

## **CHAPTER 1**

# **INTRODUCTION**

### **1.1 Course Objectives**

- Students are expected to independently identify a real time scenario problem related to the field of computer science.
- Carry out a mini project on the problem defined.
- Mini project should have all the requirement Specification, implementation and the outcomes Screenshots and also include conclusion and references.
- Projects should be coded (or programmed) individually and independently.
- The code developed towards the project will be reviewed by a panel of examiners at multiples levels.
- At the end of the project completion the student should be able to understand all the core concepts of the language used and along with the database.

### **1.2 Problem Statement**

Every Industry, School, Office workspace always used some form of register books or entry IN and OUT form, Leave form, Attendance sheet etc. The online hostel management system is web-based software to provide college students accommodation to the university hostel more efficiently. This project also keeps details of the hostellers and applied students. It is headed by Warden. He will be the administrator. For accommodate a large number of students into hostel. This document is intended to minimize human works and make hostel allocation is an easier job. All over the world this traditional system has been followed and still is being followed. So i am developing a software program which would does all this requirement digitally into the system and store it in proper databases where access will be given to specific login. Which would help us move ahead

in technology in daily life and also reduce the use of paper and Hence saving the Environment.

### **1.3 Outcomes of the Project Work carried out**

Software Program will be giving these outputs:

- Login form for user and admin Access.
- Admin can add students, view their details and delete the student who left the hostel. Here warden is the admin.
- Software will take automatically all inputs stored in databases for frequent user access.
- Admin will be responsible for the creation of userid and password for students.
- Students will login with provided password and id.
- Students must write a leave before going home that is been implemented in the form by using java swing, student should fill the details such as usn, room no, father no, location, till date etc.
- Students are also able to view the leaves submitted by other students and take note of the unique id of their leave.
- While going out on daily basis there is another form to be filled by student called IN/OUT form which will keep track of students who are inside and outside the hostel.
- Firstly, students fill out form and take their unique id to be entered while coming in.
- Out form includes things like usn, phone no, location, etc. date and time will be automatically entered in database using current timestamp function.
- While coming in student should enter their unique id from out form to enter the hostel, this will be able to delete the out form in the database helping warden to filter out how many are still outside and who already came in.

## CHAPTER 2

# JAVA FEATURES AND OOPS CONCEPT

## 2.1 FEATURES

Java is an object-oriented, cross platform, multi-purpose programming language produced by Sun Microsystems. First released in 1995, it was developed to be a machine independent web technology. It was based on C and C++ syntax to make it easy for programmers from those communities to learn. Since then, it has earned a prominent place in the world of computer programming. Java has many characteristics that have contributed to its popularity:

The prime reason behind creation of Java was to bring portability and security feature into a computer language. Beside these two major features, there were many other features that played an important role in molding out the final form of this outstanding language. Those features are:

### 1) Simple

Java is easy to learn and its syntax is quite simple, clean and easy to understand. The confusing and ambiguous concepts of C++ are either left out in Java or they have been re-implemented in a cleaner way.

*Ex:* Pointers and Operator Overloading are not there in java but were an important part of C++.

### 2) Object Oriented

In java everything is Object which has some data and behavior. Java can be easily extended as it is based on Object Model.

### 3) Robust

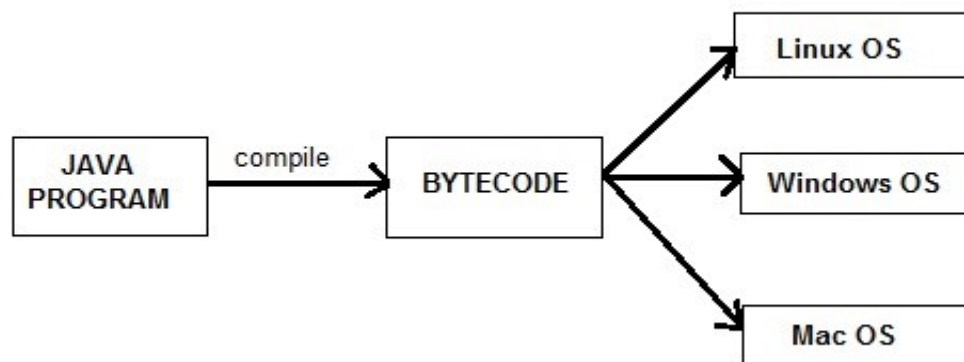
Java makes an effort to eliminate error prone codes by emphasizing mainly on compile time error checking and runtime checking. But the main areas which Java improved were

Memory Management and mishandled Exceptions by introducing automatic Garbage Collector and Exception Handling.

#### 4) Platform Independent

Unlike other programming languages such as C, C++ etc. which are compiled into platform specific machines. Java is guaranteed to be write-once, run-anywhere language.

On compilation Java program is compiled into bytecode. This bytecode is platform independent and can be run on any machine, plus this bytecode format also provide security. Any machine with Java Runtime Environment can run Java Programs.



#### 5) Secure

When it comes to security, Java is always the first choice. With java secure features it enables us to develop virus free, temper free system. Java program always runs in Java runtime environment with almost null interaction with system OS, hence it is more secure.

#### 6) Multi-Threading

Java multithreading feature makes it possible to write program that can do many tasks simultaneously. Benefit of multithreading is that it utilizes same memory and other resources to execute multiple threads at the same time, like While typing, grammatical errors are checked along.

### **7) Architectural Neutral**

Compiler generates bytecodes, which have nothing to do with a particular computer architecture, hence a Java program is easy to interpret on any machine.

### **8) Portable**

Java Byte code can be carried to any platform. No implementation dependent features. Everything related to storage is predefined, example: size of primitive data types.

### **9) High Performance**

Java is an interpreted language, so it will never be as fast as a compiled language like C or C++. But Java enables high performance with the use of just-in-time compiler.

## **2.2 OOPS CONCEPTS**

Object Oriented Programming is a programming concept that works on the principle that objects are the most important part of your program. It allows users create the objects that they want and then create methods to handle those objects. Manipulating these objects to get results is the goal of Object-Oriented Programming. Object Oriented Programming popularly known as OOP, is used in a modern programming language like Java.

Object means a real-world entity such as a pen, chair, table, computer, watch, etc. Object-Oriented Programming is a methodology or paradigm to design a program using classes and objects. It simplifies the software development and maintenance by providing some concepts:

### **1) Class**

The class is a group of similar entities. It is only a logical component and not the physical entity. For example, if you had a class called “Expensive Cars” it could have objects like

Mercedes, BMW, Toyota, etc. Its properties(data) can be price or speed of these cars. While the methods may be performed with these cars are driving, reverse, braking etc.

## **2) Object**

An object can be defined as an instance of a class, and there can be multiple instances of a class in a program. An Object contains both the data and the function, which operates on the data. For example - chair, bike, marker, pen, table, car, etc.

## **3) Inheritance**

Inheritance is an OOPS concept in which one object acquires the properties and behaviours of the parent object. It's creating a parent-child relationship between two classes. It offers robust and natural mechanism for organizing and structure of any software.

## **4) Polymorphism**

Polymorphism refers to the ability of a variable, object or function to take on multiple forms. For example, in English, the verb "run" has a different meaning if you use it with "a laptop," "a foot race, and" business. &duo Here, we understand the meaning of "run" based on the other words used along with it. The same also applied to Polymorphism.

## **5) Abstraction**

An abstraction is an act of representing essential features without including background details. It is a technique of creating a new data type that is suited for a specific application. For example, while driving a car, you do not have to be concerned with its internal working. Here you just need to concern about parts like steering wheel, Gears, accelerator, etc.

## **6) Encapsulation**

Encapsulation is an OOP technique of wrapping the data and code. In this OOPS concept, the variables of a class are always hidden from other classes. It can only be accessed using the methods of their current class. For example - in school, a student cannot exist without a class.

## **7) Association**

Association is a relationship between two objects. It defines the diversity between objects. In this OOP concept, all object has their separate lifecycle, and there is no owner. For example, many students can associate with one teacher while one student can also associate with multiple teachers.

## **8) Aggregation**

In this technique, all objects have their separate lifecycle. However, there is ownership such that child object can't belong to another parent object. For example, consider class/objects department and teacher. Here, a single teacher can't belong to multiple departments, but even if we delete the department, the teacher object will never be destroyed.

## **9) Composition**

A composition is a specialized form of Aggregation. It is also called "death" relationship. Child objects do not have their lifecycle so when parent object deletes all child object will also delete automatically. For that, let's take an example of House and rooms. Any house can have several rooms. One room can't become part of two different houses. So, if you delete the house room will also be deleted.

## 2.3 Other Features.

- **Platform independence** - Many languages are compatible with only one platform. Java was specifically designed so that it would run on any computer, regardless if it was running Windows, Linux, Mac, Unix or any of the other operating systems.
- **Simple and easy to use** - Java's creators tried to design it so code could be written efficiently and easily.
- **Multi-functional** - Java can produce many applications from command-line programs to applets to Swing windows (basically, sophisticated graphical user interfaces).
  - Mobile applications (especially Android apps)
  - Desktop applications
  - Web applications
  - Web servers and application servers
  - Games
  - Database connection

## 2.4 WEB DEVELOPMENT WITH JAVA

Java has strong support for web development. While Java on the desktop, with the notable exception of Eclipse RCP based application was never a huge success, Java is frequently used at the server side.

If you develop a web application (independent of the programming language you are using), you typically put your web application on a dedicated server (and not your local computer). The web application runs on the server and people can access it there. The server is either a real machine (with CPU, memory, hard disk, etc.) or a virtual server which is basically a machine which is separated by software into smaller machines.

It is possible to use your local computer as a server, but usually you want to have a fixed server which runs 24 hours per day, 7 days per week so that web clients can always reach your server under a pre-defined address.



## CHAPTER 3

# REQUIREMENTS AND DESIGN

### 3.1 Project Requirements

- Login form for specific user and admin.
- System Time and Date will be automatically taken into form instead of writing it every time in the form.
- The program would be able to take inputs to form in a web portal.
- Store the form in databases specified for each different form.
- It will be able to take digital signatures for verification of user.
- Attendance and absentees' sheet will be automatically entered into the proper databases.

### 3.1 Hardware Requirements

The section of hardware configuration is an important task related to the software development insufficient random-access memory may affect adversely on the speed and efficiency of the entire system. The process should be powerful to handle the entire operations. The hard disk should have sufficient capacity to store the file and application.

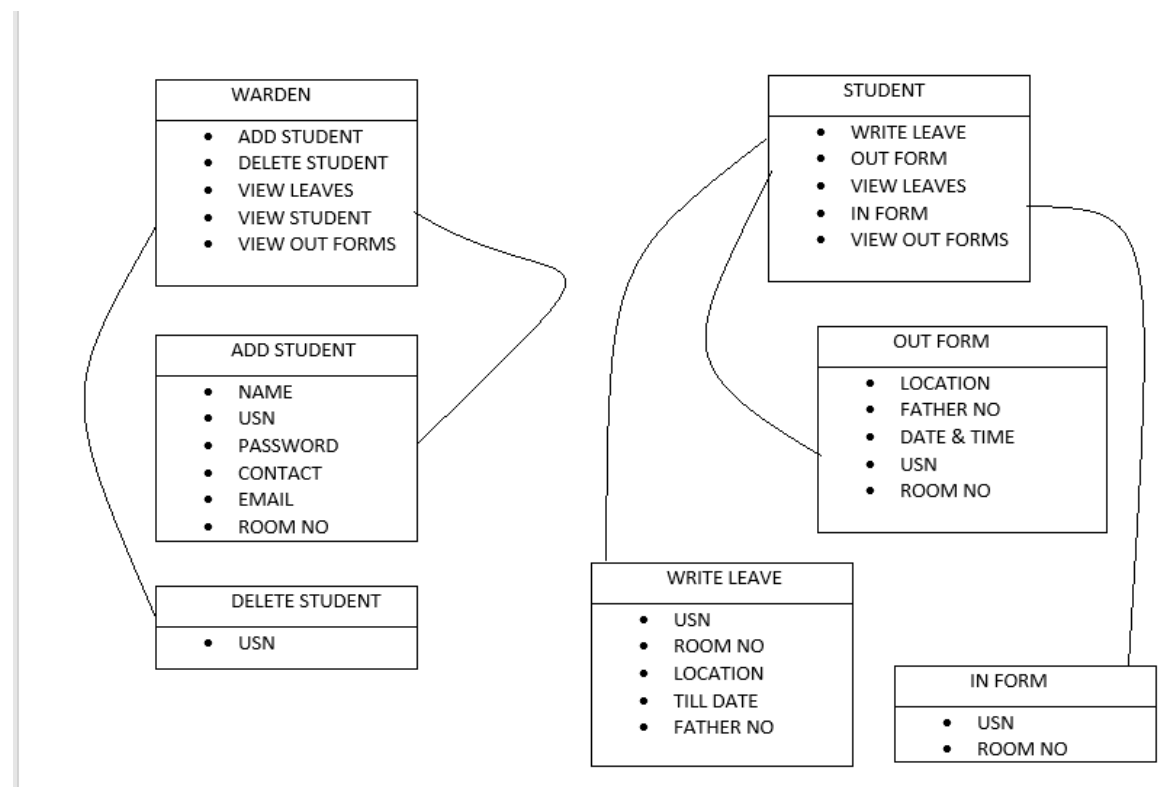
Processor speed	1.4 GHz Onwards
Cache size	512 KB
Processor	Pentium IV and above
System memory	256 Mb recommended
RAM	512 MB(Minimum)
Network card	Any card can provide a 100mbps speed
Hard disk	80Gb
Modem	56.6 Kbps

### 3.2 Software Requirements

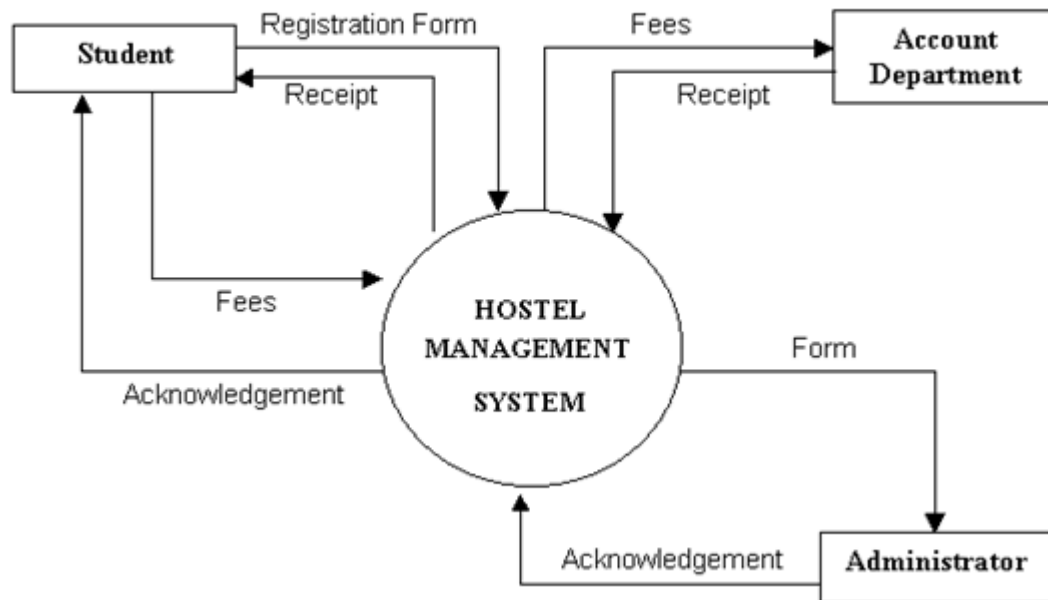
A major element in building a system is the section of compatible software since the software in the market is experiencing in geometric progression. Selected software should be acceptable by the firm and one user as well as it should be feasible for the system. This document gives a detailed description of the software requirement specification. The study of requirement specification is focused specially on the functioning of the system. it allows the developer or analyst to understand the system, function to be carried out the performance level to be obtained and corresponding interfaces to be established.

Front end tool	Java Swing and windows builder
Operating System	Windows 7 and above.
Backend	Xampp My SQL
Java version	Eclipse 8.0

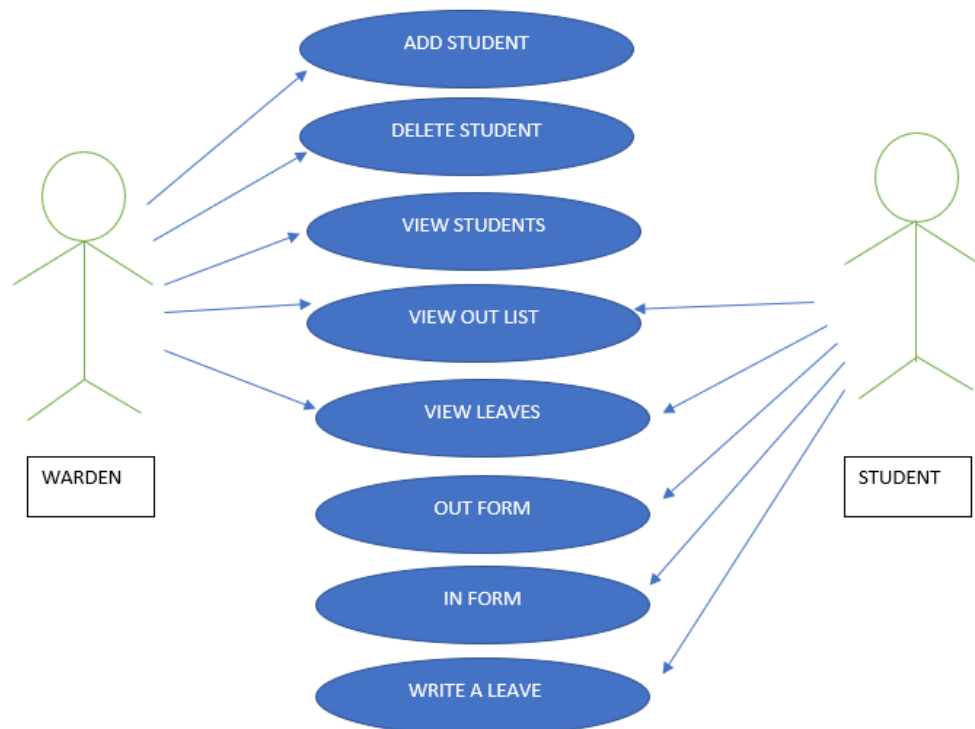
### 3.3 Class Diagram.



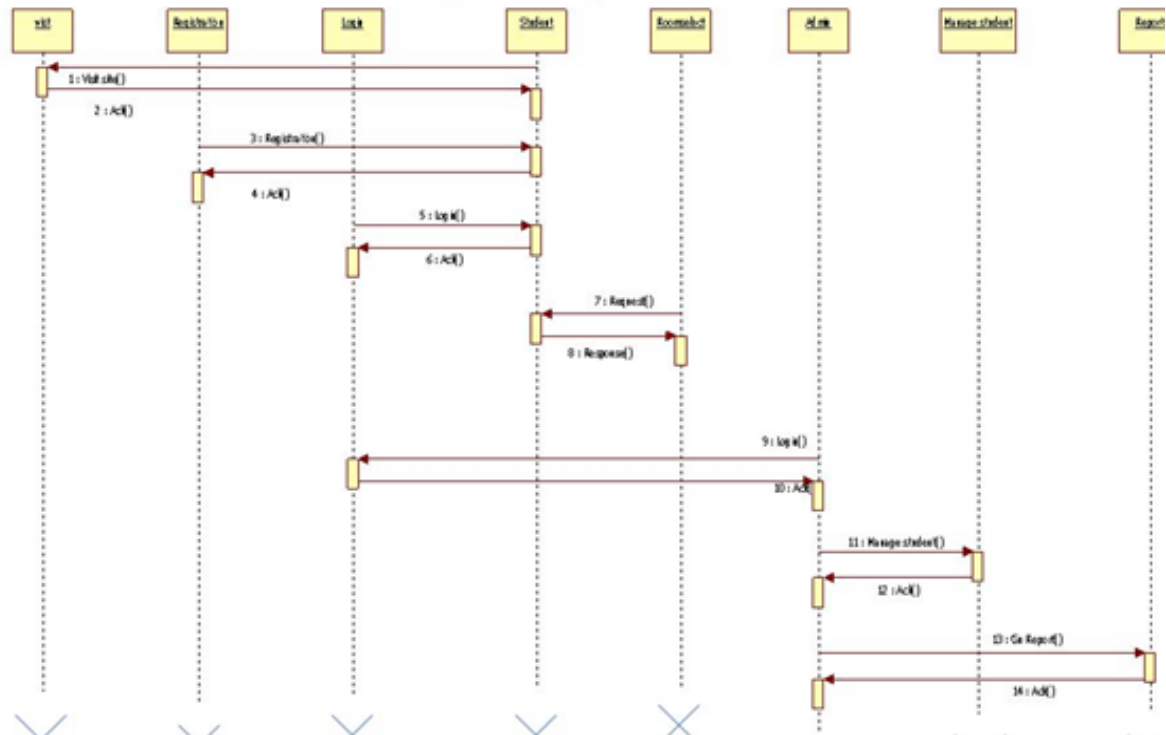
### 3.4 Data Flow Diagram.



### 3.5 Use Case Diagram.



### 3.6 Sequence Diagram.



## CHAPTER 4

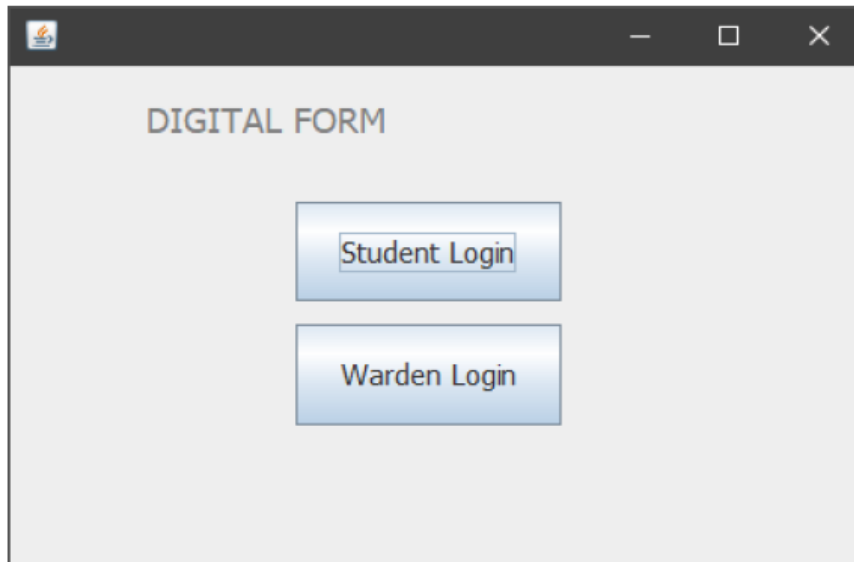
### IMPLEMENTATION

- Hostel system requires high database management strategy as it stores the data of the student present, went out and previous students list.
- In this software application we are trying to build a eco friendly system which would eliminate the use of paper for daily register book for going out and leave form filling.
- The software will have 2 logins one student and one warden.
- Warden is able add student, give password and login details to student.
- Warden has access to old student details and she/he can delete the student from the hostel.
- Student can access the system by the usn and password given by the warden
- Students should write leave while going for vacations or holidays for home with full details such as location, room no, father no, usn, and till date etc.
- Students should write OUT form while going for day with friends for eating shopping with full details such as location, room no, student no, usn etc.
- Student once entering the hostel should fill IN form for the deletion of OUT form from the database and helps warden only to see students who are still outside.
- Both Warden and Student are able view leave forms, OUT forms of other students.
- Every form will have an unique id which will help deletion of data from the database.

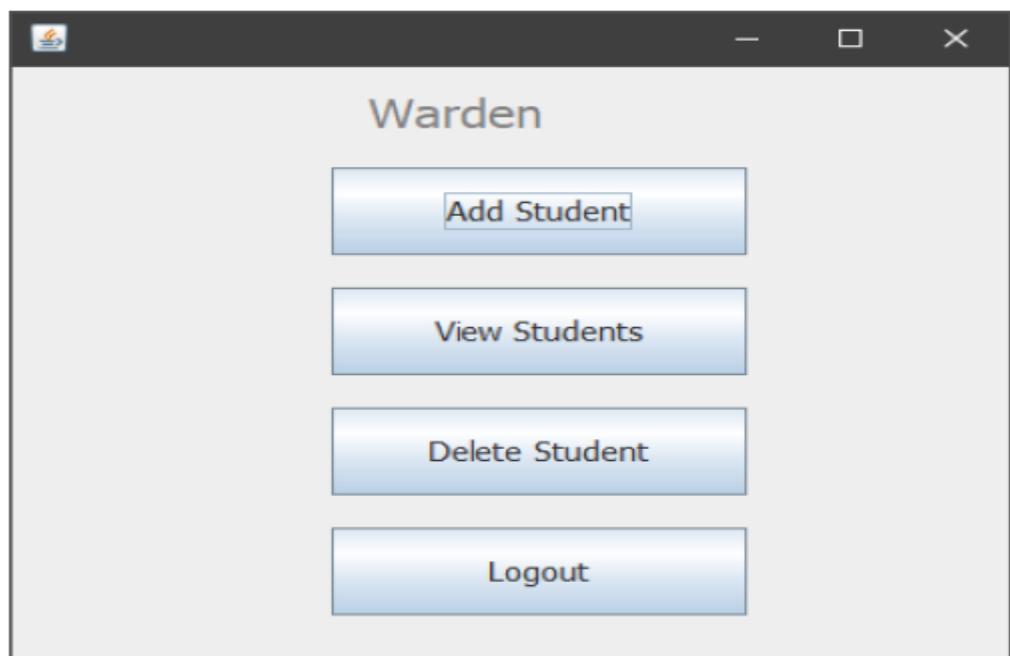
## CHAPTER 5

### SCREENSHOTS

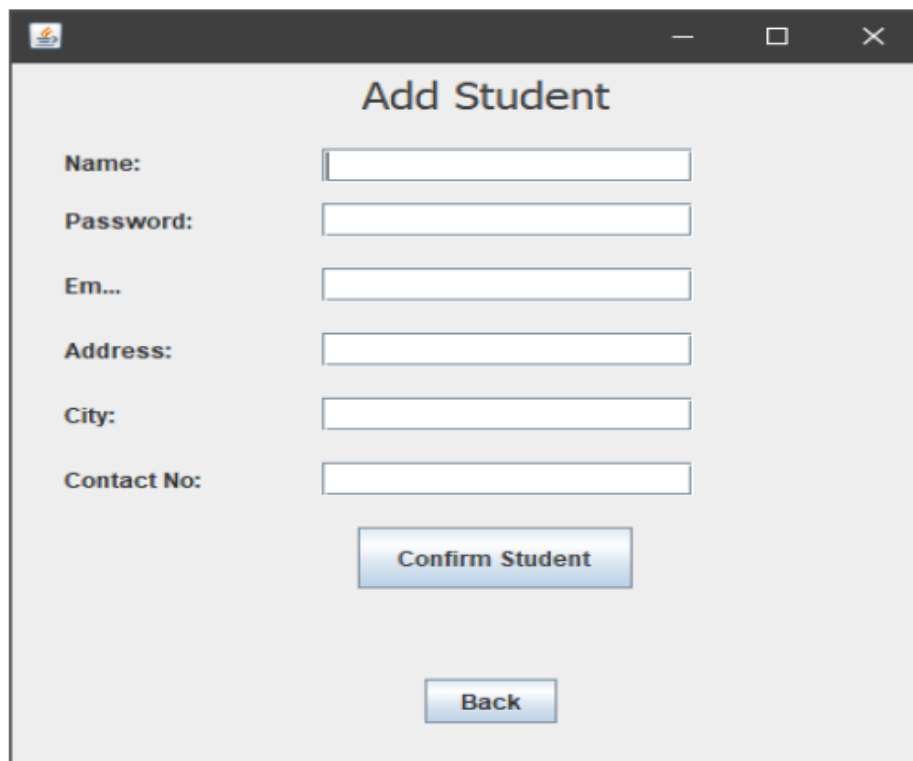
#### 5.1. LOGIN FORM



#### 5.2. WARDEN



### 5.3. ADD STUDENT

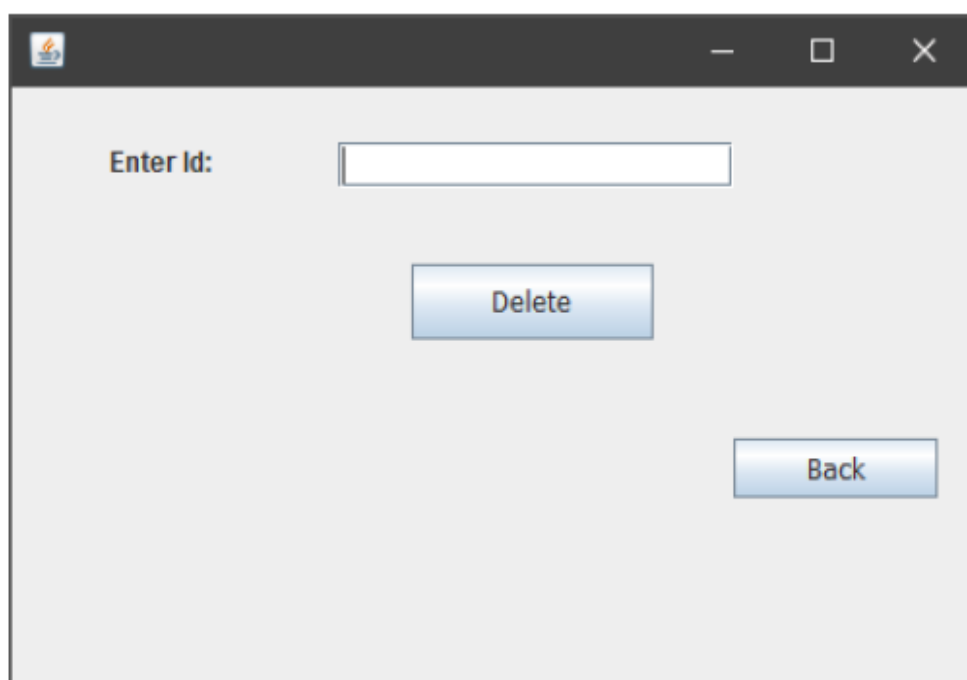


The screenshot shows a window titled "Add Student" with a standard Windows-style title bar (minimize, maximize, close buttons). The form contains six input fields, each with a label to its left: "Name:", "Password:", "Em...", "Address:", "City:", and "Contact No:". Below the input fields are two buttons: "Confirm Student" and "Back".

Field Label	Input Type
Name:	Text
Password:	Text
Em...	Text
Address:	Text
City:	Text
Contact No:	Text

Buttons: Confirm Student, Back

### 5.4. DELETE STUDENT

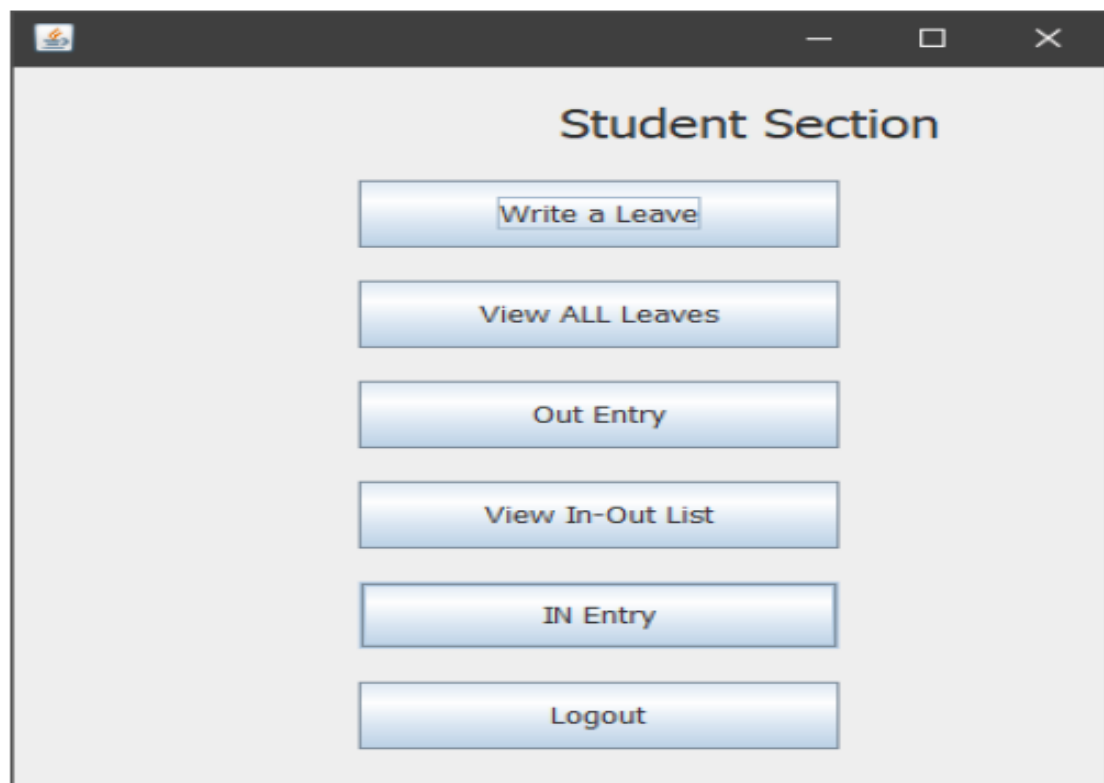


The screenshot shows a window for deleting a student. It features a single input field labeled "Enter Id:". Below the input field are two buttons: "Delete" and "Back".

Field Label	Input Type
Enter Id:	Text

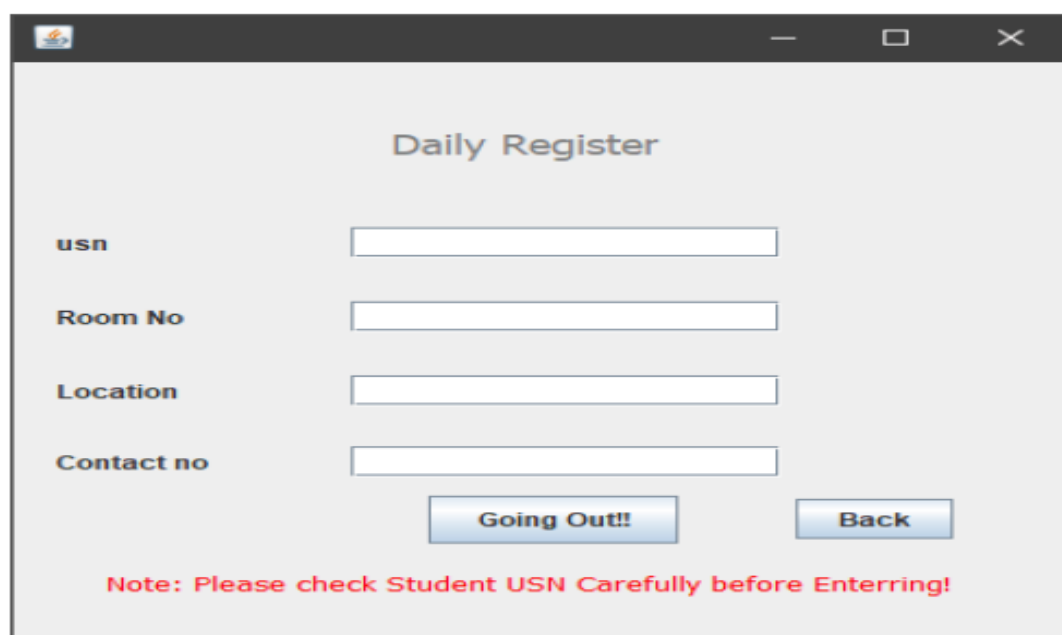
Buttons: Delete, Back

## 5.5. STUDENT



The screenshot shows a window titled "Student Section". Inside the window, there are six blue buttons arranged vertically in the center. The buttons are labeled: "Write a Leave", "View ALL Leaves", "Out Entry", "View In-Out List", "IN Entry", and "Logout". The window has a standard operating system title bar with a minimize button, a maximize button, and a close button.

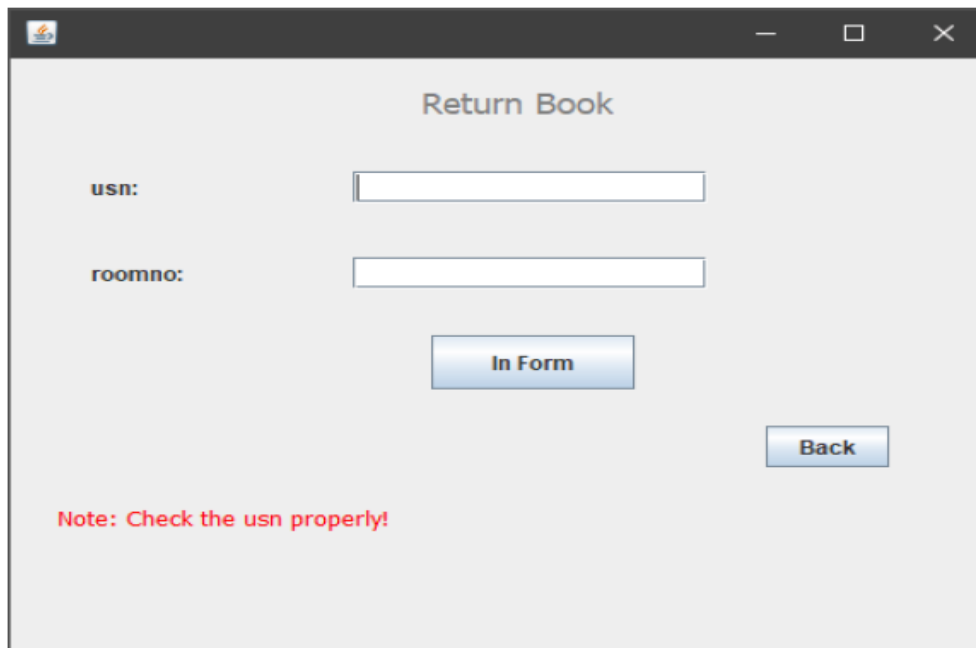
## 5.6. OUT FORM



The screenshot shows a window titled "Daily Register". Inside the window, there are four labels on the left side, each followed by a text input field: "usn", "Room No", "Location", and "Contact no". Below the input fields, there are two blue buttons: "Going Out!!" and "Back". At the bottom of the window, there is a red text note: "Note: Please check Student USN Carefully before Enterring!". The window has a standard operating system title bar with a minimize button, a maximize button, and a close button.



### 5.7. IN FORM



The screenshot shows a window titled "Return Book" with a standard Windows-style title bar (minimize, maximize, close buttons). The form contains two input fields: "usn:" and "roomno:". Below these fields are two buttons: "In Form" and "Back". A red note at the bottom left reads "Note: Check the usn properly!".

Return Book

usn:

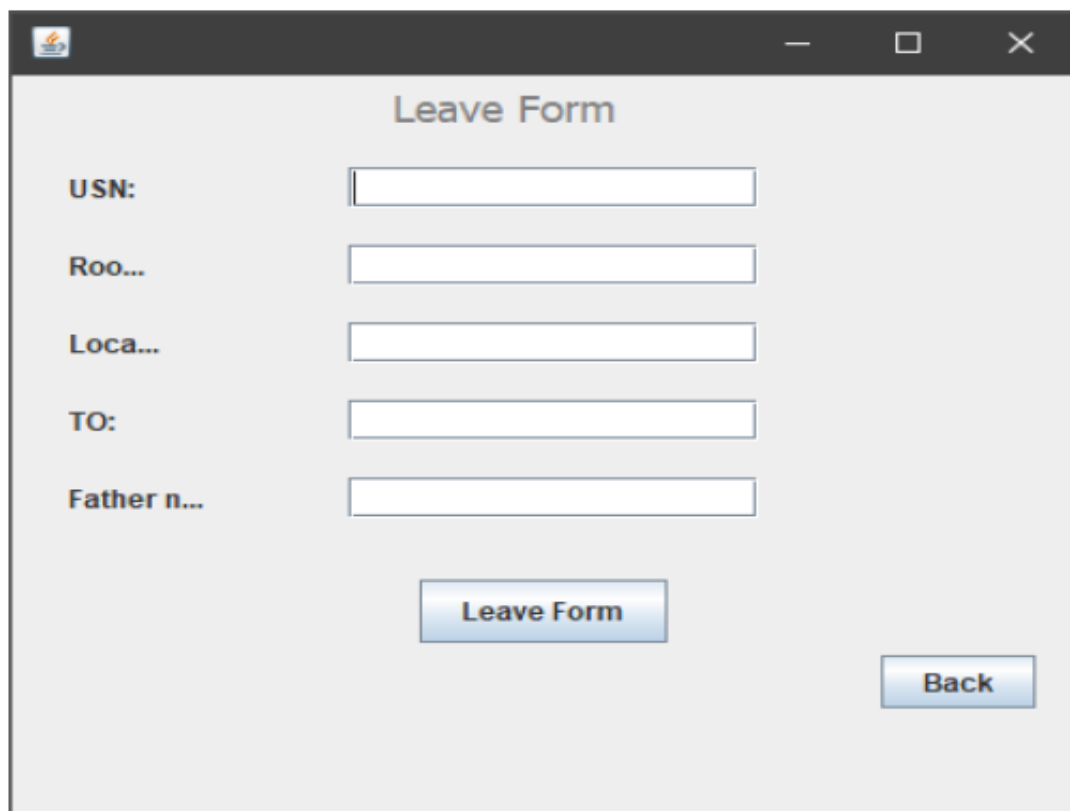
roomno:

In Form

Back

Note: Check the usn properly!

### 5.8. LEAVE



The screenshot shows a window titled "Leave Form" with a standard Windows-style title bar (minimize, maximize, close buttons). The form contains five input fields: "USN:", "Roo...", "Loca...", "TO:", and "Father n...". Below these fields are two buttons: "Leave Form" and "Back".

Leave Form

USN:

Roo...

Loca...

TO:

Father n...

Leave Form

Back

## CHAPTER 6

### CONCLUSION AND FUTURE WORK

To conclude the description about the project the project, developed using ASP.net with JAVA SWING and SQL SERVER is based on the requirement specification of the user and the analysis of the existing system, with flexibility for future enhancement. DIGITAL REGISTRATION FORM SYSTEM is very useful for hostel allotment, hotel booking, and other places where books are used for entering the data. This hostel form software is designed for people who want to manage various activities in the hostel.

For the past few years the numbers of educational institutions are increasing rapidly. Thereby the numbers of hostels are also increasing for the accommodation of the students studying in this institution. And hence there is a lot of strain on the person who are running the hostel and software's are not usually used in this context.

This particular project deals with the problems on managing a hostel and avoids the problems which occur when carried manually. Identification of the drawbacks of the existing system leads to the designing of computerized system that will be compatible to the existing system with the system which is more user friendly and more GUI oriented. Every Industry, School, Office workspace always used some form of register books or entry IN and OUT form, leave form, Attendance sheet etc. Software program which would does all this requirement digitally into the system and store it in proper databases where access will be given to specific login. Which would help us move ahead in technology in daily life and also reduce the use of paper and Hence saving the Environment.

Hence, DRFA (DIGITAL REGISTRATION FORM APPLICATION) is tool which can be used across the world for storing the data of people in database instead of paper or book which would help us reduce the usage of paper and helps mankind to move towards Digitalization and save the reduction of trees and helps in reducing global warming and saves environment.

## CHAPTER 7

### REFERENCES

- James R Groff and Paul N Weinberg, "Complete reference SQL". Second Edition.
- Agnes Molnar and Adam Granicz , " .Net with Visual C# 2010".
- Css files, from W3 schools.
- [http://w3 school.com/JAVA\\_file](http://w3school.com/JAVA_file) , Hostel form, from cusat hostel.
- [http://cusat.ac.in/hostel\\_form](http://cusat.ac.in/hostel_form) SWING files, from Wikipedia, the free encyclopedia.
- [http://www.en.wikipedia.org/wiki/Wireless\\_security](http://www.en.wikipedia.org/wiki/Wireless_security).