Programming 2 Final Project

Robert

30018755

Programming 2 Final Project

Final Project

# Table of Contents

[Table of Contents i](#_Toc56461600)

[Introduction 1](#_Toc56461601)

[What does the Program do? 1](#_Toc56461602)

[What development Methodology? 1](#_Toc56461603)

[Analysis 2](#_Toc56461604)

[What Data is required? 2](#_Toc56461605)

[What processes need to be performed while the program is running? 2](#_Toc56461606)

[What output is generated when the program closes? 2](#_Toc56461607)

[Project Plan 3](#_Toc56461608)

[What tasks were completed to develop this project? 3](#_Toc56461609)

[List the job title for each task, what order were the tasks in? 3](#_Toc56461610)

[What physical resources were required to complete this project? 3](#_Toc56461611)

[Draw a Gantt char to show the above info. 3](#_Toc56461612)

[Algorithm Design 4](#_Toc56461613)

[Test Data and Evidence 5](#_Toc56461614)

[Pictures from Testing. 10](#_Toc56461615)

[Recommendations and Improvements 18](#_Toc56461616)

## Introduction

### What does the Program do?

When the program first open it loads different binary files into arrays and populates the list. The user is able to add new customers, drones and transactions to the arrays that then get saved into binary files when the program closes.

When a value from the list box is single clicked the text boxes will populate with the appropriate values. If any of the buttons are pressed without all the appropriate text boxes filled with some values then an error message will display.

The Drone Add button will save the current values of the drone class to an Array.

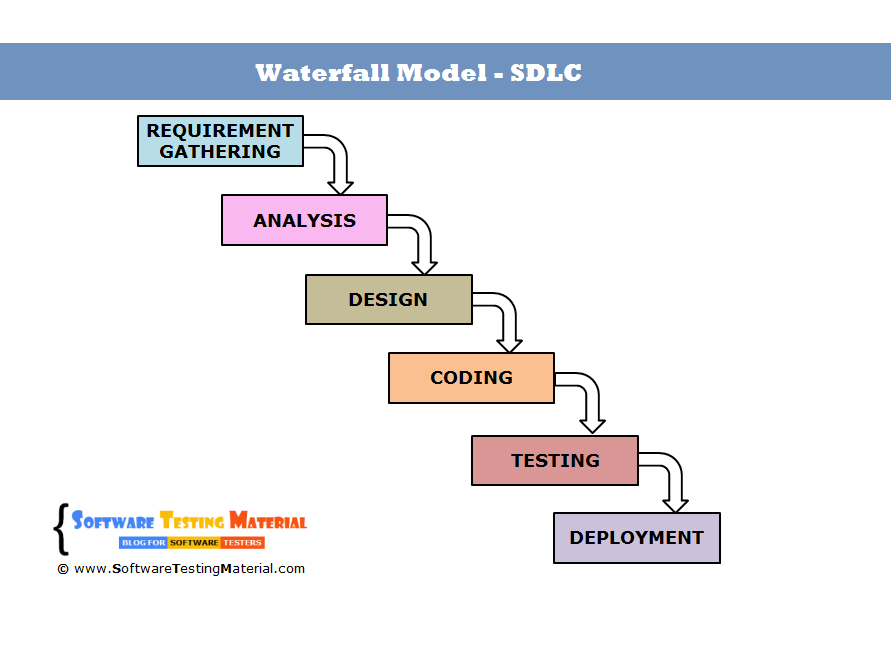
The Customer Add Button will save the current values of the customer to an Array. There will also be a message box displaying allowing the user the option of creating a default customer to save.

The Transaction Button will save the current values to the Transaction Array. The user can only type in the Transaction ID. Before the Transaction ID button is clicked the User must use the list boxes of the Drone and Customer class to put the applicable values into to transaction ID section.

Double clicking the Drone Serial Number text box will clear all the drone boxes and allow the user to add a new transaction. Double clicking the Customer ID text box will clear all the Customer Text boxes to allow the user to add a new customer. Double Clicking the Transaction ID will clear all the transaction text boxes and allow the user to add a new Transaction.

### What development Methodology?

The Methodology that I used was the Waterfall method. This is a good method for a smaller project because each stage are so much smaller the risks of changes or massive errors are reduced. It is also easier to follow compared to the other methodologies.



(What Is SDLC Waterfall Model?, 2020)

## Analysis

### What Data is required?

The data required for the program are 3 binary files (if the 3 binary files are not in the correct locations 3 empty binary files will be created). All the data types produced by the text boxes are strings. All the appropriate text boxes must be filled before the Add buttons are pressed.

### What processes need to be performed while the program is running?

While the program is running all the appropriate text boxes must be filled before clicking the add buttons. All Message Box instructions must be followed. Single clicking the list boxes will fill the appropriate text boxes and double clicking the leading text boxes (Drone Serial Number, Customer ID, Transaction ID) will clear all the text boxes and allow for new data.

### What output is generated when the program closes?

The output produced by the program are 3 new binary files or changes to 3 binary files that were already present before the program was initiated. (drones.dat / customer.dat/ transaction.dat).

## Project Plan

### What tasks were completed to develop this project?

* The client requirements were gathered. The resources were gathered and an overall time scale was documented and planned out for the entire project.
* The requirements are analyzed to make sure it Is feasible and a sing off from the client is provided. All the tasks will be allocated to the staff and the coding can begin.
* The software was the developed and coded along side the documentation that will enable the testing later on.
* All the testing is completed by the staff and the product is presented to the client in a usable state.
* Once the client signs off on the product it is then implemented.

### List the job title for each task, what order were the tasks in?

1. Client requirements are documented.
2. Feasibility report produced.
3. Requirement analysis.
4. Staff allocation.
5. Software development.
6. Testing Documentation.
7. Testing.
8. Client Sign off.
9. Project handed over and implementation.

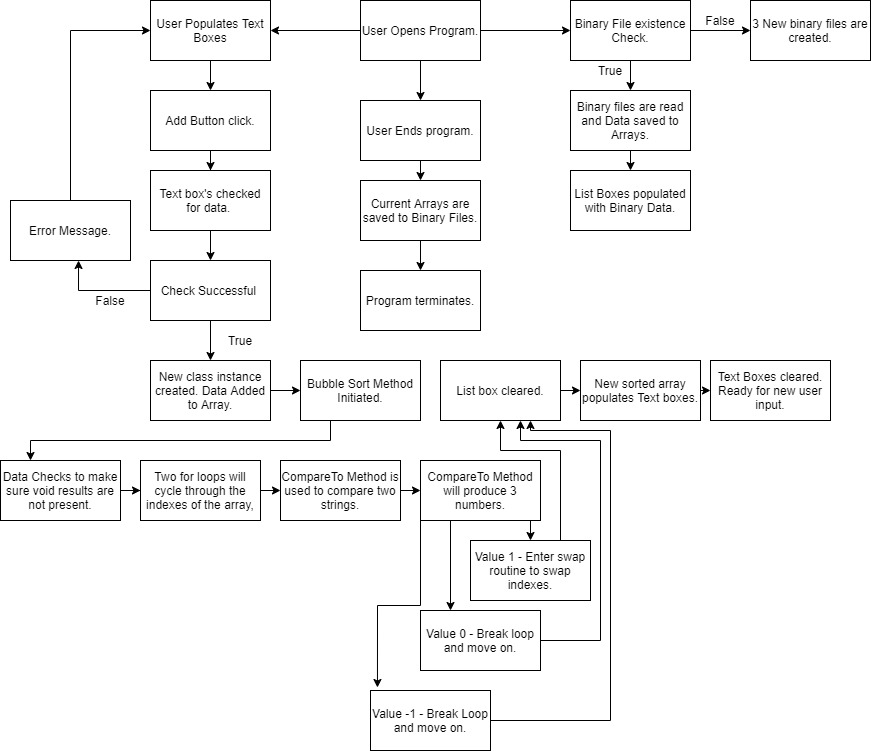
### What physical resources were required to complete this project?

The physical resources used for this project are computers for each staff member. Printers for the documentation. Phone lines for communication.

The Software used for this project are Microsoft Word (Documentation), OutLook (Email communication), Visual Studios (Coding).

### Draw a Gantt char to show the above info.

## Algorithm Design

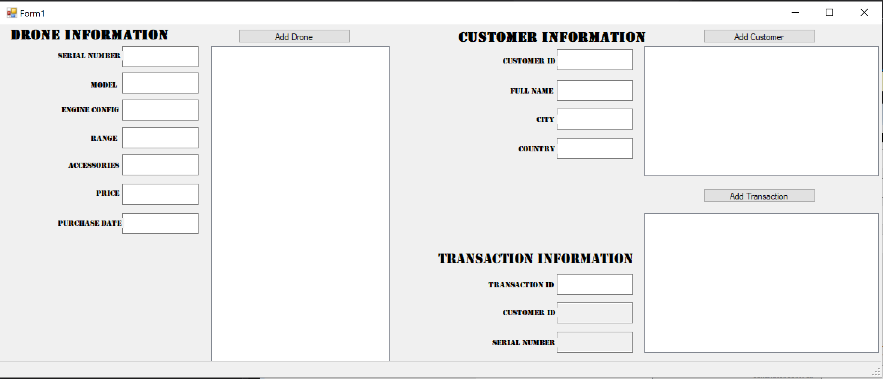


## Test Data and Evidence

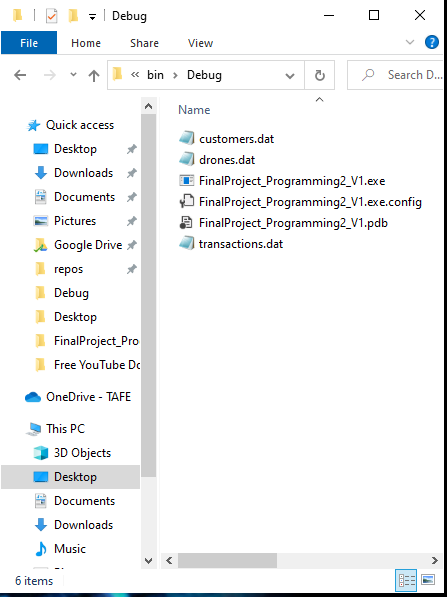
The test data will be displayed in a table format for easy viewing. Many different scenarios are presented and tested. Screenshots will be supplied where applicable. The expected outcome as well as the actual outcome will also be presented.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test Number** | **Expected Outcome** | **Actual Outcome** | **Pass / Fail** | **Evidence.** |
| 1. Loading Up Form. | Form will present itself empty. | Form presented itself empty ready for use. | Pass. | Capture 1. |
| 2. Creating new Drone binary file. On Start up. | New Drone File will appear in Binary File of Program. | Drone.Dat file appears in Binary File. | Pass. | Capture 2. |
| 3. Creating new Customer File. On Start up. | New Drone File will appear in Binary File of Program. | Customer.Dat file appears in Binary File. | Pass. | Capture 2. |
| 4. Creating new Transaction. On Start up. | New Drone File will appear in Binary File of Program. | Customer.Dat file appears in Binary File. | Pass. | Capture 2. |
| 5. Clicking Drone add Button with zero inputs. | Error Messaging requesting Data. | Error Message Requesting data into the text boxes. | Pass. | Capture 3. |
| 6. Clicking Customer Add Button with zero inputs | Request asking to make a default customer. | Dialogue box displays requesting to make a default customer. | Pass | Capture 4. |
| 6b. Clicking Yes on new window. | Multiple User notifications confirming default user. | Multiple confirmations of default user + Customer text boxes populating with default customer. | Pass | Capture 5. |
| 6c. Clicking No on new window. | User notification that no further action will be taken. | User notification that no further action will be taken. | Pass | Capture 6. |
| 7. Clicking Transaction Button with zero Inputs. | Error Messaging requesting Data. | Error Message requesting Data. | Pass. | Capture 7. |
| 8. Clicking Drone Button without all Inputs filled (Half). | Error Messaging requesting Data. | Error Message Requesting data. | Pass. | Capture 8. |
| 9.Clicking Customer Button without all inputs filled.(Half) | Error Messaging requesting Data. | Error Message Requesting Data. | Pass. | Capture 9. |
| 1. Clicking Transaction Button without all Inputs filled. (Half) | Error Messaging requesting Data. | Error Message Requesting Data. | Pass. | Capture 10. |
| 11.Clicking Drone Button with all Inputs filled in. | All the data will appear in the appropriate displaybox in alphabetical order. | All the data will appear in the appropriate displaybox in alphabetical order. | Pass. | Capture 11. |
| 12.Clicking Customer Button with all Inputs Filled in. | All the data will appear in the appropriate displaybox in alphabetical order. | All the data will appear in the appropriate displaybox in alphabetical order. | Pass. | Capture12. |
| 13.Clicking Transaction Button with all inputs filled in. | All the data will appear in the appropriate displaybox in alphabetical order. | All the data will appear in the appropriate displaybox in alphabetical order. | Pass | Capture 13. |
| 14.Double Left Clicking Transaction Serial Number text box with all inputs filled in. | All the appropriate Text boxes will clear themselves of data. | All the appropriate Text boxes will clear themselves of data. | Pass. | Capture 14. |
| 15. Double Left Clicking Customer ID Text Box with all inputs filled in. | All the appropriate Text boxes will clear themselves of data. | All the appropriate Text boxes will clear themselves of data. | Fail. First Text box will not clear. | Capture 15. |
| 16. Double Left Clicking Transaction ID with all boxes filled in. | All the appropriate Text boxes will clear themselves of data. | All the appropriate Text boxes will clear themselves of data. | Pass. | Capture 16. |
| 17. Clicking one Index inside Drone Display Box. | All the data associated with that entry will be displayed in the appropriate display box. | All the data associated with that entry will be displayed in the appropriate display box. | Pass. | Capture 17. |
| 18. Clicking One Index of Customer Display Box. | All the data associated with that entry will be displayed in the appropriate display box. | All the data associated with that entry will be displayed in the appropriate display box. | Pass. | Capture 18. |
| 19. Clicking One Index of Transaction Display Box. | All the data associated with that entry will be displayed in the appropriate text box. | All the data associated with that entry will be displayed in the appropriate display box. | Pass. | Capture 19. |
| 20. Adding multiple entries into Drone Display Box (Correct Order). | The ID, Engine Config and Price will be displayed in the display Box. All the data will be stored in the array. | All appropriate data was stored and displayed in List Box. | Pass. | Capture 17. |
| 21. Adding multiple entries into Customer Display Box (Correct Order) | All the appropriate data will be stored into the display box in order. | All appropriate data was stored and displayed in List Box. | Pass. | Capture 18. |
| 22. Adding Multiple Entries into Transaction Display Box (Correct Order). | All the appropriate data will be stored into the display box in order. | All appropriate data was stored and displayed in List Box. | Pass. | Capture 19. |
| 23.Adding Multiple entries into Drone Display Box (Wrong Order). | All the appropriate data will be stored into the display box in order. | All appropriate data was stored and displayed in List Box. It is also sorted alphabetically. | Pass. | Capture 17. |
| 24. Adding Multiple Entries into Customer Display Box (Wrong Order). | All the appropriate data will be stored into the display box in order. | All appropriate data was stored and displayed in List Box. It is also sorted alphabetically. | Pass. | Capture 18. |
| 25. Adding Multiple Entries into Transaction Display Box (Wrong Order). | All the appropriate data will be stored into the display box in order. | All appropriate data was stored and displayed in List Box. It is also sorted alphabetically. | Pass. | Capture 19. |
| 26.Right Clicking empty Customer ID box. | Error message will be provided informing the user to input data into the text boxes. | Nothing. | Failed. No Error Message Displayed. | Capture 20. |
| 27. Right Clicking Filled Customer ID Box (other entries Missing) | Error message will be provided informing the user to input data into the text boxes. | Nothing. | Failed. No Error Message Displayed. | Capture 21. |
| 28. Right Clicking Filled Customer ID Box (Data not currently in Array) | Error message will display saying entry not found. | Error message will display saying entry not found. | Pass. | Capture 22. |
| 29. Right Clicking Filled Customer ID Box (Data present in Array). | The entry will be found and a message will be displayed with the information. | Found Entry. Message Displayed. | Pass. | Capture 23. |
| 30. Closing Application form Via the X button in top right corner. | Message informing user that all the current files are saved. | Message informing user that all the current files are saved. | Pass. | Capture 24. |
| 31.Opening program again with all files present (Filled). | All the data stored in the binary files will be loaded into the text boxes and arrays. | All the data stored in the binary files will be loaded into the text boxes and arrays. | Pass. | Capture 25. |
| 32. Opening program again with 2 files present (Filled). | Only the present files will display data. Any files not present will have empty versions created. | Only the present files will display data. Any files not present will have empty versions created. | Pass. | Capture 26. |
| 33. Opening Program again with 1 file Present (Filled). | Only the present files will display data. Any files not present will have empty versions created. | Only the present files will display data. Any files not present will have empty versions created. | Pass. | Capture 27. |
| 34. Opening program again with zero files present. | Only the present files will display data. Any files not present will have empty versions created. | Only the present files will display data. Any files not present will have empty versions created. | Pass. | Capture 28. |
| 35. Opening program again with 3 files present (Empty). | Only the present files will display data. Any files not present will have empty versions created. | Only the present files will display data. Any files not present will have empty versions created. | Pass. | Capture 28. |
| 36. Opening Program again with 2 files present (Empty). | Only the present files will display data. Any files not present will have empty versions created. | Only the present files will display data. Any files not present will have empty versions created. | Pass. | Capture 28. |
| 37. Opening Program again with 1 file present (Empty). | Only the present files will display data. Any files not present will have empty versions created. | Only the present files will display data. Any files not present will have empty versions created. | Pass. | Capture 28. |
| 38. Adding Entries into array past cap. (Drone) | Out of index error message will display. | Program Crashes due to out of index exception. | Fail | Capture 29. |
| 39 Adding Entries into array past cap. (Customer) | Out of index error message will display. | Program Crashes due to out of index exception. | Fail | Capture 29. |
| 40 Adding Entries into array past cap. (Transaction) | Out of index error message will display. | Program Crashes due to out of index exception. | Fail | Capture 29 |

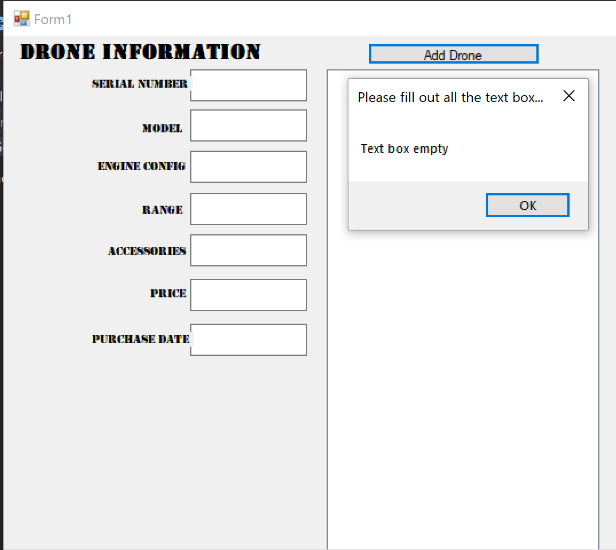
### Pictures from Testing.



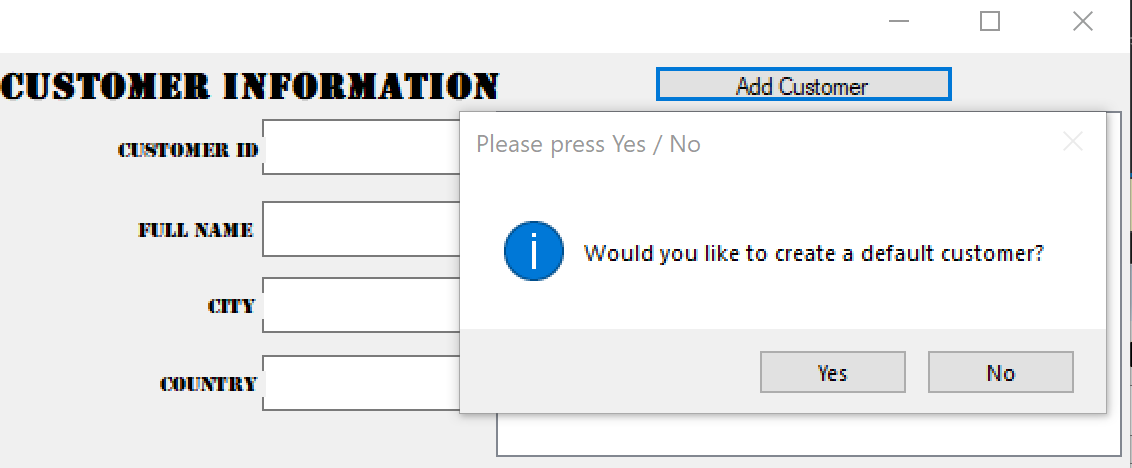
Capture 1.



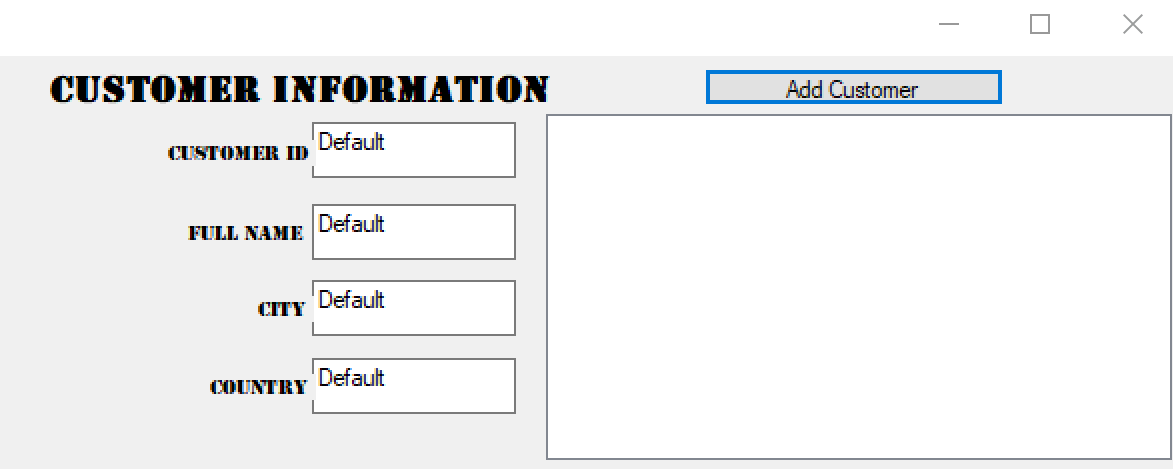
Capture 2.



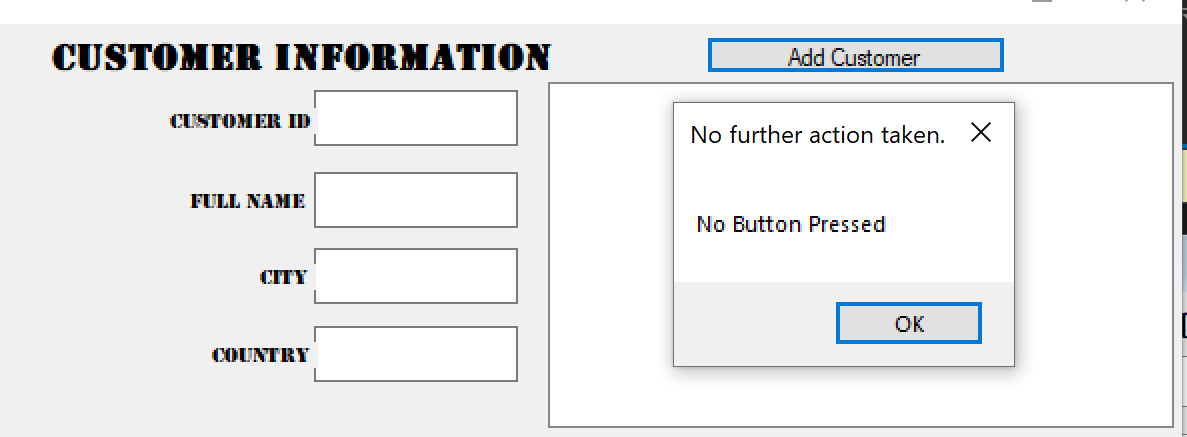
Capture 3.



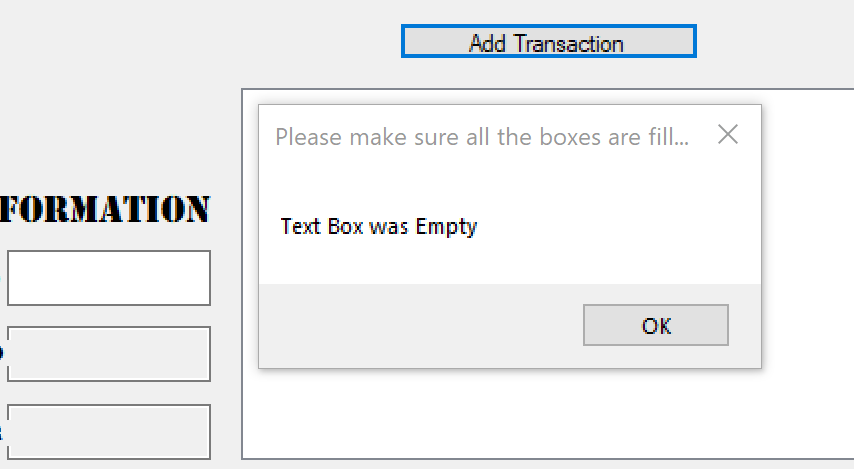
Capture 4.



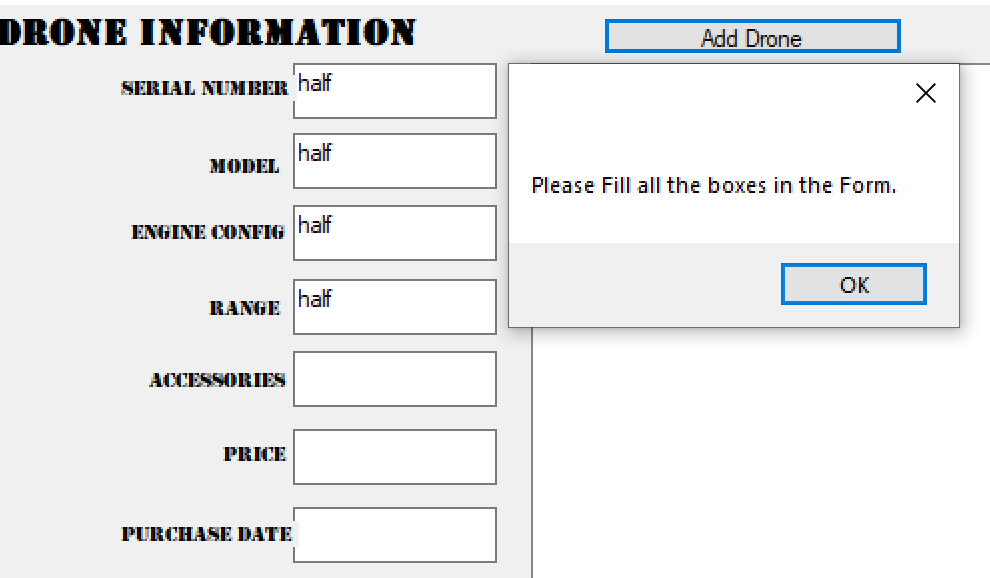
Capture 5.



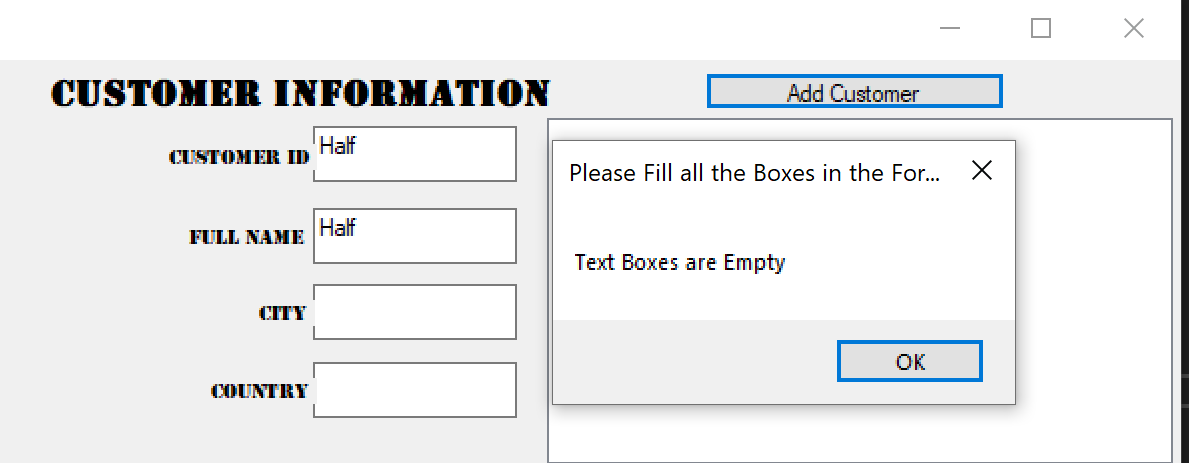
Capture 6.



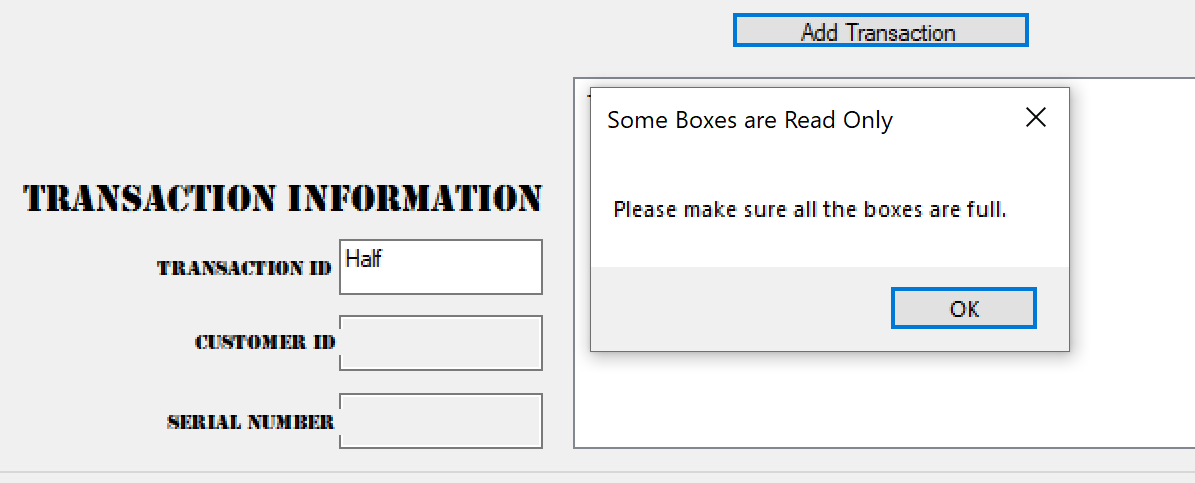
Capture 7.



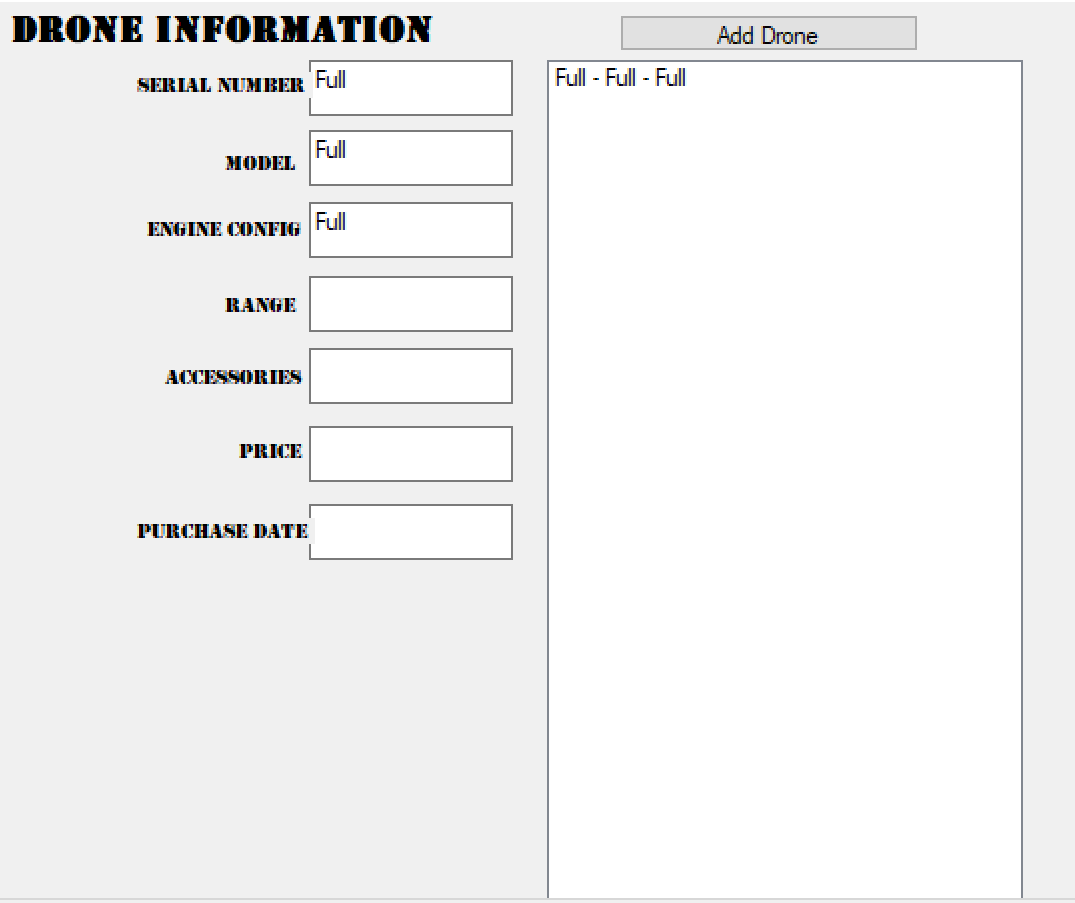
Capture 8.



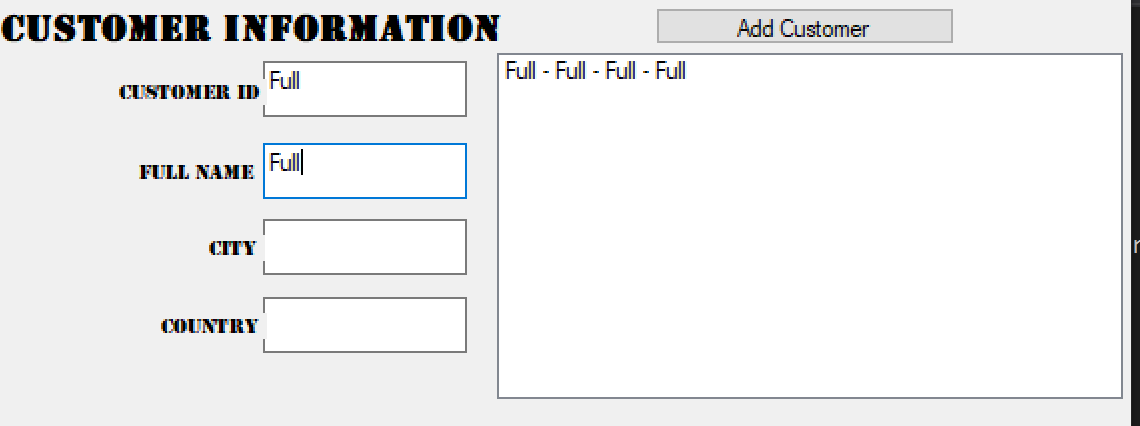
Capture 9.



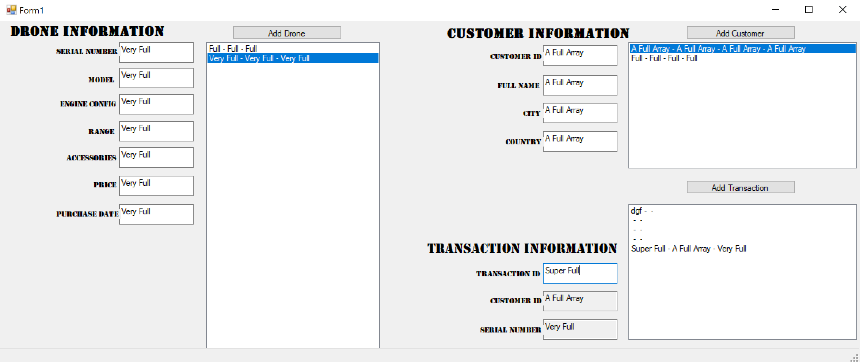
Capture 10.



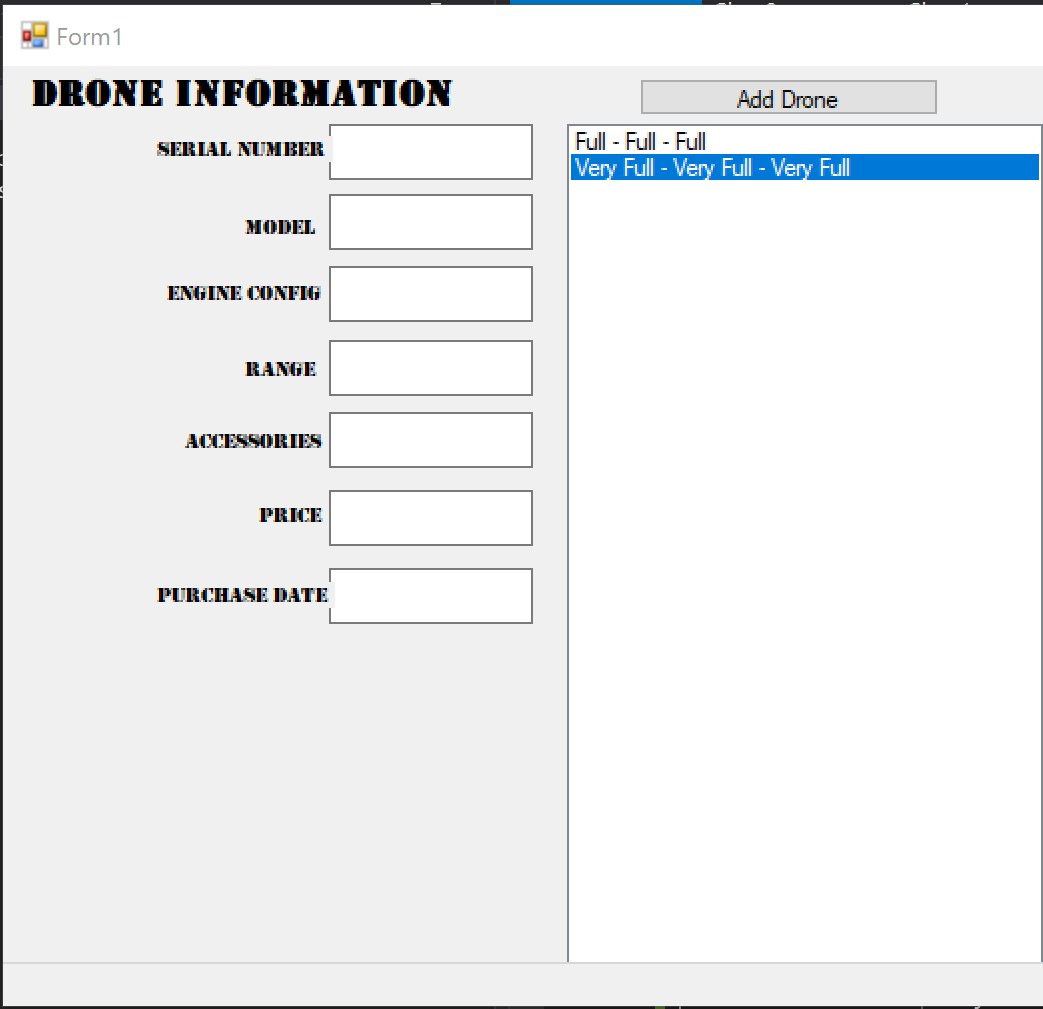
Capture 11.



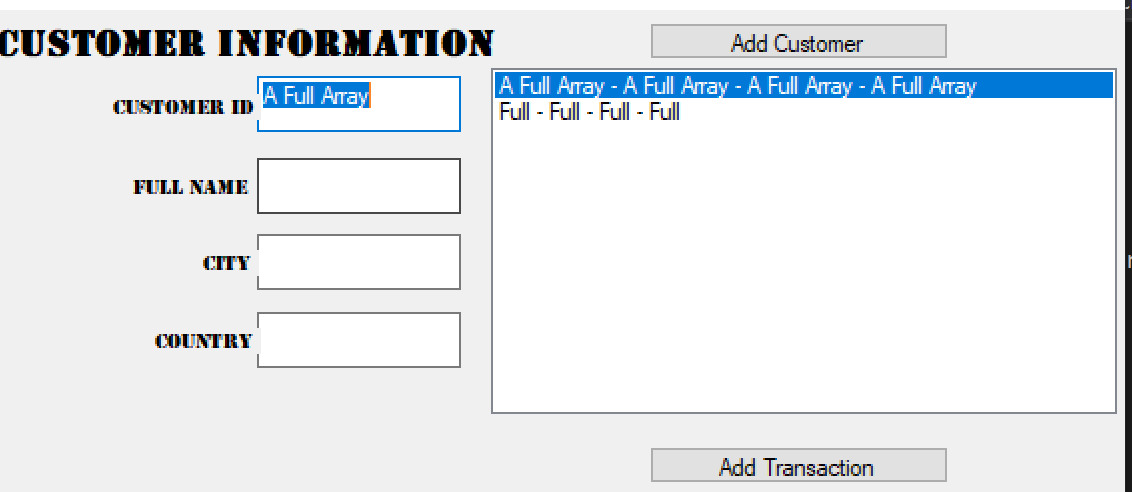
Capture 12.



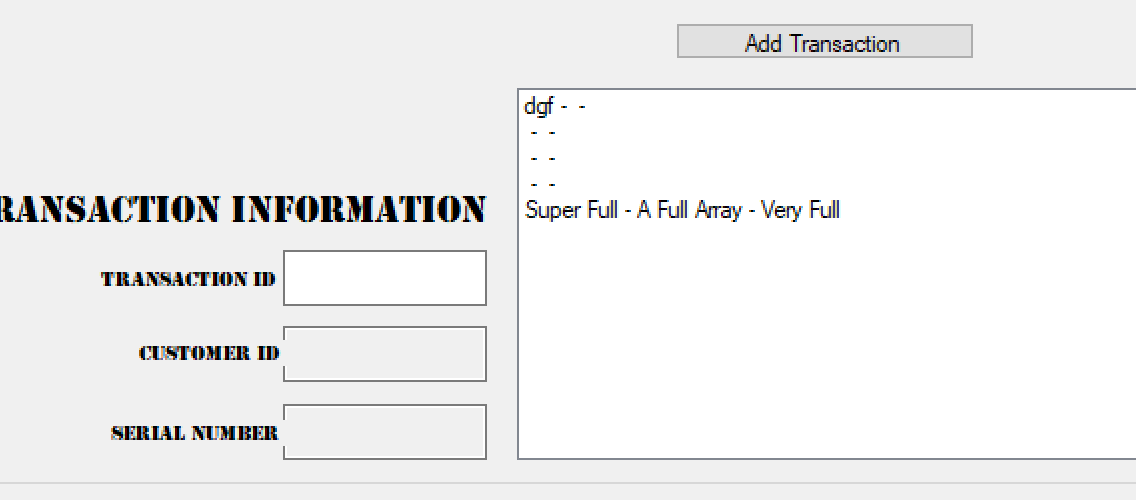
Capture 13.



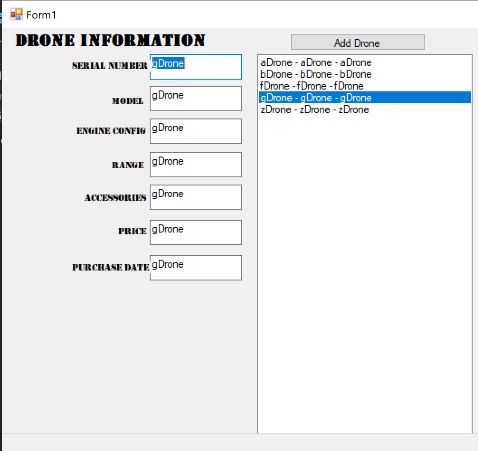
Capture 14.



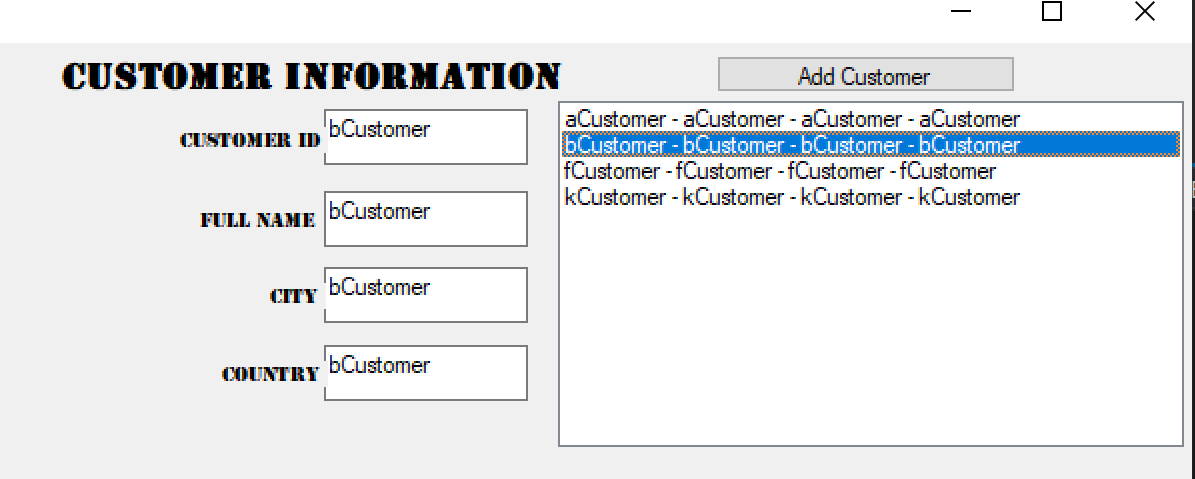
Capture 15.



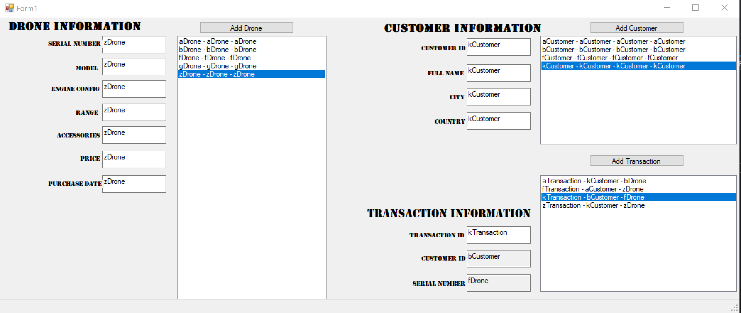
Capture 16.



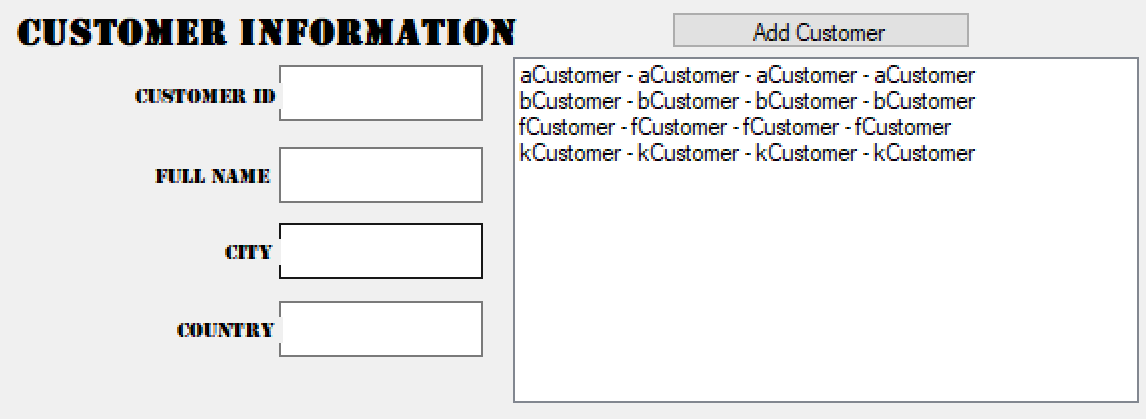
Capture 17.



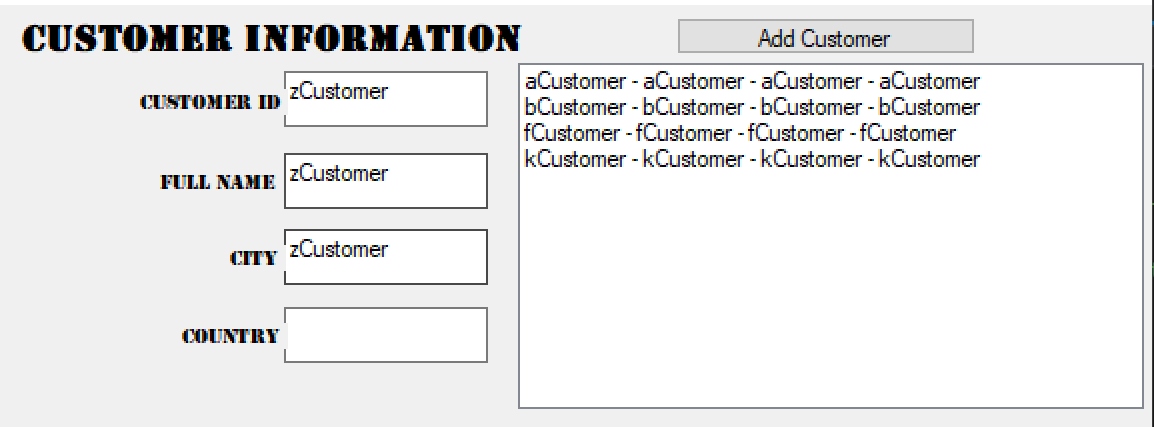
Capture 18.



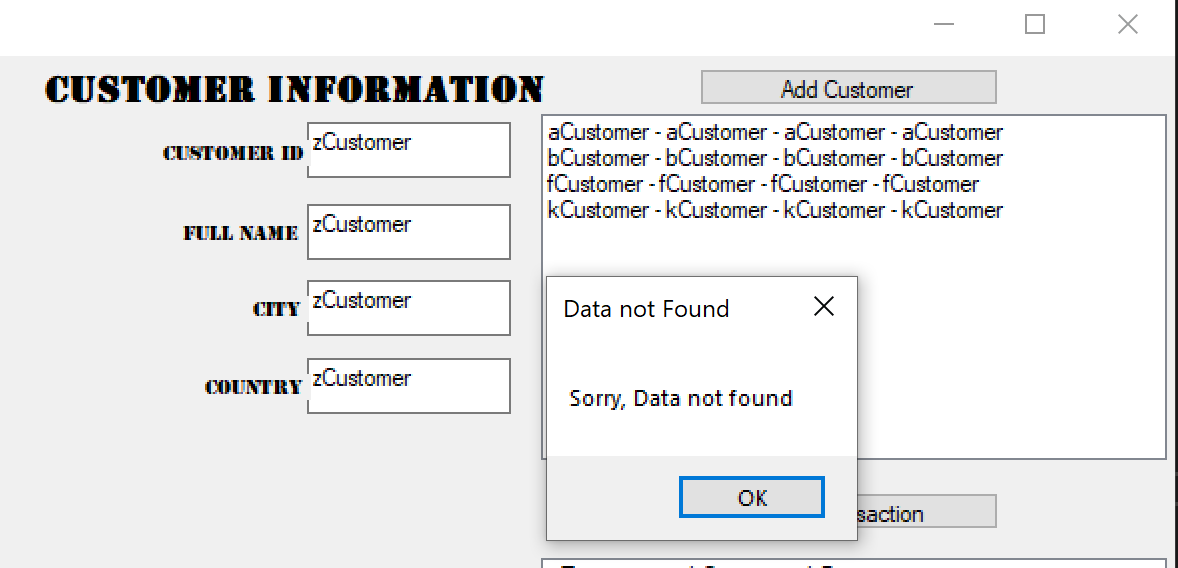
Capture 19.



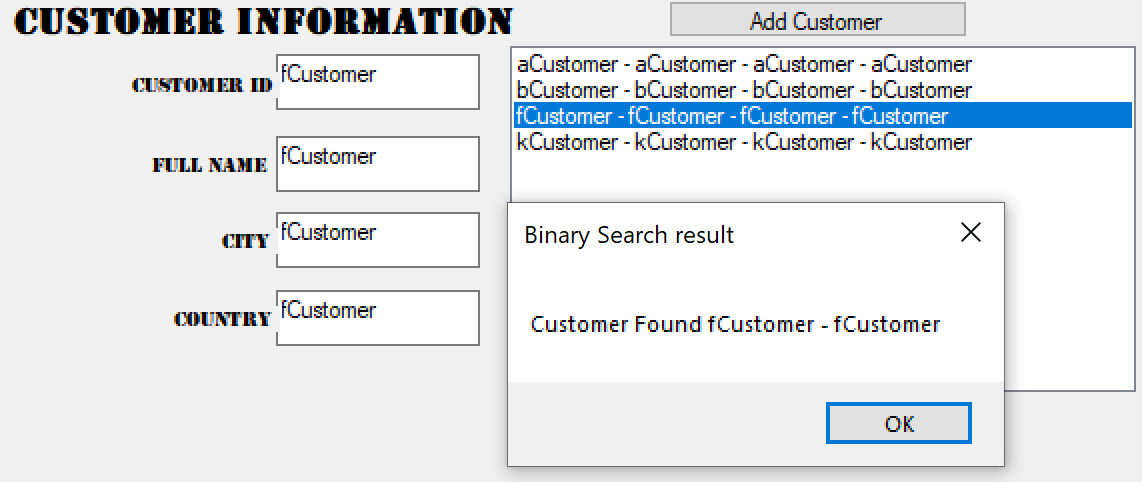
Capture 20.



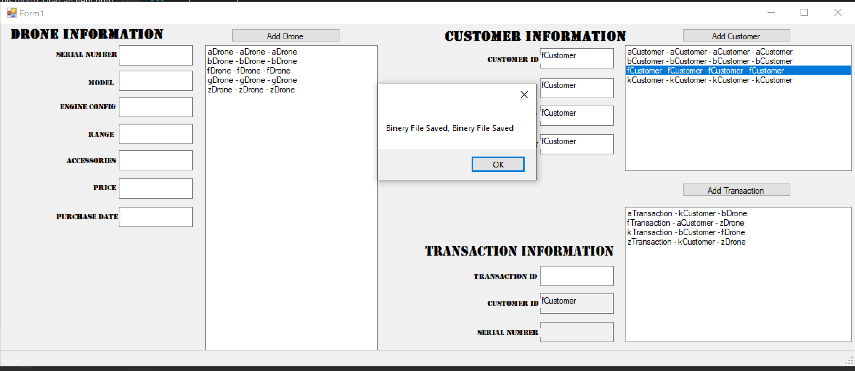
Capture 21.



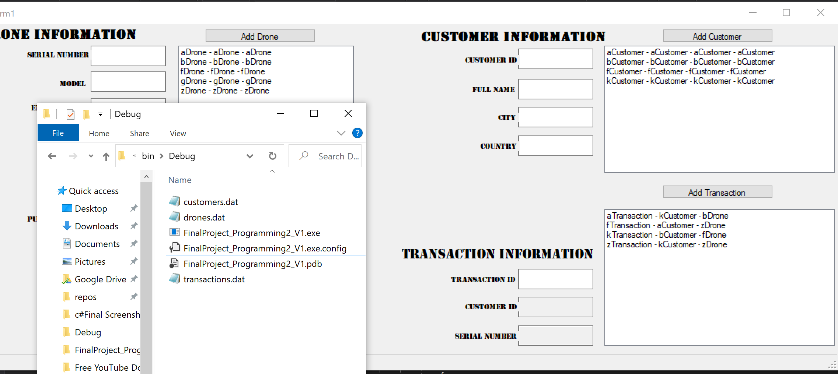
Capture 22.



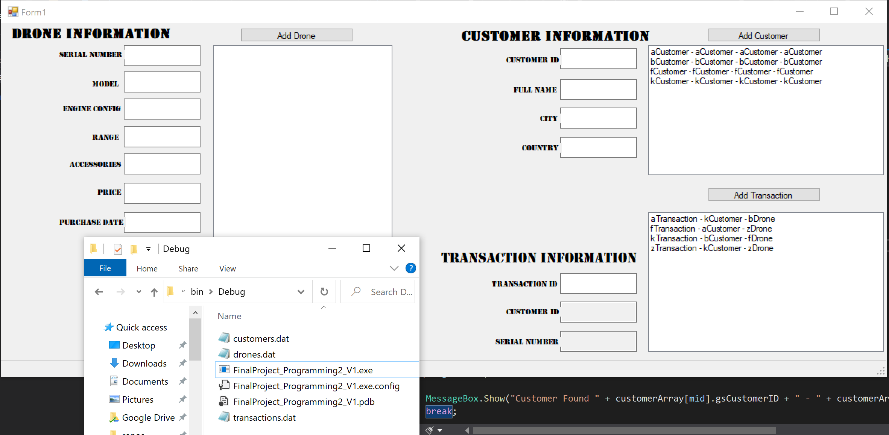
Capture 23.



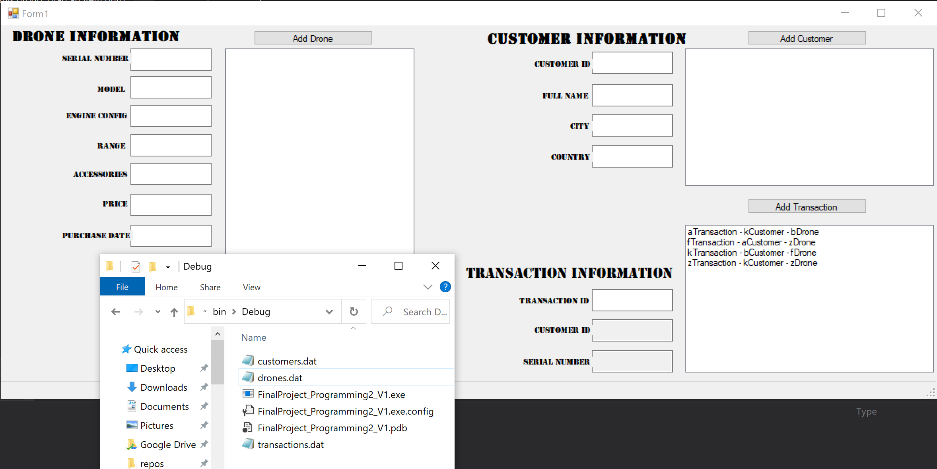
Capture 24



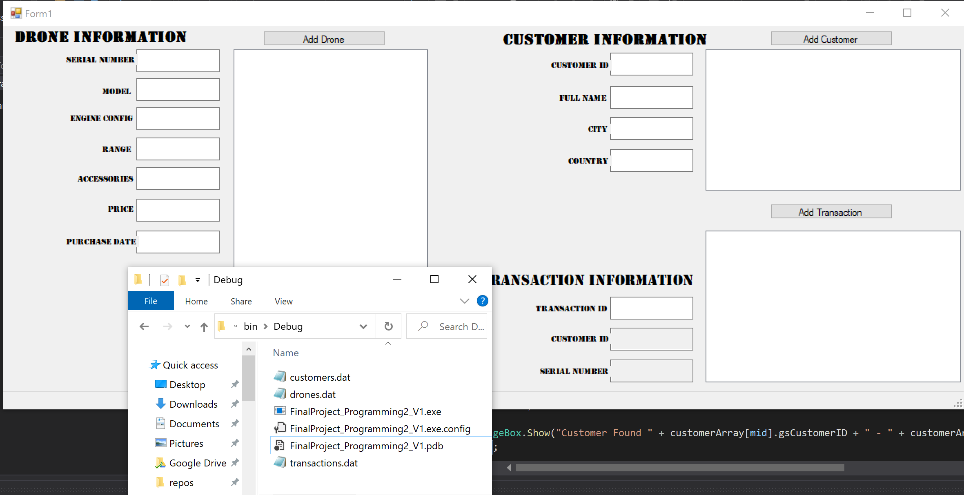
Capture 25.



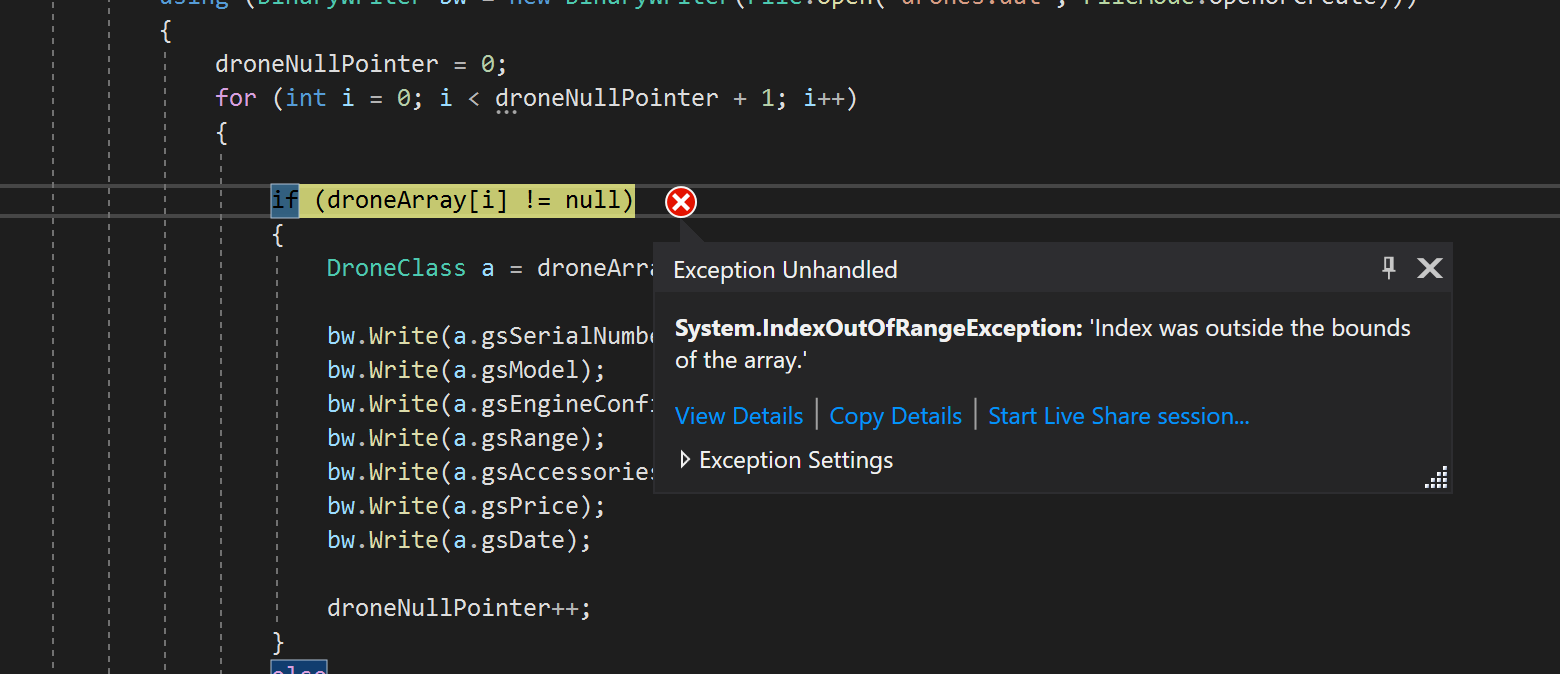
Capture 26.



Capture 27.



Capture 28.



Capture 29.

## Recommendations and Improvements

#### Recommendations

Some Recommendations to improve the program could be more functionality in regards to deleting objects from the various arrays. A Delete button would have been great in order to allow the user to fix mistakes in the entry of data.

A binary search could have been applied to all sections of the program not just the Customer section.

#### Limitations and Error Trappings.

Having two classes for the customer and drones but not for the transaction caused some implementation issues. Lots of space could be saved by having similar functionality across all classes.

Catching all the null references proved quite difficult especially when it came to checking each value before being added to the arrays.

The usability of the program is quite low since it only works if you perform certain actions in very certain order.