز الطباعة مع ما يسمى (Job Submission module) للسماح للمستخدمين بارسال طلبات الطباعة تلقائيا إلى مركز نسخ.

يتيح النظام للمدير المتجر رصد ومراقبة اوامر الطباعة المقدمة من قبل المستخدمين بسهولة. النظام

يسمح أيضا لمدير المتجر اسقاط اوامر النسخ إلى الجهاز المحدد من تطبيق طباعة الغرف.."

و بعد اطلعنا على هذا النظام بشكل دقيق تبين انه معقد و كثير التفاصيل و لا يلائم احتياجاتنا .

حيث انه لا يتيح تخزين بيانات الطلاب بينما يسمح لهم بالدخول على شبكة محلية و ارسال اوامر الطباعة خاصتهم بشكل مباشر الى مركز الطباعة و هذه الطريقة لا تحقق مبتغانا ، فالمحاضرات يتم توزيعها من مركز الطباعة على شكل ورقي ، و الطلاب ليس لديهم نسخ الكترونية فيها ، بالاضافة الى

العبي الاضافي الذي يلقيه هذا النظام على عاتق الطلاب حيث أن كل طالب مسؤول عن جمع محاضراته بشكل الكتروني و ارسال اوامر نسخ بكلّ منها ، و هذا لا يلائم حاجتنا بالطبع .
حيث أننا بحاجة الى نظام يدير بيانات الطلاب و يقوم بتقسيم المحاضرات الموجودة و توزيعها على الطلاب حسب اختصاصاتهم و المواد المسجلة ، و الطالب هنا مهمته فقط اعطاء ال(ID) الخاص به لمستخدم النظام ، فيقوم النظام بجمع كل ما يلزمه من المحاضرات بشكل تلقائي و يطبع وصل فيها ، ليقوم بجمعها موظف اخر في المكتبة من الاماكن المخصصة لها .
بالاضافة الى برمجيات اخرى

ولكن لم نستطيع الحصول على المصدر(Source Code) الخاص بأيّ منها ، بالاضافة لأنها لا تدعم بالتحديد العمليات و الخصائص المطلوبة وفق طريقة العمل في الجامعات في سوريا . ولم نجد منها ما يدعم اللغة العربية.
وهنا بدأت فكرة أتمتة مركز الخدمات الطلابية بتطوير نظام واحد متكامل لحل هذه المشاكل من تنظيم عمل المركز وتنسيق عمليات الطباعة والبيع فيه .
وتخفيف عبئ الانتظار على الطلاب وتنسيق محاضراتهم بطريقة مؤتمتة تحفظ لهم المحاضرات المأخوذة والغير مأخوذة .

1.3- أهمية استخدام هذا النظام:

تكمّن أهمية استخدام هذا النظام ، في تقليص الازدحام المستمر في المكتبة وذلك من خلال تسهيل وتسريع عملية البيع ، و ضمان وصول المحاضرات المطلوبة الى كل الطلاب بشكل تلقائي ، مما يضمن لهم الحصول على المقررات بشكل كامل ، فمع كثرة المواد و صعوبتها و كثافة المحاضرات يصعب على الطالب عند دخوله المكتبة أن يتذكر المحاضرات المطلوبة منه بشكل كامل و غالبا ما يقع في الخطأ أو النسيان ، اما باستخدام هذا النظام سوف نسجل المحاضرات و ننسبها للمواد ، و بالتالي عند ادخال ID الطالب سوف ينفذ النظام استعلام بسيط يجلب المحاضرات الواجب على الطالب الحصول عليها بشكل دقيق و آني .

1.4- الاستبيان:

وللتأكد من حاجتنا لهذا النظام في الجامعة ، قمنا باعداد ورقة استبيان بسيط حول مكتبة الخدمات الطلابية و توزيعها على شريحة كافية من الطلاب ، و كانت ورقة الاستبيان على الشكل التالي :

زميلنا المحترم / زميلتنا المحترمة ..

تحية طيبة ..

نحن مجموعة من طلاب كلية هندسة الحاسوب و المعلوماتية ، نعمل على جمع معلومات حول امكانية تطوير نظام اتمتة خالص بمكتبة المطبوعات لدينا في الجامعة. نطلب مساعدتك في الاجابة عن الاسئلة التالية شاكرين لك تعاونك .
"ضع دائرة حول الاجابة التي تختارها".

I. ما رأيك في جودة الخدمة في مكتبة المطبوعات في الجامعة ؟

1-جيدة جداً. 2-جيدة. 3-لا بأس بها. 4-رديئة.

II. كم من الوقت تنتظر غالبا لتحصل على ما يلزمك من المحاضرات ؟

1-أقل من 5 دقائق. 2-من 5 الى 10 دقائق. 3-أكثر من 10 دقائق.

III. هل تعاني عند ذهابك الى المكتبة من مشكلة في تذكر ما يلزمك من المحاضرات ؟

1-لا أبدا. 2-أحيانا. 3-غالباً.

IV. هل ترى انه من الضروري استخدام نظام حاسوبي يدير عملية بيع المحاضرات ضمن المكتبة ؟

1-نعم بالتأكيد. 2-لا أدري. 3-لا.

V. ماذا تتوقع ان يقدم لك هذا النظام ؟ (سؤال اختياري) .

.....
.....
.....
.....

رسم توضيحي 3

وكانت نتيجة الاستبيان على الشكل التالي :

I. ما رأيك في جودة الخدمة في مكتبة المطبوعات في الجامعة ؟

1-جيدة جداً. 2-جيدة. 3-لا بأس بها. 4-رديئة.

3% 22% 30% 35%

I. كم من الوقت تنتظر غالبا لتحصل على ما يلزمك من المحاضرات ؟

1-أقل من 5 دقائق. 2-من 5 الى 10 دقائق. 3-أكثر من 10 دقائق.

12% 49% 39%

ا. هل تعاني عند ذهابك الى المكتبة من مشكلة في تذكر ما يلزمك من المحاضرات ؟

1-لا أبدا. 2-أحيانا. 3-غالباً.

11% 48% 41%

ا. هل ترى انه من الضروري استخدام نظام حاسوبي يدير عملية بيع المحاضرات ضمن المكتبة ؟

1-نعم بالتأكيد. 2-لا أدري. 3-لا.

76% 16% 8%

و من الواضح أن النتائج مشجعة فعلاً ، و تدل على حاجة واضحة لاستخدام نظام حاسوبي لإدارة أعمال المكتبة .

1.5- صعوبة تطوير مثل هذا النظام.

تكمن صعوبة تطوير مثل هذا النظام في تصميم قاعدة البيانات المثالية التي تشمل العلاقات الصحيحة ما بين الاجزاء الرئيسية و هي (الطلاب ، المواد ، المحاضرات).

بالاضافة الى تصميم واجهات تدعم اللغة العربية و تكون سهلة الاستخدام و واضحة .

كما يتوجب دعم النظام بادوات تسهل ادارة المكتبة مثل ادوات جرد و تغيير الاسعار و الى ما هنالك .

1.6- النتيجة المفترض تحقيقها:

النتيجة المفترض الوصول اليها هي (نظام لإدارة و بيع محاضرات الطلاب) ، سهل الاستخدام و سريع يقلل احتمالية الوقوع في الخطأ الى الحدود الدنيا ، ويحقق المتطلبات التي قمنا بجمعها والمبينة في الفقرة التالية.

1.7- متطلبات نظام أتمتة أعمال مكتبة الخدمات الطلابية في الجامعة :

- 1- إضافة وتخزين الكليات .
- 2- إضافة و تخزين المواد لكل كلية .
- 3- إضافة وتخزين بيانات الطلاب .
- 4- تخزين المواد المسجلة للطلاب .
- 5- اظهار عدد الطلاب في كل مادة .
- 6- إضافة و تخزين المحاضرات لكل مادة .
- 7- البحث عن المحاضرات المطلوبة لكل طالب عن طريق ادخال الرقم الجامعي ID .
- 8- بيان المحاضرات المأخوذة والغير مأخوذة من قبل الطالب .
- 9- حجز المحاضرات المطلوب شراؤها وعدد نسخ كل منها.
- 10- طباعة وصل بالمشتريات .
- 11- اظهار رسائل للمستخدم بنقص عدد نسخ محاضرة معينة .
- 12- جرد للمبيعات اليومية .

الفصل الثاني

2. الأدوات المستخدمة

2.1- المفاهيم التي يمكن الاستفادة منها:

في الحقيقة النظام الذي نود بناؤه ، يمكن ان يعتبر نوعا ما مزجا لنظامين شائعين و هما :

نظام ادارة مبيعات	نظام ادارة مكتبة
Sales Management System	Library Management System

كوننا نريد تصميم نظام لإدارة مطبوعات في مكتبة (Library Management System) ،
بالإضافة الى عملية بيع هذه المطبوعات (Sales Management System) .

2.2- لغة البرمجة المستخدمة:

C# وهي لغة بسيطة و حديثة ، غرضية التوجه و تعتبر من لغات البرمجة الآمنة (type-safe programming).
لهذه اللغة جذور في عائلة C و بالتالي فهي مألوفة ل C ، C++ ، و ايضا مألوفة لمبرمجي الJava
C# محددة المعايير من قبل ECMA الدولية، المعيار ECMA-334 المحدد من قبل ISO/IEC
مترجم هذه اللغة (Compiler) التابع ل(.NET Framework) هو تطبيق محقق لهذه المعايير[1]
مجالات تطبيقها :
- تطبيقات الوندوز.
- تطبيقات الانترنت.
- تطبيقات الموبايل .
- تطبيقات العمل على لينوكس و ذلك باستخدام مكتبات مخصصة لنظام لينوكس.
- تطبيقات التعامل مع قواعد البيانات.
- تطبيقات الوسائط المتعددة و الالعاب.

2.3- بيئة قواعد البيانات المستخدمة :

SQL Server 2008: يتضمن أدوات متقدمة رسومية و ميزات قوية لإعداد التقارير و عمليات البحث
المتقدمة تستند الى نص ، توفر هذه النسخة أدوات قوية و موثوق بها ، و ادارة و حماية للبيانات ، بالإضافة
الى سرعة الأداء.
SQL Server مثالي لتطبيقات الخادم الصغيرة و مخازن البيانات المحلية.

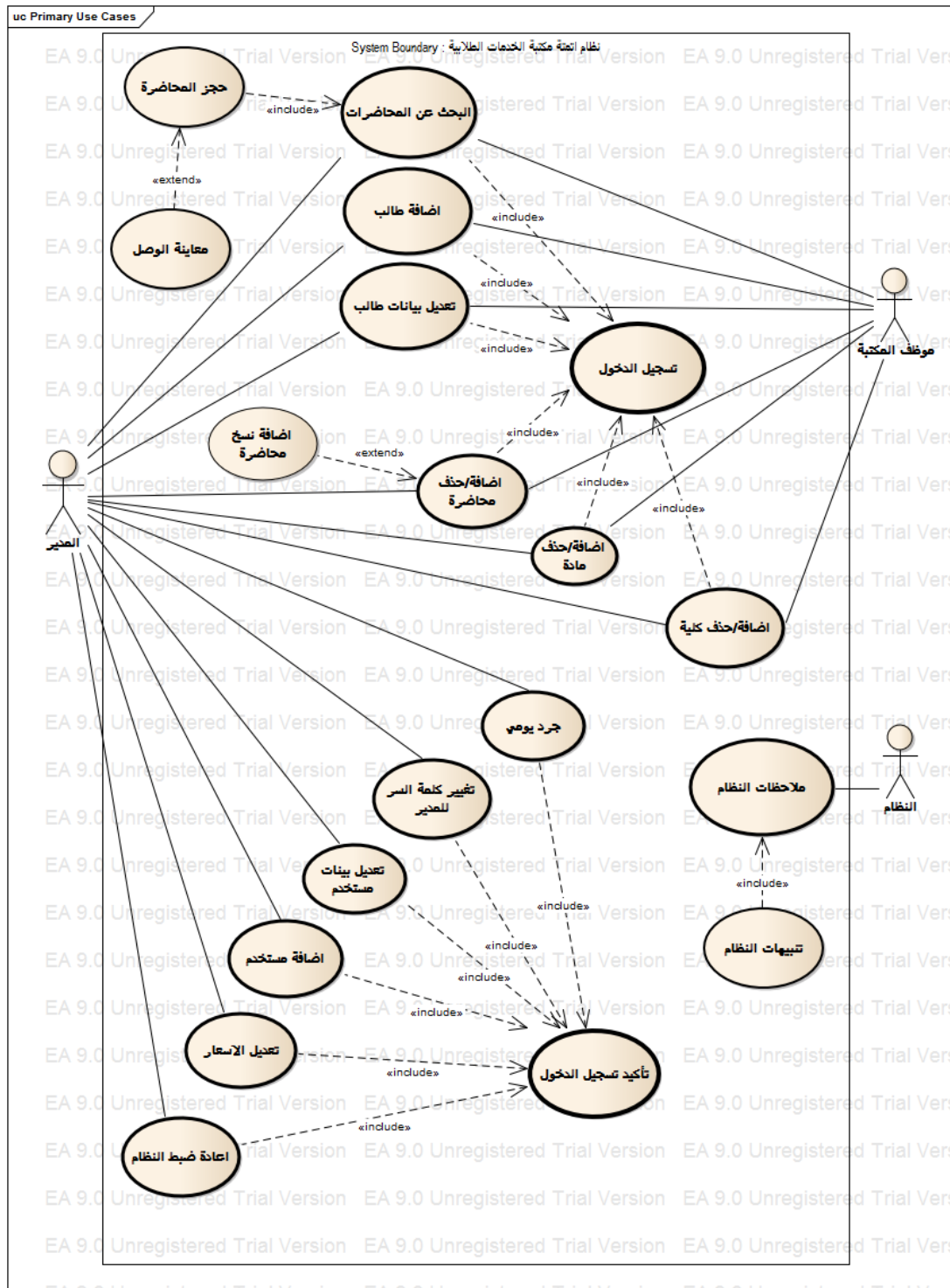
2.4- برنامج إعداد الدراسة التحليلية :

Enterprise Architect version 9.0 ، برنامج لإنشاء مخططات ال UML و غيرها.

الفصل الثالث

3. الدراسة التحليلية

3.1- مخطط حالات الاستخدام (Use Case Diagram): [2]



رسم توضيحي 4

3.2- توصيف حالات الاستخدام : (Use Cases Description):

جدول 1

ID:		01
Name:		تسجيل الدخول الى البرنامج
Description:		تتيح للمدير وموظف المكتبة تسجيل الدخول الى البرنامج
Actors:	Primary:	المدير
	Secondary:	الموظف
Pre-Condition:		تشغيل البرنامج
Main Scenario		يقوم المدير او الموظف بادخال اسم المستخدم وكلمة المرور الخاصة به
Exceptions:		عند ادخال اسم المستخدم او كلمة المرور بشكل خاطئ تظهر رسالة خطأ ولا يتم الدخول الى البرنامج

جدول 2

ID:		011
Name:		البحث عن المحاضرات
Description:		تتيح للمدير وموظف المكتبة البحث عن محاضرات للطلاب .

Actors:	Primary:	المدير
	Secondary:	الموظف والوقت
Pre-Condition:		ادخال رقم الطالب المراد البحث عن محاضرات له .
Main Scenario:		<p>1- ادخال رقم الطالب :</p> <p>عند ادخال الرقم يقوم النظام بجلب اسم الطالب ومرحلته ومواده المسجلة في النظام.</p> <p>2- عند تحديد احد المواد يظهر قائمة باسماء المحاضرات المأخوذة والغير مأخوذة من قبل الطالب.</p>
Exceptions:		عند ادخال رقم غير موجود تظهر رسالة خطأ بأن الرقم غير موجود.

جدول 3

ID:		0111
Name:		حجز المحاضرة
Description:		تتيح للمدير وموظف المكتبة حجز المحاضرة للطالب.
Actors:	Primary:	المدير.
	Secondary:	الموظف.
Pre-Condition:		القيام بالبحث عن محاضرات الطالب.
Main Scenario:		<p>1-يقوم المدير او الموظف بحجز محاضرة معينة للطالب وذلك بالنقر المزدوج على اسمها، فتظهر في حقل المحاضرات المحجوزة.</p> <p>2-يمكن تعديل عدد النسخ المطلوبة من كل محاضرة محجوزة.</p> <p>3-يمكن التراجع عن حجز محاضرة معينة.</p>
Exceptions:		عند ادخال عدد النسخ المطلوبة وكان هذا العدد اكبر من عدد النسخ الموجودة تظهر رسالة خطأ بأنه لايمكن حجز هذا العدد من النسخ .

جدول 4

ID:		01111
Name:		معاينة الوصل
Description:		تتيح للمدير وموظف المكتبة معاينة وصل بالمحاضرات المحجوزة ،

		وطباعته.
Actors:	Primary:	المدير.
	Secondary:	الموظف.
Pre-Condition:		حجز محاضرة واحدة على الأقل لطالب معين.
Main Scenario:		عند قيام المدير أو الموظف باختيار معانية الوصل ، تظهر واجهة تحوي وصل بالمحاضرات المحجوزة بالإضافة لعدد نسخ كل منها و سعرها ، و المبلغ الكلي.
Exceptions:		

جدول 5

ID:		012
Name:		إضافة طالب
Description:		تتيح للمدير وموظف المكتبة اضافة طالب جديد على قاعدة البيانات
Actors:	Primary:	المدير.
	Secondary:	الموظف.
Pre-Condition:		ادخال بيانات الطالب بشكل كامل.
Main Scenario:		يقوم المدير او الموظف بادخال بيانات الطالب : الرقم والاسم والكلية والاختصاص والمرحلة والفصل. ثم يختار (إضافة طالب).
Exceptions:		عند ادخال رقم موجود مسبقا تظهر رسالة خطأ بأن الرقم موجود ولا يمكن قبول هذا الرقم. عند ترك أحد الحقول فارغة تظهر رسالة خطأ ليقوم المستخدم بملئ تلك الحقول .

جدول 6

ID:		013
Name:		تعديل بيانات طالب

Description:		تتيح للمدير وموظف المكتبة تعديل بيانات الطالب، والمواد المسجلة.
Actors:	Primary:	المدير.
	Secondary:	الموظف.
Pre-Condition:		ادخال رقم الطالب المراد تعديل بياناته.
Main Scenario:		<p>1- يقوم المدير او الموظف بادخال رقم الطالب المراد تعديل بياناته.</p> <p>2- تظهر البيانات الحالية ، مع امكانية تعديل كل مهنا.</p> <p>في حال الرغبة في تعديل احداها يقوم مدير النظام او الموظف بتغييرها.</p> <p>3- عند اختيار (تعديل بيانات الطالب) ، يحدث النظام حقول البيانات ، ويتيح امكانية تعديل المواد المسجلة.</p> <p>4- يقوم المدير أو الموظف بتعديل المواد المسجلة في حال الرغبة في ذلك.</p>
Exceptions:		عند اضافة مادة مسجلة مسبقا للطالب تظهر رسالة خطأ بأن هذه المادة مسجلة لا يمكن اضافتها مرة اخرى.

جنول 7

ID:		014
Name:		حذف طالب
Description:		تتيح للمدير وموظف المكتبة حذف طالب من قاعدة البيانات.
Actors:	Primary:	المدير.
	Secondary:	الموظف.
Pre-Condition:		ادخال رقم الطالب المراد حذفه.
Main Scenario:		<p>يقوم المدير او الموظف بادخال رقم الطالب المراد حذفه ثم اختيار حذف الطالب ، تظهر رسالة تأكيد الحذف باختيار نعم يتم حذف الطالب ، و باختيار لا يتم التراجع عن العملية .</p>
Exceptions:		ادخال ID غير موجود.

جنول 8

ID:		015
Name:		اضافة / حذف محاضرة

Description:		تتيح للمدير وموظف المكتبة اضافة محاضرة جديدة او حذفها من قاعدة البيانات.
Actors:	Primary:	المدير.
	Secondary:	الموظف.
Pre-Condition:		تحديد الكلية والمادة المراد اضافة محاضرة لها. في حالة الاضافة تحديد الكلية و المادة و المحاضرة الموجودة ، في حالة الحذف.
Main Scenario:		يقوم المدير او الموظف بادخال اسم المحاضرة بعد تحديد الكلية والمادة وعدد اوراق المحاضرة وعدد النسخ في حالة الاضافة او يحدد المحاضرة المراد حذفها من قاعدة البيانات في حالة الحذف.
Exceptions:		عند ترك أحد الحقول فارغة تظهر رسالة خطأ بأن أحد الحقول فارغة و يجب ملئ تلك الحقول.

جدول 9

ID:		0151
Name:		اضافة نسخ محاضرة
Description:		تتيح للمدير وموظف المكتبة اضافة نسخ لمحاضرة معينة.
Actors:	Primary:	المدير.
	Secondary:	الموظف.
Pre-Condition:		تحديد المحاضرة المراد اضافة نسخ منها.
Main Scenario:		يقوم المدير او الموظف بتحديد الكلية والمادة واسم المحاضرة المراد اضافة النسخ اليها ، فيظهر العدد المتبقي من النسخ بالاضافة الى عدد الطلاب في المادة. ثم يقوم بادخال عدد النسخ المراد اضافتها، و يختار تعديل.
Exceptions:		عند ترك أحد الحقول فارغة تظهر رسالة خطأ بأن أحد الحقول فارغة و يجب ملئ تلك الحقول

جدول 10

ID:		016
Name:		اضافة/حذف مادة

Description:		تتيح للمدير وموظف المكتبة اضافة مادة جديدة أو حذف مادة من قاعدة البيانات.
Actors:	Primary:	المدير.
	Secondary:	الموظف.
Main Scenario:		يقوم المدير او الموظف بعد تحديد الكلية بادخال اسم المادة والقسم ورقم المادة والمرحلة والفصل واسم المحاضر.في حالة الاضافة. وفي حالة الحذف يكفي اختيار الكلية و مادة معينة واختيار حذف.
Exceptions:		عند ادخال رقم للمادة موجود مسبقا تظهر رسالة خطأ بأن الرقم موجود ولا يمكن اعطاء هذا الرقم لمادة جديدة. عند ترك أحد الحقول فارغة تظهر رسالة خطأ بأن أحد الحقول فارغة و يجب ملئ تلك الحقول.

جدول 11

ID:		017
Name:		اضافة/حذف كلية
Description:		تتيح للمدير وموظف المكتبة اضافة كلية جديدة او حذف كلية من قاعدة البيانات.
Actors:	Primary:	المدير.
	Secondary:	الموظف.
Main Scenario		يقوم المدير او الموظف بادخال اسم الكلية ويقوم النظام باعطاء رقم تلقائي لها . في حالة الاضافة. ويقوم باختيار الكلية و اختيار حذف في حال الرغبة في الحذف.
Exceptions:		عند ترك حقل اسم الكلية فارغاً تظهر رسالة خطأ بأن الحقل فارغ و يجب ملؤه.

ID:		02
Name:		تأكيد تسجيل الدخول كمدير
Description:		تتيح للمدير الدخول الى خيارات النظام.
Actors:	Primary:	المدير.
	Secondary:	لا يوجد.
Main Scenario:		عند الذهاب الى واجهة خيارات النظام : تظهر نافذة تسجيل الدخول و تكون خيارات النظام غير متاحة ليقوم المدير باخال اسم المستخدم وكلمة المرور الخاصة به لكي يدخل الى الخيارات الموجودة في هذه الواجهة، وتصبح متاحة.
Exceptions:		تظهر رسالة خطأ اذا كان احد من اسم المستخدم أو كلمة المرور فارغ . تظهر رسالة خطأ اذا كان اسم المستخدم وكلمة المرور لا يعودون لمدير النظام حصرا.

ID:		021
Name:		جرد يومي
Description:		تتيح للمدير فقط الاطلاع على المبيعات في كل يوم والربح الصافي فيه.
Actors:	Primary:	المدير.
	Secondary:	لا يوجد.
Pre-Condition:		تسجيل الدخول كمدير الى خيارات النظام بشكل صحيح.
Main Scenario:		يقوم المدير بتحديد اليوم المراد معرفة المبيعات فيه فتظهر قائمة بالفواتير المباعة و تواريخ ووقت البيع واسم المستخدم الذي باع المحاضرت واسم الطالب الذي اشتراها.
Exceptions:		عند تحديد يوم معين ولا يوجد مبيعات في هذا اليوم تظهر رسالة بأن لا مبيعات في هذا اليوم.

ID:		022
Name:		تعديل الاسعار
Description:		تتيح للمدير تعديل سعر رأس مال الورقة وسعر مبيعها.
Actors:	Primary:	المدير.
	Secondary:	لا يوجد.
Pre-Condition:		تسجيل الدخول كمدير الى خيارات النظام بشكل صحيح.
Main Scenario:		يقوم المدير بتحديد خيار تعديل الاسعار لتظهر له قائمة بالاسعار القديمة سعر رأس مال الورقة الواحدة وسعر مبيعها ليقوم بتعديلها.
Exceptions:		تظهر رسالة خطأ اذا ترك أحد الحقول فارغة.

ID:		023
Name:		اعادة ضبط النظام
description		تتيح للمدير فقط اعادة ضبط النظام من أجل بداية فصل جديد.
Actors:	Primary:	المدير.
	Secondary:	لا يوجد.
Pre-Condition:		تسجيل الدخول كمدير الى خيارات النظام بشكل صحيح.
Main Scenario:		يقوم المدير بتحديد خيار اعادة ضبط النظام لتفريغ جداول العلاقات في قاعدة البيانات : جدول الربط بين الطالب ومواده. جدول الربط بين المادة ومحاضراتها. جدول الربط بين الطالب ومحاضراته. ويتم ذلك من أجل اعادة ملئها عند بداية الفصل الجديد نظرا لتغير محاضرات ومواد الطالب من فصل لآخر.
Exceptions:		تظهر رسالة تأكيد للمدير بعملية اعادة الضبط فاذا اختار الغاء الأمر لا يتم افراغ محتويات الجداول المذكورة من قاعدة البيانات.

ID:		024
Name:		إضافة مستخدم
Description:		تتيح للمدير إضافة مستخدم جديد للنظام.
Actors:	Primary:	المدير.
	Secondary:	لا يوجد.
Pre-Condition:		تسجيل الدخول كمدير الى خيارات النظام بشكل صحيح.
Main Scenario :		يقوم المدير بتحديد خيار إضافة مستخدم وذلك باختيار اسم مستخدم وكلمة سر خاصة به ، ليستطيع الدخول بهما الى البرنامج ضمن السماحيات المصرحة له كمستخدم عادي (موظف في المكتبة) و هي استخدام كافة الأدوات ما عدا خيارات النظام المتاحة للمدير حصراً.
Exceptions:		تظهر رسالة خطأ للمدير اذا كان أحد الحقول فارغة.

ID:		025
Name:		تعديل بيانات مستخدم
Description:		تتيح للمدير التعديل على بيانات المستخدم .
Actors:	Primary:	المدير .
	Secondary:	لا يوجد .
Pre-Condition:		تسجيل الدخول كمدير الى البرنامج بشكل صحيح
Main Scenario:		يقوم المدير بتحديد خيار تعديل بيانات مستخدم ثم يحدد مستخدم معين ويعدل كل من اسم المستخدم أو كلمة السر الخاصة به. كما يمكن للمدير حذف مستخدم معيا بتحديد و اختيار (حذف).

ID:	026
Name:	تغيير كلمة مرور المدير
Description:	تتيح للمدير تغيير كلمة السر الخاصة به.
Actors:	المدير.
primary	لا يوجد.
secondary	
Pre-Condition:	تسجيل الدخول كمدير الى خيارات النظام بشكل صحيح.
Main Scenario	يقوم المدير باختيار (تغيير كلمة المرور) لتظهر قائمة يقوم فيها بادخال اسم المستخدم الخاص به وكلمة المرور القديمة وكلمة المرور الجديدة وتأكيدها مرة اخرى. و من ثم يختار تحديث لكي تتم العملية.
Exceptions:	تظهر رسالة خطأ اذا لم يتطابق اسم المستخدم (المدير) المدخل مع كلمة المرور ولا تتغير كلمة المرور.

ID:	003
Name:	ملاحظات النظام
Description:	عملية مراقبة عدد النسخ لكل محاضرة.
Actors:	النظام.
Primary:	لا يوجد.
Secondary:	
Pre-Condition:	بيع نسخ من هذه المحاضرة و تخريجها من قاعدة البيانات.
Main Scenario:	يقوم النظام بمراقبة عدد النسخ المتبقية من المحاضرة ، وذلك بعد كل عملية بيع أي عملية اخراج من قاعدة البيانات.
Exceptions:	لا يوجد عملية بيع قيد الاجراء

ID:		031
Name:		تنبيهات النظام
Description:		رسائل تدل المدير الى نقص في عدد النسخ لمحاضرة معينة.
Actors:	Primary:	النظام.
	Secondary:	لا يوجد.
Pre-Condition:		تسجيل نقص في نسخ محاضرة معينة (أقل من 3 نسخ) من قبل عملية ملاحظات النظام.
Main Scenario:		عند وصول عدد نسخ المحاضرات الى أقل من ثلاثة يلاحظ ذلك النظام و يظهر رسالة تنبيه في كافة نوافذ البرنامج بأن عدد النسخ للمحاضرة التالية أصبح قليل مع ذكر العدد المتبقي .
Exceptions:		لا يظهر أي تنبيه اذا لم يصل عدد النسخ إلى أقل من 3.

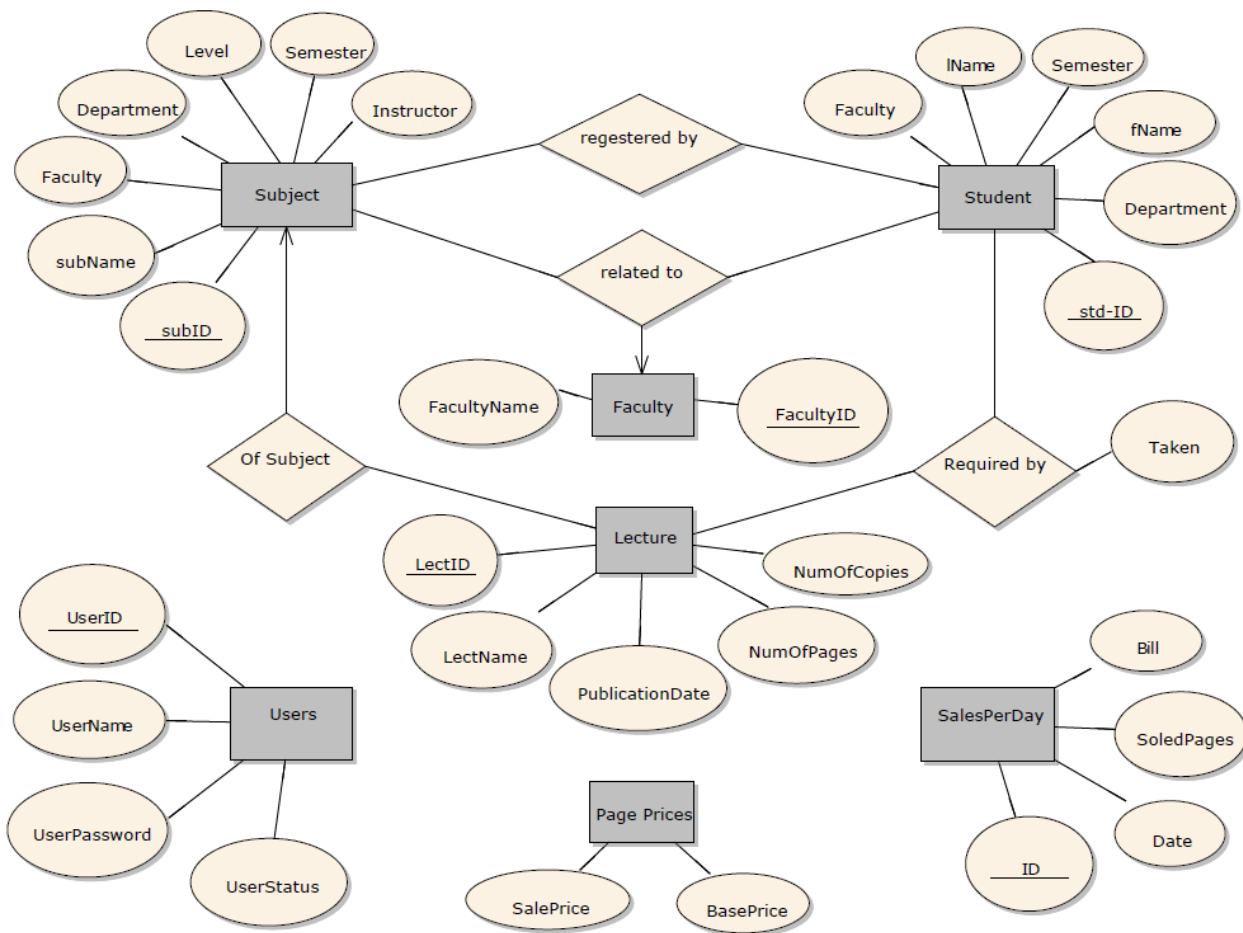
الفصل الرابع

4. تصميم قاعدة البيانات

4.1 - مخطط الكيانات و العلاقات (Entity Relationship Diagram): [3]

المكونات الأساسية لقاعدة البيانات المطلوبة : الكيانات (Entities)، والعلاقات في ما بينها (Relationships).

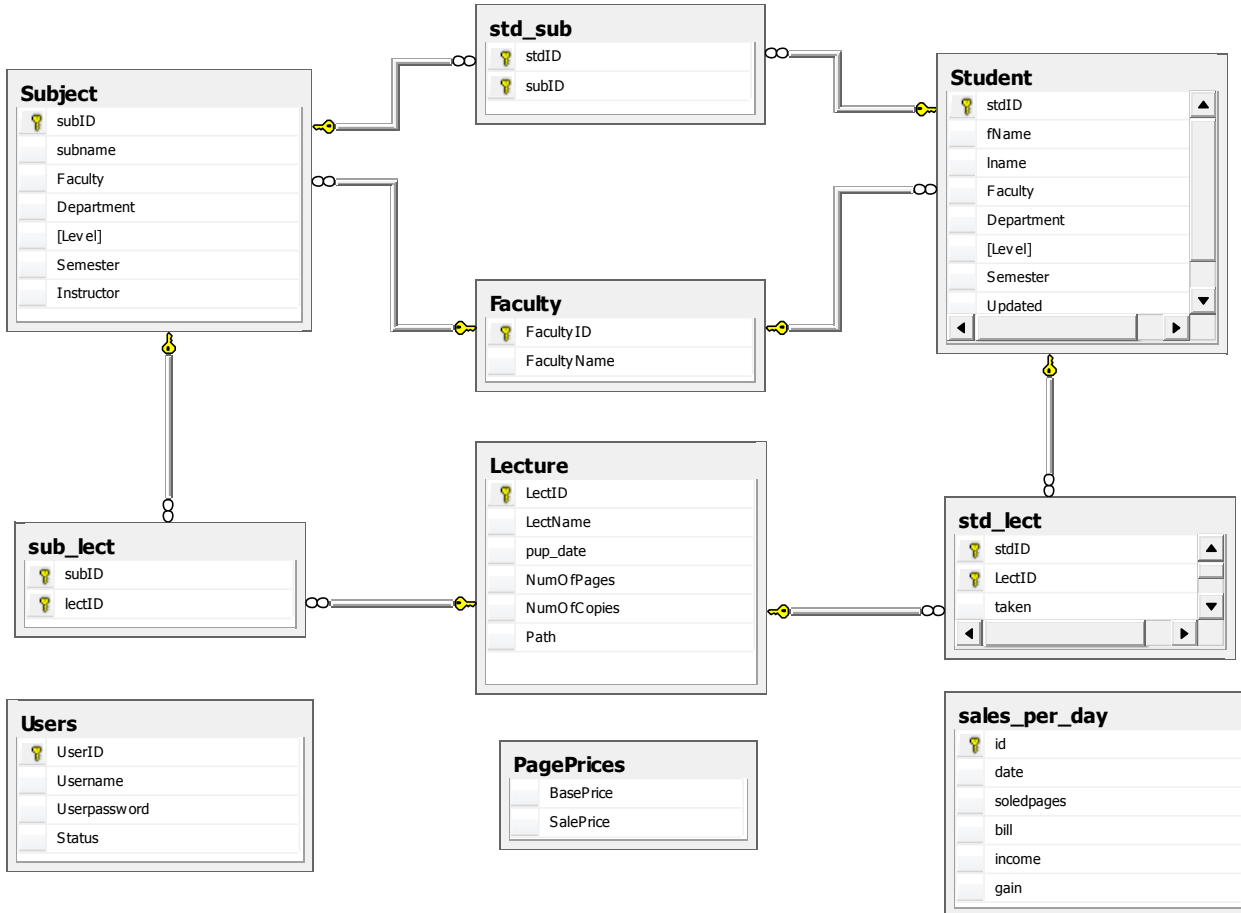
وهي موضحة أدناه في مخطط (ERD) (Entity Relationship Diagram) :



رسم توضيحي 5

4.2- بناء قاعدة البيانات:

و لتطبيق قاعدة البيانات المبينة في المخطط (Figure 1) اخترنا بيئة SQL Server 2008 ، و قمنا ببناء قاعدة البيانات الموضحة في المخطط أدناه : [4]



رسم توضيحي 6

الفصل الخامس

5. فحص النظام

5.1- الطريقة المتبعة:

بدئنا بتاريخ 2013/11/1 بعملية فحص النظام (Testing).

وقمنا بذلك من خلال افتراض اوقات دوام معينة و تجريب النظام بكافة ادواته و ميزات من اضافة طلاب و تعديل بياناتهم و المواد المسجلة ، الى اضافة المواد و المحاضرات ، وتعديل خيارات النظام (ادارة حسابات المستخدمين - تعديل الاسعار - إعادة ضبط النظام). و كانت نتائج الفحص على الشكل التالي :

5.2- الدخول الى البرنامج:

عند فتح الملف التنفيذي يظهر النظام على الشكل التالي :



رسم توضيحي 7

و تكون الواجهة الرئيسية للبرنامج غير متاحة قبل اتمام عملية تسجيل الدخول.

و عند ادخال اسم المستخدم و كلمة المرور بشكل صحيح تختفي واجهة الدخول و تصبح الواجهة الرئيسية متاحة للاستخدام . وفي حال الادخال الخاطئ يظهر النظام رسالة بالخطأ الحاصل .. كما هو موضح.



رسم توضيحي 8



بالإضافة الى ان واجهة الدخول مزودة بزر لإغلاق للنظام .
بعد اتمام عملية تسجيل الدخول تظهر واجهة البرنامج و تتكون مفعلة و هنا سوف نتحدث قليلا عن الواجهة بشكل عام .

The screenshot shows the SPU Services Library system interface. At the top, there is a navigation bar with tabs: "البحث عن الملاحظات", "إضافة/تعديل طالب", "إدارة الموجودات", and "خيارات النظام". Below this, there is a search bar with a magnifying glass icon and a dropdown menu for "البحث عن الطالب". The main area is divided into two sections: "ملاحظات النظام" (System Notes) on the left and a table of books on the right. The table has columns for "المرحلة", "الرقم", "الاسم", "المحاضرة", "تاريخ الرفع", "مأخوذ", "حجز", "العدد المتبقي", "السعر", "العدد", and "حذف". The "ملاحظات النظام" section contains two yellow boxes with text: "يوجد من محاضرة (عقيدة) 0 نسخة فقط" and "يوجد من محاضرة (تراسل 01) نسخة فقط". Below the table, there is a button labeled "معاينة الوصل". At the bottom, there is a logo for "SPU SERVICES LIBRARY".

رسم توضيحي 9

هنا استخدمنا اداة تسمى (Tab Control)[5] و التي تتيح لنا ترتيب الواجهات خلف بعضهم البعض ضمن نفس النافذة و ذلك سهل استخدام النظام بشكل كبير حيث ان فتح اكثر من نافذة لاطهار الواجهات يكون مزعج غالبا وكما نلاحظ فإننا اخترنا الواجهة الأولى (الافتراضية) -واجهة البحث عن المحاضرات - كونها الواجهة الأكثر استخداما في النظام .
و يمكننا التنقل بين الواجهات بشكل سهل كما هو موضح في الصورة اعلاه.
لنقوم باستعراض الواجهات و توضيح آلية عمل كل منها بالأمثلة :

5.3- اضافة / تعديل طالب :

The screenshot shows the "Add/Edit Student" form in the SPU Services Library system. The form is divided into two main sections: "إضافة طالب" (Add Student) on the left and "تعديل طالب" (Edit Student) on the right. The "إضافة طالب" section has a search bar and a dropdown menu for "البحث عن الطالب". The "تعديل طالب" section has a search bar and a dropdown menu for "البحث عن الطالب". The form contains several input fields for student information: "Student number", "First Name", "Last Name", "Faculty", "Department", "Semester", "Level", "الاسم الأول", "الاسم الأخير", "اللقب", "القسم", "الفصل", and "المسجلة". There are also buttons for "إضافة طالب", "تعديل طالب", and "إلغاء". At the bottom, there is a logo for "SPU SERVICES LIBRARY".

رسم توضيحي 10

كما هو موضح في الصورة قمنا باستخدام الأداة (Group Box) [6] لكل وظيفة من وظائف هذه الواجهة و لنبدأ بعملية اضافة طالب :

أضافة الطالب

Student number 4090673 رقم الطالب *

First Name مرح الإسم الأول *

Last Name الحلبي الإسم الأخير *

Faculty هندسة الحاسوب والمعلوماتية الكلية *

Department هندسة البرمجيات القسم *

Semester 1 الفصل *

Level 5 السنة *

تعديل بيانات طالب

أضف الطالب

ملاحظة

تمت اضافة الطالب بنجاح

OK

رسم توضيحي 11

و هنا قمنا باضافة بيانات طالب كما هو موضح .
و الآن سوف نقوم باضافة المواد المسجلة هنا نوم بادخال ID الطالب و ثم النقر على زر تعديل
بيانات الطالب الذي يتيح لنا اضافة المواد المسجلة :

البحث عن طالب الرقم 4090733

البحث عن طالب الاسم

أضافة طالب

تعديل المواد

معلومات الطالب

Student number 4090733 رقم الطالب *

First Name حسان الإسم الأول *

Last Name الباروكي الإسم الأخير *

Faculty هندسة الحاسوب والمعلوماتية الكلية *

Department هندسة البرمجيات القسم *

Semester 1 الفصل *

Level 5 السنة *

تعديل بيانات طالب

المادة المستوى الفصل

التسويق 2 2

النظم الخبيرة 5 1

خوارزميات البحث الذكية 5 1

معالجة الصور وتحليلها 5 1

شبكات الحاسوب 4 1

ادارة الشبكات الحاسوبية 4 1

الاتصالات النقلة واللاسكية 5 1

إنكليزي 4 2

إنكليزي 3 1

الرياضيات المتقطعة 2 1

الجبر الخطي ونظرية المثلثات 1 1

فيزياء 1 1

تحليل رياضي 1 1

تحليل رياضي 2 2

التحليل العددي 2 2


الإحصاء والاحتمالات 2 2

ادارة المشروع 3 1

أضف المواد

رسم توضيحي 12

بالنقر على زر تعديل بيانات الطالب يقوم النظام بالاستعلام عن المواد الموجودة في الخطة الدراسية ضمن كليته و يعرضها على شكل (Data Grid View) [7] مع امكانية ترتيبها حسب المادة (الترتيب الأبجدي) ، المستوى ، الفصل الدراسي.

عندما يختار مستخدم النظام المادة المسجلة و يضغط على الزر  لتظهر المادة في حقل "المواد المطلوب تسجيلها" على شكل (List Box) [8] و ذلك على الشكل التالي:

تعديل المواد

المادة	المستوى	الفصل
اللغة الفرنسية 1	1	2
اللغة الفرنسية 2	2	2
المعالجات الصغيرة	4	1
النظم الخيرة	5	1
أمن الشبكات	5	1
أمن نظم المعلومات	4	1
إنكليزي 3	3	1
إنكليزي 4	3	2
بحوث عمليات	4	2
برمجة التطبيقات الشبكية	5	1
برمجة النظم	4	1
بنيان الحواسيب	4	2
بيئات الحاسوب	3	2
تحليل رياضي 1	1	1
تحليل رياضي 2	1	2
تحليل وتصميم النظم	4	1
تأسيس المعطيات	4	1

شبكات الحاسوب
نظم الزمن الحقيقي
ضمان جودة البرمجيات
نظم المعلومات الإدارية
هندسة البرمجيات المتقدمة
النظم الخيرة

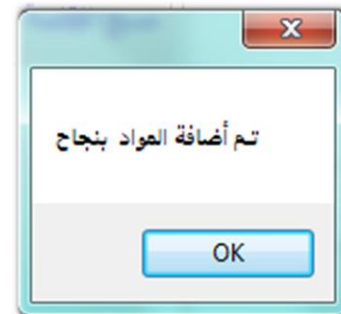
مسح القائمة

مواد المسجلة

أضف المواد

رسم توضيحي 13

مع امكانية التراجع عن اضافة احداها بالنقر على الزر  وعند التأكد من المواد المراد اضافتها يقوم المستخدم بالنقر على زر "أضف المواد" و تظهر الرسالة :



وهكذا تكون قد اكتملت بيانات الطالب حسان الباروكي و أصبحت لدينا في النظام على الشكل التالي :

تعديل المواد

+

← ×

مسح القائمة

معلومات الطالب

Student number	4090733	رقم الطالب *
First Name	حسان	الإسم الأول *
Last Name	الباروكي	الإسم الأخير *
Faculty	هندسة الحاسوب و المعلوماتية	الكلية *
Department	هندسة البرمجيات	القسم *
Semester	1	الفصل *
Level	5	السنة *



المواد المسجلة

- شيكات الحاسوب
- النظم الخبيزة
- هندسة البرمجيات المتقدمة
- نظم الزمن الحقيقي
- ضمان جودة البرمجيات
- نظم المعلومات الإدارية

رسم توضيحي 14

بالنسبة لحذف طالب :بعد ادخال ال ID والبحث عن طالب ، يتم الحذف كالتالي:

حذف الطالب



و بالنقر على

يتم الحذف و تظهر الرسالة التأكيد التالية :

هل تريد حذف الطالب؟

Yes No

و بالنقر على yes يتم الحذف و تظهر

تم حذف الطالب

OK

الرسالة :

البحث عن طالب الرقم 01090103

معلومات الطالب

Student number	01090103	رقم الطالب *
First Name	علاء	الإسم الأول *
Last Name	السلامة	الإسم الأخير *
Faculty	هندسة الحاسوب و المعلوماتية	الكلية *
Department	عام	القسم *
Semester	1	الفصل *
Level	4	السنة *





رسم توضيحي 15

كما زدونا هذه الواجهة بأداة للبحث عن طالب عن طريق اسمه الأول و يكون البحث عن طريق الاسم على الشكل التالي :

البحث عن طالب الرقم 4090733

أضف طالب

البحث عن طالب الاسم مهند

تعديل المواد

معلومات الطالب

Student number 4090724

First Name مهند

Last Name حاتم

Faculty هندسة الحاسوب والمعلوماتية

Department هندسة البرمجيات

Semester 2

Level 5

رقم الطالب *

الإسم الأول *

الإسم الأخير *

الكلية *

القسم *

الفصل *

المسنة *

المادة المستوى

هندسة البرمجيات المتقدمة 5

نظم التشغيل الموزعة 5

نظم الزمن الحقيقي 5

ضمان جودة البرمجيات 5

تطبيقات انترنت 5

نظم استرجاع البيانات 5

أمن الشبكات 5

البرمجة التفرعية 5

برمجة التطبيقات الشبكية 5

نظم المعلومات الإدارية 5

مهارات الحاسوب 1

اللغة العربية 1

مهارات التواصل 1

مقدمة في علم البيئة 1

حوار الحضارات 1

اللغة الفرنسية 1

مداخل الى القاعة 3

مواد المسجلة

النظم الخبيرة

شبكات الحاسوب

هندسة البرمجيات المتقدمة


ضمان جودة البرمجيات

نظم الزمن الحقيقي

نظم التشغيل

أضف المواد

رسم توضيحي 16

وهنا عالجنا مشكلة اضافة المادة مرتين فعند اختيار مادة موجودة مسبقا والنقر على زر  يظهر النظام رسالة بأن المادة موجودة مسبقا و لا تتم اضافتها .

تعديل المواد

المادة المستوى الفصل

التسويق 2 2

النظم الخبيرة 1 5

خوارزميات البحث الذكية 1 5

معالجة الصور وتحليلها 1 5

شبكات الحاسوب 1 4

ادارة الشبكات الحاسوبية 1 4

الاتصالات النقاله واللاسكية 1 5

الاحتمالات 2 3

نقطة 1 3

نظرية المعمل... 1 2

1 1

1 1

1 1

2 1

2 2

2 2

2 2

ادارة المشاهير 1 3

مواد المسجلة

شبكات الحاسوب

نظم الزمن الحقيقي

ضمان جودة البرمجيات

نظم المعلومات الإدارية

هندسة البرمجيات المتقدمة

النظم الخبيرة

أضف المواد

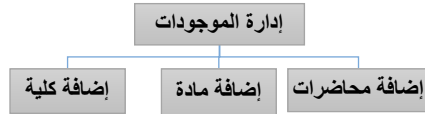
هذه المادة مسجلة بالفعل

OK

رسم توضيحي 17

كما هو موضح جانبا عندما حاول المستخدم اضافة مادة شبكات الحاسوب لم تتم اضافتها و اظهر النظام رسالة بأن المادة مسجلة بالفعل

5.4- واجهة إدارة الموجودات :



رسم توضيحي 18

بالنقر على كل من الأزرار التالية

يظهر ضمن الواجهة ال (Group Box) الخاص بها
ولنبدأ باستعراض إضافة كلية :

رسم توضيحي 19

عند النقر على زر اضافة بدون ملئ حقل اسم الكلية يظهر النظام رسالة بأن حقل اسم الكلية فارغ.

اضافة كلية

اسم الكلية

اضافة

الرجاء ادخال اسم الكلية

OK

رسم توضيحي 20

لنقم بإضافة كلية إدارة الأعمال :

اضافة كلية

اسم الكلية

إدارة الأعمال

اضافة

تم أضافة الكلية بنجاح

OK

رسم توضيحي 21

وبالنسبة الى اضافة مادة فتتم العملية على الشكل التالي :

اضافة مواد

الكلية *

رقم المادة *

اسم المادة *

القسم

المستوى

الفصل

اسم المحاضر

هندسة الحاسوب و المعلوماتية

هندسة الحاسوب و المعلوماتية

الطب البشري

طب الأسنان

الصيدلة

هندسة البترول

إدارة الأعمال

أضف

رسم توضيحي 22

نقوم باختيار الكلية المطلوب اضافة مادة الى الخطة الدراسية الخاصة بها .. و ثم نملأ البيانات الخاصة بهذه المادة و من ثمّ النقر على زر إضافة على الشكل التالي :

اضافة مواد

الكلية *	هندسة الحاسوب و المعلوماتية
رقم المادة *	URBE202
اسم المادة *	منهج البحث لعلمي
القسم	هندسة البرمجيات
المستوى	1
الفصل	2
اسم المحاضر	

أضف

تم إضافة العادة بنجاح

OK

رسم توضيحي 23

وفي حال حاول المستخدم اضافة مادة اخرى بنفس الرقم فتظهر رسالة الخطأ بأن رقم المادة المدخل محجوز لمادة اخرى موجودة .. كما هو مبين في الصورة التالية :

اضافة مواد

الكلية *	هندسة الحاسوب و المعلوماتية
رقم المادة *	URBE202
اسم المادة *	نظم التشغيل الموزعة
القسم	هندسة البرمجيات
المستوى	5
الفصل	1
اسم المحاضر	

أضف

خطأ بالإدخال

رقم العادة موجود مسبقاً

OK

رسم توضيحي 24

و أخيرا واجهة اضافة المحاضرات نستخدمها اما لاضافة محاضرة جديدة أو لتعديل عدد نسخ محاضرة معينة.
لنبدأ باضافة محاضرة ولتكن خاصة بمادة شبكات الحاسوب و اسمها شبكات 01

اضافة محاضرات

الكلية	هندسة الحاسوب و المعلوماتية	أضافة محاضره
المادة	شبكات الحاسوب	اضافة نسخ لمحاضره
اسم المحاضرة	شبكات 01	عدد الطلاب في هذه المادة 4
عدد الصفحات	20	أضف
عدد النسخ	6	
اضافة محاضرة أخرى		

رسم توضيحي 25

فور تعيين كلية و مادة معينة يعرض النظام عدد الطلاب المسجلين لهذه المادة كما هو مبين في الدائرة الحمراء ، مما يعطي المستخدم (مدير أو مستخدم عادي) فكرة عن عدد النسخ الذي يتوجب عليه طباعته من هذه المحاضرة –هنا اخترنا عدد نسخ 6 باعتبار انه لا بد من وجود طلاب مسجلين للمادة و لكنهم لم يزوروا المكتبة لهذا الفصل بعد و بالتالي فهم غير مسجلين لهذه المادة في نظام المكتبة-.



وبالنقر على زر أضف تتم اضافة المحاضرة. وتظهر الرسالة المبينة جانبا.

5.5- البحث عن المحاضرات :

تتضمن هذه الواجهة كما هو واضح في الصورة حقل لإدخال رقم الطالب (ID) ، وقمنا بمعالجة هذا الحقل لكي يقبل الأرقام فقط و هنا عند ادخال ID الطالب ، يجلب النظام المواد المسجلة من قبل الطالب و يعرض المحاضرات الصادرة في كل منها و هنا تحجز المحاضرات المراد بيعها و تتم معاينة الوصل و سنعرض هذه الخطوات عن طريق مثال:
لنفترض ان طالبة مرح الحلبي زارت المكتبة ، سيقوم مدير المكتبة هنا بإدخال الID الخاص بها والنقر على زر البحث لنحصل على النتيجة التالية :

الطالب

الاسم
مرح الحلبي

الرقم
4090673

المرحلة
5

المحاضرة	تاريخ الرفع	مأخوذ	حجز	العدد المتبقي	المحاضرة	السعر	العدد	حذف
مواد الطالب								
النظم الخبيرة								
شيكات الحاسوب								
نظم التشغيل								
هندسة البرمجيات المتقدمة								
نظم الزمن الحقيقي								

معاينة الوصل

رسم توضيحي 26

بالنقر على كل مادة من المواد الظهارة في حقل مواد الطالب تظهر المحاضرات الخاصة بهذه المادة على الشكل :

الطالب

الاسم
مرح الحلبي

الرقم
4090673

المرحلة
5

المحاضرة	تاريخ الرفع	مأخوذ	حجز	العدد المتبقي	المحاضرة	السعر	العدد	حذف
مقدمة في النظم...	AM 11:24:29 11/4/2013	✓	✓	4				
النظم الخبيرة01	AM 11:24:45 11/5/2013	✓	✓	3				
النظم الخبيرة02	AM 11:26:27 11/5/2013	✓	✓	5				
النظم الخبيرة03	AM 11:26:52 11/6/2013	✗	✗	10				
النظم الخبيرة04	AM 11:27:13 11/8/2013	✗	✗	13				

معاينة الوصل


رسم توضيحي 27

هنا تم النقر على مادة النظم الخبيرة و كما هو واضح فقد تم عرض المحاضرات الخاصة بهذه المادة و هنا نلاحظ كل محاضرة مع تاريخ صدور ها و حقل (مأخوذ) الذي يتضمن اشارة ✓ اذا كانت المحاضرة مأخوذة مسبقا من قبل الطالب و اشارة ✗ في حال كانت المحاضرة جديدة و لم يحصل عليها الطالب بعد.

الخطوة التالية تكون عملية حجز المحاضرات المراد الحصول عليها ، ويتم ذلك بالنقر المزدوج على المحاضرة لتظهر في الجدول الى اليسار و الذي يتضمن المحاضرات المحجوزة و عدد النسخ المطلوبة و السعر مع امكانية تعديل عدد النسخ المطلوبة و امكانية حذف احداها.

بعد حجز مجموعة من المحاضرات تظهر على الشكل التالي:

و هنا نستطيع تعديل عدد النسخ المطلوبة بالنقر على حقل العدد لاحد المحاضرات و ادخال رقم جديد .

كما نستطيع حذف محاضرة معينة من القائمة المحجوزة اي التراجع عن عملية بيعها و ذلك بالنقر على  الى اليسار من كل محاضرة

المحاضرة	السعر	العدد	حذف
النظم الخبيرة03	63.0000	2	
نظم التشغيل02	35.0000	1	
نظم الزمن الحقيقي01	28.0000	2	
نظم الزمن الحقيقي02	35.0000	1	
نظم الزمن الحقيقي03	84.0000	1	

رسم توضيحي 28

نستطيع ايضا ان نحجز محاضرة تما اخذها مسبقا من قبل الطالب على افتراض انه اضاعها او انه يريد نسخة ثانية منها لسبب ما و هنا فقط تظهر رسالة على الشكل التالي :

المحاضرة	تاريخ الرفق	مأخوذ	حجز	العدد المتبقي
شبكات01	AM 2:04:55 11/27/2013		<input checked="" type="checkbox"/>	2
شبكات02	AM 11:27:42 11/2/2013		<input checked="" type="checkbox"/>	6
شبكات03	AM 11:29:02 11/3/2013		<input checked="" type="checkbox"/>	9
شبكات04	AM 11:31:39 11/4/2013		<input type="checkbox"/>	9



رسم توضيحي 29

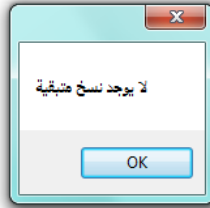
بالنقر على زر (Yes) تتم اضافة المحاضرة الى المحاضرات المحجوزة او يمكن ان نسميها (سلّة المشتريات). و أخيرا اصبحت الاخيرة على الشكل التالي :

المحاضرة	السعر	العدد	حذف
النظم الخبيزة03	63.0000	2	
نظم التشغيل02	35.0000	1	
نظم الزمن الحقيقي01	28.0000	2	
نظم الزمن الحقيقي02	35.0000	1	
نظم الزمن الحقيقي03	84.0000	1	
شبكات01	140.0000	1	

رسم توضيحي 30

وهنا عالجنا حالة استثنائية و هي عندما تنفذ النسخ الخاصة بمحاضرة ما هنا عند النقر المزدوج على المحاضرة تظهر رسالة الخطأ التالية:

المحاضرة	تاريخ الرفع	مأخوذ	حجز	العدد المتبقي
شبكات01	AM 2:04:55 11/27/2013		<input checked="" type="checkbox"/>	1
شبكات02	AM 11:27:42 11/2/2013		<input checked="" type="checkbox"/>	0
شبكات03	AM 11:29:02 11/3/2013		<input checked="" type="checkbox"/>	9
شبكات04	AM 11:31:39 11/4/2013		<input type="checkbox"/>	9



معاينة الوصل

رسم توضيحي 31

نلاحظ هنا انه بعد النقر المزدوج على محاضرة شبكات 02 - لم يتم حجزها - وظهرت رسالة بانه ما من نسخ متبقية. وفي حال حجزنا محاضرة شبكات 01 و عدلنا عدد النسخ الى 2 مثلا

المحاضرة	تاريخ الرفع	مأخوذ	حجز	العدد المتبقي	المحاضرة	السعر	العدد	حذف
شبكات 01	AM 2:04:55 11/27/2013	✓	<input checked="" type="checkbox"/>	1	النظم الخبيرة 03	63.0000	1	✗
شبكات 02	AM 11:27:42 11/2/2013	✓	<input checked="" type="checkbox"/>	0	نظم التشغيل 02	35.0000	1	✗
شبكات 03	AM 11:29:02 11/3/2013	✓	<input checked="" type="checkbox"/>	9	نظم الزمن الحقيقي 01	28.0000	1	✗
شبكات 04	AM 11:31:39 11/4/2013	✗	<input type="checkbox"/>	9	نظم الزمن الحقيقي 02	35.0000	1	✗
					نظم الزمن الحقيقي 03	84.0000	1	✗
					شبكات 01	140.0000	2	✗

العدد الموجود أقل من العدد المطلوب

OK

معاينة الوصل

رسم توضيحي 32

ظهرت رسالة الخطأ بأن العدد المتوفر من النسخ أقل من العدد المطلوب كما هو واضح و عن النقر على زر (OK) يعود العدد الى ما كلن عليه:

المحاضرة	السعر	العدد	حذف
النظم الخبيرة 03	63.0000	1	✗
نظم التشغيل 02	35.0000	1	✗
نظم الزمن الحقيقي 01	28.0000	1	✗
نظم الزمن الحقيقي 02	35.0000	1	✗
نظم الزمن الحقيقي 03	84.0000	1	✗
شبكات 01	140.0000	1	✗

رسم توضيحي 33

بالنقر على زر معاينة الوصل يظهر وصل للبيع و هنا قمنا بحفظ الوصل بقاعدة البيانات على شكل نص معد ب (XML)[9] ، يظهر الوصل على الشكل التالي :

MyMessageBox

رقم الفاتوره : 1
AM التاريخ : 11/29/2013 4:06:32

الى	مرج	الحلبي
المحاضرة <td>العدد <td>السعر </td></td>	العدد <td>السعر </td>	السعر
النظم	2	126
الخبرة 03	1	35
نظم التشغيل 02	2	56
الزمن الحقيقي 01	1	35
الزمن الحقيقي 02	1	84
الزمن الحقيقي 03	1	140
شيكات 01		



المجموع الكلي = 476 ليره سوريه
اسم المستخدم mohannad

III

Print

Close

رسم توضيحي 34

هنا يعرض مستخدم النظام (مدير ، أو موظف) هذا الوصل على الطلب و في حال وجود خطأ في المحاضرات المطلوبة يمكن تعديل هذا الوصل بالنقر على  ، اما في حال موافقة الطالب على الشراء يقوم المستخدم بالنقر على زر طباعة الوصل  هنا يطبع الوصل ، و تخرج العينات المباعة من قاعدة البيانات ، يستلم الطالب الوصل بعد دفع المبلغ المطلوب و يذهب بالوصل الى موظف آخر ليقوم الأخير باعطائه المحاضرات الموجودة ضمن الوصل .

5.6- خيارات النظام :

تظهر الواجهة على الشكل التالي و هي مزودة بحماية اضافية فتسجيل الدخول كمدير للنظام حصراً
مطلوب عند كل دخول الى هذه الواجهة ،
تظهر الواجهة على الشكل التالي :



رسم توضيحي 35

اداة تسجيل الدخول هذه خاصة بمدير النظام حصراً و لا تتيح لأية مستخدم عادي الدخول
وعند ادخال بيانات مستخدم عادي تظهر الرسالة التالية :



رسم توضيحي 36

بعد تسجيل الدخول كمدير للنظام تصبح واجهة خيارات النظام متاحة :

وهنا نلاحظ الادوات التالية :



بالنقر على كل زر تظهر الواجهة الخاصة به ، و بالنقر على خروج تقفل الواجهة و تصبح غير متاحة و يطلب تسجيل الدخول من جديد .

لنبدأ باستعراض هذه الادوات :

i. الجرد اليومي :

خصصنا جدول في قاعدة البيانات لهذه الأداة و اسمه جدول المبيعات اليومية (SalesPerDay) و هنا عند كل عملية بيع تسجل الفاتورة على شكل نص مصاغ ب (XML) ، بالإضافة لعدد الأوراق المباعة و المبلغ الوارد و الربح .

تظهر الاداة على الشكل التالي :







رسم توضيحي 37

وهنا نستطيع استعراض مبيعات يوم معين باختياره من التقويم و يقوم النظام بعرض المبلغ الوارد ، والربح الصافي ، اضافة الى امكانية استعراض كل فاتورة على حدى و يتم ذلك على الشكل التالي :

جرد يومي

مبيعات في تاريخ : AM 12:00:00 11/10/2013

Sunday , November 10, 2013


عرض	عدد الاوراق المباعة	الرقم
	150	50
	11	51
	93	70
	206	71

النتائج

3703 المبلغ الوارد

2622 الربح الصافي

رسم توضيحي 38

نستطيع استعراض كل فاتورة على حدى عن طريق النقر على زر  الي اليسار من كل فاتورة مبينة في الجدول و تظهر في حقل عرض الفواتير على الشكل التالي :

جرد يومي

مبيعات في تاريخ : AM 12:00:00 11/10/2013




Sunday , November 10, 2013

ارشف مكتبة الخدمات الطلابية

رقم الفاتوره : 3
PM التاريخ : 2:25:13 11/10/2013

الى	اخرى	كراج
المحاضرة <td>العدد <td>السعر </td></td>	العدد <td>السعر </td>	السعر
تصويق 1	2	220
تصويق 2	1	110
تصويق 3	1	110
نظم خبيرة 1	1	250
نظم خبيرة 2	1	130
فيكات 01	1	200
introduction	3	480

المجموع الكلي = 1500 ليره سوريه
اسم المستخدم hassan

عرض	عدد الاوراق المباعة	الرقم
	150	50
	11	51
	93	70
	206	71

النتائج

3703 المبلغ الوارد

2622 الربح الصافي

رسم توضيحي 39

ii. اما بالنسبة لتعديل المعطيات فهذه الاداة تتيح الاجراءات التالية :

أضافة مستخدم

تعديل المستخدمين

تغير كلمة السر

تغير الاسعار

و عند النقر على كل منها يظهر الحقل الخاص بها .

لنبدأ باضافة مستخدم :

أضافة مستخدم

اسم المستخدم	<input type="text"/>
كلمة السر	<input type="text"/>
تأكيد كلمة السر	<input type="text"/>

أدخال

رسم توضيحي 40

و تتم على الشكل التالي :

أضافة مستخدم

اسم المستخدم	samer
كلمة السر	samer
تأكيد كلمة السر	samer

أدخال

تمت الأضافة بنجاح

OK

رسم توضيحي 41

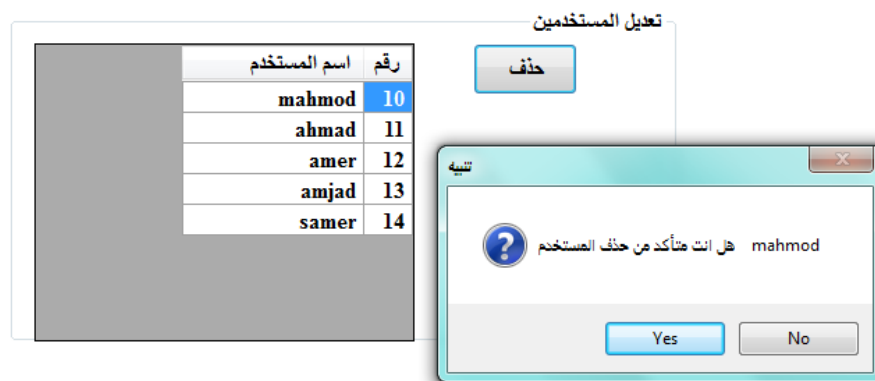
هنا اضعنا مستخدم عادي للنظام (موظف) يستطيع استعمال كل ميزات النظام باستثناء خيارات النظام .



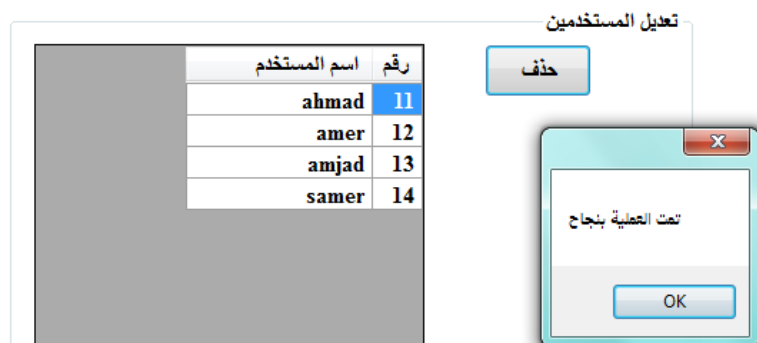
تعديل المستخدمين :

رسم توضيحي 42

هنا نستطيع حذف مستخدم معين على الشكل :



رسم توضيحي 43

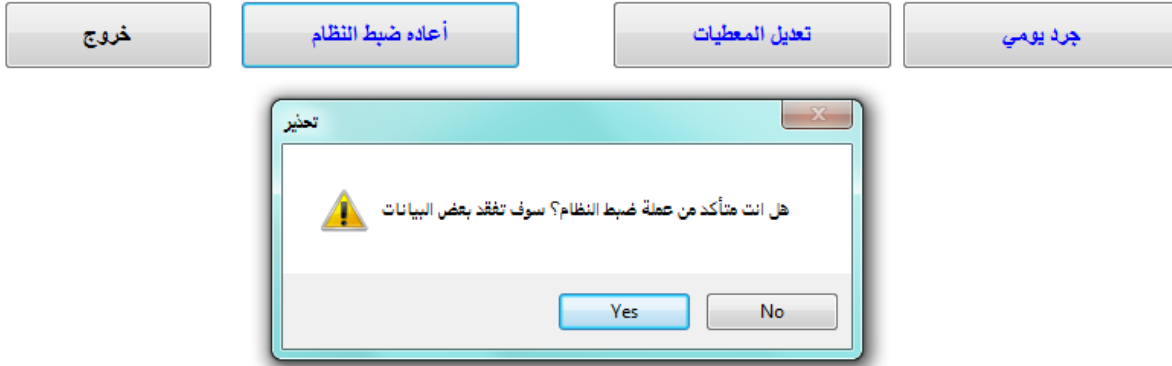


رسم توضيحي 44

iii. إعادة ضبط النظام : تتم هذه العملية عند بداية كل فصل دراسي و تتضمن :

تفريغ جدول المبيعات اليومية و تفريغ جدول المحاضرات ، و جدول المواد المسجلة لكل طالب و ايضا جدول المحاضرات المطلوبة من كل طالب ، مع الاحتفاظ ببيانات الطلاب و الكليات و المواد.

و تتم عند النقر على زر إعادة ضبط النظام فتظهر رسالة تحذير على الشكل :



رسم توضيحي 45

و عند تأكيد العملية تتم إعادة ضبط النظام لتبدأ اعمال المكتبة من جديد بتسجيل المواد الجديدة للطلاب و المحاضرات الجديدة و الخ ...

الفصل السادس

6. المقترحات المطروحة لتطوير النظام

- 1- ربط البرنامج بشبكة الانترنت أو شبكة محلية تكون ضمن نطاق مبنى الجامعة حيث يكون لكل طالب حساب خاص به وتصله رسائل الكترونية بشكل دوري بالمحاضرات الجديدة المطالب بها أو بإعلان معين يرغب أحد المحاضرين بإخبار الطلاب به أو ببرنامج الدوام الخاص بكليته .
- 2- ربط البرنامج مع واجهات عرض ضوئية و أجهزة صوت تنظم الدور بحالات الازدحام من مبدأ النافذة الواحدة.
حيث يحمل الوصل المطبوع للطالب رقم الدور مع الوقت المتوقع للانتظار.
وعندما يحين دور الطالب المعني يظهر رقم وصله على شاشة العرض مع رقم الشباك الذي سيستلم منه محاضراته بالإضافة لنداء صوتي لصاحب الوصل .
- 3- تطوير تطبيق Android يتم فيه اختيار المحاضرات المطلوبة وطلبها من المكتبة عن طريق شبكة محلية ليتم حجزها في النظام وتجهيزها لحين حضور الطالب لاستلامها.

المراجع

Books:

- [1]- The C# Programming Language Fourth Edition page 1 - .1.introduction
 - [2]- UML basics page 2 - use case diagram
 - [3]- Fundamentals of Database Systems(Elmasri,Navathe) page 52 - Ch3:Data Modeling Using ERD
 - [4]- Fundamentals of Database Systems(Elmasri,Navathe) page 100
 - [5]- C#.2010.for.Programmers.4th.Edition.Nov.2010page 481 - Tabcontrol
 - [6]- C#.2010.for.Programmers.4th.Edition.Nov.2010page 417 - GroupBoxes
 - [7]- C#.2010.for.Programmers.4th.Edition.Nov.2010..... page 598 - DataGridView
 - [8]- C#.2010.for.Programmers.4th.Edition.Nov.2010page 598 - ListBox Control
 - [9]- C#.2010.for.Programmers.4th.Edition.Nov.2010page 24 - XML Basics
- VISUAL QUICKSTART GUIDE SQL Second Edition

Websites:

www.google.com

<http://stackoverflow.com/>

<http://www.codeproject.com/>

https://www.ischool.utexas.edu/technology/tutorials/webdev/xml_dtds/03_xml.html

الملحق (الكود)

المكاتب المستخدمة:

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Linq;
using System.Text;
using System.Windows.Forms;
using System.Data.SqlClient;
using System.Security.Cryptography;
using System.Drawing;
using System.Drawing.Printing;
using System.Globalization;
using System.Diagnostics;
using System.Xml;
using System.IO;
using System.Xml.Serialization;
using System.Web;
using System.Data.SqlClient;
```

الصفوف المستخدمة :

1-DBdialer

ويستخدم لتنظيم الوصول لقاعدة البيانات ، ويحتوي على (Connection String).

```
public class DBdialer
{
    public DB_Dealer(){ }

    public static string GetConnectionString()
    {
        return @"Data Source=.;Initial Catalog=LibraryDB;Integrated Security=True";
    }
}
```

2-Bill

لتنظيم عملية البيع ضمن فاتورة و صياغة ملف XML للفاتورة ليتم حفظه في قاعدة البيانات.

```
public class Bill
{
    PrintDocument pdoc = null;
    int billNo;
    DateTime billDate;
    String total_text, Destination, DrawnBy;
    float Amount;

    public int BillNo
    {
        set { this.billNo = value; }
        get { return this.billNo; }
    }
    public DateTime BillDate
    {
        set { this.billDate = value; }
        get { return this.billDate; }
    }
}
```

```

    }

    public String Total
    {
        set { this.total_text = value; }
        get { return this.total_text; }
    }
    public String destination
    {
        set { this.Destination = value; }
        get { return this.Destination; }
    }
    public float amount
    {
        set { this.Amount = value; }
        get { return this.Amount; }
    }
    public String drawnBy
    {
        set { this.DrawnBy = value; }
        get { return this.DrawnBy; }
    }

    public Bill()
    {
    }
    public Bill(int billNo, DateTime billDate, String total_text, String Destination,
float Amount, String DrawnBy)
    {
        this.billNo = billNo;
        this.billDate = billDate;
        this.total_text = total_text;
        this.Destination = Destination;
        this.Amount = Amount;
        this.DrawnBy = DrawnBy;
    }
    public void print()
    {
        PrintDialog pd = new PrintDialog();
        pdoc = new PrintDocument();
        PrinterSettings ps = new PrinterSettings();
        Font font = new Font("Courier New", 15);

        PaperSize psize = new PaperSize("Custom", 100, 200);
        //ps.DefaultPageSettings.PaperSize = psize;

        pd.Document = pdoc;
        pd.Document.DefaultPageSettings.PaperSize = psize;
        //pdoc.DefaultPageSettings.PaperSize.Height = 320;
        pdoc.DefaultPageSettings.PaperSize.Height = 820;

        pdoc.DefaultPageSettings.PaperSize.Width = 520;

        pdoc.PrintPage += new PrintPageEventHandler(pdoc_PrintPage);

        DialogResult result = pd.ShowDialog();
        if (result == DialogResult.OK)
        {
            PrintPreviewDialog pp = new PrintPreviewDialog();
            pp.Document = pdoc;
            result = pp.ShowDialog();
            if (result == DialogResult.OK)
            {
                pdoc.Print();
            }
        }
    }
}

```

```

    }

    void show()
    {

    }

    void pdoc_PrintPage(object sender, PrintPageEventArgs e)
    {
        Graphics graphics = e.Graphics;
        Font font = new Font("Courier New", 10);
        float fontHeight = font.GetHeight();
        int startX = 50;
        int startY = 55;
        int Offset = 40;

        graphics.DrawString("الطلّابية الخدمات مكتبة في بك مرحبا", new Font("Courier New", 14),
new SolidBrush(Color.Black), startX, startY + Offset);
        Offset = Offset + 20;
        graphics.DrawString("الفاثوره رقم " + this.BillNo, new
Font("Courier New", 14), new SolidBrush(Color.Black), startX, startY + Offset);
        Offset = Offset + 20;
        graphics.DrawString("التاريخ : " + this.BillDate, new Font("Courier New",
12), new SolidBrush(Color.Black), startX, startY + Offset);
        Offset = Offset + 20;
        String underline = "-----";
        graphics.DrawString(underline, font, new SolidBrush(Color.Black), startX,
startY + Offset);

        ///////////////
        Offset = Offset + 20;

        graphics.DrawString("الى " + destination, font, new
SolidBrush(Color.Black), startX, startY + Offset);

        Offset = Offset + 20;
        String points_line = ".....";
        graphics.DrawString(points_line, font, new SolidBrush(Color.Black), startX,
startY + Offset);

        ///////////////

        Offset = Offset + 20;
        String BaseLine = "المحاضره\t\t\t\t\tالعدد\t\t\t\t\tالسعر";
        graphics.DrawString(BaseLine, font, new SolidBrush(Color.Black), startX, startY
+ Offset);

        int h_size = 0;
        int height = 392;
        String Total_text = this.total_text;
        string[] records = Total_text.Split('~');
        foreach (string record in records)
        {
            h_size++;
            string[] info_s = record.Split(',');
            string main_info = "";

            if (record != "")
            {
                for (int i = 2; i >= 0; i--)
                {

                    main_info += info_s[i] + "\t\t\t\t";

                }
            }
            main_info += "\n";
        }
    }

```

```

        Offset = Offset + 20;
        graphics.DrawString(main_info, font, new SolidBrush(Color.Black), startX,
startY + Offset);
        if (h_size > 5)
        {
            height += 15;
        }
    }

    Form1.form.pictureBox_Amer.Height = height;
    ////////////////

    Offset = Offset + 20;
    underLine = "-----";
    graphics.DrawString(underLine, font, new SolidBrush(Color.Black), startX,
startY + Offset);

    Offset = Offset + 20;
    String Grosstotal = "                السوريه ليره " + this.amount + " الكلي المجموع ";

    Offset = Offset + 20;

    graphics.DrawString(Grosstotal, font, new SolidBrush(Color.Black), startX,
startY + Offset);
    Offset = Offset + 20;
    String DrawnBy = this.drawnBy;
    graphics.DrawString("                المستخدم اسم " + DrawnBy, font, new
SolidBrush(Color.Black), startX, startY + Offset);
    }

    public void Convert_To_XML(int _n_lectures_pages)
    {
        //try
        //{
            XmlSerializer xs = new XmlSerializer(typeof(Bill));
            StringWriter sw = new StringWriter();
            xs.Serialize(sw, this);
            string BillXML = sw.ToString();
            sw.Close();
            decimal base_price = 0;
            // DeSerializeObject<Bill>(objectXML);
            //Update to datadase
            System.Data.SqlClient.SqlConnection connection = new
System.Data.SqlClient.SqlConnection(DB_Dealer.GetConnectionString());
            System.Data.SqlClient.SqlCommand command;
            connection.Open();
            string prices = ("select PaperPrice,Paper_base_Price from Users where
Status = '"+Admin+"' ").ToString();
            command = new System.Data.SqlClient.SqlCommand(prices, connection);

            System.Data.SqlClient.SqlDataReader reader= command.ExecuteReader();
            if (reader.Read())
            {
                base_price = Convert.ToDecimal(reader[1]);
            }
            reader.Close();

            string query_insert_sales_per_say = " INSERT INTO
sales_per_day([date],soledpages,bill,income,gain) VALUES ('" + DateTime.Now + "',''+
_n_lectures_pages + "','N'" + BillXML + "',''+ this.amount + "',''+
Convert.ToDecimal(base_price*_n_lectures_pages) + "')";
            command = new System.Data.SqlClient.SqlCommand(query_insert_sales_per_say,
connection);
            command.ExecuteNonQuery();

        //}
    }

```



```

        //catch (Exception ex)
        //{

        //}

    }

    static public Bill DeSerializeObject<Bill>(string fileName)
    {
        if (string.IsNullOrEmpty(fileName)) { return default(Bill); }
        Bill BillOut = default(Bill);
        try
        {
            XmlSerializer serializer = new XmlSerializer(typeof(Bill));
            StringReader reader = new StringReader(fileName);
            BillOut = (Bill)serializer.Deserialize(reader);
            reader.Close();
        }
        catch (Exception ex)
        {
        }
        return BillOut;
    }
}

```

الاجرائيات المستخدمة:

سوف نستخدم حالات الاستخدام (Use Cases) بتسلسلها لنعرض الاجرائيات المستخدمة لتحقيق كل منها:

01

تسجيل الدخول الى البرنامج

```
public partial class Form2 : Form
{
    public Form2()
    {
        InitializeComponent();
    }

    public void Form2_Load(object sender, EventArgs e){}

    public void button1_Click(object sender, EventArgs e)
    {
        if (textBox1.Text != "" || textBox2.Text != "")
        {
            try
            {
                SqlConnection connection = new SqlConnection(DB_Dealer.GetConnectionString());
                connection.Open();
                string query = ("select Userpassword,PaperPrice from Users where Username='" +
textBox1.Text.Trim() + "'").ToString();
                SqlCommand command = new SqlCommand(query, connection);
                SqlDataReader reader = command.ExecuteReader();
                if (reader.Read())
                {
                    //reader.Read();
                    if ((string.Compare(reader[0].ToString(), Form1.MD5Hash(textBox2.Text),
true) == 0))
                    {
                        Form1.form.Enabled = true;
                        this.Close();
                        Form1.form.Visible = true;
                        Form1.form.label_User_name.Text = textBox1.Text;
                        // Form1.form.label62.Text = reader[1].ToString(); ;
                    }
                    else
                    {
                        MessageBox.Show("المرور كلمة في خطأ", "خطأ", MessageBoxButtons.OK,
MessageBoxIcon.Error);
                    }
                }
                else
                {
                    MessageBox.Show("المستخدم اسم في خطأ", "خطأ", MessageBoxButtons.OK,
MessageBoxIcon.Error);
                }
                connection.Close();
            }
            catch (Exception ex)
            {
                //{ MessageBox.Show(ex.ToString()); }
                { throw (ex); }
            }
        }
        else
        {
            // MessageBox.Show("الحقول كل ملء الرجاء");
            MessageBox.Show("الحقول كل ملء الرجاء", "خطأ", MessageBoxButtons.OK,
```

```

MessageBoxIcon.Error);
    }
}

private void textBox2_KeyPress(object sender, KeyPressEventArgs e)
{
    if (e.KeyChar == 13)
    {
        if (!textBox2.AcceptsReturn)
        {
            button1.PerformClick();
        }
    }
}

private void button2_Click(object sender, EventArgs e)
{
    Form1.form.Enabled = true;
    this.Close();
    Form1.form.Visible = true;
    Form1.form.label_User_name.Text = textBox1.Text;
}

private void button3_Click(object sender, EventArgs e)
{
    Form1.form.Close();
}
}

```

011

البحث عن المحاضرات

```

private void button3_Click(object sender, EventArgs e)
{
    number_of_taken_lectures = "";
    dataGD_lectures.Rows.Clear();
    dataGridV_token_lecture.Rows.Clear();
    int student_id = -1;
    if (txtb_student_search.Text == "")
    {
        MessageBox.Show("للتألب رقم اءءال الرءاء", "ملاءة", MessageBoxButtons.OK,
        MessageBoxIcon.Information);
        txtb_student_search.Focus();
    }
    else
    {
        SqlConnection connection = new SqlConnection(DB_Dealer.GetConnectionString());
        connection.Open();
        string query = ("select * from Student where stdID='" +
        Convert.ToInt32(txtb_student_search.Text) + "'").ToString();
        string query_price = ("select PaperPrice from Users").ToString();
        SqlCommand command = new SqlCommand(query, connection);
        SqlDataReader reader = command.ExecuteReader();
        if (reader.Read())
        {
            student_id = Convert.ToInt32(reader[0]);
            student_fname = reader["fName"].ToString();
            student_lname = reader["lname"].ToString();
            student_level = reader["Level"].ToString();
            //student_faculty = reader["Faculty"].ToString();
            get_student_info(student_id);
            reader.Close();
            command = new SqlCommand(query_price, connection);
            reader = command.ExecuteReader();

```

```

        reader.Read();
        paperprice = Convert.ToDouble( reader["PaperPrice"]);
    }
    else
    {
        MessageBox.Show("خطأ", "آخر رقم ادخال الرجاء \nموجود غير أدخلته الذي الرقم",
        MessageBoxButtons.OK, MessageBoxIcon.Error);
        txtb_student_search.Text = "";
        txtb_student_search.Focus();
    }

    reader.Close();
    connection.Close();
}
}
}

```

0111

حجز المحاضرة

```

private void dataGD_lectures_CellDoubleClick(object sender, DataGridViewCellEventArgs e)
{
    bool flag_mes = false ;
    bool flag_pass = true ;

    if (Convert.ToInt32(Convert.ToInt32(dataGD_lectures.Rows[e.RowIndex].Cells[5].Value)) == 0)
    {
        MessageBox.Show("متبقية نسخ يوجد لا");
    }
    else if (e.ColumnIndex > -1 && e.RowIndex > -1)
    {
        foreach (DataGridViewRow row in dataGridV_token_lecture.Rows)
        {
            if (row.Cells[0].Value.ToString() ==
dataGD_lectures.Rows[e.RowIndex].Cells[0].Value.ToString())
            {
                flag_pass = false;
                break;
            }
        }

        if (flag_pass)
        {
            if
(Convert.ToBoolean(((DataGridViewCheckBoxCell)dataGD_lectures.Rows[e.RowIndex].Cells[Col_t.Name]).Value)
== true)
            {
                if (dataGD_lectures.Rows[e.RowIndex].Cells[Col_token.Name].Value == pic_true)
                {
                    flag_mes = true;
                }
            }
            DialogResult dialogResult = DialogResult.Yes;

            if (flag_mes)
            {
                string data = "اخبره مره المحاضره هذا اخذ تريد هل ";
                data += dataGD_lectures.Rows[e.RowIndex].Cells[1].Value.ToString();
                dialogResult = MessageBox.Show(data, "تحذير", MessageBoxButtons.YesNo,
MessageBoxIcon.Question);
            }
        }
    }
}

```



```

private void bt_add_std_Click(object sender, EventArgs e)
{
    if (textB_student_id.Text == "")
        MessageBox.Show("للطالب رقم إدخال الرجاء", "تنبيه", MessageBoxButtons.OK,
        MessageBoxIcon.Warning);
    else if (tb_level.Text == "" || tb_dep.Text == "" || textB_student_id.Text == "" ||
    tb_lname.Text == "" || tb_fa.Text == "" || tb_dep.Text == "" || tb_semester.Text == "")
    {
        MessageBox.Show("المطلوبه المعومات جميع ادخال الرجاء", "تنبيه", MessageBoxButtons.OK,
        MessageBoxIcon.Warning);
    }
    else {
        SqlConnection connection = new SqlConnection(DB_Dealer.GetConnectionString());
        connection.Open();
        string query = ("select * from Student where stdID=" +
        Convert.ToInt32(textB_student_id.Text) + "").ToString();
        SqlCommand command = new SqlCommand(query, connection);
        SqlDataReader reader = command.ExecuteReader();
        if (!reader.Read())
        {
            Student new_student = new Student(Convert.ToInt32(textB_student_id.Text),
            tb_fn.Text, tb_lname.Text, tb_dep.Text, tb_fa.SelectedValue.ToString(), tb_level.Text,
            tb_semester.Text);

            new_student.insert();
            textB_student_id.Text = "";
            tb_fn.Text = "";
            tb_fa.Text = "";
            tb_lname.Text = "";
            tb_semester.Text = "";
            tb_dep.Text = "";
            tb_level.Text = "";
            MessageBox.Show("بنجاح الطالب اضافة تمت", "ملاحظة", MessageBoxButtons.OK,
            MessageBoxIcon.Information);
        }
        else MessageBox.Show("موجود الرقم هذا", "خطأ", MessageBoxButtons.OK,
        MessageBoxIcon.Error);
        reader.Close();
        connection.Close();
    }
}

```

```

private void button1_Click(object sender, EventArgs e)
{
    if( tb_level.Text == "" || tb_semester.Text == "" || tb_fn.Text == "" || tb_lname.Text
    == "" || tb_fa.Text == "" || tb_dep.Text == "" )
    {
        MessageBox.Show("المعومات جميع ادخال الرجاء", "خطأ", MessageBoxButtons.OK, MessageBoxIcon.Error);
    }
    else
    {
        dataGridV_subjects.Rows.Clear();
        SqlConnection connection = new SqlConnection(DB_Dealer.GetConnectionString());
        connection.Open();
    }
}

```

```

        string query_update = ("update Student set fName=N'" + tb_fn.Text + "', lname= N'"
+ tb_lname.Text + "' , Faculty= N'" + tb_fa.SelectedValue.ToString() + "' , Department= N'" +
tb_dep.Text + "' , Level= '" + tb_level.Text + "',Semester='" + tb_semester.Text + "' where stdID
= '" + student.id + "'").ToString();
        SqlCommand command = new SqlCommand(query_update, connection);
        command.ExecuteNonQuery();
        connection.Close();
        groupB_edit_sub.Enabled = true;
        MessageBox.Show("بنجاح التحديث تم", "ملاحظة", MessageBoxButtons.OK,
        MessageBoxIcon.Information);
        subject();
    }

}

//chose lectures to be added:
private void but_Add_sub_Click(object sender, EventArgs e)
{
    bool flag = true ;
    for (int i = 0; i < listB_Select_sub.Items.Count; i++)
    {
        if (listB_Select_sub.Items[i].ToString() ==
dataGridV_subjects.Rows[dataGridV_subjects.CurrentRow.Index].Cells[1].Value.ToString() )
        {
            flag = false;
            break;
        }
    }

    for (int i = 0; i < listB_his_subject.Items.Count; i++)
    {
        if (listB_his_subject.Items[i].ToString() ==
dataGridV_subjects.Rows[dataGridV_subjects.CurrentRow.Index].Cells[1].Value.ToString())
        {
            flag = false;
            MessageBox.Show("بالفعل مسجلة المادة هذه");
            break;
        }
    }

    if(flag)
listB_Select_sub.Items.Add(dataGridV_subjects.Rows[dataGridV_subjects.CurrentRow.Index].Cells[1].V
alue.ToString());

}

//remove Regestered lecture
private void but_remove_sub_Click(object sender, EventArgs e)
{
    if(listB_Select_sub.SelectedIndex >= 0)
listB_Select_sub.Items.RemoveAt(listB_Select_sub.SelectedIndex);

}

//Update Regestered Lectures:
private void button1_Click_2(object sender, EventArgs e)
{
    string sub_id = "";
    SqlConnection connection = new SqlConnection(DB_Dealer.GetConnectionString());
    SqlCommand command;
    SqlDataReader reader = null;
    connection.Open();
    for (int i = 0; i < listB_Select_sub.Items.Count; i++)
    {
        string query_select_subject = ("SELECT * from Subject WHERE (subname = N'" +
listB_Select_sub.Items[i].ToString() + "')").ToString();
        command = new SqlCommand(query_select_subject, connection);
        reader = command.ExecuteReader();
    }
}

```

```

        if (reader.Read()) sub_id += reader[0].ToString() + ",";
        reader.Close();
    }

    string[] ids = sub_id.Split(',');
    foreach (string id in ids)
    {
        if (id != "")
        {
            string query_insert_subject_to_student = ("insert into std_sub (stdID, subID )
values ('" + student.id + "','" + id + "')").ToString(); //string query_select
= ("select LectName , NumOfPages * ' ' + paperprice + ' ' FROM Lecture WHERE (LectID = ' ' +
Convert.ToInt16(num) + ' ')").ToString();
            command = new SqlCommand(query_insert_subject_to_student, connection);
            command.ExecuteNonQuery();
            string query_select_student = ("SELECT subID,lectID from sub_lect WHERE
(subID = ' ' + id + ' ')").ToString();
            command = new SqlCommand(query_select_student, connection);
            reader = command.ExecuteReader();

            string let_ids = "";
            while (reader.Read())
            {
                let_ids += reader["lectID"] + ",";
            }
            reader.Close();

            string[] id_s = let_ids.Split(',');
            foreach (string id_ in id_s)
            {
                if (id_ != "")
                {
                    string query_insert_lect_to_student = "insert into std_lect
(stdID,lectID,token) values ('" + textB_student_id.Text + "','" + id_ + "','" + "0" + "')";
                    SqlCommand command_3 = new SqlCommand(query_insert_lect_to_student,
connection);
                    command_3.ExecuteNonQuery();
                }
            }
        }
    }
    if(listB_Select_sub.Items.Count!=0)
    {
        MessageBox.Show("بنجاح المواد اضافة تم");
        Fill_his_subject();
        connection.Close();
    }
    Clear_Subject_list();
}

```

014

حذف طالب

```

private void button2_Click_1(object sender, EventArgs e)
{
    if (tb_id.Text == "")
    {
        MessageBox.Show("الطالب رقم ادخال الرجاء", "خطأ", MessageBoxButtons.OK, MessageBoxIcon.Error);
    }
    else if (tb_id.Text != "")
    {
        DialogResult dialogResult1 = MessageBox.Show("ملاحظة", "الطالب؟ حذف تريد هل",
        MessageBoxButtons.YesNo, MessageBoxIcon.Question);
    }
}

```



```

        if (dialogResult1 == DialogResult.Yes)
        {
            SqlConnection connection = new SqlConnection(DB_Dealer.GetConnectionString());
            connection.Open();
            string query_delete_stdlect = ("delete from std_lect where stdID='" +
Convert.ToInt32(textB_student_id.Text) + "'").ToString();
            string query_delete_stdsub = ("delete from std_sub where stdID='" +
Convert.ToInt32(textB_student_id.Text) + "'").ToString();
            string query_delete_student = ("delete from Student where stdID='" +
Convert.ToInt32(textB_student_id.Text) + "'").ToString();
            SqlCommand command = new SqlCommand(query_delete_stdlect, connection);
            SqlDataReader reader = command.ExecuteReader();
            reader.Read();
            reader.Close();
            command = new SqlCommand(query_delete_stdsub, connection);
            reader = command.ExecuteReader();
            reader.Read();
            reader.Close();
            command = new SqlCommand(query_delete_student, connection);
            reader = command.ExecuteReader();
            reader.Read();
            reader.Close();
            connection.Close();
            MessageBox.Show("الطالب حذف تم", "تنبيه", MessageBoxButtons.OK,
MessageBoxIcon.Information);
            tb_fa.Text = "";
            tb_id.Text = "";
            tb_level.Text = "";
            tb_lname.Text = "";
            tb_semester.Text = "";
            tb_dep.Text = "";
            tb_fn.Text = "";
            textB_student_id.Text = "";
            listB_his_subject.Items.Clear();
            listB_Select_sub.Items.Clear();
        }
    }
}

```

015

اضافة محاضرة

```

private void but_add_lect_Click(object sender, EventArgs e)
{
    Lecture lecture = new Lecture(textB_lect_name.Text, textB_lect_number_pages.Text,
textB_lect_number_copies.Text);
    int id_lecture = lecture.store_lecture();
    SqlConnection connection = new SqlConnection(DB_Dealer.GetConnectionString());
    connection.Open();
    string query_select_sub_id = ("SELECT * from Subject where (subname LIKE N'%" +
comboB_subjects.Text + "%')").ToString();
    SqlCommand command = new SqlCommand(query_select_sub_id, connection);
    SqlDataReader reader = command.ExecuteReader();
    string sub_id = "";
    if (reader.Read())
    {
        sub_id = reader["subID"].ToString();
    }
    reader.Close();
    if (textB_lect_name.Text != "" && textB_lect_number_pages.Text != "" &&
textB_lect_number_copies.Text != "")
    {
        string query_insert_lect_to_subject = "insert into sub_lect (subID,lectID) values ('" +

```

```

sub_id + "','" + id_lecture + "');";
command = new SqlCommand(query_insert_lect_to_subject, connection);
command.ExecuteNonQuery();
string query_select_student = ("SELECT stdID,subID from std_sub WHERE (subID = '" +
sub_id + "')").ToString();
SqlCommand command_2 = new SqlCommand(query_select_student, connection);
reader = command_2.ExecuteReader();
string std_id = "";
while (reader.Read())
{
    std_id += reader["stdID"] + ",";
}
reader.Close();

string[] ids = std_id.Split(',');
foreach (string id in ids)
{
    if (id != "")
    {
        string query_insert_lect_to_student = "insert into std_lect
(stdID,lectID,token) values ('" + id + "','" + id_lecture + "','" + "0" + "')";
        SqlCommand command_3 = new SqlCommand(query_insert_lect_to_student,
connection);
        command_3.ExecuteNonQuery();
    }
}
MessageBox.Show("جديدة محاضرة اضافة تم");
connection.Close();

textB_lect_number_copies.Text = "";
textB_lect_number_papes.Text = "";
textB_lect_name.Text = "";
comboBox_subjects.Text = "";
}
else MessageBox.Show("خطأ", "الحقول جميع إدخال الرجاء",
"بالادخال", MessageBoxButtons.OK, MessageBoxIcon.Error);
}

```

0151

اضافة نسخ محاضرة

```

private void but_edit_number_of_copies_Click(object sender, EventArgs e)
{
    if (comboBox_faculty.SelectedValue != null)
    {
        if (comboBox_lectures.Text != "")
        {
            if (textB_lect_number_copies.Text != "")
            {
                SqlConnection connection = new SqlConnection(DB_Dealer.GetConnectionString());
                connection.Open();
                string query_update = ("update Lecture set NumOfCopies += '" +
Convert.ToDecimal(textB_lect_number_copies.Text) + "' where (LectName LIKE N'%" + comboBox_lectures.Text
+ "%')").ToString();
                SqlCommand command_1 = new SqlCommand(query_update, connection);

                command_1.ExecuteNonQuery();
                //this.Close();
                MessageBox.Show("بنجاح التعديل تم");
            }
            else MessageBox.Show("النسخ عدد أدخل");
        }
    }
}

```

```

private void but_add_subject_Click(object sender, EventArgs e)
{
    SqlConnection connection = new SqlConnection(DB_Dealer.GetConnectionString());
    connection.Open();
    string query = ("select * from Subject where subID='" + (textB_sub_id.Text) +
    """).ToString();
    string query_insert_lect_to_student = "insert into Subject (subID , subname ,
    Faculty , Department , Level , Semester , Instructor) values (N'" + textB_sub_id.Text + "',N'" +
    textB_sub_name.Text + "',N'" + comboB_faculty_2.SelectedValue + "',N'" + textB_sub_depar.Text + "', '" +
    textB_level.Text + "', '" + textB_semester.Text + "',N'" + textB_Instructor.Text + "' )";
    SqlCommand command = new SqlCommand(query, connection);
    SqlDataReader reader = command.ExecuteReader();
    if (!reader.Read())
    {
        if (textB_sub_id.Text != "" && textB_sub_name.Text != "" &&
        textB_sub_depar.Text != "")
        {
            reader.Close();
            command = new SqlCommand(query_insert_lect_to_student, connection);
            command.ExecuteNonQuery();
            MessageBox.Show("بنجاح المادة اضافة تم");
            clear_subjects();
        }
        else
            MessageBox.Show("المطلوبة الحقول جميع ادخل خطأ", MessageBoxButtons.OK,
            MessageBoxIcon.Error);
    }
    else
        MessageBox.Show("مسبقا موجود المادة رقم", "بالإدخال خطأ", MessageBoxButtons.OK,
        MessageBoxIcon.Error);

    reader.Close();
    connection.Close();
}

```

```

private void button4_Click_1(object sender, EventArgs e)
{
    if (textB_faculty_name.Text != "")
    {
        SqlConnection connection = new SqlConnection(DB_Dealer.GetConnectionString());
        string query_insert_lect_to_student = "insert into Faculty (FacultyName) values (N'" +
        textB_faculty_name.Text + "' )";
        connection.Open();
        SqlCommand command_3 = new SqlCommand(query_insert_lect_to_student, connection);
        command_3.ExecuteNonQuery();
        MessageBox.Show("بنجاح الكلية اضافة تم");
        connection.Close();
        textB_faculty_name.Text = "";
    }
    else
    {
        MessageBox.Show("الكلية اسم ادخال الرجاء");
    }
}

```

```

private void button1_Click_3(object sender, EventArgs e)
{
    if (textB_username.Text != "" || textB_password.Text != "")
    {
        try
        {
            SqlConnection connection = new SqlConnection(DB_Dealer.GetConnectionString());
            connection.Open();
            string query = ("select Userpassword from Users where Username='" +
textB_username.Text.Trim() + "'and status ='Admin'").ToString();
            SqlCommand command = new SqlCommand(query, connection);
            SqlDataReader reader = command.ExecuteReader();
            if (reader.Read())
            {
                //reader.Read();
                if ((string.Compare(reader[0].ToString(), MD5Hash(textB_password.Text), true)
== 0))
                {
                    panel_password.Visible = false;
                    groupB_system_option.Enabled = true;
                    //this.Close();
                }
                else
                {
                    MessageBox.Show("المروور كلمة في خطأ");
                }
            }
            else
            {
                MessageBox.Show("المدخلات في خطأ");
            }
            connection.Close();
        }
        catch (Exception ex)
        { throw (ex); }
    }
    else
    {
        MessageBox.Show("الحقول كل ملء الرجاء");
    }
}

```

```

private void fill_bill_g()
{
    dataGV_sell_per_day.Rows.Clear();
    DateTime date = dateTimePicker1.Value;
    SqlConnection connection = new SqlConnection(DB_Dealer.GetConnectionString());
    SqlCommand command;
    SqlDataReader reader = null;
    connection.Open();
    string query = ("SELECT id,date,soledpages , bill ,income ,gain FROM sales_per_day WHERE
(date = '" + date + "'").ToString();

    command = new SqlCommand(query, connection);
    reader = command.ExecuteReader();
    bool flag = false;

```

```

decimal income=0;
decimal gain=0;
while (reader.Read())
{
    dataGV_sell_per_day.Rows.Add(reader[0], reader[1], reader[2],reader[3]);
    flag = true;
    income +=Convert.ToDecimal(reader[4]);
    gain += Convert.ToDecimal(reader[5]);
}
if (flag)
{
    label_sell_daily.Text = Convert.ToSingle(income).ToString();
    label_gain_daily.Text = Convert.ToSingle(gain).ToString();
}
else
{
    label_sell_daily.Text = "اليوم هذا مبيعات لا";
}
reader.Close();

connection.Close();

}

```

022

تعديل الاسعار

```

private void button5_Click_1(object sender, EventArgs e)
{
    if (textB_buying_price.Text == "" || textB_selling_price.Text == "")
    {
        MessageBox.Show("الحقول جميع املئ الرجاء");
    }
    else
    {
        SqlConnection connection = new SqlConnection(DB_Dealer.GetConnectionString());
        connection.Open();
        string query_update = ("update Users set PaperPrice = '" +
Convert.ToDecimal(textB_selling_price.Text) + "',Paper_base_Price = '" +
Convert.ToDecimal(textB_buying_price.Text) + "' where Status = '" + "Admin" + "'").ToString();
        SqlCommand command_1 = new SqlCommand(query_update, connection);

        command_1.ExecuteNonQuery();
        MessageBox.Show(" بنجاح التحديث تم ");
    }
}
}

```

```
private void but_reset_Click(object sender, EventArgs e)
{
    DialogResult dialogResult = MessageBox.Show("البيانات بعض تفقد سوف النظام؟ ضبط عملة من متأكد انت هل",
        "تحذير", MessageBoxButtons.YesNo, MessageBoxIcon.Warning);
    if(dialogResult == DialogResult.Yes)
    {
        groupB_edit_option.Visible = false ;
        groupB_g_daily.Visible = false;
        groupB_edit_prices.Visible = false;
        groupB_edit_main_password.Visible = false;
        groupB_add_users.Visible = false;
        groupB_edit_users.Visible = false;
        SqlConnection connection = new SqlConnection(DB_Dealer.GetConnectionString());
        SqlCommand command;
        string Delete_Lecture = "delete from Lecture";
        string Delete_std_lect = "delete from std_lect";
        string Delete_std_sub = "delete from std_sub";
        string Delete_sub_lect = "delete from sub_lect";
        connection.Open();

        command = new SqlCommand(Delete_std_lect, connection);
        command.ExecuteNonQuery();
        command = new SqlCommand(Delete_std_sub, connection);
        command.ExecuteNonQuery();
        command = new SqlCommand(Delete_sub_lect, connection);
        command.ExecuteNonQuery();

        command = new SqlCommand(Delete_Lecture, connection);
        command.ExecuteNonQuery();
        connection.Close();
    }
}
```

```
private void but_enter_user_Click(object sender, EventArgs e)
{
    if (textB_new_u_c_password.Text == "" || textB_new_u_password.Text == "" ||
    textB_new_user.Text == "")
    {
        MessageBox.Show("الحقول جميع ملئ الرجاء");
    }
    else if (textB_new_u_c_password.Text != textB_new_u_password.Text)
    {
        MessageBox.Show("متطابقه غير السر كلمة");
        textB_new_u_c_password.Text = ""; textB_new_u_password.Text = "";
    }
    else
    {
        SqlConnection connection = new SqlConnection(DB_Dealer.GetConnectionString());
        SqlCommand command;
        connection.Open();
        string query_insert_lect_to_student = "insert into Users
        (Username,Userpassword,PaperPrice ) values ('" +textB_new_user.Text + "','" +
        MD5Hash(textB_new_u_password.Text) + "','" + "0" + "')";
```

```

        command = new SqlCommand(query_insert_lect_to_student, connection);
        command.ExecuteNonQuery();
        connection.Close();
        MessageBox.Show("بنجاح الأضافة تمت");
    }
}

```

025

حذف مستخدم

```

private void but_delete_users_Click(object sender, EventArgs e)
{
    if (dataGridV_edit_users.Rows.Count > 0)
    {
        string data = "المستخدم حذف من متأكد انت هل ";
        data +=
dataGridV_edit_users.Rows[dataGridV_edit_users.CurrentRow.Index].Cells[1].Value.ToString();
        DialogResult dialogResult = MessageBox.Show(data,
            "تنبيه", MessageBoxButtons.YesNo, MessageBoxIcon.Question);
        if (dialogResult == DialogResult.Yes)
        {
            SqlConnection connection = new SqlConnection(DB_Dealer.GetConnectionString());
            SqlCommand command;
            string Delete_user = "delete from Users where UserID =" +
dataGridV_edit_users.Rows[dataGridV_edit_users.CurrentRow.Index].Cells[0].Value + "";

            connection.Open();

            command = new SqlCommand(Delete_user, connection);
            command.ExecuteNonQuery();

            dataGridV_edit_users.Rows.RemoveAt(dataGridV_edit_users.CurrentRow.Index);
            connection.Close();
            MessageBox.Show("بنجاح العملية تمت");
        }
    }
}

```

026

تغيير كلمة مرور المدير

```

private void but_enter_new_main_password_Click(object sender, EventArgs e)
{
    if (textB_c_main_password.Text == "" || textB_c_new_password.Text == ""
||textB_cc_new_password.Text=="")
    {
        MessageBox.Show("الحقول جميع املئ الرجاء");
    }
    else if (textB_c_new_password.Text != textB_cc_new_password.Text)
    {
        MessageBox.Show("متطابقه غير السر كلمة");
        textB_new_u_c_password.Text = ""; textB_new_u_password.Text = "";
    }
    else{
        SqlConnection connection = new SqlConnection(DB_Dealer.GetConnectionString());
        SqlConnection connection_2 = new SqlConnection(DB_Dealer.GetConnectionString());
        connection.Open();
        SqlDataReader reader = null;
        string query = ("select Username , Userpassword from Users where Status =" + "Admin" +
""").ToString();
        SqlCommand command = new SqlCommand(query, connection);
    }
}

```

```

        reader = command.ExecuteReader();

        if (reader.Read())
        {
            if ((string.Compare(reader[1].ToString(), MD5Hash(textB_c_main_password.Text), true)
== 0))
            {
                string query_update = ("update Users set Userpassword = '" +
MD5Hash(textB_new_u_password.Text) + "', Username = '" + textB_user_name.Text + "' where Status = '" +
"Admin" + "'").ToString();
                reader.Close();
                SqlCommand command_1 = new SqlCommand(query_update, connection);
                command_1.ExecuteNonQuery();
                MessageBox.Show("السر كلمة تغير تم");
                reader.Close();
            }
            else
            {
                MessageBox.Show("الحالية المرور كلمة في خطأ");
            }
        }

        connection.Close();
    }
}

```

003

ملاحظات/تشبيهات النظام

```

private void get_notes()
{
    listBox_notes.Items.Clear();
    listBox_notes1.Items.Clear();
    listBox_notes2.Items.Clear();
    SqlConnection connection = new SqlConnection(DB_Dealer.GetConnectionString());
    connection.Open();
    string query_select_NOTES = ("SELECT * from Lecture where (NumOfCopies < '" + 3 +
"')").ToString();
    SqlCommand command = new SqlCommand(query_select_NOTES, connection);
    SqlDataReader reader = command.ExecuteReader();
    string text = "";
    while (reader.Read())
    {
        text += " محاضره من يوجد ( " + reader[1].ToString() + " ) " + reader[4].ToString() + " نسخة  

فقط \n";

        listBox_notes.Items.Add(text);
        listBox_notes1.Items.Add(text);
        listBox_notes2.Items.Add(text);
        text = "";
        listBox_notes.Items.Add("-----\n");
        listBox_notes1.Items.Add("-----\n");
        listBox_notes2.Items.Add("-----\n");
    }
}

```