**Project Report**

**On**

**“STUDENT RESULT MANAGEMENT SYSTEM”**

**Submitted to:**

**Faculty of Computer Science & Applications**

**Graphic Era Hill University, BHIMTAL**

**In the partial fulfilment for the award of degree of Bachelor of ComputerScience**

**Submitted By: piyush palariya**

**Project Guide: Dr. Sudhanshu Maurya**



**Graphic Era Hill University, BHIMTAL**

NOTE

FOR DIRECTLY OPENING THE PROJECT VISIT

https://sresultmsystem.000webhostapp.com/Login.php

**Details of student  --**

1. Student name-- Doodle

Roll id--03

Student Class: BCA V

2. Student name-- Mamta

Roll id--04

Student Class: BCA VI

**For Admin Panel**

Login Details for

admin : admin

Password : student1@

**ACKNOWLEDGEMENT:**

It gives me a great sense of pleasure to present the report of the BscCS project undertaken during IV year of my graduation. There are many people who have helped me directly or indirectly in the successful completion of this project.I would like to take this opportunity to thank them all.

I am very thankful to my project mentor Dr. Sudhanshu Maurya who has been an inspiring guide and committed caretaker for his unflinching devotion. The encouragement and support by him, especially in carrying out this project motivated me much more to complete this project successfully.

I would like to thank all of my friends for their help and constructive criticism during the project period.

I owe a deep sense of gratitude to Dr. M.C Lohani, Honorable Director, for his keen interest on me at every stage of my project. His prompt inspirations, timely suggestions with kindness, enthusiasm and dynamism has enabled me to complete my project. I am also thankful to Graphic Era Hill University of for providing me an opportunity to work in excellent working environment together with required resources.

**PROJECT SUMMARY**

**Title:**Student Result Management System

**Definition**: To provide result in an organised manner.

**Developed At**:Graphic Era Hill University, Bhimtal

**Team Size**:Four Member

**Team Member**:

* + Chandraprakash
  + Piyush Palariya
  + Rajat Singh Samant
  + Rashmi Pathani

**CERTIFICATE**

This is to certify that this project entitled “**STUDENT RESULT MANAGEMENT SYSTEM**” in partial fulfilment for the award ofthe degree of **Bachelor in Computer Science** to the “Graphic Era Hill University” and is a record of my own investigation carriedunder the Guidance of Dr. Sudhanshu Maurya Sir. The matter embodied in this project work has not been submitted earlier for award of any degree or diploma to the best of my knowledge and belief.

**Signature of the students Signature of the Guide**

**Chandraprakash (1984001) Dr. Sudhanshu Maurya**

**Piyush Palariya ( 1984002)**

**Rajat Singh Samant ( 1984005)**

**Rashmi Pathani ( 1984006)**

**DECLARATION**

I hereby declare that the project work entitled “**STUDENT RESULT MANAGEMENT SYSTEM**” is being submitted to Faculty of Computer Science & Applications is the authentic record of our own project work done under the guidance of Dr. Sudhanshu Maurya Sir, Graphic Era Hill University, Bhimtal.

**Signature of Student:**

Piyush palariya

**Table of Content**

1. Abstract of the Project.
2. System Requirements
3. Product Definition
4. Product Statement
5. Functions Provided
6. Processing Environment
7. Feasibility Analysis
8. Project Plan
   1. Development Schedule
   2. Programming Language and Development Tools
9. System Requirements Specification
10. Developing/ Operating/ Maintenance Environment
11. Functional and Non-Functional Requirements
12. Data Dictionary
13. Design
14. Use Case
15. Data Flow Diagram
16. ER Diagram
17. Testing
18. Implementation
19. Project Legacy
    1. Current Status
    2. Future Recommendations
    3. Bibliography
20. Conclusion

**Abstract of the Project**

This project is aimed at developing a web application that depicts the management of student’s result of their examination. The main objective of developing this project is to automate student result management. This application will greatly simplify and speed up the result preparation and management process.

This project is useful for students and institutions for getting the examination result in a simple and accurate manner.

The system is intended for the students. And the privileges that are provided to students are to read and execute his/her result by providing their roll number.

This application will manage the information about various students in various subjects in different semesters. The manual method of students’ academic result processing was found to be tedious, especially when carried out for a large number of students, this makes the entire process time-consuming and error prone.

The whole result analyser will be under the control of the administrator as the admin has the full privilege to read, write and execute the result.

The aim of this project is to provide the examination result to the students in the simplest way.

**System Requirements**

**Product Definition:**

Student Result Management System is a technological opportunity for schools, colleges, universities and the coaching centre institutions searching for a secure, simple and alternative solution to the conventional paper-based exam results evaluation, reporting and distribution.

**Problem Statement:**

To develop a system that will manage the following points:

1. Information about the various Students.
2. Information about subjects offered in various semesters.
3. Information about students enrolled in various semesters.
4. Marks obtained by Students in various Semesters.
5. Generation of Result

**Function to be provided:**

* **Admin Functions/ Privileges**

1. He/ She is a super user who enjoys all the privileges.
2. Add Update Display Delete Semesters.
3. Add Update Display Delete Subjects and their Full Marks.
4. Create/Approve/Disapprove students account.
5. View Results of each student and their Percentage.
6. Update Records.

* **Student Functions/Privileges**

1. He/ She should have their rollno with them.
2. Can view their Semester Results for all Subjects from anywhere around the globe.

**Processing Environment:**

* **Software Requirements**

1. Microsoft Windows 7/8/10 or Linux.
2. XAMPP (MySQL, Apache, PHP)
3. Visual Studio Code or any other text editor
4. Chrome or any other web browser

* **Hardware Requirements**

1. Intel Processor 2.0 GHz or above.
2. 2 GB RAM or more.

**Feasibility Analysis:**

It is the process of confirming that a strategy, plan or design is possible and makes sense. This is usually used to validate assumptions, constraints, decisions, approaches and business cases.

1. **FINANCIAL FEASIBILITY:**

The project is financially feasible as the only cost involved is having a computer with the minimum requirements such as internet, electricity, etc.

For the users to access the application, the only cost involved will be in getting access to the Internet.

1. **TECHNICAL FEASIBILITY:**

Validating that a given technology can support requirements or that a goal is technically possible. It is basically used to see whether the existing system is sufficient or any further additional requirements will be required.

1. **OPERATIONAL FEASIBILITY:**

The feasibility of deploying and operating a project. Its basic work is to check that the software performs all the operations that are required by it.

User does not require any sort of technical knowledge or any special training to use the product.

**Project Plan**

* + 1. **Development Schedule**
    2. **Programming Languages and Development Tools**

**DEVELOPMENT SCHEDULE**

Steps used to develop a website:

* + - Creating a UI.
    - Scripting (Server and Client End).
    - Creating a Backend or Database.

**PROGRAMMING LANGUAGES AND DEVELOPMENT TOOLS:**

* **PHP**

1. **Definition of PHP:**

PHP can be defined as a programming language for Database access from the web’s browser. In other words, it is an HTML embedded scripting language. It focuses on the logic of how a page responds to user input and not how the page looks that i.e. not the primary appearance of the page.

PHP runs on the server side, which means that the web server that send an HTML file to a user’s browser, will carry out the instructions found in the embedded PHP code first, and then send the output of the PHP code along with the HTML code. The result is a webpage with dynamic content.

1. **History of PHP:**

PHP development began in 1994 when Rasmus Lerdorf wrote several Common Gateway Interface (CGI) programs in C, which he used to maintain his personal homepage. He extended them to work with web forms and to communicate with databases. He extended them to work with web forms and to communicate with databases, and called this implementation “Personal Home Page/Forms Interpreter” or PHP/FI.

PHP/FI could be used to build simple, dynamic web applications. To accelerate bug reporting and improve the code, Lerdorf initially announced the release of PHP/FI as “Personal Home Page Tools (PHP Tools) version 1.0” on the Usenet discussion group

Comp.infosystems.www.authoring.cgi on June 8, 1995. This release already had the basic functionality that PHP has today. This included Perl-like variables, form handling, and the ability to embed HTML.

**Example of PHP:**

1. <!DOCTYPE html>  
   <html>  
   <body>  
     
   <h1>My first PHP page</h1>  
     
   <?php  
   echo "Hello World!";  
   ?>  
     
   </body>  
   </html>
2. <!DOCTYPE html>  
   <html>  
   <body>  
     
   <?php  
   ECHO "Hello World!<br>";  
   echo "Hello World!<br>";  
   EcHo "Hello World!<br>";  
   ?>  
     
   </body>  
   </html>

* **HTML**

**HTML** stands for **Hyper Text Markup Language**, which is the most widely used language on Web to develop web pages. **HTML** was created by Berners-Lee in late 1991 but "HTML 2.0" was the first standard HTML specification which was published in 1995. HTML 4.01 was a major version of HTML and it was published in late 1999. Though HTML 4.01 version is widely used but currently we are having HTML-5 version which is an extension to HTML 4.01, and this version was published in 2012.

HTML (Hypertext Markup Language) is the set of markup symbols or codes inserted in a file intended for display on a World Wide Web browser page. The markup tells the Web browser how to display a Web page’s words and images for the user. Each individual markup code is referred to as an element (but many people also refer to it as a tag). Some elements come in pairs that indicate when some display effect is to begin and when it is to end.

**Applications of HTML**

As mentioned before, HTML is one of the most widely used language over the web. I'm going to list few of them here:

* **Web pages development** - HTML is used to create pages which are rendered over the web. Almost every page of web is having html tags in it to render its details in browser.
* **Internet Navigation** - HTML provides tags which are used to navigate from one page to another and is heavily used in internet navigation.
* **Responsive UI** - HTML pages now-a-days works well on all platform, mobile, tabs, desktop or laptops owing to responsive design strategy.
* **Offline support-** HTML pages once loaded can be made available offline on the machine without any need of internet.
* **Game development**- HTML5 has native support for rich experience and is now useful in gaming development arena as well.

**Example of HTML**

* 1. <!DOCTYPE html>

<html>

<head>

<!-- head definitions go here -->

</head>

<body>

<!-- the content goes here -->

</body>

</html>

* 1. <!DOCTYPE html>

<html>

<head>

<title>Page Title</title>

</head>

<body>

<h1>This is a Heading</h1>

<p>This is a paragraph.</p>

</body>

</html>

* **CSS**

Cascading Style Sheets (CSS) are a collection of rules we use to define and modify web pages. CSS are similar to styles in Word. CSS allow Web designers to have much more control over their pages look and layout. For instance, you could create a style that defines the body text to be Verdana, 10 point. Later on, you can easily change the body text to Times New Roman, 12 points by just changing the rule in the CSS. Instead of having to change the font on each page of your website, all you need to do is redefine the style on the style sheet, and it will instantly change on all the pages that the style sheet has been applied to. With HTML styles, the font change would be applied to each instance of that font and have to be changed in each spot.

CSS can control the placement of text and objects on your pages as well as the look of those objects.

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, and reduce complexity and repetition in the structural content.

Separation of formatting and content also makes it feasible to present the same markup page in different styles for different rendering methods, such as on-screen, in print, by voice (via speech-based browser or screen reader), and on Braille-based tactile devices. CSS also has rules for alternate formatting if the content is accessed on a mobile device.

**Example of CSS**

1. body {

background-color: lightblue;

}

h1 {

color: white;

text-align: center;

}

p {

font-family: verdana;

font-size: 20px;

}

1. \*{

margin: 0;

padding: 0;

box-sizing: border-box;

list-style: none;

text-decoration: none;

font-family: 'josefin sans', sans-serif;

}

body{

background-color: darkslategray;

width: 100%;

height: 100%;

}

* **JavaScript**

JavaScript is a programming language commonly used in web development. It was originally developed by Netscape as a means to add dynamic and interactive elements to websites. While JavaScript is influenced by Java, the syntax is more similar to C and is based on ECMA Script, a scripting language developed by Sun Microsystems.

JavaScript is a client-side scripting language, which means the source code is processed by the client’s web browser rather than on the web server. This means JavaScript functions can run after a webpage has loaded without communicating with the server. For Example, a JavaScript function may check a web form before it is submitted to make sure all the required fields have been filled out. The JavaScript code can produce an error message before any information is actually transmitted to the server.

As a multi-paradigm language, JavaScript supports event-driven, functional, and imperative programming styles. It has application programming interfaces (APIs) for working with text, dates, regular expressions, standard data structures, and the Document Object Model (DOM). However, the language itself does not include any input/output (I/O), such as networking, storage, or graphics facilities, as the host environment (usually a web browser) provides those APIs.

JavaScript engines were originally used only in web browsers, but they are now embedded in some servers, usually via Node.js. They are also embedded in a variety of applications created with frameworks such as Electron and Cordova.

Although there are similarities between JavaScript and Java, including language name, syntax, and respective standard libraries, the two languages are distinct and differ greatly in design.

**Example of JavaScript**

1. function factorial(n) {

if (n === 0)

return 1; // 0! = 1

return n \* factorial(n - 1);

}

factorial(3); // returns 6

1. let counter = (function() {

let i = 0; // private property

return { // public methods

get: function() {

alert(i);

},

set: function(value) {

i = value;

},

increment: function() {

alert(++i);

}

};

})(); // module

counter.get(); // shows 0

counter.set(6);

counter.increment(); // shows 7

counter.increment(); // shows 8

* **XAMPP**

XAMPP is an abbreviation for cross-platform, Apache, MySQL, PHP and Perl, and it allows you to build WordPress site offline, on a local web server on your computer. This simple and lightweight solution works on Windows, Linux, and Mac – hence the “cross-platform” part.

XAMPP is a compilation of free software (comparable to a Linux distribution), it's free of charge and it's free to copy under the terms of the GNU General Public Licence. But it is only the compilation of XAMPP that is published under GPL. Please check every single licence of the contained products to get an overview of what is, and what isn't, allowed. In the case of commercial use please take a look at the product licences (especially MySQL), from the XAMPP point of view commercial use is also free.

The use of XAMPP is to test the clients or your website before uploading it to the remote web server. This XAMPP server software gives you the suitable environment for testing MYSQL, PHP, Apache and Perl projects on the local computer.

**BOOTSTRAP**

Bootstrap is a free and open-source CSS framework directed at responsive, mobile-first front-end web development. It contains CSS- and (optionally) JavaScript-based design templates for typography, forms, buttons, navigation, and other interface components.

Bootstrap is the sixth-most-starred project on GitHub, with more than 135,000 stars, behind freeCodeCamp (almost 307,000 stars) and marginally behind Vue.js framework. According to Alexa Rank, Bootstrap is in the top-2000 in the USA while vuejs.org is in the top-7000 in the USA.

Bootstrap, originally named Twitter Blueprint, was developed by Mark Otto and Jacob Thornton at Twitter as a framework to encourage consistency across internal tools. Before Bootstrap, various libraries were used for interface development, which led to inconsistencies and a high maintenance burden. According to twitter developer Mark Otto:

A super small group of developers and I got together to design and build a new internal tool and saw an opportunity to do something more. Through that process, we saw ourselves build something much more substantial than another internal tool. Months later, we ended up with an early version of Bootstrap as a way to document and share common design patterns and assets within the company.

After a few months of development by a small group, many developers at Twitter began to contribute to the project as a part of Hack Week, a hackathon-style week for the Twitter development team. It was renamed from Twitter Blueprint to Bootstrap, and released as an open source project on August 19, 2011. It has continued to be maintained by Mark Otto, Jacob Thornton, and a small group of core developers, as well as a large community of contributors.

**System Requirement Specifications:**

* **Developing/ Operating/ Maintenance Environment**

1. **Developing Environment:**

During the development phase of validation of an automation system is the phase where a new system is specified, build, verified and handed over to the system owner.

1. **Operating Environment:**

After a successful developing phase a validated system is available. The operation phase of validation will ensure that the validated state of the system is maintained during the life cycle of the system.

1. **Maintenance Environment:**

The maintenance phase of the software occurs after the product is in full operation. During this phase, errors or defects may exist, which would require repairs during additional testing of the software. Monitoring the performance of the software is also included during the maintenance phase.It also includes making changes to improve a system’s performance, correct problems, enhance security, or address user requirements.

* **Functional Requirements**

Once the registered Student enters the valid rollnowhich is given by the administrator, he/she will be provided with the following services.

1. View Student Details
2. View the marks obtained by them.
3. View Result if declared.
4. Back to Home Page.

On clicking these options we navigate to respective pages.

**View Student Details:**

This will show the students their details filled by the Administrator, during the time of registration.

**View Marks Obtained:**

This will show the students their marks according to their semesters.

**View Result if declared:**

This will show the student their Result including percentage, marks in each subjects and their total marks.

**Back to Home Page:**

This will take the user back to the main page.

* **Non-Functional Requirements**

The non-functional requirements the system should satisfy are as follows:

1. **Security Issues:** In this system unauthorized users cannot access the data and cannot perform any operation, because the system does not allow them to login. So security is provided.
2. **User Interface:** The system is designed in such a way that instructions are given clearly to navigate through the System. Warnings and error messages are provided throughout the system.
3. **Scalability:** The system is scalable in the sense, it supports large amounts of data that can be stored and retrieved.

* **DATA DICTIONARY**

A data dictionary is a file or a set of files that includes a database's metadata. The data dictionary hold records about other objects in the database, such as data ownership, data relationships to other objects, and other data.

The data dictionary is an essential component of any relational database. Ironically, because of its importance, it is invisible to most database users. Typically, only database administrators interact with the data dictionary.

The data dictionary in general contains information about the following −

1. Names of all the database tables and their schemas.
2. Details about all the tables in the database, such as their owners, their security constraints, when they were created etc.
3. Physical information about the tables such as where they are stored and how.
4. Table constraints such as primary key attributes, foreign key information etc.
5. Information about the database views that are visible.

* **Tables used in the database:**

1. **Admin Table**

## Table structure for table admin

|  |  |  |  |
| --- | --- | --- | --- |
| **Column** | **Type** | **Null** | **Default** |
| ***Id*** | int(11) | No |  |
| username | varchar(100) | Yes | NULL |
| password | varchar(100) | Yes | NULL |
| updation date | Date | Yes | NULL |

1. **Classes Table**

## Table structure for table tblclasses

|  |  |  |  |
| --- | --- | --- | --- |
| **Column** | **Type** | **Null** | **Default** |
| ***Id*** | int(11) | No |  |
| ClassName | varchar(80) | Yes | NULL |
| ClassNumericValue | int(4) | Yes | NULL |
| Section | varchar(5) | Yes | NULL |
| CreationDate | date | Yes | current\_timestamp() |
| UpdationDate | date | Yes | NULL |

1. **Result Table**

## Table structure for table tblresult

|  |  |  |  |
| --- | --- | --- | --- |
| **Column** | **Type** | **Null** | **Default** |
| ***Id*** | int(11) | No |  |
| StudentId | int(11) | Yes | NULL |
| ClassId | int(11) | Yes | NULL |
| SubjectId | int(11) | Yes | NULL |
| Marks | int(11) | Yes | NULL |
| PostingDate | date | No | current\_timestamp() |

1. **Students Table**

## Table structure for table tblstudents

|  |  |  |  |
| --- | --- | --- | --- |
| **Column** | **Type** | **Null** | **Default** |
| ***StudentId*** | int(11) | No |  |
| StudentName | varchar(100) | Yes | NULL |
| RollId | varchar(100) | Yes | NULL |
| StudentEmail | varchar(100) | Yes | NULL |
| Gender | varchar(10) | Yes | NULL |
| DOB | varchar(100) | Yes | NULL |
| ClassId | int(11) | Yes | NULL |
| RegDate | date | No | current\_timestamp() |
| UpdationDate | date | Yes | NULL |
| Status | int(1) | Yes | NULL |

1. **Subject Combination**

## Table structure for table tblsubjectcombination

|  |  |  |  |
| --- | --- | --- | --- |
| **Column** | **Type** | **Null** | **Default** |
| ***Id*** | int(11) | No |  |
| Classid | int(11) | Yes | NULL |
| Subjectid | int(11) | Yes | NULL |
| Status | int(1) | Yes | NULL |
| CreationDate | date | No | current\_timestamp() |
| UpdationDate | date | Yes | NULL |

1. **Subjects**

## Table structure for table tblsubjects

|  |  |  |  |
| --- | --- | --- | --- |
| **Column** | **Type** | **Null** | **Default** |
| ***Id*** | int(11) | No |  |
| SubjectName | varchar(100) | No |  |
| SubjectCode | varchar(100) | Yes | NULL |
| CreationDate | date | No | current\_timestamp() |
| UpdationDate | date | Yes | NULL |

**DESIGN**

1. **Use Case Diagram**
2. **Data Flow Diagram**
3. **ER-Diagram**

**USE CASE DIAGRAM:**

1. Use case diagram consist of use cases and actors and shows the interaction between them.
2. The main purpose is to show the interaction between the use cases and the actors.
3. To represent the system requirements from user’s perspective.
4. The use cases are the functions that are to be performed in the module.

## Purpose of Use Case Diagrams

The purpose of use case diagram is to capture the dynamic aspect of a system. However, this definition is too generic to describe the purpose, as other four diagrams (activity, sequence, collaboration, and State chart) also have the same purpose. We will look into some specific purpose, which will distinguish it from other four diagrams.

Use case diagrams are used to gather the requirements of a system including internal and external influences. These requirements are mostly design requirements. Hence, when a system is analysed to gather its functionalities, use cases are prepared and actors are identified.

When the initial task is complete, use case diagrams are modelled to present the outside view.

In brief, the purposes of use case diagrams can be said to be as follows −

* Used to gather the requirements of a system.
* Used to get an outside view of a system.
* Identify the external and internal factors influencing the system.
* Show the interaction among the requirements are actors.

**ADMIN SYSTEM**

**Use Case Diagram between Admin and System**

**Data Flow Diagram**

A Data Flow Diagram (DFD) is a graphical representation of the “flow” of data through an Information System. A data flow diagram can also be used for the visualization of Data Processing. It is common practice for a designer to draw a context-level DFD first which shows the interaction between the system and outside entities. This context-level DFD is then “exploded” to show more detail of the system being modelled.

A DFD represents flow of data through a system. Data flow diagrams are commonly used during problem analysis. It views a system as a function that transforms the input into desired output. A DFD shows movement of data through the different transformations or processes in the system.

Dataflow diagrams can be used to provide the end user with a physical idea of where the data they input ultimately has an effort upon the structure of the whole system from order to dispatch to restock how any system is developed can be determined through a dataflow diagram. The appropriate register saved in database and maintained by appropriate authorities.

**Data Flow Diagram Notation**

**Function**

**File/Database**

**Input/Output**

**Flow**

**DFD 0-Level**

**Request**

**Request**

**Student**

**ADMIN**

**Response**

**Response**

**DFD 1-Level ADMIN SIDE**

**Login D/B**

**ADMIN**

**Class D/B**

**Subject D/B**

**Student D/B**

**Result D/B**

**User D/B**

**Admin D/B**

**DFD 1-Level STUDENT SIDE**

**Result D/B**

**Login D/B**

**STUDENT**

**DFD 2-Level Admin Side**

**Class Management**

**DFD 2-Level Admin Side**

**ADMIN**

**Classes D/B**

**Admin D/B**

**Subject Management**

**DFD 2-Level Admin Side**

**Subjects D/B**

**ADMIN**

**Admin D/B**

**Student Management**

**Students D/B**

**ADMIN**

**Admin D/B**

**DFD 2-Level Admin Side**

**Result Management**

**ADMIN**

**Result D/B**

**Admin D/B**

**DFD 2-Level ADMIN SIDE**

**USER MANAGEMENT**

**ADMIN**

**ADMIN D/B**

**User D/B**

**DFD 2-Level ADMIN SIDE**

**CHANGE PASSWORD**

**ADMIN D/B**

**ADMIN**

**Entity Relationship Diagram (ER-Diagram):**

An entity-relationship (ER) diagram is a specialized graphic that illustrates the interrelationships between entities in a database. ER diagrams often use symbols to represent three different types of information. Boxes are commonly used to represent entities. Diamonds are normally used to represent relationships and ovals are used to represents attributes.

An entity-relationship model (ERM) in software engineering is an abstract and conceptual representation of data. Entity-relationship modelling is a relational schema database modelling method, used to produce a type of conceptual schema or semantic data model of a system, often of a system, often a relational database, and its requirements in a top-down fashion.

**Symbols used in this ER-Diagram:**

**Entity:** Entity is a “thing” in the real world with an independent existence. An entity may be an object with a physical existence such as person, car or employee. Entity symbol is as follows.

**Attributes:** Attributes are the properties of entities. Attributes are represented by means of ellipses. Every ellipse represents one attribute and is directly connected to its entity (rectangle). Attribute symbol is as follows.

**Types of Attributes**

1. **Composite Attribute:**If the attributes are **composite**, they are further divided in a tree like structure. Every node is then connected to its attribute. That is, composite attributes are represented by ellipses that are connected with an ellipse. Attribute symbol is as follows.
2. **Multivalued Attribute:** These attributes are depicted by double ellipse. Attribute symbol is as follows.
3. **Derived Attribute:** Theseattributes are depicted by dashed ellipse. Attribute symbol is as follows.

Manages

Admin

Class id

Student Id

Subject Combination

Id

Subject

Class

Result

Student

**TESTING**

Testing is a process of executing a program with the intent of finding an error. Testing is a crucial element of software quality assurance and presents ultimate review of specification, design and coding.

System Testing is an important phase. Testing represents an interesting anomaly for the software. Thus a series of testing are performed for the proposed system before the system is ready for user acceptance testing.

A good test case is one that has a high probability of finding an undiscovered error. A successful test is one that uncovers an undiscovered error.

**Testing Principles:**

* All tests should be traceable to end user requirements.
* Tests should be planned long before testing begins.
* Testing should begin on a small scale and progress towards testing in large.
* Exhaustive testing is not possible.
* To be most effective testing should be conducted by an independent third party.

**Test Case Design**

**White Box Testing:**

White box testing is a testing case design method that uses the control structure of the procedure design to derive test cases. All independents path in a module are exercised at least once, all logical decisions are exercised at once, execute all loops at boundaries and within their operational bounds exercise internal data structure to ensure their validity. Here the customer is given three chances to enter a valid choice out of the given menu. After which the control exits the current menu.

**Black Box Testing:**

Black Box Testing attempts to find errors in following areas or categories, incorrect or missing functions, interface error, errors in data structures, performance error and initialization and termination error. Here all the input data must match the data type to become a valid entry.

**Unit Testing:**

Unit testing is essentially for the verification of the code produced during the coding phase and the goal is test the internal logic of the module/program. In the Generic code project, the unit testing is done during coding phase of data entry forms whether the functions are working properly or not. In this phase all the drivers are tested they are rightly connected or not.

**Integration Testing:**

All the tested modules are combined into sub systems, which are then tested. The goal is to see if the modules are properly integrated, and the emphasis being on the testing interfaces between the modules. In the generic code integration testing is done mainly on table creation module and insertion module.

**Validation Testing:**

This testing concentrates on confirming that the software is error-free in all respects. All the specified validations are verified and the software is subjected to hard-core testing. It also aims at determining the degree of deviation that exists in the software designed from the specification; they are listed out and are corrected.

**WEB DEVELOPMENT TESTING:**

Web application testing, a software testing technique exclusively adopted to test the applications that are hosted on web in which the application interfaces and other functionalities are tested.

**TESTING TECHNIQUES:**

* 1. **Functionality Technique**: The below are some of the checks that are performed but not limited to the below list:
* Verify there is no dead page or invalid redirects.
* First check all the validations on each field.
* Wrong inputs to perform negative testing.
* Verify the workflow of the system.
* Verify the data integrity.
  1. **Usability Testing**: To verify how the application is easy to use with.
* Test the navigation and controls.
* Content checking.
* Check for user intuition
  1. **Interface Testing**: Performed to verify the interface and the dataflow from one system to other.
  2. **Compatibility Testing**: Compatibility testing is performed based on the context of the application.
* Browser compatibility
* Operating system compatibility
* Compatible to various devices like notebook, mobile, etc.
  1. **SYSTEM TESTING:**

This testing is a series of different tests whose primary is to fully exercise the computer-based system. This involves

Implementing the system in a simulated production environment and testing it.

Introducing errors and testing for error handling.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No.** | **Test Case Title** | **Description** | **Expected Outcome** | **Result** |
| 1 | Successful User Verification | The login to the system should be tried with the given roll Id assigned by the admin. | Login should be successful and the user should be able to see his/her result. | Successful |
| 2 | Unsuccessful User Verification due to wrong roll no | Login to the system with a wrong roll Id. | Login should fail with an error ‘Invalid Roll Id’ | Successful |
| 3 | Unsuccessful User Type due to non-selection of User Type | Login to the system without Selecting a valid user type. | Login Should Fail with an error ‘Invalid user Type’ | Successful |
| 4 | Successful Page Navigation | Navigation of Pages after Clicking ‘Search’ option to the Result Page. | It Should Display a Result page for the registered users. | Successful |
| 5 | Successful Email Validation | Validation of Proper Email while Registration is needed because further Communication is based on Email | Registration Should fail and display error message ‘Invalid Email is Entered’ | Successful |
|

**IMPLEMENTATION:**

The implementation phase involves installing approved application into production environments, primary tasks include announcing the implementation schedule, training end users, and installing the product. Additionally, organization should input and verify data, configure and test system and security parameters, and conduct post implementation reviews.

Management should circulate implementation schedules to all affected parts and should notify users about implementation responsibilities.

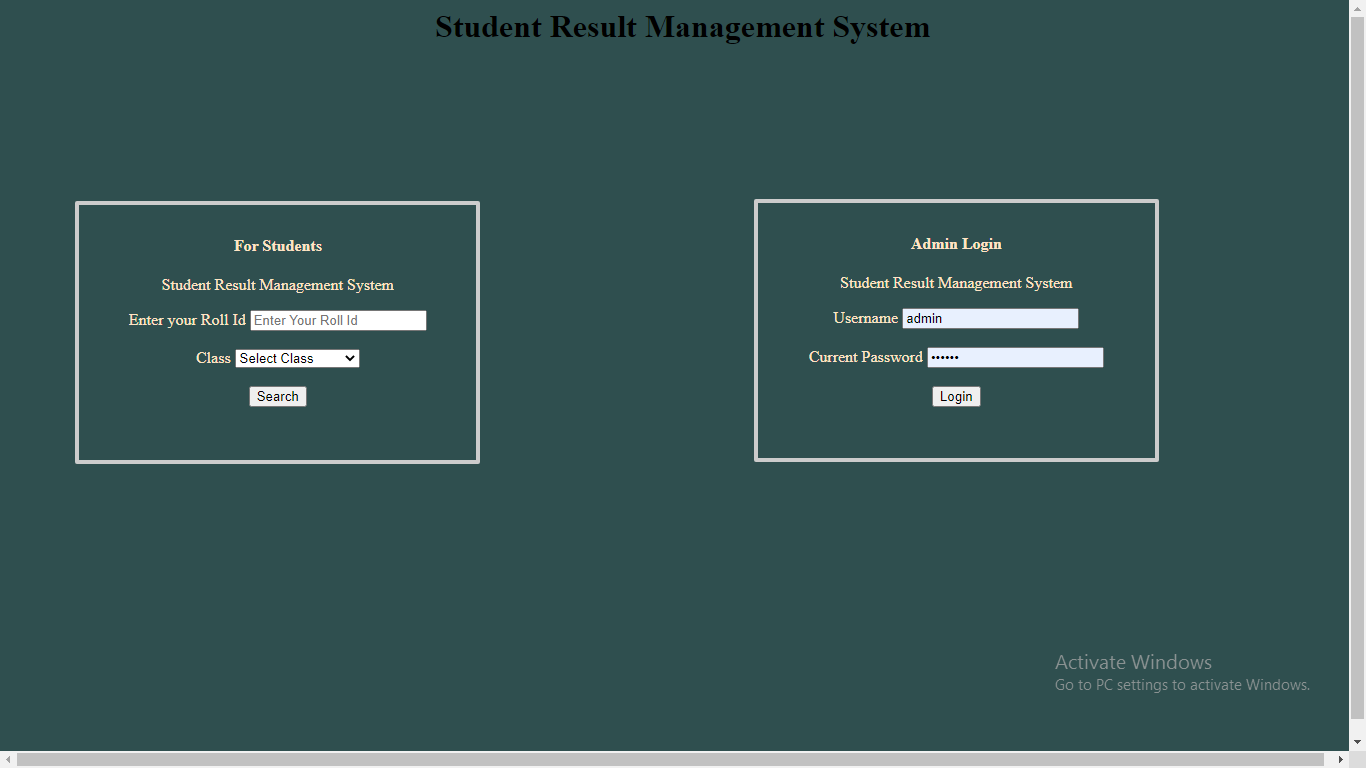
After organization installs the product, pre-existing data is manually input or electronically transferred to a new system. Verifying the accuracy of the input data and security configurations is a critical part of the implementation process.

**Forimplementationof the website project:**

1. The website can be installed on a computer or a server which has Php and MYSQL installed in it.
2. The owner of the website should be properly trained to use all the features of the website, giving details of each features of the website.
3. To show the accuracy of the website and conformance of the website to the requirements of the owner or users of the website.

**PAGE LAYOUT**

1. **Login Page**

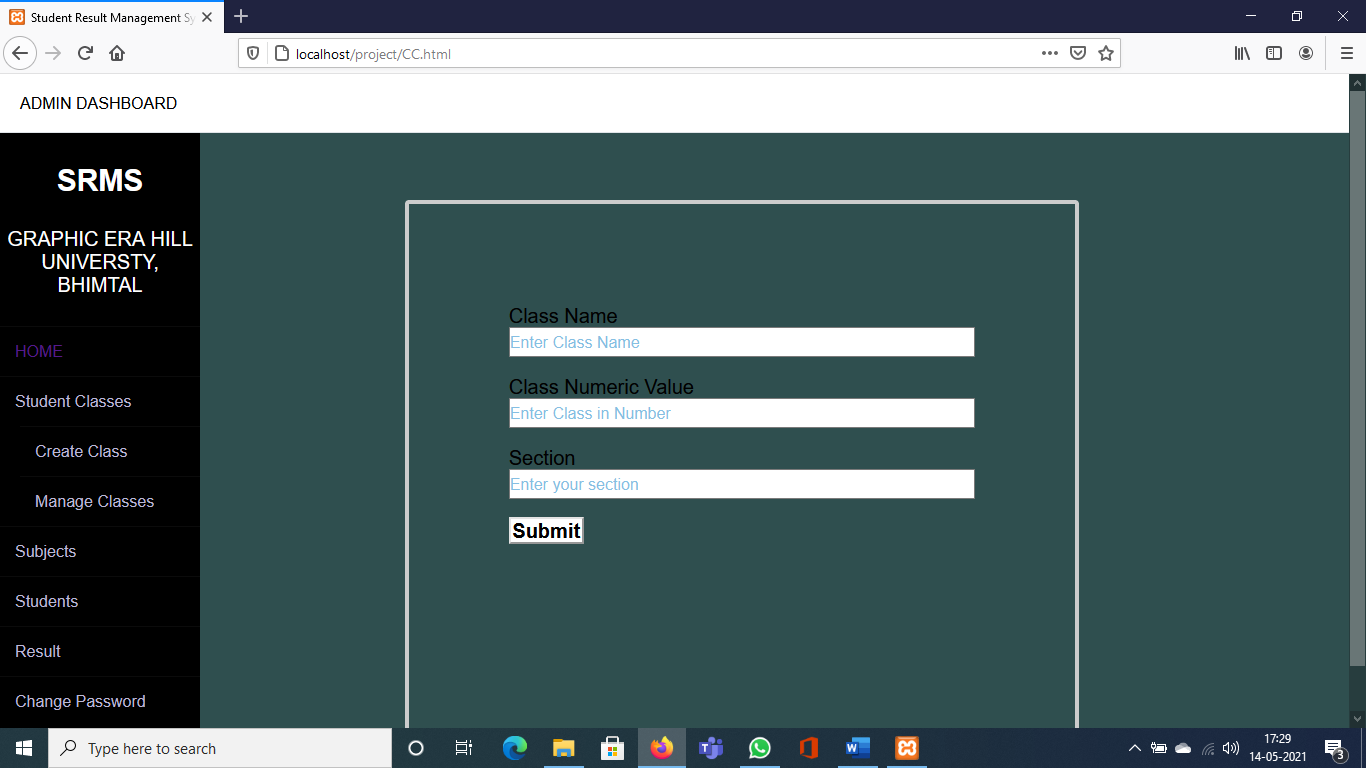
****

1. **Admin Page**

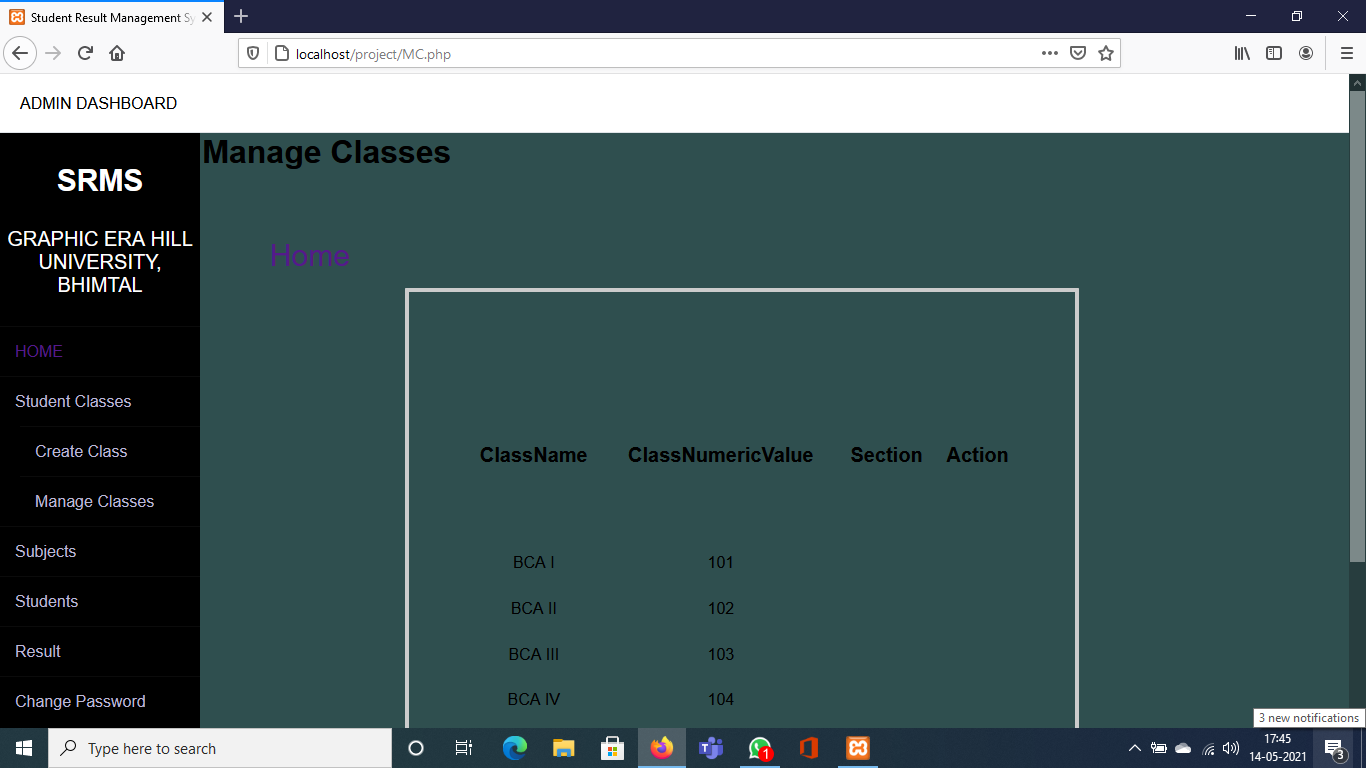
Graphical user interface, text

Description automatically generated

1. **Add Class**

****

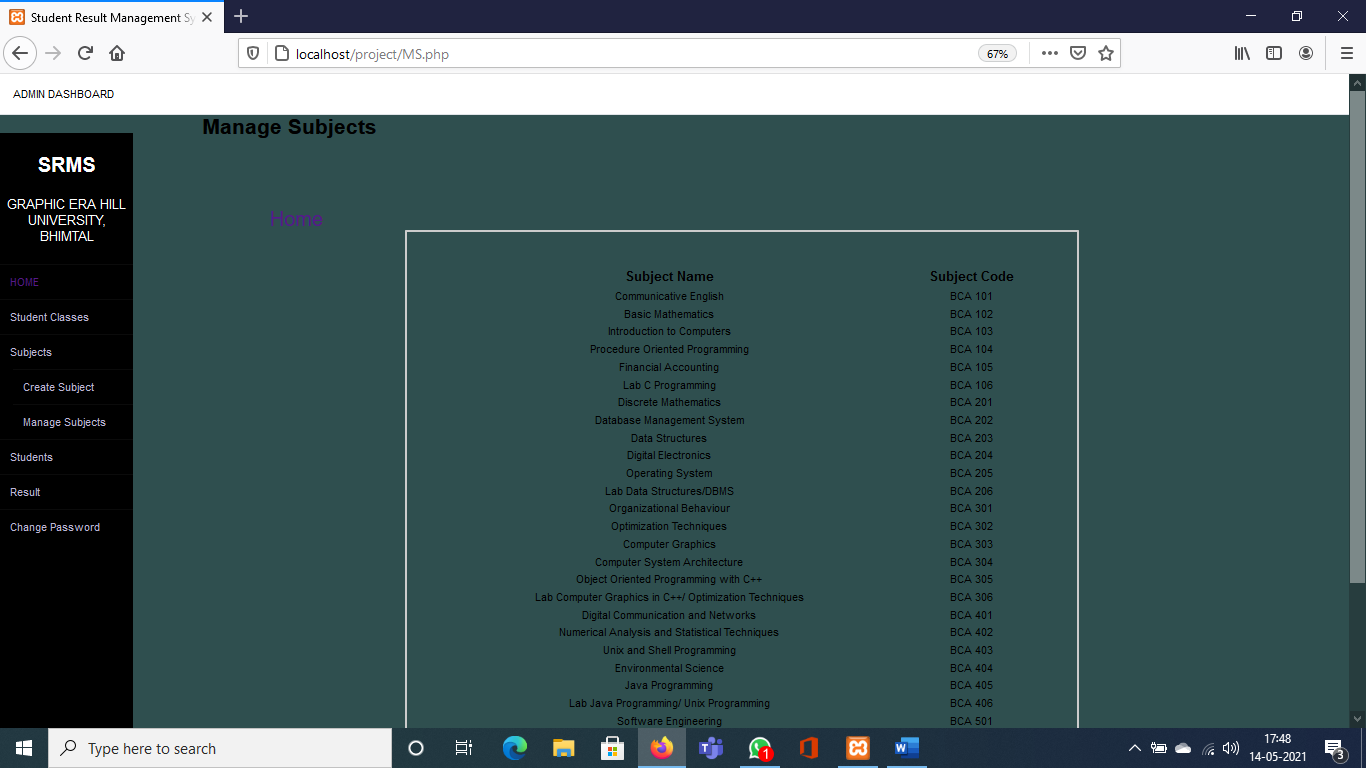
1. **Manage Class**

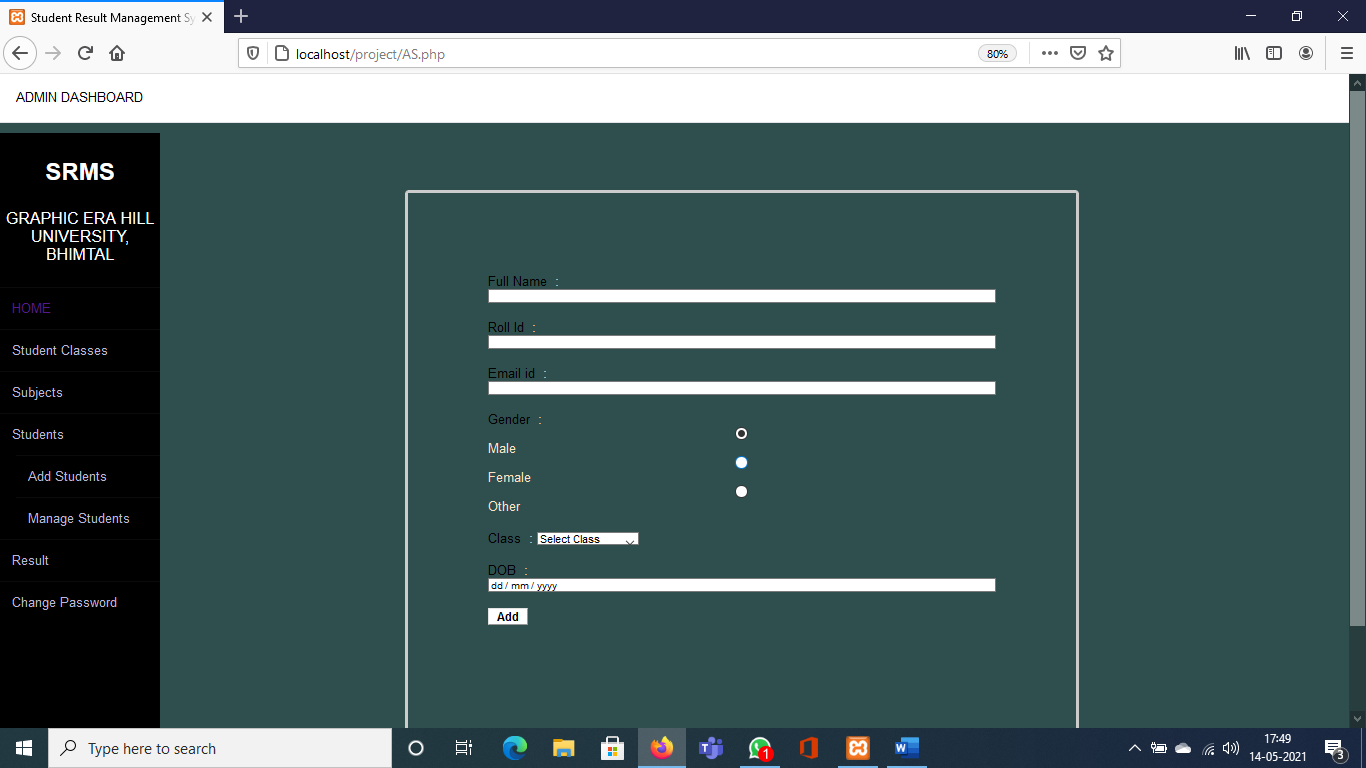
****

1. **Add Subject**

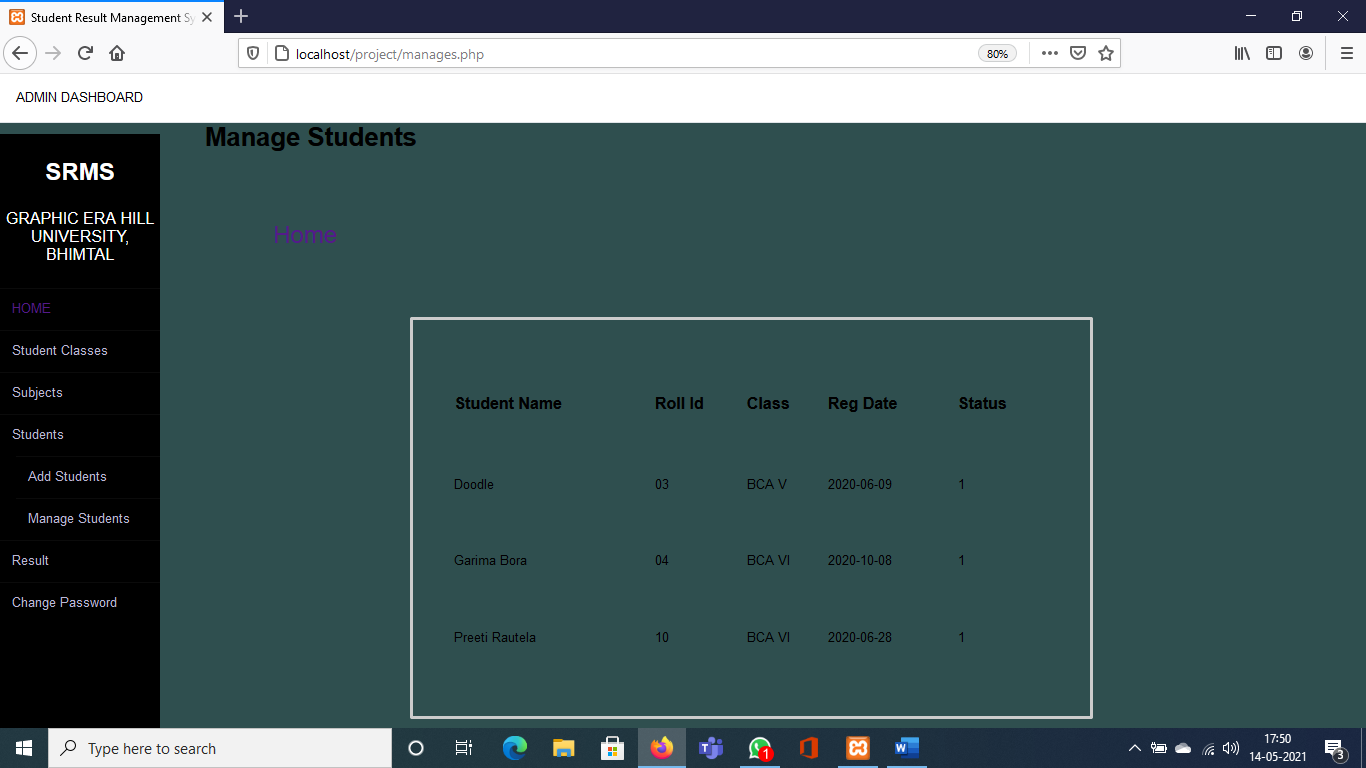
**A screenshot of a computer

Description automatically generated**

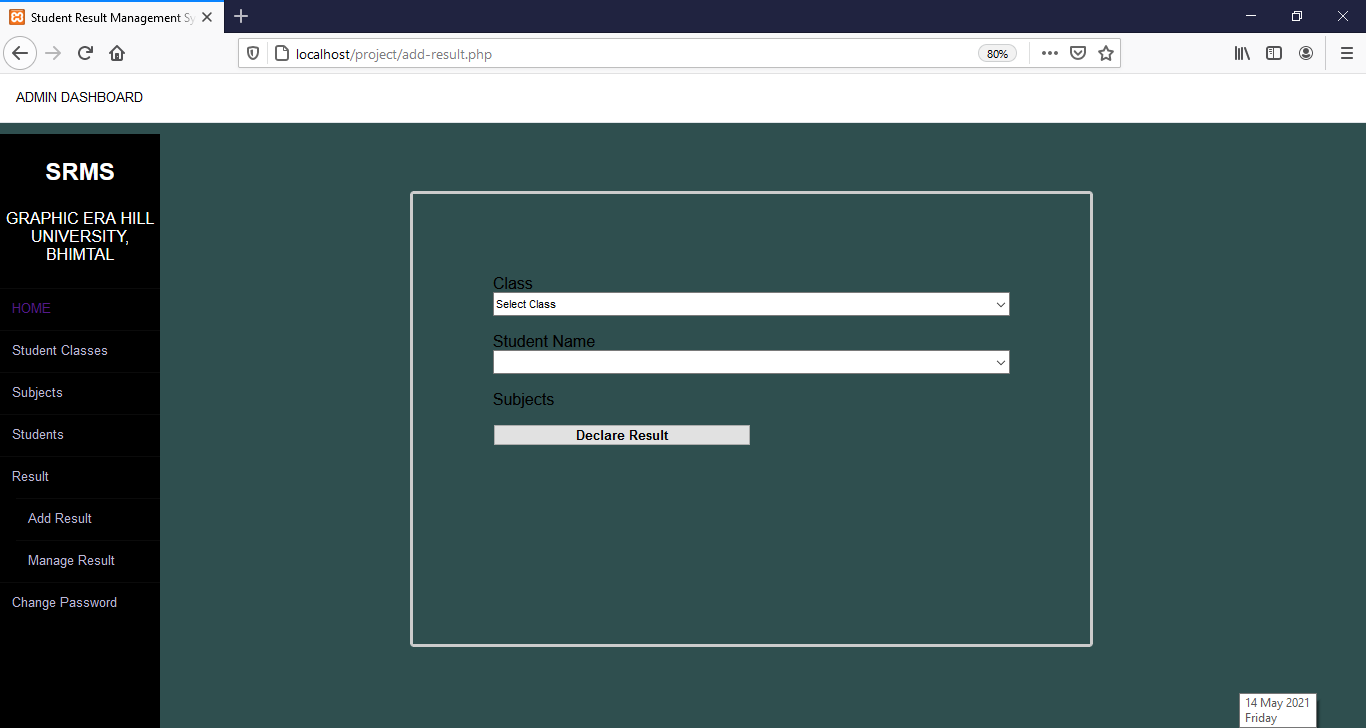
1. **Manage Subject**
2. **Add Students**

****

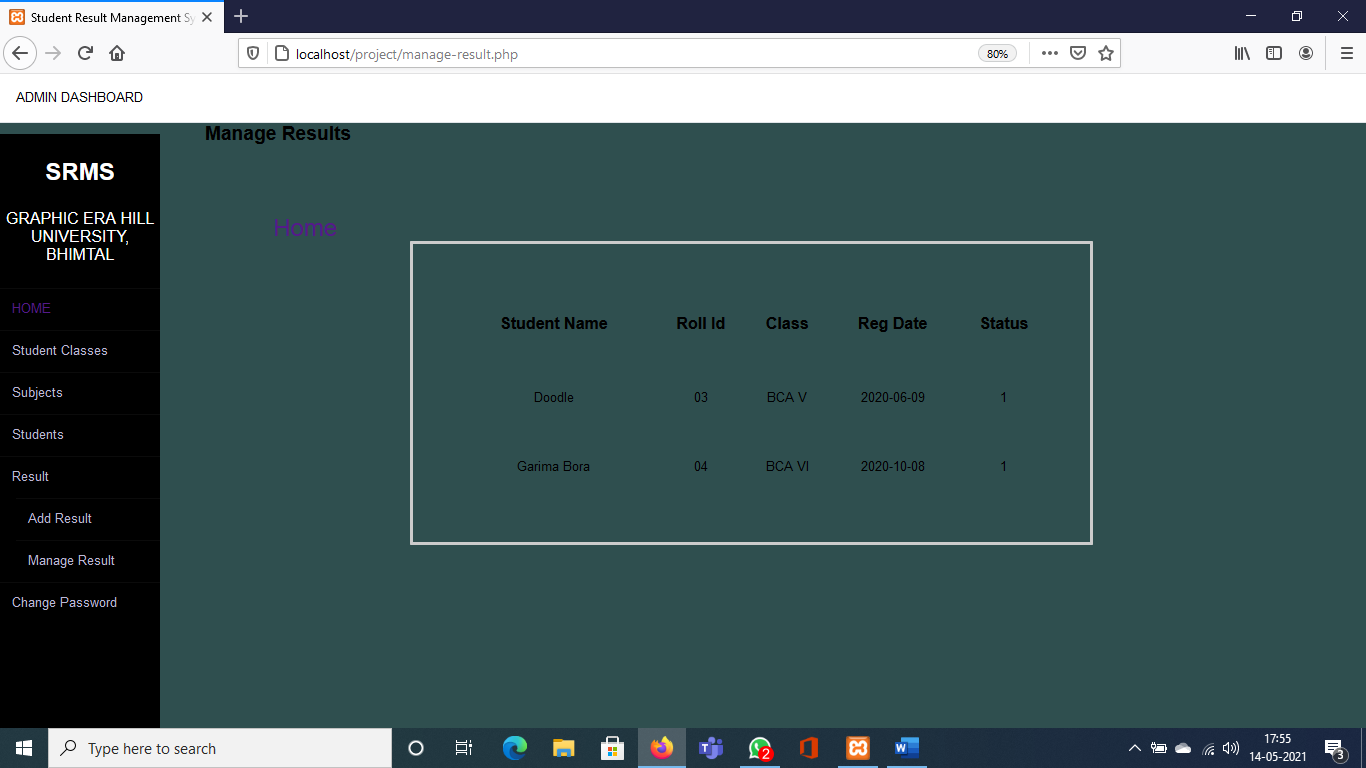
1. **Manage Students**

****

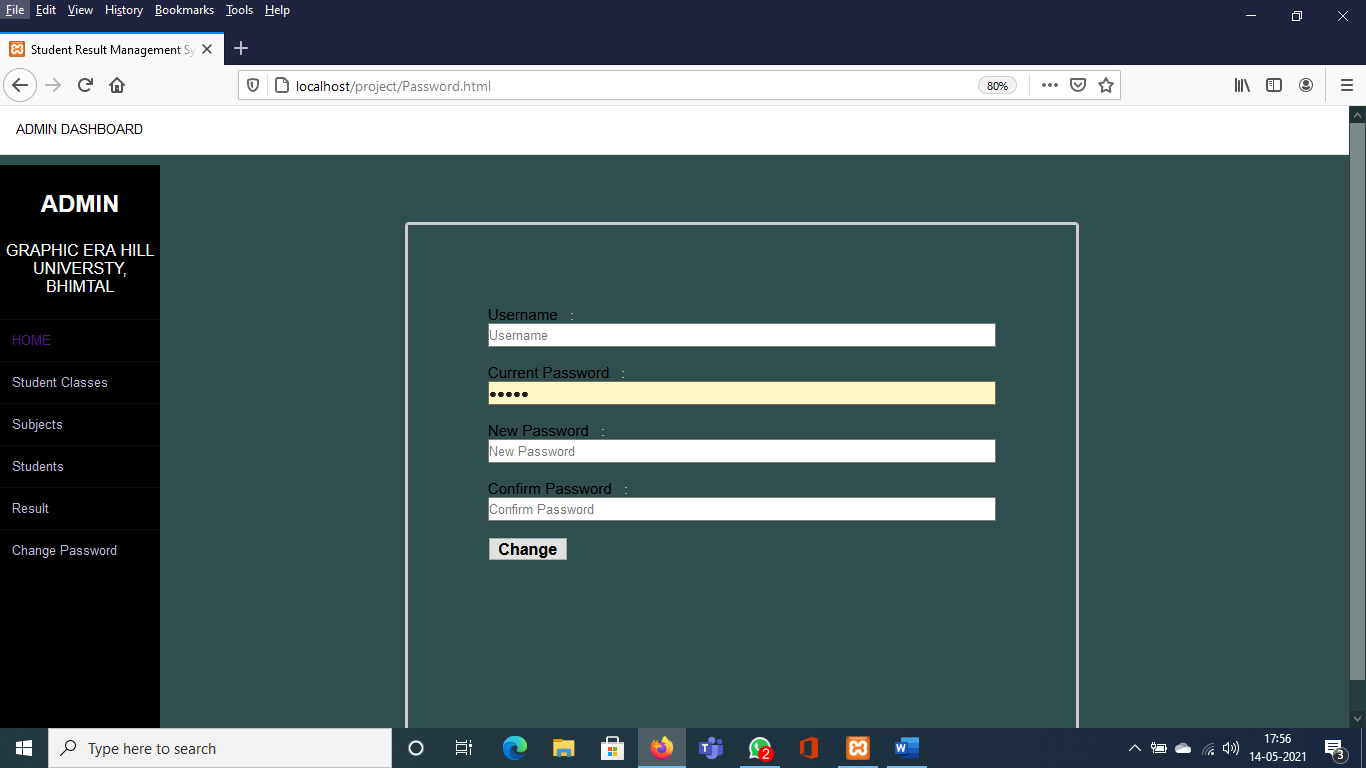
1. **Add Result**



1. **Manage Result**

****

1. **Change Password**

****

**PROJECT LEGACY**

1. **Current Status**
2. **Future Recommendations**
3. **Bibliography**

**Current Status of the Project:**

The current status of this project is at its final stage, where including designing, Front-end and Back-end process is completed successfully.

At this stage the website is well-designed and is completely ready to work in force.

**Future Recommendations:**

Previously, data used to be saved manually and include much more paper work. But, currently this project supports to inherit data and show the result in the mean time without extra manual effort. Students are not needed to visit college for viewing there results they can view it in their respective places.

The future scope is that data can be fetched, parsed in other formats like csv, odt etc. Visualisation can be provided to represent data in graphical format. Various representation like pie-chart, graph, etc.

**Bibliography:**

[www.Google.com](http://www.Google.com)

[www.javapoint.com](http://www.javapoint.com)

[www.slideshare.com](http://www.slideshare.com)

**CONCLUSION:**

This package was designed in such a way that future modifications can be done easily.

The following conclusions can be deduced from the development of this project:

* 1. Automation of the entire system improves the efficiency.
  2. It provides a friendly graphical user interface which proves to be better when compared to the existing system.
  3. It gives appropriate access to the authorized users depending on their permission.
  4. It efficiently overcomes the delay in communication.
  5. Updating of information becomes so easier.
  6. System security, data security and reliability are the striking features.
  7. The system has adequate scope for modification in future if it is necessary.