

**GROUP MEMBERS**

**INDEX NUMBER FULL NAME**

10541912 Bandara K G Prageeth K K

10541922 Algawatta Shamika Dilshan

10541954 Nayanajith W K Chathura

10541973 Ratnayake K Basura N B (**Team Lead**)

10541994 Bavantha J Lokuhettige

**Degree:** BSc (Honours) Software Engineering

**Stage:** 2

**Module:** PROJ202SL

**Batch:** 14.2

**THE**

**FORE RUNNERS**

Believe everything is possible

CLMS 1.0 – College Library Management System

Project Report

Table of Contents

[Acknowledgement 1](#_Toc445053358)

[Declaration 2](#_Toc445053359)

[Executive Summary 3](#_Toc445053360)

[UML Diagrams 5](#_Toc445053361)

[Activity Diagrams 9](#_Toc445053362)

[Entity Relationship Diagram 14](#_Toc445053363)

[Normalized Database 16](#_Toc445053364)

[Implementation Plan **Error! Bookmark not defined.**](#_Toc445053365)

[Member Responsibilities and Tentative **Error! Bookmark not defined.**](#_Toc445053366)

[Evaluation **Error! Bookmark not defined.**](#_Toc445053367)

[Functionality **Error! Bookmark not defined.**](#_Toc445053368)

[Database and File Design **Error! Bookmark not defined.**](#_Toc445053369)

[Core Functionality **Error! Bookmark not defined.**](#_Toc445053370)

[GUI Designs **Error! Bookmark not defined.**](#_Toc445053371)

[Implementation of Features 27](#_Toc445053372)

[Personal Reflection 30](#_Toc445053373)

# Acknowledgement

We would like to thank Mr. Viraj Edirisinghe for giving us the freedom to choose a scenario of our liking and to build the project based on our own assumptions and ideas and also for his heartfelt support and guidance throughout the successful completion of our project.

**Team ForeRunners**

18th April, 2016

# Declaration

This is to certify that the work contained herein are the sole and original creations of **Ratnayake K Basura N B**, **Algawatta Shamika Dilshan**, **Bandara K G Prageeth K K**, **Bavantha J Lokuhettige**, and **Nayanajith W K Chathura** and that no other external or referential creation/ source by a third-party, other than those cited, were taken-up/ included in the making of this Project. All ownership of this project resides with **TheForeRunners** and all the copyrights of the libraries implemented in the project belong to their respective owners.

# 

# Executive Summary

The scenario given to this project is a College Library Management System specifically for college and university students. This system allow users to manage resources in the library. The members and library staff both can use the system but with different authorities and powers.

The thing that separates this Library Management System from other systems is that the system is highly automated and requires only little human interactions

**E.g. Electronic card, Resource Scanner**

Since this is a group project, we have decided to broaden the scope of the taken scenario by some amount, such that it’ll allow each member to explore as many server-side techniques as possible, within the provided constraints. The functions of the application will be described in the proceeding sections of this document.

By trying to implement these various functions, we believe that we’ll be able to get the most out of working as a team; broadening our knowledge individually and striving together to meet the common goal of delivering a perfectly working distributed application.

# Analysis and Design

## Problem Identification

The major problem we found was the library staff has to manipulate so much of things manually, which sometimes return in mistakes, system errors and sometimes frauds.

The Main problems are classified as below,

* **Inefficiency and Ineffectiveness**

Library staff has to undergo too much library resource issuing, accept returns and register members and etc. This way the library staff can be reduced and also the efficiency and effectiveness doubled.

* **Fraud Memberships**

The library cards were easily printable by members and fraud occurred.

* **System Crash**

The entire library has a normal centralized database, which crashes due to heavy number of users accessing the system.

Considering the above identified problems and situations, we formulated an idea that would satisfy both the above mentioned problems and the project criteria that we are to accomplish.

## The Proposed Solution

We believe all problems can be remedied through technology, the answer to this problem is all computer science. By building a distributed application that can function as a library management system but without the above realized weaknesses, we can finally obtain a system that any and all can use without any limitations.

**Expected benefits from the new system**

* **Effective and Efficiency**

The librarians scan resources and library identification cards to easily issue resources to members and also to accept returns. So the librarian has to provide only a very limited amount of user inputs.

* **Legitimate Memberships**

Unique library identification card and resource identification is generated to uniquely identify members and resources. So frauds can be dealt easily and precisely.

* **System Stability**

The system is divided into parts and the database is divided into two to avoid system crash.

## 

# UML Diagrams

## Class Diagrams





## 

## Use Case

## 

## Activity Diagrams

# 

# 







# Database

## Entity Relationship Diagram

**Database – Resource Member**



**Database – Transactions**

****

## 

## Normalized Database

**Database Name**: ResourceMember

**Table:** regCodes

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| nic | sid | regCode | regDate | activated | pic |

**Table:** loginDetails

|  |  |  |  |
| --- | --- | --- | --- |
| regCode | username | password | email |

**Table:** personalDetails

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| username | firstName | lastName | street | city |

**Table:** contactDetails

|  |  |  |
| --- | --- | --- |
| username | mobile | home |

**Table:** contactDetails

|  |  |  |
| --- | --- | --- |
| username | mobile | home |

**Table:** books

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| isbn | title | fname | lname | genre | published | pic | publisher | lang | bType |

**Table:** newspapersMags

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| nmId | title | genre | published | lang | pic | publisher | bType |

**Table:** comics

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| coId | title | genre | published | lang | pic | publisher | volume |

**Table:** vidDoc

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| vdId | title | genre | lang | fname | lname | duration | pic | vdType |

**Table:** pastPapers

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ppID | title | Programme | eyear | semester | fname | lname | eDate | pic |

**Table:** resourceCodes

|  |  |  |
| --- | --- | --- |
| rID | rCode | rType |

**Database Name**: LibraryTransactions

**Table:** issueBook

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| ibNo | memberId | bookId | bType | issueDate | returnDate | returned |

**Table:** issueComic

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| icNo | memberId | comicId | issueDate | returnDate | returned |

**Table:** issuePastP

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ipNo | memberId | pastPId | issueDate | returnDate | returned |

**Table:** issueVidD

# 

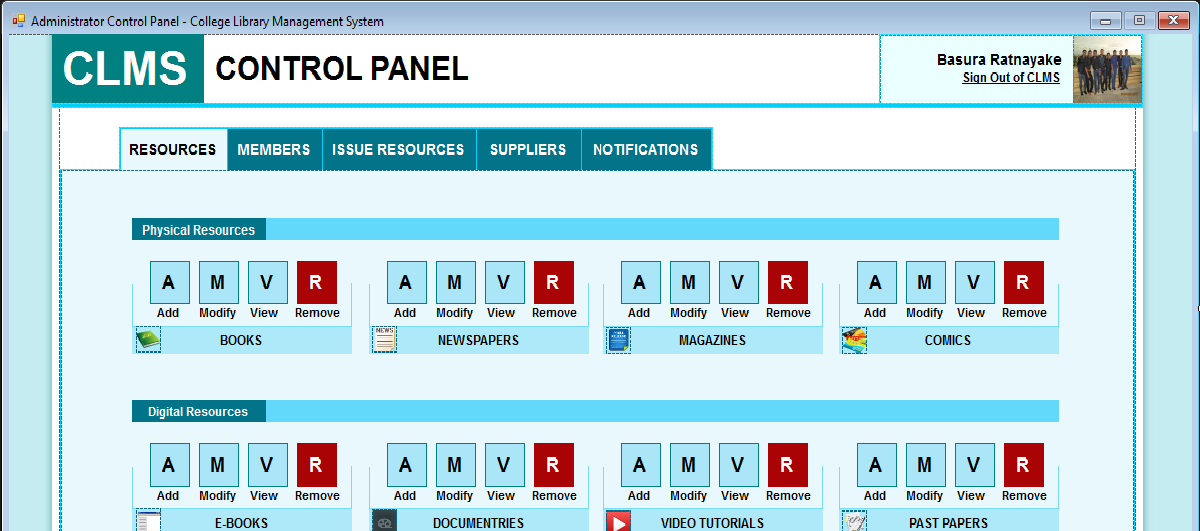
# System Description

The scenario for the project is a library management system where individuals can search and take home resources from the library by the librarian permission. The Graphical User Interface of the system is implemented by using MS Visual Studio 2015 and the back-end data storage of the system is performed by MS SQL Server database and file handling, the reason for using the MS SQL Server is because the system is in need of a portable data storage platform. The coding and implementation phase of the system is done using the MS Visual Studio 2015.

**Control Panel**

The control panel consists of 5 main tabs as following

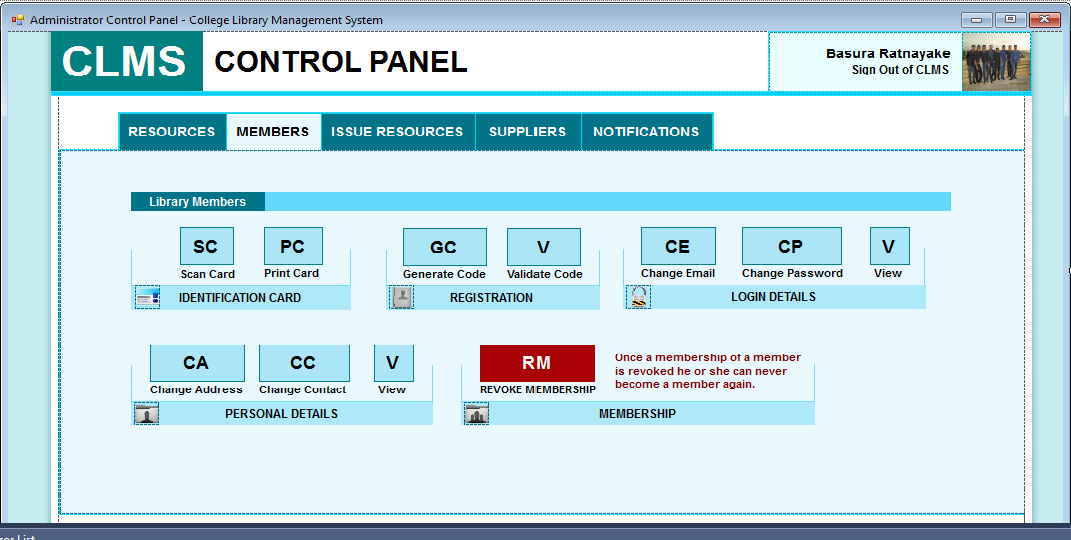
Resources, Members, Issue Resources, Suppliers, Notifications

Resources

Resources tab consists of 8 categories and each category consist with 4 different features (options) as following

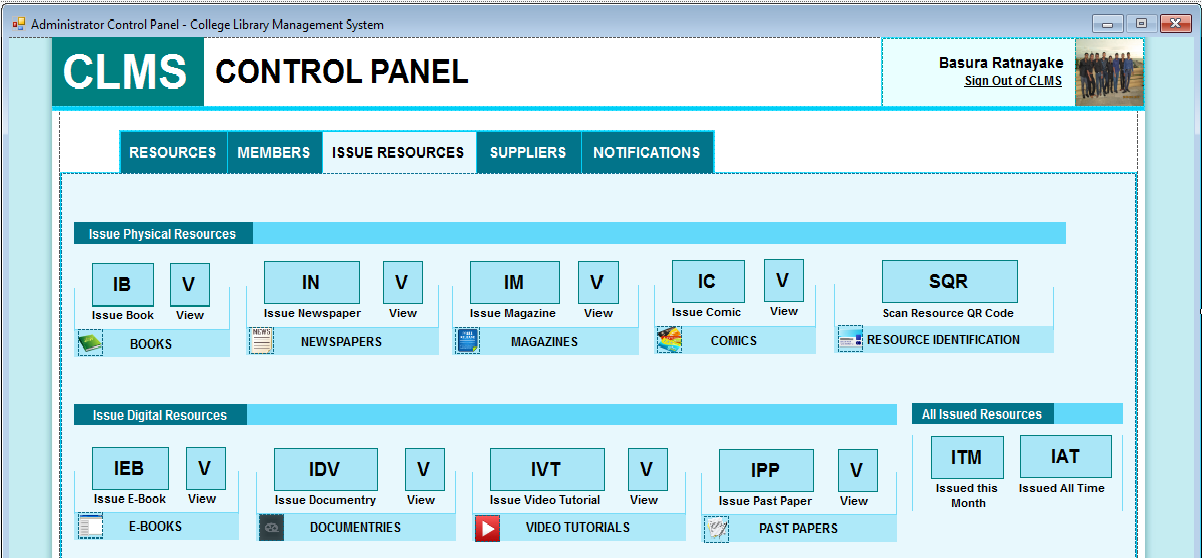
* Add
* Add resources to the database server with fields specified.
* Modify
* Modify resources to the database server with fields specified.
* View
* View resources to the database server with fields specified.

* Remove
* View resources to the database server with fields specified.

**Members**

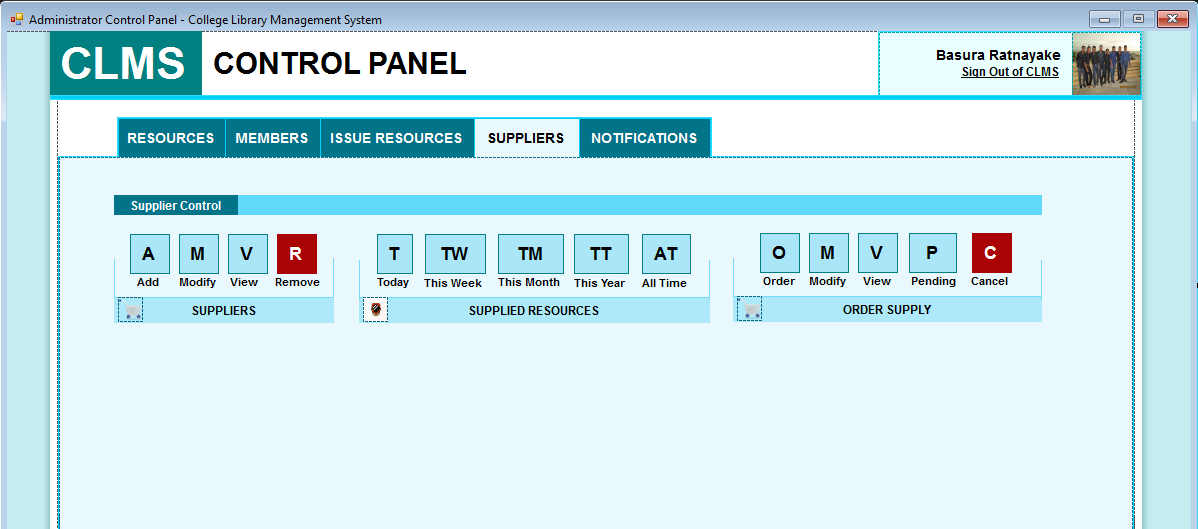
Members tab gives the information of the library members. Library members consists of 5 different categories as following

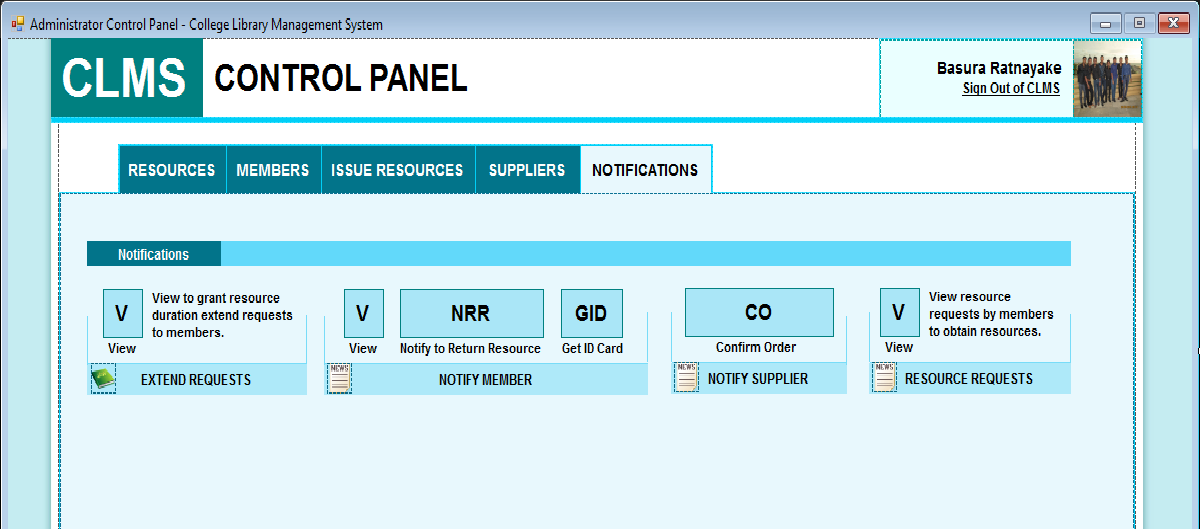
* Identification Card
* Registration
* Login Details
* Personal Details
* Membership

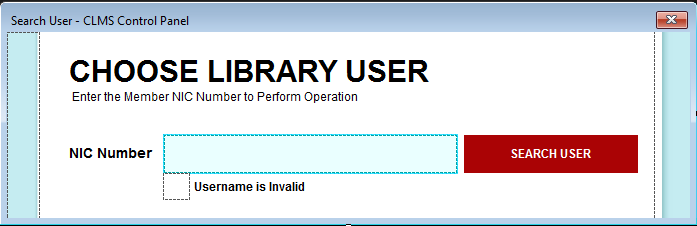
**Issue Resources**

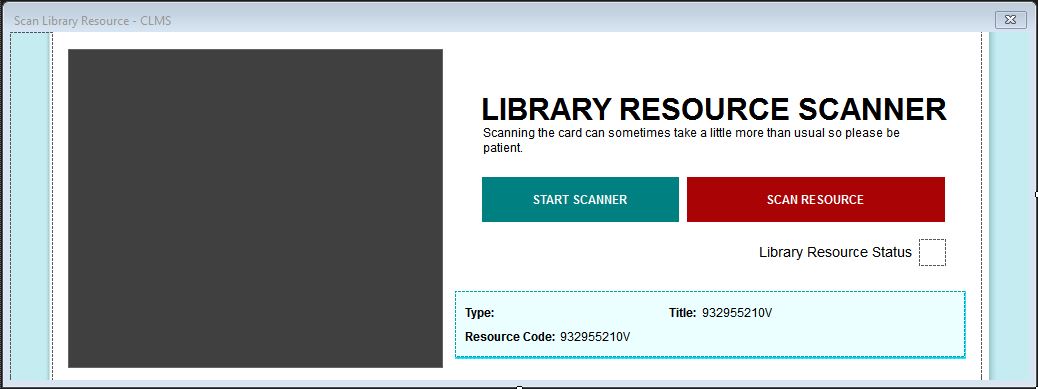
Issue Resources is categorized with Issue Physical Resources, Issue Digital Resources and All Issued Resources.

* Issue Physical Resources
* Physical resources include Books, Newspapers, Magazines, Comics, Resource Identification.
* Issue Digital Resources
* Digital Resources include with E-Books, Documentaries, Video tutorials and Past papers.
* All Issued Resources
* Here we have two features to check the resources issued this month and Issued all time.

**Suppliers**

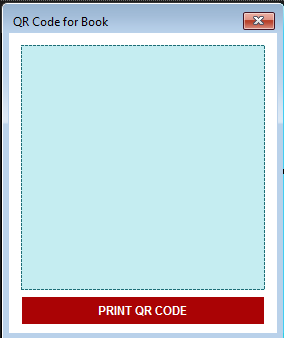
**Notifications**

**Search User**

**Scan Library Resource**

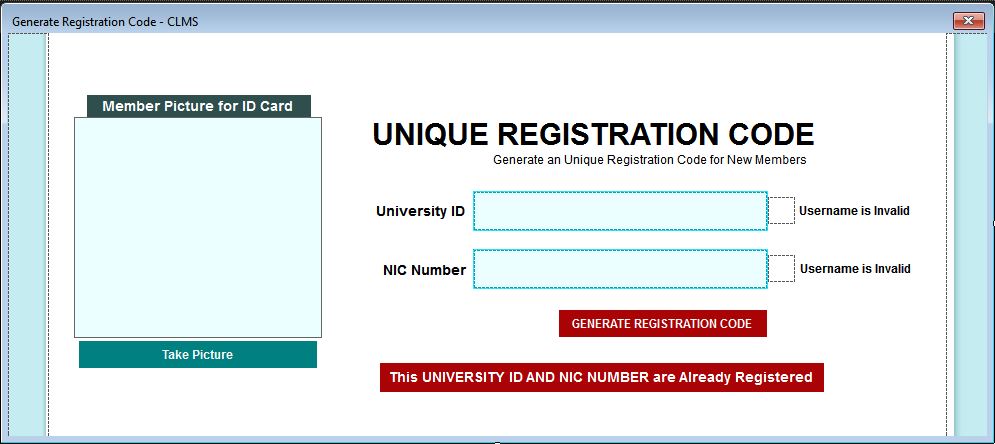
By using this interface the librarian can scan the library card of the members. Here there are 2 buttons as Start Scanner and Scan Resource.

**QR Code for Book**

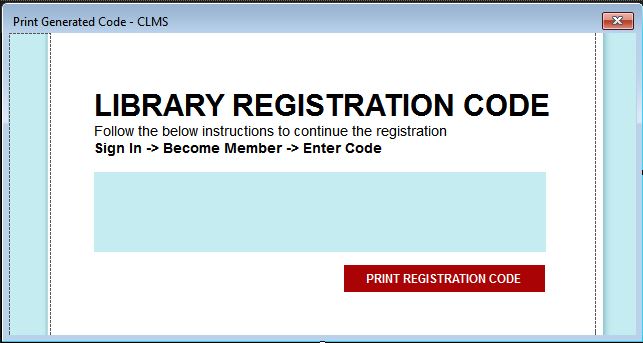


This interface helps to print a QR code for books that has been taken by the library members. A button name as PRINT QR CODE is provided in order to print the QR CODE.

**Unique Code Generation for New Member**



**Unique Code View**

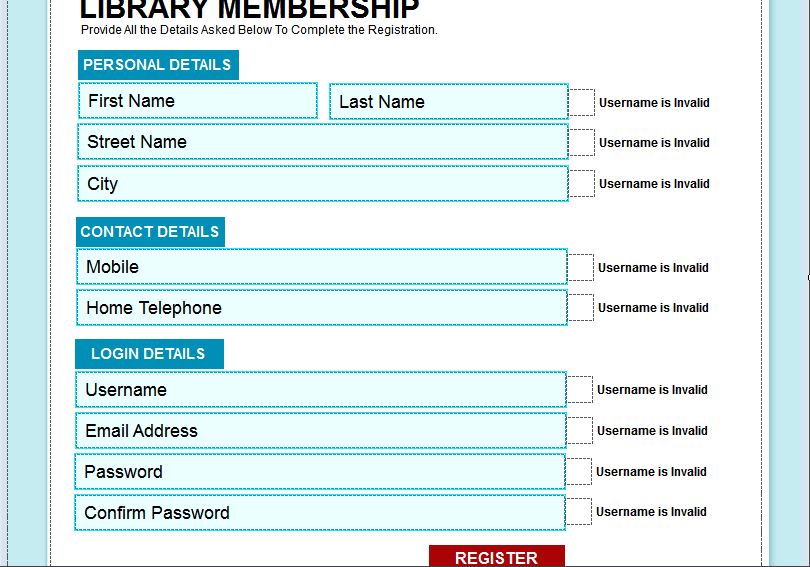


**Validate Registration Code**

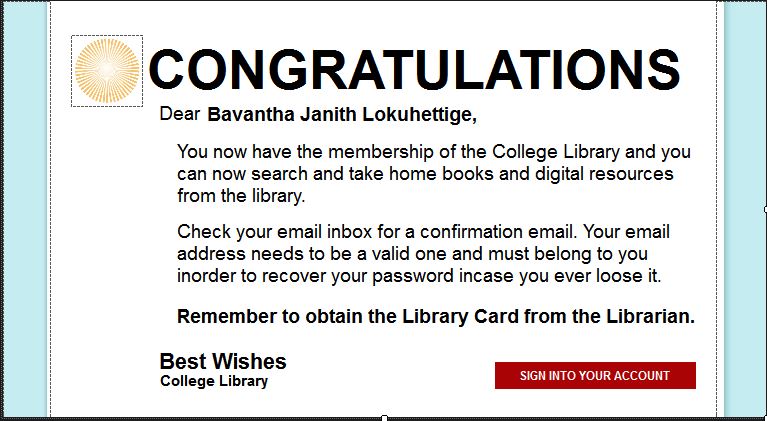


Validate the newly generated code of the new member.

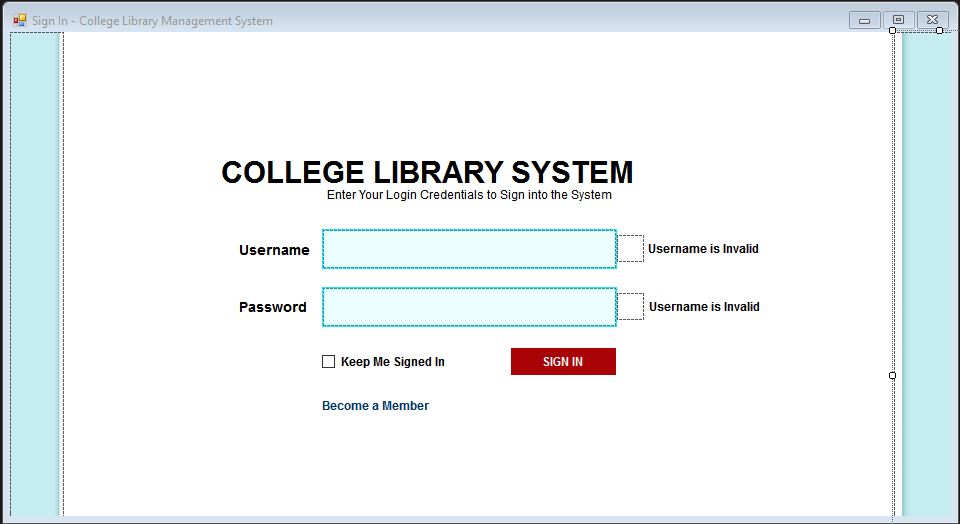
**Register Member**



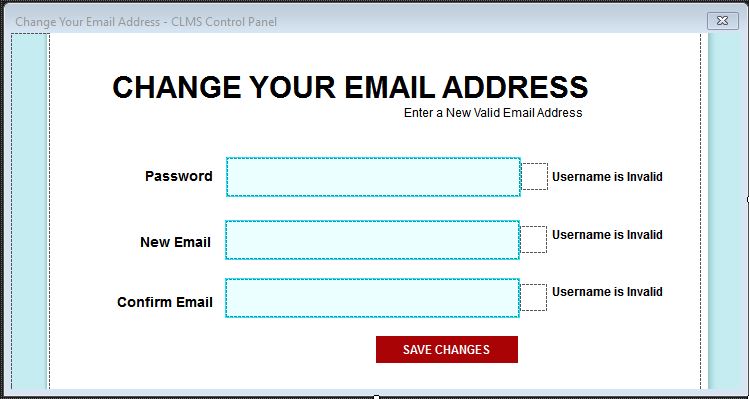
**Registration Successful**

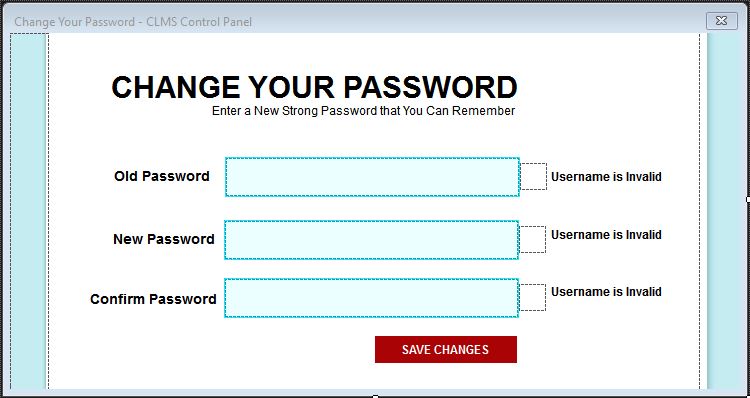


**Sign In Page**

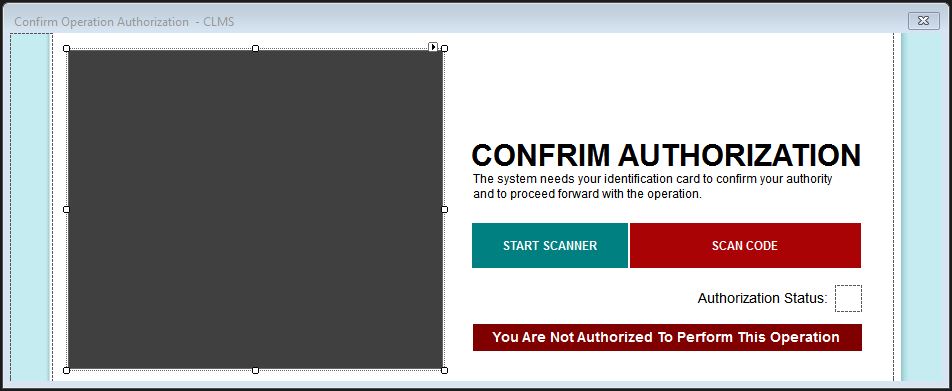


**Change User Details**





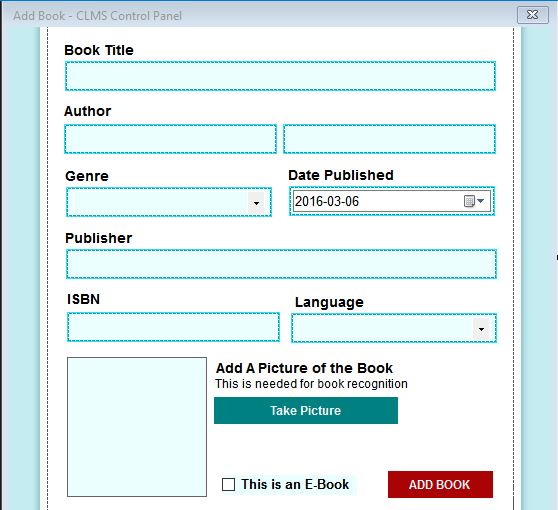
**Special Permission Authorization Grant**



**Modify Added Resources**

# 

**Add New Resource to the System**



# Implementation of Features

* **Effective and Convenient Search of Material**

An administrator or a member can easily search books, journals, magazine and etc. without the hassle of searching each and every shelf in the library. All materials are sorted and categorized, a customized search can also be performed.

* **Minimize the Physical Storage Area**

Need very limited physical storage area for the server of the system. The remaining physical free space can be allocated for computers, so members can search and read books with convenience.

* **Manage Members and Materials with Convenience**

Administrators can modify or remove stored materials without difficulty. The librarian doesn’t need to worry about borrowed books, when the duration of a borrowed book is finished the data in the member’s flash drive will self-destruct automatically.

* **Data Security**

All data stored are protected with an encryption that can only be decrypted by the administrator username and password. The database access keys are changed regularly to prevent unauthorized access via the internet.

* **Cost Reduction**

The institute can reduce expenses on employees employed at the library with the help of the system. A library can be managed by just one employee with the help of the system.

* **Easy to Use Graphical User Interface (GUI)**

Users can use the system with ease with the help of the elegant design and the structure of the system GUI.

* **Portability**

Physical books cannot be shared by multiple users but with eBooks any number of users can access the same book without complications.

* **Borrowing Period**

A member can extend the period of the borrowed books by informing the librarian

* **Member Registration**

The member will get an electronic card after registering with the system. The member should always keep the card with him or her to get access with the system.

# Quality

The library management system is successfully completed and implemented all the mentioned functionalities of the system. We have used high quality C# codes to build the core functionalities of the system. MS SQL Server is used to implement the server side database of the system and MS SQL Server is used to handle the database aspects of the system.

If a certain method needs to be optimized or debugged, it can be easily done by changing the methods in the core classes (Validations, Encryptions, and GU Effects) because every important function is declared in the core classes and used inside other classes. I used the divide and conquer principle to the coding to make the process simple and easy to understand and implement.

The System communicates with three main servers to function the system and to provide needed guidance to work the system.

OOP Concepts Inheritance, Polymorphism, Encapsulation, Abstraction are used.

File Handling, Exception Handling and Database connection are also implemented in the system.

## 

# Personal Reflection

As mentioned earlier, this project is a group work and we have been able to complete the project successfully according to the given scenario. By doing this project we have gained the knowledge of building a library management system. While doing the project we got the experience of working as a team and to overcome problems while doing the project. The time we spend to complete this project was very enjoyable. Finally we have to say that this project was a great opportunity to show our work and next time we will come with a more advance system in the future.

We managed to implement the full functionality of the system as stated above. The experience we gained really changed my coding style and my way of approaching a problem to find a solution.

Webcam video recording, that’s the name everyone uses but its actually capturing images at miliseconds from your webcam and compiling them together to form a video and you have to implement audio recording in the background as a thread to record audio. It took me a long while to realize this small concept, but finally found a nice API to do the task.

We gained knowledge about working with sound and video processing in Visual Studio C# and also managed to handle file and exception processing. The main knowledge stream we gained was through the use of different API and libraries.