NSBM

Smart Library System

TEAM 04

Content

Team Members

Name	Student ID no.
WWSRWWMDDP Samaranayaka	21013228
TMDK Tennakoon	21009439
RV Abeykoon	21006199
AKRP Arsakulasooriya	21010128

Acknowledgement

We would like to express our heartfelt gratitude to the Lecturer, Mrs. Sulari Fernando for giving us this group task as we gained more knowledge on the module "Object Oriented Programming with C#" through it.

Introduction

The target of the assignment is to computerize an existing manual system using Windows Forms (.NET framework) by connecting it to a SQL database. It should also contain the CRUD operations.

We planned in turning a manual library management system into a computerized one. Hence, we came up with "NSBM Smart Library System". This project consists of two systems:

Operations of the systems

1. Librarian's system:
   Where the librarian can
   - Log in to the system
   - Add members, Update member details and Delete members to/from the library database.
   - Add books, Update book details and Delete books to/from the library database.
   - Check who wants to borrow a book.
   - Check whether the book is available in the library at the time of need.
   - Get the location of the book (address of the book including the section, the shelf, the rack and the slot no.) in the library.
   - Confirm lending the book.
   - Notify the system of the return of a book where the system updates the necessary tables.

Unlike normal computerized library systems, we planned in introducing a system to the user (in this case a library member) as well.

2. Member's system:

Where the member or any user can

- Search for books by book title
- Search for books by author name
- Search for books by category
- Check the book's availability

Where only a registered member can

- Borrow a book
- Request the librarian to lend the book

Benefits achieved

- Like our university, the system follows the green concept which is developed to protect our environment. It reduces the usage of paper which is used when databases are handled manually.
- The librarian can get information by just moving his/her fingers.
- The librarian need not have drawers and boxes loaded with folders piled up in his/her office thus reducing the unnecessary usage of physical space.
- The librarian can get filtered results of member details and/or book details from the database.
- The librarian need not waste time rearranging the books as a responsible person or the librarian him/herself will be taking and returning the books from and back to the shelf him/herself.
- The member need not waste time searching for a book from shelf to shelf.
- The member can check the availability of a book.

- The books will not get worn away as quickly as when the system is manually managed as the books won't be touched by the members to read their description as the blurb can be viewed/read through the system.
- This system is the best way to use a library especially in situations like the covid-19 pandemic.

Characteristics of the system

1. Librarian's system

It contains 5 forms

- Log in form
- Menu form
- Member management form
- Book management form
- Borrow and Return management form

The interfaces of the Librarian's system are set with eye friendly, dark colours and with a simple arrangement of buttons and tools as he/she must work with it for almost the whole day.

2. Member's system

It contains 7 forms

- Welcome form
- Home form
- Search form
- Log in and Borrow form
- Category form
- Book details view form
- Help form

The interfaces of the Member's system are set with user friendly, eye catching, yet soothing colours and with a creative arrangement of buttons and tools as the member should not feel the complexity of the system.

About the database

We used Microsoft Access to build the database and we connected the forms to the database with the help of the System.Data.OleDb namespace. The users, both the librarian and members (when allowed) can create, update and delete data in the database via the forms (thus achieving the aim of having CRUD operations in our project).

OOP concepts used

- Inheritance
- Encapsulation
- Exception handling
- Use of objects, methods, attributes and classes

Improvements

- The system can be improved to generate the nearest available empty slot, so that once a book is returned you don't have to always go to the relevant slot (which could be far away) and instead put it into the nearest empty slot where the system will remember it's location.

- This system, with minor improvements can also be installed in a book shop to fulfill similar tasks which comes in very handy during situations like covid-19.

Diagrams

1. Entity Relationship Diagram

2. Class Diagram

Library
-adrdess
-tel.no

1..1

1..1

1..3   1..*

Libraran
-name
-username
-password

-lend_books()
-add_data()
-delete_data()
-update_data()

>>
1..1 Manage 1..*

Books
-name
-id
-price
-description
-category

<<
1..*   Write   1..1

Authors
-name
-address
-id

1..1

Manage

1..*

1..1

Members
-name
-address
-tel.no
-birthday
-gender
-username
-password

-search_books()
-borrow_books()

1..1

Borrowed By
-borrowed_date
-due_date
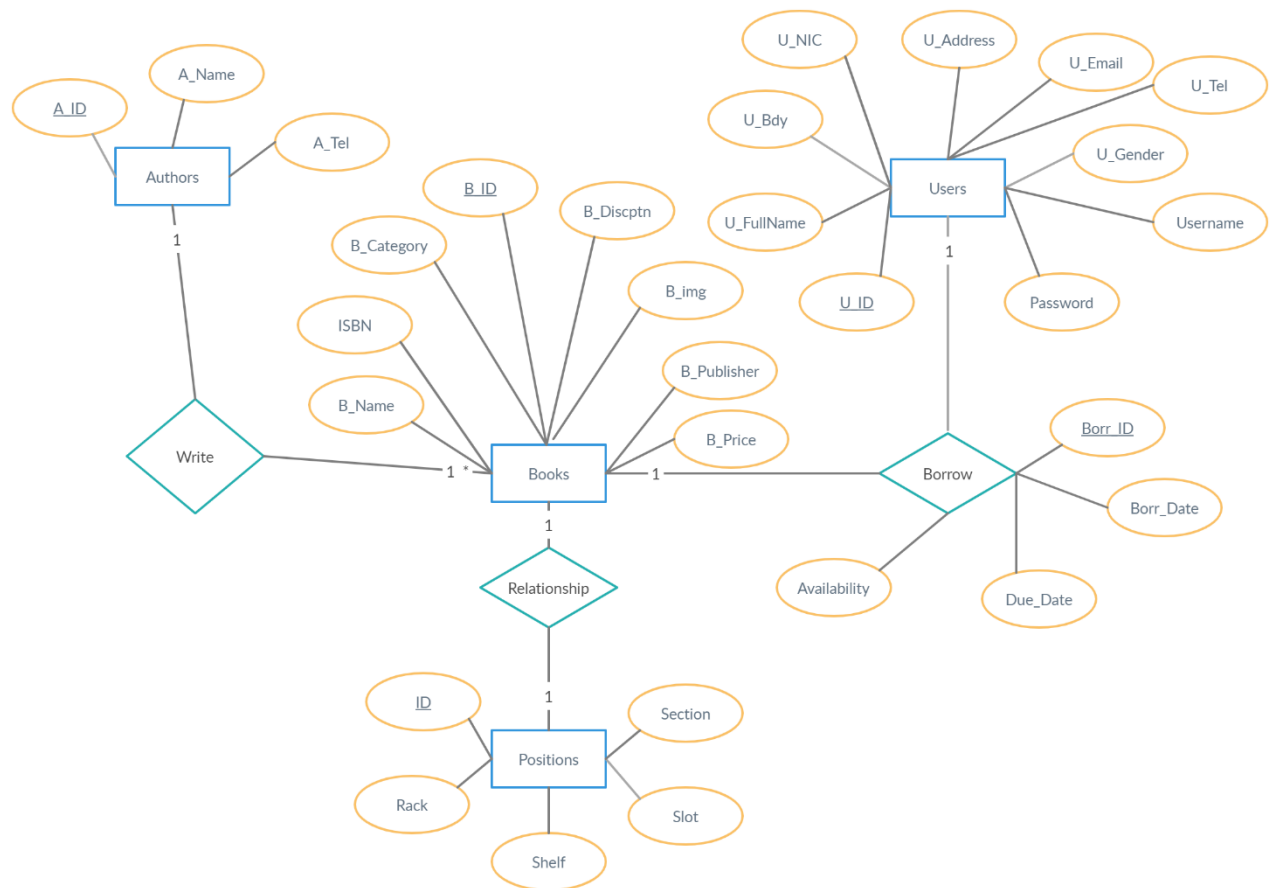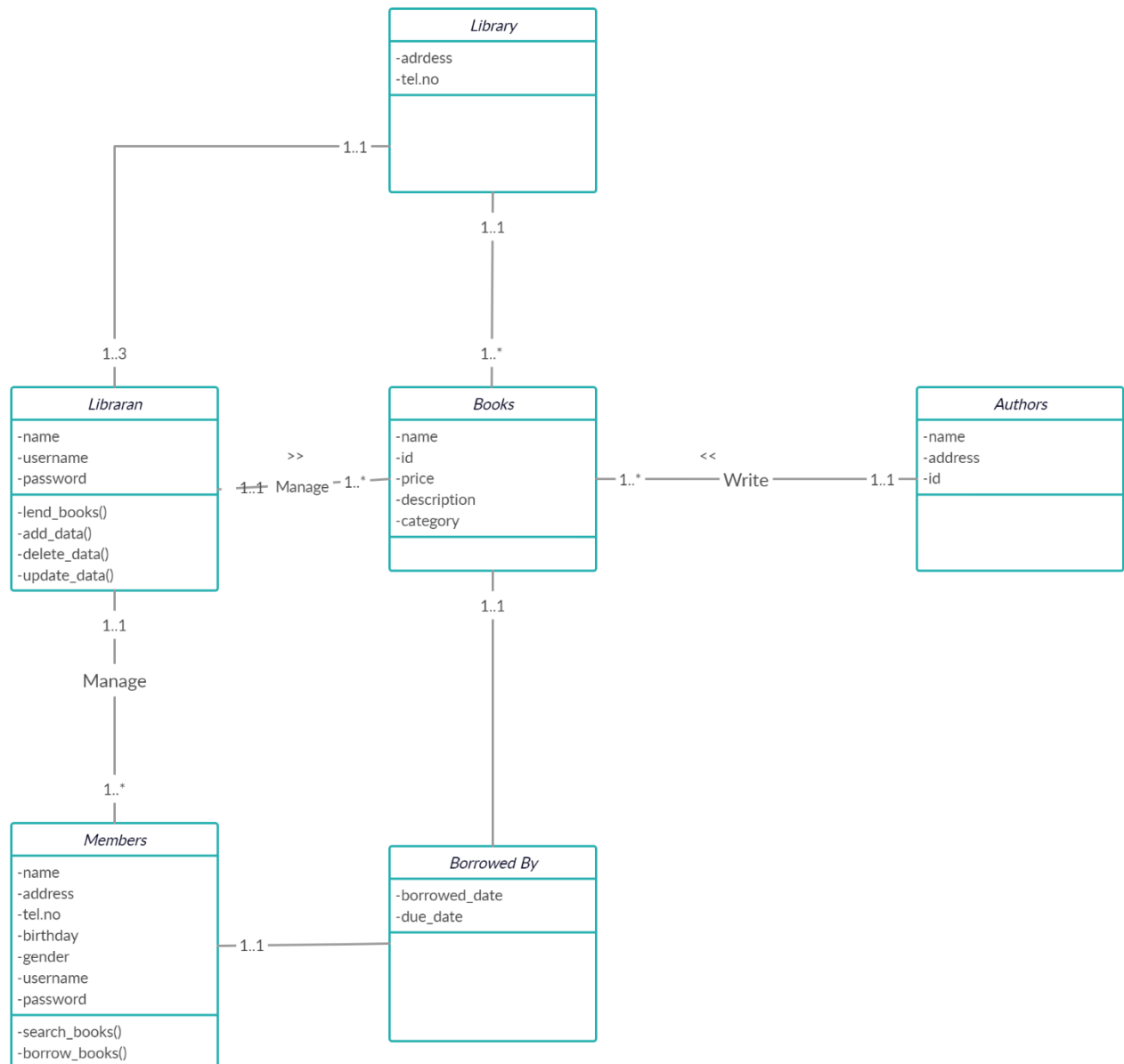
3. Use Case Diagram
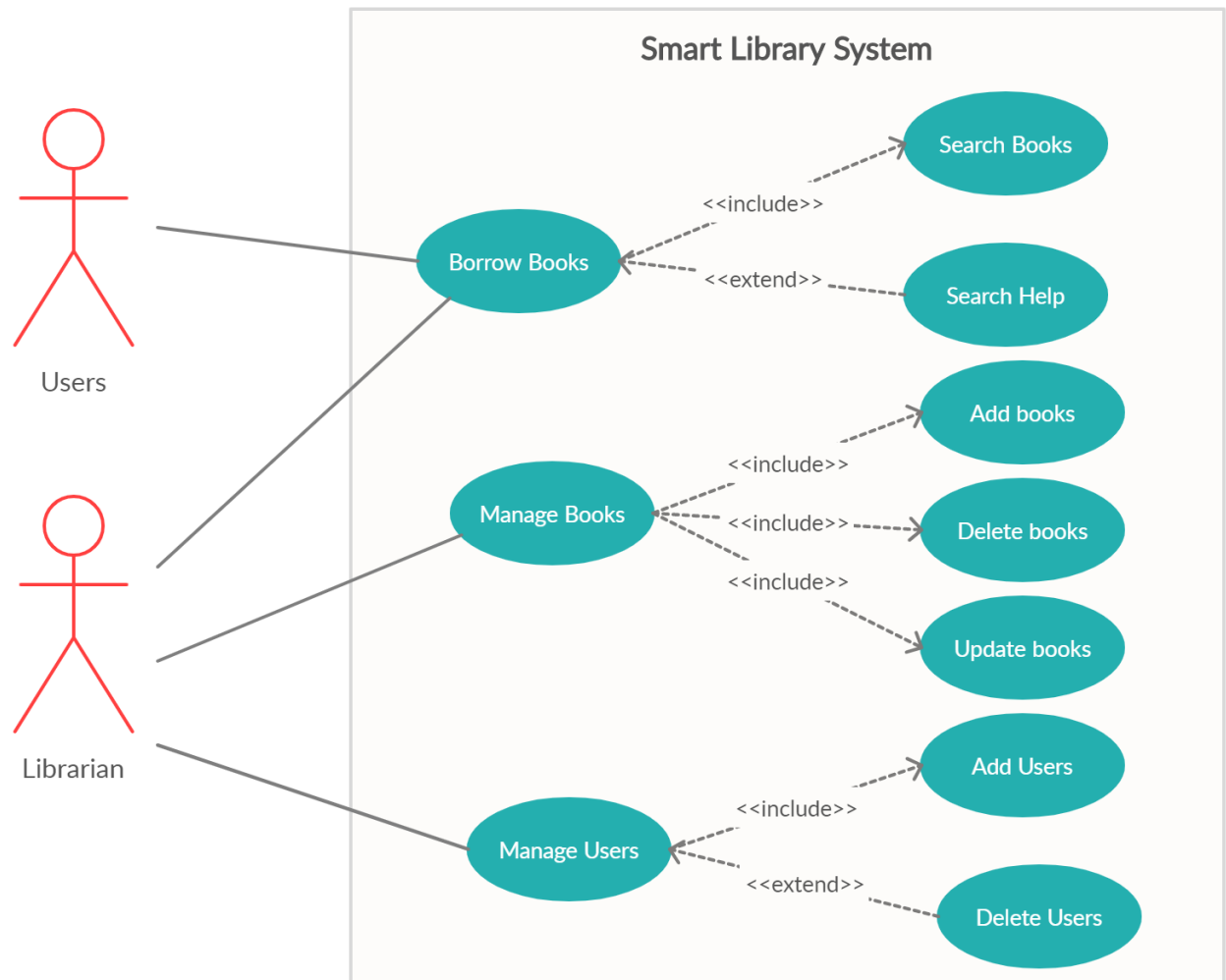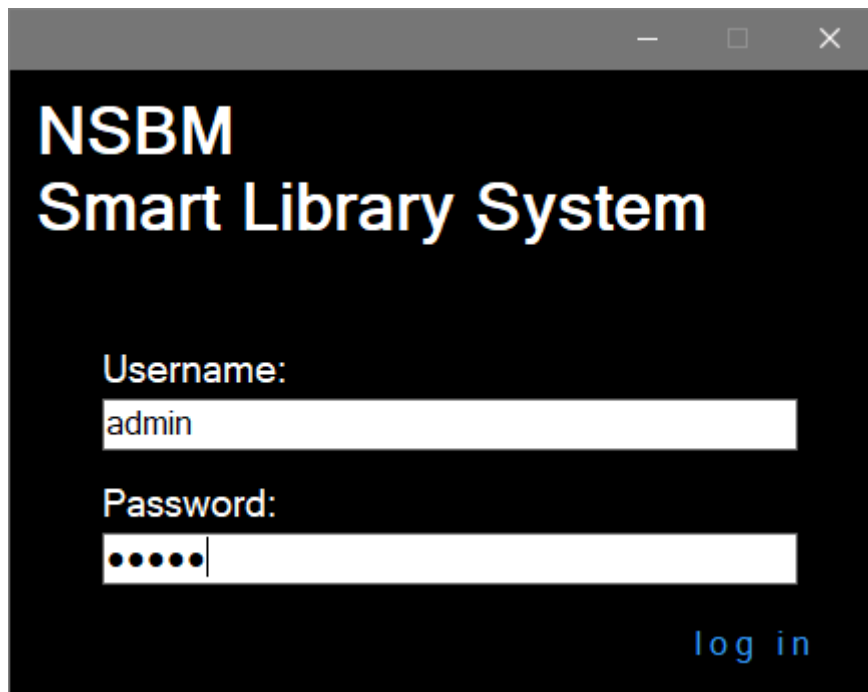
Samples of the developed system

1. Librarian's system

It contains 5 forms

- Log in form: the form validates whether the entered username and password are equal to the hard-coded values and it heads to the menu form if "true" else display a message asking to correct the mistake. (used if condition)

- Menu form
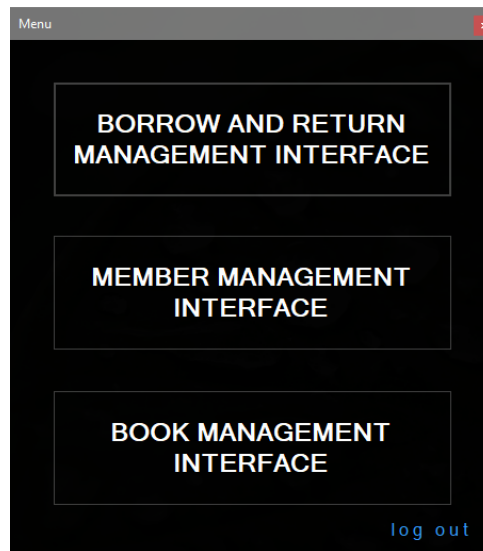
- Member management form – As mentioned in detail before this interface grants the librarian access to add and delete members and also he/she can update a required value by entering the member ID and then selecting the field name from the dropdown list containing the value to be updated and finally mentioning the new value.

- Book management form – the processes are like that of member management

- Borrow and Return management form – the top gridview of this form displays the borrow requests from the members. By entering the book ID (B_ID) we can get the position/location of the book displayed in the bottom gridview. When we enter the book ID at the bottom textbox and enter lent the "Availability" filed value of the relevant book turns from 'In Library' to 'Lent'. And finally, when a book is returned and after the entering and submitting the book id the record with that field gets deleted from the Borrowed table (top gridview). The refresh button refreshes the top gridview to display the updated/new requests.

2. Member's system

It contains 7 forms

- Welcome form

- Home form : the "Borrow a Book" button directs the user to the borrow form, the "Search a Book" button directs the user to the search form, the "Walk through our library" button directs the user to the category form whereas the "?" button directs the user to the Help form.
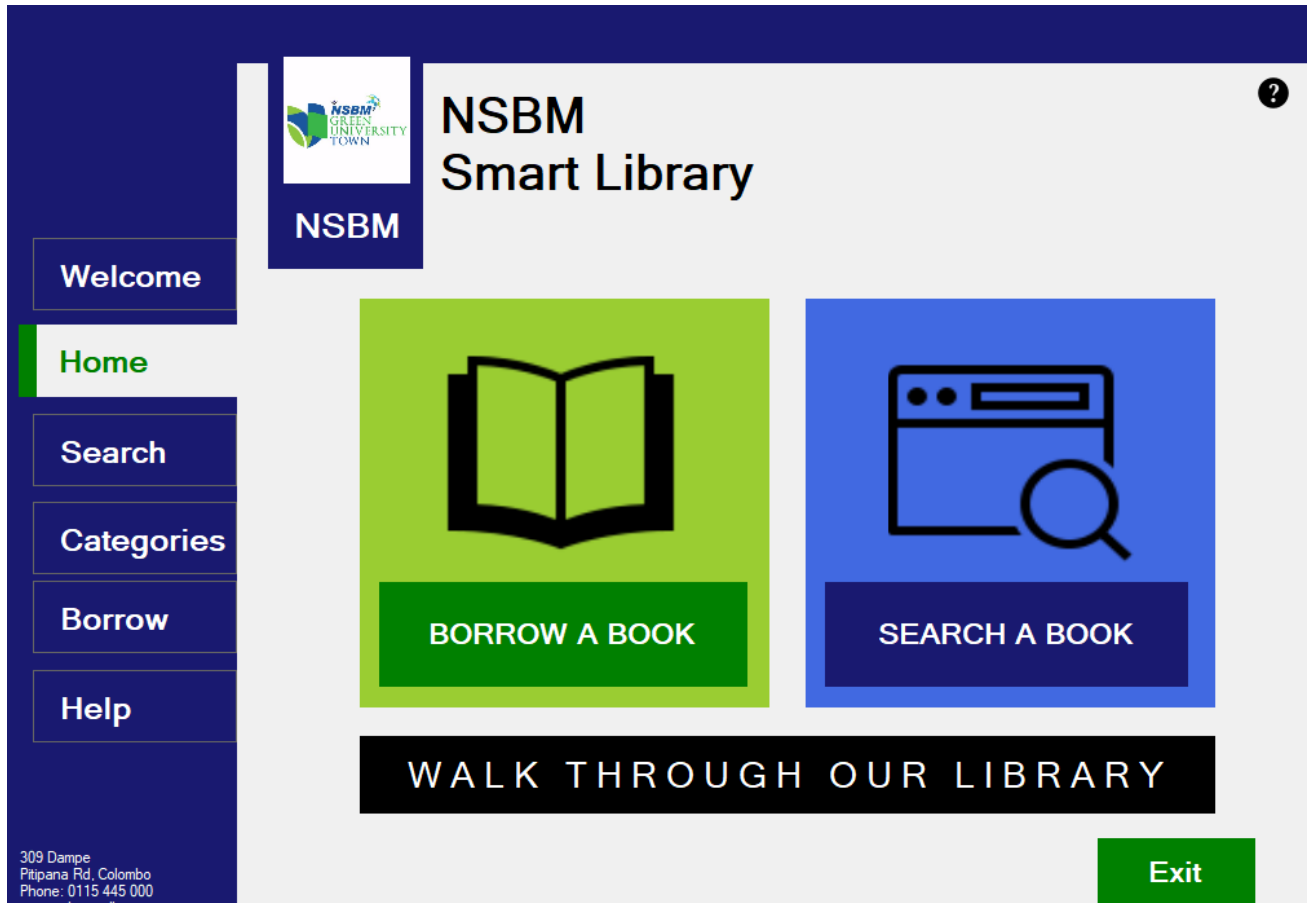
- Search form – this form allows users to browse the books by either author name or book name via the gridview below. If the user finds the book, he/she was searching for he/she can enter the relevant book ID in the relevant textbox and click borrow where he/she would be directed to the borrow page to confirm the act. If the user is unable to find a book the button below the gridview directs the user to the category form.

- Log in and Borrow form – this form sees to it that only members can borrow books by asking the member to enter his/her username and password. The form is coded so that the Book ID field will automatically contain the Book ID entered in the search form. When the required fields are filled and the borrow button is clicked the system checks the validity and if the data are true the system updates the borrow table and finally directs the user back to the "Welcome" form.

- Category form – all the books in the library fall under at least one of the below shown categories. Thus this form contain buttons named after the categories so that when you click a button (say Translations button) all the books in the database that come under this category (Translations) will be displayed in the "Book details view form" 's gridview.
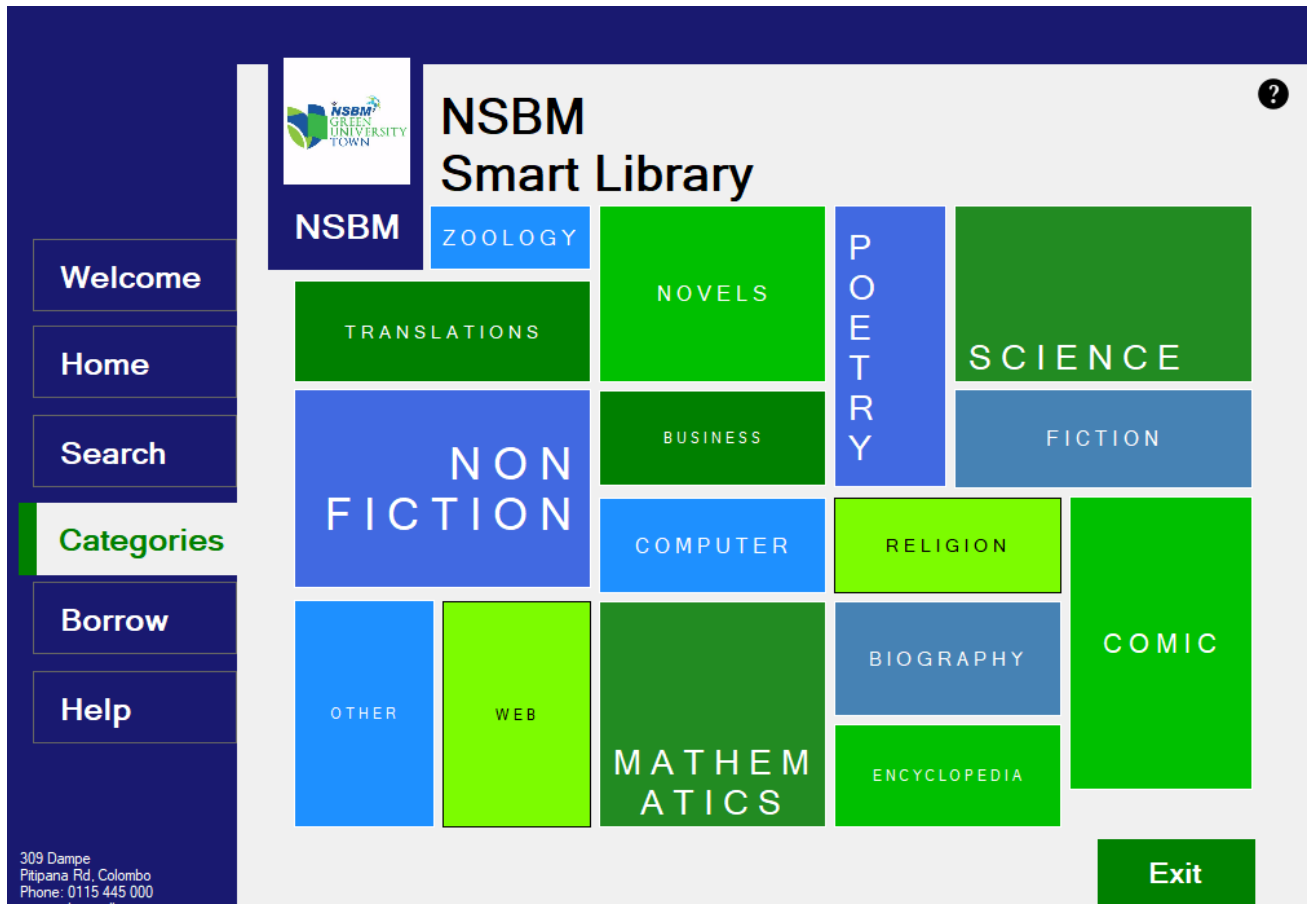
- Book details view form – this is the form mentioned above. As you can see it's a pop-up form. The top textbox is coded to contain the category name and the gridview as mentioned above shows the details of the books (including the availability of the book; whether it is "In Library" or "Lent") with the category as "Translation".

- Help form – this form contains instructions on how to use the system (note that the given image in the form is a dummy image)

Workload Matrix

Name	Student Id no.	Assigned Task	
		System	Report
WWSRWWMDDP Samaranayaka	21013228	- Database creation - Linking database with forms - Writing query	- Use case diagram - ER diagram - Class diagram - Introduction
RV Abeykoon	21006199	- Created user interfaces - Linked forms - Added data to databases	- Use case diagram - Introduction
TMDK Tennakoon	21009439	- Database creation - Linking database with forms - Writing query - Created user interfaces - Linked forms	- Introduction - Samples of the developed system
AKRP Arsakulasooriya	21010128	- Linking database with forms - Writing query - Created user interfaces	Introduction