



جامعة طرابلس
كلية تقنية المعلومات
قسم هندسة البرمجيات

اسم المشروع : WorkPal

Freelancer Task Manager Report

إعداد الطالبات :

- سجي جمال الدين علي بن موسى 2231808291
- بشائر جمعة غوار 2231816091

تحت إشراف : د. مي مفتاح البعباع

1. نظرة عامة على النظام (Overview of the system) :

يواجه العاملون المستقلون صعوبة في إدارة مشاريعهم المتعددة، والحفاظ على تحفيزهم تجاه الأهداف طويلة المدى ، وتنظيم مهامهم . لذلك قررنا إنشاء نظام يعمل كمنصة مركزية تساعد العاملين المستقلين على تنظيم سير أعمالهم اليومية وتتبع إنجازاتهم العملية والتطويرية ، يهدف المشروع إلى تسهيل متابعة المشاريع والمهام ، وتنظيم الأهداف على شكل خطوات قابلة للإنجاز، مع واجهة مستخدم سهلة وسلسة .

2. التقنيات والادوات المستخدمة (Technologies and tools used) :

- **لغة البرمجة - Java** : لبناء التطبيق بالكامل و **JDBC** للتعامل مع قاعدة البيانات.
- **الواجهة الرسومية (GUI) - Java Swing** : لإنشاء النوافذ، الجداول، الأزرار، مربعات النصوص، وقوائم الاختيار .
- **قاعدة البيانات MySQL** : لتخزين بيانات المستخدمين ، المشاريع ، المهام، الأهداف ، وخطوات الأهداف . **phpMyAdmin** : لإدارة قاعدة البيانات بسهولة.
- **الاتصال بقاعدة البيانات - JDBC (Java Database Connectivity)** : للاتصال بقاعدة البيانات وإجراء العمليات (CRUD) على الجداول.
- **إدارة النسخ والتحكم بالإصدار - GitHub** : لمتابعة التغييرات في المشروع، التعاون بين الفريق ، وإدارة النسخ المختلفة.
- **أدوات التطوير IDE - NetBeans** : بيئة التطوير المتكاملة لكتابة كود Java، إدارة المشروع ، وتجربة النماذج.
- **إدارة الجلسات** : استخدام **SessionManager** داخل المشروع لإدارة تسجيل الدخول والخروج وحفظ حالة المستخدم.
- **تنسيقات البيانات: SimpleDateFormat** : للتعامل مع التواريخ في المشروع (startDate, endDate, dueDate) .

3 . الميزات الأساسية في النظام (Core System Features) :

- تسجيل الدخول الى النظام .
- اضافة مشروع .
- اضافة مهام الى المشروع بعد انشاءه .

- تحديث و عرض كل المهام وعرض المشاريع المرتبطة بها .
- تعديل المهام المحددة .
- حذف المهام المختارة .
- اضافة هدف و وصف له .
- تحديث وعرض الهدف في الجدول .
- اختيار هدف وتعديله او حذفه .
- في واجهة اخرى يتم اختيار الهدف من الاهداف المضافة سابقاً واختيار المهام التي تساعد على تحقيق هذا الهدف .
- اضافة علامة الانتهاء (Completed mark) على المهمة عند الانتهاء من تنفيذها .
- اختيار مهمة وحذفها من جدول خطوات الهدف .
- تسجيل الخروج .
- النظام يحتفظ بعملية تسجيل الدخول ففي حال عدم تسجيل المستخدم للخروج , يحتفظ النظام بالبيانات التي اضافها سابقاً بدون تسجيل الدخول في كل مرة .

4 . لقطات الشاشة (Sample screenshots) :

1 . واجهة تسجيل الدخول (Login Form) : يقوم المستخدم بادخال بياناته للمرة الاولى ليتمكن من الدخول الى النظام , بهدف الخصوصية وحفظ بيانات كل مستخدم على حدى .

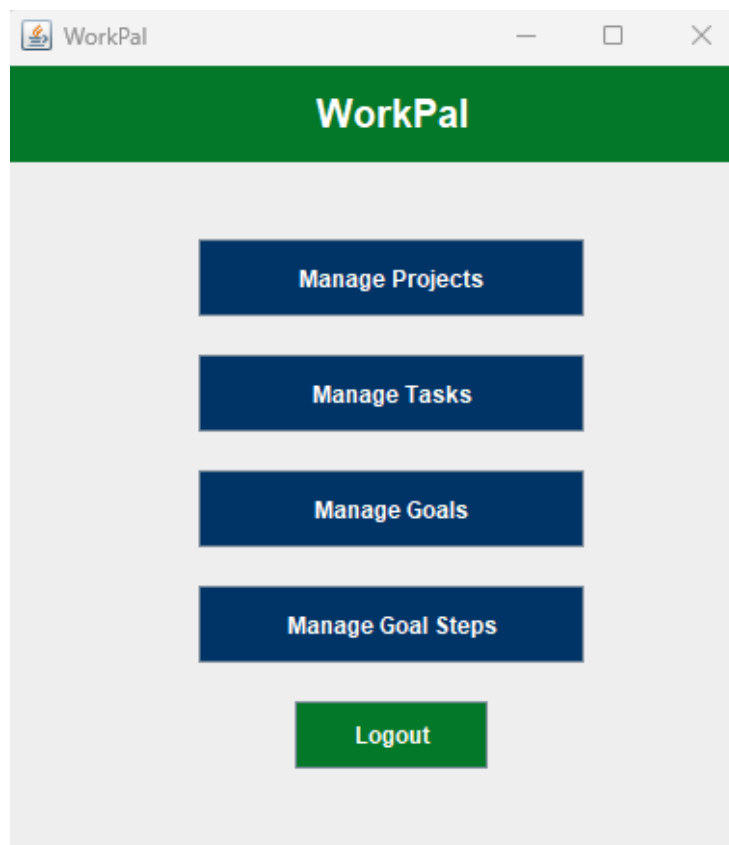
The screenshot shows a web browser window titled 'Login'. The page has a dark blue header with the text 'Login form'. Below the header, there are three input fields: 'Username:' with the value 'Saja', 'Password:' with masked characters '.....', and 'Email:' with the value 'sajabenmussa@email.com'. At the bottom of the form is a green button labeled 'Confirm'.

عملية ادخال بيانات المستخدم

This screenshot shows the same 'Login form' as the previous one, but with an error message overlay. The error message is in a white box with a red 'X' icon and the text 'Error Please fill all fields!'. It has an 'OK' button. The 'Confirm' button is still visible at the bottom.

رسالة تنبيه في حالة عدم تسجيل الدخول بشكل صحيح ..

2 . **الواجهة الرئيسية (Main Form)** : بعد عملية تسجيل الدخول ينتقل النظام الى الواجهة الرئيسية لادخال المشاريع والمهام و الاهداف .



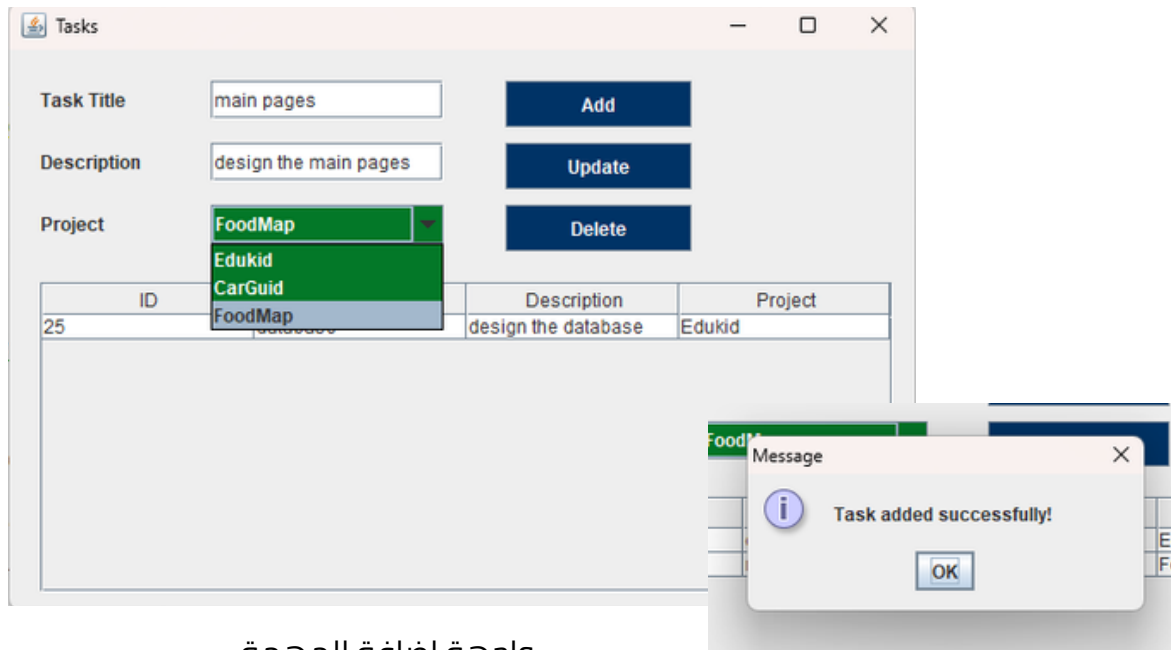
الواجهة الرئيسية

3 . **واجهة المشروع (Project Form)** : يقوم المستخدم باضافة المشروع الذي يريد العمل عليه وتحديد وصف له وتاريخ البداية و النهاية .

عملية ادراج مشروع

رسالة تأكيد الحفظ

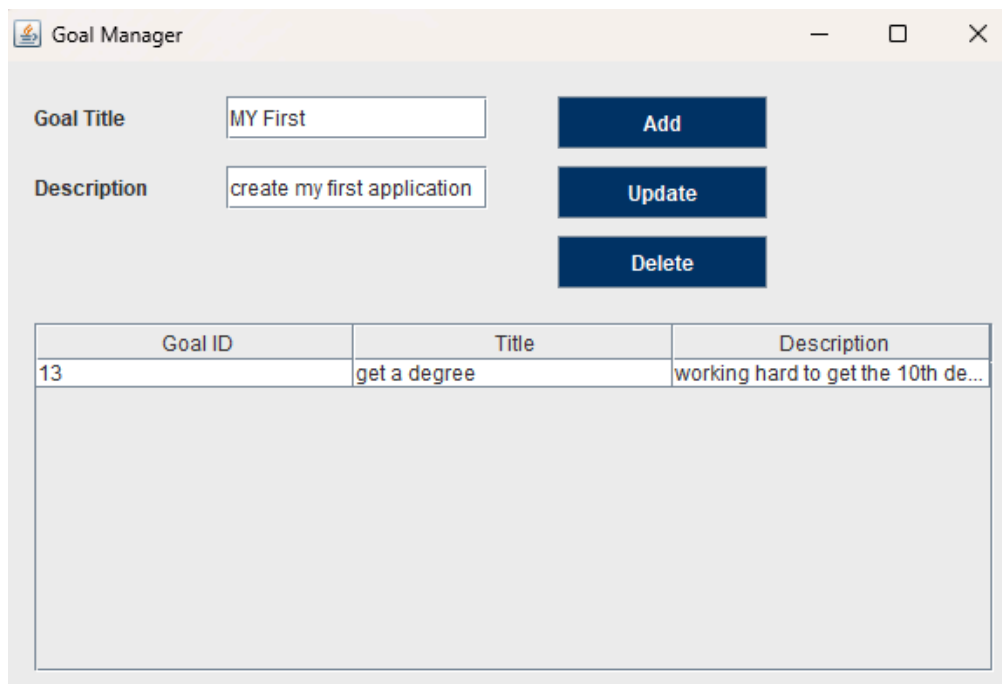
4 . **واجهة المهام (Tasks Form)** : بعد اضافة المشروع يقوم المستخدم باضافة مهام للمشروع , يتم عرض جميع المشاريع التي ادخلها المستخدم سابقاً ويختار منها , ثم تعرض المهام في الجدول .



واجهة اضافة المهمة

تأكيد الاضافة

5 . **واجهة الهدف (Goal Form)** : يحدد المستخدم الهدف الذي يسعى لتحقيقه , ويتم عرضه في الجدول .



واجهة اضافة الهدف

6 . واجهة خطوات الهدف (Goal Steps Form) : يقوم المستخدم باختيار هدف من الاهداف المضافة سابقا , ثم يحدد المهام التي ستساعد على تحقيق هذا الهدف كخطوات للهدف , ثم تعرض في الجدول وعند الانتهاء من تنفيذها يحدد المستخدم المهمة على انها مكتملة بواسطة CheckBox .

Step ID	Goal ID
33	14
34	14
35	14
36	14

واجهة اضافة خطوات للهدف

Step ID	Goal ID	Description	Completed
33	14	CRUD	<input checked="" type="checkbox"/>
34	14	details	<input checked="" type="checkbox"/>
35	14	database	<input type="checkbox"/>
36	14	search	<input type="checkbox"/>
37	14	main pages	<input checked="" type="checkbox"/>

انشاء mark على المهام المستوفية

5 . كيفية تنفيذ كل مطلوب في المشروع (Implementation of Course Topics) :

- **Exception Handling** : تم استخدام آلية معالجة الاستثناءات (try & catch) أثناء الاتصال بقاعدة البيانات وفي عمليات الإضافة والتعديل والحذف , وذلك لمنع توقف البرنامج عند حدوث أخطاء , مع إظهار رسائل تنبيه للمستخدم .
- **Collections Framework** : تم استخدام ArrayList لتخزين البيانات التي يتم جلبها من قاعدة البيانات مثل المشاريع، المهام، والأهداف، ثم تمريرها إلى الواجهات الرسومية لعرضها داخل الجداول للمستخدم .
- **JDBC**: تم استخدام JDBC في كلاس **DBConnection** وكلاسات **DAO** لتنفيذ عمليات **CRUD** على قاعدة البيانات، مثل إضافة مشروع جديد، جلب المهام، تعديل البيانات، وحذف السجلات.
- **Swing & Event Handling** : تم بناء واجهات المستخدم باستخدام Java Swing مثل **ProjectForm , TaskForm , MainForm , GoalForm** وغيرها مع استخدام Event Handling لتنفيذ العمليات عند الضغط على الأزرار **Add , Update , Delete** .
- **I/O Streams** : تم استخدام I/O Streams لقراءة أو كتابة البيانات في ملفات خارجية، مثل حفظ أو تصدير بعض بيانات النظام، مما يسمح بالاحتفاظ بالبيانات أو مشاركتها
- **Multithreading** : تم استخدام Multithreading عند تحميل البيانات من قاعدة البيانات إلى الجداول بحيث تتم العملية في الخلفية دون التأثير على استجابة واجهة المستخدم , مما يمنع تجمد التطبيق أثناء التنفيذ.

6 . توزيع المهام (Task distribution) :

• بشار غوار :

1. بناء كلاسات **TaskDAO , TaskForm , GoalStepForm , GoalForm**
2. ربط كلاسات الـ forms مع DAO
3. إنشاء كلاس **MainDashboard** لعرض بقية النماذج
4. تنفيذ نظام تسجيل الدخول والخروج وإدارة الجلسات (**SessionManager**)
5. إنشاء مخطط الكائنات العلائقية للنظام **ERD**

• سجي بن موسى :

1. تصميم وانشاء قاعدة البيانات MySQL وربطها بالنظام
2. انشاء المستودع على الـ GitHub واستضافة الفريق ورفع الـ packages
3. تصميم الشكل المبدئي لكل واجهة من واجهات المستخدم
4. انشاء كلاسات , RegisterForm , DBConnection , ProjectDAO , ProjectForm , GoalDAO
5. اعادة بناء كلاس GoalStepForm لاكتشاف عدم تحقيقه للمطلوب
6. اختبار النظام واكتشاف الازخام
7. اضافة الالوان والتنسيقات لجميع كلاسات النماذج الـ UI
8. اجراء التعديلات وتصحيح المشاكل على جميع الكلاسات لضمان عملها بشكل متكامل

• مشترك :

1. اعداد التوثيق وكتابة التقرير النهائي للمشروع
2. تصحيح الازخام

5. الدروس المستفادة والتحديات (Lessons Learned & Challenges) :

اهم الدروس المستفادة هو أهمية تنظيم هيكل المشروع وتقسيمه إلى طبقات واضحة مثل الواجهة (UI) و الكلاسات الخاصة بالبيانات (Model) ، وطبقة التعامل مع قاعدة البيانات (DAO) ، مما ساعد على سهولة فهم الكود وصيانته. كذلك اكتسبنا خبرة في التعامل مع واجهات Swing وفهم آلية معالجة الأحداث (Event Handling) . بالإضافة إلى ذلك، تعلمنا كيفية استخدام المجموعات (Collections) مثل ArrayList لتنظيم البيانات داخل النظام . وأخيرًا، تعلمنا أهمية العمل الجماعي واستخدام مستودعات GitHub لإدارة التعديلات ومشاركة العمل بين أعضاء الفريق ، مما ساعد على تتبع التغييرات وتجنب اخطاء النسخ واللصق .

ومن التحديات التي واجهتنا مثل تنظيم الكود وربط الطبقات : في البداية واجهنا صعوبة في تنظيم المشروع بين الواجهات المختلفة مما أدى إلى بعض الالتباس عند الربط ، وكما واجهنا بعض المشاكل في التعامل مع GitHub عند دمج الكلاسات مع بعضها .