|  |  |
| --- | --- |
|  | Clean and Go  Garcia, Lemuel Aldwin P.  Mandapat, Gerome A. |

1. Table of Contents

[A Objectives 1](#_Toc527343813)

[B Program Requirements 1](#_Toc527343814)

[C System Requirements 1](#_Toc527343815)

[C.1 Hardware Requirements 1](#_Toc527343816)

[C.1.1 Processor 1](#_Toc527343817)

[C.1.2 Memory 1](#_Toc527343818)

[C.1.3 Storage 1](#_Toc527343819)

[C.2 Software Requirements 1](#_Toc527343820)

[C.2.1 Operating System 1](#_Toc527343821)

[C.2.2 Software Requirements 2](#_Toc527343822)

[D Scope and Features 3](#_Toc527343823)

[D.1 Login 3](#_Toc527343824)

[D.2 View Laundry Requests 3](#_Toc527343825)

[D.3 Add Laundry Requests 3](#_Toc527343826)

[D.4 View Customers Information 3](#_Toc527343827)

[E Limitations 4](#_Toc527343828)

[E.1 No Means of Retrieving Login Credentials if Forgotten 4](#_Toc527343829)

[E.2 No Sign-up form 4](#_Toc527343830)

[E.3 Class Diagram 5](#_Toc527343831)

[F Flowchart 7](#_Toc527343832)

[F.1 Program Flowchart 7](#_Toc527343833)

[G User Manual 8](#_Toc527343834)

[H Analysis 15](#_Toc527343835)

[I Conclusion 15](#_Toc527343836)

List of Figures

[Figure F.1.a: Class Diagram 5](#_Toc527395129)

[Figure G.1.a: Class Diagram 6](#_Toc527395130)

[Figure F.1.a: Program Flowchart 7](#_Toc527395131)

# Objectives

* To use SQLite as a database for a program
* To create a program using python programming language
* To apply the 3-tier architecture in creating a program

# Program Requirements

* The program should have a login interface wherein only registered users may access it.
* The program should have a logout function so that users won’t have to close the program to login a different user.
* The user should be able to view all ongoing and completed laundry requests.
* The user should be able to add a laundry request.
* The user should be able to add a new customer profile.
* The user should be able to view all customer profiles
* The user should be able to update the status of the laundry request.
* The user can indicate whether the customer would pay in full or in installment.
* The user should be able to update the balance left of the owner for the requested laundry services.
* The types of services to be done on the laundry should be specified.
* The program should calculate the total cost of the laundry request depending on the weight and the services to be done to the laundry.

# System Requirements

## Hardware Requirements

### Processor

Processor Core Count : 4

### Memory

Memory Capacity : 8GB

### Storage

Disk Size : 890GB

## Software Requirements

### Operating System

Operating System Name : Windows 8.1

Architecture Type : 64-bit

Edition : Single Language

### Software Requirements

Software Application # 1 : Microsoft Visual Studio 2017 Community Edition

Software Application # 2 : Microsoft .NET Framework 4.6

# Scope and Features

## Login

The program would require the user to enter a username and password. The password should match the registered password of the given username.

## View Laundry Requests

The program allows the user to view the laundry requests, either ongoing or completed. The user can change status of the request through the ongoing requests table, once the status is changed, the request would be moved onto the completed orders table and removed from the ongoing requests table.

## Add Laundry Requests

The program allows the user to add laundry requests. The requests include the owner name, the weight, date received, claim mode, services, payment method either installment or full payment.

## View Customers Information

The program allows the user to view all the customer related information stored. This includes the customer name, contact number, email, and date of last order.

# Limitations

## No Means of Retrieving Login Credentials if Forgotten

If ever that a user forgets his/her password, currently, there is no function that would allow the user to change the password incase he forgets it.

## No Sign-up form

No sign-up form for new users. Only existing accounts are usable. The sign-up form was not made due to time constraints.

# Class Diagram

## Class Diagram

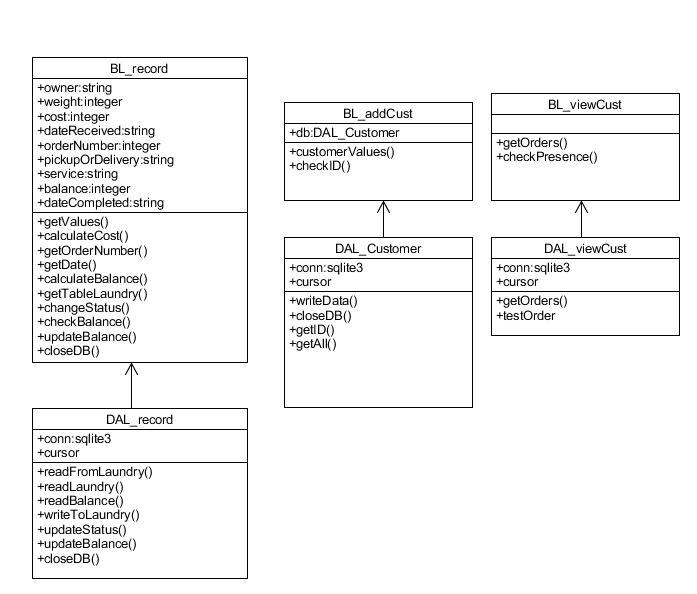


Figure F.1.a: Class Diagram

# Entity Relationship Diagram

## Entity Relationship Diagram

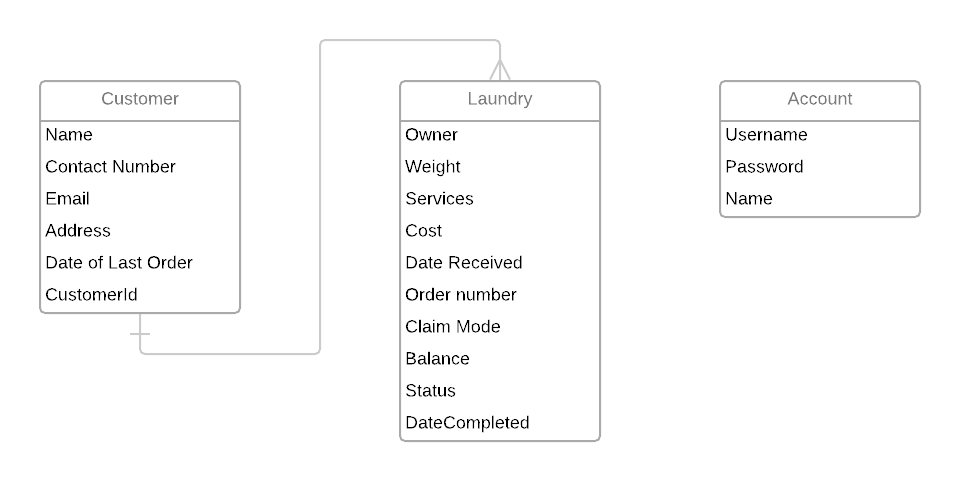


Figure G.1.a: Entity Relationship Diagram

# Flowchart

## Program Flowchart

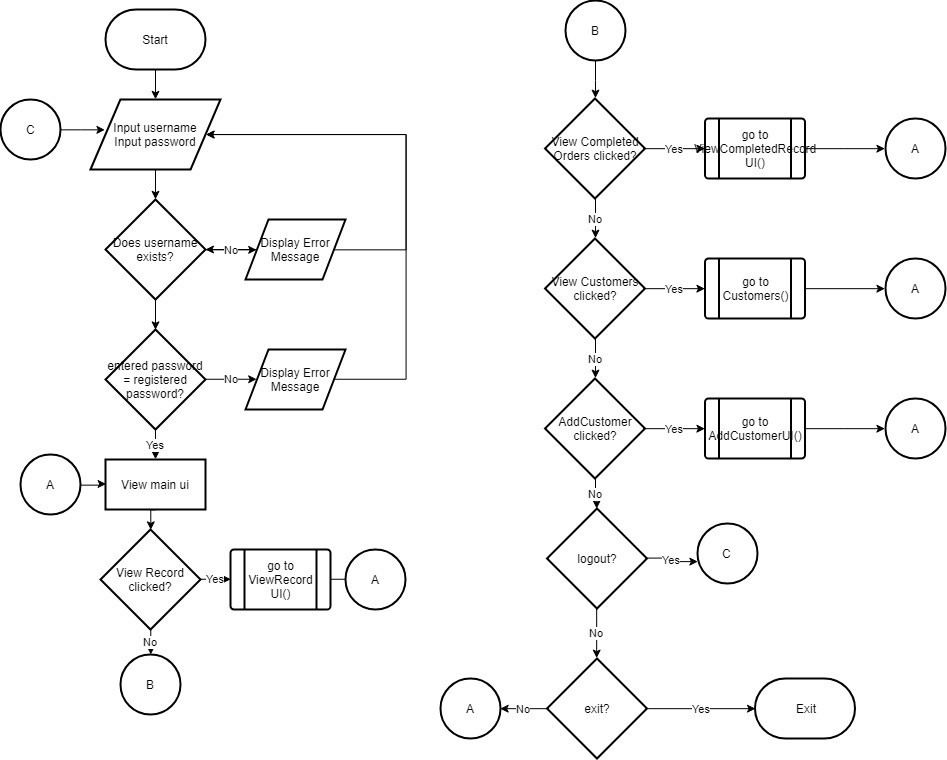
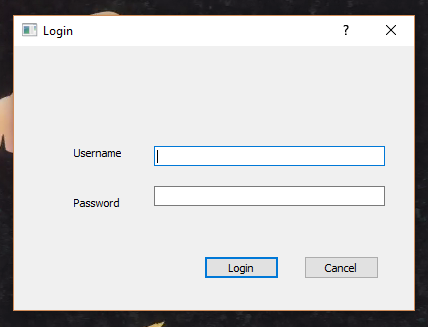


Figure F.1.a: Program Flowchart

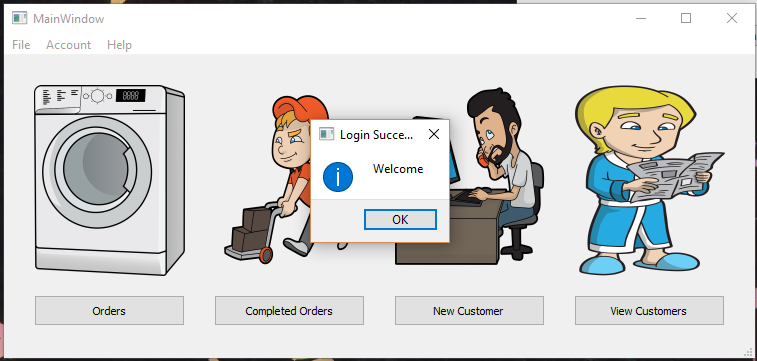
The figure shows the overall flow of the program. The user would need to login, inputting his/her registered account username and password. After successfully logging in, the user would be seeing the main interface. There, the user can view the ongoing laundry requests, completed laundry requests, and customer information. The user also has the option to logout from the menu bar under Account menu or exit the program from the File menu.

# User Manual

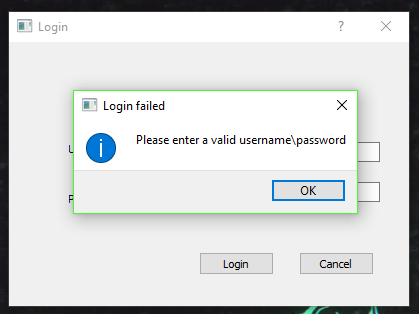
1. The user needs to login to his/her respective account. If the user has no registered account yet, he/she needs to register for one.



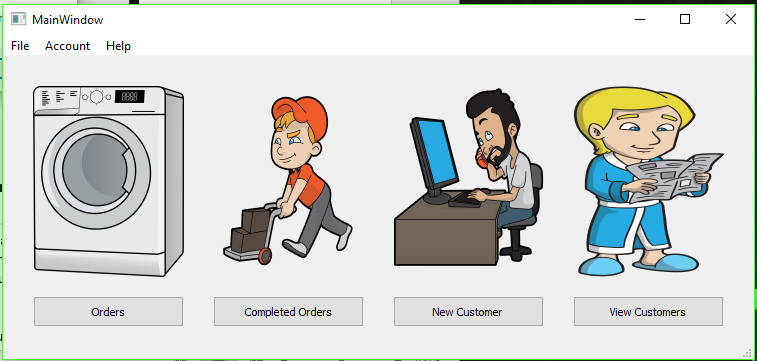
If the user enters the correct set of username and password, a message would be prompted upon successful login.



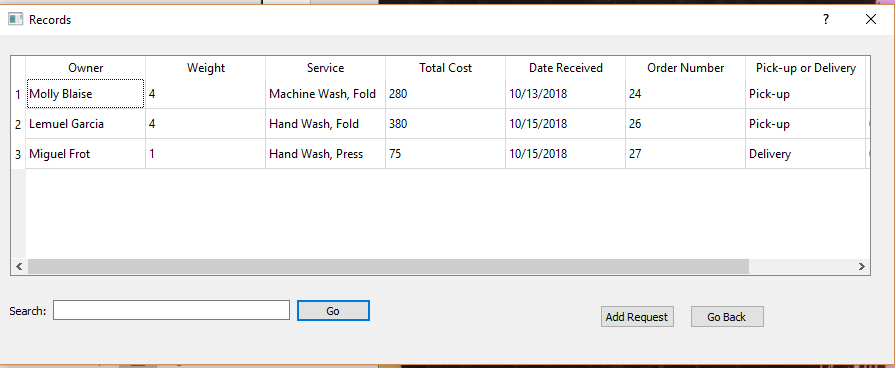
If the user enters a wrong username or password, an error message would be shown.



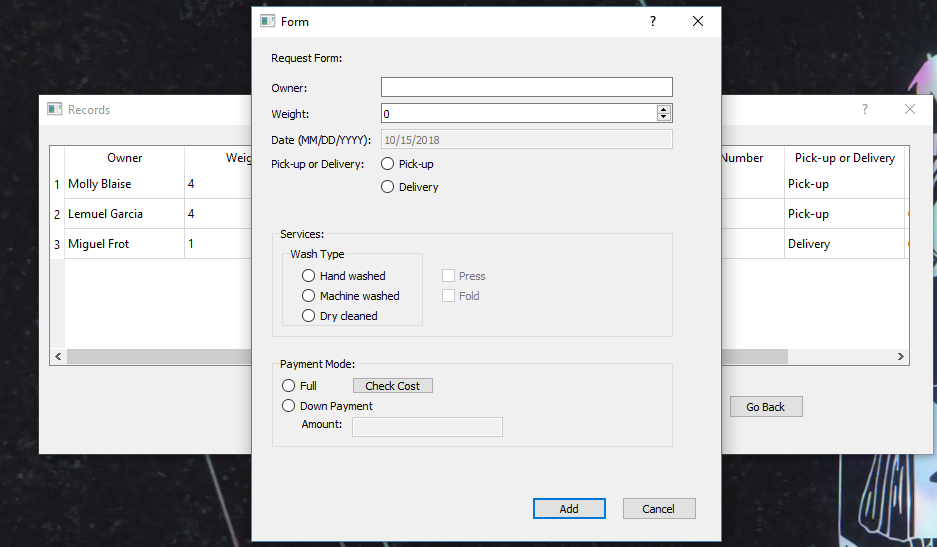
1. After logging in the user would be able to view all the ongoing requests, view all finished request, add a new customer profile, and view all customer profiles. In the menu bar of the main interface, there are the File and Account menu. In the File menu, the user may exit the application, and under the Account menu, the user may logout and then login into a different account.



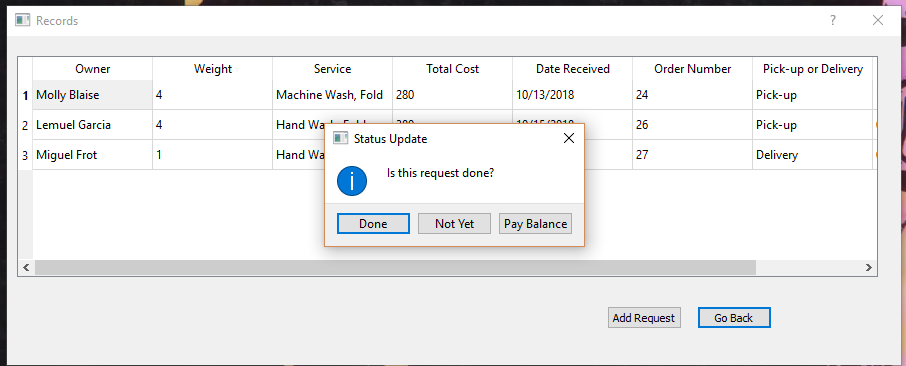
1. If the user selects the Order button, he/she could view all the ongoing requests, add a request on the list, edit its status, or update the remaining balance of the customer for a certain order.



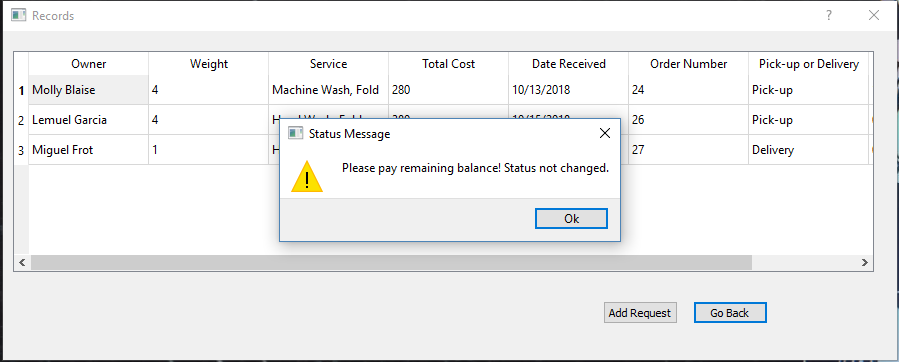
To add a request on the list the user should click the “Add” button

  
 Clicking the “Back” button would return the user to the main interface

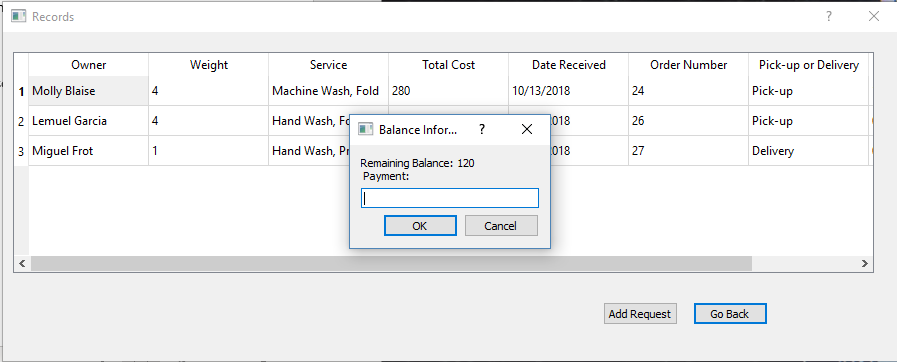
1. The user can also change the status of the laundry request from ongoing to done by clicking on any of the item on the row of the request the user wants to change the status. The user would be prompted if he wants to change the status to Done, pay remaining balance, or cancel action.



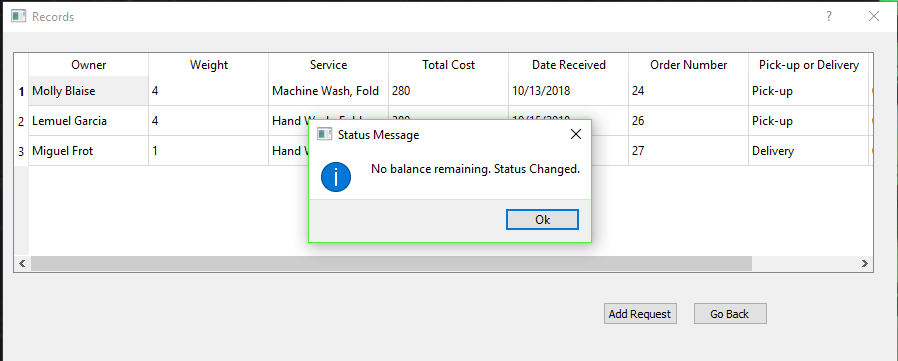
If the status change “Done” is selected, the program would check if the owner still has a remaining balance, if there is then the program would prompt the user to pay remaining balance first.

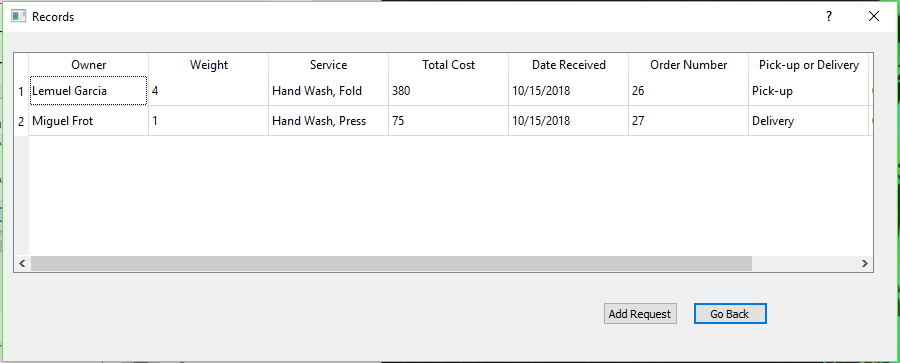


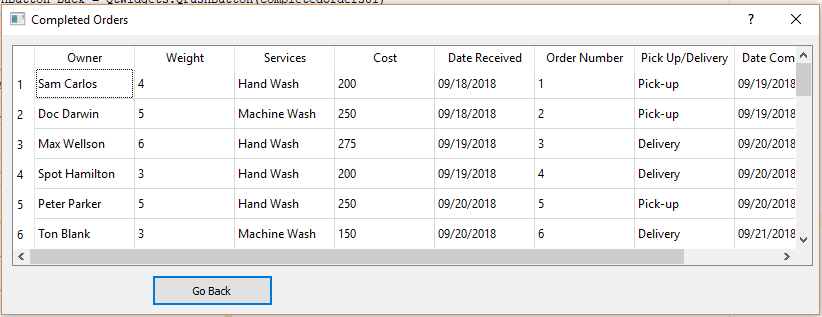
If pay balance is selected, the user would be prompted how much would the customer be paying, and the message would also display the remaining balance of the owner.

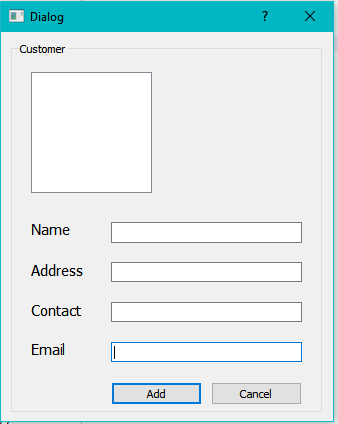


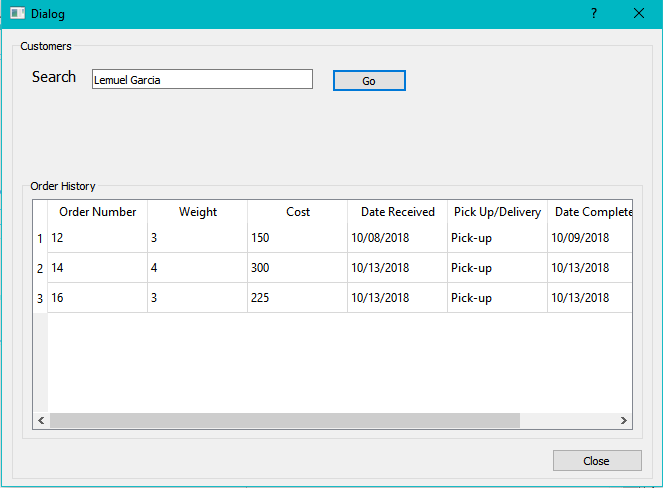
If no balance remains then status would be successfully changed, then it would be removed from the table.





1. Going to the Completed Orders interface would show the user the list of completed orders
2. The add Customer form, which allows user to add a new Customer to the database.



6. The View customers form, which allows users to view existing users and their order history. If a customer does not exist in the database, a dialog will appear.

# Analysis

The program gives user an efficient means of storing laundry orders, with the important information such as the customer, contact details and others. This allows the users to easily access any information needed through the use of the different forms in the application.

# Conclusion

The program created is able to store all the laundry requests of the shop. Using the program the user can also view all the ongoing and accomplished laundry requests, add a customer, or view all customer information. The three-tier architecture was successfully applied in the program wherein it is divided into three parts, the graphical user interface, the business logic, and the data access layer wherein the graphical user interface may only communicate with the database through the business logic layer.

