EXECUTIVE SUMMARY

This project is entitled as "Student Internal Assessment Marks Management System". The main aim of this system is to develop the web application software to store and retrieve the student's marks. We are able to see the all the student's marks in a organized way. Student IA marks management system has been designed to store and retrieve marks of all students mark in the educational institution in most efficient manner. Given the continuous rise in student population, tertiary institutions storing the student marks with the help of computer programs makes it easy to organize and manage huge data. Storing student's information and their academic record is very important. This application helps to the department staffs and students to know the marks in the internal assessment. This project is to be developed using HTML and CSS for Front end MySQL for Back end and JSP to connect these technologies.

INTRODUCTION

The “IA MARKS MANAGEMENT” has been developed to overrie the problems prevailing in the practicing manual system.This software is supporte to eliminate and in some cases reduce the hardships faced by existing system.Moreover this system is designed for the particular need of the company to carry out operations in a smooth an effective manner.

The purpose of IA Marks Management system is to automate the existing manual system by the help of computerized equipments and full fledged computer software fulfilling their requirements so that they are valuable data information can be stored for a longer period with accessing and manipulation of the same.

The application is reduced as much as possible to avoid errors while entering the data it also provides error message while entering invalid data. No formal knowledge is needed for the user to use the system does by this all the truth it is user friendly.

IA Marks Management system as described above can lead to error free secure reliable and fast management system.It can assess the user to concentrate on their other activities rather to concentrate on record keeping.Thus it will help organisation in better utilisation of resources.

INDUSTRY PROFILE / COMPANY PROFILE

AUDAZ VENTURES PVT. LTD., with the headquarter in New Delhi and Corporate Office in Bengaluru was established in the year 2020. It is a service - based company which provides service and solutions to 62 institutes across India. It has its presence in more than 12 cities across India.

Directors AUDAZ VENTURES PVT. LTD., are Mr. Man Mohan and Mr. Rahul Oberoi.

Its services include

- Software solutions
- Digital marketing
- SAP software
- ERP software
- Blockchain software
- Placement related training to the Engineering Graduates and many more
- Placement opportunities to students

OBJECTIVES AND SCOPE OF THE STUDY

Objectives

The main objective of the project on the IA Marks Management System is to manage the details of Student IA marks. It manages all the information about student marks ,insertion, deletion and updation can be done. Also marks can be viewed of each student individually. The project is totally built at administrative level and thus only the administrator is guaranteed the access. The purpose of the project is to build an application program to reduce the manual work for managing the student marks

Scope

- It may help collecting perfect management in details in a very short time, the collection will be obvious simple and sensible
- It will help the person to know the management of a past year perfectly and vividly
- It also helps in current all works relative to IA management
- It will also be reduced the cost of collecting the management and collection procedure will go on smoothly.
- In computer system it is not necessary to create the manifest but we can directly printed which saves our time.
- To assess the staff in capturing the effort spent on their respective working areas
- To utilise resources in a efficient manner for increasing the productivity through automation
- The system generates types of information that can be used for various purposes.

THEORETICAL BACKGROUND

MySQL

MySQL is an open-source relational database management system (RDBMS). "SQL", the abbreviation **for Structured Query Language**. A relational database organizes data into one or more data tables in which data types may be related to each other; these relations help structure the data. SQL is a language programmers use to create, modify and extract data from the relational database, as well as control user access to the database. In addition to relational databases and SQL, an RDBMS like MySQL works with an operating system to implement a relational database in a computer's storage system, manages users, allows for network access and facilitates testing database integrity and creation of backups.

HTML

The **Hyper Text Markup Language**, or HTML is the standard markup language for documents designed to be displayed in a web browser. Web browsers receive HTML documents from a web server or from local storage and render the documents into multimedia web pages. HTML describes the structure of a web page semantically and originally included cues for the appearance of the document. HTML elements are the building blocks of HTML pages. With HTML constructs, images and other objects such as interactive forms may be embedded into the rendered page. HTML provides a means to create structured documents by denoting structural semantics for text such as headings, paragraphs, lists,links, quotes and other items. HTML elements are delineated by tags, written using angle brackets. Tags such as <p> surround and provide information about document text and may include other tags as sub-elements. Browsers do not display the HTML tags, but use them to interpret the content of the page.

CSS

Cascading Style Sheets (CSS) is a style sheet language used for describing the presentation of a document written in a markup language such as HTML. CSS is a cornerstone technology of the World Wide Web, alongside HTML and JavaScript. CSS is designed to enable the separation of presentation and content, including layout, colours, and fonts. This separation can improve content accessibility; provide more flexibility and control in the specification of presentation characteristics; enable multiple web pages to share formatting by specifying the relevant CSS in a separate .CSS file, which reduces complexity and repetition in the structural content; and enable the .CSS file to be cached to improve the page load speed between the pages that share the file and its formatting.

JAVA SERVER PAGE (JSP)

JSP technology is used to create web application just like Servlet technology. It can be thought of as an extension to Servlet because it provides more functionality than servlet such as expression language, JSTL, etc.

A JSP page consists of HTML tags and JSP tags. The JSP pages are easier to maintain than Servlet because we can separate designing and development. It provides some additional features such as Expression Language, Custom Tags, etc.

The JSP pages follow these phases:

- Translation of JSP Page
- Compilation of JSP Page
- Classloading (the classloader loads class file)
- Instantiation (Object of the Generated Servlet is created).
- Initialization (the container invokes jspInit() method).
- Request processing (the container invokes _jspService() method).
- Destroy (the container invokes jspDestroy() method).

RESEARCH METHODOLOGY

1. What is Java?
2. Discuss about Editor of java IDE like Eclispse, Netbeans etc. of JAVA.
3. Versions of Java.
4. Structure of Java Program.
5. Discuss about Memory Management and Compilation Program.
6. Discuss about API.
7. Discuss of Third party tools.
8. Discuss of Various Database Connectivity with Java Language.
9. Describe Java Security Regions and attacks on code.
10. Discuss about application and Project or software and implements in Java.
11. Future of JAVA.

HYPOTHESIS

Research work mentioned all the aspect relative to Java Programming Language.This is an attempt to investigate how students develop project and applications in java an understandings of all concepts in „Programming disciplines. In order to have a reasonable focus, its scope is limited to learning computer programming, and centrally, coming to understand the concepts of object-oriented programming (OOP) in Java. It is restricted to novices – students with little or no previous experience of programming.

Why Java?

While some investigation in this thesis looks at general notions of computers and the execution of short pieces of pseudo-code, the focus will be on the Java language and associated OOP concepts. There are several reasons for this. Java is currently a very widely used language, for desktop applications, client-server systems (using „Java Enterprise Edition) and mobile phone applications on Android.

Java is already familiar with these concepts, which of course is not true for novices.
A second aspect of Java is that it involves a set of inter-dependent ideas, which cannot be isolated. As a fairly „pure OOP language, even the simplest executable program makes reference to a host of unfamiliar concepts, represented by the keywords public, class,

static, void and. Beyond the first program, OOP entails a number of ideas which are distinct but closely linked, such as class and object. This presents a difficulty in designing a learning sequence, which one would ideally like to lead the student to encounter new ideas one at a time – impossible in Java because the concepts are dependent upon each other.

Concepts

What are the concepts used in OOP? Armstrong (2006) examined 239 sources published between 1966 and 2005 to identify the most commonly used set of OOP ideas. The 'top ten' were in order:

1. Inheritance
2. Object
3. Class
4. Encapsulation
5. Method
6. Message Passing
7. Polymorphism
8. Abstraction
9. Instantiation
10. Attribute

WORK PLANAND METHODOLOGY

In this dissertation our main aim is to study and implement and make own structure or an Algorithm to improve the Securityand how student develop Applications or projects using java and how many library provided for security and packages in java ,how to make project or web application more robust, fast, and reliable in java and discuss various tools. Complete dissertation will be covered in a Five Chapters. The entire Chapter further divided into number of modules. Abstract of Five chapters is as follows:

1: Introduction of JAVA Language.

2: Literature Survey

3: Problem Definition

4: Implementation of Techniques for Make Java Application or other.

5: Discussion & Result

ANALYSIS AND INTERPRETATION OF DATA

Java Database Connectivity (JDBC) is an application programming interface (API) for the programming language Java, which defines how a client may access a database. It is a Java-based data access technology used for Java database connectivity. It is part of the Java Standard Edition platform, from Oracle Corporation. It provides methods to query and update data in a database, and is oriented toward relational databases. A JDBC-to-ODBC bridge enables connections to any ODBC-accessible data source in the Java virtual machine (JVM) host environment. These drivers fall into one of the following types:

1. Type 1 that calls native code of the locally available ODBC driver.
2. Type 2 that calls database vendor native library on a client side. This code then talks to database over the network.
3. Type 3 the pure-java driver that talks with the server-side middleware that then talks to the database.
4. Type 4, the pure-java driver that uses database native protocol.

Steps for connecting Java programs to database :

- Import the packages import java.sql.*
- Load the drivers using the forName() method of class Class.
- Register the drivers using Driver Manager.
- Establish a connection using the connection class object.
- Create a statement.
- Execute the query.
- Closed the connection.

FINDINGS

Figure-1: Home Page

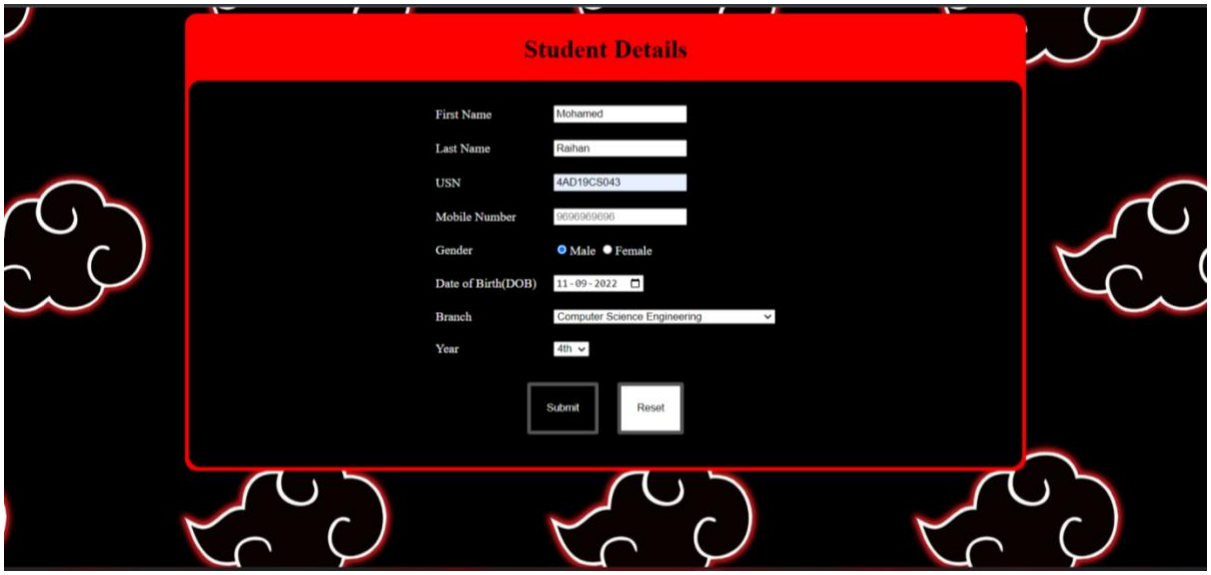

Figure-2: Student Registration Form

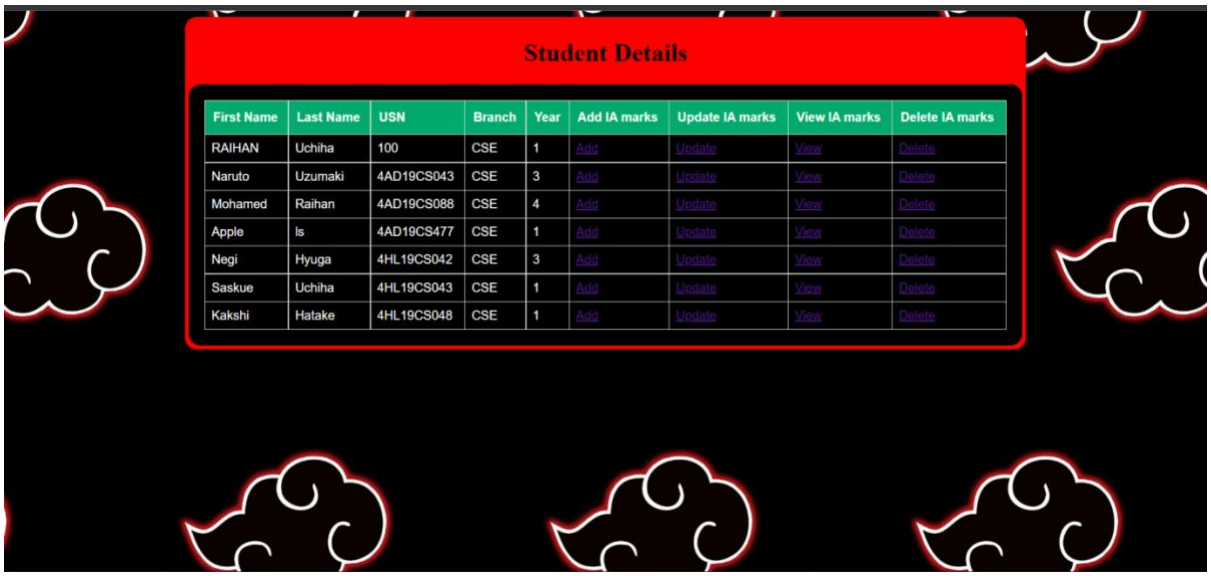

Figure-3: Student Information Table

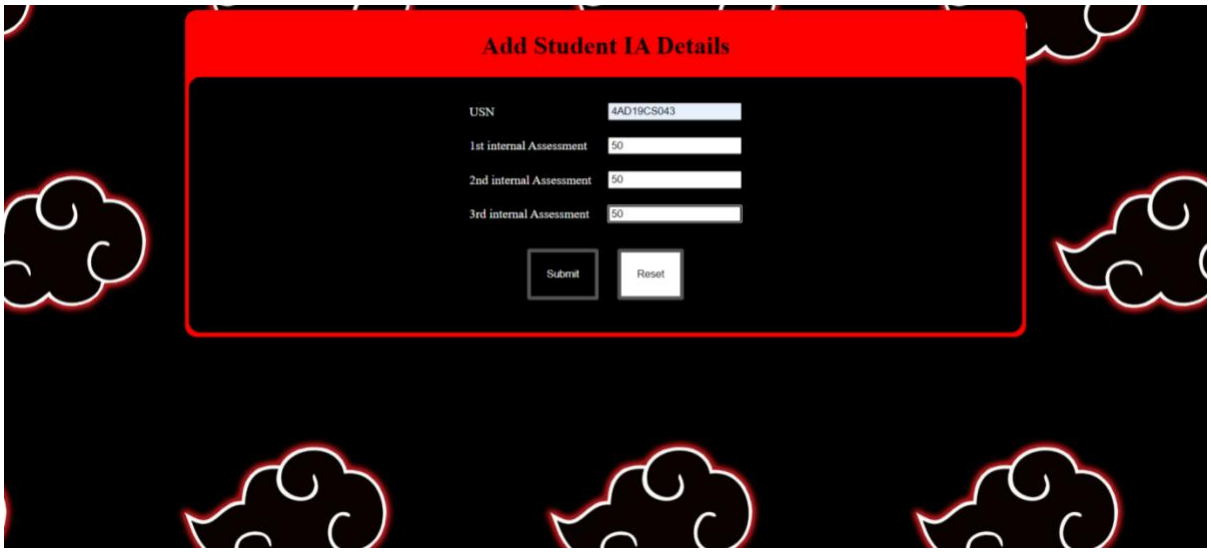

Figure-4: Student IA Marks Update Form

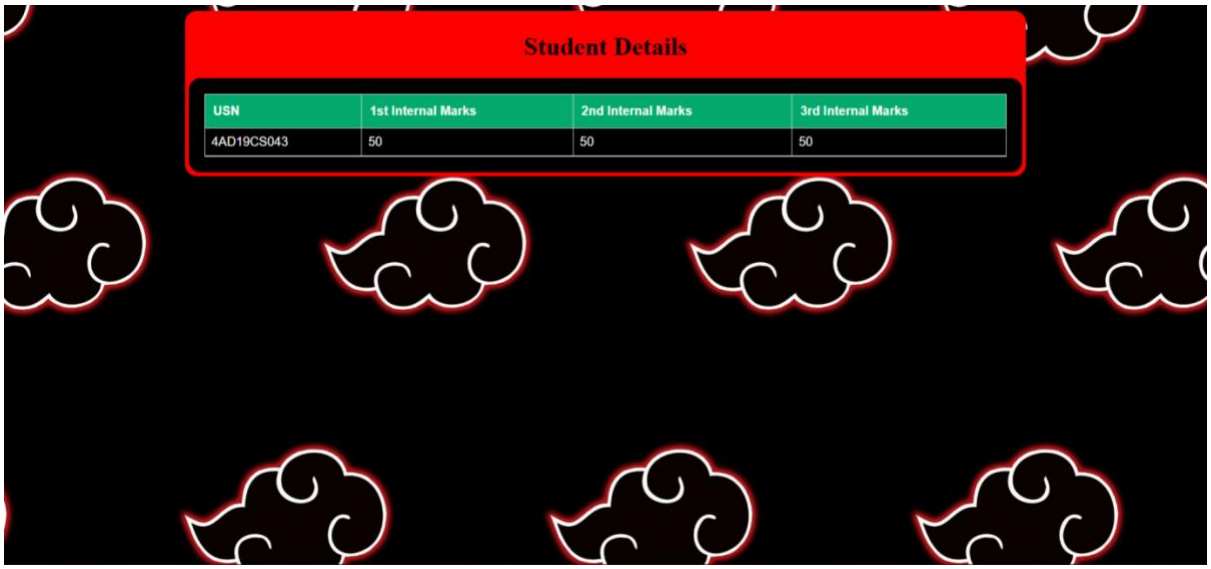

Figure-5: Student IA Marks Table

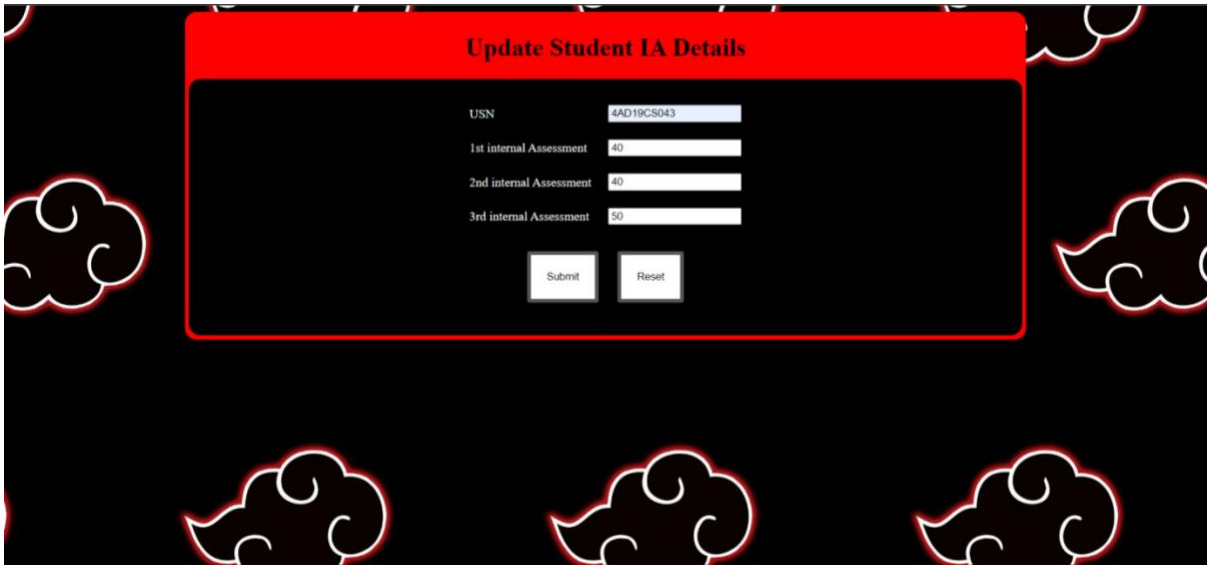

Figure-6: Student IA Marks Update Form

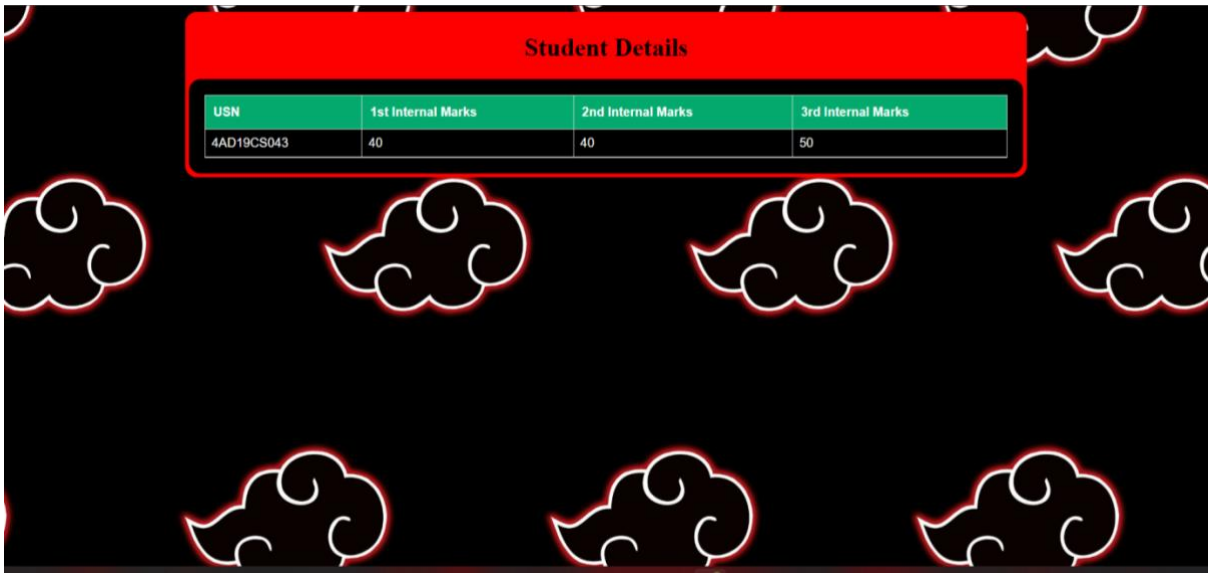

Figure-6: Student IA Marks Update Table

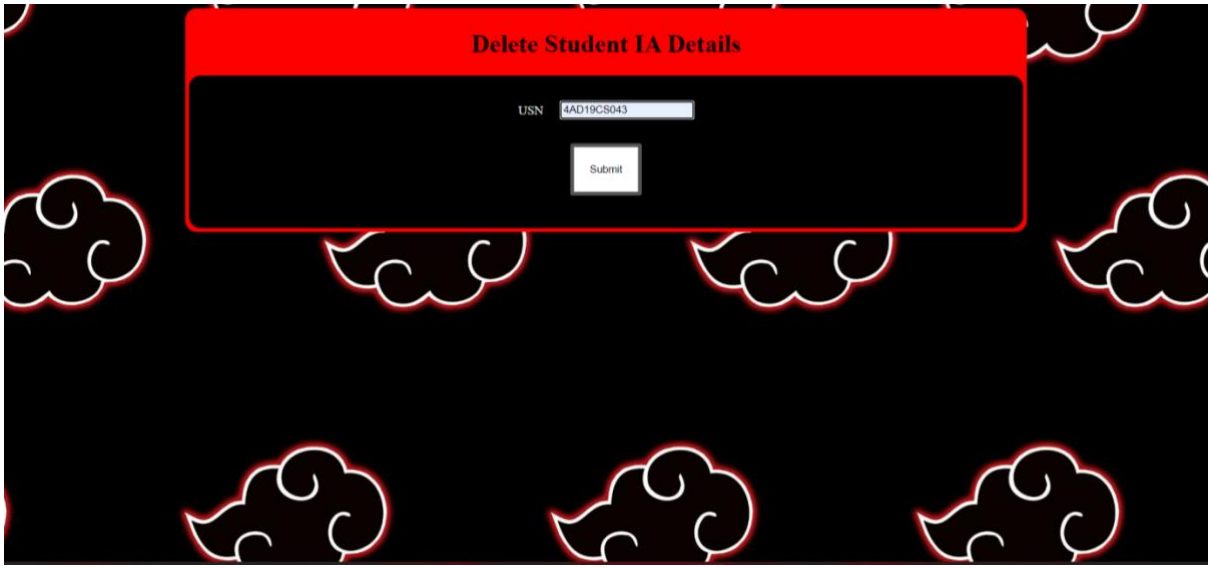

Figure-7: Student Information Delete Form

SUGGESTIONS / RECOMMENDATIONS

In a nutshell, it can be summarised that the future scope of the project circles around maintaining information regarding:

- We can add marks in the future.
- We can give more advanced software for a marks management including more facilities.
- We will host the platform online service to make it accessible worldwide.
- Integrate multiple load balancers distribute the load of the system.
- Create the master and slave database structure to reduce the overall rating based queries.
- Implement the backup mechanism for taking backup of codebase and database on regular basis on different servers.
-

The above mentioned points enhancements which can be done to increase their applicability and uses of the project.

PSEUDO CODE

Index.html File:

```
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title>IA MARKS DETAILS</title>
<link rel="stylesheet" href="styles.css">
</head>
<body>
<center>
<div class="form" style="width:23%;">
        <h1 style="color:red;">IA MARKS DETAILS</h1>
        <h2 style="color:white">ATME COLLEGE OF ENGINEERING </h2>
</div>
<img src="./itachi.jpg" height="100%"width="17%"></img>

</br>
<a href = "form.html">
<button class="button"> ADD DETAILS</button></a>
<a href= "view.jsp">
<button class="button"> VIEW DETAILS</button></a>
</center>

</body>
</html>
```

Add.jsp File:

```
<%@ page import="java.sql.*" %>
<%@ page language="java" contentType="text/html; charset=UTF-8"
 pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title>Insert title here</title>
<link rel="stylesheet" href="styles.css">
</head>
<body>
<% String name=request.getParameter("name");
String usn=request.getParameter("usn");
String branch=request.getParameter("branch");
String year=request.getParameter("year");
String phone=request.getParameter("phone");
String lname=request.getParameter("lname");
String gender=request.getParameter("gender");
String dob=request.getParameter("dob");
```

```
String query = "insert into student(name,usn,branch,year,lname,phone,gender,dob)
values("'+name+"',"'+usn+"',"'+branch+"',"+year+","'+lname+"',"+phone+","'+gender+"',"'+d
ob+"');";
Class.forName("com.mysql.cj.jdbc.Driver");
Connection con = DriverManager.getConnection("jdbc:mysql://localhost:3307/atme", "root",
"1234");
System.out.println("Connection Established successfully");
PreparedStatement st = con.prepareStatement(query);
st.executeUpdate(query);
st.close();
con.close();
%>
<center>
<div class="form" style="width:23%;">
<h1 style="color:red;text-align:center;"> DETAILS STORED SUCCESSFULLY</h1>
</div>
</br>
<a href = "form.html">
<button class="button"> ADD DETAILS</button></a>
<a href= "view.jsp">
<button class="button"> VIEW DETAILS</button></a>
<a href= "index.html">
<button class="button"> HOME</button></a>
</center>
</body>
</html>
```

Addia.jsp File:

```
<%@ page import="java.sql.*" %>
<%@ page language="java" contentType="text/html; charset=UTF-8"
   pageEncoding="UTF-8"%>
   <!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<link rel="stylesheet" href="styles.css">
<title>Insert title here</title>
</head>
<body>
<%
String usn=request.getParameter("usn");
String ia1=request.getParameter("ia1");
String ia2=request.getParameter("ia2");
String ia3=request.getParameter("ia3");
String query = "insert into iamarks values("'+usn+"',"+ia1+","+ia2+","+ia3+");";
        Class.forName("com.mysql.cj.jdbc.Driver");
Connection con = DriverManager.getConnection("jdbc:mysql://localhost:3307/atme", "root",
"1234");
System.out.println("Connection Established successfully");
```

```
PreparedStatement st = con.prepareStatement(query);
st.executeUpdate(query);
   st.close();
   con.close();
%>
<center>
<div class="form" style="width:23%;">
<h1 style="
color:red;
text-align:center;"> DETAILS STORED SUCCESSFULLY</h1>
</div>
</br>
<a href = "form.html">
<button class="button"> ADD DETAILS</button></a>
<a href= "view.jsp">
<button class="button"> VIEW DETAILS</button></a>
<a href= "index.html">
<button class="button"> HOME</button></a>
</center>
</body>
</html>
```

View.jsp File:

```
<%@ page import="java.sql.*" %>
<%@ page language="java" contentType="text/html; charset=UTF-8"
   pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
<head>
<title>Hello, world!</title>
  <meta charset="UTF-8" />
  <meta name="viewport" content="width=device-width,initial-scale=1" />
  <meta name="description" content="" />
  <link rel="stylesheet" type="text/css" href="styles.css" />
  <link rel="icon" href="img/logo.png">
  <script src="script.js"></script>
<title>Insert title here</title>
</head>
<body>
<center>
  <div class="contents" style="width :70%" >
  <h1>Student Details</h1>
  <div class="form">
  <table id="student">
<tr>
<th>First Name</th>
<th>Last Name</th>
<th>USN</th>
<th>Branch</th>
```

```
<th>Year</th>
<th>Add IA marks</th>
<th>Update IA marks</th>
<th>View IA marks</th>
<th>Delete IA marks</th>
</tr>
<%
String query = "select * from student";
Class.forName("com.mysql.cj.jdbc.Driver");
Connection con = DriverManager.getConnection("jdbc:mysql://localhost:3307/atme", "root",
"1234");
System.out.println("Connection Established successfully");
Statement st = con.createStatement();
ResultSet rs = st.executeQuery(query);
while(rs.next())
{
String name = rs.getString("name");
String lname = rs.getString("lname");
String usn =rs.getString("usn");
String branch=rs.getString("branch");
int year=rs.getInt("year");
%>
<td >
<% out.println(name);%>
</td>
<td >
<% out.println(lname);%>
</td>
<td >
<% out.println(usn );%>
</td>
<td >
<% out.println(branch );%>
</td>
<td >
<% out.println(year );%>
</td >
<td ><a href="insert.html">
<% out.println("Add" );%>
</a></td>
<td ><a href="update.html">
<% out.println("Update" );%>
</a></td>
<td >
<a href="viewia.html">
<% out.println("View" );%>
</a></td>
<td><a href="delete.html">
<% out.println("Delete" );%>
```

```
</a></td>
</tr>
<%
}
st.close();
con.close();   %>
</table>
</div></div></center>
</body>
</html>
```

Viewia.jsp File:

```
<%@ page import="java.sql.*" %>
<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
<head>
<title>Hello, world!</title>
  <meta charset="UTF-8" />
  <meta name="viewport" content="width=device-width,initial-scale=1" />
  <meta name="description" content="" />
  <link rel="stylesheet" type="text/css" href="styles.css" />
  <link rel="icon" href="img/logo.png">
  <script src="script.js"></script>
<title>Insert title here</title>
</head>
<body>
<center>
  <div class="contents" style="width :70%" >
  <h1>Student Details</h1>
  <div class="form">
  <table id="student">
<tr>
<th>USN</th>
<th>1st Internal Marks</th>
<th>2nd Internal Marks</th>
<th>3rd Internal Marks</th>
</tr>
<%
String usn1=request.getParameter("usn");
String query = "select * from iamarks where usn ='"+usn1+"';";
Class.forName("com.mysql.cj.jdbc.Driver");
Connection con = DriverManager.getConnection("jdbc:mysql://localhost:3307/atme", "root",
"1234");
System.out.println("Connection Established successfully");
PreparedStatement st = con.prepareStatement(query);
ResultSet rs = st.executeQuery(query);
rs.next();
```

```
String usn =rs.getString("usn");
String ia1 =rs.getString("ia1");
String ia2 =rs.getString("ia2");
String ia3 =rs.getString("ia3");
%>
<td>
<% out.println(usn );%>
</td>
<td>
<% out.println(ia1 );%>
</td>
<td>
<% out.println(ia2 );%>
</td>
<td>
<% out.println(ia3 );%>
</td>
</tr>
<%
st.close();
con.close();   %>
</table>
</div></div></center>
</body>
</html>
```

Update.jsp File:

```
<%@ page import="java.sql.*" %>
<%@ page language="java" contentType="text/html; charset=UTF-8"
   pageEncoding="UTF-8"%>
   <!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<link rel="stylesheet" href="styles.css">
<title>Insert title here</title>
</head>
<body>
<%
String usn=request.getParameter("usn");
String ia1=request.getParameter("ia1");
String ia2=request.getParameter("ia2");
String ia3=request.getParameter("ia3");
String query = "update iamarks set ia1="+ia1+",ia2="+ia2+",ia3="+ia3+" where
usn="'+usn+"'";
Class.forName("com.mysql.cj.jdbc.Driver");
Connection con = DriverManager.getConnection("jdbc:mysql://localhost:3307/atme", "root",
"1234");
System.out.println("Connection Established successfully");
```

```
PreparedStatement st = con.prepareStatement(query);
st.executeUpdate(query);
st.close();
con.close();
%>
<center>
<div class="form" style="width:23%;">
<h1 style="
color:red;
text-align:center;"> DETAILS STORED SUCCESSFULLY</h1>
</div>
</br>
<a href = "form.html">
<button class="button"> ADD DETAILS</button></a>
<a href= "view.jsp">
<button class="button"> VIEW DETAILS</button></a>
<a href= "index.html">
<button class="button"> HOME</button></a>
</center>
</body>
</html>
```

Delete.jsp File:

```
<%@ page import="java.sql.*" %>
<%@ page language="java" contentType="text/html; charset=UTF-8"
   pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
<head>
<title>Hello, world!</title>
  <meta charset="UTF-8" />
  <meta name="viewport" content="width=device-width,initial-scale=1" />
  <meta name="description" content="" />
  <link rel="stylesheet" type="text/css" href="styles.css" />
  <link rel="icon" href="img/logo.png">
  <script src="script.js"></script>
<title>Insert title here</title>
</head>
<body>
<%
String usn1=request.getParameter("usn");
String query = "delete  from iamarks where usn ='"+usn1+"';";
String query1 = "delete  from student where usn ='"+usn1+"';";
Class.forName("com.mysql.cj.jdbc.Driver");
Connection con = DriverManager.getConnection("jdbc:mysql://localhost:3307/atme", "root",
"1234");
System.out.println("Connection Established successfully");
PreparedStatement st = con.prepareStatement(query);
st.executeUpdate(query);
```

```
PreparedStatement st1 = con.prepareStatement(query1);
st1.executeUpdate(query1);
st1.close();
st.close();
con.close();%>
<%String redirectURL = "view.jsp";
response.sendRedirect(redirectURL); %>
</body>
</html>
```

CONCLUSION

Nowadays, it is a manual process for updating details and bigger registers consumes more and more time and incorrect calculation leads military entrepreneur in huge loss. To overcome this, to save time as well as money with correct automation of customer measurement detail management and day-to-day updating of customer information and use talks of design, we go with the software application. It is simple and easy to use and and details are not accepted when not entered correctly.

LIMITATIONS

Although I have put my best efforts to make the software flexible easy to operate without limitations, it cannot be ruled out even by me. Though the software presents a broad range of options to you, to its users some intricate options could not be covered into it, partly because of logistic and partly due to lack of sophistication. Paucity of time was also a major constraint, thus it was not possible to make the software foolproof and dynamic. Lack of time also compelled me to ignore some parts as storing old result of the candidate etc.

BILBLIOGRAPHY

1. Welling, L. and Thomson, L., 2003. Java and MySQL Web development.
2. Arnold, Ken, James Gosling and David Holmes(2005). The Java programming Language, 3rd edition. NJ: Prentice Hall.
3. Sams Publishing. Cornelio, A. and Navathe, S.B., 1988, February. A tool for integrating conceptual schemas and user views. In Proc. IEEE Fourth Int'l Conf. Data Eng.(ICDE'88).

4 .Gillenson, M.L., 2008. Fundamentals of database management systems.

5. DuBois, P., 2008. MySQL. Pearson Education.

6.Core Java, MySql, and JavaScripts, www.w3schools.com