Kristu Jayanti College
AUTONOMOUS Bengaluru
Reaccredited A++ Grade by NAAC | Affiliated to Bengaluru North University

LOST AND FOUND

Project Report submitted in partial fulfillment of the requirements for the award of the degree of

BACHELOR OF COMPUTER APPLICATIONS (BCA)

Submitted By

SAMARPAN DAHAL

21BCAA54

Under the guidance of

Prof. Mary Jacob

DEPARTMENT OF COMPUTER SCIENCE (UG)
BCA PROGRAME
KRISTU JAYANTI COLLEGE (Autonomous)
K. Narayanapura, Kothanur P.O., Bangalore – 560077

Kristu Jayanti College
AUTONOMOUS Bengaluru
Reaccredited A++ Grade by NAAC | Affiliated to Bengaluru North University

DEPARTMENT OF COMPUTER SCIENCE (UG)

CERTIFICATE OF COMPLETION

This is to certify that the project entitled **"LOST AND FOUND"** has been satisfactorily completed by **SAMARPAN DAHAL, 21BCAA54** in partial fulfillment of the award of the Bachelor of Computer Applications degree requirements prescribed by Kristu Jayanti College (Autonomous) Bengaluru (Affiliated to Bangalore North University) during the academic year 2022 -23.

Internal Guide ***Head of the Department***

Valued by Examiners

1:_____ ***Centre:*** **Kristu Jayanti College**

2:_____ ***Date:***

Kristu Jayanti College
AUTONOMOUS Bengaluru
Reaccredited A++ Grade by NAAC | Affiliated to Bengaluru North University

DECLARATION

I, SAMARPAN DAHAL, 21BCAA54 hereby declare that the project work entitled 'LOST AND FOUND' is an original project work carried out by me, underthe guidance of PROF.MARY JACOB

This project work has not been submitted earlier either to any University/Institution or any other body for the fulfillment of the requirement of a course of study.

Signature
(Samarpan)

Bengaluru
Date:

ACKNOWLEDGEMENT

The success of the project depends upon the efforts invested. It's my duty to acknowledge and thank the individuals who has contributed to the successful completion of the project.

I take this opportunity to express my profound and wholehearted thanks to **FR. DR. AUGUSTINE GEORGE, PRINCIPAL, KRISTU JAYANTI COLLEGE**, **BANGALORE**
for providing ample facilities made to undergo my project successfully.

I express my deep sense of gratitude and sincere thanks to our Head of the Department **PROF. SEVUGA PANDIAN** for his valuable advice.

I feel immense pleasure to thank my respected guide **PROF. MARY JACOB** for sustaining Interest and providing dynamic guidance in aiding me to complete this project immaculately and impeccably and for being the source of my strength and confidence.

It is my duty to express my thanks to all Teaching and Non-Teaching Staff members of Computer science department who offered me help directly or indirectly by their suggestions. The successful completion of my project would not have been possible without my parent's Sacrifice, guidance, and prayers. I take this opportunity to thank everyone for their continuous Encouragement. I convey my thankfulness to all my friends who were with me to share my happiness and agony.

Last but not the least I thank almighty God for giving me strength and good health throughout my project and enabling me to complete it successfully.

TABLE OF CONTENTS

1. INTRODUCTION

1.1 PROBLEM DEFINITION:

These days many of us lose our valuable items but no such proper platform is available for returning these lost items. So, it is mandatory to develop a system that can help to overcome these problems.. As this system's functionality is described here, whenever a person will lose his item, he should register and after successful login a form will appear for describing his lost item. If someone finds a item he will go to the found section and update the details of the items. The details of found items will be stored in database along with the photo. whoever has lost the items can search in the database if his/her item has been found and updated.

1.2 SCOPE OF THE PROJECT:

- Our goal is to return the owner their lost items.
- Developing a user-friendly software which helps the user to find there lost items.

MODULES IN THE PROJECT:

- **Login**
- **Register**
- **Found section**
- **Lost section**
- **Claimed**
- **Report**

Register:

This module is used to create an account in the project, and we need reg no, user name, email and password to create an account.

Login page:

This module is used to login and we need user’s reg no and password to login into the form.

Found Section:

This module stores the details of items which is found by the user. The details of found items can be stored in database as user fills the details of found items. This section contains unique ID for each items found ,item name, item color, brand name , key description, item condition, place found , and founders details.

Founders details contain (name, regno, email, phone no, class, section, course,)

Lost Section:

This module is used to search the lost items by adding the details and searching in the database.

(L item name , L item colour , L brand name , L date , L key desc , Lost place , L item condition).If item has been found and is present in the database then matching item is shown. User can claim his/her item by giving their details. Once the item is claimed it goes in claimed section and the details gets stored in claimed database.

Claimed Section:

Once the user identifies the item then they can claim item in this section. Claimed items gets stored in claimed database.

Report:

Report can be generated by admins. Report contains all the items found and items claimed for today and this month.

2. SYSTEM STUDY

2.1 EXISTING SYSTEM:

The Existing system is system which helps people or users to find their lost items and it maintain the details which is updated in the system.

Some of the main problems in the existing system is:

- Percentage of accuracy is less.
- If found item name is given wrong, then finding the correct item is difficult.

2.2 FEASIBILITY STUDY:

A feasibility study is an analysis of how successfully a project can be completed, accounting for factors that affect it such as economic, technological, legal and scheduling factors.

Project managers use feasibility studies to determine potential positive and negative outcomes of a project before investing a considerable amount of time and money into it.

A feasibility study tests the viability of an idea, a project or even a new business. The goal of a feasibility study is to place emphasis on potential problems that could occur if a project is pursued and determine if, after all significant factors are considered, the project should be pursued.

The project "**LOST AND FOUND "** can be designed and developed using .Net framework. The content, language is feasible to use and the portal can be developed based on the requirements.

Components of feasible study:

- Technical feasibility
- Operational feasibility
- Schedule Feasibility

Every project is feasible for given unlimited resources and infinitive time. Feasibility study is an evaluation of the proposed system regarding its workability, impact on the organization, ability to meet the user needs and effective use of resources.

Thus, when a new application is proposed it normally goes through a feasibility study before it is approved for development. Feasibility and risk analysis are related in many ways. The feasibility analysis in this project has been discussed below based on the above mentioned components of feasibility.

SCHEDULE:

Time duration of this project requires 53 days covering 5 days for requirement analysis, 8 days for system design, 8 days for I/O Design , 20 days for implementation phase, 6 days for Testing phaseand 6 days for documentation.

Resource persons are SAMARPAN DAHAL and MITESH MALVIYA.V

2.3 PROPOSED SYSTEM:

The proposed system has got the following advantages over the existing system

- It help user to find there lost items.
- It is easy to handle
- It save time.

3.SYSTEM DESIGN

In the design phase the architecture is established. This phase starts with the requirement document delivered by the requirement phase and maps the requirements into an architecture. The architecture defines the components, their interfaces and behaviours. The deliverable design document is the architecture. The design document describes a plan to implement the requirements. This phase represents the ``how'' phase

Details on computer programming languages and environments, machines, packages, application architecture, distributed architecture layering, memory size, platform, algorithms, data structures, global type definitions, interfaces, and many other engineering details are established.

The design may include the usage of existing components. Analysing the trade-offs of necessary complexity allows for many things to remain simple which, in turn, will eventually lead to a higher quality product. The architecture team also converts the typical scenarios into a test plan. In our approach, the team, given a complete requirement document, must also indicate critical priorities for the implementation team.

A critical implementation priority leads to a task that has to be done right . If it fails, the product fails. If it succeeds, the product might succeed. At the very least, the confidence level of the team producing a successful product will increase. This will keep the implementation team focused. Exactly how this information is conveyed is a skill based on experience more than a science based on fundamental foundations . System design is the process of defining the architecture components, modules, interfaces, and data for a system to satisfy specified requirements.

Systems design could be seen as the application of systems theory to product development. There is some overlap with the disciplines of systems analysis , systems architecture and systems engineering.

If the broader topic of product development blends the perspective of marketing, design, and manufacturing into a single approach to product development," then design is the act of taking the marketing information and creating the design of the product to be manufactured. Systems design is therefore the process of defining and developing systems to satisfy specified requirements of the user.

Until the 1990s, systems design had a crucial and respected role in the data processing industry. In the 1990s, standardization of hardware and software resulted in the ability to build modular systems.

The increasing importance of software running on generic platforms has enhanced the discipline of software engineering. Object-oriented analysis and design methods are becoming the most widely used methods for computer systems design. The UML has become the standard language in object-oriented analysis and design. It is widely used for modelling software systems and is increasingly used for high designing non softwaresystems and organizations.

ARCHITECTURAL DESIGN:

The architectural design of a system emphasizes the design of the system architecture that describes the structure, behavior and more views that system and analysis.

LOGICAL DESIGN:

The logical design of a system pertains to an abstract representation of the data flows, inputs and outputs of the system. This is often conducted via modelling, using an over-abstract (and sometimes graphical) model of the actual system. In the context of systems, designs are included. Logical design includes entity-relationship diagrams(ER diagrams).

PHYSICAL DESIGN:

The physical design relates to the actual input and output processes of the system. This is explained in terms of how data is input into a system, how it is verified/authenticated, how it is processed, and howit is displayed. In physical design, the following requirements about the system are decided.

- Input requirement
- Output requirement
- Storage requirement
- System control and backup
- Processing requirements

Put another way, the physical portion of system design can generally be broken down into three sub-tasks:

- User Interface Design
- Data Design.
- Process Design

User Interface Design is concerned with how users add information to the system and with how the system presents information back to them. Data Design is concerned with how the data is represented and stored within the system.

Finally, Process Design is concerned with how data moves through the system, and with how and where it is validated, secured and/or transformed as it flows into, through and out of the system.

At the end of the system design phase, documentation describing the three sub-tasks is produced and made available for use in the next phase. Physical design, in this context, does not refer to the tangible physical design of an information system.

To use an analogy, a personal computer's physical design involves input via a keyboard, processing within the CPU, and output via a monitor, printer, etc.

It would not concern the actual layout of the tangible hardware, which for a PC would be a monitor, CPU, motherboard, hard drive, modems, video/graphics cards, USB slots, etc.

It involves a detailed design of a user and a product database structure processor and a control processor. The H/S personal specification is developed for the proposed system.

3.1E-R DIAGRAM:

An entity relationship diagram (ERD) shows the relationships of entity sets stored in a database. An entity in this context is a component of data. In other words, ER diagrams illustrate the logical structureof databases. At first glance an entity relationship diagram looks very much like a flowchart.

It is the specialized symbols, and the meanings of those symbols, that make it unique. Because this ER tutorialfocuses on beginners .

Below are some tips that will help you build effective ER diagrams:

- Identify all the relevant entities in a given system and determine the relationships among these entities.
- An entity should appear only once in a particular diagram.
- Provide a precise and appropriate name for each entity, attribute, and relationship in the diagram.
- Terms that are simple and familiar always beats vague, technical-sounding words.

- In naming entities, remember to use singular nouns. However, adjectives may be used to distinguish entities belonging to the same class (part-time employee and full -time employee, for example).
- Meanwhile attribute names must be meaningful, unique, system independent, and easily understandable.
- Remove vague, redundant or unnecessary relationships between entities.
- Never connect a relationship to another relationship.
- Make effective use of colours.
- You can use colours to classify similar entities or to highlight key areas in your diagrams.
- You can draw entity relationship diagrams manually, especially when you are just informally showing simple systems to your peers.
- However, for more complex systems and for external audiences, you need diagramming software such as Creately's to craft visually engaging and precise ER diagrams.
- The ER Diagram Software offered by Creately as an online service is pretty easy to use and is a lot more affordable than purchasing licensed software.
- It is also perfectly suited for development teams because of its strong support for collaboration.

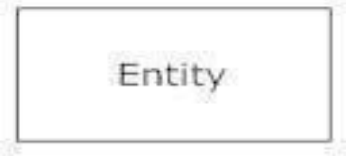
THE HISTORY OF ENTITY RELATIONSHIP DIAGRAMS:

Peter Chen developed ERDs in 1976. Since then Charles Bachman and James Martin have added some slight refinements to the basic ERD principles.

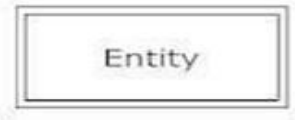
STRUCTURE OF ENTITY RELATIONSHIP DIAGRAM WITH COMMON ERD:

Notations : An entity relationship diagram is a means of visualizing how the information a system produces is related. There are five main components of an ERD:

Entities, which are represented by rectangles. An entity is an object or concept about which you want to store information.



A weak entity is an entity that must defined by a foreign key relationship with another entity as it cannot be uniquely identified by its own attributes alone.

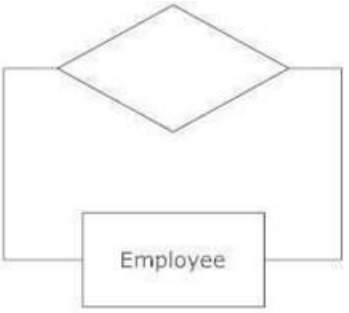


Lost and Found has 9 Entities they are

- User
- Signup
- Found item
- Lost item
- Founder Details
- Admin
- Report
- Lost
- Claimed

Actions, which are represented by diamond shapes, show how two entities share information in thedatabase.

In some cases, entities can be self-linked. For example, employees can supervise other employees.



Lost and Found has 4 Relation

- Manage
- Categories
- Has
- Generates

Sign up entity **manage** User where User has two **Categories** Lost item and Found item. Found item **has** Founder Details. Admin **generates** report of Found and Claimed.

Attributes, which are represented by ovals. A key attribute is the unique, distinguishing characteristic of the entity. For example, an employee's social security number might be the employee's key attribute.

A multivalued attribute can have more than one value. For example, an employee entity can have multiple skill values.

A derived attribute is based on another attribute. For example, an employee's monthly salary is based on the employee's annual salary.

Attributes in Lost And Found : Lost and Found has many attributes for each entity.

- **User**
  1. Regno
  2. Password

- **Signup**
  1. ID
  2. Name
  3. Email
  4. Password
  5. Reg no

- **Found item**
  1. FitemName
  2. FitemColour
  3. FBrand
  4. Fkeydesc
  5. PlaceFound
  6. Fitemcondition
  7. Fitemphoto
- **Lost item**
  1. ID
  2. LitemName
  3. LitemColour
  4. L Brand
  5. Lost Date
  6. L Key Desc
  7. Litemcondition
  8. Lost place

- **Founder Details**
    1. Fname
    2. Reg no
    3. Phone no
    4. Email
    5. Semester
    6. Course
    7. Section

- **Admin**
    1. Reg no
    2. Password

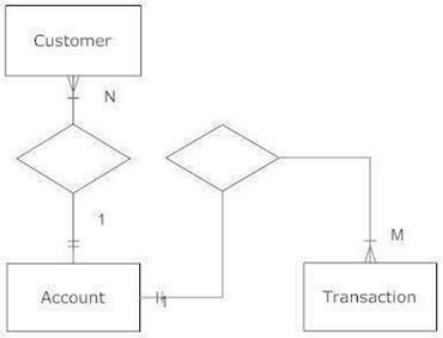
- **Found**
    1. ID
    2. Reg no
    3. itemcolour
    4. Item brand
    5. Item name
    6. Date found
    7. Place found

- **Claimed**
    1. ID
    2. C name
    3. Reg no
    4. email
    5. Phone no
    6. Sem
    7. course
    8. section
    9. Today date

Connecting lines, solid lines that connect attributes to show the relationships of entities in the ER Diagram.

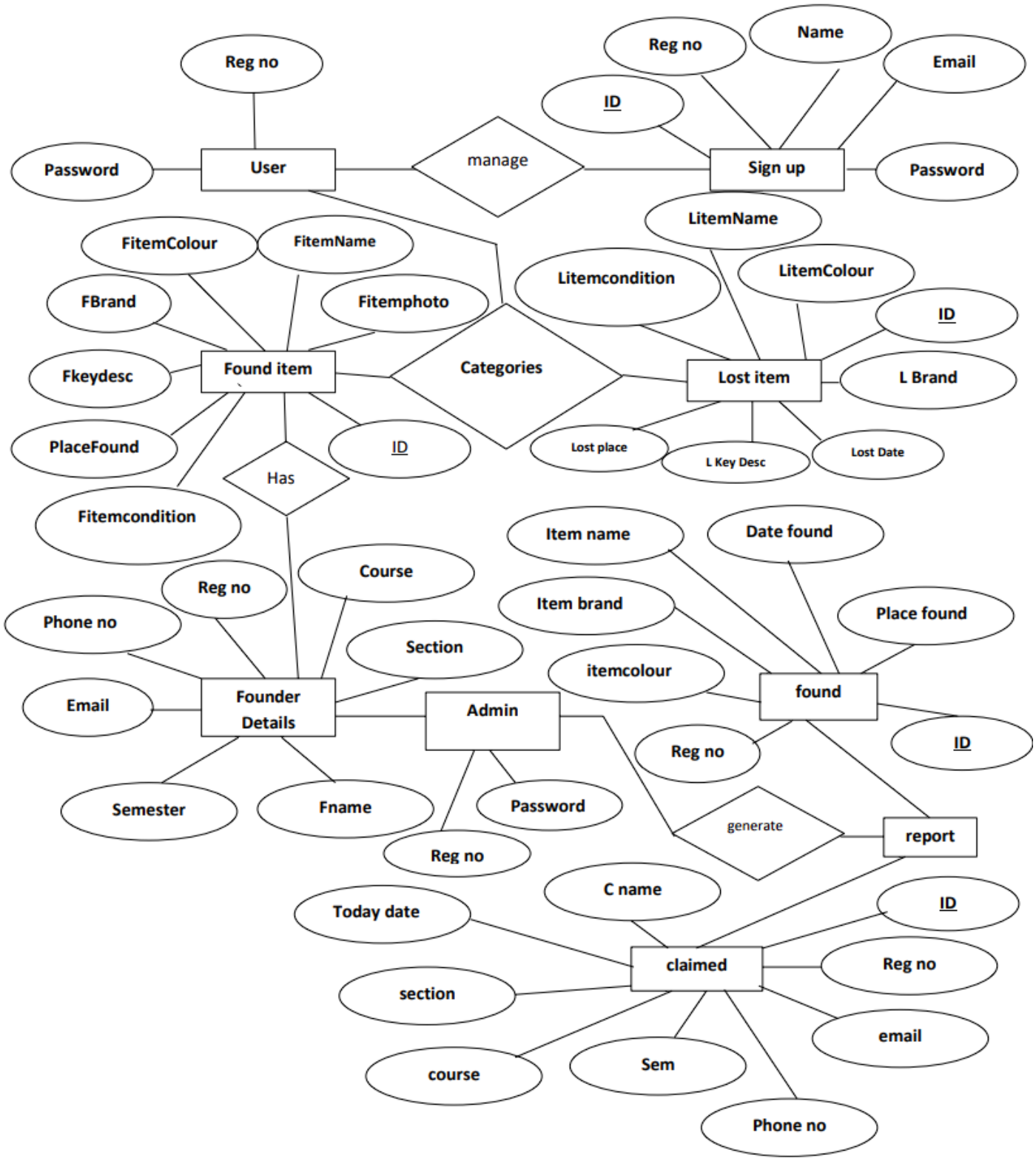
Cardinality specifies how many instances of an entity relate to one instance of another entity. Ordinality is also closely linked to cardinality. While cardinality specifies the occurrences of a relationship, ordinality describes the relationship as either mandatory or optional. In other words, cardinality specifies the maximum number of relationships and ordinality specifies the absolute minimum number of relationships.

There are many notation styles that express cardinality .



Lost And Found ER: There are 9 Entities which has relation between each other. Sign up manage user where Sign up have many attributes that are mentoned above. User has two categories Lost item and Found item . Found item has Founder details which stores the details of founder. Admin generates the report of found and claimed .

LOST AND FOUND ER DIAGRAM



3.2 DATA FLOW DIAGRAM (LEVEL 0 AND LEVEL 1):

The Data Flow Diagrams (DFDs) are used for structure analysis and design. DFDs show the flow of data from external entities into the system. DFDs also show how the data moves and is transformed from one process to another, as well as its logical storage. The following symbols are used within DFDs. For clarity, a key has been provided at the bottom of this page.

A data flow diagram (DFD) is a graphical representation of the "flow" of data through an information system, modelling its process aspects. A DFD is often used as a preliminary step to create an overview of the system, which can later be elaborated. DFDs can also be used for the visualization of data processing (structured design).

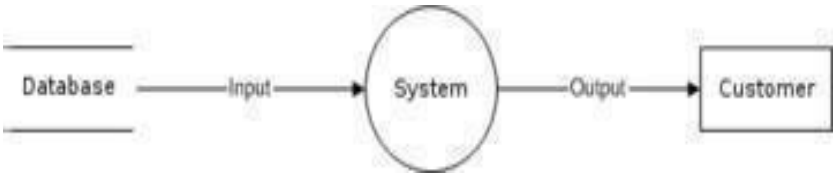
A DFD shows what kind of information will be input to and output from the system, where the data will come from and go to, and where the data will be stored. It does not show information about the timing of process or information about whether processes will operate in sequence or in parallel (which is shown on a flowchart).

HISTORY:

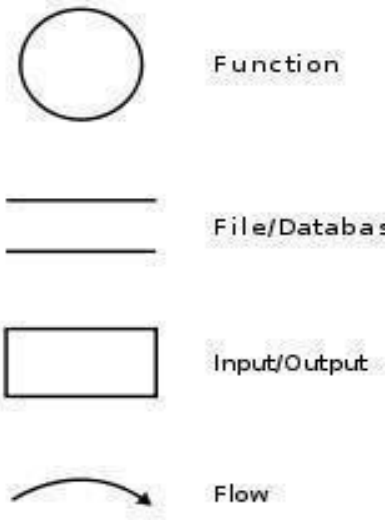
Larry Constantine, the original developer of structured design, based on Martin and Estrin's "Data Flow Graph" model of computation.

Starting in the 1970s, data flow diagrams (DFD) became a popular way to visualize the major steps and data involved in software system processes. DFDs were usually used to show data flow in a computer system, although they could in theory be applied to business process modelling. DFD were useful to document the major data flows or to explore a new high-level design in terms of data flow.

THEORY:



DATA FLOW DIAGRAM EXAMPLE:



The Data Flow Diagrams (DFDs) are used for structure analysis and design. DFDs show the flow of data from external entities into the system. DFDs also show how the data moves and is transformed from one process to another, as well as its logical storage. The following symbols are used within DFDs. For clarity, a key has been provided at the bottom of this page.

A data flow diagram (DFD) is a graphical representation of the "flow" of data through an information system, modelling its process aspects. A DFD is often used as a preliminary step to create an overview of the system, which can later be elaborated. DFDs can also be used for the visualization of data processing (structured design).

A DFD shows what kind of information will be input to and output from the system, where the data will come from and go to, and where the data will be stored. It does not show information about the timing of process or information about whether processes will operate in sequence or in parallel (which is shown on a flowchart).

Data flow diagrams are one of the three essential perspectives of the structured-systems analysis and design method. The sponsor of a project and the end users will need to be briefed and consulted throughout all stages of a system's evolution. With a data flow diagram, users are able to visualize how the system will operate, what the system will accomplish, and how the system will be implemented.

The old system's dataflow diagrams can be drawn up and compared with the new system's data flow diagrams to draw comparisons to implement a more efficient system. Data flow diagrams can beusedto provide the end user with a physical idea of where the data they input ultimately has an effect upon the structure of the whole system from order to dispatch to report. How any system is developed can be determined through a data flow diagram model.

In the course of developing a set of levelled data flow diagrams the analyst/designer is forced to address how the system may be decomposed into component sub-systems, and to identify the transaction datain the data model.

Data flow diagrams can be used in both Analysis and Design phase of the SDLC.

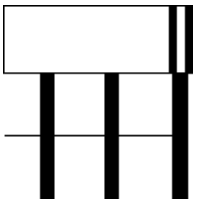
There are different notations to draw data flow diagrams (Yourdon & Coad and Gane & Sarson), defining different visual representations for processes, data stores, data flow, and external entities.

PHYSICAL VS LOGICAL DFD:

A logical DFD captures the data flows that are necessary for a system to operate. It describes the processes that are undertaken, the data required and produced by each process, and the stores needed to hold the data. On the other hand, a physical DFD shows how the system is actually implemented, either at the moment (Current Physical DFD), or how the designer intends it to be in the future (Required Physical DFD).

Thus, a Physical DFD may be used to describe the set of data items that appear on each piece of paper that move around an office, and the fact that a particular set of pieces of paper are stored together in a filing cabinet. It is quite possible that a Physical DFD will include references to data that are duplicated, or redundant, and that the data stores, if implemented as a set of database tables, would constitute an un-normalised (or denormalised) relational database. In contrast, a Logical DFD attempts to capture the data flow aspects of a system in a form that has neither redundancy nor duplication

DATA FLOW SYMBOLS AND THERE MEANINGS:

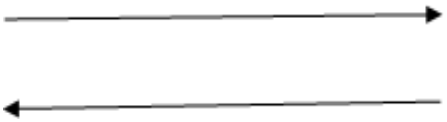


An entity. A source of data or a destination for data.

Source/Sink: Represented by rectangles in the diagram. Sources and Sinks are external entities which are sources or destinations of data, respectively.

Process: Represented by circles in the diagram. Processes are responsible for manipulating the data. They take data as input and output an altered version of the data.

Data Store: Represented by a segmented rectangle with an open end on the right. Data Stores are both electronic and physical locations of data. Examples include databases, directories, files, and even filing cabinets and stacks of paper.



In our course, we need to understand and be able to draw 2 types of Data Flow Diagrams, they are Level-0 and Level 1 DFD's. In this blog, I will hopefully make it easier to understand the differences between the two types of DFD's and help understand how to draw a DFD for the exam.

Firstly we will look at level-0 DFD's and give an example. Then we will look at Level 1 DFD's and give an example.

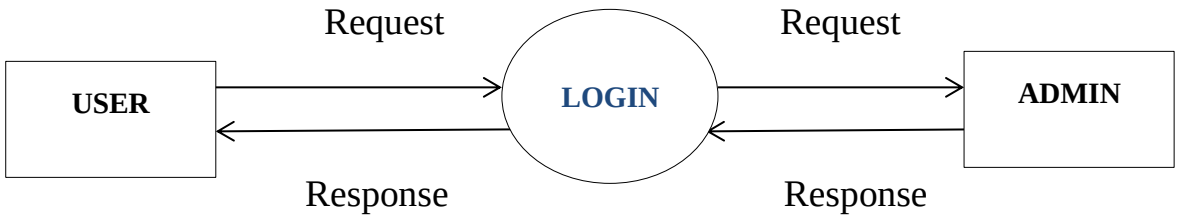
A level-0 DFD is the most basic form of DFD. It aims to show how the entire system works at a glance. There is only one process in the system and all the data flows either into or out of this process.Level- 0 DFD's demonstrates the interactions between the process and external entities. They do not contain Data Stores.

When drawing Level-0 DFD's, we must first identify the process, all the external entities and all the data flows. We must also state any assumptions we make about the system. It is advised that we draw the process in the middle of the page. We then draw our external entities in the corners and finally connect our entities to our process with the data flows.

Lost and Found Data Flow

This data flow diagram is about how the data is flowing in the project lost and found At first the user have to create a account by going to the register section which will store the user information in the data base. Then the user can login into the project and it helps user to find there lost items or whenever they find item they can update it in the found section.
To update the item which user finds, the user have to go to the found section and give a brief description of the item which the user had found with his personal information that will updated in the date base.
If the user has lost any items he can go the lost section and give a brief description of the item and search for his lost item, if the item is updated in the data base.
Below is an example of a Level-0 DFD.

LEVEL ZERO DIAGRAM FOR LOST AND FOUND:



LEVEL 1 DFD's:

Level 1 DFD's aim to give an overview of the full system. They look at the system in more detail. Major processes are broken down into subprocesses. Level 1 DFD's also identifies data stores that are used by the major processes. When constructing a Level 1 DFD, we must start by examining the Context Level DFD.

We must break up the single process into its subprocesses.

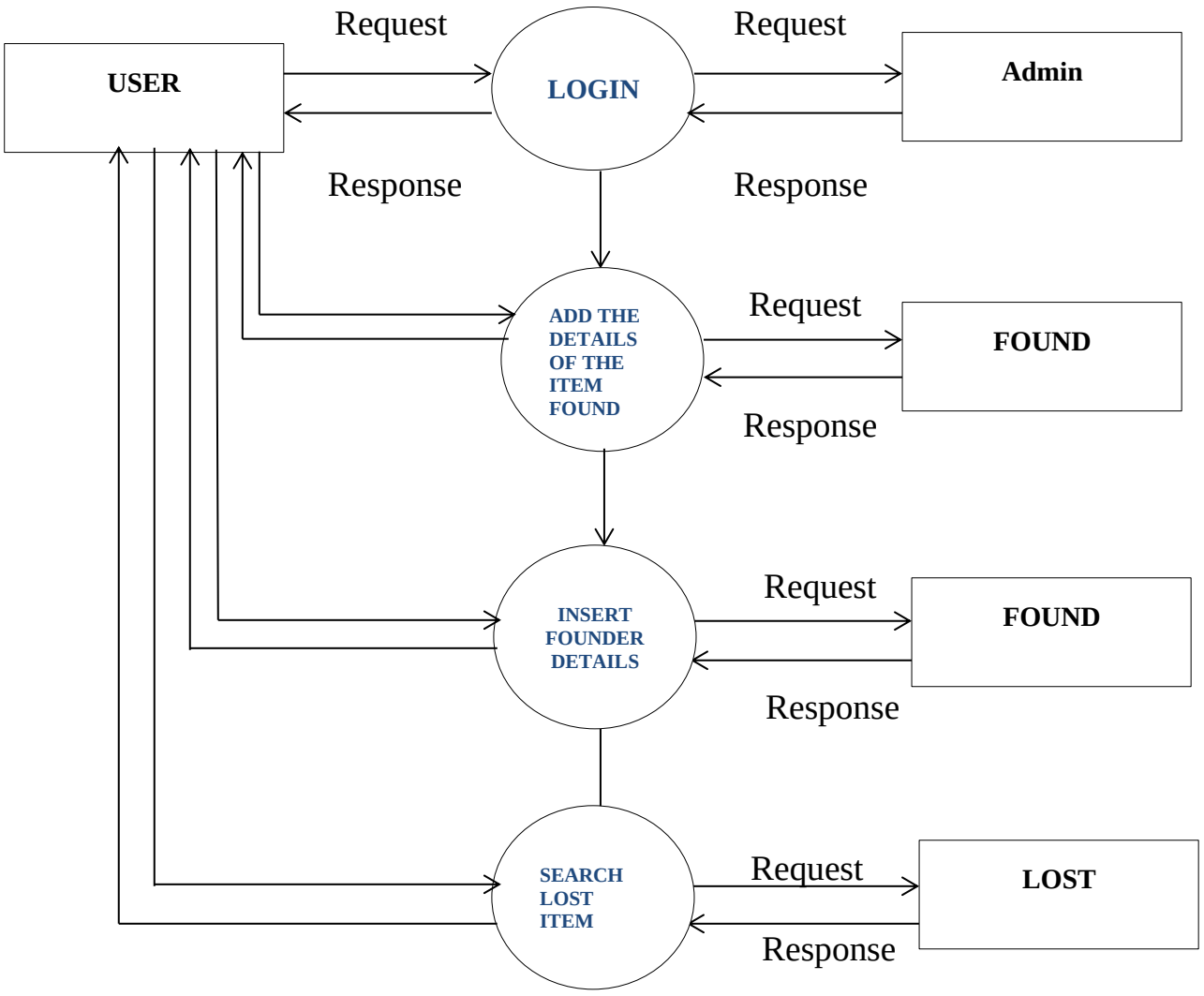
We must then pick out the data stores from the text we are given and include them in our DFD.

Like the Context Level DFD's, all entities, data stores and processes must be labelled.

We must also state any assumptions made from the text.

Below is the Level 1 DFD of LOST AND FOUND:

LEVEL ONE DIAGRAM FOR LOST AND FOUND:



3.3 GANTT CHART:

A Gantt chart is a type of bar chart, devised by Henry Gantt in the 1910s, that illustrates a project schedule. Gantt charts illustrate the start and finish dates of the terminal elements and summary elements of a project. Terminal elements and summary elements comprise the work breakdown structure of the project.

Modern Gantt charts also show the dependency (i.e., precedence network) relationships between activities. Gantt charts can be used to show current schedule status using percent-complete shadings and a vertical "TODAY" line as shown here.

Although now regarded as a common charting technique, Gantt charts were considered revolutionary when first introduced. This chart is also used in information technology to represent data that has been collected.

HISTORICAL DEVELOPMENT:

The first known tool of this type was developed in 1896 by Karol Adamiecki, who called it a harmonogram , Adamiecki did not publish his chart until 1931, however, and only in Polish, which limited both its adoption and recognition of his authorship.

The chart is named after Henry Gantt (1861– 1919), who designed his chart around the years 1910–1915.

One of the first major applications of Gantt charts was by the United States during World War I, at the instigation of General William Crozier

In the 1980s, personal computers allowed widespread creation of complex and elaborate Gantt charts. The first desktop applications were intended mainly for project managers and project schedulers.

With the advent of the Internet and increased collaboration over networks at the end of the 1990s, Gantt charts became a common feature of web-based applications, including collaborative groupware.

GANTT CHART BENEFITS:

CLARITY: One of the biggest benefits of a Gantt chart is the tool's ability to boil down multiple tasks and timelines into a single document. Stakeholders throughout an organization can easily understand where teams are in a process while grasping the ways in which independent elements come together toward project completion.

COMMUNICATION: Teams can use Gantt charts to replace meetings and enhance other status updates. Simply clarifying chart positions offers an easy, visual method to help team members understand task progress.

MOTIVATION: Some teams or team members become more effective when faced with a form of external motivation. Gantt charts offer teams the ability to focus work at the front of a task timeline, or at the tail end of a chart segment. Both types of team members can find Gantt charts meaningful as they plug their own work habits into the overall project schedule.

CO ORDINATION: For project managers and resource schedulers, the benefits of a Gantt chart include the ability to sequence events and reduce the potential for overburdening team members. Some project managers even use combinations of charts to break down projects into more manageable sets of tasks.

CREATIVITY: Sometimes, a lack of time or resources forces project managers and teams to find creative solutions. Seeing how individual tasks intertwine on Gantt charts often encourages new partnerships and collaborations that might not have evolved under traditional task assignment systems.

TIME MANAGEMENT: Most managers regard scheduling as one of the major benefits of Gantt charts in a creative environment. Helping teams understand the overall impact of project delays can foster stronger collaboration while encouraging better task organization.

FLEXIBILITY: Whether you use Excel to generate Gantt charts or you load tasks into a more precise chart generator, the ability to issue new charts as your project evolves lets you react to unexpected changes in project scope or timeline. While revising your project schedule too frequently can eliminate some of the other benefits of Gantt charts, offering a realistic view of a project can help team members recover from setbacks or adjust to other changes.

MANAGEABILITY: For project managers handling complex assignments, like software publishing or event planning, the benefits of Gantt charts include externalizing assignments. By visualizing all of the pieces of a project puzzle, managers can make more focused, effective decisions about resources and timetables.

EFFICIENCY: Another one of the benefits of Gantt charts is the ability for teams members to leverage each other's deadlines for maximum efficiency. For instance, while one team member waits on the outcome of three other tasks before starting a crucial piece of the assignment, he or she can perform other project tasks. Visualizing resource usage during projects allows managers to make better use of people, places, and things.

ACCOUNTABILITY: When project teams face major organizational change, documenting effort and outcomes becomes crucial to career success. Using Gantt charts during critical projects allows both project managers and participants to track team progress, highlighting both big wins and major failures. During professional review periods, team members who frequently exceed expectations can leverage this documentation into larger raises or bonuses.

GANTT CHART IMPORTANCE:

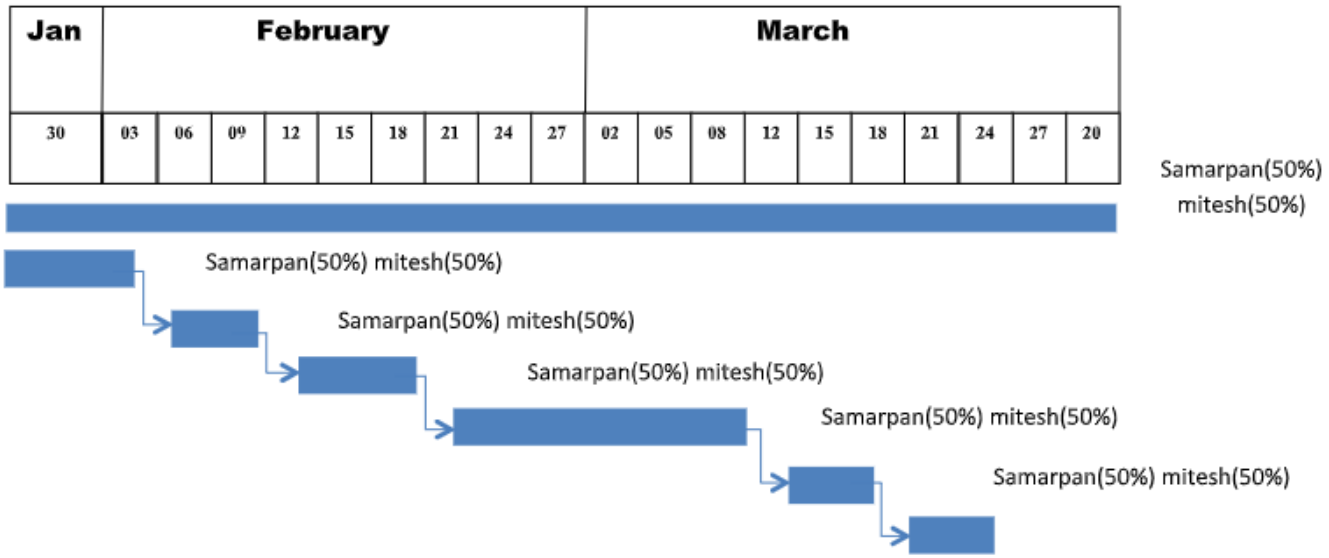
The project's summary and terminal elements, which combine to form the project's internal structure, are shown on the Gantt chart. Many charts will also depict the precedence rankings and dependencies of various tasks within the project. The charts can illustrate the start and finish project terminal elements in project management.

It can also show summary elements and terminal dependencies. The smallest task tracked as part of the project effort is known as a terminal element. Gantt chart represents the tasks in most modern project scheduling packages. However other management applications use simpler communication tools such as message boards, to-do lists and simple scheduling etc., therefore, they do not use Gantt charts as heavily.

The way to create this chart begins by determining and listing the necessary activities. Next, sketch out how you expect the chart to look. List which items depend on others and what activities take place when. For each activity, list how many man-hours it will require, and who is responsible.

Lastly, determine the throughput time. This technique's primary advantage is its good graphical overview that is easy to understand for nearly all project participants and stakeholders. Its primary disadvantage is its limited applicability for many projects, since projects are often more complex than can be effectively communicated with this chart.

Gantt chart for LOST AND FOUND



Sl no	Name	Duration	Start	Finish	Resource
1	Lost and Found	53 days	30/01/2023	23/03/2023	Samarpan Mitesh
2	Requirement Analysis	5 days	30/01/2023	03/02/2023	Samarpan Mitesh
3	System design	8 days	04/02/2023	11/02/2023	Samarpan Mitesh
4	I/O design	8 days	12/02/2023	19/02/2023	Samarpan Mitesh
5	Coding	20 days	20/02/2023	11/03/2023	Samarpan Mitesh
6	Testing	6 days	12/03/2023	17/03/2023	Samarpan Mitesh
7	Documentation	6 days	18/03/2023	23/03/2023	Samarpan Mitesh

3.3 INPUT/OUTPUT DESIGN:

```
Imports System.Text.RegularExpressions
Imports System.Data.SqlClient
Public Class Form1
```

REGISTER BUTTON

```
    Private Sub Buttonregister_Click(sender As Object, e As EventArgs) Handles Buttonregister.Click
```

'// declared for email validation //

```
        Dim regex As New Regex("^[A-Za-z0-9]+(.|_)+[A-Za-z0-9]+@+kristujayanti.com$")
        Dim allowedemail As Boolean = regex.IsMatch(Email.Text.Trim)


        Dim con As New SqlConnection
        Dim cmd As New SqlCommand
        Dim dr As SqlDataReader
        con.ConnectionString = "Data
Source=(LocalDB)\MSSQLLocalDB;AttachDbFilename=C:\Users\samar\source\repos\lostandfound\lostan
dfound\registration.mdf;Integrated Security=True"
        con.Open()
        cmd.Connection = con
        cmd.CommandType = CommandType.Text
        cmd.CommandText = "select * from Register where Regno='" & Regno.Text & "'"
        dr = cmd.ExecuteReader
        If dr.HasRows Then
            MsgBox("Regno already registered", MsgBoxStyle.Critical)
            Regno.Clear()
            con.Close()
        Else
            con.Close()
            con.Open()
            cmd = New SqlCommand("Insert Into Register values('" & Fname.Text & "','" & Regno.Text & "','"
& Email.Text & "','" & Password.Text & "')", con)



            If (Fname.Text = "" And Regno.Text = "" And Email.Text = "" And Password.Text = "") Then
                MessageBox.Show("please enter the details")
            ElseIf (Fname.Text = "" Or Regno.Text = "" Or Email.Text = "" Or Password.Text = "") Then
                MessageBox.Show("Fill all the details")

            ElseIf Not allowedemail Then
                MessageBox.Show("Please Enter College Email.")
                Email.Clear()

            Else
                cmd.ExecuteNonQuery()
```

```
            MsgBox("successfully registered.", MsgBoxStyle.Information, "Success")
            GunaPanel3.Hide()
            GunaPanel4.Show()

            Fname.Clear()
            Regno.Clear()
            Email.Clear()
            Password.Clear()
            ErrorProvider1.Clear()

        End If
        con.Close()
    End If
    con.Close()
End Sub
```

Name validation:

```
Private Sub Fname_keyPress(sender As Object, e As KeyPressEventArgs) Handles Fname.KeyPress
    If Not (Asc(e.KeyChar) = 8) Then
        Dim allowedname As String = "abcdefghijklmnopqrstuvwxyz "
        If Not allowedname.Contains(e.KeyChar.ToString.ToLower) Then
            ErrorProvider1.SetError(Fname, "Enter Alphabets")
            e.KeyChar = ChrW(0)
            e.Handled = True
        Else
            ErrorProvider1.SetError(Fname, "")
        End If
    End If


End Sub
```

Reg no validation:

```
Private Sub Regno_keyPress(sender As Object, e As KeyPressEventArgs) Handles Regno.KeyPress
    If Not (Asc(e.KeyChar) = 8) Then
        Dim allowedregno As String = "abcdefghijklmnopqrstuvwxyz1234567890"

If Not allowedregno.Contains(e.KeyChar.ToString.ToLower) Then
            ErrorProvider1.SetError(Regno, "Enter Correct Regno")



e.KeyChar = ChrW(0)
```

```
e.Handled = True

        Else
            ErrorProvider1.SetError(Regno, "")
        End If
    End If
End Sub
```

Login page:

```
Private Sub Buttonlogin_Click_1(sender As Object, e As EventArgs) Handles Buttonlogin.Click
    Dim con As New SqlConnection
    Dim cmd As New SqlCommand
    con.ConnectionString = "Data
Source=(LocalDB)\MSSQLLocalDB;AttachDbFilename=C:\Users\Minesh\Downloads\lostandfound\lostan
dfound\registration.mdf;Integrated Security=True"
    Dim objcon As SqlConnection = Nothing
    Dim objcmd As SqlCommand = Nothing
    objcon = New SqlConnection("Data
Source=(LocalDB)\MSSQLLocalDB;AttachDbFilename=C:\Users\Minesh\Downloads\lostandfound\lostan
dfound\registration.mdf;Integrated Security=True")
    objcon.Open()
    Dim stmt As String = " select * from Register where Regno='" & Lregno.Text & "' and Password='" &
Lpassword.Text & "' "
    objcmd = New SqlCommand(stmt, objcon)
    Dim reader As SqlDataReader = objcmd.ExecuteReader
    If reader.Read Then


        Me.Hide()
        Form2.Show()
        Lregno.Clear()
        Lpassword.Clear()

    Else
        MessageBox.Show("Invalid login ")
        Lregno.Clear()
        Lpassword.Clear()
    End If
End Sub


Private Sub LinkLabel1_LinkClicked(sender As Object, e As LinkLabelLinkClickedEventArgs) Handles
LinkLabel1.LinkClicked
    GunaPanel3.Show()


 GunaPanel4.Hide()
```

```
End Sub

    Private Sub LinkLabel2_LinkClicked(sender As Object, e As LinkLabelLinkClickedEventArgs) Handles
LinkLabel2.LinkClicked
        GunaPanel3.Hide()
        GunaPanel4.Show()

    End Sub
```

Form 2:

```
Imports System.Data.SqlClient
Imports System.IO
Imports System.Text


Public Class Form2
    Public user As String


    Private Sub Form2Closing(sender As Object, e As FormClosingEventArgs) Handles
MyBase.FormClosing
        Form1.Show()
    End Sub

 Private Sub Form2_Load(sender As Object, e As EventArgs) Handles MyBase.Load

        user = Form1.Lregno.Text
        Todaydate.Text = Date.Today.ToString("dd-MMMM-yyyy")
        GunaPanel6.BackColor = Color.FromArgb(100, 0, 0, 0)
        GunaPanel5.BackColor = Color.FromArgb(100, 0, 0, 0)

   GunaPanel7.BackColor = Color.FromArgb(70, 0, 0, 0)
        GunaPanel8.BackColor = Color.FromArgb(70, 0, 0, 0)
        userinfopanel.Hide()
        RichTextBox1.Hide()
        GunaPanel9.Hide()
        GunaPanel5.Hide()
      GunaPanel11.Hide()
```

Admin Login:

```
Dim admin1 As String = "21BCAA54"
        Dim admin2 As String = "1"



 If user <> admin1 AndAlso user <> admin2 Then
```

```
        Report.Hide()
    End If

    Dim con As New SqlConnection

 con.ConnectionString = "Data
Source=(LocalDB)\MSSQLLocalDB;AttachDbFilename=C:\Users\Minesh\Downloads\lostandfound\lostan
dfound\registration.mdf;Integrated Security=True"

    con.Open()
    Dim stmt As String
    stmt = "select fname,regno,email from register where regno='" & user & "'"

    Dim cmd As New SqlCommand(stmt, con)
    Dim reader As SqlDataReader
    reader = cmd.ExecuteReader
    If reader.Read() Then
        Label8.Text = reader("fname")
        Fname.Text = reader("fname")
        Regno.Text = reader("regno")
        Email.Text = reader("email")
    Else
        MessageBox.Show("error")
 End If
End Sub
```

Report button click

```
Private Sub Report_Click(sender As Object, e As EventArgs) Handles Report.Click
    GunaLabel4.Text = "Report"
    GunaPanel11.Show()
    userinfopanel.Show()
    foundpanel.Hide()
    lostpanel.Hide()
    Today.BaseColor = Color.FromArgb(100, 88, 255)
    Thismonth.BaseColor = Color.FromArgb(100, 88, 255)
    Auction.BaseColor = Color.FromArgb(100, 88, 255)

End Sub


Private Sub Buttonfound_Click(sender As Object, e As EventArgs) Handles Buttonfound.Click

    foundpanel.Show()
    userinfopanel.Hide()
    lostpanel.Hide()
    GunaPanel11.Hide()
End Sub
```

```
Private Sub Buttonlost_Click(sender As Object, e As EventArgs) Handles Buttonlost.Click
        lostpanel.Show()
        foundpanel.Hide()
        userinfopanel.Hide()
        GunaPanel11.Hide()
    End Sub

    Private Sub Phoneno_KeyPress(sender As Object, e As KeyPressEventArgs) Handles Phoneno.KeyPress
        If Not Char.IsDigit(e.KeyChar) AndAlso Not Char.IsControl(e.KeyChar) Then
            e.Handled = True 'Ignore the key press if it's not a digit or control character
            ErrorProvider1.SetError(Phoneno, "Enter a valid phone number")
        Else
            ErrorProvider1.SetError(Phoneno, "") 'Clear the error message if the input is valid
        End If
    End Sub



  Private Sub Buttonselectimage_Click(sender As Object, e As EventArgs) Handles
Buttonselectimage.Click
        If OpenFileDialog1.ShowDialog = Windows.Forms.DialogResult.OK Then
            Fitemphoto.Image = Image.FromFile(OpenFileDialog1.FileName)
        End If
    End Sub
```

Submit Button of found item:

```
Private Sub Buttonsubmit_Click(sender As Object, e As EventArgs) Handles Buttonsubmit.Click

        Dim con As New SqlConnection

  con.ConnectionString = "Data
Source=(LocalDB)\MSSQLLocalDB;AttachDbFilename=C:\Users\Minesh\Downloads\lostandfound\lostan
dfound\registration.mdf;Integrated Security=True"


Dim cmd As New SqlCommand("Insert Into Found values('" & Fitemname.Text & "','" & Fitemcolour.Text
& "','" & Fbrandname.Text & "','" & Founddate.Text & "','" & Fkeydesc.Text & "','" & placefound.Text &
"','" & Fitemcondition.Text & "',@ph,'" & Fname.Text & "','" & Regno.Text & "','" & Email.Text & "','" &
Phoneno.Text & "','" & Sem.Text & "','" & Course.Text & "','" & Section.Text & "','" & Todaydate.Text &
"')", con)
        If (Fitemphoto.Image IsNot Nothing) Then

            Dim ms As New MemoryStream
            Dim img As Image = Fitemphoto.Image
            Dim bmpimage As New Bitmap(img)
            bmpimage.Save(ms, System.Drawing.Imaging.ImageFormat.Jpeg)

  Dim data As Byte() = ms.GetBuffer
```

```
        Dim p As New SqlParameter("@ph", SqlDbType.Image)
        p.Value = data
        cmd.Parameters.Add(p)
    End If

    con.Open()

    If (Fitemname.Text = "" Or Fitemcolour.Text = "" Or Fbrandname.Text = "" Or Founddate.Text = "" Or Fkeydesc.Text = "" Or placefound.Text = "" Or Fitemcondition.Text = "" Or Fname.Text = "" Or Regno.Text = "" Or Email.Text = "" Or Phoneno.Text = "" Or Sem.Text = "" Or Course.Text = "" Or Section.Text = "") Then
        MessageBox.Show("Fill the details")

    ElseIf ErrorProvider1.GetError(Founddate) <> "" Then
        MessageBox.Show("Please fill in all required fields correctly.", "Error", MessageBoxButtons.OK, MessageBoxIcon.Error)


    ElseIf Fitemphoto.Image Is Nothing Then
        MessageBox.Show("Please upload image")
    ElseIf Phoneno.Text.Length < 10 Then
        ErrorProvider1.SetError(Phoneno, "Enter a valid phone number")


    Else cmd.ExecuteNonQuery()
        MessageBox.Show("added")
        Fitemcolour.Text = ""
        Fitemname.Text = ""
        Fbrandname.Text = ""
        Founddate.Text = ""
        Fkeydesc.Text = ""
        placefound.Text = ""
        Fitemcondition.Text = ""
        Fitemphoto.Image = Nothing
        Phoneno.Text = ""
        Sem.SelectedItem = Nothing
        Course.Text = ""
        Section.Text = ""

End If
    con.Close()

 End Sub
```

Calculate the Levenshtein Distance between two strings:

```
Function LevenshteinDistance(s1 As String, s2 As String) As Integer
    Dim d(s1.Length, s2.Length) As Integer
    For i As Integer = 0 To s1.Length
      d(i, 0) = i
    Next
    For j As Integer = 0 To s2.Length
      d(0, j) = j
    Next
    For i As Integer = 1 To s1.Length
      For j As Integer = 1 To s2.Length
        Dim cost As Integer = If(s1(i - 1) = s2(j - 1), 0, 1)
        d(i, j) = Math.Min(Math.Min(d(i - 1, j) + 1, d(i, j - 1) + 1), d(i - 1, j - 1) + cost)
      Next
    Next
    Return d(s1.Length, s2.Length)
  End Function
```

Button submit of Lost Items:

```
Private Sub GunaButton1_Click(sender As Object, e As EventArgs) Handles GunaButton1.Click
    If (Litemname.Text = "" Or Litemcolour.Text = "" Or Lbrandname.Text = "" Or Lostdate.Text = "" Or
Lkeydesc.Text = "" Or Placelost.Text = "" Or Litemcondition.Text = "") Then
      MessageBox.Show("Fill the details")
    ElseIf ErrorProvider1.GetError(Lostdate) <> "" Then



  MessageBox.Show("Please fill in all required fields correctly.", "Error", MessageBoxButtons.OK,
MessageBoxIcon.Error)

    Else Dim con As New SqlConnection("Data
Source=(LocalDB)\MSSQLLocalDB;AttachDbFilename=C:\Users\Minesh\Downloads\lostandfound\lostan
dfound\registration.mdf;Integrated Security=True")
      con.Open()
      ' //Select all items found after the specified lost date, including the image data//
      Dim cmd As New SqlCommand("SELECT * FROM found WHERE founddate >= @lostdate", con)
      cmd.Parameters.AddWithValue("@lostdate", Lostdate.Value)
      Dim reader As SqlDataReader = cmd.ExecuteReader()

      ' //Loop through all found items and add the matching items to the list//
      Dim matchingItems As New StringBuilder()
      While reader.Read()
```

```
' //Check if the item name matches the search query//
            Dim itemName As String = reader("Fitemname")
            If LevenshteinDistance(Litemname.Text, itemName) <= itemName.Length * 0.6 Then


' //Check if any two values match fitemcolour, fbrandname, and placefound//
                If (Litemcolour.Text.ToLower() = reader("Fitemcolour").ToString().ToLower() AndAlso
Lbrandname.Text.ToLower() = reader("Fbrandname").ToString().ToLower()) _
                OrElse (Litemcolour.Text.ToLower() = reader("Fitemcolour").ToString().ToLower()
AndAlso Placelost.Text.ToLower() = reader("Placefound").ToString().ToLower()) _
                OrElse (Lbrandname.Text.ToLower() = reader("Fbrandname").ToString().ToLower()
AndAlso Placelost.Text.ToLower() = reader("Placefound").ToString().ToLower()) Then

                Dim imageData As Byte() = TryCast(reader("Fitemphoto"), Byte())
                If imageData IsNot Nothing Then
                   Using stream As New MemoryStream(imageData)
                      GunaPictureBox2.Image = Image.FromStream(stream)
                   End Using
                End If


 ' //Add the matching item details to the StringBuilder//

                matchingItems.AppendLine($"Place found: {reader("Placefound")}")
                matchingItems.AppendLine($"Person Name: {reader("Fname")}")
                matchingItems.AppendLine($"Person Regno: {reader("Regno")}")


                matchingItems.AppendLine($"Person Email: {reader("Email")}")
                matchingItems.AppendLine($"Person Phoneno: {reader("Phoneno")}")
                matchingItems.AppendLine($"Person Sem: {reader("Sem")}")
                matchingItems.AppendLine($"Person Course: {reader("Course")}")
                matchingItems.AppendLine($"Person Section: {reader("Section")}")
                'matchingItems.AppendLine() ' Add a blank line between each item
             End If
          End If
       End While

'// show result of seatch items//
       Dim finalmatch = matchingItems.ToString()
       If finalmatch.Trim() = String.Empty Then
          GunaPanel5.Show()
          GunaPanel9.Show()
          Label30.Text = "No Items Found"
          RichTextBox2.Text = "No matching items found."
          GunaPictureBox2.Image = Nothing
          Buttonyes.Hide()
          Buttonno.Hide()
```

```
 Else
            GunaPanel5.Show()
            RichTextBox1.Text = "Matching items found:" & vbCrLf & finalmatch.ToString()

          End If
        End If
    End Sub

  '// button yes click to confirm lost item//
    Private Sub Buttonyes_Click(sender As Object, e As EventArgs) Handles Buttonyes.Click
        If GunaPictureBox2.Image IsNot Nothing Then
            RichTextBox1.Show()
        End If
        GunaPanel9.Hide()
    End Sub
    '// Button no click to confirm shown item is not yours //

Private Sub Buttonno_Click(sender As Object, e As EventArgs) Handles Buttonno.Click
        RichTextBox1.Hide()
        If GunaPictureBox2.Image IsNot Nothing Then
            GunaPanel9.Show()
            RichTextBox2.Text = "No matching items found."


        End If

        GunaPictureBox2.Image = Nothing
    End Sub

'// clear form on ok button click after lost search result//
    Private Sub GunaButton2_Click(sender As Object, e As EventArgs) Handles GunaButton2.Click
        GunaPanel5.Hide()
        Litemcolour.Text = ""
        Litemname.Text = ""
        Lbrandname.Text = ""
        Lostdate.Text = ""
        Lkeydesc.Text = ""
        Placelost.Text = ""
        Litemcondition.Text = ""
        GunaPictureBox2.Image = Nothing
    End Sub
```

Lost date validation:

```
    Private Sub Lostdate_ValueChanged(sender As Object, e As EventArgs) Handles
Lostdate.ValueChanged

        Dim selectedDate As Date = Lostdate.Value.Date
```

```
If selectedDate > Date.Today Then
        ErrorProvider1.SetError(Lostdate, "Selected date cannot be in the future.")
        Return

End If

  ErrorProvider1.SetError(Lostdate, "")

 End Sub
```

Found date validation:

```
Private Sub Founddate_ValueChanged(sender As Object, e As EventArgs) Handles
Founddate.ValueChanged
      Dim selectedDate As Date = Founddate.Value.Date
      If selectedDate > Date.Today Then
         ErrorProvider1.SetError(Founddate, "Selected date cannot be in the future.")
         Return
      End If
      ErrorProvider1.SetError(Founddate, "")
   End Sub

Private Sub Today_Click(sender As Object, e As EventArgs) Handles Today.Click
      GunaLabel4.Text = "Today's Report"
      todayfound.Show()
      todayclaimed.Show()

    monthlyfound.Hide()
      monthlyclaimed.Hide()
      Today.BaseColor = Color.Lime
      Thismonth.BaseColor = Color.FromArgb(100, 88, 255)
      Auction.BaseColor = Color.FromArgb(100, 88, 255)
   End Sub
   Private Sub todayfound_Click(sender As Object, e As EventArgs) Handles todayfound.Click


      Dim con As New SqlConnection("Data
Source=(LocalDB)\MSSQLLocalDB;AttachDbFilename=C:\Users\Minesh\Downloads\lostandfound\lostan
dfound\registration.mdf;Integrated Security=True")
      con.Open()

      Dim cmd As New SqlCommand("SELECT Fitemname, Fitemcolour, Fbrandname, Founddate,
placefound, Fname AS [Founders name], Regno FROM Found WHERE Todaydate = CONVERT(date,
GETDATE())", con)

      Dim adapter As New SqlDataAdapter(cmd)
      Dim table As New DataTable()
      adapter.Fill(table)
```

```
        GunaDataGridView1.DataSource = table

        GunaDataGridView1.Columns("Fitemname").HeaderText = "Item Name"
        GunaDataGridView1.Columns("Fitemcolour").HeaderText = "Item Colour"
        GunaDataGridView1.Columns("Fbrandname").HeaderText = "Brand Name"
        GunaDataGridView1.Columns("Founddate").HeaderText = "Date Found"
       GunaDataGridView1.Columns("placefound").HeaderText = "Place Found"
        GunaDataGridView1.Columns("Founders name").HeaderText = "Founders name"
        GunaDataGridView1.Columns("Regno").HeaderText = "Registration No."

        If table.Rows.Count > 0 Then
            GunaDataGridView1.DataSource = table

        End If

        con.Close()
    End Sub


 Private Sub Thismonth_Click(sender As Object, e As EventArgs) Handles Thismonth.Click
        GunaLabel4.Text = "Monthly Report"
        todayfound.Hide()
        todayclaimed.Hide()
        monthlyfound.Show()
        monthlyclaimed.Show()
        Thismonth.BaseColor = Color.Lime
        Today.BaseColor = Color.FromArgb(100, 88, 255)
        Auction.BaseColor = Color.FromArgb(100, 88, 255)
    End Sub
    Private Sub monthlyfound_Click(sender As Object, e As EventArgs) Handles monthlyfound.Click
        Dim con As New SqlConnection("Data
Source=(LocalDB)\MSSQLLocalDB;AttachDbFilename=C:\Users\Minesh\Downloads\lostandfound\lostan
dfound\registration.mdf;Integrated Security=True")
        con.Open()
        Dim cmd As New SqlCommand("SELECT Fitemname, Fitemcolour, Fbrandname, Founddate,
placefound, Fname AS [Founders name], Regno FROM Found WHERE Todaydate BETWEEN
DATEADD(day, -30, GETDATE()) AND GETDATE()", con)
        Dim adapter As New SqlDataAdapter(cmd)
        Dim table As New DataTable()
        adapter.Fill(table)

        GunaDataGridView1.DataSource = table

        GunaDataGridView1.Columns("Fitemname").HeaderText = "Item Name"
        GunaDataGridView1.Columns("Fitemcolour").HeaderText = "Item Colour"
        GunaDataGridView1.Columns("Fbrandname").HeaderText = "Brand Name"
        GunaDataGridView1.Columns("Founddate").HeaderText = "Date Found"
        GunaDataGridView1.Columns("placefound").HeaderText = "Place Found"
        GunaDataGridView1.Columns("Founders name").HeaderText = "Founders name"
        GunaDataGridView1.Columns("Regno").HeaderText = "Registration No."
```

```
        If table.Rows.Count > 0 Then
            GunaDataGridView1.DataSource = table
        End If
        con.Close()
    End Sub

Private Sub Auction_Click(sender As Object, e As EventArgs) Handles Auction.Click
        GunaLabel4.Text = "Welcome To Auction"
        todayfound.Hide()
        todayclaimed.Hide()
        monthlyfound.Hide()
        monthlyclaimed.Hide()

        Auction.BaseColor = Color.Lime
        Thismonth.BaseColor = Color.FromArgb(100, 88, 255)
        Today.BaseColor = Color.FromArgb(100, 88, 255)
    End Sub
End Class
```

4. SYSTEM CONFIGURATION

4.1 HARDWARE REQUIREMENTS:-

PROCESSOR : INTEL CORE

SPEED : 1.50GHZ

RAM : 4GB

4.2 SOFTWARE REQUIREMENTS:-

FRONTEND : VB.NET (VISUALSTUDIO)2019

BACKEND : SQL SERVER

DOCUMENTATON : MS WORD 2007

GANTT CHART : MS PROJECT 2013

OPERATING SYSTEM : WINDOWS10

5. DETAILS OF SOFTWARE

5.1 OVERVIEW OF FRONT-END:

Microsoft Visual Studio 2019 is an integrated development environment (IDE) from Microsoft. It is used to develop computer programs for Microsoft Windows, as well as web sites, web apps, web services and mobile apps. Visual Studio uses Microsoft software development platforms such as Windows API, Windows Forms, Windows Presentation Foundation, Windows Store and Microsoft Silverlight. It can produce both native code and managed code.

Visual Studio supports different programming languages and allows the code editor and debugger to support nearly any programming language, provided a language-specific service exists. Built-in languages include C, C++, Visual C++ and VB.NET. Support for other languages such as Python, Ruby, Node.js, and M among others is available via language services installed separately.

It also supports XML/XSLT, HTML/XHTML, JavaScript and CSS. Java (and J#) were supported in the past**.** Microsoft provides a free version of Visual Studio called the Community edition that supports plugins and is available at no cost for all users. Support for programming languages isadded by usinga specific VSPackage called a Language Service.
A language service defines various interfaces which the VSPackage implementation can implement to add support for various functionalities. Functionalities that can be added this way include syntax coloring, statement completion, brace matching, parameter information tooltips, member lists and error markers for background compilation.

If the interface is implemented, the functionality will be available for the language. Language services are implemented on a per-language basis. The implementations can reuse code from the parser or the compiler for the language. Language services can be implemented either in native code or managed code. For native code, either the native COM interfaces or the Babel Framework can be used. For managed code, the MPF includes wrappers for writing managed language services.

FEATURES:

- Boolean Conditions
- Automatic Garbage Collection
- Standard Library

- Assembly Versioning
- Properties and Events
- Delegates and Events Management
- Easy-to-use Generics
- Indexers
- Conditional Compilation
- Simple Multithreading

ADVANTAGES:

- The structure of the Basic programming language is very simple, particularly as to the executable code.
- VB is not only a language but primarily an integrated, interactive development environment ("IDE").
- The VB-IDE has been highly optimized to support rapid application development("RAD"). It is particularly easy to develop graphical user interfaces and to connect them to handler functions provided by the application.
- The graphical user interface of the VB-IDE provides intuitively appealing views for the management of the program structure in the large and the various types of entities (classes, modules, procedures, forms, …).
- VB provides a comprehensive interactive and context-sensitive online helpsystem.
- When editing program texts the "IntelliSense" technology informs you in a little popup window about the types of constructs that may be entered at the current cursor location.
- VB is a component integration language which is attuned to Microsoft's Component Object Model ("COM").
- COM components can be written in different languages and then integrated using VB.
- Interfaces of COM components can be easily called remotely via Distributed COM ("DCOM"), which makes it easy to construct distributed applications.
- COM components can be embedded in / linked to your application's user interface and also in/to stored documents (Object Linking and Embedding "OLE", "Compound Documents").
- There is a wealth of readily available COM components for many different purposes.
- Visual Basic is built around the .NET environment used by all Microsoft

5.2 OVERVIEW OF BACK-END:

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. The protocol layer implements the external interface to SQL

Server. All operations that can be invoked on SQL Server are communicated to it via a Microsoft-defined format, called Tabular Data Stream (TDS).

TDS is an application layer protocol, used to transfer data between a database server and a client. Initially designed and developed by Sybase Inc. for their Sybase SQL Server relational database engine in 1984, and later by Microsoft in Microsoft SQL Server, TDS packets can be encased in other physical transport dependent protocols, including TCP/IP, named pipes, and shared memory. Consequently, access to SQL Server is available over these protocols. In addition, the SQL Server API is also exposed over web services.

A Relational Data Base Management System (RDBMS) is software that: Enables you to implement a database with tables, columns and indexes. Guarantees the Referential Integrity between rows of various tables. Updates the indexes automatically. Interprets an SQL query and combines information from various tables. MySQL is a leading open source database management system.

It is a multiuser, multithreaded database management system. MySQL is especially popular on the web. It is one of the parts of the very popular LAMP platform. Linux, Apache, MySQL and PHP. MySQL database is available on most important OS platforms. It runs on BSD Unix, Linux, Windows or Mac. Wikipedia, YouTube, Facebook use MySQL. These sites manage millions of queries each day. MySQL comes in two versions.

MySQL server system and MySQL embedded system. The MySQL server software and the client libraries are dual-licensed:

Advantages:

MySQL is becoming so popular because of many good reasons:
MySQL is released under an open-source license. So you have nothing to pay to use it. MySQL is a very powerful program in its own right. It handles a large subset of the functionality of the most expensive and powerful database packages. MySQL uses a standard form of the well-known SQL data language. MySQL works on many operating systems and with many languages including PHP, PERL, C, C++, JAVA, etc. MySQL works very quickly and works well even with large data sets. MySQL is very friendly to PHP, the most appreciated language for web development. MySQL supports large databases, up to 50 million rows or more in a table. The default file size limit for a table is 4GB, but you can increase this (if your operating system can handle it) to a theoretical limit of 8 million terabytes (TB). MySQL is customizable. The open-source GPL license allows programmers to modify the MySQL software to fit ther own specific environments.

FEATURES:

- Written in C and C++.
- Tested with a broad range of different compilers.
- Works on many different platforms. .
- For portability, uses **CMake** in MySQL 5.5 and up. Previous series use GNU Automake,Autoconf, and Libtool.
- Tested with Purify (a commercial memory leakage detector) as well as with Valgrind, a GPL tool.
- Uses multi-layered server design with independent modules.
- Designed to be fully multi-threaded using kernel threads, to easily use multiple CPUs if they are available.
- Provides transactional and nontransactional storage engines.
- Fixed-length and variable-length string types.

Statements and Functions

mysql> SELECT CONCAT(first_name, ' ', last_name) -> FROM citizen

-> WHERE income/dependents > 10000 AND age > 30;Full support for SQL GROUP BY and ORDER BY clauses. Support for andGROUP_CONCAT()).

Support for LEFT OUTER JOIN and RIGHT OUTER JOIN with both standard SQL and ODBC syntax. Support for aliases on tables and columns as required by standard SQL. Some were changed (affected) to, return the number of rows matched instead by setting a flag when connecting to the server. Support for MySQL-specific SHOW statements that retrieve information about databases, storage engines, tables, and indexes. Support for the INFORMATION_SCHEMA database, implemented according to standard SQL.

An explain statement to show how the optimizer resolves a query.

Independence of function names from table or column names. For example, ABS is a valid column name. The only restriction is that for a function call, no spaces are permitted between the function name and the "(" that follows it.You can refer to tables from different databases in the same statement.

SECURITY:

A privilege and password system that is very flexible and secure, and that enables host-based verification.

Password security by encryption of all password traffic when you connect to a server.

SCALABILITY AND LIMITS:

Support for large databases. We use MySQL Server with databases that contain 50 million records. We also know of users who use MySQL Server with 200,000 tables and about 5,000,000,000 rows.

Support for up to 64 indexes per table. Each index may consist of 1 to 16 columns or parts of columns. The maximum index width for InnoDB tables is either 767 bytes or 3072 bytes. The maximum index width for My ISAM tables is 1000 bytes. An index may use a prefix of a column for CHAR, VARCHAR, BLOB, or TEXT column types.

CONNECTIVITY:

- Clients can connect to MySQL Server using several protocols:
- Clients can connect using TCP/IP sockets on any platform.
- On Unix systems, clients can connect using Unix domain socket files.

LOCALIZATION:

The server can provide error messages to clients in many languages.

Full support for several different character sets, including latin1 (cp1252), german, big5, ujis, several Unicode character sets, and more. For example, the

Scandinavian characters “å”, “ä” and “ö” are permitted in table and column names.All

data is saved in the chosen character set.

Sorting and comparisons are done according to the chosen character set and collation

(using latin1 and Swedish collation by default). It is possible to change this when the MySQL server is started. To see an example of very advanced sorting, look at the Czech sorting code. MySQL Server supports many different character sets that can be specified at compile time and runtime.

The server time zone can be changed dynamically, and individual clients can specify their own time zone.

CLIENTS AND TOOLS:

- MySQL includes several client and utility programs. These include both command-line programs and graphical programs.
- MySQL Server has built-in support for SQL statements to check, optimize, and repair tables.
- These statements are available from the command line through the mysqlcheck client. MySQL also includes myisamch, a very fast command-line utility for performing these operations on MyISAM tables..
- MySQL programs can be invoked with the --help or -? option to obtain online assistance.

5.3 ABOUT THE PLATFORM :

Windows is a series of Operating Systems developed by Microsoft. Each version of Windows includes a Graphical User Interface, with a desktop that allows users to view files and folders in Windows. For the past two decades, Windows has been the most widely used operating system for personal computers PCs.

Microsoft Windows is designed for both home computing and professional purposes. Past versions of Windows home editions include Windows 3.0 (1990), Windows 3.1 (1992), Windows 95 (1995), Windows 98 (1998), Windows Me (2000), Windows XP (2001), and Windows Vista (2006),windows 7(2009),windows 8(2012),windows 8.1(2013)..

Windows 10 is a series of operating systems developed by Microsoft and released as part of its Windows NT family of operating systems. It is the successor to Windows 8.1, released nearly two years earlier, and was released to manufacturing on July 15, 2015, and broadly released for the general public on July 29, 2015 .Windows Vista and Windows 7 have followed the same release strategy. Windows is designed to run on standard x86 hardware, such as Intel and AMD processors. Therefore, it can be installed on multiple brands of hardware, such as Dell, HP, and Sony computers, as well as home-built PCs.

During upgrades, Windows 10 licenses are not tied directly to a product key. Instead, the license status of the system's current installation of Windows is migrated, and a "Digital license" (also known as "Digital entitlement" in version 1511 or earlier) is generated during the activationn process

.NET FRAMEWORK:

.NET Framework (pronounced as "dot net") is a software framework developed by Microsoft that runs primarily on Microsoft Windows. It includes a large class library named Framework Class Library (FCL) and provides language interoperability (each language can use code written in other languages) across several programming languages.

Programs written for .NET Framework execute in a software environment (in contrast to a hardware environment) named Common Language Runtime (CLR), an application virtual machine that provides services such as security, memory management, and exception handling.

As such, computer code written using .NET Framework is called "managed code". FCL and CLR together constitute the .NET Framework. FCL provides user interface, data access, database connectivity, cryptography, web application development, numeric algorithms, and network communications. Programmers produce software by combining their source code with .NET Framework and other libraries.

The framework is intended to be used by newest applications created for the Windows platform. Microsoft also produces an integrated development environment largely for .NET software called Visual Studio**. .**NET Framework began as proprietary software, although the firm worked to standardize the software stack almost immediately, even before its first release.

Despite the standardization efforts, developers, mainly those in the free and open-source software communities, expressed their unease with the selected terms and the prospects of any free and open-source implementation, especially regarding software patents. Since then, Microsoft has changed .NET development to more closely follow a contemporary model of a community developed software project, including issuing an update to its patent promising to address the concerns.

6.TESTING

Testing is a vital part of software development, and it is important to start it as early as possible, and to make testing a part of the process of deciding requirements. To get the most useful perspective on your development project, it is worthwhile devoting some thought to the entire lifecycle including how feedback from users will influence the future of the application.

The tools and techniques we've discussed in this book should help your team to be more responsive to changes without extra cost, despite the necessarily wide variety of different development processes. Nevertheless, new tools and process improvements should be adopted gradually, assessing the results after each step.

Testing is part of a lifecycle. The software development lifecycle is one in which you hear of a need, you write some code to fulfil it, and then you check to see whether you have pleased the stakeholders—the users, owners, and other people who have an interest in what the software does.

Hopefully they like it, but would also like some additions or changes, so you update or augment your code; and so the cycle continues. This cycle might happen every few days, as it does in Fabrikam's ice cream vending project, or every few years, as it does in Contoso's carefully specified and tested healthcare support system. Software development lifecycle Testing is a proxy for the customer.

You could conceivably do your testing by releasing it into the wild and waiting for the complaints and compliments to come back. Some companies have been accused of having such a strategy as their business model even before it became fashionable. But on the whole, the books are better balanced by trying to make sure that the software will satisfy the customer before we hand it over.

We therefore design tests based on the stakeholders' needs, and run the tests before the product reaches the users. Preferably well before then, so as not to waste our time working on something that isn't going to do the job.

In this light, two important principles become clear:

TESTS REPRESENT REQUIREMENTS: Whether you write user stories on sticky notes on the wall, or use cases in a big thick document, your tests should be derived from and linked to those requirements. And as we've said, devising tests is a good vehicle for discussing the requirements.

WE ARE NOT DONE TILL THE TEST IS PASSED: The only useful measure of completionis when tests have been performed successfully.

Those principles apply no matter how you develop your software.

SOFTWARE TESTING TYPES:

BLACK BOX TESTING – Internal system design is not considered in this type of testing. Tests arebased on requirements and functionality.

WHITE BOX TESTING – This testing is based on knowledge of the internal logic of an application's code. Also known as Glass box Testing. Internal software and code working should be known forthis type of testing. Tests are based on coverage of code statements, branches, paths, conditions.

UNIT TESTING – Testing of individual software components or modules. Typically done by the programmer and not by testers, as it requires detailed knowledge of the internal program design and code. may require developing test driver modules or test harnesses.

INCREMENTAL INTEGRATION TESTING – Bottom up approach for testing i.e continuous testing of an application as new functionality is added; Application functionality and modules should be independent enough to test separately. done by programmers or by testers.

INTEGRATION TESTING – Testing of integrated modules to verify combined functionality after integration. Modules are typically code modules, individual applications, client and server applications on a network, etc. This type of testing is especially relevant to client/server and distributed systems.

FUNCTIONAL TESTING – This type of testing ignores the internal parts and focus on the output is as per requirement or not. Black-box type testing geared to functional requirements of an application.

SYSTEM TESTING – Entire system is tested as per the requirements. Black-box type testing that is based on overall requirements specifications, covers all combined parts of a system.

END-TO-END TESTING – Similar to system testing, involves testing of a complete application environment in a situation that mimics real-world use, such as interacting with a database, using network communications, or interacting with other hardware, applications, or systems if appropriate

SANITY TESTING – Testing to determine if a new software version is performing well enough to accept it for a major testing effort. If application is crashing for initial use then system is not stable enough for further testing and build or application is assigned to fix.

REGRESSION TESTING – Testing the application as a whole for the modification in any module or functionality. Difficult to cover all the system in regression testing so typically automation tools are used for these testing types.

ACCEPTANCE TESTING -Normally this type of testing is done to verify if system meets the customer specified requirements. User or customer do this testing to determine whether to accept application.

LOAD TESTING– Its a performance testing to check system behavior under load. Testing an application under heavy loads, such as testing of a web site under a range of loads to determine at what point the system's response time degrades or fails.

STRESS TESTING– System is stressed beyond its specifications to check how and when it fails. Performed under heavy load like putting large number beyond storage capacity, complex database queries, continuous input to system or database load.

PERFORMANCE TESTING – Term often used interchangeably with 'stress' and 'load' testing. To check whether system meets performance requirements. Used different performance and load tools to do this.

USABILITY TESTING – User-friendliness check. Application flow is tested, Can new user understand the application easily, Proper help documented whenever user stuck at any point. Basically system navigation is checked in this testing.

INSTALL/UNINSTALL TESTING– Tested for full, partial, or upgrade install/uninstall processes on different operating systems under different hardware, software environment.

RECOVERY TESTIG – Testing how well a system recovers from crashes, hardware failures, or other catastrophic problems.

SECURITY TESTING – Can system be penetrated by any hacking way. Testing how well the system protects against unauthorized internal or external access.

Checked if system, database is safe from external attacks.

COMPABILITY TESTING– Testing how well software performs in a particular hardware/software/operating system/network environment and different combination s of above.

COMPARISON TESTING– Comparison of product strengths and weaknesses with previous versions or other similar products.

ALPHA TESTING– In house virtual user environment can be created for this type of testing. Testing is done at the end of development. Still minor design changes may be made as a result of such testing.

BETA TESTING – Testing typically done by end-users or others. Final testing before releasing application for commercial purpose.

Testing done for Lost And Found:

Testing names

1. User input & Validation testing
2. Button submit Test
3. Panel testing
4. Form testing
5. Logical code testing

User input & Validation testing: In this testing all the user inputs and validation made were tested some of the validation made were

Name (It should only have alphabets)

Regno (It should have alphabets and number other character not allowed)

Email(It should be college Email with @kristujayanti.com)

Dates(It should not be in future)

Phone no(It should contain 10 integer)

Button submit Test: In this testing all the button click were tested

Login Button: (Should login to next form if userid and password matches)

Register Button: (Should register new user and insert user details in database, if user already present show error)

Select image : (Should select image form folder)

Found submit button: (Should insert details in database if all inputs are valid)

Panel testing: In this testing many panels were tested when they are hidden and shown Lost panel was should hide when found panel is shown, login panel should hide when Register panel is shown and vice versa.

Form testing: On close of form2, form2 should close and form1 should appear.

Logical code testing: In this testing function of code were tested like
Calculating the Levenshtein Distance between two strings.

Select all items found after the specified lost date, including the image data.

Loop through all found items and add the matching items to the list.

Check if any two values match fitemcolour, fbrandname, and placefound.

Add the matching item details to the StringBuilder.

Test Case:

Test Name	Step	Test Data	Expected Result	Actual Result	status
Username /password	Type username &password	21BCAA54	Logged in successfully	Logged in successfully	Pass
Email validation	Type college email	21BCAA54@ kristujayanti.com	Accept email	Accept email	Pass
Email/checking	Type different email	sam@gmail.com	Do not accept email	Does not accepts email	Pass
Register form name	Type number	123	Enter alphabet	Enter alphabet only	Pass
Register form name	Type alphabet	ABC	Accept alphabet	Accepts alphabet	Pass
Found date& Lost date	Select date in future	05-07-2024	Date can not be in future	Date can not be in future	Pass
Phone no	Type alphabet& characters	ABC#$%	Do not accepts phone no	Does not accepts phone no	Pass
Null values	Enter null values	“_____”	Do not accept Null values	Does not accept Null values	Pass
Matching string	Enter string name of item	Item name	Show the matching string	Shows the matching string	Pass
Report Generator	Click report button		Show report from the database	Show’s report	pass
Two field match	Enter the data in fields		Check two matching field and show result	Check two matching field and show result	Pass

7.CONCLUSION AND FUTURE ENHANCEMENT:

CONCLUSION:

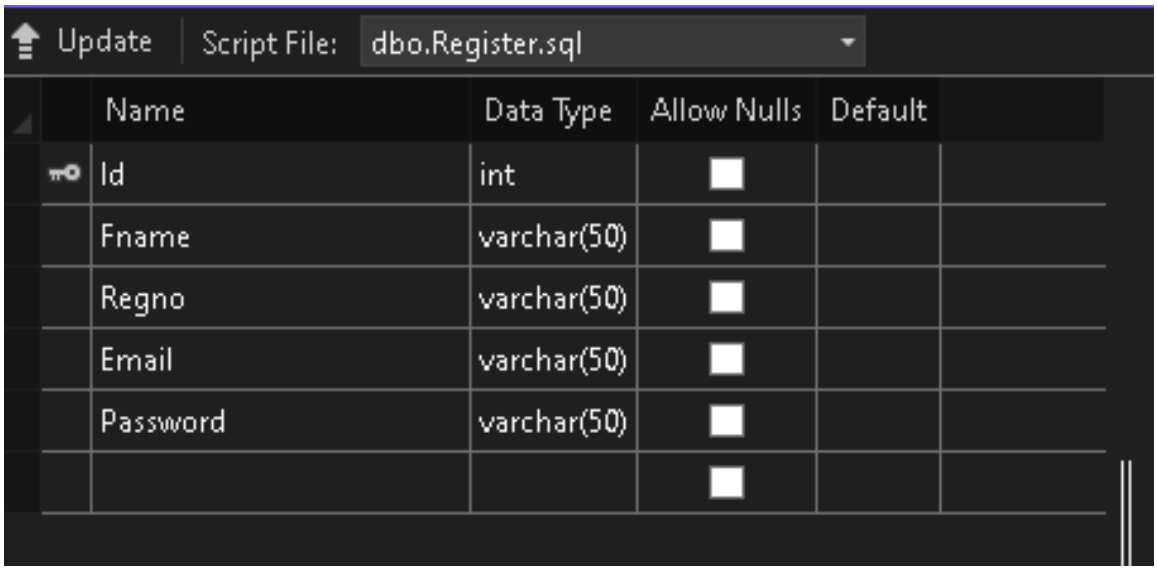
This mini project titled “LOST AND FOUND ” is a simple project which can help in finding lost items of people. This is just a first version of project developed in limited number of days. This application can find items with 70% of accuracy as of now it finds items by comparing string value i.e., item name , color ,place etc.

FUTURE ENHANCEMENT:

This project can be enhanced in future using deep learning and image processing and image recognizing which can improve its accuracy of 70% to 95% .

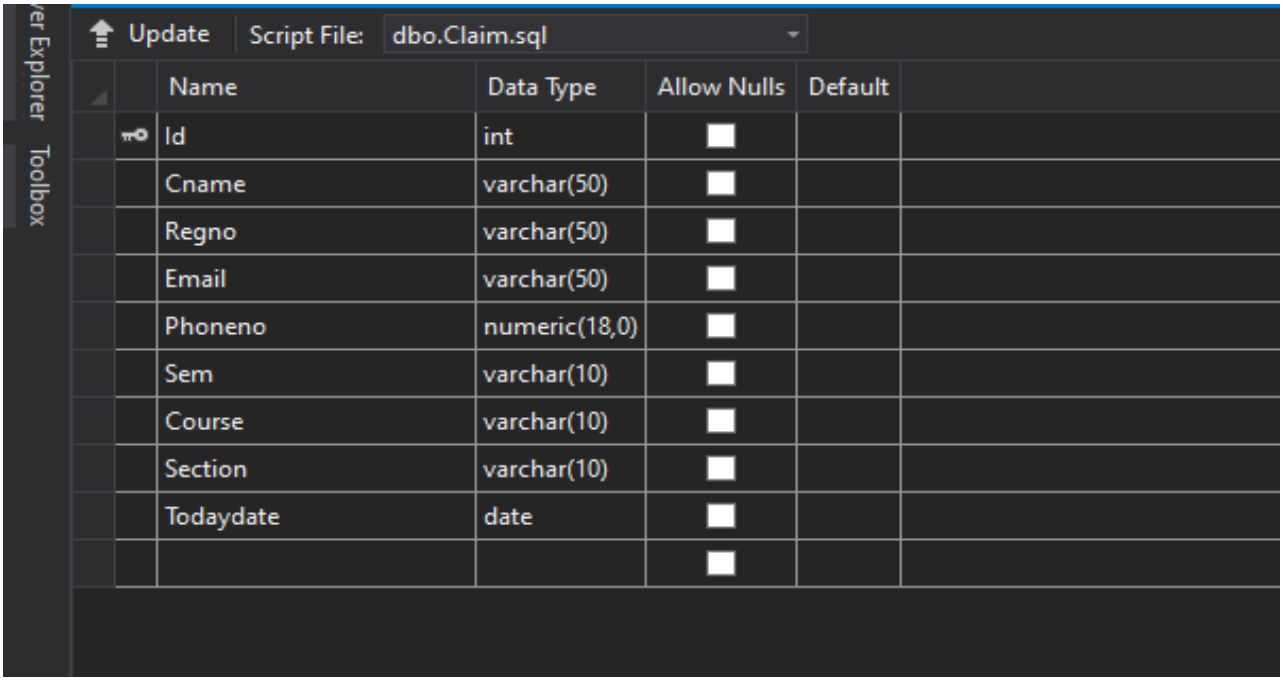
8. APPENDICES A-TABLE STRUCTURE:

Register:



Name	Data Type	Allow Nulls	Default
Id	int	☐	
Fname	varchar(50)	☐	
Regno	varchar(50)	☐	
Email	varchar(50)	☐	
Password	varchar(50)	☐	
		☐	

Claim Table:



Name	Data Type	Allow Nulls	Default
Id	int	☐	
Cname	varchar(50)	☐	
Regno	varchar(50)	☐	
Email	varchar(50)	☐	
Phoneno	numeric(18,0)	☐	
Sem	varchar(10)	☐	
Course	varchar(10)	☐	
Section	varchar(10)	☐	
Todaydate	date	☐	
		☐	

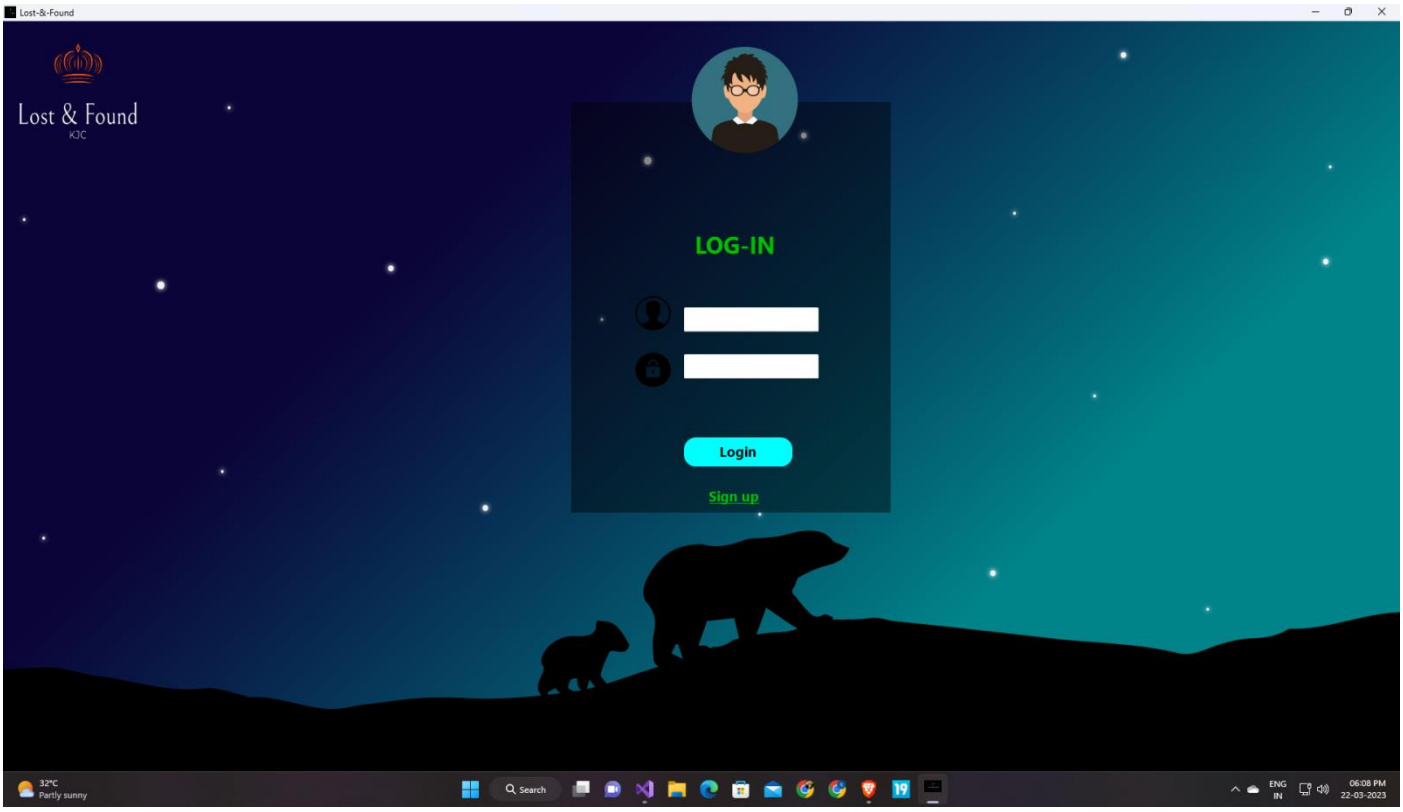
Found table :

Update | Script File: dbo.Found.sql

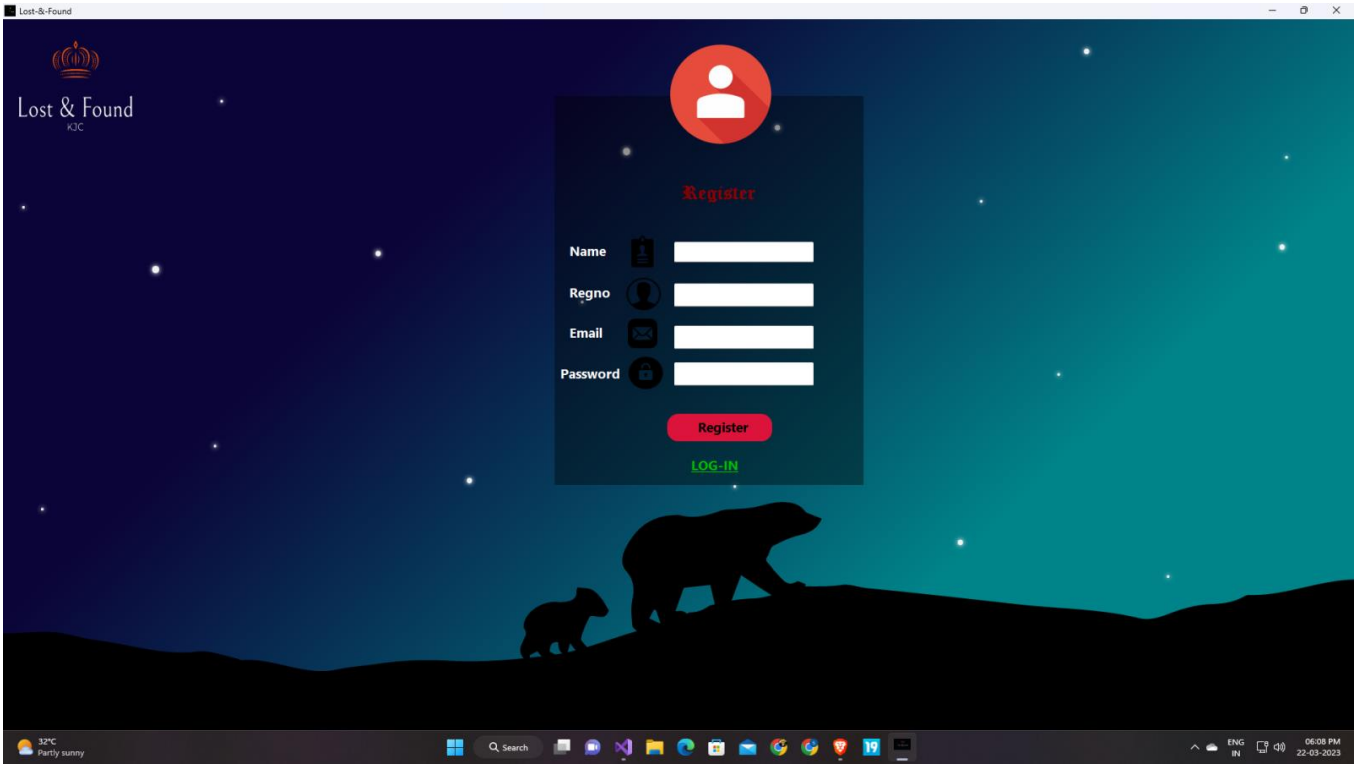
Name	Data Type	Allow Nulls	Default
Id	int	☐	
Fitemname	varchar(50)	☐	
Fitemcolour	varchar(50)	☐	
Fbrandname	varchar(50)	☐	
Founddate	date	☐	
Fkeydesc	varchar(50)	☐	
placefound	varchar(50)	☐	
Fitemcondition	varchar(50)	☐	
Fitemphoto	image	☐	
Fname	varchar(50)	☐	
Regno	varchar(50)	☐	
Email	varchar(50)	☐	
Phoneno	numeric(18,0)	☐	
Sem	varchar(10)	☐	
Course	varchar(10)	☐	
Section	varchar(10)	☐	
Todaydate	date	☐	
		☐	

9.APPENDICES-B SCREENSHOTS

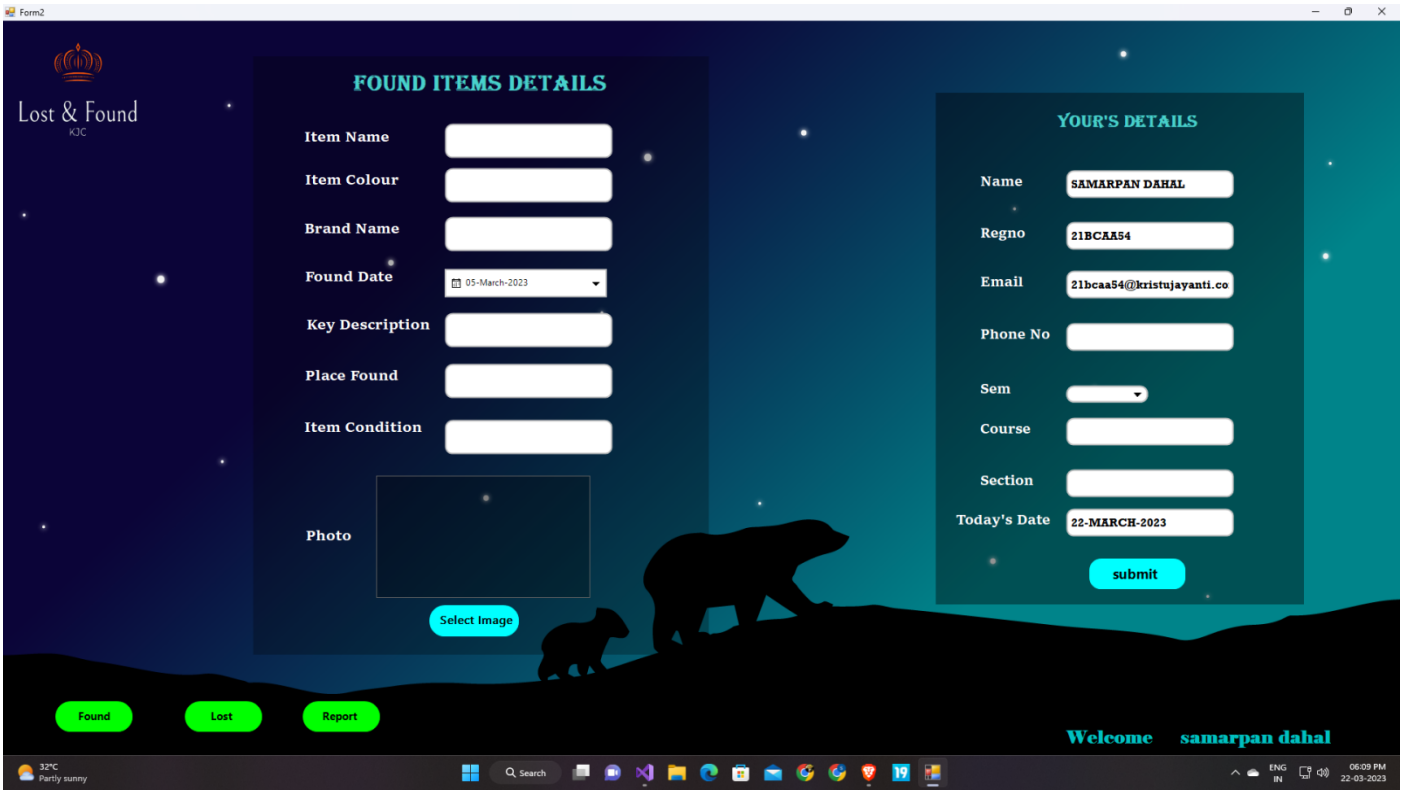
Login page:



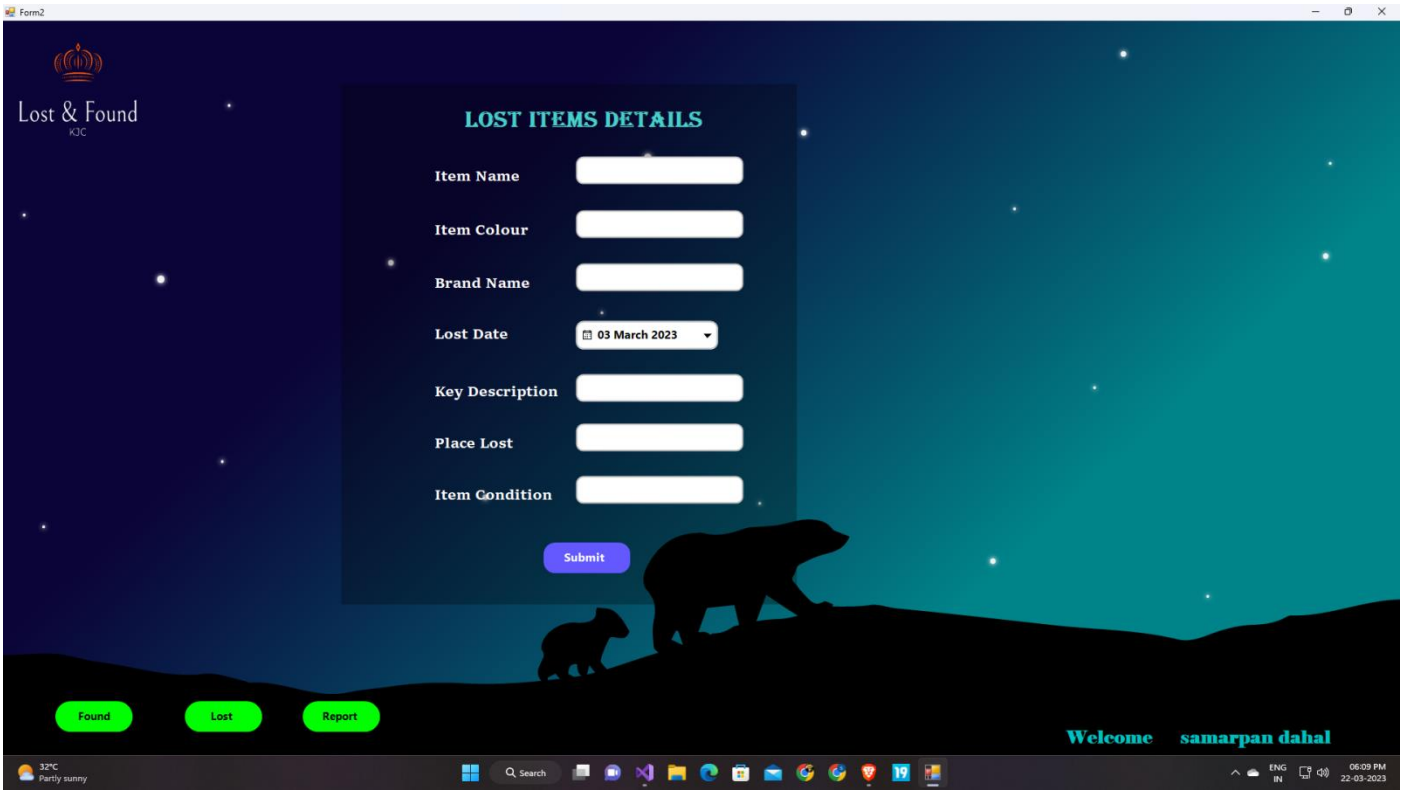
Register page:



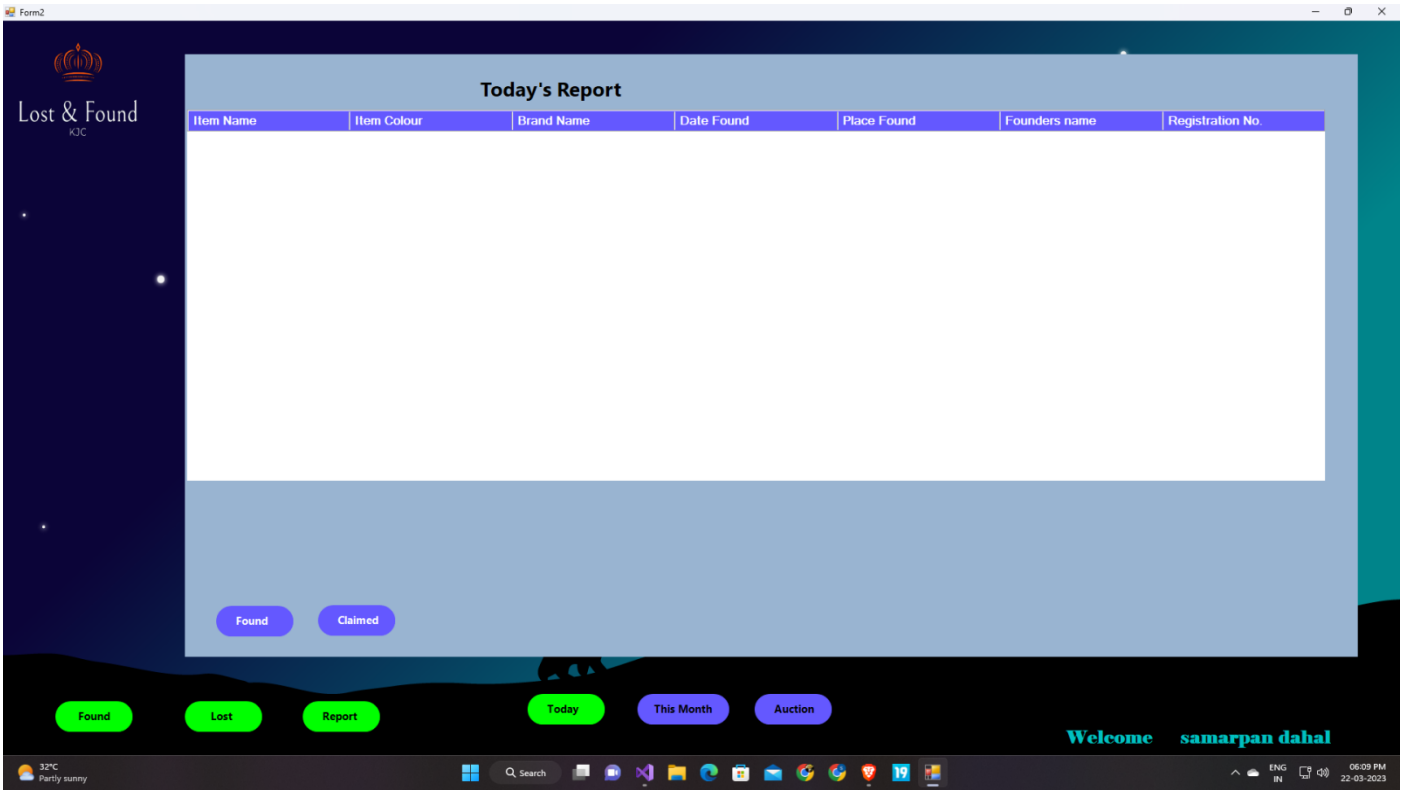
Found section:

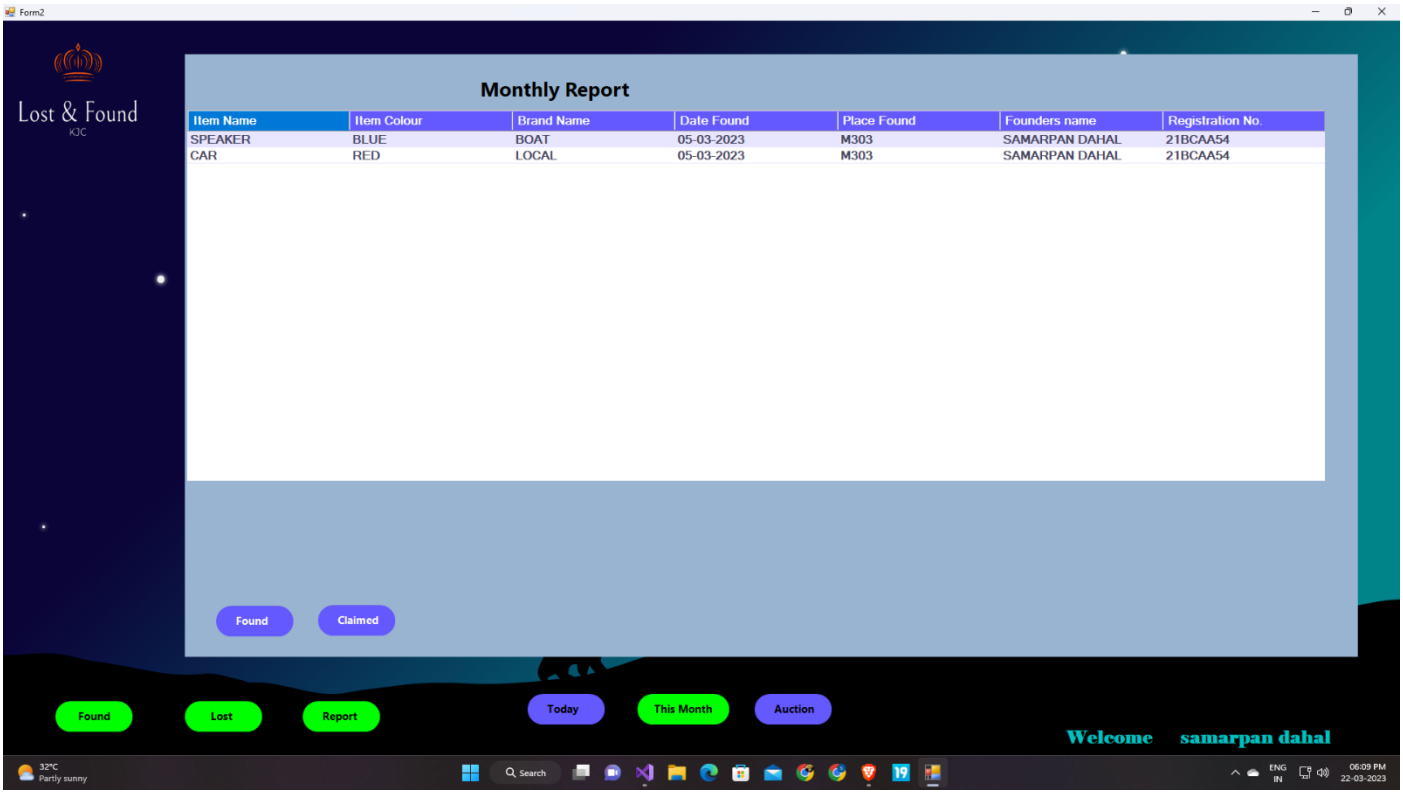


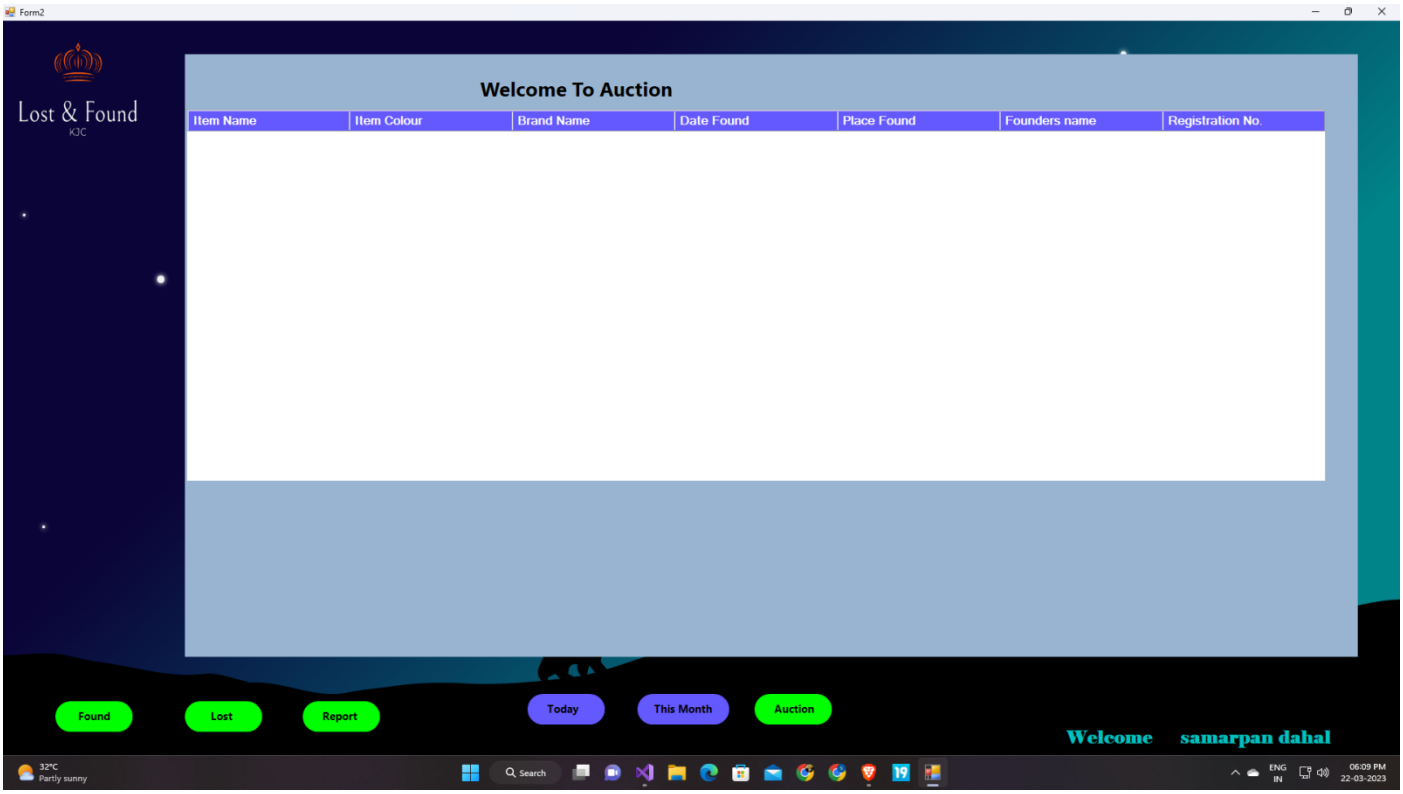
Lost section:



Report :


Form2
Lost & Found
KJC
Today's Report
Item Name
Item Colour
Brand Name
Date Found
Place Found
Founders name
Registration No.
Found
Claimed
Found
Lost
Report
Today
This Month
Auction
Welcome samarpan dahal


Form2
Lost & Found
KJC
Monthly Report
Item Name
Item Colour
Brand Name
Date Found
Place Found
Founders name
Registration No.
SPEAKER BLUE BOAT 05-03-2023 M303 SAMARPAN DAHAL 21BCAA54
CAR RED LOCAL 05-03-2023 M303 SAMARPAN DAHAL 21BCAA54
Found
Claimed
Found
Lost
Report
Today
This Month
Auction
Welcome samarpan dahal


Form2
Lost & Found
KJC
Welcome To Auction
Item Name
Item Colour
Brand Name
Date Found
Place Found
Founders name
Registration No.
Found
Lost
Report
Today
This Month
Auction
Welcome samarpan dahal
32°C
Partly sunny
Search
ENG
IN
06:09 PM
22-03-2023