

Computing A2 Coursework

Imran Rahman

December 29, 2014

Contents

| | | |
|----------|---|----------|
| 1 | Analysis | 5 |
| 1.1 | Introduction | 5 |
| 1.1.1 | Client Identification | 5 |
| 1.1.2 | Define the current system | 6 |
| 1.1.3 | Describe the problems | 6 |
| 1.1.4 | Section appendix | 7 |
| 1.2 | Investigation | 9 |
| 1.2.1 | The current system | 9 |
| 1.2.2 | The proposed system | 17 |
| 1.3 | Objectives | 22 |
| 1.3.1 | General Objectives | 22 |
| 1.3.2 | Specific Objectives | 22 |
| 1.3.3 | Core Objectives | 23 |
| 1.3.4 | Other Objectives | 23 |
| 1.4 | ER Diagrams and Descriptions | 25 |
| 1.4.1 | ER Diagram | 25 |
| 1.4.2 | Entity Descriptions | 26 |
| 1.5 | Object Analysis | 26 |
| 1.5.1 | Object Listing | 26 |
| 1.5.2 | Relationship diagrams | 28 |
| 1.5.3 | Class definitions | 29 |
| 1.6 | Other Abstractions and Graphs | 30 |
| 1.7 | Constraints | 31 |
| 1.7.1 | Hardware | 31 |
| 1.7.2 | Software | 31 |
| 1.7.3 | Time | 31 |
| 1.7.4 | User Knowledge | 31 |
| 1.7.5 | Access restrictions | 32 |
| 1.8 | Limitations | 32 |
| 1.8.1 | Areas which will not be included in computerisation | 32 |
| 1.8.2 | Areas considered for future computerisation | 32 |
| 1.9 | Solutions | 33 |
| 1.9.1 | Alternative solutions | 33 |

| | | |
|----------|---|-----------|
| 1.9.2 | Justification of chosen solution | 33 |
| 2 | Design | 35 |
| 2.1 | Overall System Design | 35 |
| 2.1.1 | Short description of the main parts of the system | 35 |
| 2.1.2 | System flowcharts showing an overview of the complete system | 37 |
| 2.2 | User Interface Designs | 44 |
| 2.3 | Hardware Specification | 54 |
| 2.4 | Program Structure | 54 |
| 2.4.1 | Top-down design structure charts | 54 |
| 2.4.2 | Algorithms in pseudo-code for each data transformation process | 56 |
| 2.4.3 | Object Diagrams | 62 |
| 2.4.4 | Class Definitions | 63 |
| 2.5 | Prototyping | 66 |
| 2.6 | Definition of Data Requirements | 70 |
| 2.6.1 | Identification of all data input items | 70 |
| 2.6.2 | Identification of all data output items | 71 |
| 2.6.3 | Explanation of how data output items are generated | 72 |
| 2.6.4 | Data Dictionary | 72 |
| 2.6.5 | Identification of appropriate storage media | 74 |
| 2.7 | Database Design | 76 |
| 2.7.1 | Normalisation | 76 |
| 2.7.2 | SQL Queries | 81 |
| 2.8 | Security and Integrity of the System and Data | 83 |
| 2.8.1 | Security and Integrity of Data | 83 |
| 2.8.2 | System Security | 83 |
| 2.9 | Validation | 83 |
| 2.10 | Testing | 87 |
| 2.10.1 | Outline Plan | 87 |
| 2.10.2 | Detailed Plan | 87 |
| 3 | Testing | 97 |
| 3.1 | Test Plan | 97 |
| 3.1.1 | Original Outline Plan | 98 |
| 3.1.2 | Changes to Outline Plan | 98 |
| 3.1.3 | Original Detailed Plan | 98 |
| 3.1.4 | Changes to Detailed Plan | 98 |
| 3.2 | Test Data | 99 |
| 3.2.1 | Original Test Data | 99 |
| 3.2.2 | Changes to Test Data | 99 |
| 3.3 | Annotated Samples | 99 |
| 3.3.1 | Actual Results | 99 |
| 3.3.2 | Evidence | 99 |
| 3.4 | Evaluation | 100 |

| | | |
|----------|--------------------------------------|------------|
| 3.4.1 | Approach to Testing | 100 |
| 3.4.2 | Problems Encountered | 100 |
| 3.4.3 | Strengths of Testing | 100 |
| 3.4.4 | Weaknesses of Testing | 100 |
| 3.4.5 | Reliability of Application | 100 |
| 3.4.6 | Robustness of Application | 100 |
| 4 | System Maintenance | 101 |
| 4.1 | Environment | 102 |
| 4.1.1 | Software | 102 |
| 4.1.2 | Usage Explanation | 102 |
| 4.1.3 | Features Used | 102 |
| 4.2 | System Overview | 102 |
| 4.2.1 | System Component | 102 |
| 4.3 | Code Structure | 102 |
| 4.3.1 | Particular Code Section | 102 |
| 4.4 | Variable Listing | 102 |
| 4.5 | System Evidence | 102 |
| 4.5.1 | User Interface | 102 |
| 4.5.2 | ER Diagram | 102 |
| 4.5.3 | Database Table Views | 102 |
| 4.5.4 | Database SQL | 102 |
| 4.5.5 | SQL Queries | 102 |
| 4.6 | Testing | 102 |
| 4.6.1 | Summary of Results | 102 |
| 4.6.2 | Known Issues | 102 |
| 4.7 | Code Explanations | 102 |
| 4.7.1 | Difficult Sections | 102 |
| 4.7.2 | Self-created Algorithms | 102 |
| 4.8 | Settings | 102 |
| 4.9 | Acknowledgements | 102 |
| 4.10 | Code Listing | 102 |
| 4.10.1 | Module 1 | 103 |
| 5 | User Manual | 104 |
| 5.1 | Introduction | 105 |
| 5.2 | Installation | 105 |
| 5.2.1 | Prerequisite Installation | 105 |
| 5.2.2 | System Installation | 105 |
| 5.2.3 | Running the System | 105 |
| 5.3 | Tutorial | 105 |
| 5.3.1 | Introduction | 105 |
| 5.3.2 | Assumptions | 105 |
| 5.3.3 | Tutorial Questions | 105 |
| 5.3.4 | Saving | 105 |
| 5.3.5 | Limitations | 105 |

| | | |
|----------|---------------------------------------|------------|
| 5.4 | Error Recovery | 105 |
| 5.4.1 | Error 1 | 105 |
| 5.4.2 | Error 2 | 105 |
| 5.5 | System Recovery | 105 |
| 5.5.1 | Backing-up Data | 105 |
| 5.5.2 | Restoring Data | 105 |
| 6 | Evaluation | 106 |
| 6.1 | Customer Requirements | 107 |
| 6.1.1 | Objective Evaluation | 107 |
| 6.2 | Effectiveness | 107 |
| 6.2.1 | Objective Evaluation | 107 |
| 6.3 | Learnability | 107 |
| 6.4 | Usability | 107 |
| 6.5 | Maintainability | 107 |
| 6.6 | Suggestions for Improvement | 107 |
| 6.7 | End User Evidence | 107 |
| 6.7.1 | Questionnaires | 107 |
| 6.7.2 | Graphs | 107 |
| 6.7.3 | Written Statements | 107 |

Chapter 1

Analysis

1.1 Introduction

1.1.1 Client Identification

My client, Shahida Rahman, is an Author, and the Director and Secretary of Perfect Publishers Ltd, which has been a self publishing company since 2005. This means that the author pays Perfect Publishers to publish their book. She published her first book through Perfect Publishers Ltd, and this was when the company was born. She is 42 years old and is a mother of 4 children. Shahida uses computers to deal with online enquiries and to publish books from all over the world. Furthermore, she also produces the royalty statements for each book twice yearly. Aside from this, she has little experience with computers. Shahida generally uses a computer for research, social networking and reading the news. Every book is outsourced to an Editor and a Cover Designer. When the book is fully edited and formatted to the right specifications, they return the ready to print files to Shahida, who sends the books off to print. They track the books and their details manually using a database on an Excel Spreadsheet. Currently, it is difficult to keep all the data up to date and it is rather disorganised. Shahida would like to be able to look up a book/number of books in the system by using the details, such as the Author/Title/Date etc. She would also like the new system to link this database with information about the royalties of each book, and when they are needed to be paid every six months. The system could send an email to her, updating her about these.

1.1.2 Define the current system

The system that is currently being used consists of Shahida entering the book and its details into the spreadsheet. These details are taken from the enquiries that she receives via email, and include; first name, last name, email, and vague details of the book. Then, the more accurate details such as; book title, size, number of pages, hardback/paperback, mat or gloss, crème or white paper, font and font size are discussed between Shahida and the customer. She also records their details in a separate spreadsheet, which includes their email, phone number, and address, upon the customer's consent. The spreadsheets are used to track current and previous customers and their books, as the details about the book and themselves are recorded in these. Subsequently, Shahida informs the customer of the price details, waits for full payment and then sends the customer an invoice. She then contacts her editor and her illustrator to start work on the book. Shahida refers to her company's website, where the calculated prices are ready for books, in order to correctly price the book, in accordance to the book's details. An ISBN number is assigned to the book, which is bought in bulks by Shahida from the ISBN Office. Once the book is finished, the book is sent off to print, and the author receives 25 copies.

1.1.3 Describe the problems

There are numerous problems with the current system. First of all, the usage of the spreadsheet makes it harder to find a customer and their details, and their book's details. This is because the spreadsheet is much disorganised. Furthermore, it is harder to keep track of the details of each book, meaning it is difficult to update the details of the book when necessary. Because there are a large number of books in the system, it is harder to find each book and then separately find the customer that the book was written by, as they are in separate spreadsheets, meaning that Shahida must manually search for them in both spreadsheets. Also, if the same author makes an enquiry about another book, their details must be entered into the spreadsheet again, because it is difficult to find where the customers details currently are due to there being many customers in the spreadsheet, having incorrectly entered their data from a given enquiry, or they might have been removed after having been there for a long time. This could cause inconsistencies in the data, because for instance, the customer may move house, meaning their address would need changing, and it would be difficult to find and update all entries where their address is recorded.

1.1.4 Section appendix

Interview Questions

1. What current system is in place?
 - Excel spreadsheets
 - Holds details of the books and authors/customers
2. What are the problems with this system?
 - Hard to keep track of everything
 - Data is duplicated for existing customers
 - Very disorganised
3. What data do you record?
 - Details about the authors - Name, email, Phone number, Address
 - Book details - Title, author, Pages, Hardcover softback, paper back, colour, size, ISBN
4. How much data is duplicated for existing customers who send another enquiry?
 - Every time an enquiry is sent, their details are added to the database
5. What should the new system accomplish?
 - To create an organised database
 - To avoid duplication
 - To be able to calculate royalties
6. What will stay the same?
 - Details that are stored
 - Storing data electronically
7. How long will the data remain in the system?
 - As long as necessary due to details required for paying royalties
8. Are hard copies of the data required?
 - No, everything is conducted electronically
9. What computing resources do you have available to you?
 - Laptop - windows 7
 - Microsoft Office
10. Is security an issue?
 - Very secure - Data isn't shared with other 3rd parties
 - kept confidential

Figure 1.1: Interview Questions: Page 1

11. Who will be using the data?

Just Shavida, and the customer it belongs to

12. Do you have a particular system in mind?

Anything that is efficient
avoids duplication of data
easy to keep track of data

I confirm that I answered these questions to help Imran Rahman investigate my current system to help with the design of his new system.

Signed:

Imran Rahman

Figure 1.2: Interview Questions: Page 2

1.2 Investigation

1.2.1 The current system

Data sources and destinations

In the current system there are three key data sources that are used. These are Shahida herself, the customer and the spreadsheets. The Customer sends the enquiry which is sent via email. After the details about the book have been discussed, they are stored in one spreadsheet, and the details of the customer are stored in a separate spreadsheet, linked to the details of the book. These details are agreed separately from the enquiry, between Shahida and the customer. Details such as the book size, page number, hardback/softback and paper type are used to calculate the cost for the customer, which is used to create an invoice which is sent to the customer. This is the first output of the system. A copy of every invoice is stored on Shahida's computer in a special folder just for invoices. Once Shahida receives full payment, the work is conducted and completed. If the customer wishes to publish another book, they send another enquiry, and their personal data is duplicated because of the details of the new book which are added. Every six months after the book has been published, the royalties must be paid to the author. The royalties are the profit that the author makes from sales of her book from bookshops. A royalty statement is created and stored in a special folder just for royalties.

| Source | Data | Example Data | Destination |
|------------------|--------------------|------------------|-----------------|
| Customer Enquiry | Forename | Peter | Shahida |
| Customer Enquiry | Surname | Parker | Shahida |
| Customer Enquiry | Email | mail@example.com | Shahida |
| Customer | Address | 1 Example Road | Shahida |
| Customer | Postcode | AB1 2CD | Shahida |
| Customer | Phone Number | 07123456789 | Shahida |
| Customer | Book Title | The Hobbit | Shahida |
| Customer | Size | Large | Shahida |
| Customer | Number of Pages | 395 | Shahida |
| Customer | Hardback/Paperback | Paperback | Shahida |
| Customer | Mat/Gloss | Gloss | Book Database |
| Customer | Creame/White Paper | White Paper | Shahida |
| Customer | Font | Times New Roman | Shahida |
| Customer | Font Size | 12 | Shahida |
| Shahida | Book Title | The Hobbit | Book Database |
| Shahida | Size | Large | Book Database |
| Shahida | Number of Pages | 395 | Book Database |
| Shahida | Hardback/Paperback | Paperback | Book Database |
| Shahida | Mat/Gloss | Gloss | Book Database |
| Shahida | Creame/White Paper | White Paper | Book Database |
| Shahida | Font | Times New Roman | Book Database |
| Shahida | Font Size | 12 | Book Database |
| Shahida | Forename | Peter | Author Database |
| Shahida | Surname | Parker | Author Database |
| Shahida | Email | mail@example.com | Author Database |
| Shahida | Address | 1 Example Road | Author Database |
| Shahida | Postcode | AB1 2CD | Author Database |
| Shahida | PhoneNumber | 07123456789 | Author Database |
| Shahida | Invoice | - | Invoice Folder |
| Shahida | Invoice | - | Customer |
| Shahida | ISBN | 9780007525492 | Book Database |
| Shahida | Date Published | 23/10/2014 | Book Database |

| Source | Data | Example Data | Destination |
|----------------|-------------------|--------------|--------------------------|
| Shahida | Price | £12.99 | Book Database |
| Customer | Payment | £1000 | Shahida |
| Shahida | Cover Preferences | - | Cover Designer |
| Shahida | Book | - | Editor |
| Editor | Completed Book | - | Shahida |
| Cover Designer | Completed Cover | - | Shahida |
| Shahida | Royalty Statement | - | Royalty Statement Folder |
| Shahida | Royalty Statement | - | Customer |
| Shahida | Royalties | £211.20 | Customer |

Algorithms

In the current system there are two Algorithms which are being used. The first sends an invoice to the customer and checks whether Shahida has recieved full payment. Once Shahida has received full payment, she, her cover designer and her editor can begin working on the book. The second algorithm consists of completing the work that is needed to be done, and checks whether the work has been completed, so that the completed book can then be sent off for printing.

Algorithm 1 First Algorithm - Sending an invoice and Checking for Payment

```

1: SET Payment TO false
2:
   Check Website for Price
   Create Invoice
   Send Invoice
3: WHILE Payment = false DO
   Check For Payment
4:   IF PaymentReceived THEN
     Payment = true
5:   END IF
6: END WHILE

```

Algorithm 2 Second Algorithm - Completing Work and Checking If Work is Completed

```
1: SET WorkComplete TO false
2: SET CoverComplete TO false
3: SET BookComplete TO false
4:
5: WHILE WorkComplete = false DO
    Get Completed Cover from Cover Designer
6:   SET CoverComplete TO true
    Get Completed Book from Editor
7:   SET BookComplete TO true
8:   IF BookComplete and CoverComplete THEN
9:     SET finished TO true
10:   END IF
11: END WHILE
```

Data flow diagrams

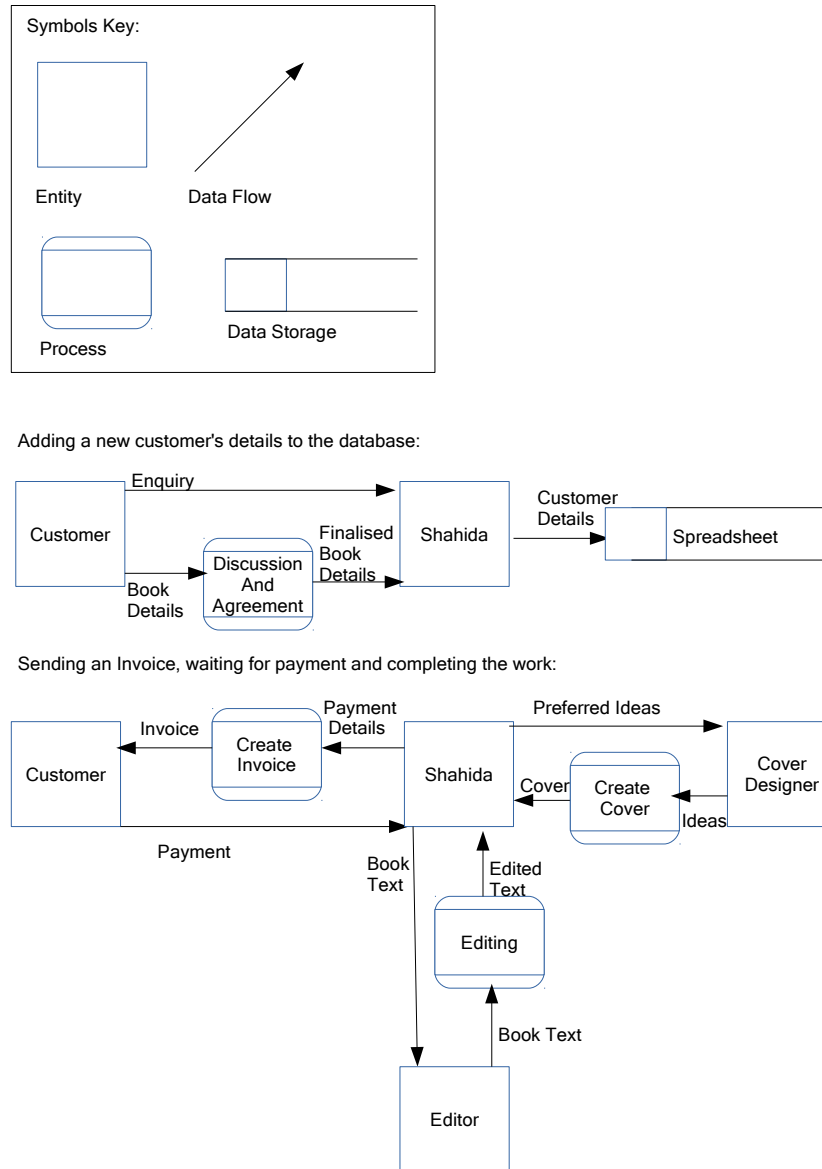


Figure 1.3: Data Flow Diagrams

Input Forms, Output Forms, Report Formats

The current system has just one input form. This is the Enquiry that is sent to the company, from an author. Also, The current system has two different output forms - The Invoice and The Royalty Statement.

The enquiry is received via email, which is sent using the company's website. The email will look like this when received:

First Name:

Last Name:

Email:

Question/Comment:

The following image is an example of the first output form, an Invoice.

PERFECT PUBLISHERS Ltd. 23 MAITLAND AVENUE, CAMBRIDGE, CB4 1TA, UK.
Tel: +44 (0) 1223 424422 Fax: +44 (0) 1223 424414

**INVOICE***"Fulfil your books' potential"***Invoice Date: 27-05-14****Author Name: Svagito Liebermeister****Title of Book: Osho Therapy****ISBN: 978-1905399-9-25**

Shipping details:
Pratibha de Stoppani
via Al Marcadello 2
CH-6988 Ponte Tresa
Switzerland

| Order Description | | PRICE |
|---|---------|----------------|
| 12 Osho Therapy @ 50% discount. Retail price £25.99 | | £155.94 |
| Shipping | Premium | £10.06 |
| TOTAL | | £166.00 |

Account details: BIC: LOYDGB21206 IBAN: GB33 LOYD 3091 7402 3570 35

SORT CODE: 30-91-74 ACCOUNT NUMBER: 02357035

Payment within 14 days. Late payment will incur a fixed penalty of £20 for the first month and £50 per month thereafter.

Company Number 5429532. VAT Number: 857 5975 58. Registered in England.
www.perfectpublishers.co.uk

Figure 1.4: Invoice Example

The following image is an example of the second output form, a Royalty Statement.

23 Maitland Avenue
Cambridge
CB4 1TA
United Kingdom
enquiries@perfectpublishers.co.uk



ROYALTY STATEMENT

Date: 01-01-14 – 30-06-14

| AUTHOR | TITLE | ISBN (13-DIGIT) |
|--------------|-------------------------|-----------------|
| Andre Corrie | Into The Mourning Light | 9781905399895 |

| LIST PRICE | DISCOUNT | WHOLESALE PRICE | QUANTITY | NET SALES | PRINT COST | NET PUB COMP |
|------------|----------|-----------------|----------|--------------|------------|----------------|
| \$15.99 | 40% | \$9.59 | 19 | \$182.21 | \$96.00 | \$91.01* |
| £9.99 | 40% | £5.99 | 69 | £413.31 | £233.10 | £158.01 |
| | | | | TOTAL | | £211.20 |

PRINT COST PER BOOK: £3.70 UK AND \$4.80 US
*\$91.01 = £53.19

1 GBP = 1.71565 USD 09-07-14

Company Number: 5429532. VAT Number: 857 5975 58. Registered in England.
www.perfectpublishers.co.uk | www.facebook.com/ppublishers | www.twitter.com/ppublishers

Figure 1.5: Royalty Statement Example

1.2.2 The proposed system

In the proposed system the Customer's information will still be received through the online form on the company's website, which Shahida receives via email. She will then enter this into the system using a new interface that will ask her for the details. This will be placed into a database. Each Customer's book will have a primary key, the ISBN number which Shahida assigns to the book. In a separate database, the author's details will be stored and the author will have a special ID number which is used only in the databases. Every book that is published by the same author will have an attribute which is the author's ID. The ID will just be a 3 digit number. The system's interface will have a search feature, which can search for book titles, authors, and author IDs.

Data sources and destinations

| Source | Data | Data Type | Destination |
|------------------|-------------------------|--------------------|--------------------|
| Customer Enquiry | Forename | String | Shahida |
| Customer Enquiry | Surname | String | Shahida |
| Customer Enquiry | Email | String | Shahida |
| Customer | Address | String | Shahida |
| Customer | Postcode | String | Shahida |
| Customer | Phone Number | String | Shahida |
| Customer | Book Title | String | Shahida |
| Customer | Size | String | Shahida |
| Customer | Number of Pages | 395 | Shahida |
| Customer | Hardback/Paperback | Paperback | Shahida |
| Customer | Mat/Gloss | Gloss | Shahida |
| Customer | Creame/White Pa- per | White Paper | Shahida |
| Customer | Font | Times New Roman | Shahida |
| Customer | Font Size | 12 | Shahida |
| Shahida | Book Title | The Hobbit | Database |
| Shahida | Size | Large | Database |
| Shahida | Number of Pages | 395 | Database |
| Shahida | Hardback/Paperback | Paperback | Database |
| Shahida | Mat/Gloss | Gloss | Database |
| Shahida | Creame/White Pa- per | White Paper | Database |
| Shahida | Font | Times New Roman | Database |
| Shahida | Font Size | 12 | Database |
| Shahida | Forename | String | Database |
| Shahida | Surname | String | Database |
| Shahida | Email | String | Database |
| Shahida | Address | String | Database |
| Shahida | Postcode | String | Database |
| Shahida | PhoneNumber | String | Database |
| Shahida | ISBN | String | Database |
| Shahida | Date Published | Date | Database |
| Shahida | Price | Real | Book Database |
| Database* | Author ID | Integer | Shahida |
| Shahida | Invoice | String | Invoice Folder |
| Shahida | Invoice | String | Customer |

| Source | Data | Data Type | Destination |
|----------------|-------------------|-----------|--------------------------------|
| Customer | Payment | Real | Shahida |
| Shahida | Cover Details | String | Cover De- signer |
| Shahida | Book | String | Editor |
| Editor | Completed Book | String | Shahida |
| Cover Designer | Completed Cover | Image | Shahida |
| Shahida | DatePublished | Date | Database |
| Shahida | Royalty Statement | - | Royalty Statement Folder |
| Shahida | Royalty Statement | - | Customer |
| Shahida | Royalties | Real | Customer |

*The Database will create a number and assign that number as an Author ID

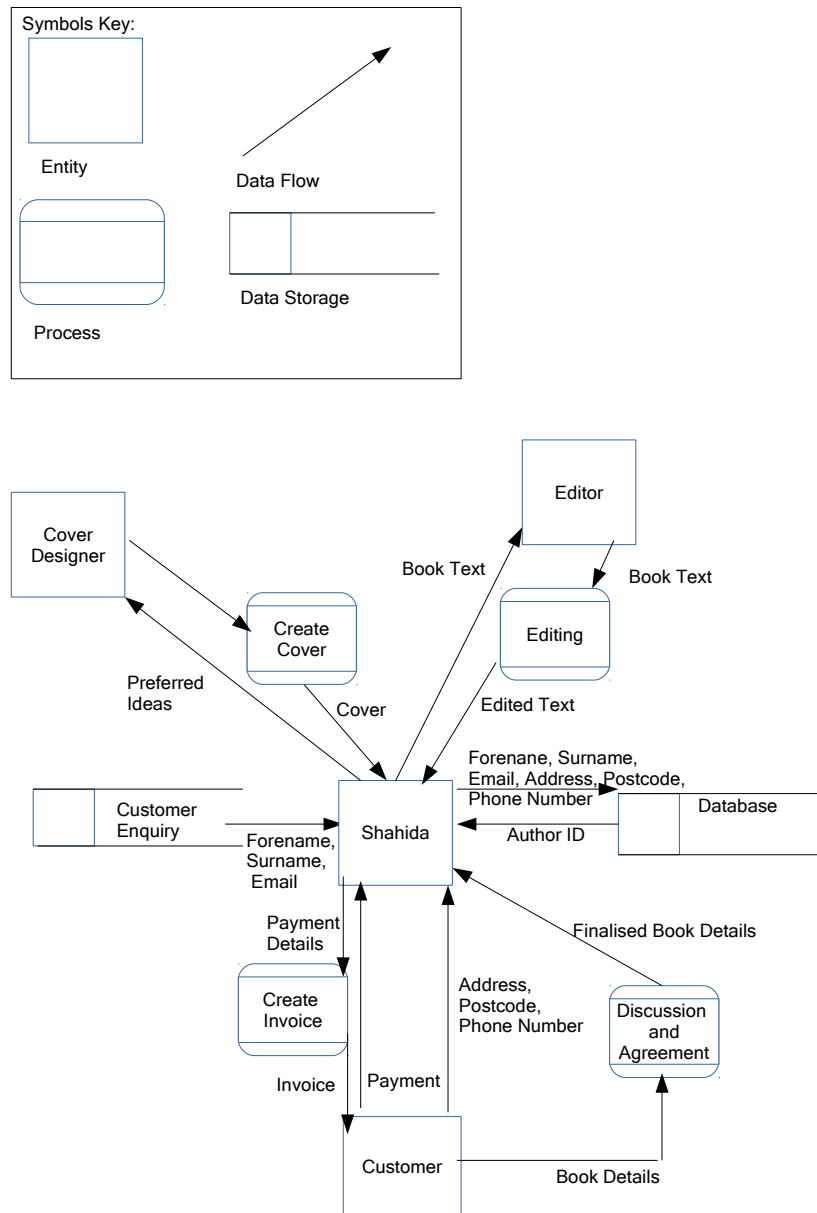
Data flow diagram

Figure 1.6: Data Flow Diagram

Data dictionary

| Name | Data Type | Length | Validation | Example Data |
|---------------|-----------|-------------------|--------------|------------------|
| FirstName | String | 2-20 Characters | Length | Jo |
| LastName | String | 2-20 Characters | Length | Williamson |
| Email | String | 7-30 Characters | Length | mail@example.com |
| PhoneNumber | String | 9-15 Characters | Format | 07123456789 |
| Address | String | 5-64 Characters | Length | Example Road |
| Postcode | String | 7 Characters | Format | AB1 2CD |
| Author ID | Integer | 1-255 | Range | 17 |
| ISBN | String | 13 Characters | Length | 9780007525492 |
| BookTitle | String | 1-127 Characters | Length | The Hobbit |
| NoOfPages | Integer | 1-1023 | Range | 395 |
| Size | String | 5 | Existence | Large |
| Back | String | 8 or 9 Characters | Existence | Paperback |
| Cover | String | 3 or 5 Characters | Existence | Gloss |
| Paper | String | 11 Characters | Existence | White Paper |
| Font | String | 1-64 Characters | Length | Arial |
| FontSize | Real | 8-64 | Numbers only | 12.5 |
| DatePublished | Date | dd/mm/yyyy | Range | 23/10/2014 |
| Price | Real | Numbers only | £12.99 | |

Volumetrics

I have conducted calculations to calculate the maximum possible size of 1 customer and book record, which is 275 Bytes. However, when a customer wishes to publish more than one book, more book records are required. As the most amount of books one customer has published with the company is 3, we can have 4 book records per customer record.

Each ASCII Character is 1 byte, each number up to 255 is 1 byte, and each number between 256 and 32768 is 2 bytes. Real Numbers such as 12.5 are 2 bytes, and a Date is 3 bytes.

Firstly, I have worked out the size of the customer record, which is 157 Bytes.

FirstName (20) + LastName (20) + Email (30) + PhoneNumber (15) + Address (64) + Postcode (7) + Author ID (1) = 157 Bytes.

I have then calculated the size of one book record, which is 118 Bytes. Book Title (1), NoOfPages (2), Size (5), Back (9), Cover (5), Paper (11), Font (64),

FontSize (2) + ISBN (13) + DatePublished (3) + Price (2) + Author ID (1) = 118 Bytes

If we have 4 book records per customer record, that would mean that the size for 1 customer with 4 books would be $157 + (5 * 118) = 747$ Bytes.

I have chosen to use a size of 100 different customer records, which would be equivalent to 74700 bytes, and $74700 / 1024 = 72.9$ Kilobytes. This is because the company rarely have more than 20 enquiries in a year. This would be a suitable number of customer records as it will last a few years before it may require resizing, which can be conducted at a later date when necessary. 72.9 KB will not be difficult for Shahida's PC to hold, as it is a very small size.

1.3 Objectives

1.3.1 General Objectives

The general objectives are:

- Organised layout for the database.
- Prevention of unnecessary duplication of data.
- Simple interface for entering data, meaning it can be conducted quickly.
- Search function to find a specific customer in the database.
- Ability to edit existing data easily and quickly.

The System must be able to prevent unnecessary duplication of data, and be able to organise data well, and this will be a priority.

1.3.2 Specific Objectives

Organising a new layout for the database:

- Be able to sort by date (ascending and descending)
- Clear tables and fields for each entity and attribute

Preventing Duplication:

- Checks to see if the data already exists
- Use of Author ID to ensure it will only be entered once

Simple interface for entering data:

- As little amount of boxes as possible
- Clearly label entry boxes

Search function and editing data:

- Data can be found using the Author ID, Author Name, or Book Title
- Can be edited upon finding the desired data

1.3.3 Core Objectives

- Organising the data using certain attributes
- Preventing Duplication

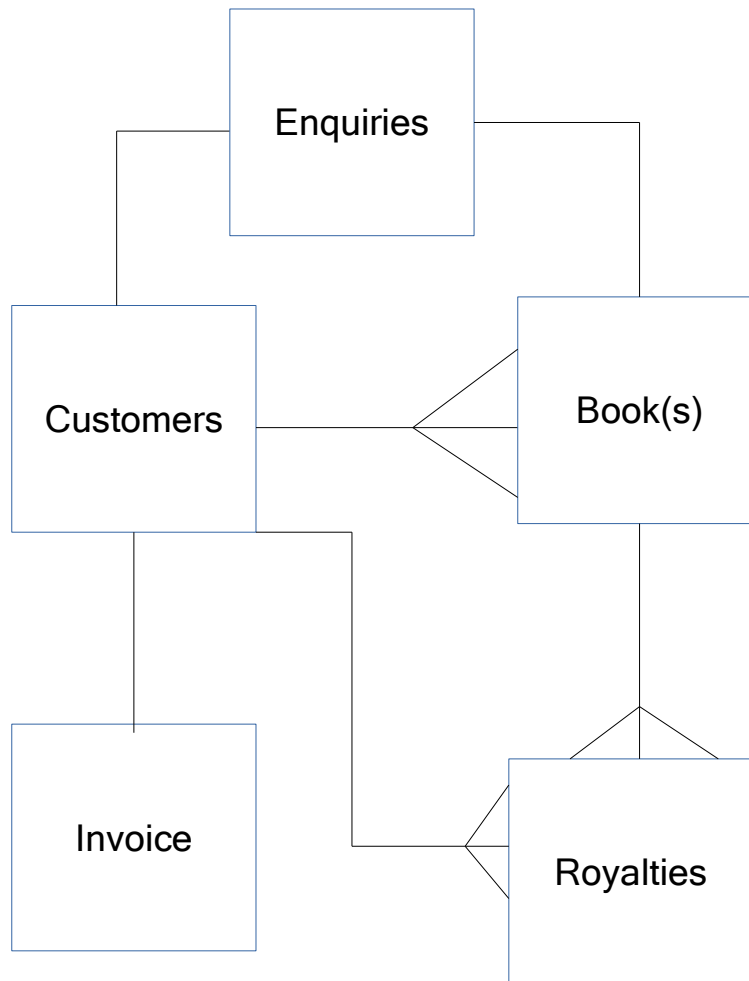
1.3.4 Other Objectives

- Searching for data using attributes
- Editing data in the database

1.4 ER Diagrams and Descriptions

1.4.1 ER Diagram

Figure 1.7: ER Diagram



1.4.2 Entity Descriptions

Customer(Author ID, *Email*, Forename, Surname, Address, Postcode, Phone Number)

Enquiry(Email, *Author ID*, Forename, Surname)

Invoice(Author ID, *ISBN Number*, Book title, Price, Forename, Surname, Address, Postcode)

Royalties(AuthorID, *ISBN Number*, Book title, Price, Forename, Surname, Address, Postcode)

Book(ISBN Number, *AuthorID*, Book Title, Pages, Size, Cover type, Colour, Back Type, Paper, Font, Font size, Date published, Price)

The database will only store data about the customers and their books, as the enquiries give details about the books and customers, and the royalties and invoices are stored seperately from the database.

1.5 Object Analysis

1.5.1 Object Listing

- Shahida
- Customer
- Editor
- Cover Designer
- Spreadsheet

1.5.2 Relationship diagrams

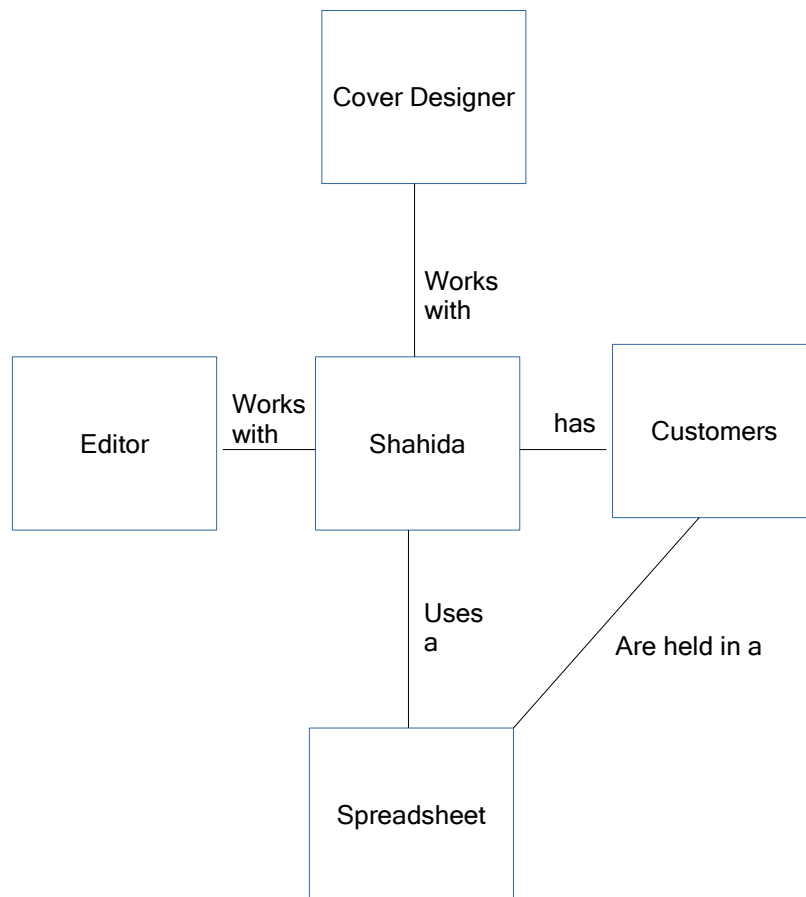


Figure 1.8: Relationship Diagram

1.5.3 Class definitions

Key:

| Label |
|------------|
| Attributes |
| Behaviours |

| Customer |
|---------------|
| Author ID |
| ForeName |
| Surname |
| Email |
| Address |
| Postcode |
| Phone Number |
| Add ForeName |
| Edit Forename |
| Add Surname |
| Edit Surname |
| Add Email |
| Edit Email |
| Add Address |
| Edit Address |
| Add Postcode |
| Edit Postcode |
| Add Phone |
| Number |
| Edit Phone |
| Number |

| Book |
|---------------------|
| Title |
| ISBN |
| Pages |
| Size |
| Cover Type |
| Back Type |
| Paper |
| Font |
| Font Size |
| Price |
| Date Published |
| Add Title |
| Edit Title |
| Add ISBN |
| Edit ISBN |
| Add Pages |
| Edit Pages |
| Add Size |
| Edit Size |
| Add Cover Type |
| Edit Cover Type |
| Add Back Type |
| Edit Back Type |
| Add Paper |
| Edit Paper |
| Add Font |
| Edit Font |
| Add Font Size |
| Edit Font Size |
| Add Date Published |
| Edit Date Published |
| Add Price |
| Edit Price |

1.6 Other Abstractions and Graphs

Graphs not required.

1.7 Constraints

1.7.1 Hardware

Shahida uses her laptop to run the company from home. The new system will need to be able to run on this machine.

Computer Specifications:

- 15.6" Display
- AMD Quad-Core A4-5000M APU (1.5GHz, 2MB cache)
- 4 GB DDR3 RAM
- 750 GB HDD, 5400 rpm
- AMD Radeon HD 8330 Graphics Card

The proposed system will have no problem with running on this machine, as it uses a small amount of CPU usage. A constraint would be the size of the screen. This is because the system will need to be based around the screen size of her laptop. As her laptop is portable, portability is not a constraint. The laptop will need enough RAM to hold the system. However, Shahida's laptop has more than enough memory for this, meaning this will not be a problem.

1.7.2 Software

Shahida would prefer that the system will run on Windows 7, as she uses this operating system for her laptop. Changing the operating system will cause difficulties, meaning it is best for the system to run on Windows 7, suiting her needs.

1.7.3 Time

Shahida does not need this system to be built quickly, but she would like it to be complete as soon as reasonably possible. Otherwise, the only deadline for this project is April 2015, which has been set by my teacher.

1.7.4 User Knowledge

Having worked in the publishing industry beforehand, being an author has also given Shahida the knowledge of how to run her current company. Aside from being able to perform basic tasks on a computer, browsing the internet and using social media, Shahida has small experience with computers.

1.7.5 Access restrictions

Shahida will be the only person who will have full access to all the data in the proposed system, and she will be the only one who can access it. This can be password protected for security reasons, meaning that only she can gain access to the database. This is also because she is the only necessary person to view, enter and edit data in the system, as her Editor and her Cover Designer do not need to use the database. As she is the only user of the database, it will be easier to keep secure. The authors will be able to make requests about personal data, such as having it removed, or receiving a copy of the personal data about them. The database will comply to the Data Protection Act 1998, as the company already does so with their current system.

1.8 Limitations

1.8.1 Areas which will not be included in computerisation

Generally, all actions require the use of a computer in the company. However, rarely, a customer does call Shahida about an enquiry, as this customer may not be so computer literate. In this case, Shahida will note down the details of the enquiry, and will enter it into the database.

1.8.2 Areas considered for future computerisation

The database could be used online, so that the authors can use their Author ID to log in and see just their details on the database. This would mean that the customers would not have to contact Shahida to receive the data held about them, as they can see the data by themselves. They will also be able to access this data from anywhere where they have access to the internet. This could also enable Shahida to access the data from other machines aside from her laptop.

1.9 Solutions

1.9.1 Alternative solutions

| Solution | Advantages | Disadvantages |
|--|--|---|
| Re-organisation of the current spreadsheet | No changes to current operating system and software required, will not cost | Current problems will still occur, Difficult to keep organised as it will require more maintenance to do so |
| Python Desktop Application with GUI | User Friendly, Clear and easy to interpret, Layout can be designed specifically for the client, Usage of buttons simplifies tasks, Minimal training needed for most levels of experience | Takes up more memory, Takes longer to create the application |
| Filing system | No electronics needed, Costs less, Minimal training needed for most levels of experience | Difficult to back up the data due to it being held on paper, data will have to be sent via post when necessary, Lots of physical space is required, more prone to damage and deterioration due to more movement |

1.9.2 Justification of chosen solution

I have chosen to use the Python Desktop Application with GUI as my solution. This is because:

- I am already familiar with the Python Programming Language, whereas I have little knowledge of how to manage a Paper Filing system or with creating advance spreadsheets.
- This will keep the system using computers and software, meaning there will not be a drastic change.
- Using the application will take less time than manually entering everything into a spreadsheet.

- This will also take less time and physical space than writing details down on paper.

Chapter 2

Design

2.1 Overall System Design

2.1.1 Short description of the main parts of the system

- Log In Window
- Main Database Interface
- Adding/Removing/Editing Customers and Entries
- Calender Interface
- Changing Password
- Search Window

Log In Window

- A window is displayed which prompts the user to input their ID and password.
- Checks the entered values with the database to identify whether the user's credentials are correct.
- Once a correct set of values are entered, the user will be granted access to the database.
- A link will be at the bottom which says "Forgotten password?". This can be clicked on and then the user will be prompted for the email address, and the corresponding password for the email address entered will be sent to that email.

- If there is no record of an email and password then the user will be prompted to create one for their corresponding email.

Main Database Interface

- This will be the "home" interface.
- A view of the Customer details in the database will be available.
- The user can select an author from the basic view of the database, and click view
- A user interface is presented with a set of options which are: View, Search Database, Add Entry, Remove an Entry, Edit an Entry, Change Password, and Log out.
- Clicking the Search Database Button will prompt a separate interface to open, and shows details which can be used to search for specific items in the database.

View Screen

- Clicking view after having selected a customer will open a new window which will show a more in depth view of it. It will show the books that have been published with them.
- There will be buttons to expand on certain fields, including Royalties, Publishing Invoices and Book Invoices. These will show in new windows. If the user has clicked on Royalties, they have the option to click on Royalties Items on the following screen, where they can see breakdown of it. If they have clicked on Book Invoices, they can click on the Book Invoice Items on the following screen to see a breakdown of these too. From the Royalties/Book Invoice screens, they can calculate and add more Royalties/Invoices.
- Customers can be searched for quickly using their AuthorID on this screen.

Adding/Removing/Editing Customers and Entries

- Clicking the Add Entry Button will prompt a separate interface to open, and contains a layout of entry boxes for required fields for entering details about the customer. After this, the user can click on the customer's new record from the menu and click edit or a book/royalties/royalty items/-book invoice/book invoice items/publication invoice, dependent on which has been selected. An existing customer can be selected using the search function.
- If a customer already exists, and details are needed to be edited or deleted, a search can be conducted to find that customer.

- Clicking the Remove Entry Button will prompt a separate interface to open, which contains a view of the database, consisting of all the customers. Three search boxes can be used for searching for their forename, surname or AuthorID. If an entry was selected beforehand, then upon clicking Remove Entry, The user will be prompted for confirmation, then asked to enter their password.
- Clicking the Edit Entry button will prompt a separate interface to open, and will contain a view of the database. An entry can be searched for using the search, selected, and once the user clicks "Edit", the user will be prompted with a text box, asking for the user to enter text. Upon confirming what the user wants to enter, they are required to enter their password. This will then be saved. If a customer entry was selected beforehand, a new window for adding entries will open first upon clicking Edit Entry, and the data about the customers will be in the fields already, ready for editing. Then, the data can be edited and saved, and the user will be prompted for confirmation then asked to enter their password.

Changing Password

- An interface will open, which will prompt the user to enter their Email, Old Password, and then the new password twice for confirmation.
- Once this has been confirmed, the interface will close, resorting back to the log in window.

Search Window

- Clicking the Search Database Button will open a separate interface, which contains a set of fields that the user can use to search the database by.
- Once the Search Button is clicked, a list of all the data entries that match the search criteria will come up in a list in the Main Window.
- The search can be refined by searching again, and an item can be selected from the search results.

2.1.2 System flowcharts showing an overview of the complete system

The following is a flowchart representing a summary of the complete system.

Figure 2.1: Flowchart 1

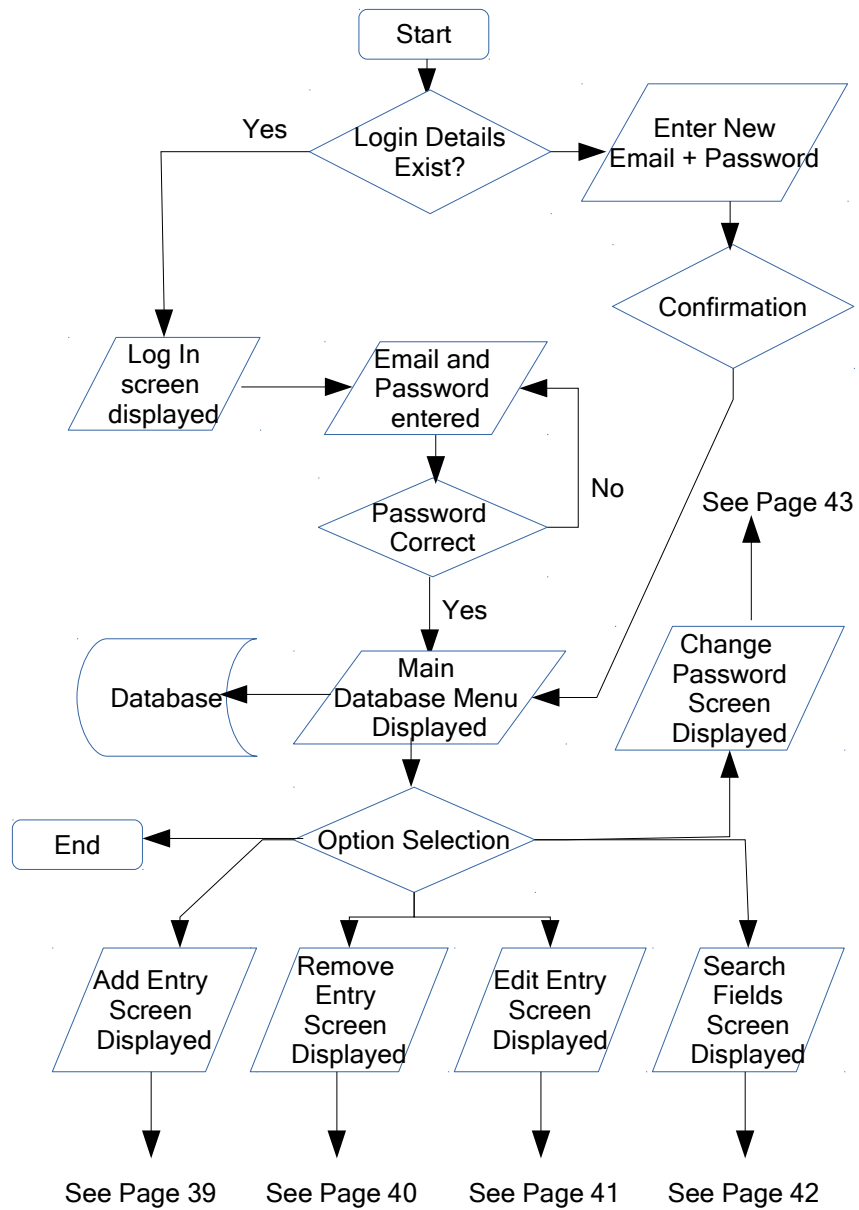


Figure 2.2: Flowchart 2

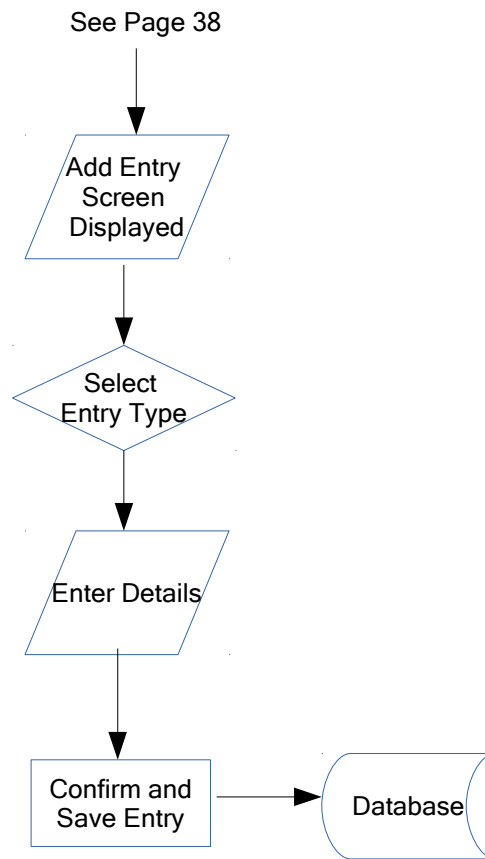


Figure 2.3: Flowchart 3

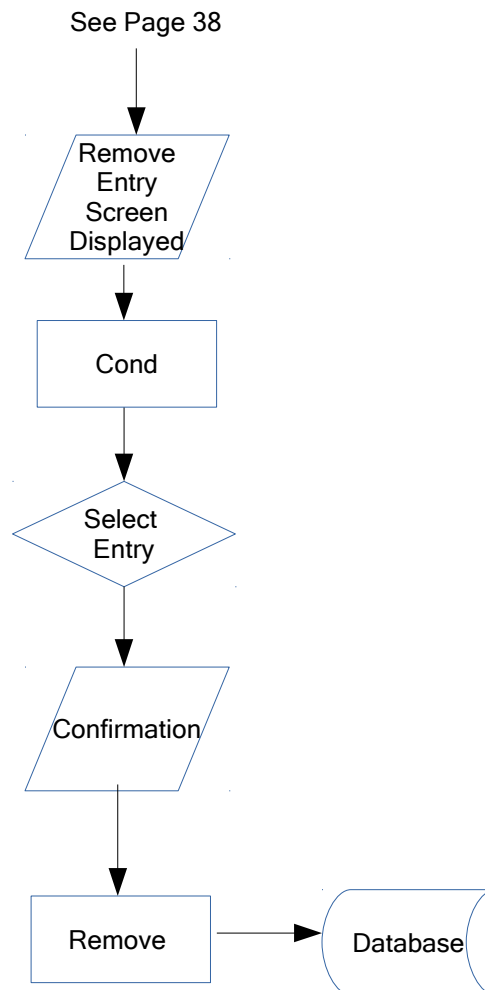


Figure 2.4: Flowchart 4

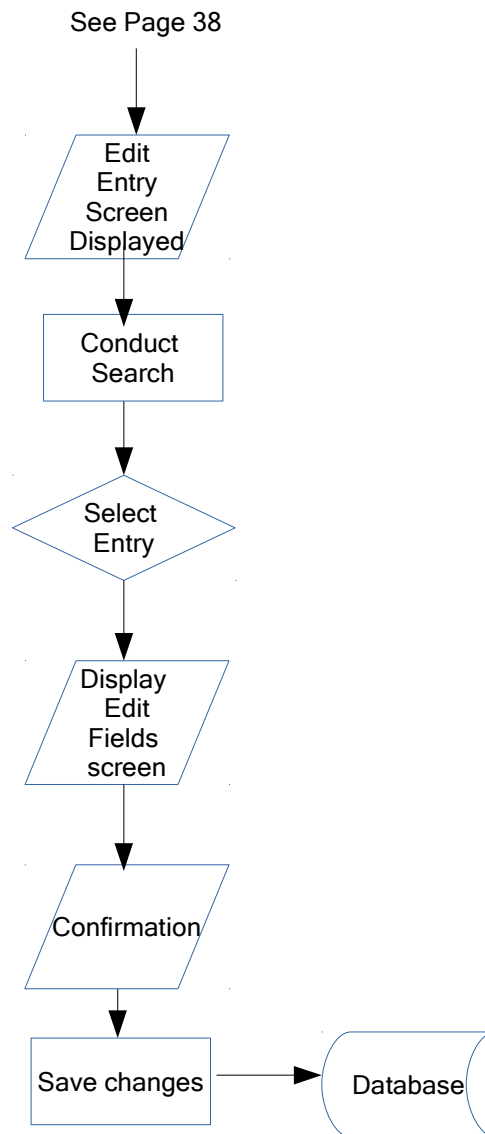


Figure 2.5: Flowchart 5

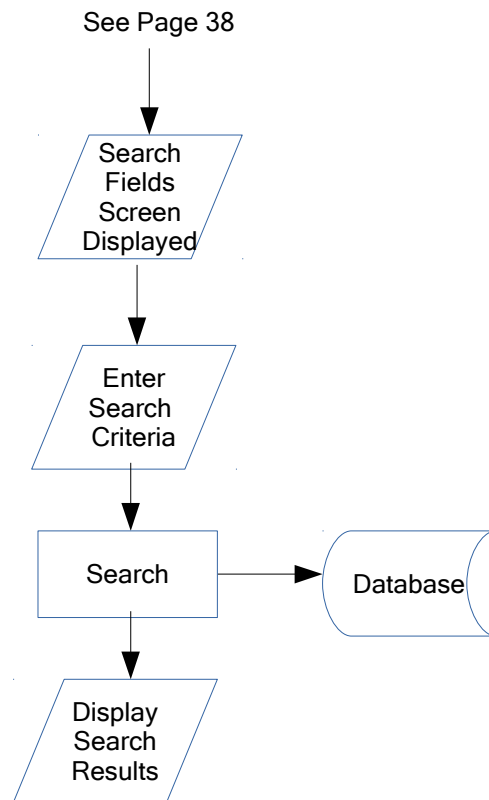
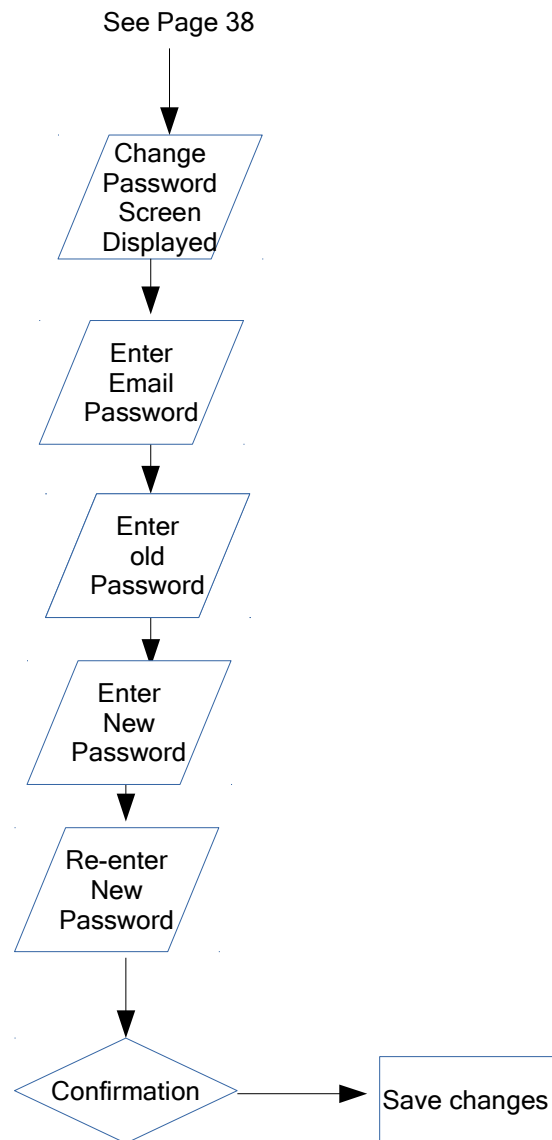


Figure 2.6: Flowchart 6



2.2 User Interface Designs

Figure 2.7: Login Screen and Main Menu

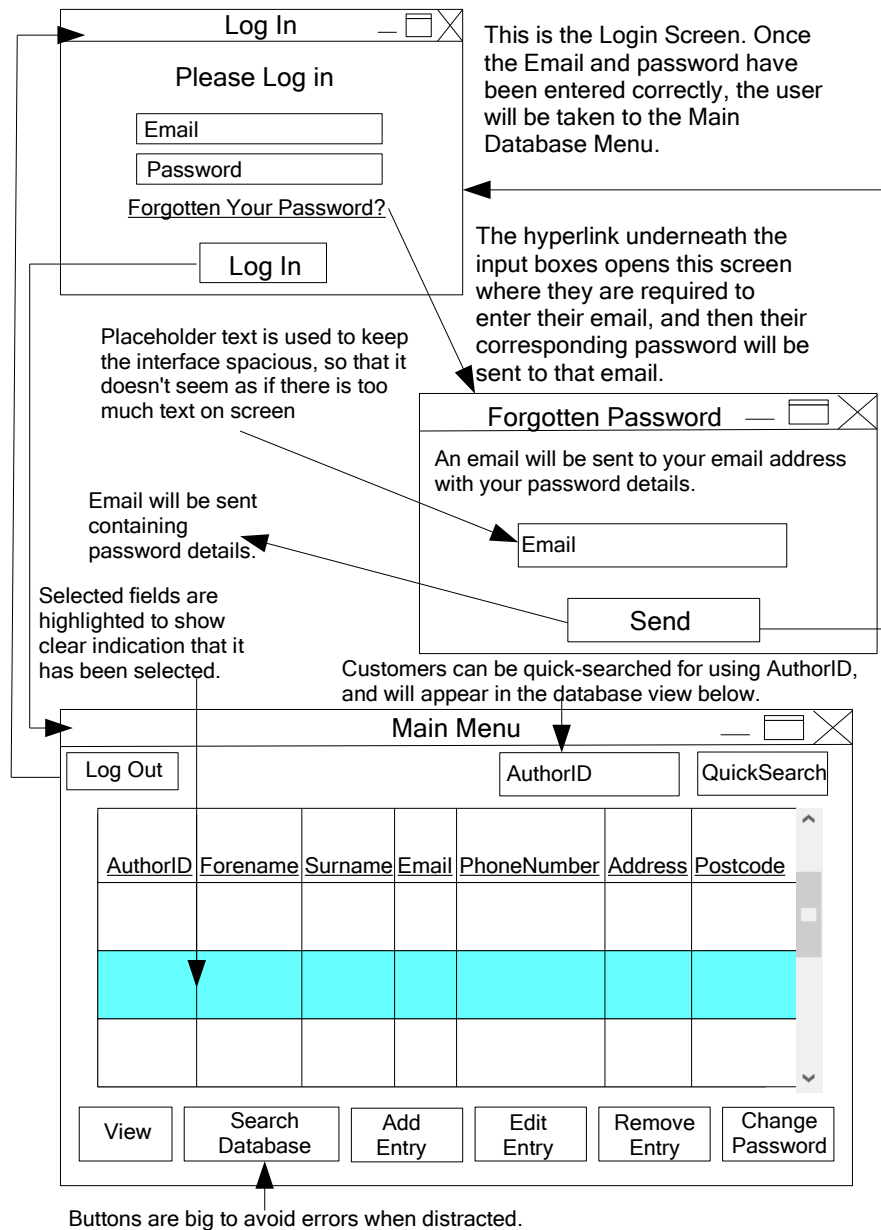


Figure 2.8: View Menu and Royalties

(From Main Menu)

View

Opens in
New window

A full view of each book from an author can be seen here, The user can see every detail about the books, or details about royalties and invoices as a whole. Buttons are spaced out to avoid clicking the wrong button.

View Menu

| AuthorID | ISBN | Book Title | NoOf Pages | Size | Back | Cover | Paper | Font | Font Size | Date Pub | Price |
|----------|------|------------|------------|------|------|-------|-------|------|-----------|----------|-------|
| | | | | | | | | | | | |
| | | | | | | | | | | | |
| | | | | | | | | | | | |

View Royalties View Book Invoices View Publishing Invoice AddBook

Opens in New window

View Royalties

| RoyaltiesID | AuthorID | RoyaltyPayment | RoyaltiesDate |
|-------------|----------|----------------|---------------|
| | | | |
| | | | |
| | | | |

View Royalties Items

Scroll bars allow lots of data to be seen without requiring large amounts of space

Figure 2.9: RoyaltiesItems and BookInvoice and Items

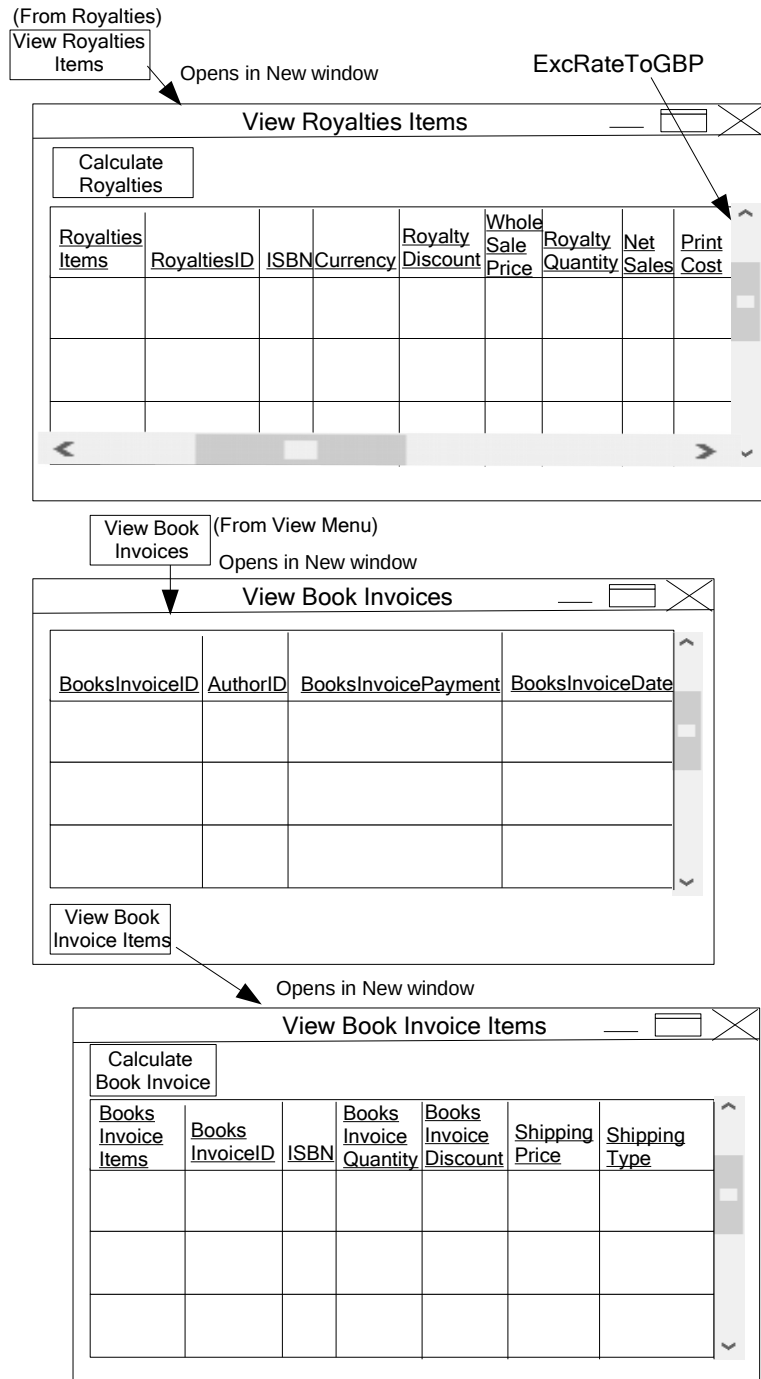


Figure 2.10: Publishing Invoice and Adding Entries

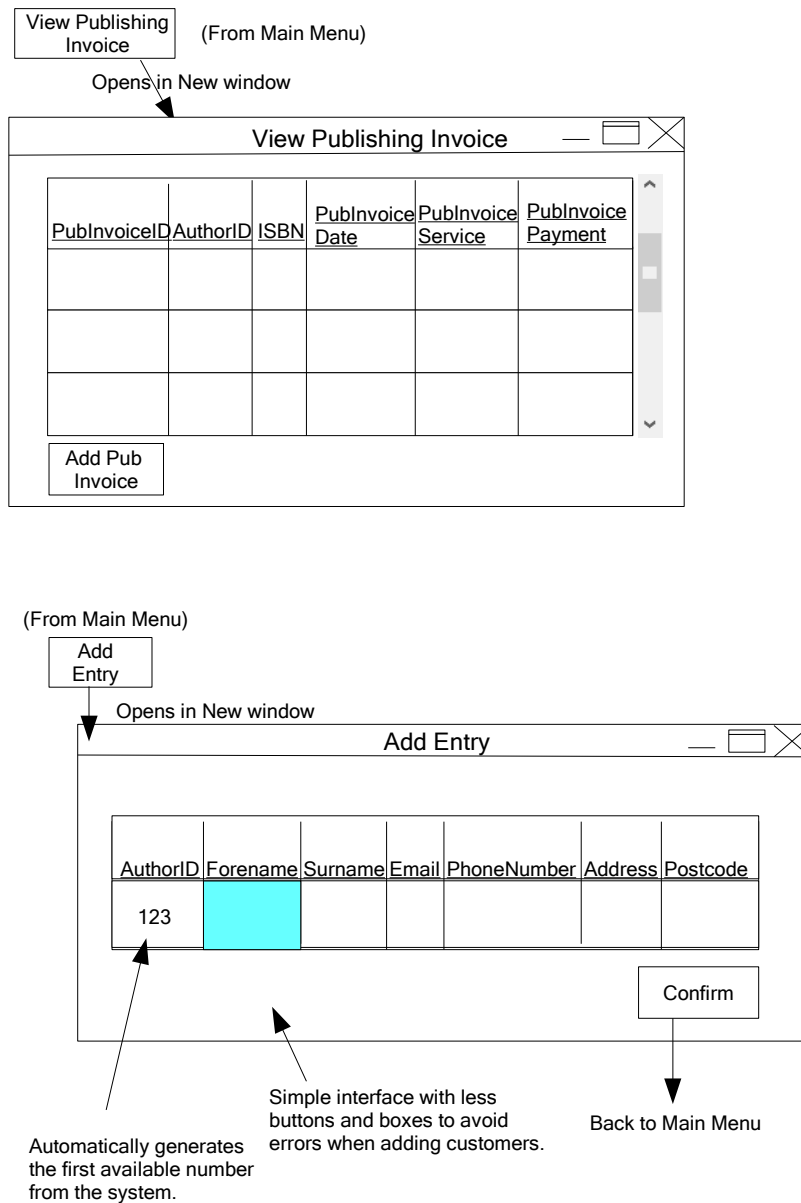


Figure 2.11: Search

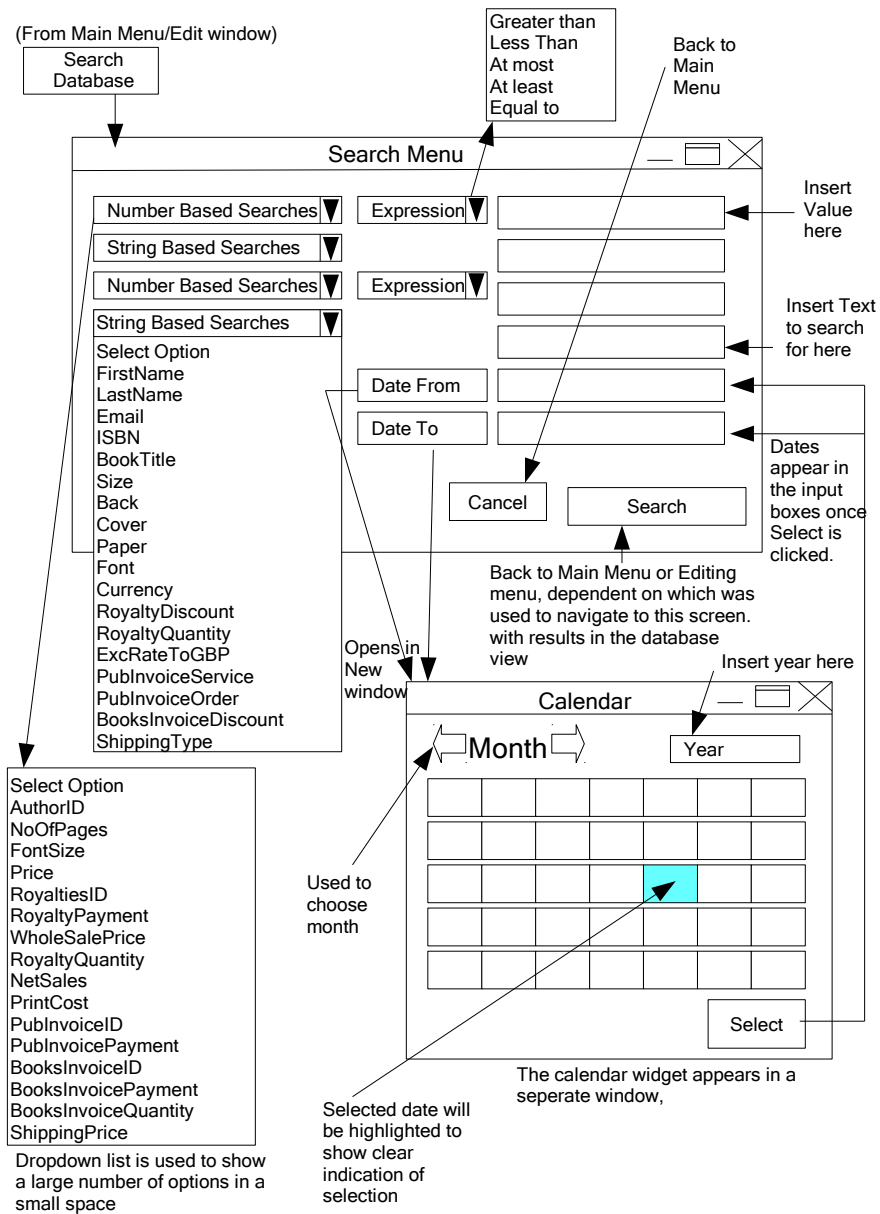


Figure 2.12: Editing Screens

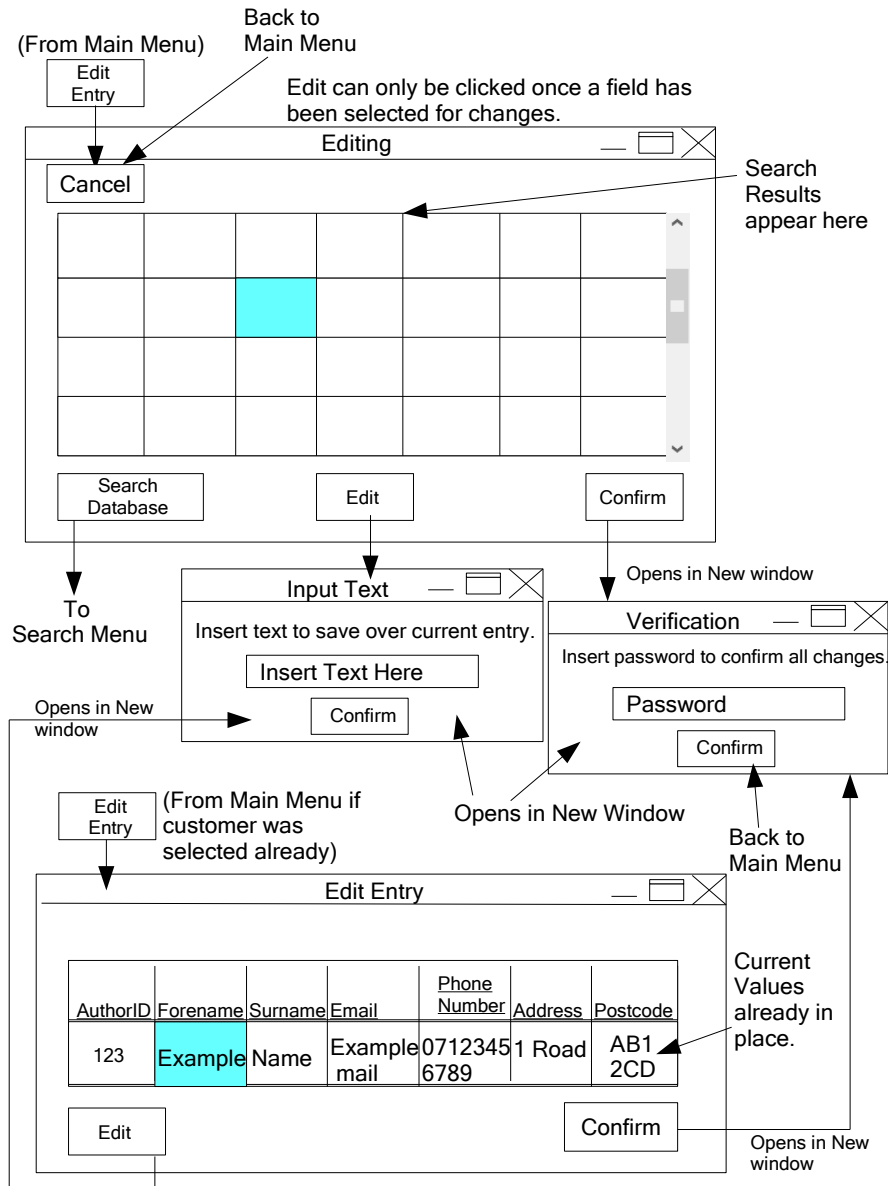
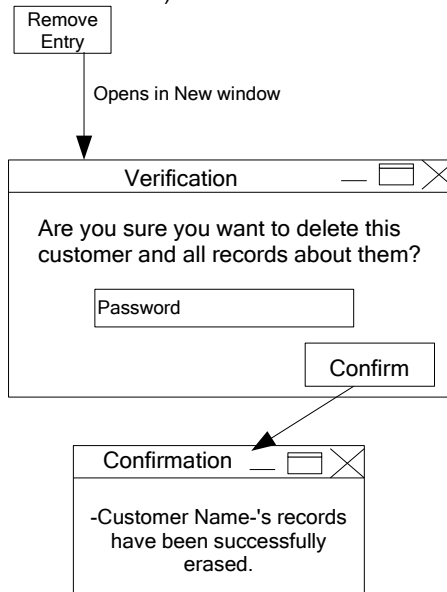


Figure 2.13: Remove Entry and Change Password

(From Main Menu)



(From Main Menu)

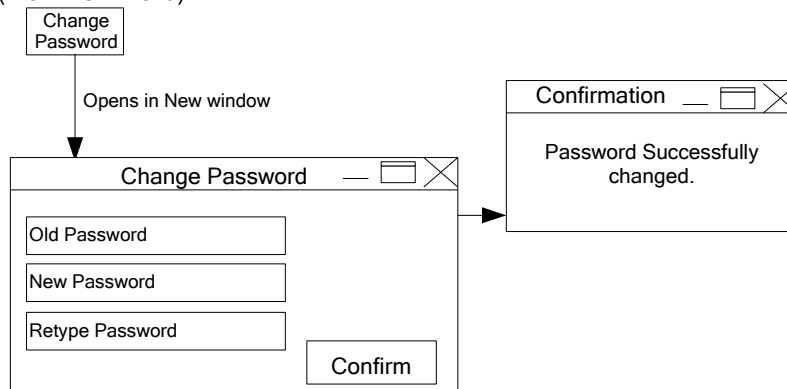


Figure 2.14: Calculations

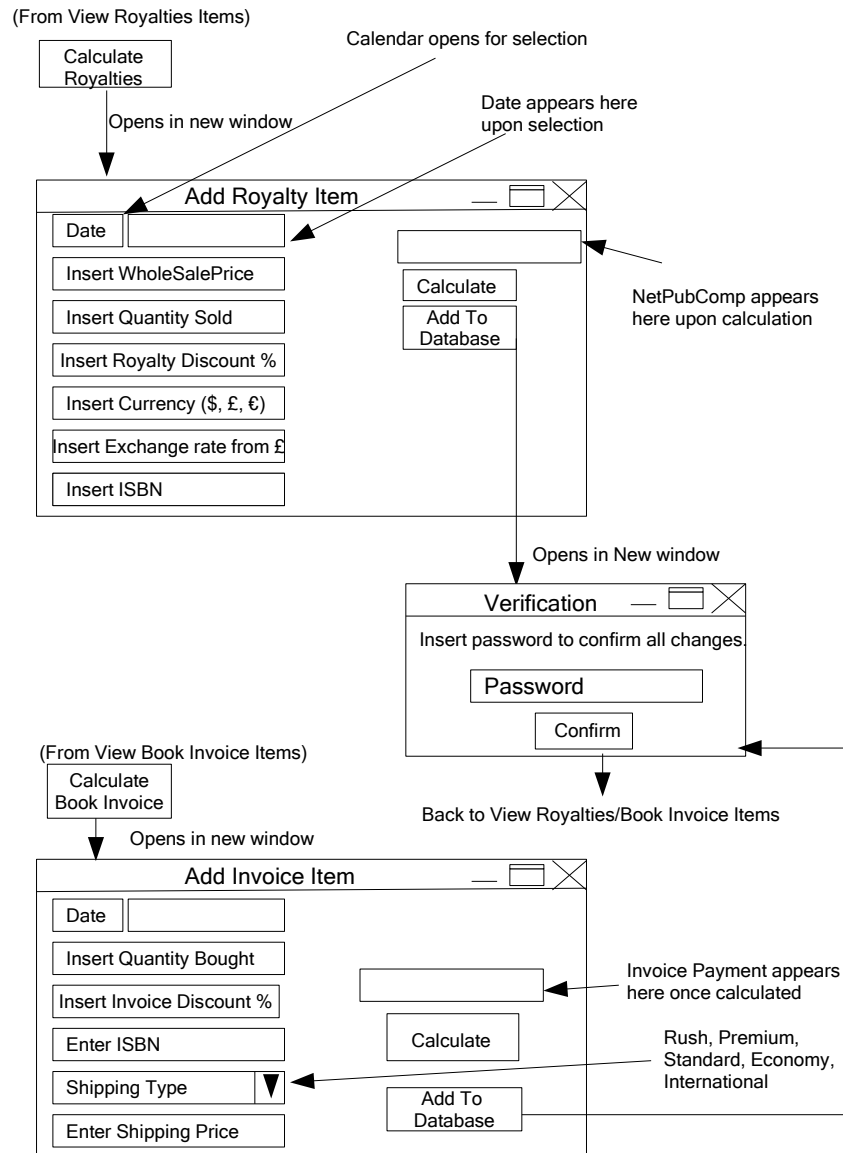


Figure 2.15: Adding Publishing Invoices

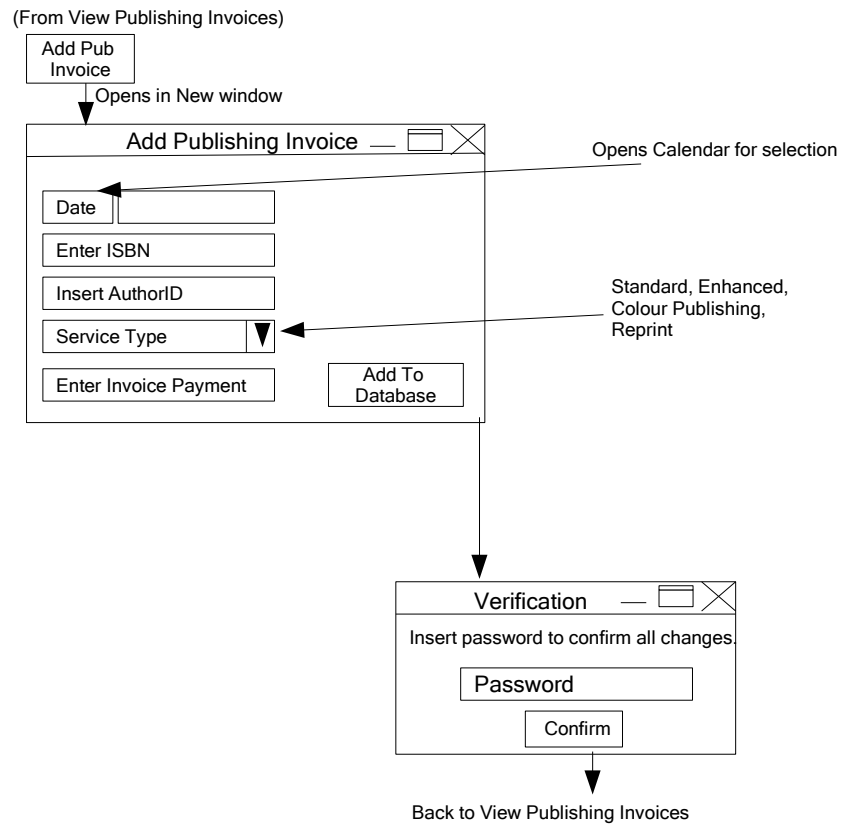
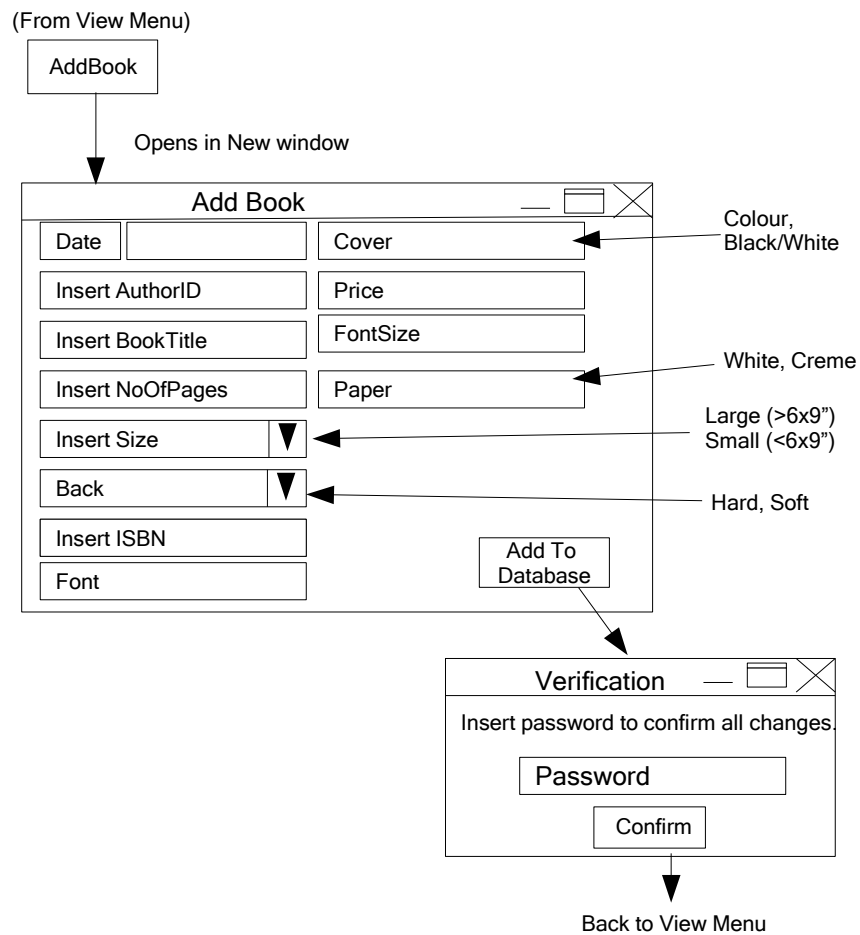


Figure 2.16: Add Book



2.3 Hardware Specification

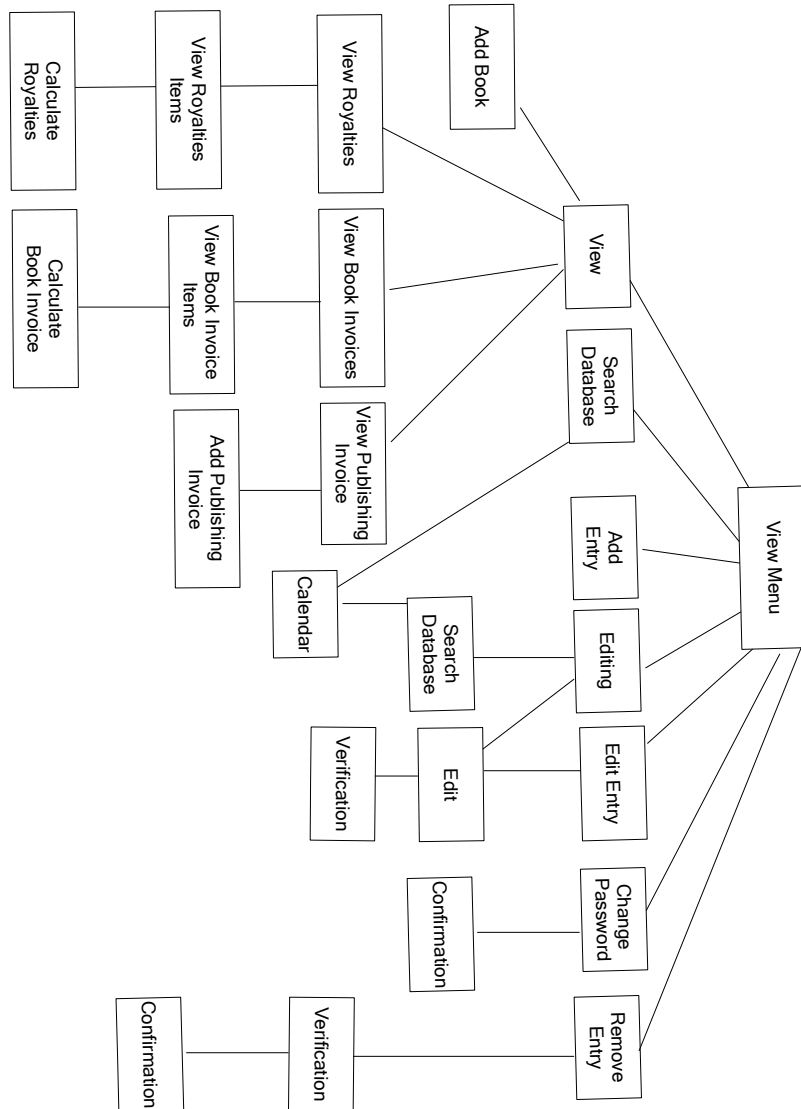
The system needs to be able to run on a laptop with a 1366 x 768, 16:9 aspect ratio screen which runs on Windows 8. This is imperative to the size of the application I will be creating because it must fit on the given screen size and can't be resizable. A mouse or touchpad will be used for navigational purposes, and for confirmation of entries. Also, a keyboard will be used for inputting information into fields for entering and editing information. All the data used by the program and its database will be held on a local hard drive, and a display is needed for the outputs of the program.

2.4 Program Structure

2.4.1 Top-down design structure charts

.

Figure 2.17: Top-down design structure chart



2.4.2 Algorithms in pseudo-code for each data transformation process

Algorithm 3 Add Customer Entry

```

1: function ADDEENTRY(CustomerTable) Loop = 1
2:   SET GetNewAuthorID TO False
3:   SET ConfirmClicked TO False NewAuthorID = 0
4:   WHILE GetNewAuthorID = false DO
      NewAuthorID = loop Get AuthorID[loop] from CustomerTable
5:   IF NewAuthorID = AuthorID[loop] THEN
      loop = loop +1
6:   ELSE
7:     SET GetNewAuthorID TO True
8:   END IF
9:   END WHILE
      FirstName = null
      LastName = null
      Email = null
      PhoneNumber = null
      Address = null
      Postcode = null
10:  WHILE len(FirstName) <= 0 DO
      INPUT FirstName
11:  END WHILE
12:  WHILE len(LastName) <= 0 DO
      INPUT LastName
13:  END WHILE
14:  WHILE len(Email) <= 0 DO
      INPUT Email
15:  END WHILE
16:  WHILE len(PhoneNumber) <= 0 DO
      INPUT PhoneNumber
17:  END WHILE
18:  WHILE len(Address) <= 0 DO
      INPUT Address
19:  END WHILE
20:  WHILE len(Postcode) <= 0 DO
      INPUT Postcode
21:  END WHILE
22:  IF ConfirmClicked == True THEN
      CONNECT to Customer Database
23:  END IF
24: END function

```

Algorithm 4 Add and Calculate Royalty Items- part 1

```

1: SET CalculateClicked TO False
2: SET AddToDatabaseClicked TO False
   Date = null
   WholeSalePrice = null
   QuantitySold = null
   Currency = null
   ExcRateFromGBP = null
   ISBN = null
   RoyaltyDiscount = null
3: WHILE len(Date) <= 0 DO
   INPUT Date
4: END WHILE
5: WHILE len(WholeSalePrice) <= 0 DO
   INPUT WholeSalePrice
6: END WHILE
7: WHILE len(QuantitySold) <= 0 DO
   INPUT LastName
8: END WHILE
9: WHILE len(Currency) <= 0 DO
   INPUT Currency
10: END WHILE
11: IF Currency <> $ THEN
12:   WHILE len(ExcRatefromGBP) <= 0 DO
   INPUT ExcRateFromGBP
13:   END WHILE
14: END IF
15: WHILE len(ISBN) <= 0 DO
   INPUT ISBN
16: END WHILE
17: WHILE len(RoyaltyDiscount) <= 0 DO
   INPUT RoyaltyDiscount
18:   IF RoyaltyDiscount > 100 THEN
   RoyaltyDiscount = null
19:   END IF
20:   IF RoyaltyDiscount < 0 THEN
   RoyaltyDiscount = null
21:   END IF
22: END WHILE

```

Algorithm 5 Add and Calculate Royalty Items- part 2

```

1: IF Cover == Colour THEN
    Size = Small
    PagePrice = 0.06 * NoOfPages
2:   IF Back == Hard THEN
    CoverPrice = 5
3:   END IF
4:   IF Back == Soft THEN
    CoverPrice = 3
5:   END IF
6: END IF
7: IF Cover == Black/White THEN
8:   IF Size == Large THEN
    PagePrice = 0.015 * NoOfPages
9:     IF Back == Hard THEN
    CoverPrice = 5
10:    END IF
11:    IF Back == Soft THEN
    CoverPrice = 1
12:    END IF
13:  END IF
14:  IF Size == Small THEN
    PagePrice = 0.01 * NoOfPages
15:    IF Back == Hard THEN
    CoverPrice = 4
16:    END IF
17:    IF Back == Soft THEN CoverPrice = 0.7
18:    END IF
19:  END IF
20: END IF
    PrintCost = (PagePrice + CoverPrice) * RoyaltyQuantity
21: IF CalculateClicked == True THEN
    NetSales = WholeSalePrice * RoyaltyQuantity
    RoyaltyPayment = NetSales - PrintCost
    OUTPUT RoyaltyPayment
22: END IF
23: IF AddToDatabaseClicked == True THEN
    CONNECT to RoyaltyItems Database
24: END IF

```

Algorithm 6 Add and Calculate Invoice Items

```

1: function ADDINVOICEITEM(BookTable)
2:   SET CalculateClicked TO False
3:   SET AddToDatabaseClicked TO False
   Date = null
   QuantityBought = null
   InvoiceDiscount = null
   ShippingType = null
   ShippingPrice = null
   ISBN = null
4:   WHILE len(Date) <= 0 DO
   INPUT Date
5:   END WHILE
6:   WHILE len(QuantityBought) <= 0 DO
   INPUT QuantityBought
7:   END WHILE
8:   WHILE len(InvoiceDiscount) <= 0 DO
   INPUT InvoiceDiscount
9:     IF InvoiceDiscount > 100 THEN
   InvoiceDiscount = null
10:    END IF
11:    IF InvoiceDiscount < 0 THEN
   InvoiceDiscount = null
12:    END IF
13:  END WHILE
14:  WHILE len(ISBN) <= 0 DO
   INPUT ISBN
15:  END WHILE
16:  WHILE len(ShippingType) <= 0 DO
   INPUT ShippingType
17:  END WHILE
18:  WHILE len(ShippingPrice) <= 0 DO
   INPUT ShippingPrice
19:  END WHILE
20:  IF CalculateClicked == True THEN
   InvoicePayment = (QuantityBought * Price * InvoiceDiscount) + Shipping Price
   OUTPUT InvoicePayment
21:  END IF
22:  IF AddToDatabaseClicked == True THEN
   CONNECT to Invoice Database
23:  END IF
24: END function

```

Algorithm 7 Add Book

```

1: function ADDINVOICEITEM(CustomerTable)
2:   SET AddToDatabaseClicked TO False
      DatePublished = null
      ISBN = null
      AuthorID = null
      BookTitle = null
      NoOfPages = null
      Size = null
      Back = null
      Cover = null
      Paper = null
      Font = null
      FontSize = null
      Price = null
3:   WHILE len(DatePublished) <= 0 DO
      INPUT DatePublished
4:   END WHILE
5:   WHILE len(ISBN) <= 0 DO
      INPUT ISBN
6:   END WHILE
7:   WHILE len(BookTitle) <= 0 DO
      INPUT BookTitle
8:   END WHILE
9:   WHILE len(Size) <= 0 DO
      INPUT Size
10:  END WHILE
11:  WHILE len(BackType) <= 0 DO
      INPUT BackType
12:  END WHILE
13:  WHILE len(NoOfPages) <= 0 DO
      INPUT NoOfPages
14:  END WHILE
15:  WHILE len(Paper) <= 0 DO
      INPUT Paper
16:  END WHILE
17:  WHILE len(Font) <= 0 DO
      INPUT Font
18:  END WHILE
19:  WHILE len(FontSize) <= 0 DO
      INPUT FontSize
20:  END WHILE
21:  WHILE len(Price) <= 0 DO
      INPUT Price
22:  END WHILE
23:  IF AddToDatabaseClicked == True THEN
      CONNECT to Book Database
24:  END IF
25: END function

```

Algorithm 8 Add and Publishing Invoice

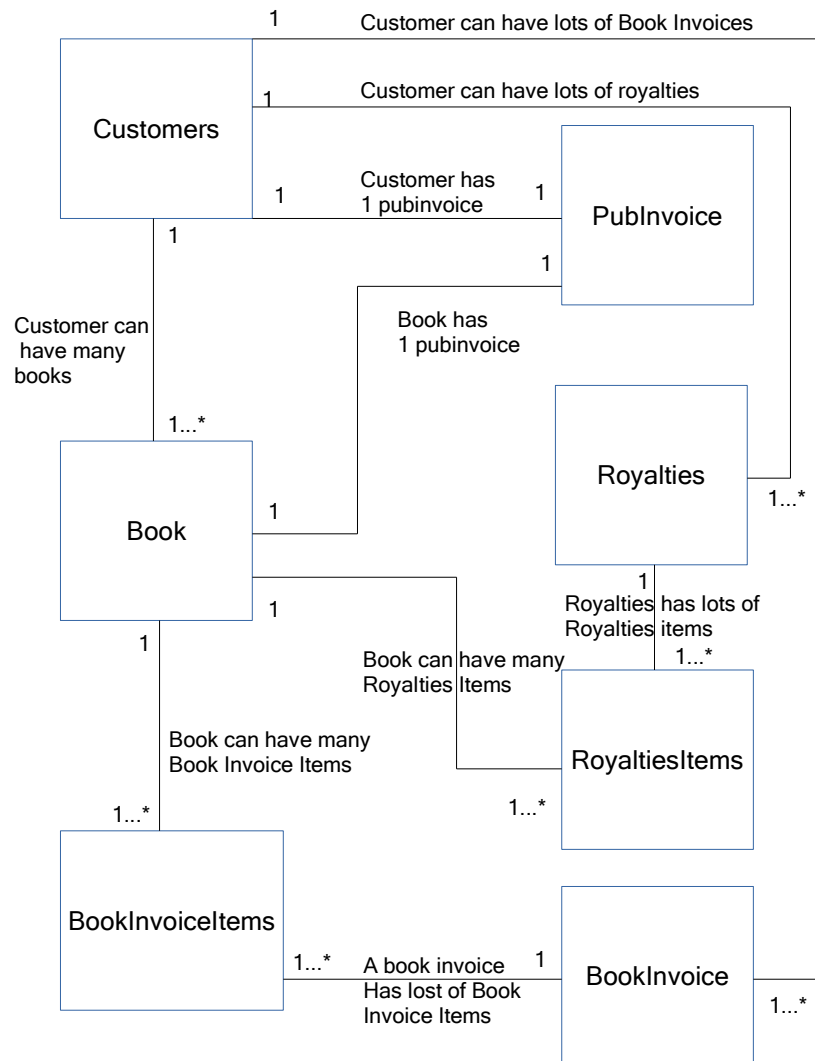
```

1: function ADDINVOICEITEM(BookTable, CustomerTable)
2:   SET AddToDatabaseClicked TO False
      Date = null
      AuthorID = null
      ServiceType = null
      InvoicePayment = null
      ISBN = null
3:   WHILE len(Date) <= 0 DO
      INPUT Date
4:   END WHILE
5:   WHILE len(AuthorID) <= 0 DO
      INPUT AuthorID
6:   END WHILE
7:   WHILE len(InvoicePayment) <= 0 DO
      INPUT InvoicePayment
8:   END WHILE
9:   WHILE len(ISBN) <= 0 DO
      INPUT ISBN
10:  END WHILE
11:  WHILE len(ServiceType) <= 0 DO
      INPUT ServiceType
12:  END WHILE
13:  IF AddToDatabaseClicked == True THEN
      CONNECT to PubInvoice Database
14:  END IF
15: END function

```

2.4.3 Object Diagrams

Figure 2.18: Object Diagram



2.4.4 Class Definitions

Key:

| Label |
|------------|
| Attributes |
| Behaviours |

| Customer |
|--|
| Author ID ForeName Surname Email Address Postcode Phone Number |
| Add ForeName Edit Forename Add Surname Edit Surname Add Email Edit Email Add Address Edit Address Add Postcode Edit Postcode Add Phone Number Edit Phone Number |

| Book |
|--------------------|
| ISBN |
| AuthorID |
| BookTitle |
| NoOfPages |
| Size |
| Cover |
| Back |
| Paper |
| Font |
| FontSize |
| Price |
| DatePublished |
| Add ISBN |
| Edit ISBN |
| Add AuthorID |
| Edit AuthorID |
| Add BookTitle |
| Edit BookTitle |
| Add NoOfPages |
| Edit NoOfPages |
| Add Size |
| Edit Size |
| Add Cover |
| Edit Cover |
| Add Back |
| Edit Back |
| Add Paper |
| Edit Paper |
| Add Font |
| Edit Font |
| Add FontSize |
| Edit FontSize |
| Add DatePublished |
| Edit DatePublished |
| Add Price |
| Edit Price |

| Royalties |
|---|
| RoyaltiesID AuthorID RoyaltyPayment RoyaltiesDate |
| Add AuthorID Edit AuthorID Add RoyaltiesDate Edit Royalties Date |

| Royalties Items |
|---|
| RoyaltiesItems RoyaltiesID ISBN Currency RoyaltyDiscount WholesalePrice RoyaltyQuantity NetSales PrintCost ExcRateFromGBP |
| Add RoyaltiesID Edit RoyaltiesID Add ISBN Edit ISBN Add Currency Edit Currency Add RoyaltyDiscount Edit RoyaltyDiscount Add WholesalePrice Edit WholesalePrice Add RoyaltyQuantity Edit RoyaltyQuantity Add ExcRateFromGBP Edit ExcRateFromGBP |

| BookInvoice |
|---|
| BookInvoiceID AuthorID BookInvoicePayment BookInvoiceDate |
| Add AuthorID Edit AuthorID Add BookInvoiceDate Edit BookInvoiceDaate |

| BookInvoiceItems |
|---|
| BookInvoiceItems BookInvoiceID ISBN BookInvoiceQuantity BookInvoiceDiscount ShippingType ShippingPrice |
| Add BookInvoiceID Edit BookInvoiceID Add ISBN Edit ISBN Add BookInvoiceQuantity Edit BookInvoiceQuantity Add BookInvoiceDiscount Edit BookInvoiceDiscount Add ShippingType Edit Shipping Type Add ShippingPrice Edit ShippingPrice |

| PubInvoice |
|--|
| PubInvoiceID AuthorID ISBN PubInvoiceDate PubInvoiceService PubInvoicePayment |
| Add AuthorID Edit AuthorID Add ISBN Edit ISBN Add PubInvoiceDate Edit PubInvoiceDate Add PubInvoiceService Edit PubInvoiceService Add PubInvoicePayment EditPubInvoicePayment |

2.5 Prototyping

It will be helpful to create a prototype of the main menu to make sure that different windows can be navigated through without any difficulty. This would give me a good idea of the flow of control between interfaces. Also, I plan to

prototype a log in screen, because this would help me identify the difficulties involved in moving to and from the main menu and log in screen.

Furthermore, I plan to prototype the adding, editing and removal of data to and from the database. I am going to prototype this in order to make sure that data can be successfully added, edited and removed to and from the database so that it can be confirmed that this can be conducted upon creation of the program.

I have already prototyped the calendar interface used for inputting the date in a box. This is just a prototype of the basic interface, which is for selecting a date and placing it in the text box. It can switch between months with a dropdown list, or using the arrows and also years.

Figure 2.19: Calendar Interface

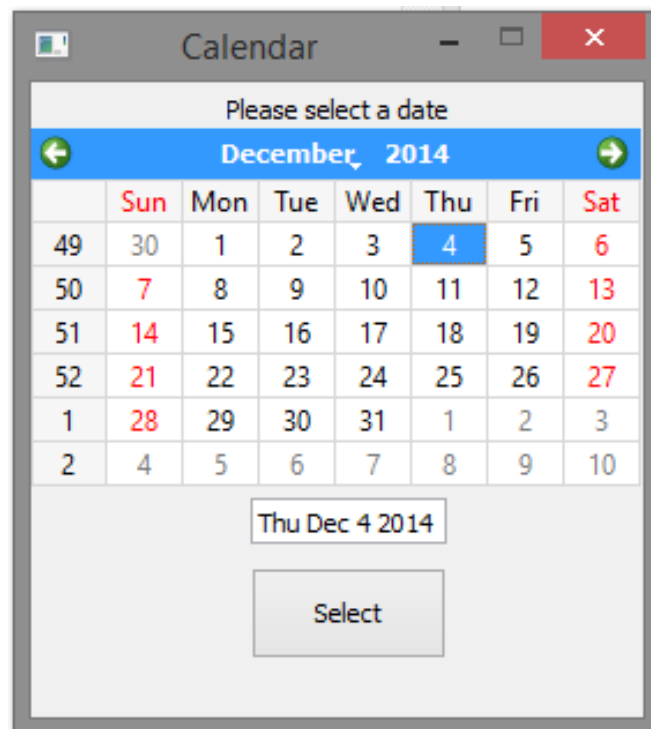
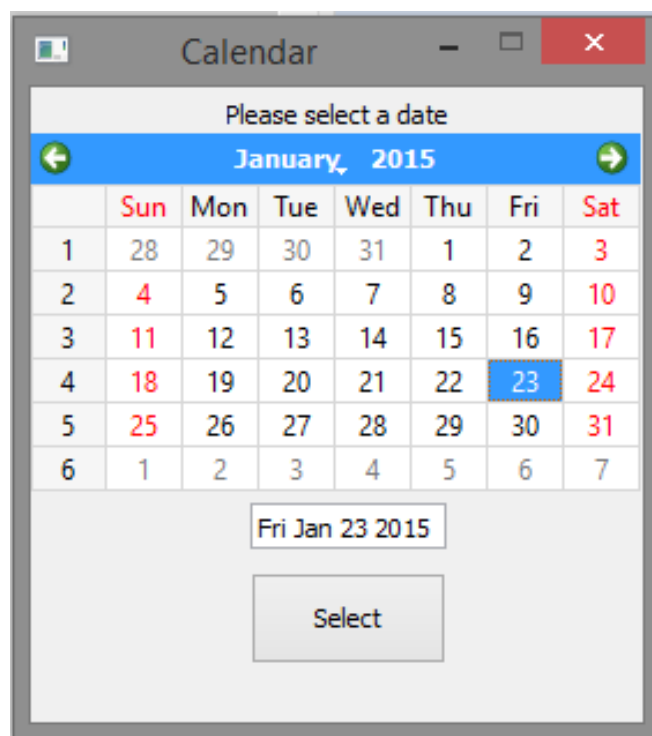


Figure 2.20: Calendar - Date Selection



2.6 Definition of Data Requirements

2.6.1 Identification of all data input items

- First Name
- Last Name
- Email
- Phone Number
- Address
- Postcode
- ISBN
- Book Title
- Number of pages
- Size
- Back type
- Cover type
- Paper type
- Font
- Font size
- Date published
- Price
- Currency
- Royalties date
- Royalty discount
- Wholesale price
- Royalty Quantity
- Exchange rate from pounds
- Publishing invoice date
- Publishing invoice service
- Publishing invoice payment
- Book invoice date

- Book invoice discount
- Book invoice quantity
- Shipping price
- Shipping type

2.6.2 Identification of all data output items

- Author ID
- Royalties ID
- Publishing Invoice ID
- Books Invoice ID
- Net Sales
- Print Cost
- Royalty Payment
- Net Publishing Compensation
- Book Invoice Payment
- Royalties Items
- Book Invoice Items

2.6.3 Explanation of how data output items are generated

| Output | How it is produced | Input Items Required to Produce |
|-----------------------------|--|---|
| Author ID | Produced by program | |
| Royalties ID | Produced by program | |
| Publishing Invoice ID | Produced by program | |
| Book Invoice ID | Produced by program | |
| Net Sales | Calculated by program using inputs | Wholesale Price, Royalty Quantity |
| Print Cost | Calculated by program using inputs | Number of Pages, Size, Back type, Cover type, Royalty Quantity |
| Net Publishing Compensation | Calculated by program using inputs/outputs | Net Sales, Print Cost |
| Royalty Payment | Calculated by program using outputs | Net Publishing Compensation |
| Book Invoice Payment | Calculated by program using inputs | Book Invoice Quantity, Book Invoice Discount, Price, Shipping Price |
| Royalties Items | Produced by program | |
| Book Invoice Items | Produced by program | |

2.6.4 Data Dictionary

There have been some changes to my data dictionary as a number of elements have been identified that need to be added to the system.

| Name | Data Type | Length | Validation | Example Data |
|-----------------|-----------|-------------------|----------------------------|------------------|
| FirstName | String | 2-20 Characters | Length | Jo |
| LastName | String | 2-20 Characters | Length | Williamson |
| Email | String | 7-30 Characters | Length | mail@example.com |
| PhoneNumber | String | 9-15 Characters | Format | 07123456789 |
| Address | String | 5-64 Characters | Length | Example Road |
| Postcode | String | 7 Characters | Format | AB1 2CD |
| AuthorID | Integer | 1-255 | Range | 17 |
| ISBN | String | 13 Characters | Length | 9780007525492 |
| BookTitle | String | 1-127 Characters | Length | The Hobbit |
| NoOfPages | Integer | 1-1023 | Range | 395 |
| Size | String | 5 | Existence | Large |
| Back | String | 8 or 9 Characters | Existence | Paperback |
| Cover | String | 3 or 5 Characters | Existence | Gloss |
| Paper | String | 11 Characters | Existence | White Paper |
| Font | String | 1-64 Characters | Length | Arial |
| FontSize | Real | 8-64 | Numbers only | 12.5 |
| DatePublished | Date | dd/mm/yyyy | Range | 23/10/2014 |
| Price | Real | 0.01-63.00 | Numbers only | £12.99 |
| RoyaltiesID | Integer | 1-255 | Numbers only | 123 |
| RoyaltiesItems | Integer | 1-511 | Numbers only | 12 |
| Currency | String | 1 Character | Pound, Dollar or Euro sign | £ |
| RoyaltyPayment | Real | 1-32767 | Numbers only | 489.92 |
| RoyaltiesDate | Date | dd/mm/yyyy | Range | 03/12/2014 |
| RoyaltyDiscount | Real | 0-100 | Numbers only | 40 |
| WholeSalePrice | Real | 0.01-63.00 | Numbers only | 7.99 |
| RoyaltyQuantity | Integer | 1- 2047 | Numbers only | 192 |

| Name | Data Type | Length | Validation | Example Data |
|---------------------|-----------|------------------|--------------|--------------|
| NetSales | Real | 0.01-32767.00 | Numbers only | 900.00 |
| PrintCost | Real | 0.01-32767.00 | Numbers only | 800.00 |
| ExcRateFromGBP | Real | 0-1027 | Numbers only | 1.67 |
| PubInvoiceID | Integer | 1-511 | Numbers only | 123 |
| PubInvoiceDate | Date | dd/mm/yyyy | Range | 23/05/2014 |
| PubInvoiceService | String | 1-127 Characters | Length | Standard |
| PubInvoicePayment | Real | 0.01-32767.00 | Numbers only | 700.00 |
| BookInvoiceID | Integer | 1-255 | Numbers only | 99 |
| BookInvoiceItems | Integer | 1-255 | Numbers only | 2 |
| BookInvoicePayment | Real | 0.01-32767.00 | Numbers only | 600.00 |
| BookInvoiceDate | Date | dd/mm/yyyy | Range | 01/01/2014 |
| BookInvoiceDiscount | Real | 0-100 | Numbers only | 50 |
| BookInvoiceQuantity | Integer | 1-2047 | Numbers only | 777 |
| Shipping Type | String | 7-8 Characters | Range | Premium |
| Shipping Price | Real | 0.01-64.00 | Numbers only | 25.00 |

2.6.5 Identification of appropriate storage media

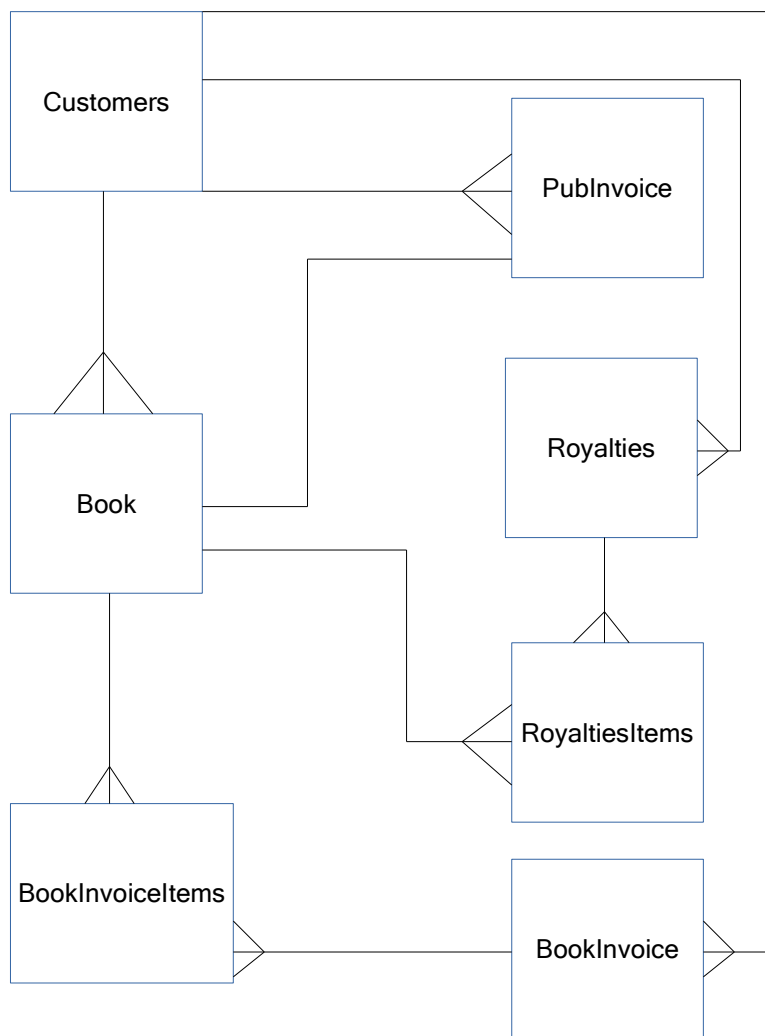
A hard drive would be an example of an appropriate storage media. For backup purposes, an external hard drive would be the optimal solution, because it is portable, meaning it can be kept off site, as well as being able to hold all data files on it. The database will be needed to be kept for long term storage. This is the data will be required to be accessed a number of times.

2.7 Database Design

2.7.1 Normalisation

ER Diagrams

Figure 2.21: ER Diagram



Entity Descriptions

Customer(Author ID, FirstName, LastName, Email, Address, Postcode, Phone Number)

Book(ISBN, *AuthorID*, Book Title, NoOfPages, Size, Cover, Paper, Back, Paper, Font, FontSize, DatePublished, Price)

BookInvoice(BookInvoiceID, *AuthorID*, InvoiceDate, InvoicePayment)

BookInvoiceItems(BookInvoiceItems, *BookInvoiceID*, *ISBN*, BooksInvoiceQuantity, BooksInvoiceDiscount, ShippingPrice, ShippingType)

Royalties(RoyaltiesID, *AuthorID*, RoyaltiesDate, RoyaltyPayment)

RoyaltiesItems(RoyaltiesItems, *RoyaltiesID*, *ISBN*, Currency, RoyaltyDiscount, WholesalePrice, RoyaltyQuantity, NetSales, PrintCost, ExcRateFromGBP)

PubInvoice(PubInvoiceID, *AuthorID*, *ISBN*, PubInvoiceDate, PubInvoiceOrder, PubInvoicePayment)

UNF to 3NF

Key:

Bold Font = Primary Key

Italics = Foreign Key

Each Column represents a new group.

First of all, I have started with the data in its unnormalised form.

| |
|---------------------|
| FirstName |
| LastName |
| Email |
| PhoneNumber |
| Address |
| PostCode |
| AuthorID |
| ISBN |
| BookTitle |
| NoOfPages |
| Size |
| Back |
| Cover |
| Paper |
| Font |
| FontSize |
| DatePublished |
| Price |
| RoyaltiesID |
| RoyaltiesItems |
| Currency |
| RoyaltyPayment |
| RoyaltiesDate |
| RoyaltyDiscount |
| WholeSalePrice |
| RoyaltyQuantity |
| NetSales |
| PrintCost |
| ExcRateFromGBP |
| PubInvoiceID |
| PubInvoicePayment |
| PubInvoiceDate |
| PubInvoiceService |
| BookInvoiceID |
| BookInvoiceItems |
| BookInvoicePayment |
| BookInvoiceDate |
| BookInvoiceDiscount |
| BookInvoiceQuantity |
| ShippingType |
| ShippingPrice |

Then, I put it into the first normal form, separating the repeating attributes and the non-repeating attributes.

| AuthorID | ISBN |
|-----------------|---------------------|
| FirstName | AuthorID |
| LastName | BookTitle |
| Email | NoOfPages |
| PhoneNumber | Size |
| Address | Back |
| PostCode | Cover |
| | Paper |
| | Font |
| | FontSize |
| | DatePublished |
| | Price |
| | RoyaltiesID |
| | RoyaltiesItems |
| | Currency |
| | RoyaltyPayment |
| | RoyaltiesDate |
| | RoyaltyDiscount |
| | WholeSalePrice |
| | RoyaltyQuantity |
| | NetSales |
| | PrintCost |
| | ExcRateFromGBP |
| | PubInvoiceID |
| | PubInvoiceDate |
| | PubInvoiceService |
| | PubInvoicePayment |
| | BookInvoiceID |
| | BookInvoiceItems |
| | BookInvoicePayment |
| | BookInvoiceDate |
| | BookInvoiceDiscount |
| | BookInvoiceQuantity |
| | ShippingType |
| | ShippingPrice |

After that, I put it into the second normal form.

| | | |
|-----------------|---------------------|---------------|
| AuthorID | ISBN | ISBN |
| FirstName | AuthorID | BookTitle |
| LastName | RoyaltiesID | NoOfPages |
| Email | Currency | Size |
| PhoneNumber | RoyaltyPayment | Back |
| Address | RoyaltiesDate | Cover |
| PostCode | RoyaltyDiscount | Paper |
| | WholeSalePrice | Font |
| | RoyaltyQuantity | FontSize |
| | NetSales | DatePublished |
| | PrintCost | Price |
| | ExcRateFromGBP | |
| | PubInvoiceID | |
| | PubInvoiceDate | |
| | PubInvoiceService | |
| | PubInvoicePayment | |
| | BookInvoiceID | |
| | BookInvoiceItems | |
| | BookInvoicePayment | |
| | BookInvoiceDate | |
| | BookInvoiceDiscount | |
| | BookInvoiceQuantity | |
| | ShippingType | |
| | ShippingPrice | |

Finally, I put the data into its third normal form.

| AuthorID | ISBN | RoyaltiesID | RoyaltiesItems |
|-----------------|-----------------|--------------------|-----------------------|
| FirstName | <i>AuthorID</i> | <i>AuthorID</i> | <i>RoyaltiesID</i> |
| LastName | BookTitle | RoyaltyPayment | <i>ISBN</i> |
| Email | NoOfPages | RoyaltiesDate | Currency |
| PhoneNumber | Size | | RoyaltyDiscount |
| Address | Back | | WholeSalePrice |
| PostCode | Cover | | RoyaltyQuantity |
| | Paper | | NetSales |
| | Font | | PrintCost |
| | FontSize | | ExcRateFromGBP |
| | DatePublished | | |
| | Price | | |

| PubInvoiceID | BooksInvoiceID | BooksInvoiceItems |
|---------------------|-----------------------|--------------------------|
| <i>AuthorID</i> | <i>AuthorID</i> | <i>BooksInvoiceID</i> |
| <i>ISBN</i> | BookInvoicePayment | <i>ISBN</i> |
| PubInvoiceDate | BookInvoiceDate | BookInvoiceQuantity |
| PubInvoiceService | | BookInvoiceDiscount |
| PubInvoicePayment | | ShippingType |
| | | ShippingPrice |

2.7.2 SQL Queries

I am using Python to format the SQL query text strings.

| SQL | Descriptions |
|--|---|
| <pre> """insert into Customer(FirstName, LastName, Email, PhoneNumber, Address, Post- code) values ({0}, {1}, {2}, {3}, {4}, {5}) """format(FirstName, LastName, Email, PhoneNumber, Address, Post- code) </pre> | <p>An example of an SQL statement which adds customer records to the database. Here, it is entering a new customer record with the attributes: FirstName, LastName, Email, PhoneNumber, Address and Postcode.</p> |
| <pre> """create table RoyaltiesItems(RoyaltiesID INTEGER, Currency REAL, RoyaltyDiscount STRING, WholeSalePrice REAL, RoyaltyQuantity INTEGER, NetSales REAL, PrintCost REAL, ExcRateFromGBP STRING PRIMARY KEY(RoyaltiesItems) FOREIGN KEY(RoyaltiesID) REF- ERENCES Royalties(RoyaltiesID) """ </pre> | <p>An example of an SQL statement that creates a new table for the Royalties. There is a primary key which is RoyaltiesItems, and there is one foreign key, which is RoyaltiesID.</p> |
| <pre> """select Customer.LastName, Book.BookTitle from Customer, Book where Price < 13.00 and Back = "Paperback" </pre> | <p>This statement will return all the LastNames and the BookTitles from the Customer table and the Book table whose book is paperback and costs less than £13.</p> |

2.8 Security and Integrity of the System and Data

2.8.1 Security and Integrity of Data

The system will store personal data about the customers and will comply to the Data Protection Act. This means that the data must be frequently updated in order to keep it up to date, so there will be a way to edit and change the information using the program. All the data in the database must be kept securely, so that it can only be granted access to someone with the use of a password. To ensure that all data stored is valid, everytime the user uses the keyboard, a check will be completed to make sure that the data is valid. I need make checks for the addition and removals of data, so that all records will have the sufficient key data.

2.8.2 System Security

The database will be password protected to make sure that only users who know the password can access the database. This will keep the number of users of the database to a minimum. This can prevent the data from being tampered with or stolen. The database will be encrypted in order to avoid unwanted people having access to the data without the use of the system., therefore the number of people who can access the data can be determined beforehand.

2.9 Validation

The system will check to make sure each entry is valid, in order to avoid any invalid entries into the database.

| Item | Example | Validation | Comments |
|-------------|------------------------|------------------------------|--|
| FirstName | John | Presence Check, Type Check | Makes sure that a name is entered, and that the name is just letters. |
| LastName | Smith | Presence Check, Type Check | Makes sure that a name is entered, and that the name is just letters. |
| Email | test@mail.com | Presence Check, Type Check | Makes sure that an email is entered and that the @ symbol comes before the '.' in the email. |
| PhoneNumber | 07123456789 | Presence Check | Makes sure that a phone number is entered. |
| Address | 1 Example Road | Presence Check | Makes sure that an address is entered. |
| Postcode | AB1 2CD | Presence Check | Makes sure that a Postcode is entered. |
| ISBN | 9780748782987 | Presence Check, Length Check | Makes sure that an ISBN is entered and is not longer than 13 digits. |
| BookTitle | Computing A2 Text Book | Presence Check | Makes sure that a Book Title has been entered. |
| NoOfPages | 395 | Presence Check, Type Check | Makes sure that a positive integer has been entered. |
| Size | Large | Presence Check | Makes sure that a Size has been entered. |
| Back | Hard | Presence Check | Makes sure that a Back type has been entered. |
| Cover | Colour | Presence Check | Makes sure that a Cover type has been entered. |
| Paper | White | Presence Check | Makes sure that a Paper type has been entered. |
| Font | Microsoft Sans Serif | Presence Check | Makes sure that a Font has been entered. |

| Item | Example | Validation | Comments |
|---------------------|-------------|---|---|
| FontSize | 12 | Presence Check, Type Check | Makes sure that a positive value has been entered, and just digits. |
| DatePublished | 25 Dec 2014 | Presence Check, Type Check | Makes sure that a Date has been entered. |
| Price | 8.99 | Presence Check, Type Check | Makes sure that a positive value has been entered, and just digits. |
| Currency | £ | Presence Check, Type Check | Makes sure that the correct symbol has been entered. |
| RoyaltiesDate | 25 Dec 2014 | Presence Check, Type Check | Makes sure that a Date has been entered. |
| RoyaltyDiscount | 50 | Presence Check, Type Check, Range Check | Makes sure that a positive value has been entered, just digits, and it must be between 0 and 100. |
| WholesalePrice | 5.99 | Presence Check, Type Check | Makes sure that a positive value has been entered, and just digits. |
| RoyaltyQuantity | 150 | Presence Check, Type Check | Makes sure that a positive integer has been entered. |
| ExcRate FromGBP | 1.67 | Presence Check, Type Check | Makes sure that a value has been entered. |
| Pub Invoice Date | 25 Dec 2014 | Presence Check, Type Check | Makes sure that a Date has been entered. |
| Pub Invoice Service | Standard | Presence Check | Makes sure that a Service has been entered. |
| Pub Invoice Payment | £1659.99 | Presence Check, Type Check | Makes sure that a positive value has been entered. |

| Item | Example | Validation | Comments |
|-----------------------|-------------|---|---|
| Book Invoice Date | 25 Dec 2014 | Presence Check, Type Check | Makes sure that a Date has been entered. |
| Book Invoice Discount | 50 | Presence Check, Type Check, Range Check | Makes sure that a positive value has been entered, just digits, and it must be between 0 and 100. |
| Book Invoice Quantity | 25 | PresenceCheck, Type Check | Makes sure that a positive integer has been entered. |
| Shipping Type | Premium | Presence Check | Makes sure that a Shipping type has been entered. |
| Shipping Price | 4.00 | Presence Check, Type Check | Makes sure that a positive value has been entered, and just digits. |
| AuthorID | 21 | Presence Check, Type Check, Look up Check | Makes sure that an integer has been entered, and will check the Customer Table to find the entry to confirm that it is valid. |
| RoyaltiesID | 21 | Presence Check, Type Check, Look up Check | Makes sure that an integer has been entered, and will check the Royalties Table to find the entry to confirm that it is valid. |
| PubInvoiceID | 21 | Presence Check, Type Check, Look up Check | Makes sure that an integer has been entered, and will check the PubInvoice Table to find the entry to confirm that it is valid. |
| BookInvoiceID | 21 | Presence Check, Type Check, Look up Check | Makes sure that an integer has been entered, and will check the Book Invoice Table to find the entry to confirm that it is valid. |

2.10 Testing

2.10.1 Outline Plan

| Test Series | Purpose of Test Series | Testing Strategy | Strategy Rationale |
|-------------|--|-------------------|---|
| 1 | Testing the flow of control between user interfaces | Top-down Testing | |
| 2 | Testing the validation of input data | Bottom-up testing | All components are to be tested after development |
| 3 | Testing the algorithms' functionality | White box testing | |
| 4 | Testing that the information has been successfully stored, and in the right places | Black box testing | |
| 5 | Testing the system and whether it meets the requirements | System testing | |

2.10.2 Detailed Plan

| Test Series | Purpose of Test | Test Description | Test Data | Test Data Type (Normal/ Erroneous/ Boundary) | Expected Result | Actual Result | Evidence |
|-------------|-----------------|------------------|-----------|--|-----------------|---------------|----------|
| | | | | | | | |

| | | | | | | | |
|-----|---|---|--------------------------|--------|---|--|--|
| 1.1 | Test the Log in button on the log in screen | This should check whether the password and email match and exist in a record | Click the log in button | Normal | If the email and password match, the main menu should open, else the program should prompt the user with an error | | |
| 1.2 | Test the View button on the main menu | This button links the main menu to the view menu. | Click the View Button | Normal | The program should open the View Menu in a new window | | |
| 1.3 | Testing the Log Out button on the Main Menu | This button links to the Login screen, where the user is required to log in again | Click the Log out button | Normal | The screen should switch to the log out screen | | |

| | | | | | | | |
|-----|---|---|----------------------------------|--------|---|--|--|
| 1.4 | Testing the Search Database button on the Main Menu | This button should prompt a separate interface to open, and show details which can be used to search for specific items in the database | Click the Search Database button | Normal | The program should open a new window consisting of the Search Database screen | | |
| 1.5 | Testing the Add Entry button on the Main Menu | This button should prompt a separate interface to open and show the Add Entry screen | Click the Add Entry button | Normal | The program should open the Add Entry screen in a new window | | |
| 1.6 | Testing the Edit Entry button from the Main Menu | This button links to the Editing screen | Click the Edit Entry button | Normal | The program should switch to the Editing screen from the Main Menu | | |

| | | | | | | | |
|-----|---|--|---|--------|---|--|--|
| 1.7 | Testing the Edit Entry button after an entry has been selected beforehand | These conditions should open the Edit Entry screen upon clicking Edit Entry, with data on the selected entry already filled in on the grid | Click on an entry, and then click Edit Entry | Normal | The program should open the Edit Entry screen in a new window | | |
| 1.8 | Testing the Remove Entry Button after an entry has been selected beforehand | These conditions should prompt the user for verification on deleting a selected customer record | Click on an entry and then click Remove Entry | Normal | The program should open a Verification window, asking the user for confirmation and verification on removing the selected customer record | | |

| | | | | | | | |
|------|---|--|---|--------|--|--|--|
| 1.9 | Testing the Change Password Button on the Main Menu | This button prompts the Change Password window to open | Click the Change Password button | Normal | The program should open the Change Password window, with fields required to be filled in in order to change the password | | |
| 1.10 | Testing the Quick Search button on the Main Menu | This button returns the customer that matches the entered AuthorID | Type in an AuthorID and click QuickSearch | Normal | The program should return the customer that matches the entered AuthorID and show it in the grid | | |

| | | | | | | | |
|-----|--|--|--|---|------------------------------|--|--|
| 2.1 | Verify that some criteria has been entered when using the search | At least one set of the input boxes and drop-down lists must have been filled in or selected from, else the program will prompt the user about the error | Number based search selection and positive number input, String Based search selection and valid text input, Nothing, No selections and valid text and number inputs | Normal, Normal, Erroneous, Erroneous | Accept, Accept, Error, Error | | |
| 2.2 | Verify that a valid email has been entered on the log in screen | The program will prompt the user telling them they have inputted an error | test@testmail.com, helloworld, test @test-mail.com, test.com@testmail | No, Normal, Erroneous, Erroneous, Erroneous | Accept, Error, Error, Error | | |

| | | | | | | | |
|-----|--|--|---|---|--|--|--|
| 2.3 | Verify that a valid AuthorID has been entered on the main menu | The user will be prompted with an error | 123, 12, 1234, a23, 123, -123, @/! | Normal, Erroneous, Erroneous, Erroneous, Erroneous, Erroneous | Accept, Error, Error, Error, Error, Error | | |
| 3.1 | Verify that all fields required are entered when adding a book | If all fields are filled in correctly, a book will be successfully added to the database | Fill all Fields Correctly and then click Add to Database, Leave Fields Blank and then click Add to Database | Normal | Verification screen should open, User will be prompted with an error | | |

| | | | | | | | |
|-----|---|--|---|--------|--|--|--|
| 3.2 | Verify that all fields required are entered and calculations are complete when adding an Invoice Item | If all fields are filled in correctly and the calculations are complete, then the invoice item will be added to the database | Fill all fields correctly and click calculate and then click Add to Database, Leave Fields Blank and click Calculate, Fill all fields correctly and click add to database | Normal | Verification screen should open, User will be prompted with an error, User will be prompted to click Calculate | | |
|-----|---|--|---|--------|--|--|--|

| | | | | | | | |
|-----|--|--|---|--------|--|--|--|
| 3.3 | Verify that all fields required are entered and calculations are complete when adding a Royalty Item | If all fields are filled in correctly and the calculations are complete, then the royalty item will be added to the database | Fill all fields correctly and click calculate and then click Add to Database, Leave Fields Blank and click Calculate, Fill all fields correctly and click add to database | Normal | Verification screen should open, User will be prompted with an error, User will be prompted to click Calculate | | |
| 4.1 | Verify that all book data has been added to the book database | All the information should be added to the correct fields in the book table | Book Information | Normal | Added to the Book Table | | |
| 4.2 | Verify that all royalty item data has been added to the book database | All the information should be added to the correct fields in the royalty item table | Royalty Items Information | Normal | Added to the Royalty Items Table | | |

| | | | | | | | |
|-----|---|---|---|--------|---|--|--|
| 4.3 | Verify that all invoice items data has been added to the Invoice Items database | All the information should be added to the correct fields in the Invoice Items table | Invoice Items Information | Normal | Added to the Invoice Items Table | | |
| 4.4 | Verify that all publishing invoice data has been added to the publishing invoice database | All the information should be added to the correct fields in the Publishing Invoice table | Publishing Invoice Information | Normal | Added to the Publishing Invoice Table | | |
| 5 | Verify that the program meets the requirements given | Run the program testing all parts to make sure they meet all of the requirements | Add entries for all possible inputs in order to test them all, view all windows, change password and log out. | Normal | Program is up to the required standards | | |

Chapter 3

Testing

3.1 Test Plan

3.1.1 Original Outline Plan

| Test Series | Purpose of Test Series | Testing Strategy | Strategy Rationale |
|-------------|------------------------|------------------|--------------------|
| Example | Example | Example | Example |

3.1.2 Changes to Outline Plan

| Test Series | Purpose of Test Series | Testing Strategy | Strategy Rationale |
|-------------|------------------------|------------------|--------------------|
| Example | Example | Example | Example |

3.1.3 Original Detailed Plan

| Test Series | Purpose of Test | Test Description | Test Data | Test Data Type (Normal/ Erroneous/ Boundary) | Expected Result | Actual Result | Evidence |
|-------------|-----------------|------------------|-----------|--|-----------------|---------------|----------|
| Example | Example | Example | Example | Example | Example | Example | Example |

3.1.4 Changes to Detailed Plan

| Test Series | Purpose of Test | Test Description | Test Data | Test Data Type (Normal/ Erroneous/ Boundary) | Expected Result | Actual Result | Evidence |
|-------------|-----------------|------------------|-----------|--|-----------------|---------------|----------|
| Example | Example | Example | Example | Example | Example | Example | Example |

3.2 Test Data

3.2.1 Original Test Data

3.2.2 Changes to Test Data

3.3 Annotated Samples

3.3.1 Actual Results

3.3.2 Evidence

3.4 Evaluation

3.4.1 Approach to Testing

3.4.2 Problems Encountered

3.4.3 Strengths of Testing

3.4.4 Weaknesses of Testing

3.4.5 Reliability of Application

3.4.6 Robustness of Application

Chapter 4

System Maintenance

4.1 Environment

4.1.1 Software

4.1.2 Usage Explanation

4.1.3 Features Used

4.2 System Overview

4.2.1 System Component

4.3 Code Structure

4.3.1 Particular Code Section

4.4 Variable Listing

4.5 System Evidence

4.5.1 User Interface

4.5.2 ER Diagram

4.5.3 Database Table Views

4.5.4 Database SQL

102

4.5.5 SQL Queries

4.6 Testing

4.6.1 Screenshots of Results

4.10.1 Module 1

Chapter 5

User Manual

5.1 Introduction

5.2 Installation

5.2.1 Prerequisite Installation

Installing Python

Installing PyQt

Etc.

5.2.2 System Installation

5.2.3 Running the System

5.3 Tutorial

5.3.1 Introduction

5.3.2 Assumptions

5.3.3 Tutorial Questions

Question 1

Question 2

5.3.4 Saving

5.3.5 Limitations

5.4 Error Recovery

5.4.1 Error 1

Chapter 6

Evaluation

6.1 Customer Requirements

6.1.1 Objective Evaluation

6.2 Effectiveness

6.2.1 Objective Evaluation

6.3 Learnability

6.4 Usability

6.5 Maintainability

6.6 Suggestions for Improvement

6.7 End User Evidence

6.7.1 Questionnaires

6.7.2 Graphs

6.7.3 Written Statements