

GRAPHICAL REPRESENTATION OF STUDENT PERFORMANCE AND PROJECT REPORTING SYSTEM

*A Real-Time Research Project Report
submitted to JNTU Hyderabad in partial fulfillment
of the requirements for the award of the degree*

BACHELOR OF TECHNOLOGY

In

COMPUTER SCIENCE AND ENGINEERING

Submitted by

JELLA KEERTHANA

22RG1A0589

KURMA SWAROOPA

22RG1A0598

KYASAGALLA SRAVANTHI

22RG1A0599

SAMALA AKSHAYA

22RG1A05B9

Under the Guidance of

Mrs. G.Sujatha

Assistant Professor



DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING
MALLA REDDY COLLEGE OF ENGINEERING FOR WOMEN
An UGC Autonomous Institution

Approved by AICTE New Delhi and Affiliated to JNTUH

An ISO 9001: 2015 Certified Institution

B. Tech. Programs CSE, ECE Accredited by NBA

IIC 5.0 with 4-star Rating; INDIAN RANKINGS 2023 NIRF Innovation Band 151-300

Maisammaguda Medchal (Dist), Hyderabad -500100, Telangana.

JUNE 2024

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING
MALLA REDDY COLLEGE OF ENGINEERING FOR WOMEN
An UGC Autonomous Institution

Approved by AICTE New Delhi and Affiliated to JNTUH

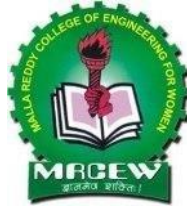
An ISO 9001: 2015 Certified Institution

B. Tech. Programs CSE, ECE Accredited by NBA

IIC 5.0 with 4-star Rating; INDIAN RANKINGS 2023 NIRF Innovation Band 151-300

Maisammaguda , Medchal (Dist), Hyderabad -500100, Telangana.

JUNE 2024



CERTIFICATE

This is to certify that the Real-Time Research Project entitled **“Graphical Representation of Student Performance and project Reporting System”** has been submitted by **JELLA KEERTHANA (22RG1A0589)** , **KURMA SWAROOPA (22RG1A0598)** , **KYASAGALLA SRAVANTHI (22RG1A0599)** , **SAMALA AKSHAYA(22RG1A05B9)** in partial fulfillment of the requirements for the award of **BACHELOR OF TECHNOLOGY** in **COMPUTER SCIENCE & ENGINEERING**. This record of bonafide work carried out by them under my guidance and supervision. The result embodied in this Real-Time Research Project report has not been submitted to any other University or Institute for the award of any degree.

Mrs. G.Sujatha

Assistant professor
Project Guide

Mrs. K. SHEETAL

Head of the Department

External Examiner

ACKNOWLEDGEMENT

The Real-Time Research Project work carried out by our team in the Department of Computer Science and Engineering, Malla Reddy College of Engineering for Women, Hyderabad. ***This work is original and has not been submitted in part or full for any degree or diploma of any other university.***

We wish to acknowledge our sincere thanks to our project guide **Mrs. G.Sujatha**, Assistant Professor, Computer Science & Engineering for formulation of the problem, analysis, guidance and her continuous supervision during the course of work.

We acknowledge our sincere thanks to **Dr.Kanaka Durga Returi**, Principal & Professor of Department of Computer Science and Engineering and **Mrs. Sheetal Kulkarni**, Head of the Department, MRCEW and all our faculties of CSE Department for their kind cooperation in making this Real-Time Research Project work a success.

We extend our gratitude to **Sri. Ch. Malla Reddy**, Founder Chairman and **Sri. Ch. Mahender Reddy**, Secretary, **Dr. Vaka Murali Mohan**, Director for their kind cooperation in providing the infrastructure for completion of our Real-Time Research Project.

We acknowledge our special thanks to the entire teaching faculty and non-teaching staff members of the Computer Science & Engineering Department for their support in making this project work a success.

JELLA KEERTHANA	22RG1A0589
KURMA SWAROOPA	22RG1A0598
KYASAGALLA SRAVANTHI	22RG1A0599
SAMALA AKSHAYA	22RG1A05B9

INDEX

Chapter	Page No.
ABSTRACT	vii
LIST OF FIGURES	viii
1. SYSTEM ANALYSIS	1
1.1 Existing System	1
1.1.1 Disadvantages of Existing System	1
1.2 Proposed System	1
1.2.1 Advantages of Proposed System	2
1.3 Introduction	2
2. LITERATURE SURVEY	3-8
3. SYSTEM DESIGN	9
3.1 System modules	9
3.1.1 Server	9
3.1.2 User	9
3.2 System Architecture	10
3.3 System Requirements	11
3.3.1 Hardware Requirements	11
3.3.2 Software Requirements	11
3.4 UML Diagrams	12
3.4.1 Use Case Diagram	13
3.4.2 Class Diagram	14
3.4.3 Sequence Diagram	15
3.4.4 Flow Chart Diagram	16
3.4.5 Data Flow Diagram	17

3.4.6 ER Diagram	18
3.4.7 Deployment Diagram	19
4.INPUT & OUTPUT DESIGN	20
4.1 Input Design	20
4.2 Output Design	21
5.SOFTWARE ENVIRONMENT	22-30
5.1Java Technology	22
5.1.1Java as Programming Language	22
5.1.2Java Platform	23
5.1.3JAVA Virtual Machine	24
5.1.4 Java Architecture	24
5.1.5 Eclipse Galileo	26
5.1.6 Apache Tomcat	26
5.1.7Servlet	26
5.1.8JDBC	27
5.1.9 JDBC Goals	28
5.2 Features of Java	29-30
6.SYSTEM STUDY	31-32
6.1Economic Feasibility	31
6.2Technical Feasibility	31
6.3Social Feasibility	32
7.SYSTEM TESTING	33-35
7.1 Types of Tests	33
7.1.1 Unit testing	33
7.1.2 Integration testing	34
7.1.3 Functional testing	34
7.1.4System testing	35
7.1.5White Box Testing	35

8.RESULTS	36-47
9.CONCLUSION AND FUTURE ENHANCEMENT	48
10.BIBILOGRAPHY	49

ABSTRACT

The Project entitled "Project Status Information System (PSIS)" deals with the various levels of project development and will account for time used in analysis, design programming, testing and verification etc. Information systems development projects range from one-person projects that take very little time and effort to multiple person,efforts costing millions of dollars. The goal of Project Status Information System (PSIS) is to prevent projects from coming in late and going over budget.

Project Status Information System (PSIS) gives the management a clear picture of the usage of time by various projects i.e. utilized time and unutilized time. Every activity, no matter how small or large, requires use of the commodity called time. There is no substitute for time as there are substitutes for other resources.

As such it is one of the most precious of resources. By analyzing the results provided by the software they might rectify the defects in utilizing time and take remedial actions. Project Status Information System (PSIS) takes time sheet as input. The input may be in non-standard format differing from project to project. Project Status Information System (PSIS) produces output in the form of reports. This output gives a clear picture of the time used at various levels of the project. In Project Status Information System (PSIS) new project information is entered by the technical manager, based on the project information project manager will assign activities to employees who are working under him.

LIST OF FIGURES

Fig. No.	Fig. Name	Page.no
3.1	System Architecture	10
3.4	Use Case Diagram	13
3.5	Class Diagram	14
3.6	Sequence Diagram	15
3.7	Flow Chart Diagram	16
3.8	Data Flow Diagram	17
3.9	ER Diagram	18
3.10	Deployment Diagram	19
8.1	Screenshot of Home Page	36
8.2	Screenshot of Student Registration Page	37
8.3	Screenshot of Student Login Page	38
8.4	Screenshot of Admin Login Page	39
8.5	Screenshot of Admin Page	40
8.6	Screenshot of Faculty Information	41
8.7	Screenshot of Student Information	42
8.8	Screenshot of Faculty Login	43
8.9	Screenshot of Faculty Home	44
8.10	Screenshot of Student project Details	45
8.11	Screenshot of Marks Entering Page	46
8.12	Screenshot of Student Graph	47

CHAPTER-1

SYSTEM ANALYSIS

1.1 EXISTING SYSTEM

- In the existing system, all the student information is added manually and the data is stored in the records.
- So it takes a lot of time and physical effort in searching and adding the information. In the existing system, there is a possibility of losing data and no proper maintenance of data existing system the students are not get their results properly.
- In manual records we can maintain less security and also we can use many files to store the data.

1.1.1 Disadvantages of Existing System

- It requires large data base and memory.
- It takes a lot of time and physical effort .
- In searching and adding the information.
- There is possibility of losing data.

1.2 PROPOSED SYSTEM

- The newly constructed system will maintain all the above details at one place and depends upon the login constraints we are using it allows us to access the details of different phases of the system and all this system is constructed with different modules.
- Project Status Information System takes time sheet as input. The input may be in non-standard format differing from project to project.
- Project Status Information System produces output in the form of reports. This output gives a clear picture of the time used at various levels of the project. In Project Status Information System new project information is entered by the technical manager, based on theninformation project manager will assign activities to employees who are working under him.

1.2.1 Advantages of Proposed System

- It takes less time to add or remove any data. Easy to view the details in single screen.

1.3 INTRODUCTION

- Project Status Information System (PSIS) deals with the various levels of project development and will account for time used in analysis, design, programming, testing and verification etc.
- Project Status Information System (PSIS) gives the management a clear picture of usage of time by projects i.e. utilized time and unutilized time.
- By analyzing the results provided by the software they might rectify the defects in utilizing time and take remedial actions.
- Project Status Information System (PSIS) takes time sheet reports as input.
- The input may be in non-standard format differing from project to project.
- Project Status Information System gives the individual report of project, which contains time used for various tasks, and also gives the picture of the unutilized time.
- In Project Status Information System (PSIS) new project information is entered by the technical manager, based on the project Information project manager will assign activities to employees who are working under him.

CHAPTER-2

LITERATURE SURVEY

Patrice Norton focused on the visual representation of data to aid in decision-making processes, while Herbert Simon, a Nobel laureate, emphasized the importance of visualizing information to enhance cognitive processes. Together, their work laid the foundation for using graphs, charts, and diagrams to present data effectively in educational settings and project management. By incorporating graphical representations into student performance tracking and project reporting systems, educators and project managers can easily interpret and communicate data, leading to more informed decision-making and improved outcomes. Visual representations help in identifying trends, patterns, and areas for improvement, making it easier to track progress and make data-driven decisions.

James A. Senn, as a publisher affiliated with Tata McGraw Hill, likely emphasized the significance of graphical representation in the context of student performance and project reporting systems. In his works, Senn may have discussed how effective design plays a crucial role in creating visually appealing and informative graphs, charts, and diagrams to convey data accurately and efficiently. Senn might have highlighted the importance of thoughtful design principles in creating graphical representations that are easy to interpret and engaging for users. By focusing on design elements such as color, layout, and clarity, educators and project managers can enhance the visual appeal of performance reports and project updates, making the information more accessible and impactful for stakeholders. Incorporating well-designed graphical elements can not only improve the understanding of data but also enhance the overall user experience, leading to more effective communication and decision-making in educational and project management contexts.

Pressman, a well-known author in software engineering, has provided valuable insights into the graphical representation of student performance and project

reporting systems through the lens of system analysis and software engineering. In his works, Pressman likely emphasizes the importance of utilizing graphical representations as a tool for system analysis and design in educational and project management contexts. Pressman may have discussed how software engineering principles, such as requirements analysis, design, implementation, testing, and maintenance, can be applied to develop effective graphical representations that enhance the understanding of student performance data and project progress. By following systematic software engineering processes, educators and project managers can create visually engaging and informative graphs, charts, and diagrams to present data in a clear and concise manner. Through the perspective of software engineering, Pressman may have highlighted the role of graphical representations in improving communication, decision-making, and problem-solving within educational settings and project management environments. By applying software engineering methodologies to the design and implementation of graphical elements, stakeholders can leverage visual representations to convey complex information, track progress, and make informed decisions based on the data presented.

Raghu RamaKrishnan is an expert in the field of database management systems. He might have discussed how graphical representations of student performance and project reporting systems can be facilitated through the use of a database management system (DBMS). In his insights, Raghu RamaKrishnan may have highlighted the role of a DBMS in storing and managing data efficiently to support the generation of graphical representations. By structuring data in a database, educational institutions and project teams can query the database to extract relevant information and present it visually through graphs, charts, and reports. Through the lens of database management systems, Raghu RamaKrishnan may have emphasized the importance of designing database schemas that allow for easy retrieval and analysis of student performance metrics and project data.

Haishuo Gu; Yaqi Guo; Huigen Yang; Peng Chen; Minggang Yao; Jiaqi Hou state that “Nowadays, police are facing with the challenge that crime increases far more rapidly than police force, which generate many problems and cause the police have to seek for support from new technical ways. In this background, the new concept of intelligence-led-policing was proposed. This concept indicated that police should actively identify and analyze the problems by collecting data, such that the potential threatens could be known in early stage and prevention resources would be deployed in more efficient ways. In this project, a typical work big data analysis in policing was introduced. Pickpocketing offence is a kind of frequently occurred crime in public transportation, and identifying offenders is a great challenge for the police. With nearly a billion of metro subway traffic records accessed from Beijing metro line company, the passengers' routine mobility pattern was analyzed and some pickpocketing suspects were detected by defining some specific rules. The results indicated that significant traveling patterns exist for routine passengers, and some exceptional individuals who has the unique pattern which is quite different from routine passengers are discovered. However, the shortage of the work still exists as the identity of cardholders is anonymous and it cannot be confirmed, but it still provides an example of big data analysis in public security, which might inspire police for further work”.

Cindy Verleysen states that “This theoretical paper is published by the EUCPN Secretariat in connection with the theme of the Maltese Presidency. The Maltese Presidency, in line with the Trio’s subject of Organised Crime, focused on two topics, namely domestic burglaries and pickpocketing, committed by mobile organised crime groups. For the Theoretical project and Toolbox, the Maltese Presidency proposed the EUCPN Secretariat to concentrate on the phenomenon ‘pickpocketing’. This Theoretical Project gives more information about ‘organised property crime’, whereof pickpocketing is just one phenomenon. Each year, thousands of citizens and especially tourists become victims of pickpockets. Luckily there are many things you can do to prevent yourself from being pickpocketed. Therefore, this paper provides a brief overview with prevention tips. Furthermore, since it is assumed that itinerant crime groups

are responsible for a significant part of the committed pickpocketing, a notable part of this paper focusses on itinerant crime groups in general. Mobile organised crime groups commit a wide variety of crimes, including pickpocketing, and are active in many crime areas. They move quickly around within and across multiple jurisdictions, which makes it tough and difficult to identify and tackle them. A cooperation within and outside the EU is important to identify and dismantle the networks behind these mobile organised crime groups.”

YAN Mi-qiao, GUO Zhong-yang, REN Zhe-hao [10] states that “This paper introduced the spatio-temporal association rules based on clustering mining to find out the spatio-temporal crime patterns of bus pickpocketing. It can be carried out through three steps. Firstly, extract time, places and other information from the case information by text extraction. Then, confirm the boarding stations and getting off stations of victims using the geocoding service and POI search capability of Amap API. Divide the bus routes into sections according to the bus stops and merge the crime time into time interval. Thirdly, the analysis of association rules based on clustering is carried out to discover the patterns of bus pickpocketing. The results prove that the proposed mining model has the following characteristics: This method can reduce the database scanning times, the candidate item sets amount and improve time efficiency of the searching. After clustering, the data in a cluster is similar and the characteristics are more obvious. On this basis, the association rules of high confidence are extracted. When the analysis was carried out, the temporal and spatial characteristics of the bus pickpocketing crime were also considered.”

Peter K. Andersson states that “The history of walking in the city has long been neglected, and the existing scholarship is largely concerned with rioting, flânerie or urban geography. This article aims to detect the behavioural patterns of pedestrian traffic in the late nineteenth century through a close study of the methods of pickpockets in London streets, with information gleaned from trial reports and writings on pickpockets. the most common ways in which pickpockets operated, as described in numerous accounts, we can see how they adapted to nineteenth- century pedestrian norms, and through this method

acquire a rough outline of what pedestrian traffic looked like, and thus how urban dwellers living in a critical historical period adapted and reacted to urban conditions on an everyday level. The evidence shows that pedestrian traffic through the century remained highly interactive, and that the modern aspects of cities identified in theories of civilizing or impoverishment of the public realm had a very limited impact at this time.”

Henri Bouma, Jan Baan, Gertjan J. Burghouts, Pieter T. Eendebak, Jasper R. van Huis, Judith Dijk, Jeroen H. C. van Rest states that “Proactive detection of incidents is required to decrease the cost of security incidents. This paper focusses on the automatic early detection of suspicious behavior of pickpockets with track-based features in a crowded shopping mall. Our method consists of several steps: pedestrian tracking, feature computation and pickpocket recognition. This is challenging because the environment is crowded, people move freely through areas which cannot be covered by a single camera, because the actual snatch is a subtle action, and because collaboration is complex social behavior. We carried out an experiment with more than 20 validated pickpocket incidents. We used a top- down approach to translate expert knowledge in features and rules, and a bottom-up approach to learn discriminating patterns with a classifier. The classifier was used to separate the pickpockets from normal passers-by who are shopping in the mall. We performed a cross validation to train and evaluate our system. In this paper, we describe our method, identify the most valuable features, and analyze the results that were obtained in the experiment. We estimate the quality of these features and the performance of automatic detection of (collaborating) pickpockets. The results show that many of the pickpockets can be detected at a low false alarm rate.”

Xia Zhao, Yong Zhang, Hao Liu, Shaofan Wang, Zhen Qian, Yongli Hu, Baocai Yin [16] states that “Detecting pickpocketing gangs on buses is critical for safety and public security department. Knowing this both in real time and from historical records would allow effective law enforcement and crime prevention. However, very little research has been devoted into identifying pickpocketing gangs in an automated and holistic manner. This research utilizes smart card

data generated by bus riders to identify pickpocketing gangs, who possess distinct characteristics from regular passenger. Particularly, we create a dataset of 1,098 pickpockets among 4.06 million bus riders in August, 2015 in Beijing automatically and efficiently based on an efficient labeling model of outliers. This model examines anomaly of passengers using the so-called relative outlier cluster factor and local outlier factor. The proposed mobility patterns of pickpockets are then learned based on supervised classification. Pickpockets from the derived dataset form a pickpocketing network, which is modeled as a graph with vertices denoted as discrete pickpockets, and edge weight quantified by a combined similarity on mobility pattern, space and time. A graphbased Louvain algorithm is adopted to detect pickpocketing gangs. Experiments are conducted on SINA microblog data to verify the detected pickpocketing gangs identified by the proposed framework. Results show that the framework detects 63 pickpocketing gangs and verifies 34 gangs by microblogs, with recall value 0.85. Findings from this research can assist police and public safety departments in the city in taking pro-active actions to track down pickpocketing gangs.”

Hao Zhang, Ole Lund, Morten Nielsen “Motivation: Receptor–ligand interactions play an important role in controlling many biological systems. One prominent example is the binding of peptides to the major histocompatibility complex (MHC) molecules controlling the onset of cellular immune responses. Thousands of MHC allelic versions exist, making determination of the binding specificity for each variant experimentally infeasible. Here, we present a method that can extrapolate from variants with known binding specificity to those where no experimental data are available.”

CHAPTER-3

SYSTEM DESIGN

3.1 System Modules

3.1.1 Server:

Admin can get login by entering valid username and password. Admin can view all the details of the users and need to authorize them. Admin can upload projects by giving all the details of the project. Admin can add all the details of student's attendance and marks. Admin can view added students and the progress in a graph.

3.1.2 User:

The student needs to enter all the details in the registration form. The student should be authorized by admin to get login into the application. A student can search projects based on the title, year and Guide name. Students can view all the details of his/her attendance and marks. Students can view graph for his attendance and marks.

Modules:

- Admin
- Faculty
- Student

Module description:

- **ADMIN:** Admin can get login by entering valid username and password. Admin can view all the details of the users and need to authorize them. Admin can upload projects by giving all the details of the project. Admin can add all the details of student's attendance and marks. Admin can view added students and the progress in a graph.
- **FACULTY:** Faculty can get the login by entering their valid username & password. He can collect all the project of the project from the students.
- **STUDENT:** The student needs to enter all the details in the registration form. The student should be authorized by admin to get login into the application.

3.2 System Architecture

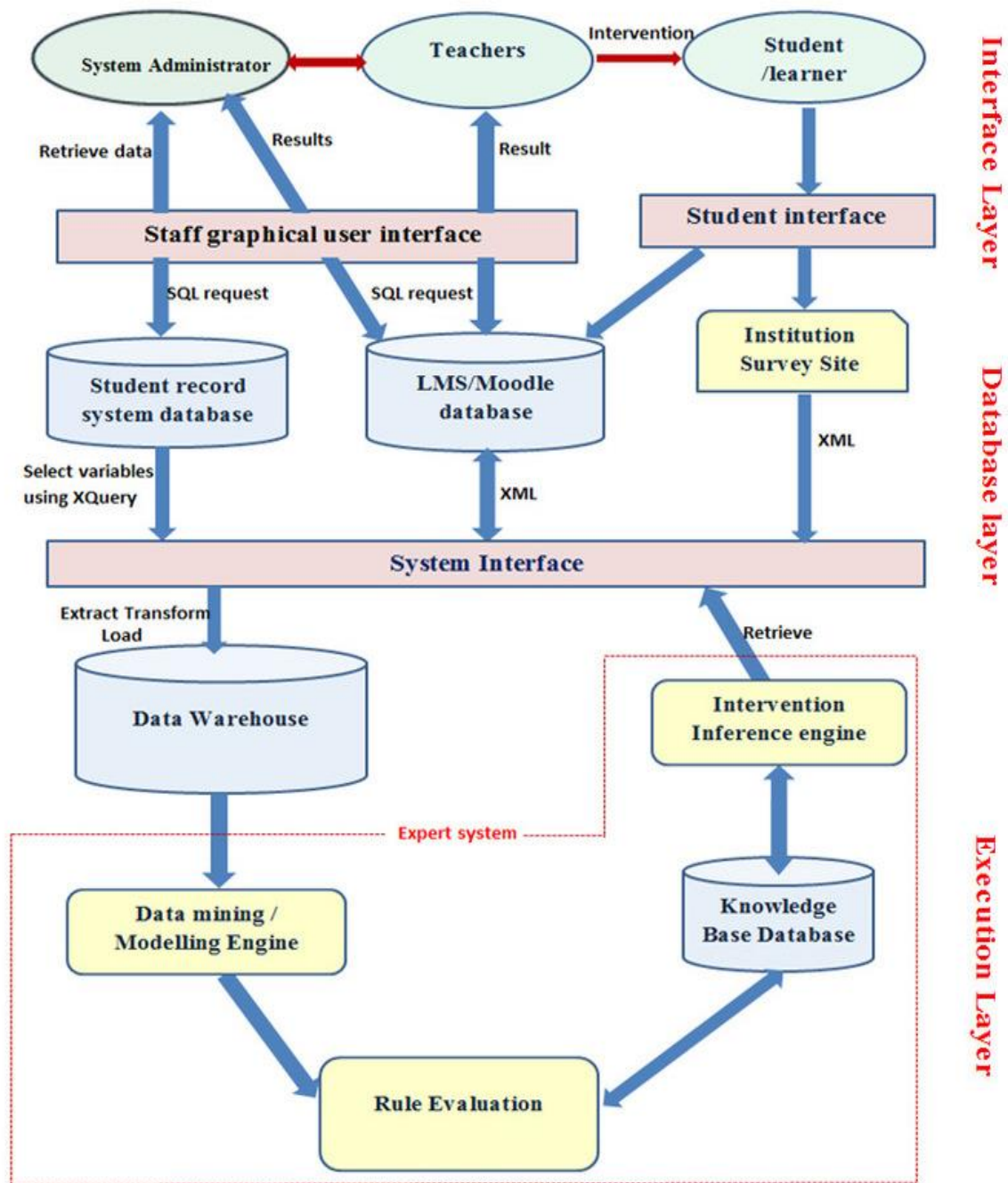


Fig:3.2 System Architecture

3.3 System Requirements

3.3.1 Hardware Requirements

- System : intel i3 2.0 GHz.
- Hard Disk : 1 TB.
- Floppy Drive : 1.44 Mb.
- Monitor : 15 VGColour.
- Mouse : Logitech.
- Ram : 4 GB.

3.3.2 Software Requirements

- Operating system: Windows XP.
- Coding Language : J2EE
- Data Base : MYSQL

3.4 UML Diagrams

UML stands for Unified Modeling Language. UML is a standardized general-purpose modeling language in the field of object-oriented software engineering. The standard is managed, and was created by, the Object Management Group. The goal is for UML to become a common language for creating models of object oriented computer software. In its current form UML is comprised of two major components: a Meta-model and a notation. In the future, some form of method or process may also be added to; or associated with, UML. The Unified Modeling Language is a standard language for specifying, Visualization, Constructing and documenting the artifacts of software system, as well as for business modeling and other non-software systems. The UML represents a collection of best engineering practices that have proven successful in the modeling of large and complex systems.

The UML is a very important part of developing objects oriented software and the software development process. The UML uses mostly graphical notations to express the design of software projects. The Primary goals in the design of the UML are as follows: Provide users a ready-to-use, expressive visual modeling Language so that they can develop and exchange meaningful models. Provide extendibility and specialization mechanisms to extend the core concepts. Be independent of particular programming languages and development process. Provide a formal basis for understanding the modeling language.

3.4.1 Use Case Diagram

A use case diagram in the Unified Modeling Language (UML) is a type of behavioral diagram defined by and created from a Use-case analysis. Its purpose is to present a graphical overview of the functionality provided by a system in terms of actors, their goals (represented as use cases), and any dependencies between those use cases. The main purpose of a use case diagram is to show what system functions are performed for which actor. Roles of the actors in the system can be depicted.

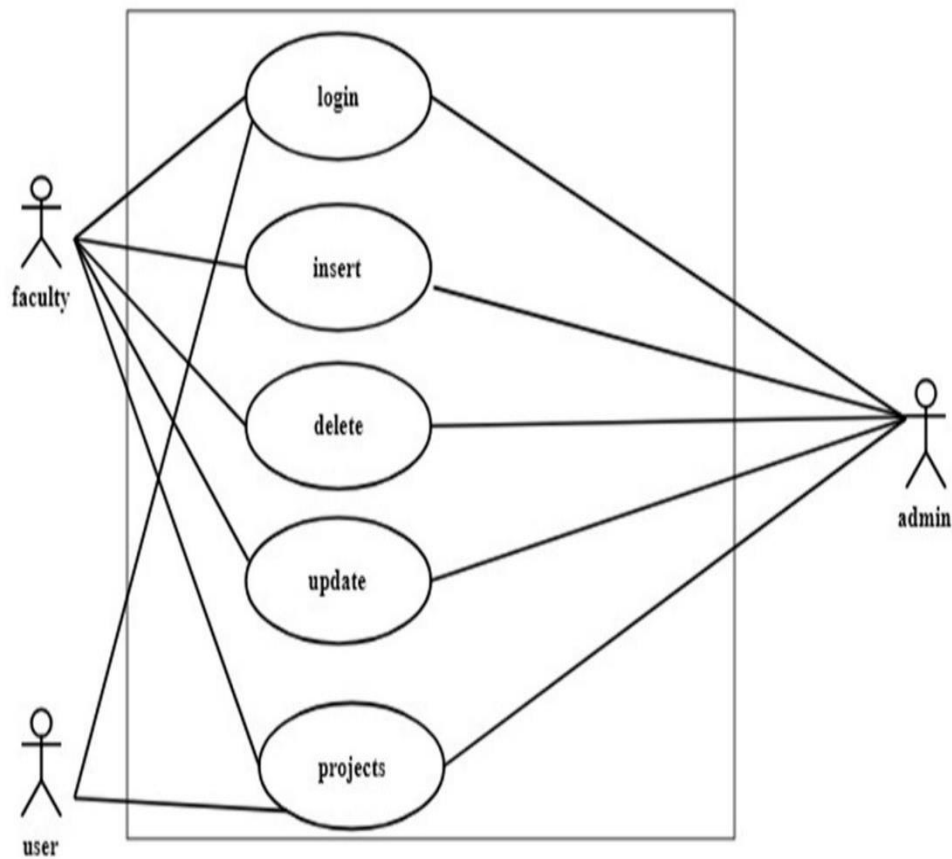


Fig:3.4 Use Case Diagram

3.4.2 Class Diagram

In software engineering, a class diagram in the Unified Modeling Language (UML) is a type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes, operations (or methods), and the relationships among the classes. It explains which class contains information.

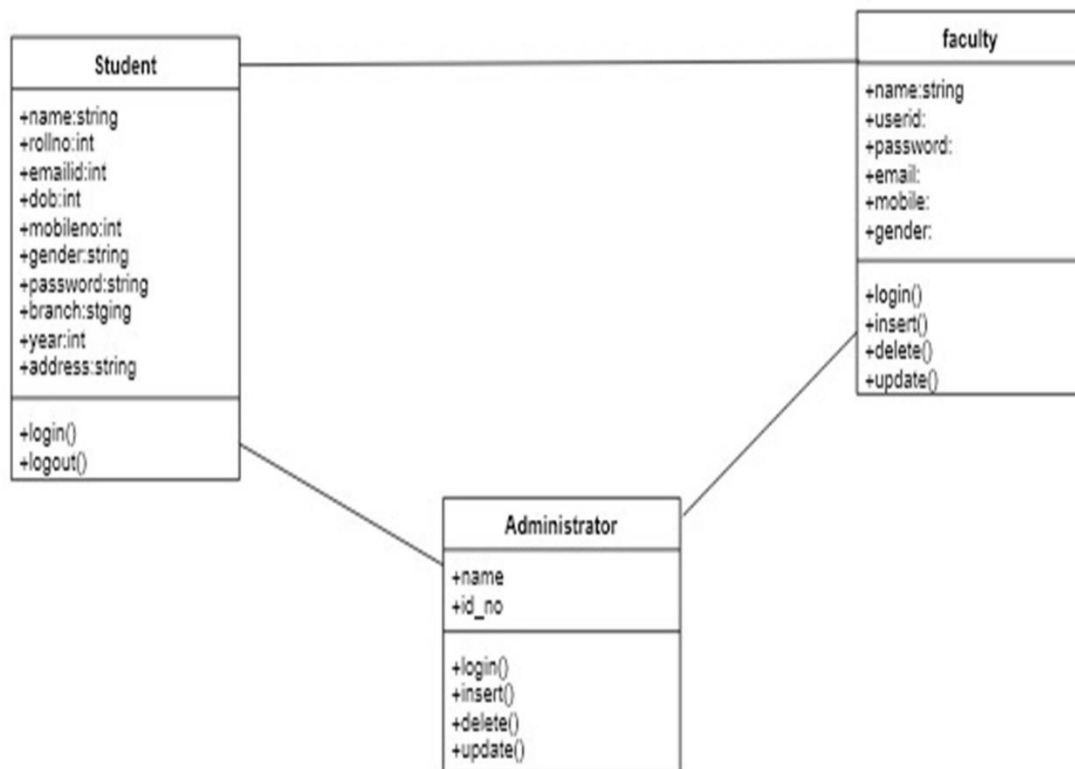


Fig:3.5 Class Diagram

3.4.3 Sequence Diagram

A sequence diagram in Unified Modeling Language (UML) is a kind of interaction diagram that shows how processes operate with one another in what order. It is a construct of a Message Sequence Chart. Sequence diagrams are sometimes called event diagrams, event scenarios, and timing diagrams.

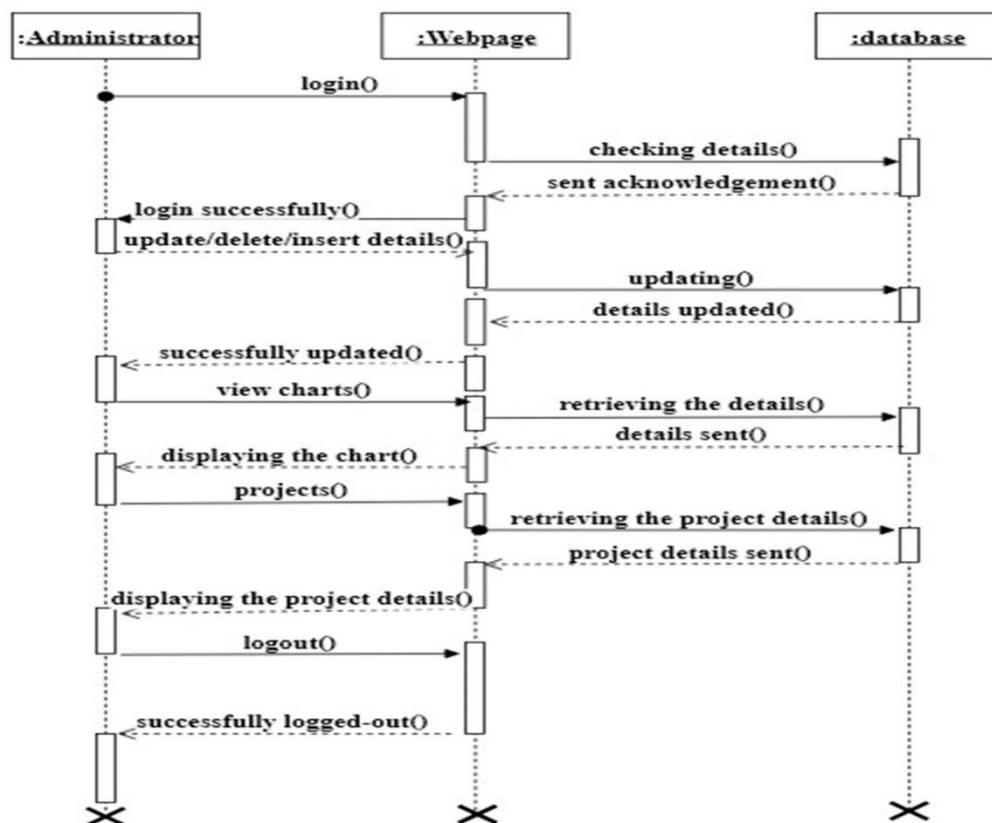


Fig:3.6 Sequence Diagram

3.4.4 Flow Chart Diagram

Flow Chart diagrams are graphical representations of workflows of stepwise activities and actions with support for choice, iteration and concurrency. In the Unified Modeling Language, activity diagrams can be used to describe the business and operational step-by-step workflows of components in a system. An activity diagram shows the overall flow of control

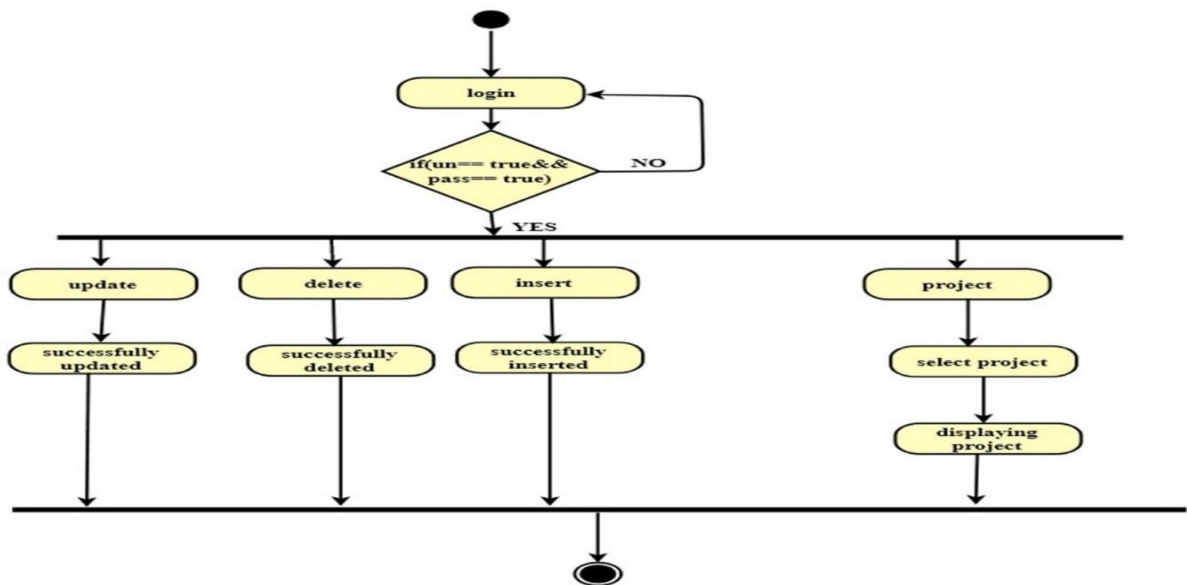


Fig:3.7 Flow Chart Diagram

3.4.5 Data Flow Diagram

DFD is also called as bubble chart. It is a simple graphical formalism that can be used to represent a system in terms of input data to the system, various processing carried out on this data, and the output data is generated by this system. The data flow diagram (DFD) is one of the most important modeling tools. It is used to model the system components. These components are the system process, the data used by the process, an external entity that interacts with the system and the information flows in the system. DFD shows how the information moves through the system and how it is modified by a series of transformations. It is a graphical technique that depicts information flow and the transformations that are applied as data moves from input to output. DFD is also known as bubble chart. A DFD may be used to represent a system at any level of abstraction. DFD may be partitioned into levels that represent increasing information flow and functional detail.

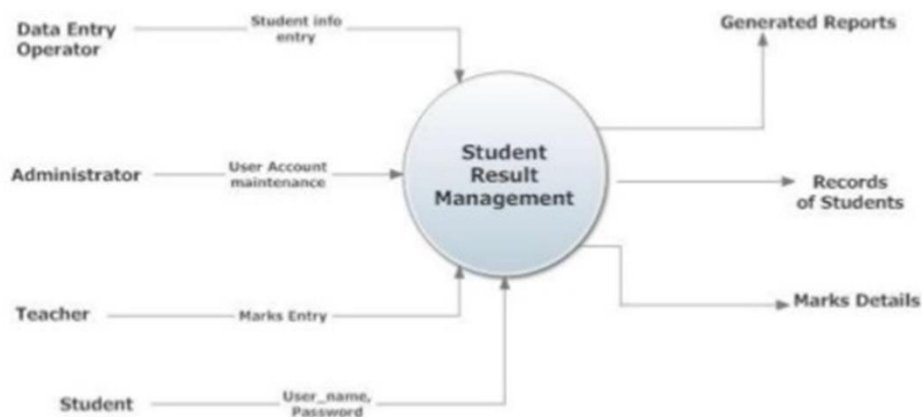


Fig:3.8 Data Flow Diagram

3.4.6 ER Diagram:

An entity relationship diagram (ERD) shows the relationships of entity sets stored in a database. An entity in this context is an object, a component of data. An entity set is a collection of similar entities. These entities can have attributes that define its properties.

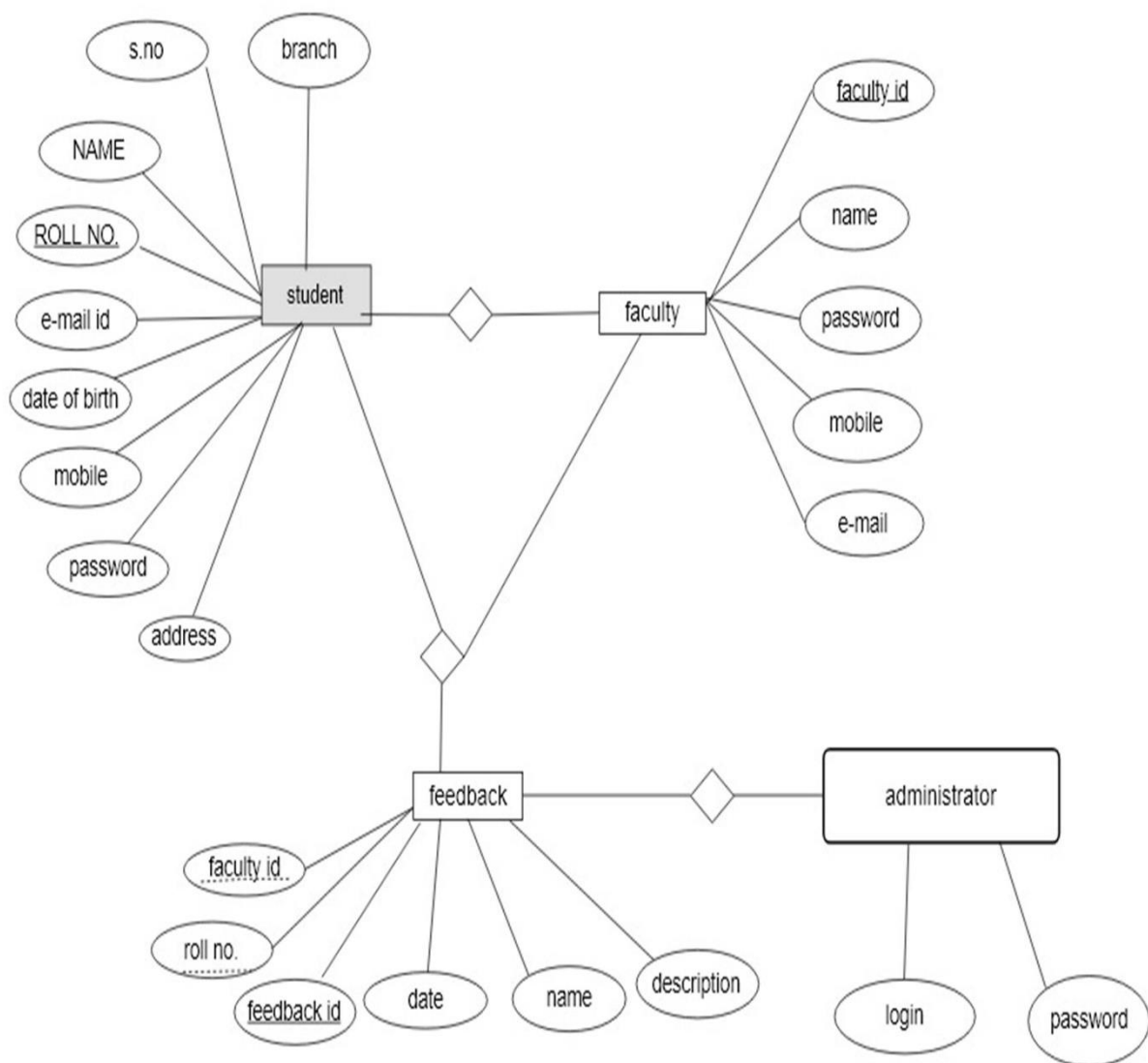


Fig:3.9 ER Diagram

3.4.7 DEPLOYMENT DIAGRAM:

A deployment diagram is a UML diagram type that shows the execution architecture of a system, including nodes such as hardware or software execution environments, and the middleware connecting them. Deployment diagrams are typically used to visualize the physical hardware and software of a system.

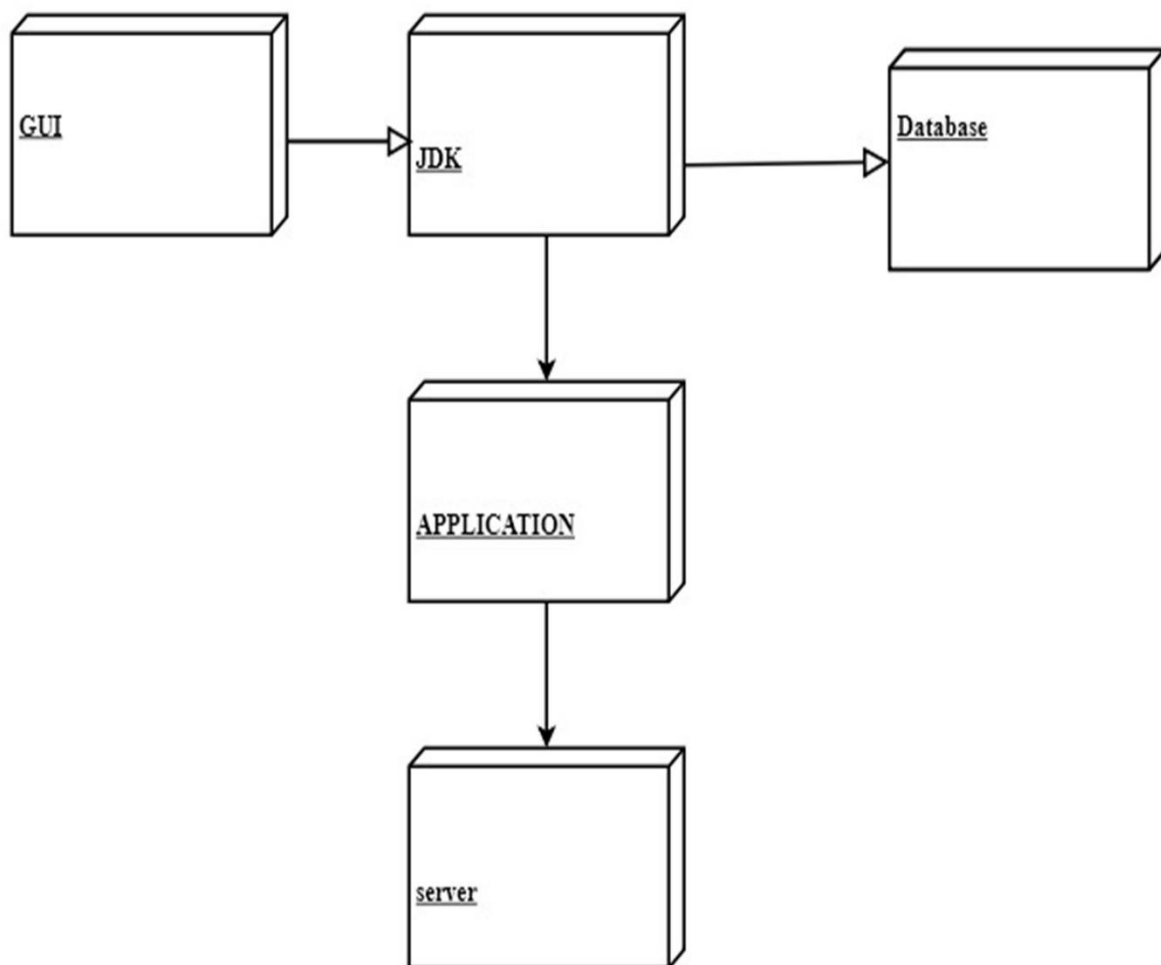


Fig 3.10: Deployment Diagram

CHAPTER-4

INPUT AND OUTPUT DESIGN

4.1 INPUT DESIGN

Input Design plays a vital role in the life cycle of software development, it requires very careful attention of developers. The input design is to feed data to the application as accurate as possible. So inputs are supposed to be designed effectively so that the errors occurring while feeding are minimized. According to Software Engineering Concepts, the input forms or screens are designed to provide to have a validation control over the input limit, range and other related validations.

This system has input screens in almost all the modules. Error messages are developed to alert the user whenever he commits some mistakes and guides him in the right way so that invalid entries are not made. Let us see deeply about this under module design.

Input design is the process of converting the user created input into a computer-based format. The goal of the input design is to make the data entry logical and free from errors. The error in the input are controlled by the input design. The application has been developed in user-friendly manner. The forms have been designed in such a way during the processing the cursor is placed in the position where must be entered. The user is also provided with in an option to select an appropriate input from various alternatives related to the field in certain cases.

Validations are required for each data entered. Whenever a user enters an erroneous data, error message is displayed and the user can move on to the subsequent pages after completing all the entries in the current page.

4.2 OUTPUT DESIGN

The Output from the computer is required to mainly create an efficient method of communication within the company primarily among the project leader and his team members, in other words, the administrator and the clients. The output of VPN is the system which allows the project leader to manage his clients in terms of creating new clients and assigning new projects to them, maintaining a record of the project validity and providing folder level access to each client on the user side depending on the projects allotted to him. After completion of a project, a new project may be assigned to the client. User authentication procedures are maintained at the initial stages itself.

A new user may be created by the administrator himself or a user can himself register as a new user but the task of assigning projects and validating a new user, rests with the administrator only. The application starts running when it is executed for the first time. The server has to be started and then the internet explorer is used as the browser. The project will run on the local area network so the server machine will serve as the administrator while the other connected systems can act as the clients. The developed system is highly user friendly and can be easily understood by anyone using it even for the first time.

CHAPTER-5

SOFTWARE ENVIRONMENT

5.1 JAVA TECHNOLOGY

Java technology is both a programming language and a platform.

5.1.1 The Java Programming Language

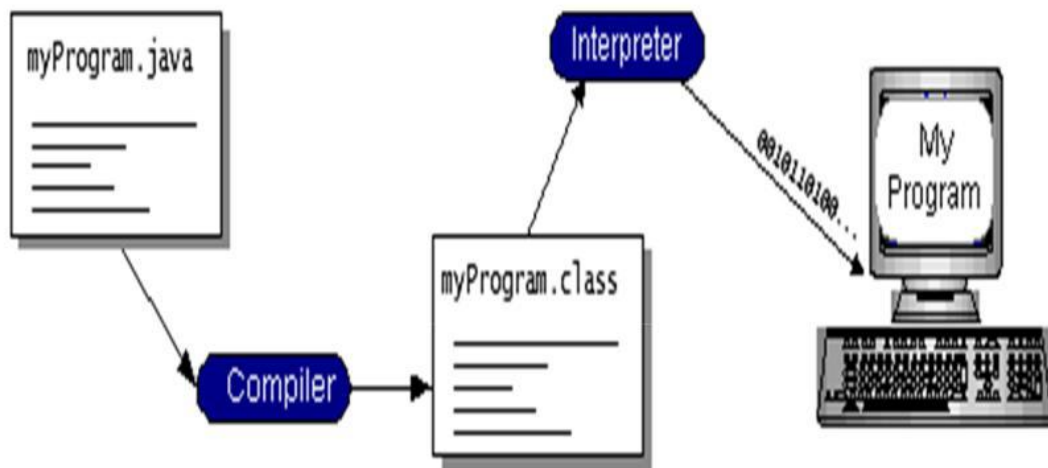
The Java programming language is a high-level language that can be characterized by all of the following buzzwords:

- Simple
- Architecture neutral
- Object oriented
- Portable
- Distributed
- High performance
- Interpreted
- Multithreaded
- Robust
- Dynamic
- Secure

With most programming languages, you either compile or interpret a program so that you can run it on your computer. The Java programming language is unusual in that a program is both compiled and interpreted. With the compiler, first you translate a program into an intermediate language called Java byte codes —the platform-independent codes interpreted by the interpreter on the Java platform. The interpreter parses and runs each Java byte code instruction on the computer. Compilation happens just once; interpretation occurs each time the program is executed.

They can think of Java byte codes as the machine code instructions for the Java Virtual Machine (Java VM). Every Java interpreter, whether it's a

development tool or a Web browser that can run applets, is an implementation of the Java VM. Java byte codes help make “write once, run anywhere” possible. You can compile your program into byte codes on any platform that has a Java compiler.



5.1.2 Java Platform

A platform is the hardware or software environment in which a program runs. It's already mentioned that some of the most popular platforms like Windows 2000, Linux, Solaris, and MacOS. Most platforms can be described as a combination of the operating system and hardware. The Java platform differs from most other platforms in that it's a software-only platform that runs on top of other hardware-based platforms.

The Java platform has two components:

- The Java Virtual Machine (Java VM)
- The Java Application Programming Interface (Java API)

They've already been introduced to the Java VM. It's the base for the Java platform and is ported onto various hardware-based platforms.

The Java API is a large collection of ready-made software components that provide many useful capabilities, such as graphical user interface (GUI) widgets. The Java API is grouped into libraries of related classes and

interfaces; these libraries are known as packages.

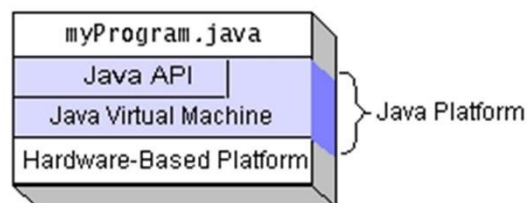
Native code is code that after you compile it, the compiled code runs on a specific hardware platform. As a platform-independent environment, the Java platform can be a bit slower than native code. However, smart compilers, well-tuned interpreters, and just-in-time byte code compilers can bring performance close to that of native code without threatening portability.

5.1.3 Java Virtual Machine (JVM):

Java is a component which provides an environment for running Java programs. JVM interprets the byte code into machine code which will be executed the machine in which the Java program runs.

5.1.4 Java architecture:

Java combines both the approaches of compilation and interpretation. First, java compiler compiles the source code into byte code. At the run time, Java Virtual Machine (JVM) interprets this byte code and generates machine code which will be directly executed by the machine in which java program runs. So java is both compiled and interpreted language.



The key that allows the Java to solve the security and portability problems is that the output of Java compiler is Byte code. Byte code is a highly optimized set of instructions designed to be executed by the Java run-time system, which is called the Java Virtual Machine (JVM). That is, in its standard form, the JVM is an interpreter for byte code.

Translating a Java program into byte code helps makes it much easier to run a program in a wide variety of environments. The reason is, once the run-time package exists for a given system, any Java program can run on it. Although Java was designed for interpretation, there is technically nothing about Java that prevents on-the-fly compilation of byte code into native

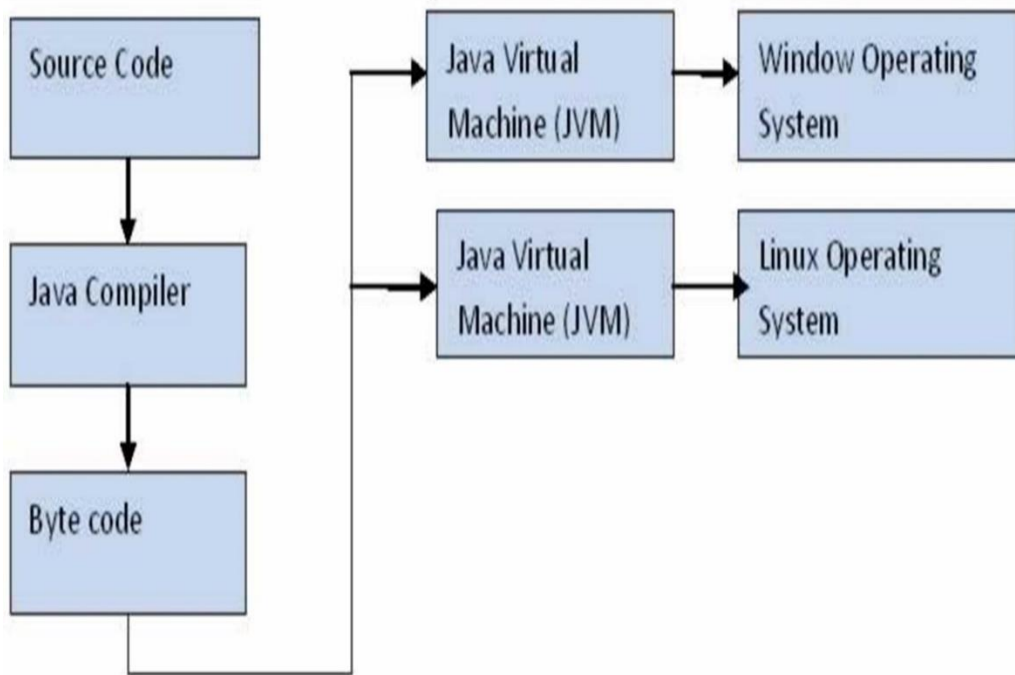
Java is platform independent:

Platform independence is one of the main advantages of Java. In another words, java is portable because the same java program can be executed in multiple platforms without making any changes in the source code. You just need to write the java code for one platform and the same program will run in any platforms.

Let's make it more clearly with the help of the following diagram. Here the same compiled Java byte code is interpreted by two different JVMs to make it run in Windows and Linux platforms.

The multi-platform environment of the Web places extraordinary demands on a program, because the program must execute reliably in a variety of systems. The ability to create robust programs was given a high priority in the design of Java. Java is strictly typed language; it checks your code at compile time and run time.

Java virtually eliminates the problems of memory management and deallocation, which is completely automatic. In a well-written Java program, all run time errors can and should be managed by your program.



5.1.5ECLIPSE GALILEO:

Eclipse is a commercially available Java EE and AJAX IDE created and maintained by the company Genuitec, a founding member of the Eclipse Foundation. Eclipse is a multi-language software development environment comprising an integrated development environment (IDE) and an extensible plug-in system. It is written mostly in java and can be used to develop applications in Java. Eclipse employs plug-ins in order to provide all of its functionality on top of (and including) the runtime system, in contrast to some other applications where functionality is typically hard coded. The runtime system of Eclipse is based on equinox an OSGi standard compliant implementation. This plug-in mechanism is a lightweight software componentry framework. In addition to allowing Eclipse to be extended using other programming languages such as c and python, the plug-in framework allows Eclipse to work with typesetting languages like LaTeX[2] networking applications such as telnet and database management system.

5.1.6APACHE TOMCAT 9.0:

Apache Tomcat (or Jakarta Tomcat or simply Tomcat) is an open source servlet container developed by the Apache Software Foundation(ASF). Tomcat implements the Java Servlet and the Java Server Pages(JSP) specifications from Sun Microsystems, and provides a “pure Java” HTTP web server environment for Java code to run.

5.1.7SERVLET:

Servlet is what you get when you cross an applet with a CGI script. From a programmatic point of view, a servlet resembles an applet; it is an executable that's written in the Java programming language, and it usually (though not always) is executed in response to an invocation from an HTML page. A user requests some information by filling out a form containing a link to a servlet and clicking the Submit button. The server locates the requested servlet. The servlet then gathers the information needed to satisfy the user's request and

constructs a Web page containing the information. That Web page is then displayed on the user's browser.

5.1.8JDBC

JDBC stands for "Java Data Base Connectivity". It is an API (Application Programming Interface) which consists of a set of Java classes, interfaces and exceptions and a specification to which both JDBC driver vendors and JDBC developers adhere when developing applications. JDBC is a very popular data access standard. RDBMS (Relational Database Management Systems) or third-party vendors develop drivers which adhere to the JDBC specification. Other developers use these drivers to develop applications which access those databases e.g. you'll use Connector J JDBC driver to access My SQL database. Since the drivers adhered to JDBC specification, the JDBC application developers can replace one driver for their application with another better one without having to rewrite their application. If they had used some proprietary API provided by some RDBMS vendor, they will not have been able to change the driver and/or database without having to rewrite the complete application.

5.1.9JDBC Goals

Few software packages are designed without goals in mind. JDBC is one that, because of its many goals, drove the development of the API. These goals, in conjunction with early reviewer feedback, have finalized the JDBC class library into a solid framework for building database applications in Java. The goals that were set for JDBC are important. They will give you some insight as to why certain classes and functionalities behave the way they do. The eight design goals for JDBC are as follows: The designers felt that their main goal was to define a SQL interface for Java. Although not the lowest database interface level possible, it is at a low enough level for higher-level tools and APIs to be created. Conversely, it is at a high enough level for application programmers to use it confidently. Attaining this goal allows for future tool vendors to "generate" JDBC code and to hide many of SQL syntax varies as you move from database vendor to database vendor. In an effort to support a wide variety of vendors, JDBC will allow any query statement to be passed through it to the underlying database driver. This allows the connectivity module to handle non-standard

functionality in a manner that is suitable for its user. The JDBC SQL API must “sit” on top of other common SQL level APIs. This goal allows JDBC to use existing ODBC level drivers by the use of a software interface. This interface would translate JDBC calls to ODBC and vice versa. Java’s acceptance in the user community thus far, the designers feel that they should not stray from the current design of the core Java system. This goal probably appears in all software design goal listings. JDBC is no exception. Sun felt that the design of JDBC should be very simple, allowing for only one method of completing a task per mechanism. Allowing duplicate functionality only serves to confuse the users of the API. Because more often than not, the usual SQL calls used by the programmer are simple SELECT’s, INSERT’s, DELETE’s and UPDATE’s, these queries should be simple to perform with JDBC. However, more complex SQL statements should also be possible

Advantages of JDBC Technology:

With JDBC technology, businesses are not locked in any proprietary architecture, and can continue to use their installed databases and access information easily -- even if it is stored on different database management systems. The combination of the Java API and the JDBC API makes application development easy and economical. JDBC hides the complexity of many data access tasks, doing most of the “heavy lifting” for the programmer behind the scenes. The JDBC API is simple to learn, easy to deploy, and inexpensive to maintain. With the JDBC API, no configuration is required on the client side. With a driver written in the Java programming language, all the information needed to make a connection is completely defined by the JDBC URL or by a Data Source object registered with a Java Naming and Directory Interface (JNDI) naming service. Zero configuration for clients supports the network computing paradigm and centralizes software maintenance. In an effort to set an independent database standard API for Java; Sun Microsystems developed Java Database Connectivity, or JDBC. JDBC offers a generic SQL database access mechanism that provides a consistent interface to a variety of RDBMSs. This consistent interface is achieved through the use of “plug-in” database connectivity modules, or *drivers*. If a database vendor wishes to have JDBC

support, he or she must provide the driver for each platform that the database and Java run on.

To gain a wider acceptance of JDBC, Sun based JDBC's framework on ODBC. As you discovered earlier in this chapter, ODBC has widespread support on a variety of platforms. Basing JDBC on ODBC will allow vendors to bring JDBC drivers to market much faster than developing a completely new connectivity solution.

JDBC was announced in March of 1996. It was released for a 90 day public review that ended June 8, 1996.

5.2 Features:

Every time you that you download a "normal" program; you are risking a viral infection. Prior to Java, most users did not download executable programs frequently, and those who did scan them for viruses prior to execution. Most users still worried about the possibility of infecting their systems with a virus. In addition, another type of malicious program exists that must be guarded against. This type of program can gather private information, such as credit card numbers, bank account balances, and passwords. Java answers both these concerns by providing a "firewall" between a network application and your computer.

Portability:

For programs to be dynamically downloaded to all the various types of platforms connected to the Internet, some means of generating portable executable code is needed .As you will see, the same mechanism that helps ensure security also helps create portability. Indeed, Java's solution to these two problems is both elegant and efficient.

The byte code:

The key that allows the Java to solve the security and portability problems is that the output of Java compiler is Byte code. Byte code is a highly optimized set of instructions designed to be executed by the Java run-time system, which is called the Java Virtual Machine (JVM). That is, in its standard form, the JVM is an interpreter for byte code.

Translating a Java program into byte code helps makes it much easier to run a program in a wide variety of environments. The reason is, once the run-time package exists for a given system, any Java program can run on it. Although Java was designed for interpretation, there is technically nothing about Java that prevents on-the-fly compilation of byte code into native code. Sun has just completed its Just In Time (JIT) compiler for byte code. When the JIT compiler is a part WEof JVM, it compiles byte code into executable code in real time, on a piece-by-piece, demand basis. It is not possible to compile an entire Java program into executable code all at once, because Java performs various run-time checks that can be done only at run time. The JIT compiles code, as it is needed, during execution.

Java is platform independent:

Platform independence is one of the main advantages of Java. In another words, java is portable because the same java program can be executed in multiple platforms without making any changes in the source code. You just need to write the java code for one platform and the same program will run in any platforms

Object-Oriented:

Java was not designed to be source-code compatible with any other language. This allowed the Java team the freedom to design with a blank slate. One outcome of this was a clean usable, pragmatic approach to objects. The object model in Java is simple and easy to extend, while simple types, such as integers, are kept as high-performance non-objects.

CHAPTER-6

SYSTEM STUDY

6.1 FEASIBILITY STUDY

The feasibility of the project is analyzed in this phase and business proposal is put forth with a very general plan for the project and some cost estimates. During system analysis the feasibility study of the proposed system is to be carried out. This is to ensure that the proposed system is not a burden to the company. For feasibility analysis, some understanding of the major requirements for the system is essential.

Three key considerations involved in the feasibility analysis are

- ECONOMICAL FEASIBILITY
- TECHNICAL FEASIBILITY
- SOCIAL FEASIBILITY

6.1ECONOMICAL FEASIBILITY

This study is carried out to check the economic impact that the system will have on the organization. The amount of fund that the company can pour into the research and development of the system is limited. The expenditures must be justified. Thus the developed system as well within the budget and this was achieved because most of the technologies used are freely available. Only the customized products had to be purchased.

6.2 TECHNICAL FEASIBILITY

This study is carried out to check the technical feasibility, that is, the technical requirements of the system. Any system developed must not have a high demand on the available technical resources. This will lead to high demands on the available technical resources. This will lead to high demands being placed on the client. The developed system must have a modest requirement, as only minimal or null changes are required for implementing this system.

6.3 SOCIAL FEASIBILITY

The aspect of study is to check the level of acceptance of the system by the user. This includes the process of training the user to use the system efficiently. The user must not feel threatened by the system, instead must accept it as a necessity. The level of acceptance by the users solely depends on the methods that are employed to educate the user about the system and to make him familiar with it. His level of confidence must be raised so that he is also able to make some constructive criticism, which is welcomed, as he is the final user of the system.

CHAPTER-7

SYSTEM TESTING

The purpose of testing is to discover errors. Testing is the process of trying to discover every conceivable fault or weakness in a work product. It provides a way to check the functionality of components, sub-assemblies, assemblies and/or a finished product. It is the process of exercising software with the intent of ensuring that the Software system meets its requirements and user expectations and does not fail in an unacceptable manner. There are various types of tests. Each test type addresses a specific testing requirement.

7.1 TYPES OF TESTS

7.1.1 Unit testing

Unit testing involves the design of test cases that validate that the internal program logic is functioning properly, and that program inputs produce valid outputs. All decision branches and internal code flow should be validated. It is the testing of individual software units of the application. It is done after the completion of an individual unit before integration. This is a structural testing, that relies on knowledge of its construction and is invasive. Unit tests perform basic tests at component level and test a specific business process, application, and/or system configuration. Unit tests ensure that each unique path of a business process performs accurately to the documented specifications and contains clearly defined inputs and expected results.

Unit testing is usually conducted as part of a combined code and unit test phase of the software lifecycle, although it is not uncommon for coding and unit testing to be conducted as two distinct phases. Test strategy and approach Field testing will be performed manually and functional tests will be written in detail.

Test objectives

- All field entries must work properly.
- Pages must be activated from the identified link.
- The entry screen, messages and responses must not be delayed.
- Features to be tested
- Verify that the entries are of the correct format

- No duplicate entries should be allowed

7.1.2 Integration testing

Integration tests are designed to test integrated software components to determine if they actually run as one program. Testing is event driven and is more concerned with the basic outcome of screens or fields. Integration tests demonstrate that although the components were individually satisfaction, as shown by successfully unit testing, the combination of components is correct and consistent. Integration testing is specifically aimed at exposing the problems that arise from the combination of components.

7.1.3 Functional test

Functional tests provide systematic demonstrations that functions tested are available as specified by the business and technical requirements, system documentation, and user manuals.

Organization and preparation of functional tests is focused on requirements, key functions, or special test cases. In addition, systematic coverage pertaining to identify Business process flows; data fields, predefined processes, and successive processes must be considered for testing. Before functional testing is complete, additional tests are identified and the effective value of current tests is determined. Software integration testing is the incremental integration testing of two or more integrated software components on a single platform to produce failures caused by interface defects. The task of the integration test is to check that components or software applications, e.g. components in a software system or – one step up – software applications at the company level – interact without error.

Test Results: All the test cases mentioned above passed successfully. No defects encountered.

Systems/Procedures: interfacing systems or procedures must be invoked. Organization and preparation of functional tests is focused requirements, key functions, or special test cases. In addition, systematic coverage pertaining to identify Business process flows; data fields, predefined processes, and successive processes must be considered for testing. Before functional testing is complete, additional tests are identified and the effective value of current tests is determined.

7.1.4 System Test

System testing ensures that the entire integrated software system meets requirements. It tests a configuration to ensure known and predictable results. An example of system testing is the configuration oriented system integration test. System testing is based on process descriptions and flows, emphasizing pre-driven process links and integration points.

7.1.5 White Box Testing

White Box Testing is a testing in which in which the software tester has knowledge of the inner workings, structure and language of the software, or at least its purpose. It is purpose. It is used to test areas that cannot be reached from a black box level.

7.1.6 Black Box Testing

Black Box Testing is testing the software without any knowledge of the inner workings, structure or language of the module being tested. Black box tests, as most other kinds of tests, must be written from a definitive source document, such as specification or requirements document, such as specification or requirements document. It is a testing in which the software under test is treated, as a black box .you cannot “see” into it. The test provides inputs and responds to outputs without considering how the software works.

7.1.7 Acceptance Testing

User Acceptance Testing is a critical phase of any project and requires significant participation by the end user. It also ensures that the system meets the functional requirements.

Test Results: All the test cases mentioned above passed successfully. No defects encountered.

CHAPTER-8

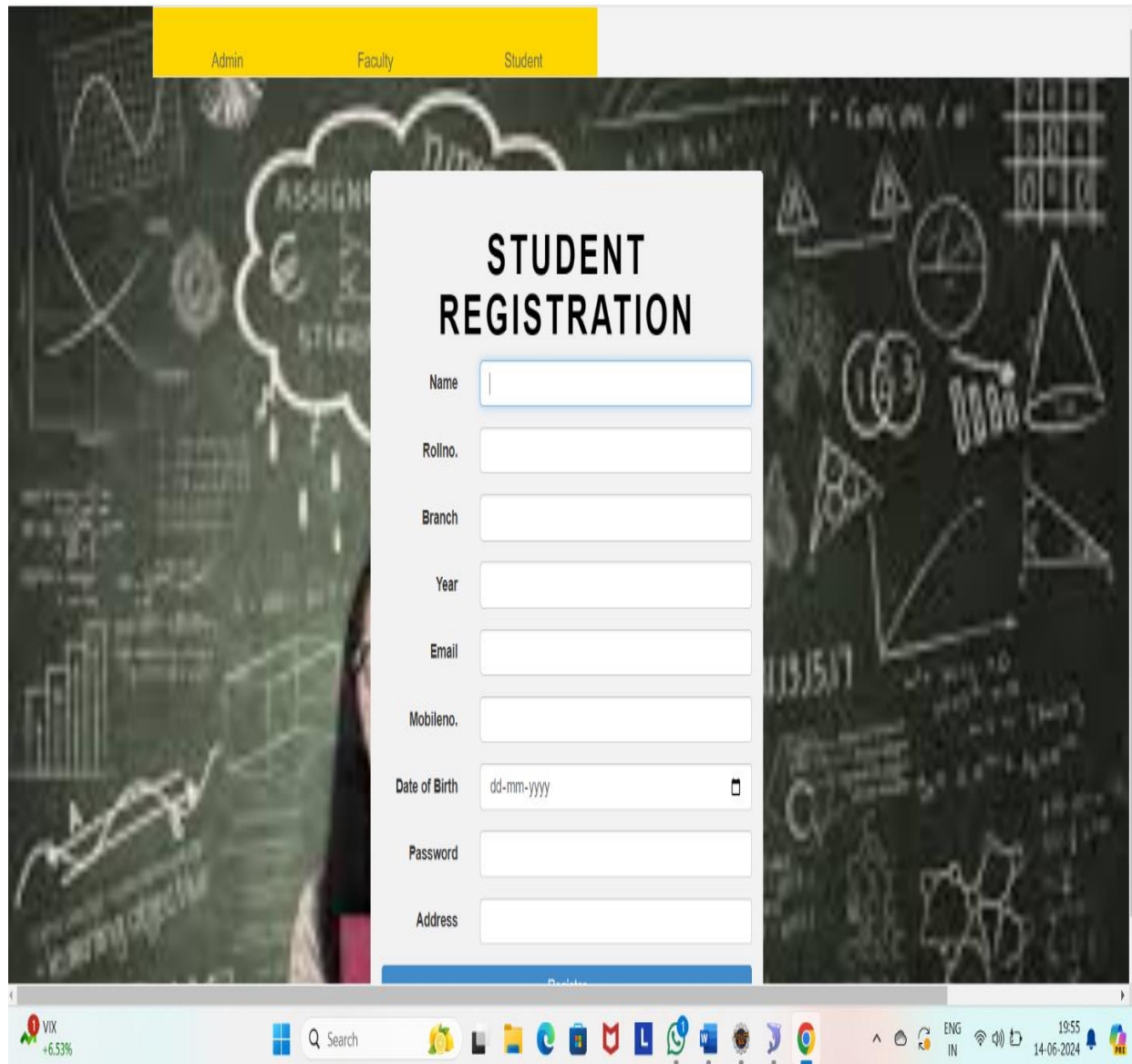
RESULTS

8.1 HOME PAGE:



Fig:8.1 Screenshot of Home Page

8.2 STUDENT REGISTRATION PAGE:



The screenshot displays a web application interface for student registration. At the top, a yellow navigation bar contains three links: 'Admin', 'Faculty', and 'Student'. The 'Student' link is currently selected, highlighted in yellow. The background of the page is a chalkboard filled with various mathematical diagrams and equations, including a sine wave, a bar chart, a line graph, and geometric shapes like triangles and circles. In the center, a white registration form is overlaid. The form is titled 'STUDENT REGISTRATION' in bold, black, uppercase letters. Below the title, there are eight input fields, each with a label to its left: 'Name', 'Rollno.', 'Branch', 'Year', 'Email', 'Mobilen.', 'Date of Birth', and 'Password'. The 'Date of Birth' field includes a calendar icon. Below these fields is an 'Address' field. At the bottom of the form, there is a blue button labeled 'Register'. The Windows taskbar is visible at the bottom of the screen, showing the Start button, a search bar, and several application icons. The system tray on the right shows the date and time as 19:55 on 14-06-2024.

Admin Faculty Student

STUDENT REGISTRATION

Name


Rollno.

Branch

Year

Email

Mobilen.

Date of Birth 

Password

Address

Register

Fig:8.2 Screenshot of Student Registration Page

8.3 STUDENT LOGIN PAGE:

Admin Faculty Student

STUDENT LOGIN

Rollno

Password*

Login

[CREATE NEW REGISTRATION?](#)

OneDrive - Personal
Office is syncing open files

84°F Mostly cloudy Search ENG IN 19:55 14-06-2024

Fig:8.3 Screenshot of Student login page

8.4 ADMIN LOGIN PAGE:

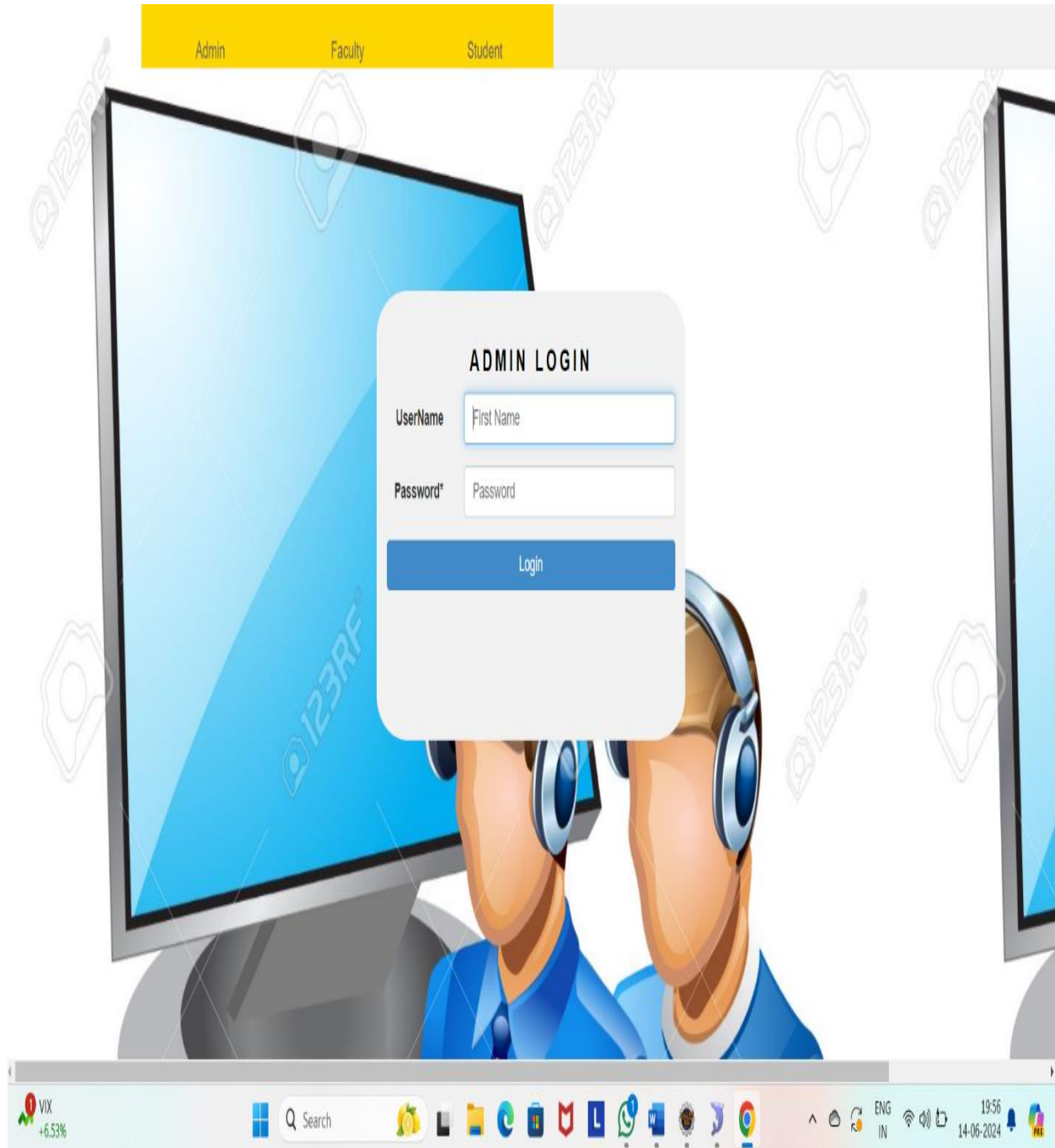


Fig8.4: Screenshot of admin login page

8.5 ADMIN PAGE:



Fig8.5 Screenshot of Admin Page

8.6 FACULTY INFORMATION:

Project Status & Student Performance

Facultyinfo

Studentinfo

Logout

Id	username	email	password	mobile
1	xcbgxf	sgf@gmail.com	vbvnbm	9341521
2	xcbgxf	sgf@gmail.com		9341521
3	xcbgxf	sgf@gmail.com		9341521
4	xcbgxf	sgf@gmail.com		9341521
5	fdgfffffffffffffff	r@gmail.com	123456	987686867
6	gbgvnbmbbmbb	rrrrrr@gmail.com	76565	986767
7	bvnbm	cv@gmail.com	786	7876876
8	veera	veera@gmail.com	veera	7702084601
9	akshaya	akshaya@gmail.com	1234	1234567899
10	keerthu	keerthu@gmail.com	1234	9389789609
11	swaroopa	swaroopa@gmail.com	1234	9876543215

USD/EUR
+0.57%

Q Search

ENG
IN

19:57
14-06-2024

Fig8.6Screenshot of Faculty Information

8.7 STUDENT INFORMATION:

Project Status & Student Performance

Facultyinfo

Studentinfo

Logout

id	username	email	password	mobile
1	xcbgxf	sgf@gmail.com	vbvnbm	9341521
2	xcbgxf	sgf@gmail.com		9341521
3	xcbgxf	sgf@gmail.com		9341521
4	xcbgxf	sgf@gmail.com		9341521
5	fdgffffffffffffffff	r@gmail.com	123456	987686867
6	gbgvnbmbbmbb	rrrrrr@gmail.com	76565	986767
7	bvnbm	cv@gmail.com	786	7876876
8	veera	veera@gmail.com	veera	7702084601
9	akshaya	akshaya@gmail.com	1234	1234567899
10	keerthu	keerthu@gmail.com	1234	9389789609
11	swaroopa	swaroopa@gmail.com	1234	9876543215

USD/EUR
+0.57%

Q Search

ENG IN

19:57
14-06-2024

Fig8.7Screenshot of Student Information

8.8 FACULTY LOGIN:

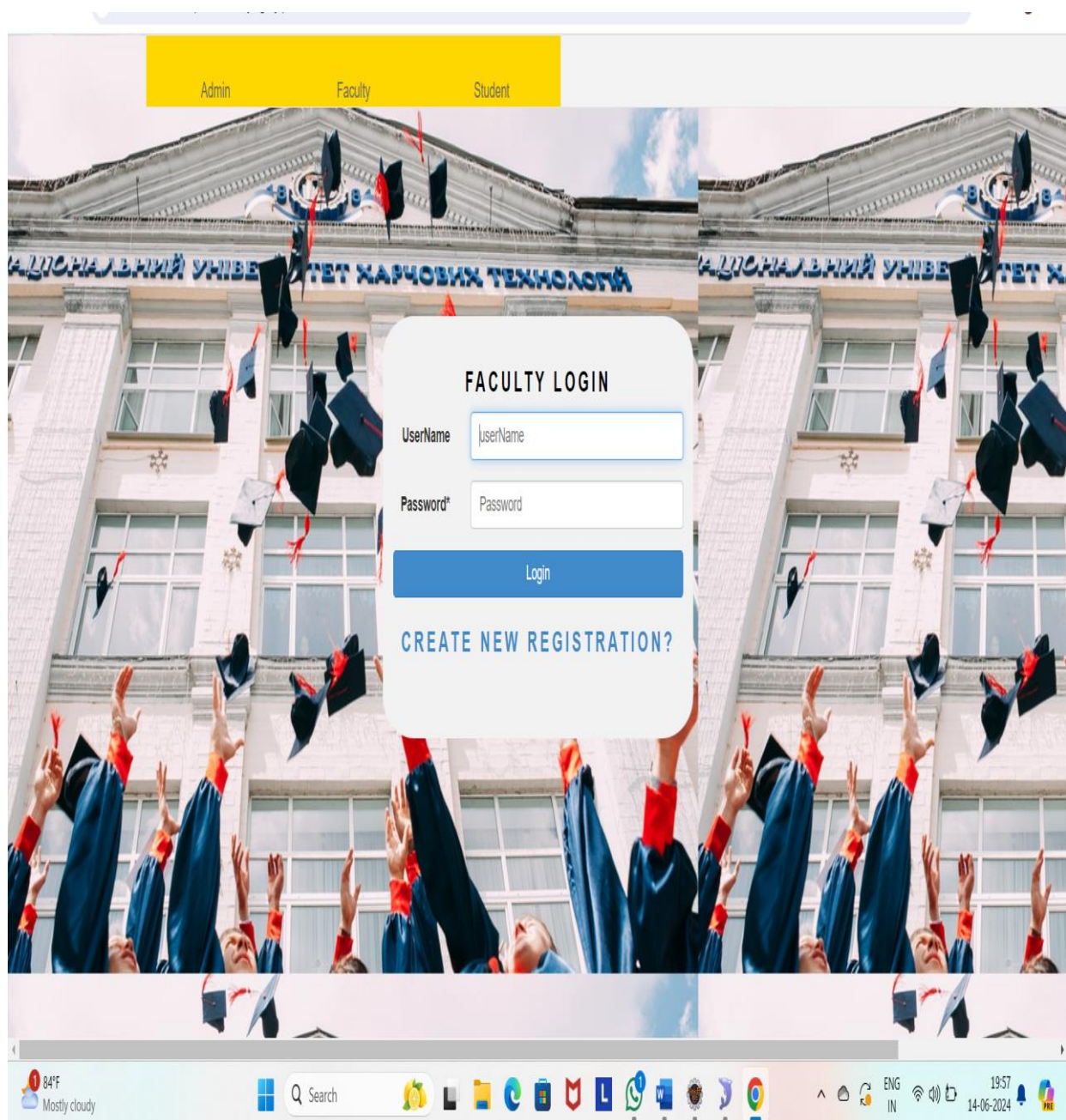


Fig 8.8: Screenshot of Faculty login

8.9 FACULTY HOME:

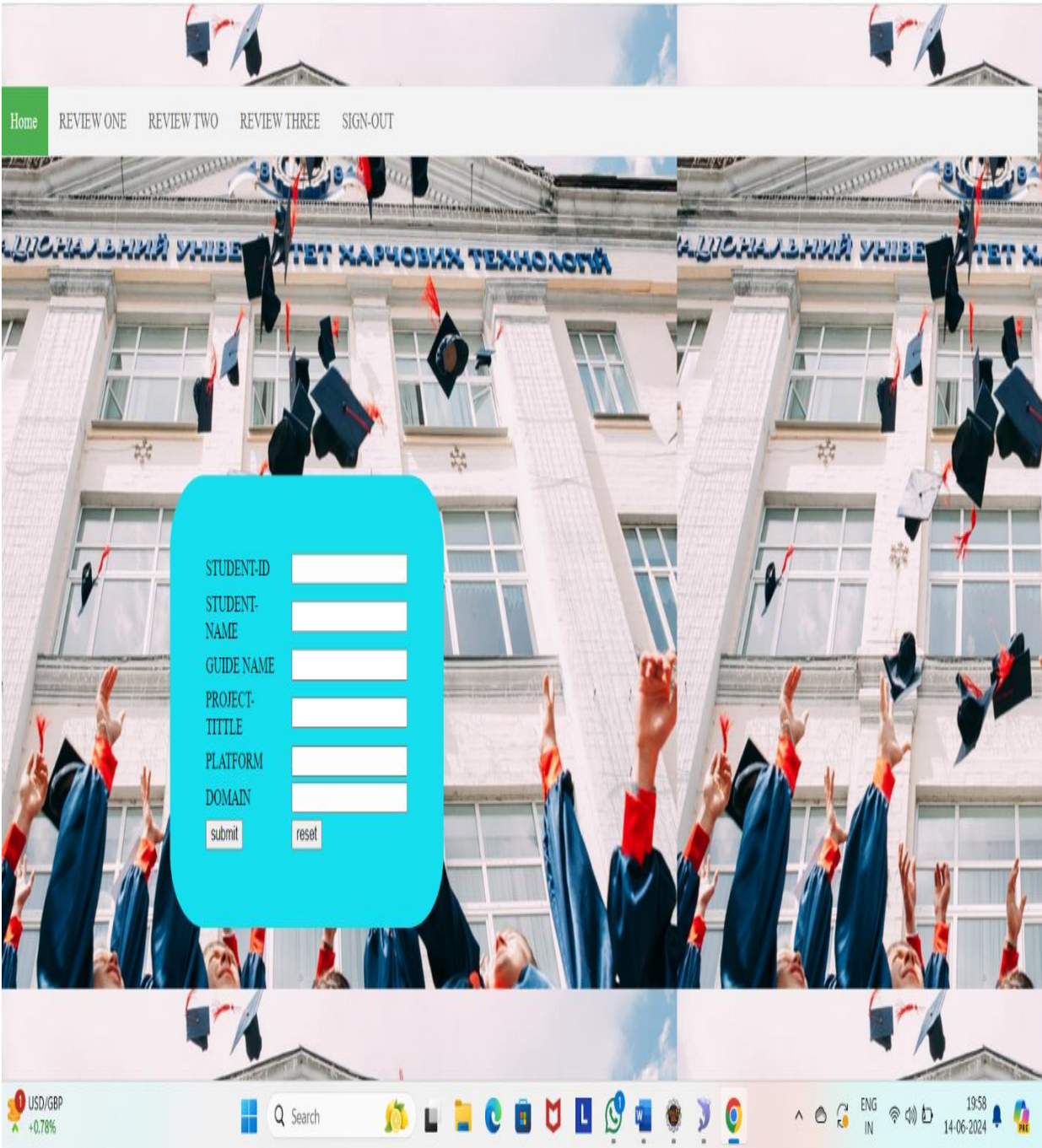


Fig:8.9 Screenshot of Faculty Home

8.10 STUDENT PROJECT DETAILS:

Project Status & Student Performance

[Home](#)
[REVIEW ONE](#)
[REVIEW TWO](#)
[REVIEW THREE](#)
[SIGN-OUT](#)

SNO	STUDENT-ID	STUDENT-NAME	GUIDE	TITTLE	PLATFORM	DOMAIN	MARKS
1	1245	veera	versontin	JAVAAVV	XAS	EFWFW	ENTER
2	599	sravanthi	akshaya	login	java	java	ENTER
3	1234	anu	akshaya	info	java	java	ENTER
4	589	keerthana	keerthu	student	java	java	ENTER
5	599	akshaya	swaroopa	title	java	java	ENTER
6	571	hai	ga	title	java	java	ENTER



Fig8.10Screenshot of Student Project Details

8.11 MARKS ENTERING

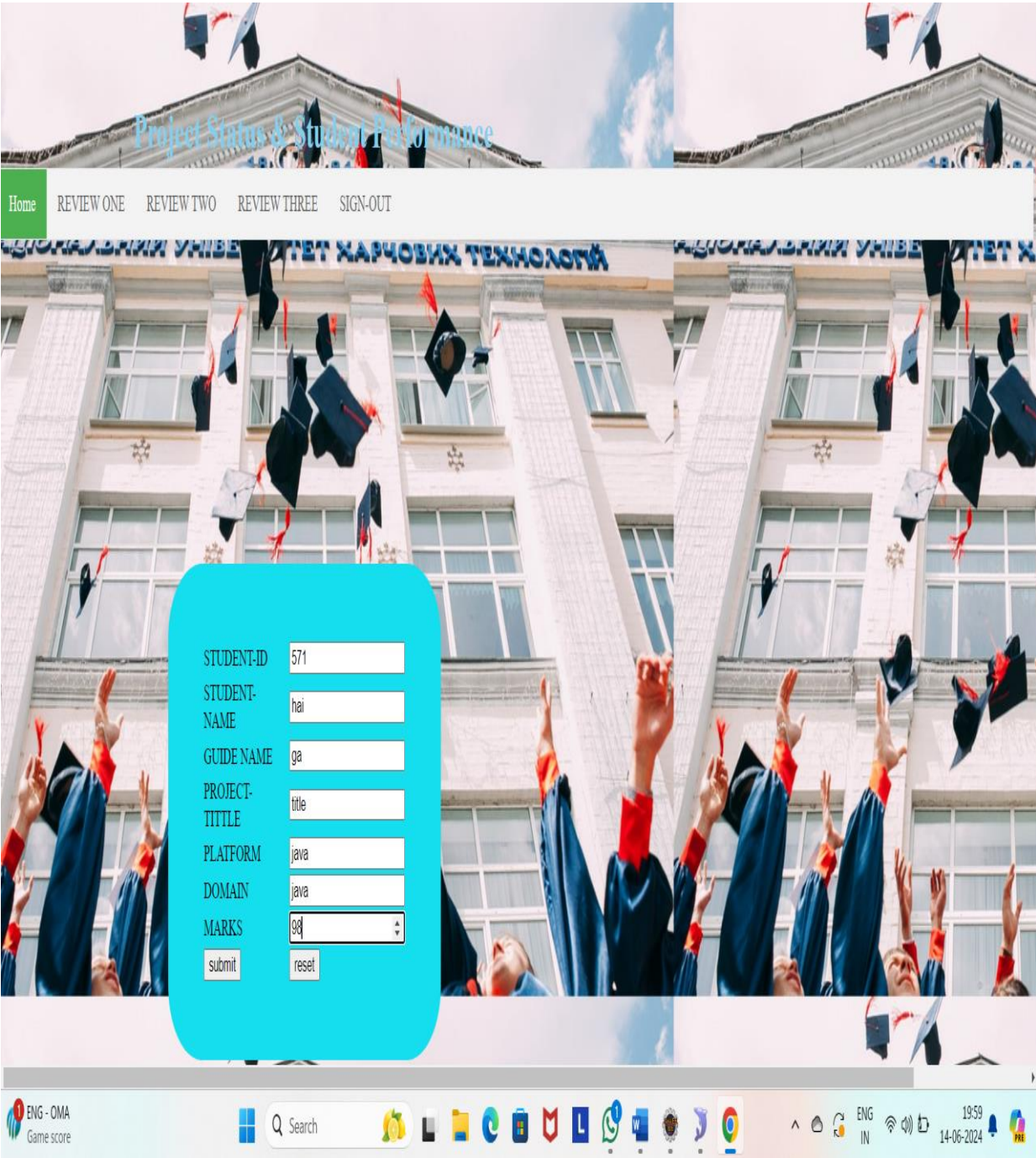


Fig8.11 Screenshot of Marks Entering Page

8.12STUDENT GRAPH:

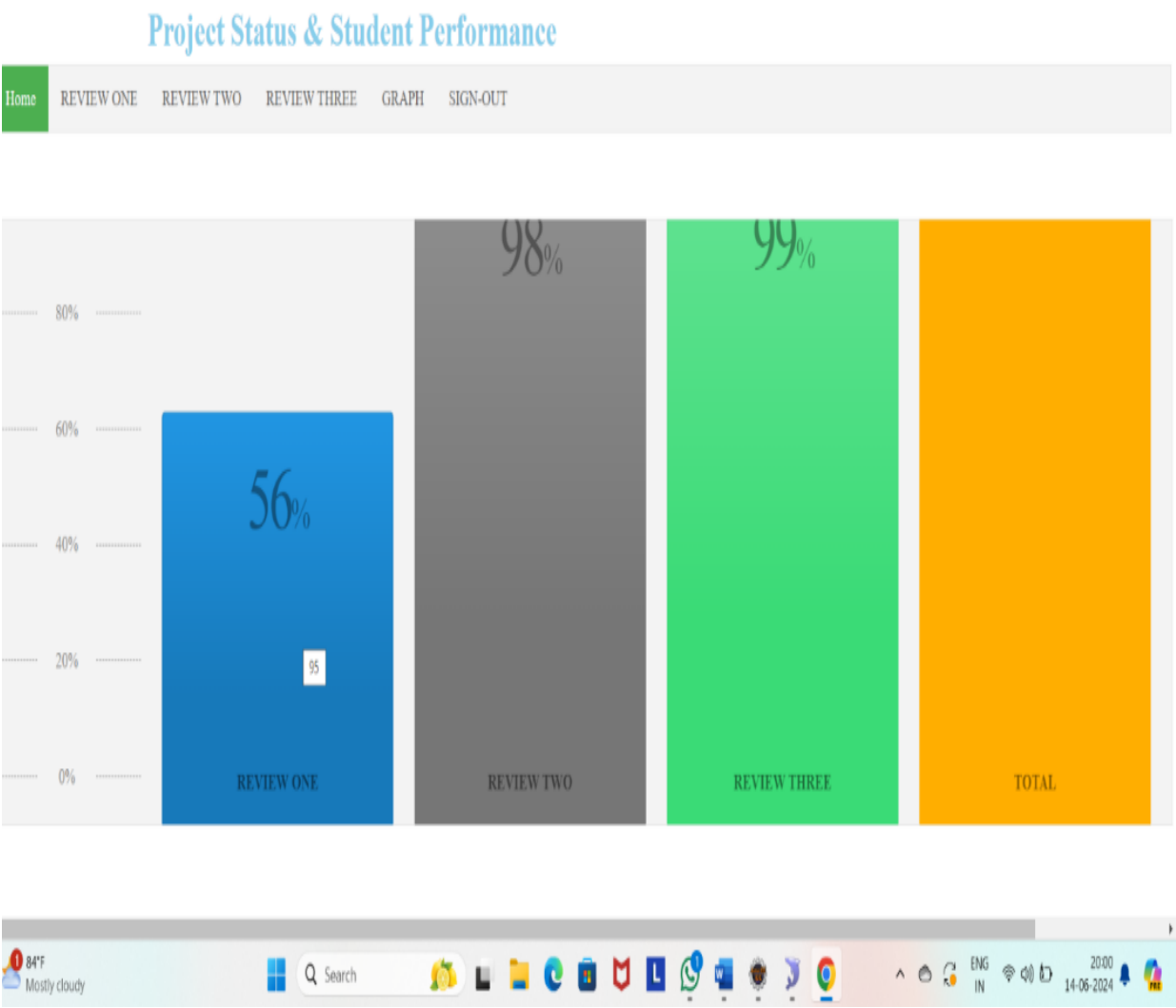


Fig8.12Screenshot of Student Graph

CHAPTER-9

CONCLUSION AND ENHANCEMENT

The project "Project Time Analysis and Reporting System" aims at accomplishing the task of allowing the project manager to maintain the project details. It also helps in maintaining the time details of each project. The system provides a graphical user interface, which helps all the employees to know the project details. It also generates reports, which gives detailed information about the clients of the company, different groups their size including the team leaders. Future enhancements for this project can be also created using Bar-charts by which the performance of each project can be better analyzed and by using the resource allocation can be done efficiently.

FUTURE ENHANCEMENT:

This project can also be done in android application. Alert message can be sent to register number with the status. In this project we can also implement the other projects like attendance and aggerigation. We can also use back end as cloud.

CHAPTER-10

REFERENCES

JAVA-2 Complete-Reference

Author: Patrice Norton & Herbert Child Publisher: Tata McGraw Hill.

System Analysis & Designing

Author: James A. Senn Publisher: Tata McGraw Hill.

Software engineering

principles Author:

Pressman.

Java Server Pages

Author: James good will.

Database Management System

Author: Raghu Rama Krishnan

https://www.google.com/search?q=eZ+school+links&rlz=1C1CHZL_enIN753IN753&oq=eZ+scho

[ol+links&aqs=chrome..69i57j9j7&sourceid=chrome&ie=UTF-8](https://www.google.com/search?q=eZ+school+links&rlz=1C1CHZL_enIN753IN753&oq=eZ+scho)

https://www.google.com/search?q=employee+status+reporting&rlz=1C1CHZL_enIN753IN753&oq=employee+status+reporting&aqs=chrome..69i57j9j7&sourceid=chrome&ie=UTF-8

[oq=employee+status+reporting&aqs=chrome..69i57j9j7&sourceid=chrome&ie=UTF-8](https://www.google.com/search?q=employee+status+reporting&rlz=1C1CHZL_enIN753IN753&oq=employee+status+reporting&aqs=chrome..69i57j9j7&sourceid=chrome&ie=UTF-8)