**SYNOPSIS**

SYNOPSIS

**1.Title of the project :**

Library Management System

**2.Objective of the project :**

The main objective of the Library Management system is discipline of the planning, organizing and managing the library tasks. Our project aims at making the task of library easy. Library Management is entering the records of new book and retrieving the details of book available in the library. We can issue book to the library member and maintain their records and can also checks how many book are issued and stock available in the library. In the project we can maintain the late fine of library member who return the issued book after the due date.

**3.Project category :**

JAVA

**4.Language and software tool used :**

* Front End : Java,Hibernate,CSS.
* Operating System : Window 7
* Back End : SQLite.
* Tools: intellij IDEA ,Scene Builder, SQLite DB Browser.

**5.Structure of the project:**

**5.1 Proposed System:**

In the proposed system, we assume that each member will be having a identity card which can be used for the library book issue, fine payment etc. whenever library member wish to take a book, the book issued by the library authority will be check both the book details as well as the student details and store it in library database. In case of retrieval of book much of human intervention can be eliminated.

**5.2 Module Description:**

1. **Administrator Module**:

This is the main module in the proposed project. The administrator can read and write information about any member. The administrator can also update, create and delete the record of membership and books as per requirement and implementation plans.

The following are the sub module in the administrator module.

**User Manager:-**  Here admin can manage user related tasks like add new , update , delete , view all etc there are following sub modules:

* **Add new :** Here new user can be added.
* **Delete :** A user can be deleted from database.
* **Update :** User related information can be modified and saved again.
* **View All** : All registered users can be seen from here.

**Books Manager:-** In thissection Admin can manage book related tasks

* **Issue :-**Admin can issue a book using this section.
* **Submit :** Book can be submitted using this module.
* **Add new :** New book can be added here.
* **Delete :** A Book can be delete here.
* **View All :** Admin can see all available books .
* **View issued :** Admin can see all issued books here.

**6. Future scope of the project:**

We can consider much future scope to this application. The following are some of there.

* Online use of the library can be good feature for the Library Management system.
* Advanced fine payment system can be added.
* Inventory system can be used to maintain the books of the library.

**7. Hardware requirement :**

* Operating system: Window 7
* Hard disks: 40GB
* RAM: 256 MB

**8. Software requirement :**

* Java , Hibernate, CSS ,SQL languages
* Intellij IDEA
* SQLite DB Browser
* Java, hibernate and sql Jar files should be in the running machine.

**SOFWARE REQUIREMENT SPECIFICATION**

**1. Introduction :**

A development process consist of various phases, each phase ending with a defined output. The main reason for having a phased process is that it breaks the problem of developing software into successfully performing a set of phases, each handling a different concern of software development. This ensures that the cost of development is lower than what it would have been if the whole problem was tackled together. Furthermore, a phased process allows proper checking for quality and progress at some defined points during the development (end of process).Without this one would have to wait until the end to see what software has been produced.

Any problem solving in software consist of these steps:-

* **Requirement Analysis** :

Requirement Analysis is done in order to understand the problem the software system is to solve.

There are two major activities in this phase: problem understanding or analysis and requirement specification. In problem analysis, the aim is to understand the problem and its context, and the requirements of the new system that is to be developed. Once the problem is analyzed and the essentials understood, the requirements must be specified in the requirement specification document. The requirements document must specify all functional and performance requirements; the formats of inputs and outputs etc.

* **Software Design :**

The purpose of design phase is to plan a solution of the problem

Specified by the requirements document. This phase is the first step in moving from the problem domain to solution domain.

The design activity often results in three separate outputs: architecture design, high level design and detailed design.

* **Coding :**

The main goal of coding phase is to translate the design of the

System into code in a given programming language. The coding phase affects both testing and maintenance profoundly. The goal of coding should be to reduce the testing and maintenance effort, hence during coding the focus should be on developing programs that are easy to read and understand.

* **Testing :**

The function of testing is to detect the defects in the

Software. The main goal testing is to uncover requirement, design and coding errors in the programs.

The main goal of the requirement phase is to produce the software requirement specification (SRS), which accurately capture the client’s requirements. SRS is

A document that describes what the software should do. The basic purpose of SRS

is to bridge the communication gap between the clients, the end users and the

Software developers. Another purpose is helping user to understand their own needs.

**1.1 Purpose:**

The SRS typically contains the brief description of the project. The purpose of the requirement document is to specify all the information required to design, develop and test the software.

* The purpose of this project is to provide a friendly environment to maintain the details of books and library members.
* The main purpose of this project is to maintain easy circulation system using computers and to provide different reports.

**1.2 Scope:**

The document only covers the requirements specifications for the Library Management System. This document does not provide any references to the other component of the Library Management System. All the external interfaces and the dependencies are also identified in this document.

**Feasibility study**: The overall scope of the feasibility study was to provide sufficient information to allow a decision to be made as to whether the Library Management System project should proceed and if so, its relative priority in the context of other existing Library Management Technology.

The feasibility study phase of this project had undergone through various steps which as describe as under:

* Identity the origin the information at different level.
* Identity the expectation of user from computerized system.
* Analyze the drawback of existing system(manual system)

**1.3 Definition, Acronyms, Abbreviation:**

* JAVA -> platform independence
* SQL -> Structured query Language
* DFD -> Data Flow Diagram
* CFD -> Context Flow Diagram
* ER -> Entity Relationship
* IDE -> Integrated Development Environment
* SRS -> Software Requirement Specification

**1.4 Reference:**

* An Integrated Approach Software Engineering Third Edition by Pankaj Jalote.
* Java :- Balaguru swamy
* SQL :- JosephL Jorden

**1.5 Overview:**

The implementation of Library Management starts with entering and updating master records like book details, library information. Any further transaction like book issue, book return will automatically update the current books.

**2. Overall Description :**

**2.1 Product Perspective:**

The proposed Library Management System will take care of the current book detail at any point of time. The book issue, book return will update the current book details automatically so that user will get the update current book details.

**2.2 Product function:**

* The main purpose of this project is to reduce the manual work.
* This software is capable of managing Book Issues, Returns, and Calculating/Managing Fine. Generating various Reports for Record-Keeping according to end user requirements

**2.3 User characteristics:**

Administration module: The following are the sub module in the **User Manager:-**  Here admin can manage user related tasks like add new , update , delete , view all etc there are following sub modules:

* **Add new :** Here new user can be added.
* **Delete :** A user can be deleted from database.
* **Update :** User related information can be modified and saved again.
* **View All** : All registered users can be seen from here.

**Books Manager:-** In thissection Admin can manage book related tasks

* **Issue :-**Admin can issue a book using this section.
* **Submit :** Book can be submitted using this module.
* **Add new :** New book can be added here.
* **Delete :** A Book can be delete here.
* **View All :** Admin can see all available books .
* **View issued :** Admin can see all issued books here.
* **2.4 General Constraints:**

Any update regarding the book from the library is to be recorded to have update & correct values.

**2.5 Assumption and dependencies:**

All the data entered will be correct and up to date. This software package is developed using java as front end which is supported by sun micro system. Microsoft SQL server 2005 as the back end which is supported

by Window 7.

**3. Specific Requirement:**

**3.1 External Interface Requirement:**

The user should be simple and easy to understand and use. Also be an interactive interface .The system should prompt for the user and administrator to login to the application and for proper input criteria

**3.1.1 User Interface:**

The software provides good graphical interface for the user any administrator can operate on the system, performing the required task such as create, update, viewing the details of the book.

* Allows user to view quick reports like Book Issues/Returned etc in between particular time.
* Stock verification and search facility based on different criteria.

**3.1.2 Hardware interface:**

* Operating system : window
* Hard disk :40 GB
* RAM : 256 MB
* Processor : Pentium(R)Dual-core CPU

* + 1. **Software interface :**
* Java language
* Net beans IDE 7.0.1
* MS SQL server 2005

**3.1.4 Communication interface:**

Window

**3.2 Functional requirements:**

* + - Book entry: In this module we can store the details of the books.
    - Register student: in this module we can keep the details of the new student.
    - Book issue: This module is used to keep a track of book issue details.
    - Book return: This module enables to keep a track of return the books.

**3.3 Performance requirements:**

The capability of the computer depends on the performance of the software. The software can take any number of inputs provided the database size is larger enough. This would depend on the available memory space.

* 1. **Design constraints :**

Each member will be having a identity card which can be used for the library book issue, fine payment etc. whenever library member wish to take a book, the book issued by the library authority will be check both the book details as well as the student details and store it in library database. In case of retrieval of book much of human intervention can be eliminated.

* 1. **System attributes :**

* **Maintainability**: There will be no maintained requirement for the software. The database is provided by the end user and therefore is maintained by this user.
* **Portability**: The system is developed for secured purpose, so it is can’t be portable.
* **Availability**: This system will available only until the system on which it is install, is running.
* **Scalability:** Applicable.

**3.6 Other requirements:**

There are no other requirements.

**DATABASE DESIGN**

**1. Introduction:**

The database is a collection of information and is systematically stored in tables in the form of rows and columns. The table in the database has unique name that identifies its contents. The database in turn is further described in detail giving all the fields used with the data types, constraints available, primary key and foreign key.

Database design is used to manage large of information. In this database we describe the entire 4 table available in the software, which are used to store all the records.

**2. Data types and its description:**

Fields in database table have a data type used in database table are explained below.

1. **Integer**: one optional sign character (+ or -) followed by at least one digit (0-9). Leading and trailing blanks are ignored. No other character is allowed.
2. **Varchar**: It is used to store alpha numeric characters. In this data type we can set the maximum number of characters up to 8000 ranges by defaults SQL server will set the size to 50 characters range.
3. **Data/time**: Data/time data type is used for representing date or time.

**Database Name: bookData**

|  |  |  |
| --- | --- | --- |
| **Field Name** | **Data Type** | **Relation** |
| id | integer | Primary key |
| isbn | Text | Not null |
| subject | Text | Not null |
| name | Text | Not null |
| author | Text | Not null |
| publisher | Text | Not null |
| edition | integer | Not null |
| pages | Integer | Not null |
| addDate | Text | Not null |
| shelf | Integer | Not null |
| language | Text | Not null |
| description | Text | Not null |
| photo | Blob | Not null |

**Database Name: userData**

|  |  |  |
| --- | --- | --- |
| **Field Name** | **Data Type** | **Relation** |
| userId | Text | Primary key, Foreign key |
| fristName | Text | Not null |
| lastName | Text | Not null |
| DOB | Text | Not null |
| mobile | text | Not null |
| email | Text | Not null |
| password | Text | Not null |
| photo | Blob | Not null |

**Database Name: issueBookData**

|  |  |  |
| --- | --- | --- |
| **Field Name** | **Data Type** | **Relation** |
| id | Integer | Primary key |
| userId | Text | Foreign key |
| bookId | Integer | Foreign key |
| issuedDate | Text | Not Null |

**Database Name: adminData**

|  |  |  |
| --- | --- | --- |
| **Field Name** | **Data Type** | **Relation** |
| id | Integer | Primary key |
| password | text | Not Null |
| email | text | Not null |
| name | Text | Not null |
| mobile | text | Not null |

**Entity Relationship Diagram:**

Entity Relationship Diagram is used in modern database software engineering to illustrate logical structure of database. It is a relational schema database modeling method used to

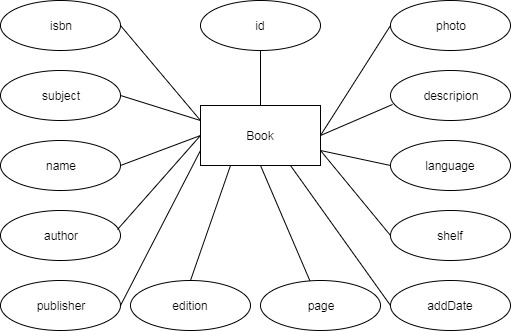
Model a system and approach. This approach commonly used in database design. The diagram created using this method is called ER-diagram.

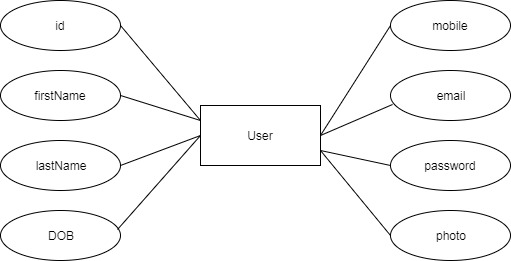
The ER-diagram depicts the various relationships among entities, considering each object as entity. Entity is represented as rectangle shape and relationship represented as diamond shape. It depicts the relationship between data object. The ER-diagram is the notation that is used to conduct the data modeling activity.

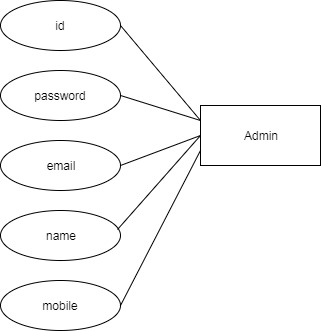
**Entity:** Entity is the things which we want to store information. It is an elementary basic building block of storing information about business process. An entity represents an object defined within the information system about which you want to store information. Entities are distinct things in the enterprise.

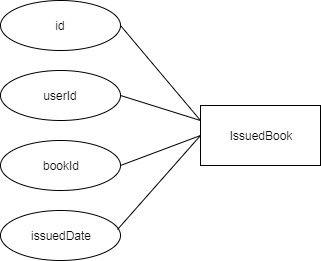
**Relationship:** A relationship is normal connection or association between entities used to relate two or more entities with some common attributes or meaningful interaction between the object.

**Attributes:** Attributes are the properties of the entities and relationship descriptor of the entity. Attributes are elementary pieces of information attached to an entity.

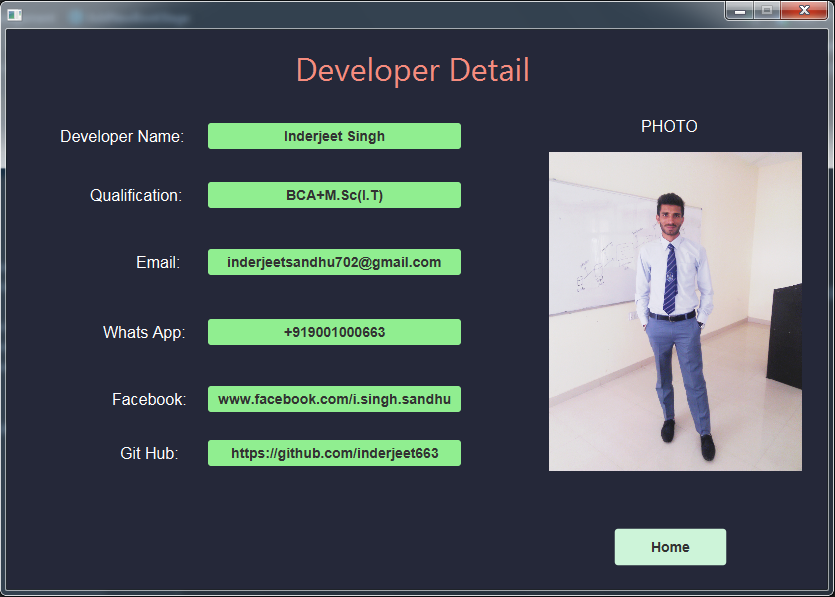
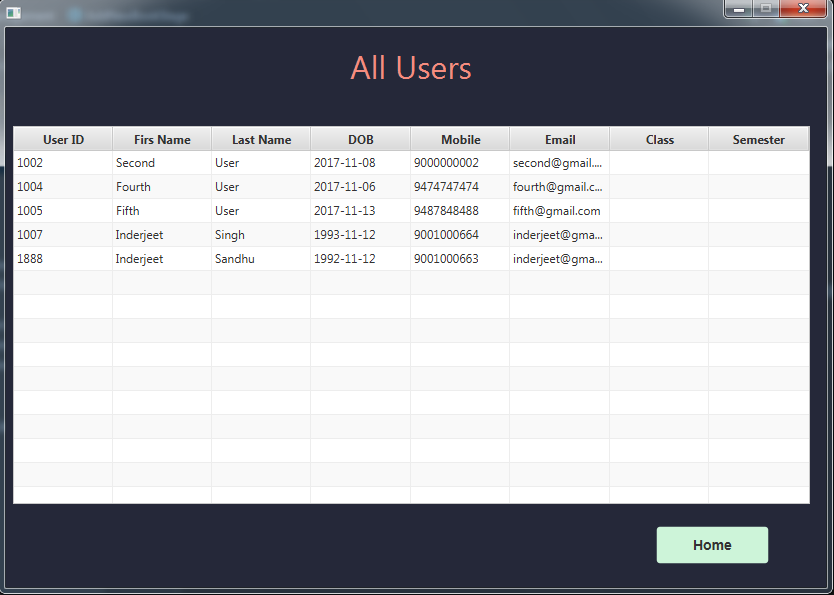
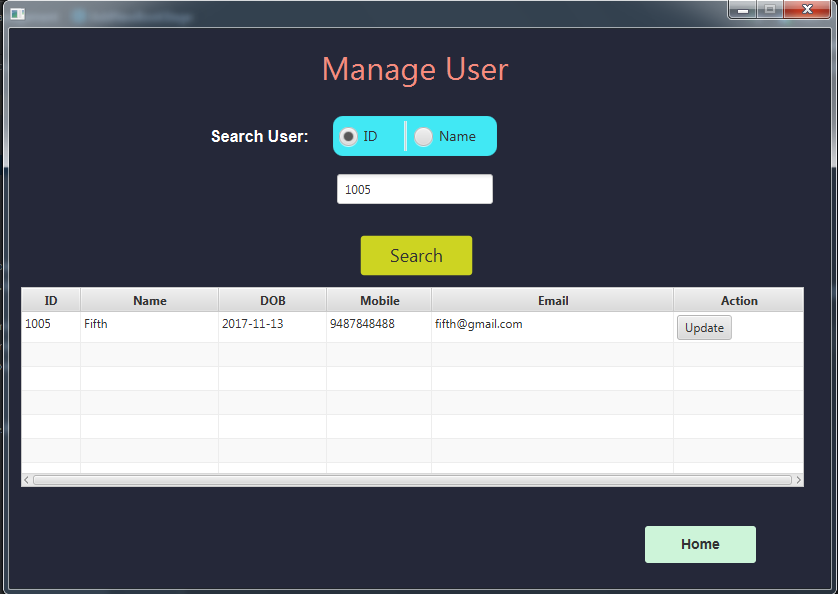
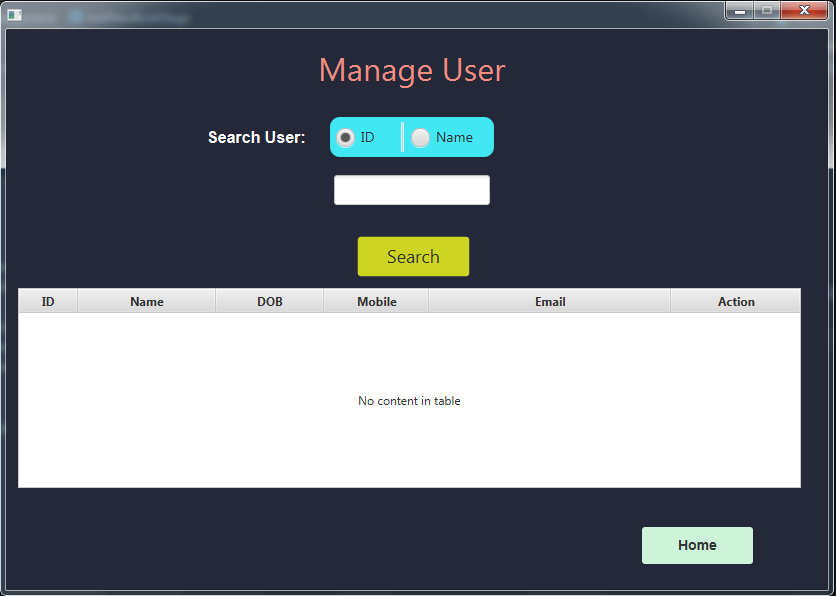
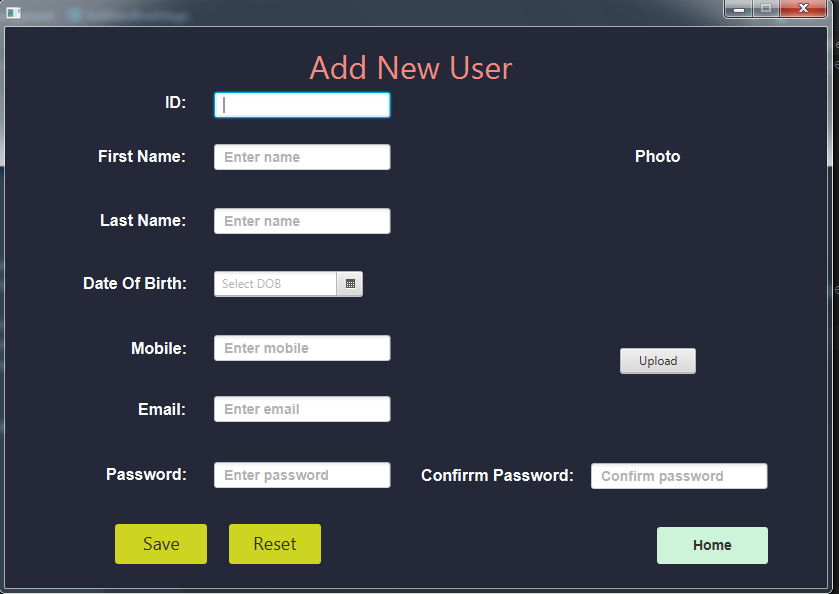
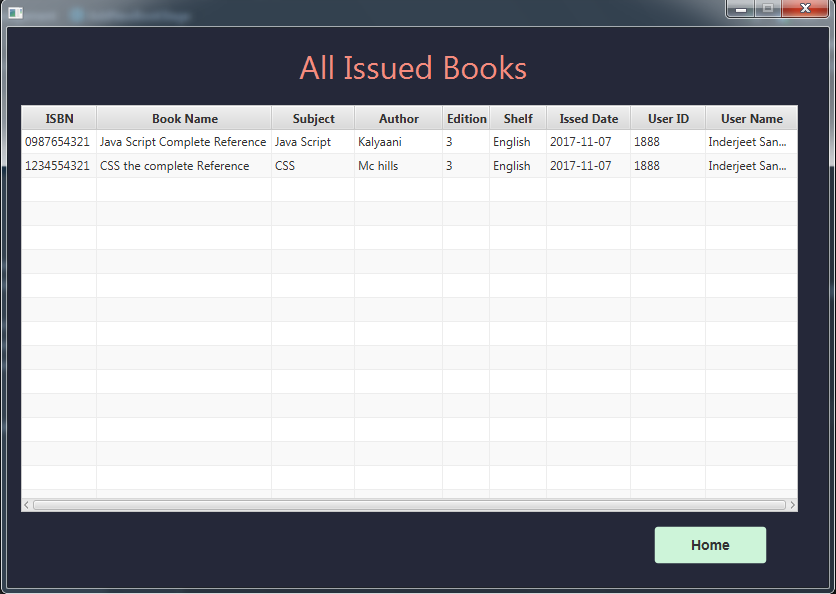
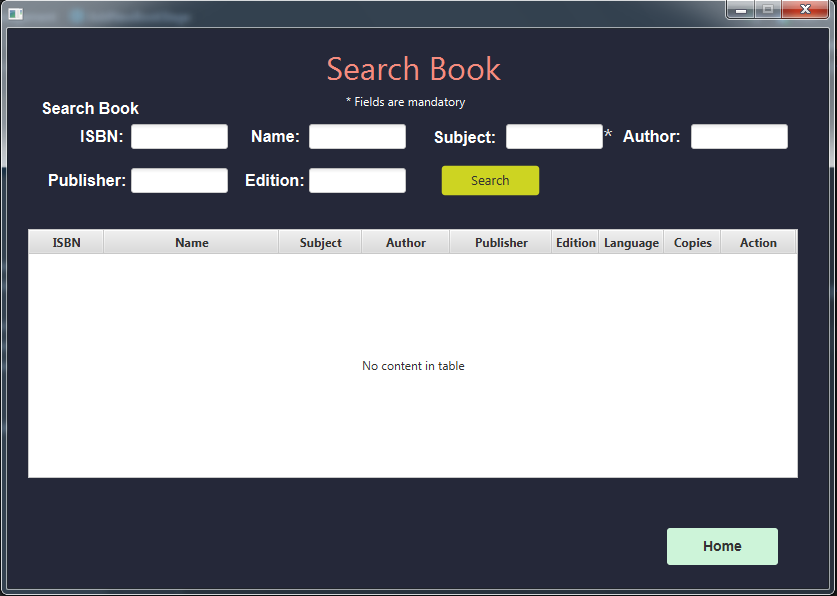
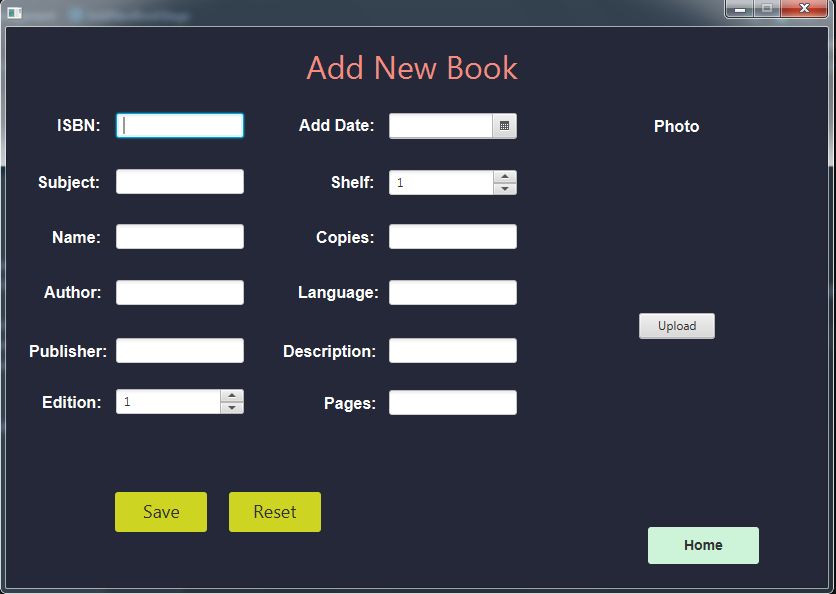
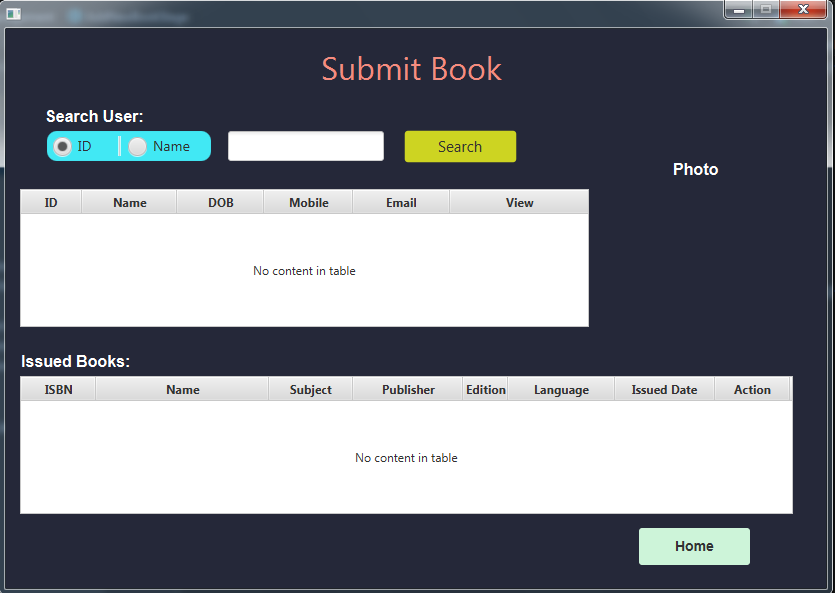
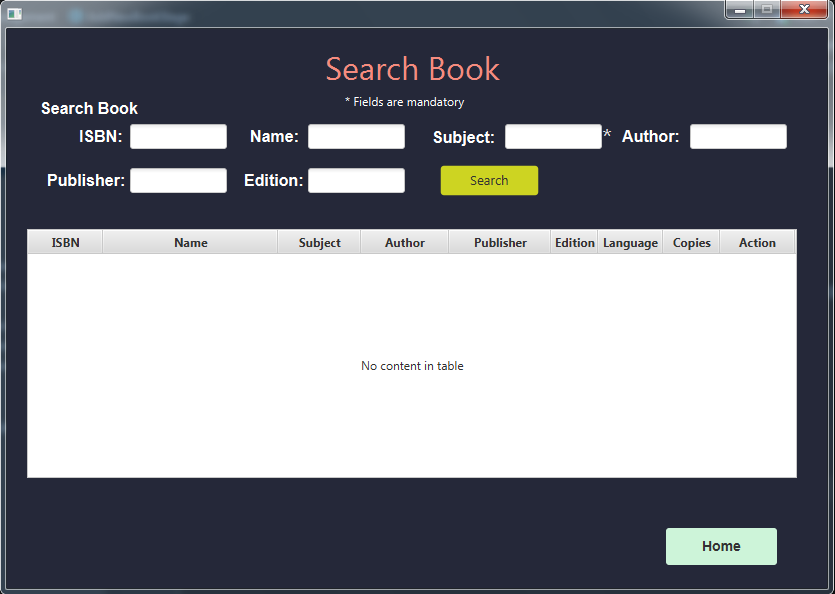
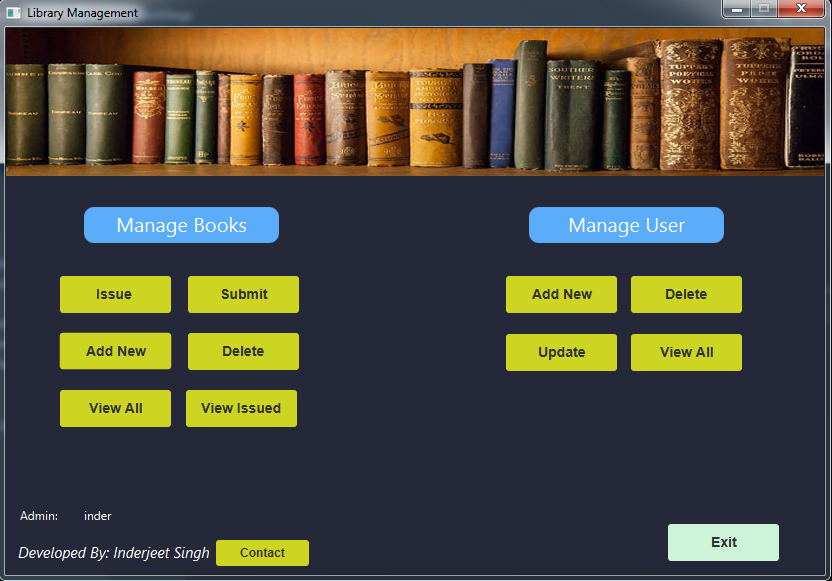
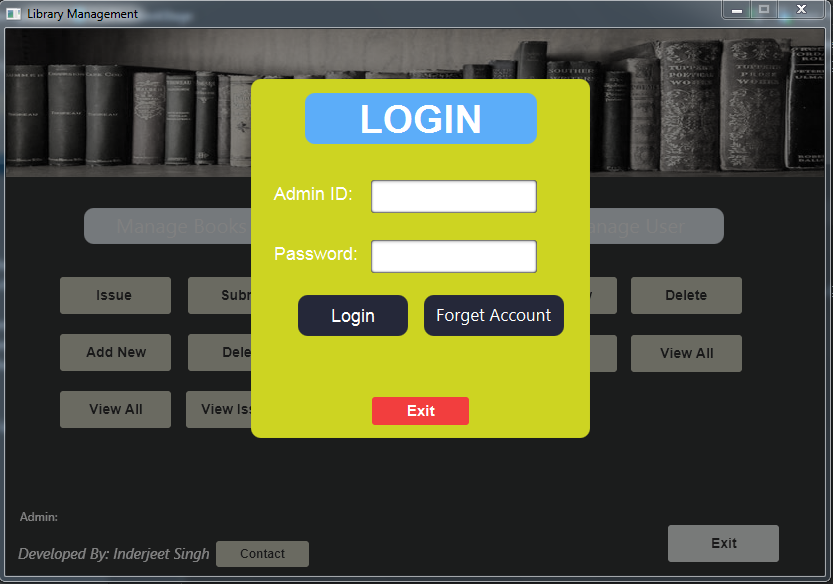


****

****

****

**Snap Shots**

****