

A PROJECT REPORT ON

Digital LogBook

**Submitted in partial fulfilment of the requirements for the award of
the degree of**

**BACHELOR OF COMPUTER APPLICATION
OF
BANGALORE NORTH UNIVERSITY**



Submitted by

Ankit Kumar Yadav(R2012508)

Ijaz Ahmad S(R2012536)

Akshay S Nair(R2012506)

2022-2023

GUIDED BY

Ms. Helaria

ASSISTANT PROFESSOR

DEPARTMENT OF COMPUTER SCIENCE



Kasturi Nagar, Bangalore –
560043

CERTIFICATE

This is to certify that **Ankit Kumar Yadav(R2012508), Ijaz Ahmad S(R2012536), Akshay S Nair(R2012506)** are a Bonafide students of New Horizon College and has carried out a project entitled **DIGITAL LOGBOOK** under the guidance of Ms. HELARIA. This project report has been submitted during the academic year 2022-2023 in partial fulfilment of requirements of the Degree in Bachelor of Computer Application, Bangalore North University.

Ms. HELERIA
Assistant Professor

Dr T. LOGESWARI
Guide Associate Professor
Head Of the Department

Dr BASVARAJ BIRADAR
Principal

Date: March 2023
Place: NHCK,
Bangalore

DECLARATION

We, **Ankit Kumar Yadav(R2012508)** ,**Ijaz Ahmad S(R2012536)** and **Akshay S Nair(R2012)** do hereby declare that the project work entitled **DIGITAL LOGBOOK** is a Bonafide work carried out by us under the guidance of **Ms Helaria** .This project is our original work and as not been presented for any other university. This project has been submitted as part fulfilment of requirements for the Degree of Bachelor of Computer Application, Bangalore North University.

Date: March 2023

Place: NHCK
Bangalore

Ankit Kumar Yadav
Ijaz Ahmad S
Akshay S Nair

ACKNOWLEDGEMENT

It is a great pleasure to express my sincere gratitude to those who have contributed to my learning, through their support, encouragement and guidance.

I sincerely thank the Chairman, **Dr Mohan Magnani** of —**New Horizon College** for providing me the necessary facilities for the completion of the project.

With great pleasure I place on record my deep gratitude to the Principal, **Dr. Basvaraj Biradar** for providing all the facilities for the completion of the project.

I am grateful for the co-operation and valuable suggestions rendered by my guide **Ms. Helaria**, Assistant Professor for her sincere assistance, valuable guidance and careful reviews, without which my effort would have been in vain.

Thanks for the inspiration given to me in my task by her kind appreciation and suggestion through my project.

I also express my gratitude to all the faculty members of the Department of Computer Science.

Finally, I thank my parents, friends and colleagues who have always been a great inspiration to the success of my project.

Ankit Kumar Yadav(R2012508)
Ijaz Ahmad(R2012536)
Akshay S Nair(R2012506)

CONTENTS

SL. NO	SUBJECT	PG. NO
1.	SYNOPSIS	8
2.	INTRODUCTION	10
3.	FUNCTIONAL SPECIFICATION	13
4.	REQUIREMENTS SPECIFICATION <ul style="list-style-type: none"> • Hardware Requirements • Software Requirements 	16
5.	SYSTEM ANALYSIS <ul style="list-style-type: none"> • Existing System • Proposed System 	18
6.	PERT CHART	21
7.	TECHNOLOGIES USED <ul style="list-style-type: none"> • Microsoft Visual Studio 2010 [Front End] • Microsoft SQL Server Management Studio 2008 [Back End] 	23
8.	SYSTEM DESIGN <ul style="list-style-type: none"> • System design • Database design 	38
9.	SYSTEM IMPLEMENTATION <ul style="list-style-type: none"> • Entity-relationship diagram (ERD) • Table structure 	43
10.	SCREEN SHOTS <ul style="list-style-type: none"> □ Output screens 	51
11.	CODING	57
12.	TESTING	67
13.	FUTURE ENHANCEMENT	71
14.	CONCLUSION	73
15.	BIBLIOGRAPHY	75

LIST OF FIGURES

Figure No.	Figures	Pg. No.
1.	Pert Chart	
2	Entity Relationship Diagram	
.		
3.	Splash Screen	
4.	Login	
5.	Logged	
6.	Register	
7.	Admin Login	
8.	Admin Logged	
9.	Delete user	
10.	Test Cases	

LIST OF TABLES

Table No.	Table Contents	
1.	User Logs	
2.	Admin Logs	
3.	User Register	
4.	Report/Feedback	

SYNOPSIS

SYNOPSIS

A computer lab digital logbook is a digital record-keeping system that is used in computer labs to track the usage of the facility by students and staff. The logbook typically contains information such as the date and time of entry and exit from the lab, the user's name or ID number, and the purpose of their visit.

The purpose of a digital logbook is to provide an efficient and accurate method of tracking usage of the lab, as well as monitoring the activities of the users. It can also help with security and safety by allowing administrators to quickly identify who is in the lab at any given time.

In addition to tracking usage, a digital logbook can also be used to monitor equipment usage and maintenance. This can help administrators identify which equipment is being used the most and may need to be replaced or serviced.

Some digital logbook systems may also allow for the recording of specific activities within the lab, such as printing or software usage. This can provide valuable data for administrators to analyze and use to optimize the lab's resources.

Overall, a computer lab digital logbook can be a valuable tool for managing and optimizing the use of computer lab resources, as well as ensuring the safety and security of the facility and its users.

INTRODUCTION

INTRODUCTION

A digital logbook is a computer-based system that is used to record and track data in a computer lab. This system replaces the traditional paper-based logbook that is used to document all activities and actions performed in the lab. The digital logbook can be accessed and updated by authorized personnel from any location using a computer or a mobile device.

Digital logbooks are essential in computer labs as they provide an accurate record of all the activities that take place in the lab. This includes the use of hardware and software, maintenance and repairs, system updates and configurations, as well as any incidents or problems that may occur. This information is critical for tracking and improving the performance of the lab, ensuring compliance with regulations, and for auditing purposes.

In addition to tracking lab activities, digital logbooks can also be used to monitor lab inventory. This includes tracking the availability and usage of hardware components, such as computers, printers, and networking devices, as well as software licenses and other resources. By maintaining accurate inventory records, lab managers can ensure that the lab is properly equipped and that resources are being used efficiently.

Digital logbooks are also beneficial for managing lab security. Access to the lab can be controlled through the logbook, with only authorized personnel being granted entry. The logbook can also be used to monitor and track the use of sensitive information and data, ensuring that it is only accessed by authorized personnel.

One of the major advantages of using a digital logbook is the ability to generate reports and analytics. The system can generate reports on lab usage, inventory, and incidents, which can be used to identify trends, patterns, and areas for improvement. This information can be used to optimize the lab's performance, improve efficiency, and reduce costs.

Digital logbooks also offer a number of practical benefits. They are user-friendly, easy to update, and can be accessed from any location with an internet connection. They eliminate the need for paper-based logs, which can be lost or damaged, and reduce the time and effort required to maintain accurate records.

In conclusion, digital logbooks are essential in computer labs as they provide an accurate record of lab activities, inventory, and security. They offer a range of benefits, including the ability to generate reports and analytics, improve efficiency, and reduce costs. By adopting a digital logbook system, lab managers can ensure that their lab is operating at its best, and that they have the necessary information to make informed decisions.

FUNCTIONAL SPECIFICATION

1) Splash Screen

2) Login

3) Logged

4) Register

5) Admin Login

6) Admin Page

7) Delete user

8 Selection

1. Splash Screen

Loadin Screen to the Home page

2. Login

The user can get SignIn here to make entry.

3.Logged

The Logged user details are displayed here.

4.Register

On this form New user can rigister himself.

5.Admin Login & Register

Admin can login here & new admin can be registered too

6.Admin Page

Admin can login here & new admin can be registered too

7.Delete users

Admin can login here & new admin can be registered too

8. selection

User specific selections can be made here.

REQUIREMENT SPECIFICATION

REQUIREMENT SPECIFICATION

The purpose of this SRS document is to provide a detailed overview of our software product, its parameters and goals. This document describes the project's target audience and its user interface, hardware and software requirements. The Software Requirement Specification is documented in such a way that it breaks the deliverables into smaller components. The information is organized in such a way that the developers will not only understand the boundaries within which they need to work, but also what functionality needs to be developed and in what order. These two points are particularly important in the process of software development. If your development team do not understand that there are certain constraints on their work, as for example the code must be tightly written so that it will compile and run quickly, then you will run into problems later on when the code might deliver the functionality required, but no one will ever see it because it takes so long to load.

1. HARDWARE REQUIREMENTS

- Processor Intel Pentium IV 2.66
- Main Memory GHz 512 MB
- Hard Disk Capacity 160 GB
- Keyboard 101 keys
- Monitor VGA with resolution

2. SOFTWARE REQUIREMENTS

- Operating System Windows 10 Front
- End Visual Basic 6.0
- Back End SQL SERVER

SYSTEM ANALYSIS

SYSTEM ANALYSIS

INTRODUCTION TO SYSTEM ANALYSIS

A software requirements specification (SRS) is a description of a software system to be developed. It lays out functional and non-functional requirements and may include a set of use cases that describe user interactions that the software must provide. Software requirements establishes the basis for an agreement between customers and contractors or suppliers (in market-driven projects, these roles may be played by the marketing and development divisions) on what the software product is to do as well as what it is not expected to do. Software requirements specification permits a rigorous assessment of requirements before design can begin and reduces later redesign. It should also provide a realistic basis for estimating product costs, risks, and schedules. Used appropriately, software requirements specifications can help prevent software project failure.

1. EXISTING SYSTEM

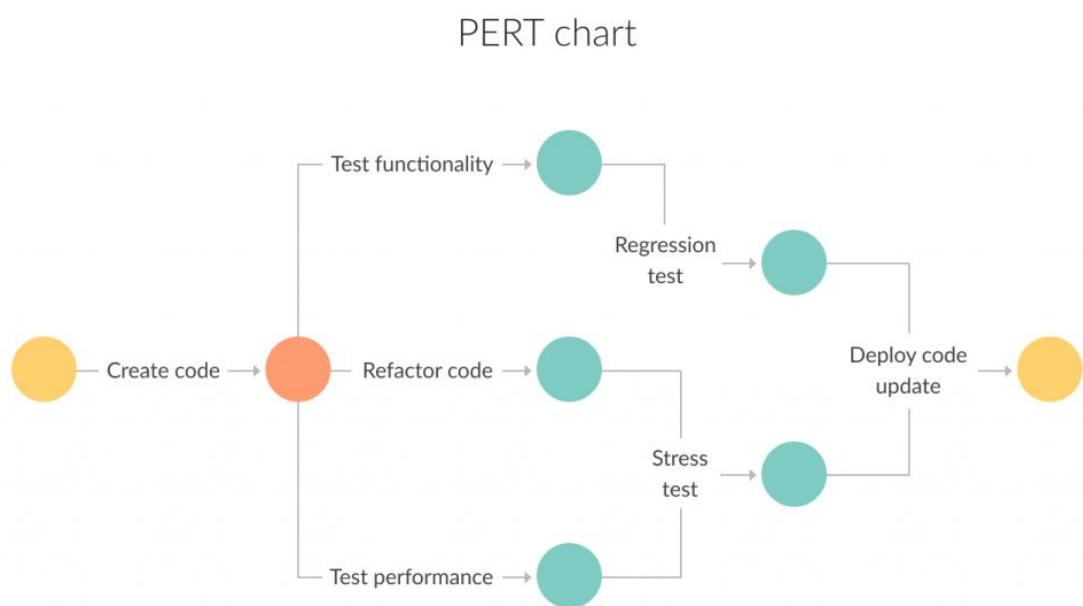
1. The data is written manually in books.
2. Duplication of data occurs in this system.
3. Data is not secured.
4. Time consuming.
5. External resources such as pen and papers are required.
6. Chances of loosing the Data.

2. PROPOSED SYSTEM

1. This system helps to overcome the shortenings of the existing system.
2. It is more secure and safety.
3. Using this application ,one can add,modify,update,save,delete details.
4. It is more reliable,fast,user-friendly and information.
5. Has added Security.
6. Backup of data is made.
7. can be accessed from diffrent devices.
8. No external resources are required.

PERT CHART

PERT CHART



TECHNOLOGIES USED

TECHNOLOGIES USED

INTRODUCTION TO VB

"Visual Basic is a third-generation event-driven programming language and integrated development environment(IDE) from Microsoft for its Component Object Model(COM) programming model first released in 1991 and declared legacy during 2008.

Microsoft intended Visual Basic to be relatively easy to learn and use."Visual Basic was derived from BASIC, a user-friendly programming language designed for beginners, and it enables the rapid application development(RAD) of graphical user interface(GUI) application, access to database using Data Access Object, Remote Data Object, or ActiveX Data Object, and creation of ActiveX controls and object.

A programmer can create an application using the components provided by the Visual Basic program itself. Over time the community of programmers developed third-party components. Programs written in Visual Basic can also use the Windows API, which requires external function declaration.

The final release was version 6 in 1998(now known simply as Visual Basic).

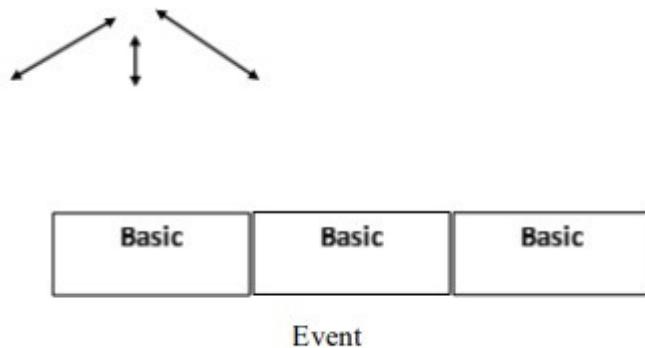
On April 8, 2008, Microsoft stopped Visual Basic 6.0 IDE. The Microsoft Visual Basic term still maintains compatibility for Visual Basic 6.0 application on Windows vista, Windows Server 2008 including R2, Windows 7, Windows 8, Windows 8.1, Windows Server 2012 and Windows 10 through its "It Just Works" program. In 2014, some software developers still preferred Visual Basic 6.0 over its successor, Visual Basic.NET. In 2014 some developers lobbied for a new version of Visual Basic 6.0. In 2016 Visual Basic 6.0 won the technical impact award at The19th Annual D.I.C.E. Awards. A dialect of Visual Basic, Visual Basic for Application (VBA), is used as a macro or scripting language within several Microsoft application, including Microsoft Office.

Some of the main features of Visual Basic 6.0 are listed below:

- Access features allow you to create database and front-end applications for most popular database formats, including Microsoft SQL Server, Oracle, Microsoft Access and other enterprise level database.
- It includes a GUI environment for making windows-based applications.
- X technology allows you to use the functionality provided by other application such as MS Word, MS Excel and other Windows applications.
- Internet capabilities make it easy to provide access to documents and applications across the Internet from within your application.
- It provides a vital link to graphical environment and allows you to develop applications based on standard Windows features: Dialog Boxes, Command buttons, pull down menus, Scroll bars, Selection lists etc. It also allows creating robust applications that fully make use of the graphical user interface.
- A multitude of wizards and other graphical tools aid developers new to Visual Basic.
- ADO-compliant data-bound controls.
- Hierarchical record sets and the Flex Grid Control.
- Visual Basic is an event driven programming language.
- Visual Basic allows you to adopt more of parallel approach, with independent sections of code for each option that the user may select. This is known as Event driven programming language.
- Visual data tools (VDTs).
- ADO Data Control (ADODC)
- Data report design and Data form wizards.
- It also helps the user with the SQL editor.
- By connecting it with Oracle, SQL statements can be run and terminated through Visual Basic 6.0.
- Visual Basic Component creation.

- The language is very easy and it provides a very user-friendly environment while programming in Visual Basic 6.0.
- Packaging and Deployment wizard.
- Allows for the creation of p-code and native code EXE files. p-code is a tokenized form of your source code that will be broken down at runtime into
- Machine code, which is why Visual Basic will create this intermediate forms.
- Can be extended easily through the use of windows API calls, hundreds of third-party controls and DLLs, and integration with other windows applications through COM and DCOM.
- Has a shorter learning curve and development time than C/C++, Delphi, and even Power Builder.
- Used by most of the office suite tool as macro language. With the rest to follow. Other companies as well are starting to support VBA in their products, such as AutoCAD, Vision, CorelDraw, SAP, and many others.
- Allows for rapid application development and is excellent for business applications. Has an excellent integrated help facility and book online as well as it includes good debugging facilities and have many wizards that help automated repetitive tasks.
- Object-based development is possible using class modules and rapid application development (RAD).
- Allows for the creation of COM components such as Active X controls, DLLs, and Exec's. Can integrate with the Internet on both the server side and the client side.
- Can create Active X Automation server. Integrates with Microsoft transaction server. Can run server either on the same machine or remotely on another computer. This allows for true distributed processing.

What is Visual Basic?



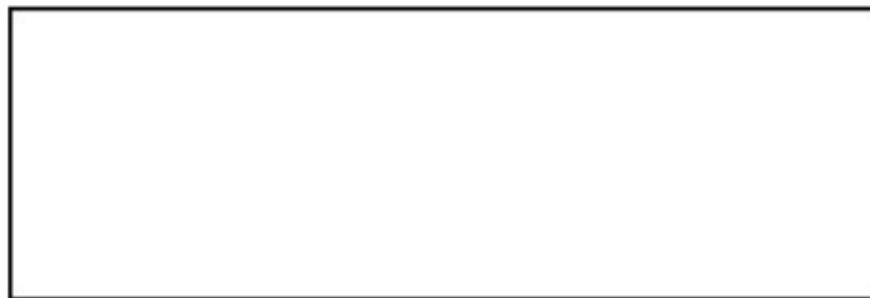
All Windows applications are event-driven. For example, nothing happens in Word until you click on a button, select a menu option, or type some text. Each of these actions is an event.

Structure of a Visual Basic Application Project (.VBP, .MAK)

Module 1 (.BAS)

Form 3 (.FRM)

Form 2 (.FRM)



Form 1 (.FRM)

Control 3 Control 3 Control 1 Control 1 Control 2 Control 3 Control 1 Control 2



Steps in Developing Application

- The Visual Basic development environment makes building an application a straightforward process. There are three primary steps involved in building a Visual Basic application:
 1. Draw the user interface by placing controls on the form
 2. Assign properties to controls
 3. Attach code to control events (and perhaps write other procedures) These same steps are followed whether you are building a very simple application or one involving many controls and many lines of code



We assume you have Visual Basic 6 installed and operational on your computer. If you don't, you need to do this first. To start Visual Basic:

Click on the Start button on the Windows task bar.

Select Programs, then Microsoft Visual Basic 6

Click on Visual Basic 6 For now, just click Open – we are starting a new project. Later, once you have saved some projects, they can be opened using the Existing and Recent tabs.

The Visual Basic development environment will start. Application (Project) is made up of:

Drive List Box	Command Button	List Box	Vertical Scroll Bar
File List Box	Lines	Data Tool	Option Button
Picture Box	Text Box	Pointer	Label

Timer	Horizontal Scroll Bar	Combo Box	Check Box
Frame	Directory List Box	Shapes	Image Box

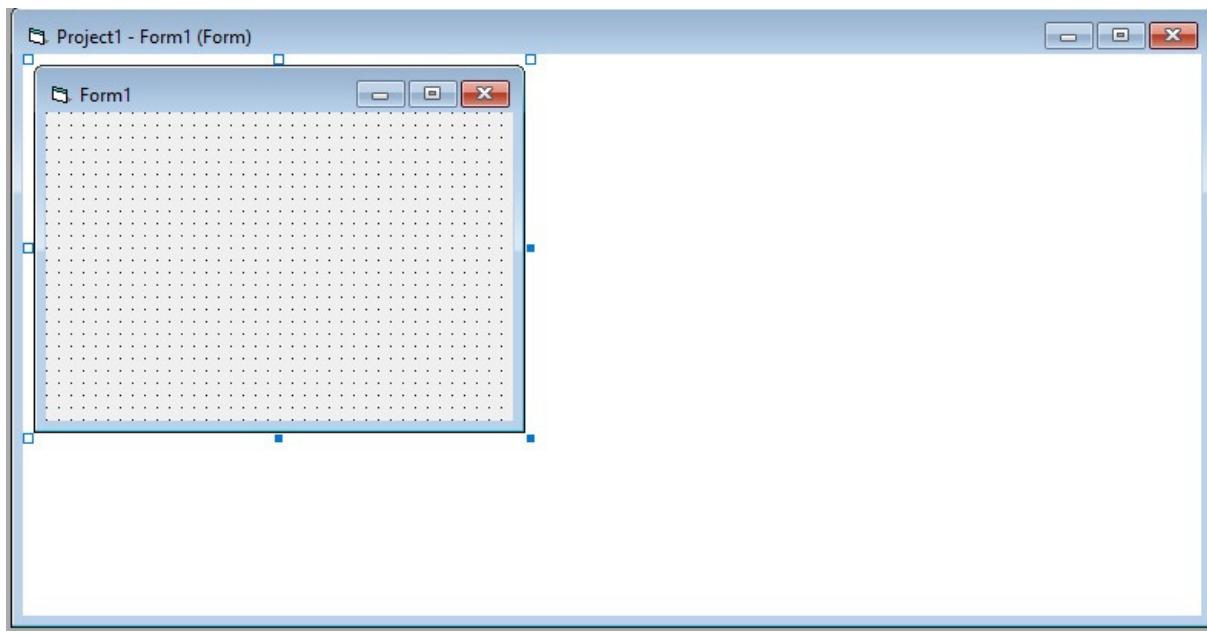


Object Linking Embedding

You can add menus, tool bars, status bars, text boxes, etc. to blank window.

Disadvantages of Visual Basic 6.0: -

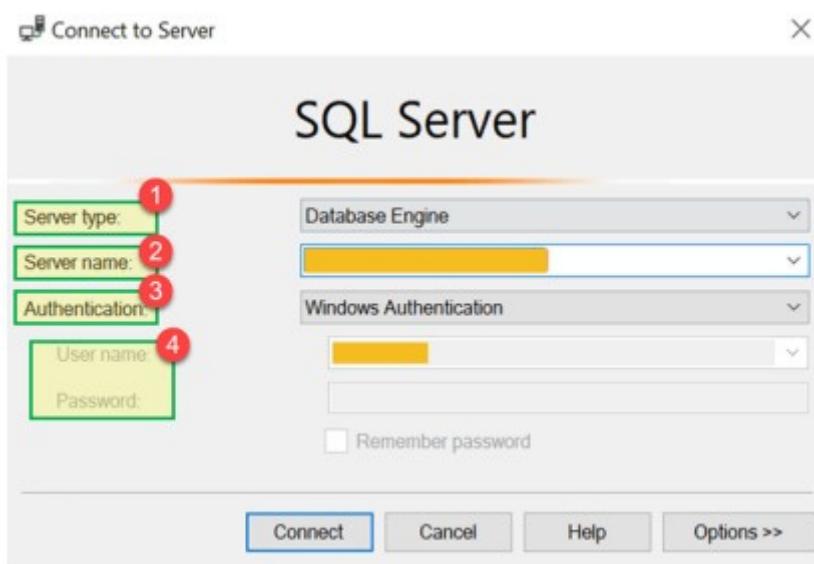
The programs that are developed utilize more memory. Visual Basic requires specific operating system, which supports visual programming. Graphical User Interface is provided by visual basic, which takes some more time than other non-visual programming for execution of the program



Introduction to SQL SERVER

Microsoft SQL Server is a relational database management system developed by Microsoft. As a database server, it is a software product with the primary function of storing and retrieving data as requested by other software applications—which may run either on the same computer or on another computer across a network (including the Internet). Microsoft markets at least a dozen different editions of Microsoft SQL Server, aimed at different audiences and for workloads ranging from small single-machine applications to large Internet-facing applications with many concurrent users.. Storage space allocated to a database is divided into sequentially numbered pages, each 8 KB in size. A page is the basic unit of I/O for SQL Server operations. A page is marked with a 96-byte header which stores metadata about the page including the page number, page type, free space on the page and the ID of the object that owns it. The page type defines the data contained in the page. This data includes: data stored in the database, an index, an allocation map, which holds information about how pages are allocated to tables and indexes; and a change map which holds information about the changes made to other pages since last backup or logging, or contain large data types such as image or text. While a page is the basic unit of an I/O operation, space is actually managed in terms of an extent which consists of 8 pages. A database object can either span all 8 pages in an extent ("uniform extent") or share an extent with up to 7 more objects ("mixed extent"). A row in a database table cannot span more than one page, so is limited to 8 KB in size. However, if the data exceeds 8 KB and the row contains varchar or varbinary data, the data in those columns are moved to a new page (or possibly a sequence of pages, called an allocation unit) and replaced with a pointer to the data. For physical storage of a table, its rows are divided into a series of partitions (numbered 1 to n). The partition size is user defined; by default all rows are in a single partition. A table is split into multiple partitions in order to spread a database over a computer cluster. Rows in each partition are stored in either B-tree or heap structure. If the table has an associated, clustered index to allow fast retrieval of rows, the rows are stored in-order according to their index values, with a B-tree providing the index.

The data is in the leaf node of the leaves, and other nodes storing the index values for the leaf data reachable from the respective nodes. If the index is non-clustered, the rows are not sorted according to the index keys. An indexed view has the same storage structure as an indexed table. A table without a clustered index is stored in an unordered heap structure. However, the table may have non-clustered indices to allow fast retrieval of rows. In some situation the heap structure has performance advantages over the clustered structure.

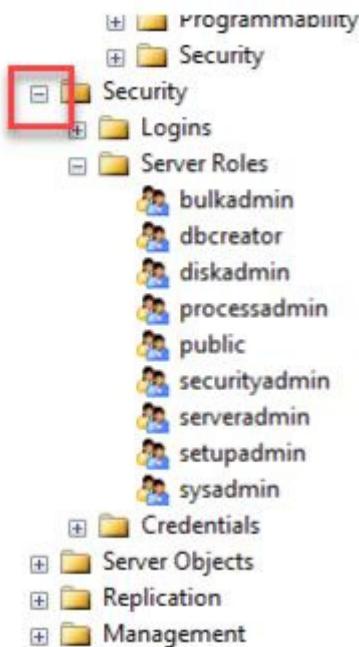


Let's understand each of the above fields

Server type: This is an option to select one out of four available MS SQL services option. We will be working on 'Database Engine' for creating and working with Database. Other Server type includes Analysis, Reporting & Integration Services.

Server name: This is Server's name where MS SQL Server is installed and need to establish the connection with that server. Generally, we use the server name as "Machine name\Instance." Here Instance is the name given to SQL Server instance while SQL server installation

Authentication: This is defaulted to "Windows Authentication" if we use "Windows Authentication" during SQL Server Installation. Else, if we select 'Mixed Mode (Windows Authentication & Windows Authentication)' then Authentication will be defaulted to "SQL Server Installation."



2) Databases Selection Dropdown

This dropdown allows the user to select the Database in which we will be running our queries.

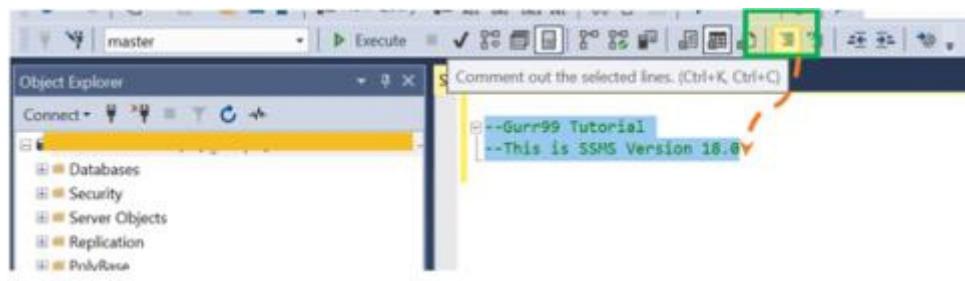
3) Query Editor Here we can write all our queries. MS SQL server provides interactive suggestions for tables, columns, etc. for easy queries creations and much more.

4) Execute button This button will finally execute the query and return the results. SSMS Tips and Issues

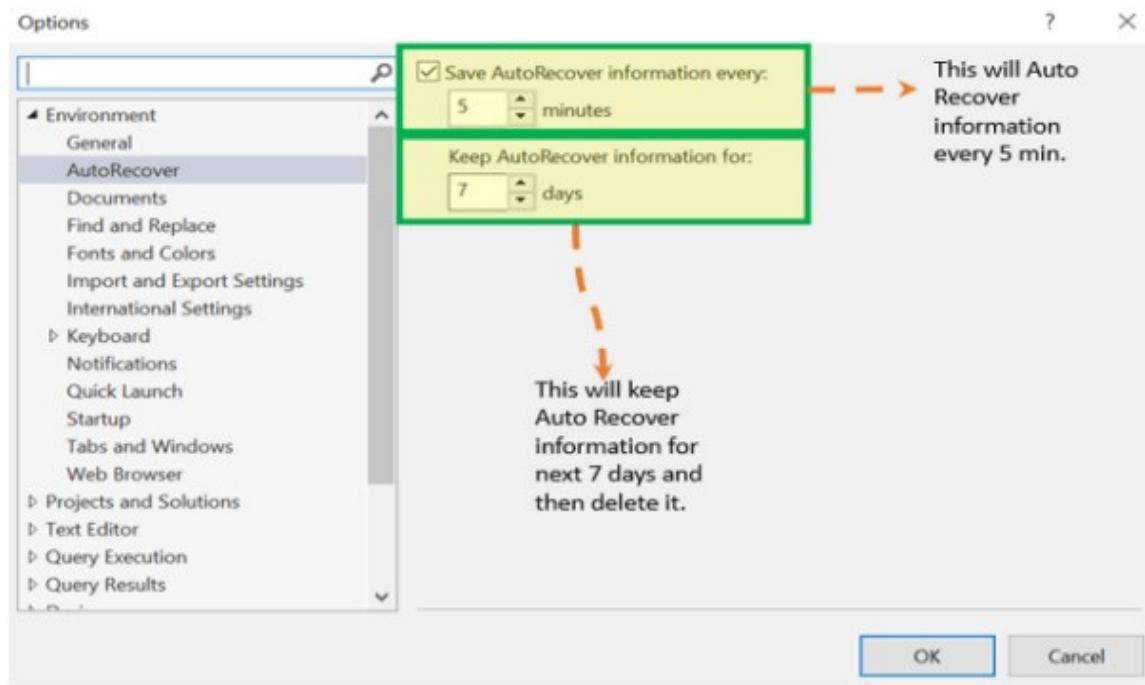
- Management Studio is a standalone product. It does not correspond to any specific version of SQL Server.

For example, we can use SMMS Version 18 with SQL Server 2017, SQL Server 2016 as well.

- Large codes reduce readability. Use comments for better Readability. Put "--" in front of any line to comment it out.
- Group comment: We can comment out the group of line by selecting them all and clicking on Icon shown in below image.

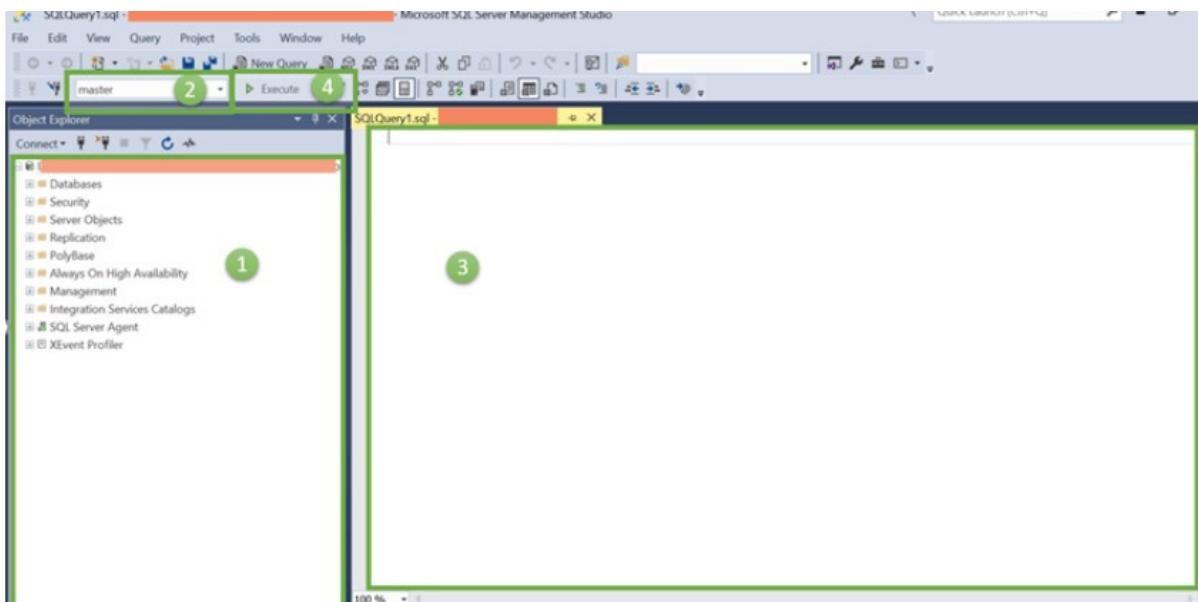


- Unexpected shutdown, a system failure can cause unexpected data loss. Set 'Auto recover' option checked to minimize data loss. We can even customized time interval to Auto Recover data and the number of days this information to be saved before deleting it.
- Tools> Options>Environment>AutoRecover



In case of failure, the popup window will appear with the name • We often need the result of our query to be saved in text format for future references.

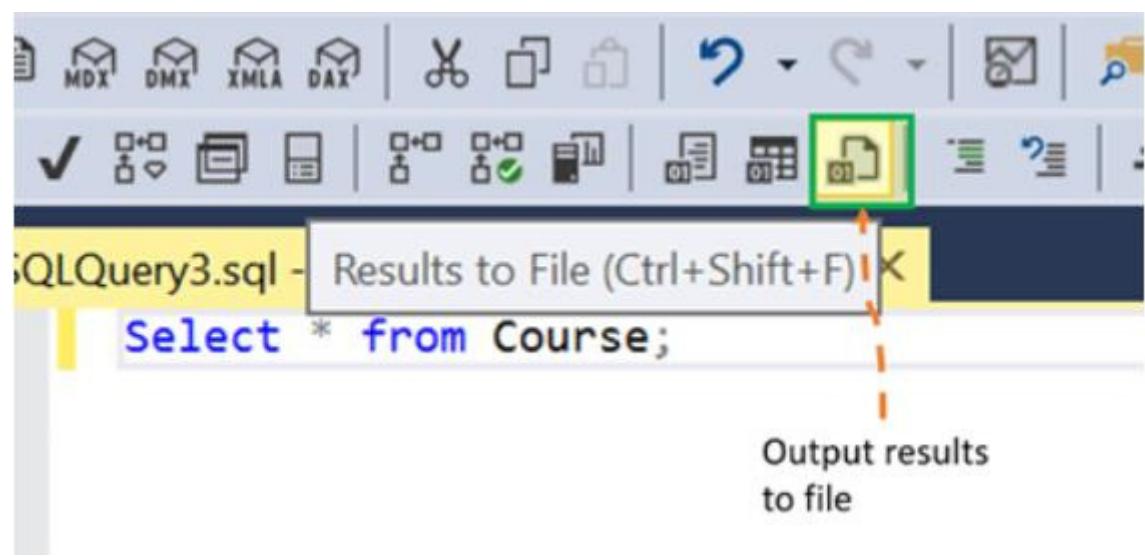
User name\Password: If Authentication is selected other than "Windows Authentication" like "SQL server Installation" then these two fields will be required. Click on 'Connect.' Now you will be connected to 'Data Management Studio.' 'Connect to Server' screen will appear in a similar way as described in the previous section



Let's discuss each section in detail.

1) Object Explorer

The Object Explorer provides a tree view of the database objects contained in the server. This section shows all the Databases, Security, Server Object for quick reference. To view the components of each object, just click the + icon located to the left of the object which will expand it.



SYSTEM DESIGN

SYSTEM DESIGN

INTRODUCTION TO SYSTEM DESIGN:

The design phase is the life cycle phase in which the detailed design of the selected system in the study phase is accomplished. In the design phase, the technical specifications are prepared for the performance of all allocated tasks. It also includes the construction of programs and program testing. In the design phase, the first step is to determine the output is to be produced and in what format. Second, input data and master files have to be designed to meet the requirements of proposed output. The system analyst has to define the methods of capturing and input programs and format of the output and its use by the users.

SYSTEM FLOW CHART:

A graphic representation of a system showing the overall flow of control in the processing at the job level; specifies what activities must be done to convert from a physical to logical model is known as a system flowchart. Thus, it summarizes what operations are undertaken and where and when they take place. Normally in a system flowchart input from outside are shown to the left and outputs to the right. Symbols representing the operations undertaken and the documents used are then placed in the appropriate places which gives a general flow of data from top to bottom and left to right. Arrows are used on the connecting lines to indicate the logical flow or sequence where the flow is not in the standard direction. No interaction is implied by crossing lines. Decisions which lead to different actions can also be shown

DATA FLOW DIAGRAM:

A data flow diagram is graphic representation of a system that shows data flows to, from and within the system, processing functions that change the date in some manner, and the storage of this data. They are networks of related system function that indicated form where information is revived and to where it is sent. An external entity is the originator or receiver of data or information. A data store symbol portrait a file or database in which data resides. A process is depicted by a circle sometimes it is called a bubble or transform. Process portraits the transformation of the content of status of data

DATABASE DESIGN:

This activity deals with the design of the physical database. The designer begins to concentrate on file design or how data should be organized around user requirements. How data are organized depends on the data and response requirements that determine hardware configurations. An integrated approach to file design is the database. The general theme is to handle information as an integrated whole, with a minimum of redundancy and improved performance, type and size of data structure used. The objectives of data base are accuracy and integrity, privacy and security of data etc

The general theme behind a database is to handle information as an integrated whole. A database is a collection of interrelated data stored with minimum redundancy to serve many users quickly and effectively. After designing input and output, the analyst must concentrate on database design or how data should be organized around user requirements. The general objective is to make information access, easy quick, inexpensive and flexible for other users.

CODE DESIGN:

Codes can provide brief identification of data items and replace longer descriptions that would be more awkward to store and to manipulate.

INPUT DESIGN: -

Input design is processing of converting the user-oriented description of the inputs of the system. The goal of designing input data to make data entry as easy logical and free from errors as possible.

In entering data, operators need to know the following.

1. The allocated space for each field.
2. Field sequence which must match that in the source document.
3. The format in which data fields are entered

for example, filling out the data field is required through the edited format mm/dd/ yy When we approach input data design, we design source documents that capture the data and then select the media used to enter them into the computer. There are different ways in which data can be introduced into the system such as

- a. The data is converted into a machine sensible form by some realistic source document and types in the relevant items using a keyboard connected to the system.
- b. The document can be read directly by a machine, and this converts information held in the human sensible form into a machine-readable form without need for human investigations.
- c. Data entered into a system through a keyboard. This is done interactively by the person using the system. The field name must be documented. The field name must be known to data entry operator or users so that the data entry will not exceed the allocated space. Our system contains the following inputs:

OUTPUT DESIGN: -

primary consideration in the design of all output is the information requirement and other objective of the users. It is the most important and direct source of information to the user. A major form of output is a hard copy. Print out should be designed around the output requirements of the user. Each output should be given a specific name or title. The output data is displayed on the visual display unit and output can be redirected to printers and or sorted in a file for later use. Here, in this system, program is designed so as to generate a number of relevant outputs displayed in various kinds of user- defined tables in an easily readable and comprehensive manner which can be readily read and understood by the user. So, no further attempt has been made to generate reports which of course could have been easily implemented into the system.

PROCEDURE DESIGN: -

When program become very long, they are divided into smaller programs or modules. These smaller programs can be written, tested and debugged separately. This technique of programming is known as modular programming. The advantages of modular programming are.

1. It is easy to write, test and debug a module.
2. Generally the modules of common nature are prepared, which can be used at many places.
3. The programmer can use the previously written programs.
4. If a change is to be made, it is made in the particular module; the entire program is not affected.

SYSTEM IMPLEMENTATION

SYSTEM IMPLEMENTATION

PROGRAM DEVELOPMENT

In the case of program development first of all the problem is defined. It includes input-output specifications, requirements, execution times, accuracy etc. A necessary system flowchart is expended to show additional detail input and out files are identified, and computer programs logic flowchart are prepared for each computer program component. An algorithm can also write to solve the problem.

The following are the stages for the development of software.

1. Problem definition
2. Program design
3. Coding
4. Debugging
5. Testing
6. Documentation
7. Maintenance, Extension, and Redesign.

The criteria for evolution of a program are reliability, speed hardware cost, programming time and cost of use error tolerance and extensibility. A good program should utilize memory and times efficiently. An interface should be simple and less costly as far as possible to perform a ascertain task. Good design and clear documentation make a program simple and it can be used by others.

SOFTWARE SELECTION

Software selection is critical aspect of system development. These are two ways of acquiring software custom-made or “off-the-shelf” packages. Today’s trend towards purchasing packages, which represent roughly 10 percent of what are costs to, developed same in house. Prior to selecting the software, the project team must setup criteria for software selection. Software readability brings up the concept of modularity. Functionally, it is definition of the e facilities, performance and other factors that the

user requires in the finished product. Capacity refers to the capability of the software package to handle the user's requirements. The criterion, usability refers to the effort required to operate, prepare the input and interpret the output program. Serviceability focuses on documentation and vendor support lost is major consideration. The other criteria are flexibility, security, performance, and ownership.

SECURITY FEATURES

Every candidate system must provide built in for security and integrity of data. Without safeguards against unauthorized access and natural disasters, a system could be so vulnerable as to threaten survival of the organization. To do an adequate job on security, a system analyst must analyse the risks, exposure and costs and specify measures such as passwords and encryption to provide protection. In addition, backup copies of software and recovery re start procedures must be available when needed.

1.Table Scheme

1.Users

	Column Name	Data Type	Allow Nulls
▶	user_id	int	<input type="checkbox"/>
	username	varchar(50)	<input type="checkbox"/>
	password	varchar(50)	<input type="checkbox"/>
	lab	varchar(50)	<input checked="" type="checkbox"/>
	system_number	varchar(50)	<input checked="" type="checkbox"/>
	subjectr	varchar(50)	<input checked="" type="checkbox"/>
	upTime	varchar(50)	<input checked="" type="checkbox"/>
			<input type="checkbox"/>

2.Admin Users

	Column Name	Data Type	Allow Nulls
▶	id	int	<input type="checkbox"/>
	username	varchar(255)	<input type="checkbox"/>
	password	varchar(255)	<input checked="" type="checkbox"/>
			<input type="checkbox"/>

3.Feedback

Column Name	Data Type	Allow Nulls
name	varchar(50)	<input type="checkbox"/>
lab	varchar(50)	<input type="checkbox"/>
system	varchar(50)	<input type="checkbox"/>
description	varchar(50)	<input type="checkbox"/>
		<input type="checkbox"/>

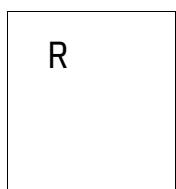
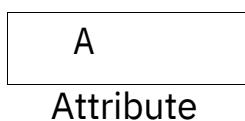
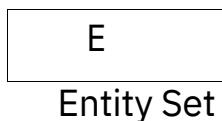
4.Report

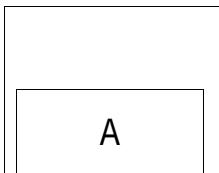
Column Name	Data Type	Allow Nulls
name	varchar(50)	<input type="checkbox"/>
type	varchar(50)	<input type="checkbox"/>
freq	varchar(50)	<input type="checkbox"/>
lab	varchar(50)	<input type="checkbox"/>
sys	varchar(50)	<input type="checkbox"/>
descri	varchar(50)	<input type="checkbox"/>
		<input type="checkbox"/>

2. ENTITY-RELATIONSHIP(ER) DIAGRAM

The entity-relationship (E-R) data model perceives the real world as consisting of basic objects, called entities and relationships among those objects. It was developed to facilitate database design by allowing specification of an enterprise schema, which represents the overall logical structure of a database. The E-R model is one of the several semantic data models; the semantic aspect of the model lies in representation of the meaning of the data. The E-R model is very useful in mapping the meanings and interactions of the real world enterprise onto a conceptual schema. Because of this usefulness, many database design tools draw on concepts from the E-R model.

Symbols mainly used in E-R diagram:





1.

Primary Key

E-R data model employs three basic notions: entity sets, relationship sets and attributes.

Entity set an entity is a thing or object in the real world that is distinguishable

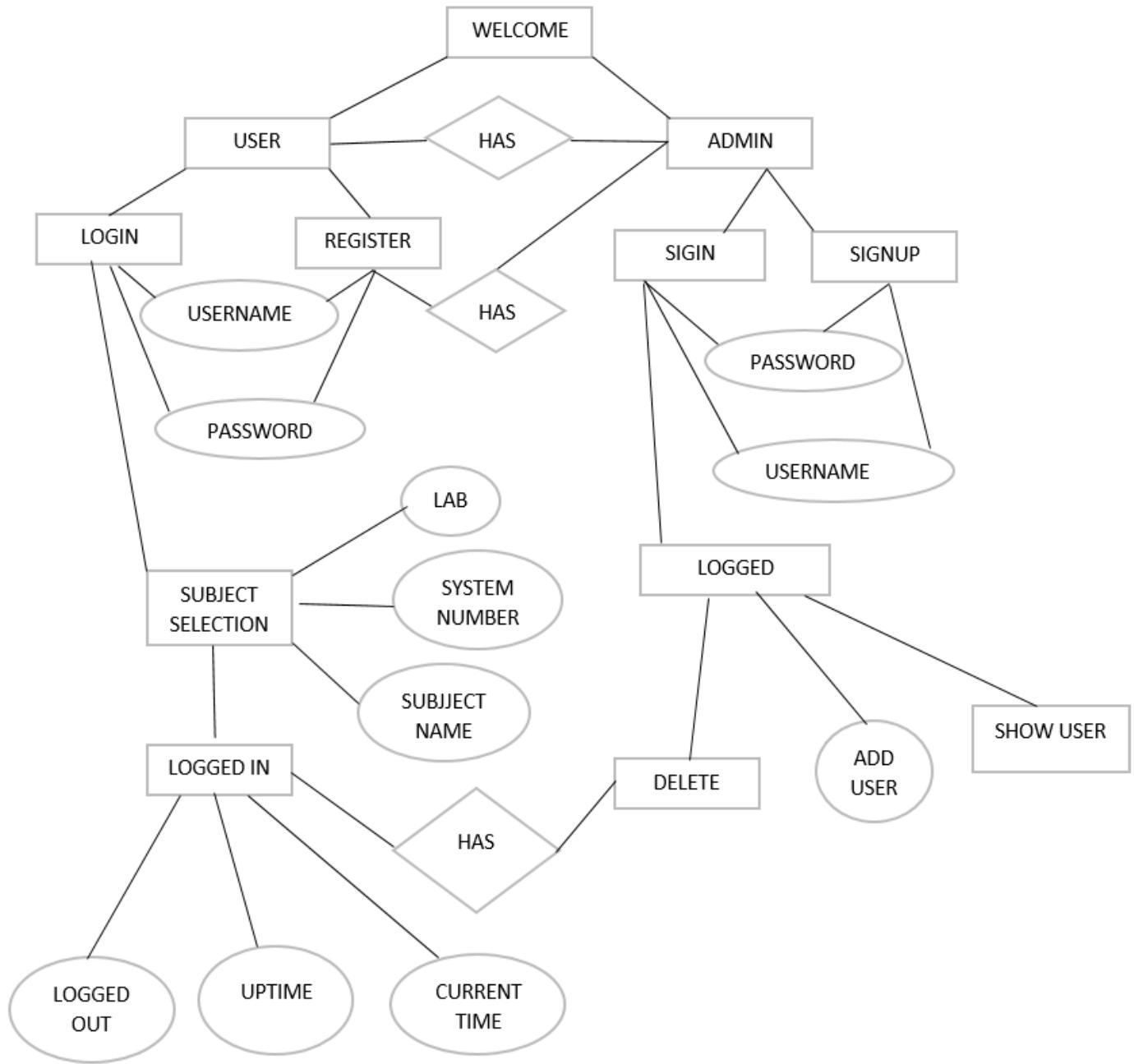
from all other objects. An entity has a set of properties and values of some properties uniquely define the entities. An entity set is the set of all the entities of same type that share same properties or attributes.

Attributes attributes are the descriptive properties possessed by each member of an

entity set. Each entity may have its own value for each attribute. For each attribute,

there is a set of permitted values, called as domain or value set of an attribute.

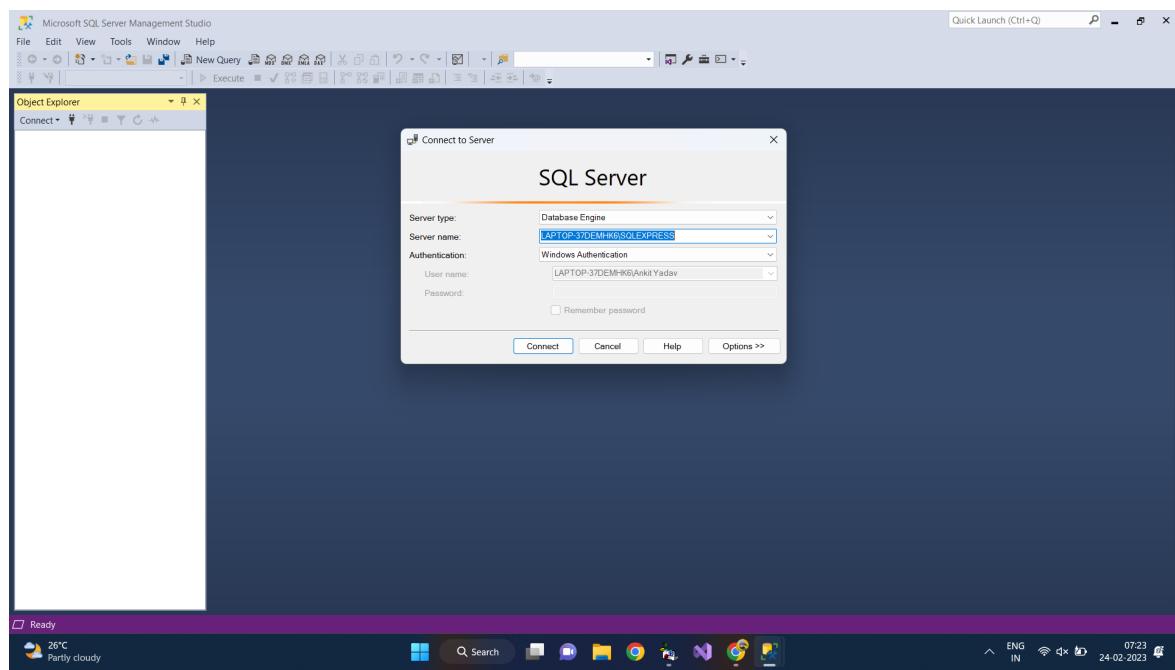
ER-DIAGRAM



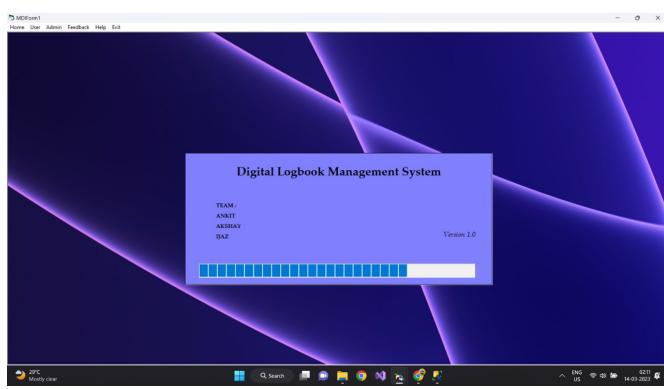
SCREEN SHOTS

SCREEN LAYOUT

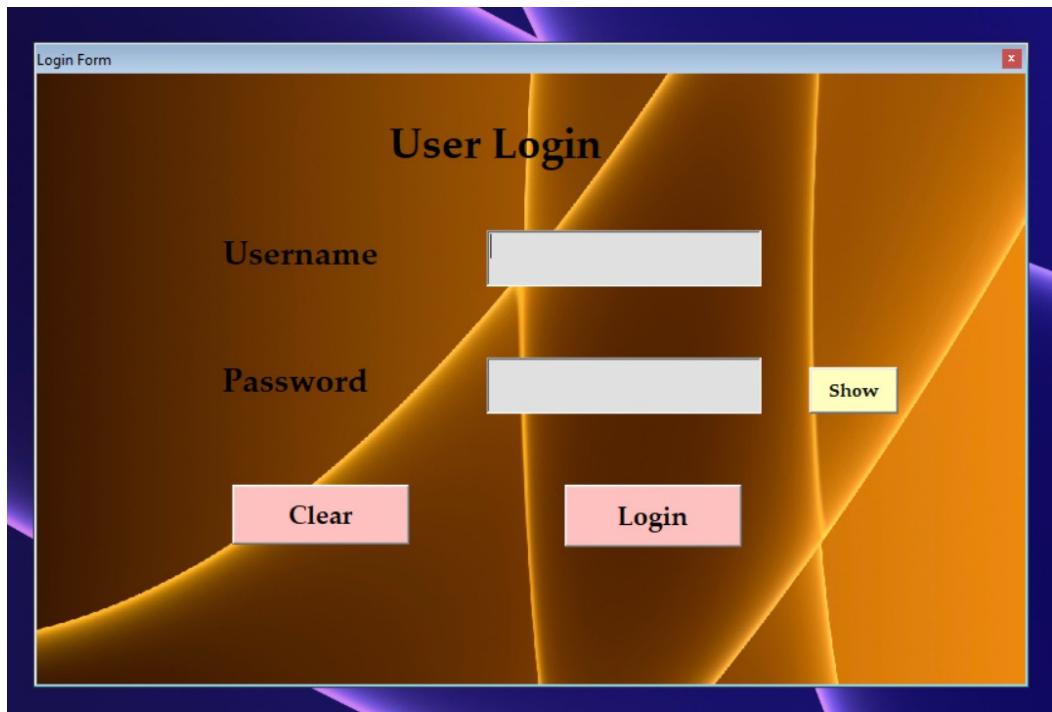
1) SQL Server Connection



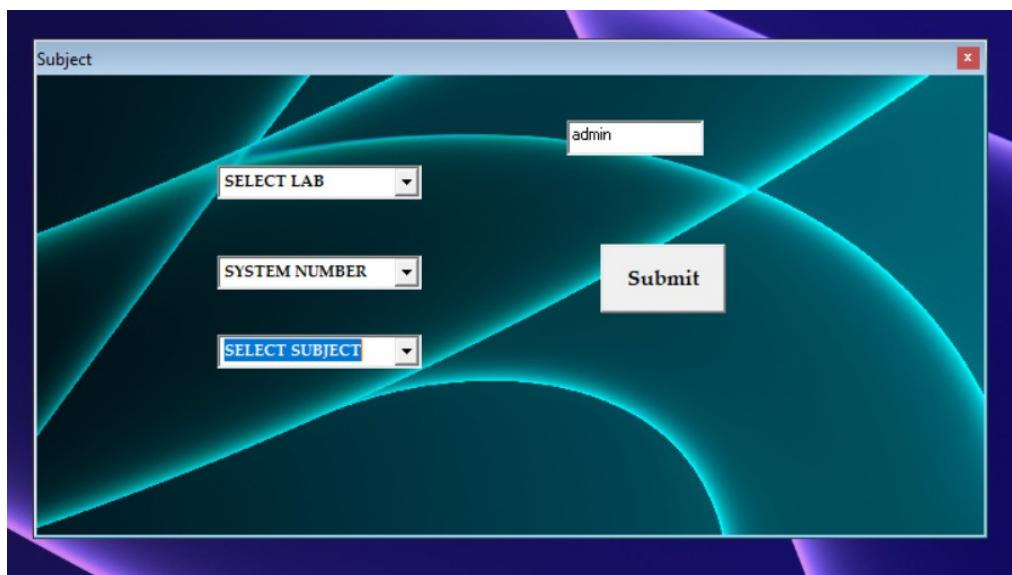
2) Splash Screen



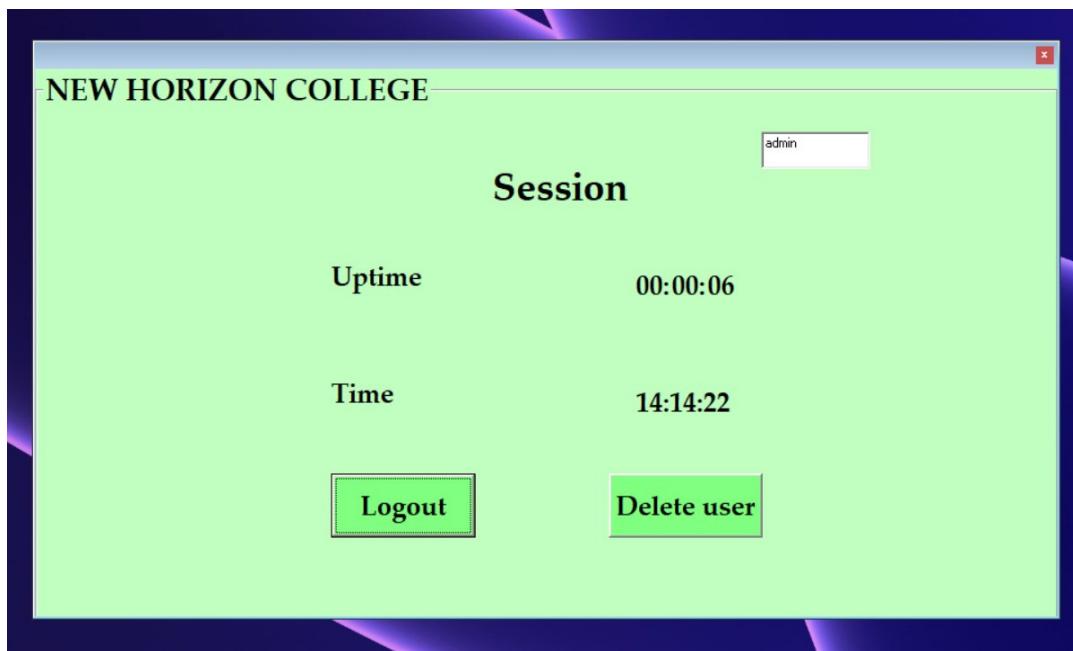
3) Login



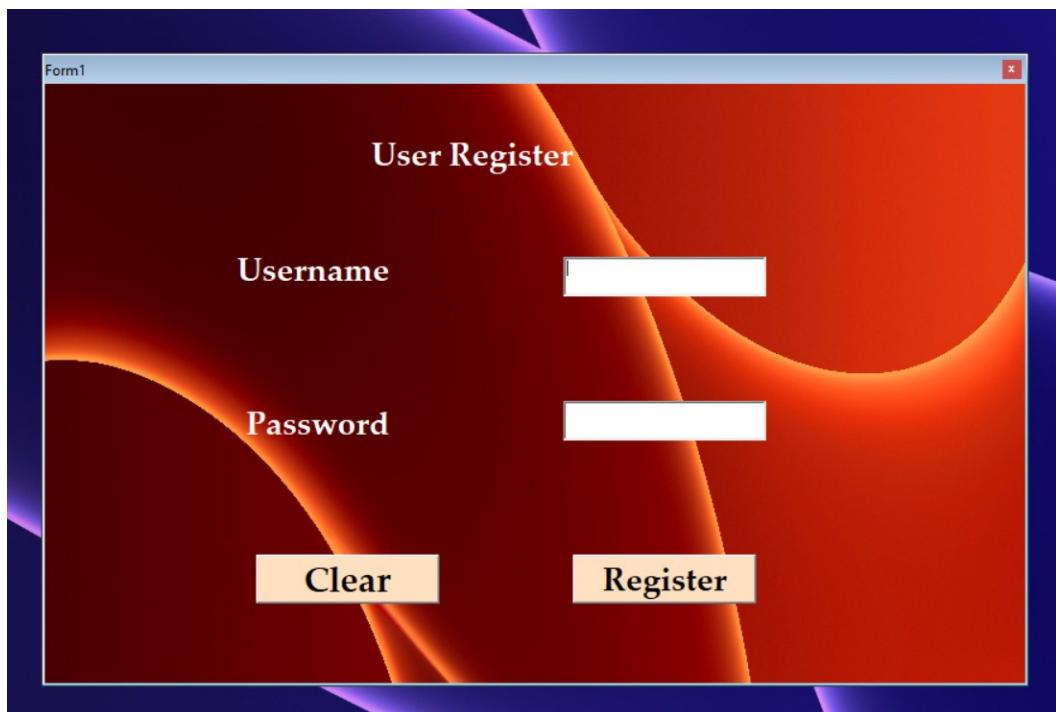
4) Subject Selection



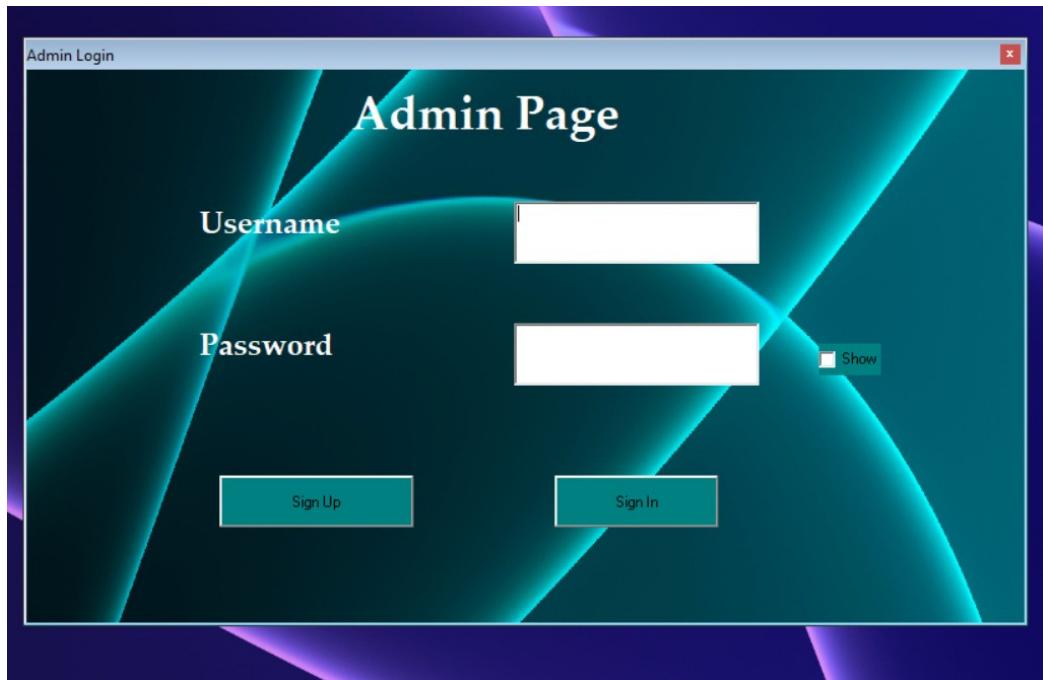
5) Logged



6) Register



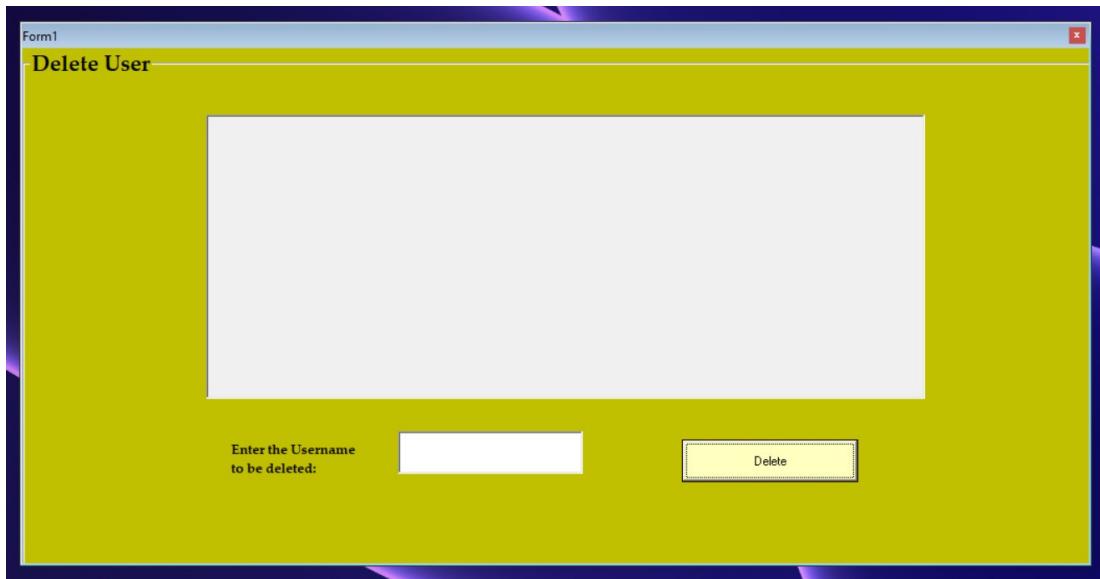
7) Admin Login



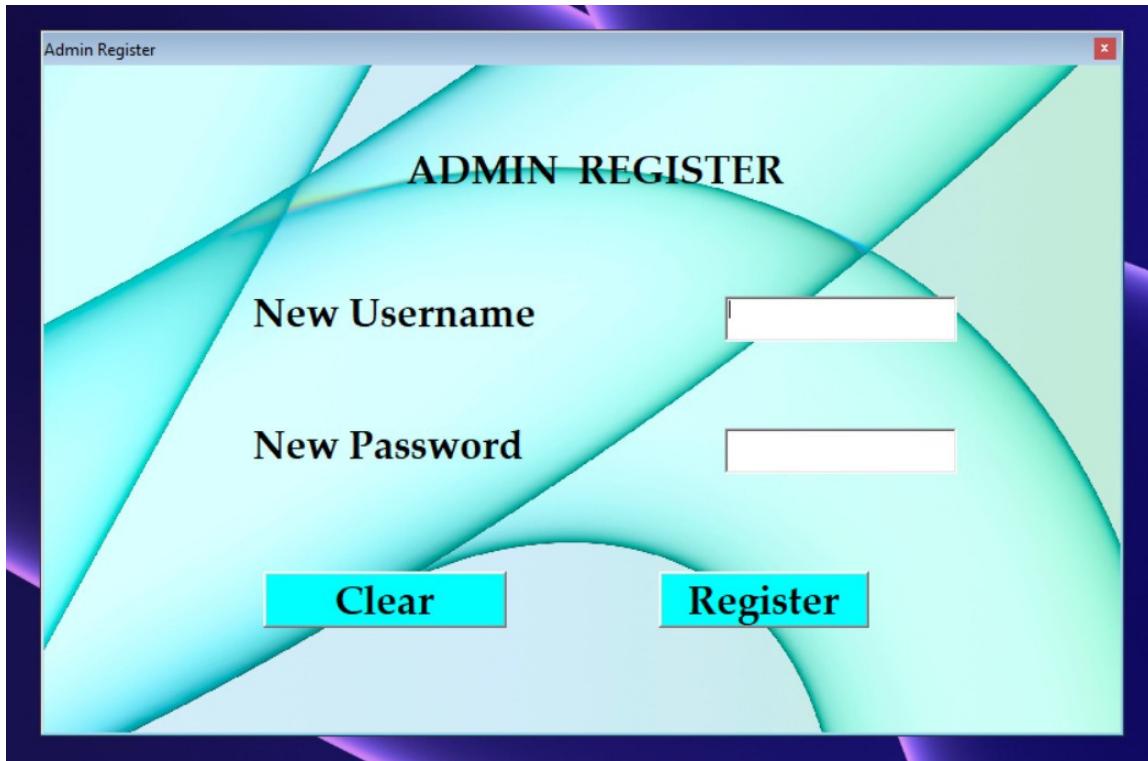
8) Admin Logged



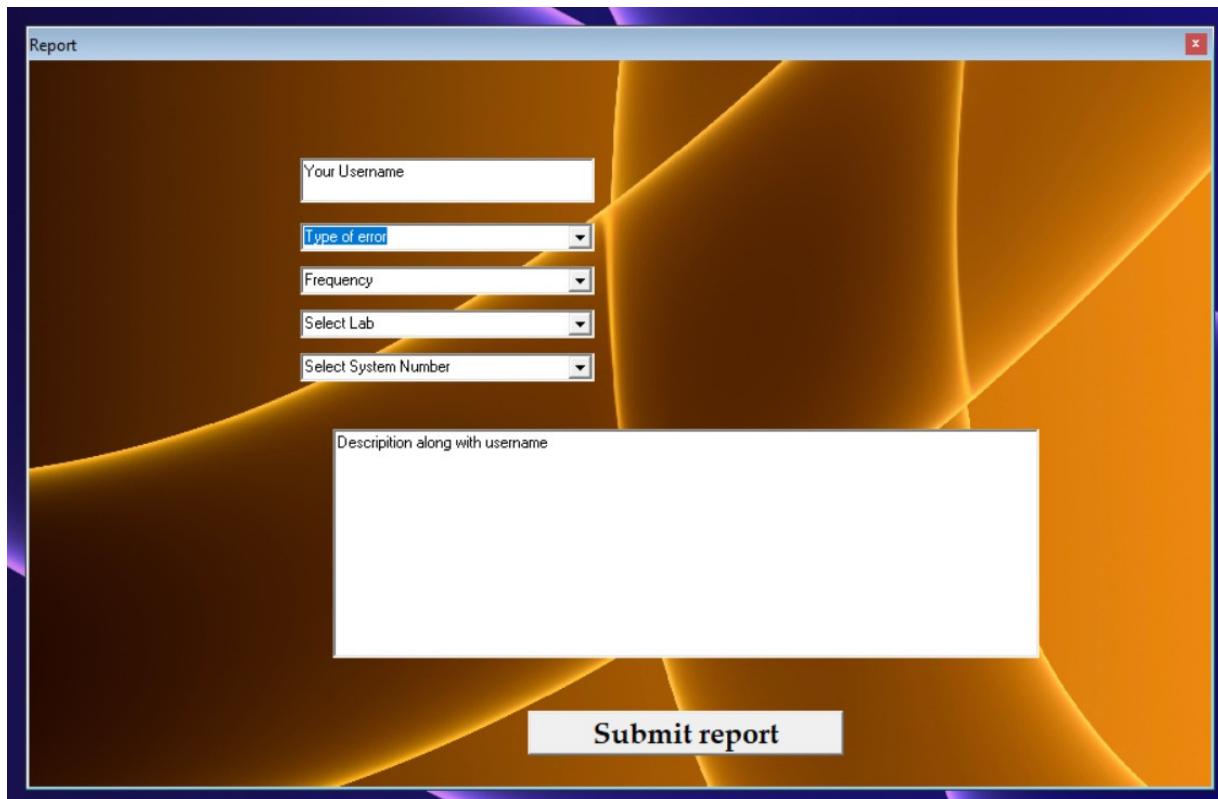
9) Delete Users



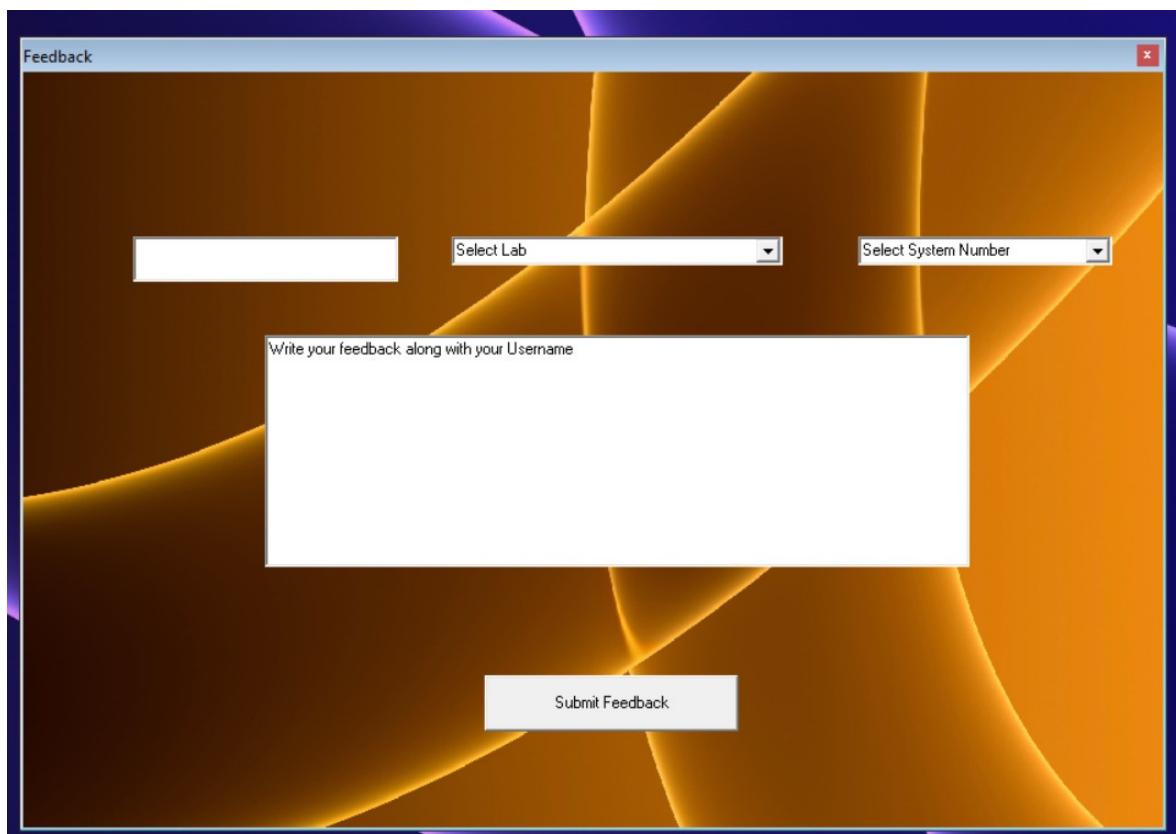
10) Admin Register



11) Error Report



12) Feedback



CODING

Splash screen

```
Private Sub Command1_Click()
frmlogin.Show
End Sub
```

```
Private Sub Form_Load()
Me.Left = (Screen.Width - Me.Width) / 2
Me.Top = (Screen.Height - Me.Height) / 2
End Sub
```

Connection Code:

(Connection String)

```
Dim con As New ADODB.Connection
Dim rs As New ADODB.Recordset
```

(Establishing the connection)

```
Private Sub Form_Load()
con.Open _
"Provider=sqloledb;" & _
"Data Source=LAPTOP-37DEMHK6\SQLEXPRESS;" & _
"Initial Catalog=stud_db;" & _
"Trusted_Connection=yes;" 'CONNECTION IS DONE/OPEN
End Sub
```

(Closing the Connection)

```
Private Sub Form_Unload(Cancel As Integer)
con.Close
End Sub
```

Login

```
Private Sub CmdSignup_Click()
FrmRegister.Show
End Sub
```

```
Private Sub Command1_Click()
Form1.Show
End Sub
```

```
Private Sub Command2_Click()
deleteuser.Show
End Sub
```

```
Private Sub Command3_Click()
FrmRegister.Show
End Sub
```

```
Private Sub CmdClear_Click()
TxtUserId.Text = ""
TxtPassword.Text = ""
End Sub
```

```
Private Sub Check1_Click()
If Check1.Value = 1 Then
    TxtPassword.PasswordChar = ""
ElseIf Check1.Value = 0 Then
    TxtPassword.PasswordChar = "*"
End If
End Sub
```

```
Private Sub CmdSignin_Click()
rs.Open "select stud_pass from stud_db.dbo.stud_table where stud_name = "" &
TxtUserId.Text & "", con, adOpenKeyset
```

```
If rs.RecordCount = 0 Then
    MsgBox "user not found"
Else

    If rs(0) = TxtPassword.Text Then
        MsgBox "Password Verified"
        con.Close
        FrmLogged.Show
        Unload Me
```

```
Else
MsgBox "Wrong Password"
End If
End If
End Sub
```

Logged

```
Dim sec, min, hrs As Integer
```

```
Private Sub Command1_Click()
frmlogin.Show
End Sub
```

```
Private Sub Command2_Click()
MsgBox "Please Continue With Admin Login"
Admin.Show
End Sub
```

```
Private Sub Form_Load()
Me.Left = (Screen.Width - Me.Width) / 2
Me.Top = (Screen.Height - Me.Height) / 2

Timer1.Interval = 1000
Timer2.Interval = 100

End Sub
```

```
Private Sub Timer1_Timer()
sec = sec + 1
Label4.Caption = Format(sec, "00")
If sec = 60 Then
min = min + 1
sec = 0
Label3.Caption = Format(min, "00")
End If
```

```
If min = 60 Then  
    hrs = hrs + 1  
    min = 0  
    Label2.Caption = Format(hrs, "00")  
End If  
End Sub
```

```
Private Sub Timer2_Timer()  
    Label1 = Time  
  
End Sub
```

Register

```
Dim con As New ADODB.Connection
Dim rec As New ADODB.Recordset

Private Sub Command1_Click()
Admin.Show
End Sub

Private Sub reg_btn_Click()

If TbUsername.Text = "" And TbPass.Text = "" Then
    MsgBox "Fields Can't Be Empty"
Else
    Dim sql As String

    sql = "insert into stud_db.dbo.stud_table (stud_name, stud_pass) values (" & _
    sql = sql & """ & TbUsername.Text & ""","
    sql = sql & """ & TbPass.Text & "")"
    con.Execute sql

    MsgBox "Registered"
    TbUsername.Text = ""
    TbPass.Text = ""
End If

End Sub

'Establishing the DB connection
Private Sub Form_Load()

Me.Left = (Screen.Width - Me.Width) / 2
Me.Top = (Screen.Height - Me.Height) / 2

con.Open _
"Provider=sqloledb;" & _
"Data Source=LAPTOP-37DEMHK6\SQLEXPRESS;" & _
"Initial Catalog=stud_db;" & _
"Trusted_Connection=yes;" "CONNECTION IS DONE/OPEN
End Sub

Private Sub Form_Unload(Cancel As Integer)
con.Close
End Sub
```

Admin Login

```
Dim con As New ADODB.Connection  
Dim rs As New ADODB.Recordset
```

```
Private Sub Check1_Click()  
If Check1.Value = 1 Then  
    Text2.PasswordChar = ""  
ElseIf Check1.Value = 0 Then  
    Text2.PasswordChar = "*"  
End If  
End Sub
```

```
Private Sub Command1_Click()
```

```
rs.Open "select username from stud_db.dbo.admin_t where username = '" & Text1.Text & "' ",  
con, adOpenKeyset
```

```
If rs.RecordCount = 0 Then  
    MsgBox "Admin Not Exist"  
Else  
    If rs(0) = Text2.Text Then  
        MsgBox "Admin Password Verified"  
        con.Close  
        Admin.Show  
        Unload Me  
    Else  
        MsgBox "Wrong Password"  
  
    End If  
End If  
  
End Sub
```

```
Private Sub Command2_Click()  
Dim pin As Integer  
pin = InputBox("Please Enter Admin Pin To Proceed", Testting, 0)  
If pin = 705 Then  
    If Text1.Text = "" And Text2.Text = "" Then  
        MsgBox "Fields Can't Be Empty"  
    Else  
        Dim sql As String
```

```
sql = "insert into stud_db.dbo.admin_t (username, password) values ("  
    sql = sql & """ & Text1.Text & """,  
    sql = sql & """ & Text2.Text & "")"  
con.Execute sql
```

```
MsgBox "New Admin Registered"  
Text1.Text = ""  
Text2.Text = ""  
End If
```

```
Else  
    MsgBox "Wrong Admin Pin"  
End If
```

```
End Sub
```

```
Private Sub Form_Load()  
Me.Left = (Screen.Width - Me.Width) / 2  
Me.Top = (Screen.Height - Me.Height) / 2
```

```
con.Open _  
"Provider=sqloledb;" & _  
"Data Source=LAPTOP-37DEMHK6\SQLEXPRESS;" & _  
"Initial Catalog=stud_db;" & _  
"Trusted_Connection=yes;" 'CONNECTION IS DONE/OPEN  
End Sub
```

```
Private Sub Form_Unload(Cancel As Integer)  
'con.Close  
End Sub
```

Admin

```
Dim con As New ADODB.Connection  
Dim rec As New ADODB.Recordset  
  
Private Sub Command1_Click()  
rec.Open "select * from [stud_db].[dbo].[stud_table]", con, adOpenKeyset  
  
For i = 1 To rec.RecordCount  
stud_table = stud_table & "User Name : " & rec.Fields("stud_name").Value & vbCrLf  
stud_table = stud_table & "User Password : " & rec.Fields("stud_pass").Value & vbCrLf  
  
rec.MoveNext  
Next i  
  
Label1.Caption = stud_table  
  
rec.Close  
  
End Sub  
  
Private Sub Command2_Click()  
deleteuser.Show  
End Sub  
  
Private Sub Command3_Click()  
FrmRegister.Show  
End Sub  
  
Private Sub Form_Load()  
  
Me.Left = (Screen.Width - Me.Width) / 2  
Me.Top = (Screen.Height - Me.Height) / 2  
  
con.Open _  
"Provider=sqloledb;" & _  
"Data Source=LAPTOP-37DEMHK6\SQLEXPRESS;" & _  
"Initial Catalog=stud_db;" & _  
"Trusted_Connection=yes;"  
End Sub  
  
Private Sub Form_Unload(Cancel As Integer)  
con.Close  
End Sub
```

Delete User

```
Dim con As New ADODB.Connection
Dim rec As New ADODB.Recordset

Private Sub Command1_Click()
rec.Open "select * from stud_db.dbo.stud_table where stud_name= '" & Text1.Text & "'", con,
adOpenStatic, adLockReadOnly
sql = "delete from stud_db.dbo.stud_table where stud_name= '" & Text1.Text & "'"

If rec.EOF() Then
MsgBox "No Such Record Not Found"
rec.Close
con.Close
Else

con.Execute sql
MsgBox "Record Deleted"
rec.Close
con.Close
End If

End Sub

Private Sub Form_Load()

Me.Left = (Screen.Width - Me.Width) / 2
Me.Top = (Screen.Height - Me.Height) / 2
'Establishing the DB connection
con.Open _
"Provider=sqloledb;" & _
"Data Source=LAPTOP-37DEMHK6\SQLEXPRESS;" & _
"Initial Catalog=stud_db;" & _
"Trusted_Connection=yes;" 'CONNECTION IS DONE/OPEN

rec.Open "select * from [stud_db].[dbo].[stud_table]", con, adOpenKeyset

For i = 1 To rec.RecordCount
stud_table = stud_table & "User Name : " & rec.Fields("stud_name").Value & vbCrLf
stud_table = stud_table & "User Password : " & rec.Fields("stud_pass").Value & vbCrLf

```

```
rec.MoveNext
```

```
Next i
```

```
Label1.Caption = stud_table
```

```
rec.Close
```

```
End Sub
```

TESTING

TESTING

Introduction to Testing

Software testing is an investigation conducted to provide stakeholders with information about the quality of the software product or service under test.[\[1\]](#) Software testing can also provide an objective, independent view of the software to allow the business to appreciate and understand the risks of software implementation. Test techniques include the process of executing a program or application with the intent of finding software bugs (errors or other defects), and verifying that the software product is fit for use. Software testing involves the execution of a software component or system component to evaluate one or more properties of interest.

Types of Testing

The general testing process is the creation of a testing strategy(which sometimes includes the execution of the testing case), creation of a test plan/design(which usually includes test case and test procedure) and the execution of tests. Test are inputs that have been devised to test the system.

1. Black-Box Testing
2. White Box Testing
3. Alpha Testing
4. Beta Testing
5. Clean Room Testing

1. Black-Box Testing:-

Black box Testing is a software testing method in which the internal structure/design/implementation of the item being tested is not known to the tester. These tests can be functional or non-functional. It focuses on the functional requirements of the software. In black-box testing, the cases are derived from the specification of the system, the implementation details of the system are not taken into considerations.

2. White Box Testing:-

White-box testing, sometimes called Glass box testing or structural testing or clear box testing. White box testing is used to test areas that cannot be reached from a black box testing. White-box testing user an internal perspective of the system to design the test cases based on the internal structure.

3. Alpha Testing :-

Alpha testing is simulated for actual operational testing by potential users customers or an independent test team at the developers site. Alpha testing is often employed for off-the-self software as a form of internal acceptance testing before the software goes to beta testing.

4. Beta Testing

Beta testing comes after alpha testing. Version of the software, known as beta versions, are related to a limited audience outside of the programming team. The software is released to groups of people so that further testing can ensure the products has few faults or bugs.

5. Clean Room Testing

Clean room testing makes use of incremental development model. The initial increment delivered to the customer with the critical functionalities and less important features are added in later increments. The initial increment is released for experimentation with an intention to identify the requirements problem. These problems are removed in later increments. New increments are added to the existing ones and the integrated system is validated. As new increments are added, the initial increments are also released.

TEST CASES

Test Case-ID	Check Item	Testing	Test Case Objectives	Steps to Execute	Expected Result
TC-1	Username and password	Output Testing	Leave all Fields as Empty and click Login	Click Login Button	Error Message Displays —you are not a valid admin
TC-2	Upload issues	Data Testing	Incorrect Format	Click save button	Upload the issue

Figure03: Test Cases for Validating Login of User

FUTURE ENHANCEMENT

FUTURE ENHANCEMENT

A digital logbook can be enhanced with features that make it more user-friendly and efficient. For example, a digital logbook could include features that allow users to add notes and comments as they are working on a lab program, and to track their progress over time. Additionally, a digital logbook could include features that allow users to share their projects with others, and to access their projects from any computer or mobile device.

- Fix bugs
- categories users
- Restricted system access
- Restricted time access
- Extra time request
- Enhance UX and UI
- Cloud storage
- System sharing
- QR code support
- Biometric access

CONCLUSION

CONCLUSION

In conclusion, developing a digital logbook for a computer lab is a useful and efficient solution for managing lab activities and resources. A digital logbook can help lab administrators keep track of equipment usage, maintenance schedules, and user activity. It can also provide users with an easy-to-use platform for requesting resources and reporting issues.

By digitizing the logbook, lab administrators can reduce the risk of data loss, improve data accuracy, and streamline communication. Digital logbooks can also provide insights into lab usage patterns and help identify areas for improvement.

To develop a digital logbook for a computer lab, it is important to consider the specific needs and requirements of the lab. A user-friendly interface, robust data management capabilities, and effective communication channels are all important features to include.

Overall, a digital logbook can be a valuable tool for managing a computer lab and improving lab efficiency and productivity.

BIBLIOGRAPHY

BIBLIOGRAPHY

“Visual Basic 6 Black ”

Author: Steven Holzner Publication:

The coriolis Group “Visual
Programming” – Edition: 2016

Author: Srikanth

Publication:- Skyward

Github:-

ankit705yadav

Websites:-

geeksforgeeks.com

stackoverflow.com

vbforum.com

youtube.com

codeproject.com

“<https://msdn.microsoft.com>