BAIUST RESULT SYSTEM

A Project Submitted

By
Fatema Jahan Jinia
ID: 1103071
MD. Tariqul Islam Ridoy
ID: 1103038
Rafid Shadman
ID: 1103097

Under the Supervision of

Md. Rifat Hossain
Lecturer
Department of Computer Science and Engineering
Bangladesh Army International University of Science and Technology

Bachelor of Science in Computer Science and Engineering

Bangladesh Army International University of Science and Technology

Fall, 2019 Semester

BAIUST RESULT SYSTEM

A Project submitted to the Department of Computer Science and Engineering Faculty, Bangladesh Army International University of Science and Technology in partial fulfillment of the requirements for the degree of Bachelor of Science in Computer Science and Engineering.

Fatema Jahan Jinia	**ID: 1103071**
Md. Tariqul Islam Ridoy	**ID: 1103038**
Rafid Shadman	**ID: 1103097**

Department of
Computer Science and Engineering

Bangladesh Army International University of Science and Technology

Fall Semester 2019

June, 2020

Declaration

This is to certify that this project is my original work. No part of this work has been submitted elsewhere partially or fully for the award of any other degree or diploma. Any material reproduced in this project has been properly acknowledged.

Student's name & signature

_ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _

Fatema Jahan Jinia

_ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _

Md. Tariqul Islam Ridoy

_ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _

Rafid Shadman

Approval

The Project titled "**BAIUST Result System**" has been submitted to the following respected members of the Board of Examiners in partial fulfillment of the requirements for the degree of Bachelor of Science in Computer Science and Engineering on June 2020 by the following student and has been accepted as satisfactory.

Authors

Fatema Jahan Jinia	**ID: 11003071**
Md. Tariqul Islam Ridoy	**ID: 11003038**
Rafid Shadman	**ID: 11003097**

_ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _

Supervisor
Md Rifat Hossain
Lecturer
Department of CSE
Bangladesh Army International University of Science and Technology

_ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _

External Supervisor
Dr. Fakir Sharif Hossain
Assistant Professor
Department of EEE
Bangladesh Army International University of Science and Technology

_ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _

Mohammad Asaduzzaman Khan

Head, Department of CSE

Bangladesh Army International University of Science and Technology

Acknowledgement

First and foremost, I would like to thank God Almighty for giving me the strength to finish this work. The satisfaction that accompanies the successful completion of this Project would be incomplete without the mention of people whose ceaseless cooperation made it possible, whose constant guidance and encouragement crown all efforts with success. I am grateful to my honorable project Supervisor **Md. Rifat Hossain, Lecturer**, Department of Computer Science and Engineering, Bangladesh Army International University of Science & Technology (BAIUST), for the guidance, inspiration and constructive suggestions which were helpful in the preparation of this project. I also convey special thanks and gratitude to **Mohammad Asaduzzaman Khan**, honorable head of the Department of Computer Science and Engineering, Bangladesh Army International University of Science & Technology (BAIUST), for his kind advice. I would also like to extend my gratitude to all of my teachers for their valuable guidance in every step of mine during the four years of learning stage. Finally I would like to thank my friends, senior, juniors and the staffs of the department for their valuable suggestion and assistance that has helped in successful completion of the project.

Table of Contents

List of Figure

Abstract

BAIUST Result System gives a straightforward interface to the students for viewing their own results. It might be utilized by instructive industries to keep track of the results of all the students. It manages all kind of results of all students according to their semester. Student can view their result anytime by using given user id and password. Addition and update of any kind of result will be done by admin.

Chapter 1: Introduction

1.1 Introduction

The impact of computers and internet, on our lives today is probably much more than we really know. Getting information and quickly turning it into a product that consumers want is the essential key to staying in business and all of this is done nowadays using computers and applications or information systems.

Online result system is an advantage for any kind of institution. It can provide easy access to the result for everyone. It creates an environment with less error possibility and fewer manual work that will be much efficient than the traditional record system.

1.2 Motivation

BAIUST Result system is aimed to create an easy and automated way for the students to see their results.

This is a software that has the perspective of attaining attraction of those Universities which don't have good performing software for keeping their result secure and make their result easily accessible for the students

1.3 Contribution of the Project

In this project, we shall give our attention on the following specifications. The objectives of the project are mentioned as follows:

- To know the current status of every students
- To know the Result of individual students
- To know or update and add the result of the students

1.4 Organization of the Project

Literature review: It contains the description of fundamental concept of BAIUST Result System.

Methodology of our work: It contains the information of the students according to the university rules and described how the project is implemented in which method.

Implementation of our work: This chapter contains the details description of the implementation procedure of the project work. This chapter contains how to give input and what is its output.

Conclusion and recommendations: This chapter concludes my project paper and some recommendations are given here for the future improvement of this work.

1.5 Conclusion

In this chapter we have given a basic introduction of the project. In the next chapter we will describe the theoretical concept related with the application of development of BAIUST result system.

Chapter 2: Literature Review

Today in the university result of the students are published and handled manually which is not very efficient. Students have to be physically present to view their results. To keep track of the student records is a tedious work in case of updating them. There is a big chance for more manual errors.

2.1 Existing Work on the System

The system which is used nowadays has some drawbacks which need to be improved for better performance. The existing system which we use in our University is traditional process that includes a complete manual process. Now-a-days every institution is being dependent on the online system for most of the work process. It is much easier and accessible from anywhere. In our university all the results are being maintained in records. As a result for a increasing number of students it becomes very difficult to maintain this large number of data manually. Maintaining the records manually leads to error prone and required more man power and it consumes more time for processing the records.

2.2 Drawbacks of the Existing System

- The existing system less user friendly because the retrieval of data is very slow and data is not maintained efficiently. The use of some technology can be time consuming.
- It has a lot of manual work. All calculations to generate report is done manually so there is greater chance of errors.
- Here the students has to suffer a lot if there is a loss of some report then it may cause a lot of problem.
- This is time consuming also due to exaggerating calculation. Even after that there is some miscalculation which is very frustrating for the Students.
- The present system was very less secure.

Chapter 3: System Overview

3.1 Proposed System

The purpose is to design a web application which contains up to date information and Results of the students; which will be more efficient.

3.2 Benefits of the Proposed System

- Providing the online interface for students
- Increasing the efficiency of university result management.
- Decrease time required to access and deliver student results.
- To make the system more secure.
- Decrease time spent on non-value added tasks.
- To make the framework more secure.

3.2.1 System Architecture Diagram

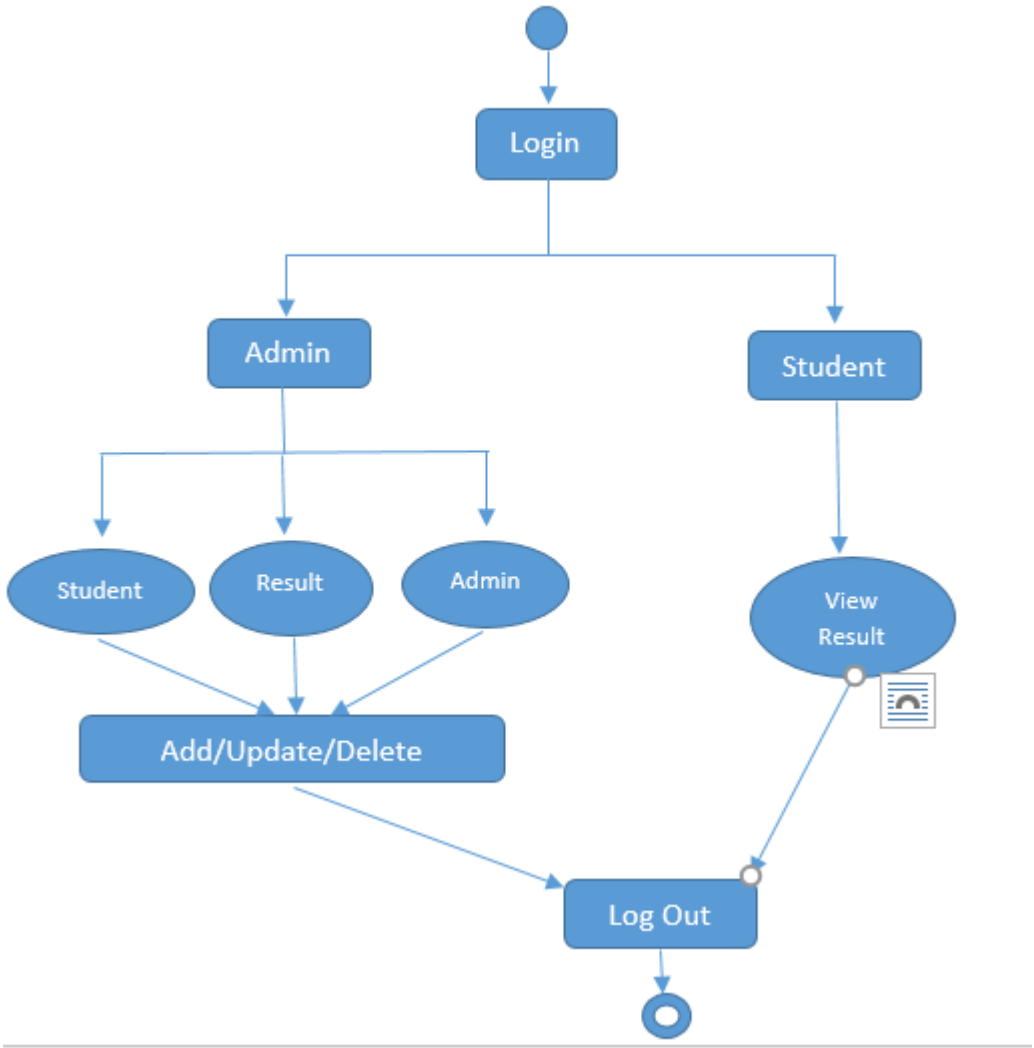


Fig 1: System Architecture Diagram of the Project

3.2.2 Modules

1. User Module: In this module we are authenticating the users by providing username and password. If username and password is valid then they will be taken to their screens.

2. Result Entry Module: The purpose of Result Entry Module is to enter Result. In this module Admin can add/update/delete result according to term & semester of every individual student.

3. Backlog Module: In this module students with backlogs are added to the list by the admin.

4. Student Module: This module allows the students to see their whole result by using user id and password. They can also change their password in this module.

5. Auto Calculation Module: Automatically calculate the total CGPA by calculating GPA.

3.2.3 Workflow

The detailed workflow of this application is as follows: The application is divided into 5 sub-groups:

1. **Admin:** Admin will maintain the main works of the system like authorization, addition or update students as well as their results. At first admin will add a student to the database. Then he can add result of the previously added student.
2. **Students:** An individual student can see his/her result by simply using his/her id and password.

3.3 XAMPP Server

XAMPP is a free and open-source cross-platform web server solution stack package developed by Apache Friends, consisting mainly of the Apache HTTP Server, MariaDB database, and interpreters for scripts written in the PHP and Perl programming languages.

3.3.1 Apache

The Apache server is set up to run through configuration files, in which directives are added to control its behavior. In its idle state, Apache listens to the IP addresses identified in its config file

(HTTPdconf). The browser then connects to a DNS server, which translates the domain names to their IP addresses.

3.3.2 MySQL

XAMPP only offers MySQL (Database Server) & Apache (Webserver) in one setup and you can manage them with the xampp starter. But at first you have to set/change the MySQL Root password. Start the Apache server and type localhost or 127.0.0.1 in your browser's address bar.

3.3.3 PHP

PHP is a popular general purpose scripting language that is especially suited to Web development. PHP code is usually processed on a web server by a PHP interpreter implemented as a module, a daemon or as a Common Gateway Interface (CGI) executable. On a web server, the result of the interpreted and executed PHP code – which may be any type of data,

Though it is very simple, it offers advanced features of a professional programmer. OOP is followed in PHP.

Chapter 4: Implementation Details

4.1 Methodology

4.1.1 Introduction

The goal of this project is to develop a web based system for the betterment of every student as well as the admins. This chapter will illustrate the whole frame work of the method.

4.1.2 2 Used Methodology

We have used Incremental Method on the proposed system.

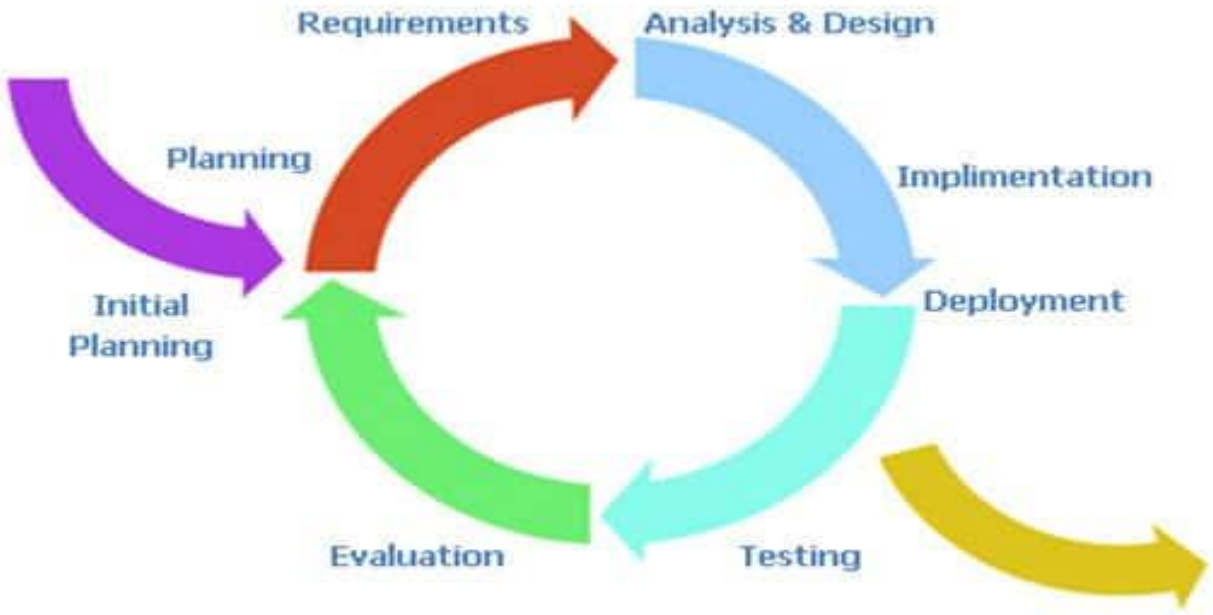


Fig 2: Incremental Model of the project

The incremental build model is a method of software development where the model is designed, implemented and tested incrementally (a little more is added each time) until the product is finished. It involves both development and maintenance. The product is defined as finished when it satisfies all of its requirements. This model combines the elements of the waterfall model with the iterative philosophy of prototyping.

4.1.3 Conclusion

Here, we have presented the whole methodology of the project. Implementation process and requirements will be discussed next.

4.2 Data Collections

After doing some research in the students we collected our data. Then the surveying data helped us to make an architectural design to implement to software. Moreover, the data collection phase is the first step to implement the software.

4.3 System Requirements

Front end:

- **HTML5:** HTML5 is a markup language used for structuring and presenting content on the World Wide Web. It is the fifth and latest major version of HTML that is a World Wide Web Consortium (W3C) recommendation.
- **CSS 3:** Cascading Style Sheets is a style sheet language used for describing the presentation of a document written in a markup language like HTML.
- **Bootstrap 4:** A free and open-source front-end web framework. It contains HTML and CSS-based design templates. We use many of the bootstrap classes, navigations, forms, buttons and interface components etc.

- **JavaScript:** JavaScript, often abbreviated as JS, is a high-level, interpreted programming language that conforms to the ECMAScript specification. It is a programming language that is characterized as dynamic, weakly typed, prototype-based and multi-paradigm.
- **JQuery:** A JavaScript library designed to simplify HTML DOM tree traversal and manipulation, as well as event handling, CSS animation, and Ajax. It is free, open-source software using the permissive MIT License. Web analysis indicates that it is the most widely deployed JavaScript library by a large margin

Database:

- **MySQL:** MySQL is an open-source relational database management system (RDBMS). 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.
  PDO (PHP Data Objects) is a general database abstraction layer with support for MySQL among many other databases. It provides prepared statements, and significant flexibility in how data is returned

Difference between PDO and MySQL:

	PDO	**MySQL**
Database support	12 different drivers	MySQL only
API	OOP	OOP+procedural
Connection	Easy	Easy
Named parameter	Yes	No
Object mapping	Yes	Yes
Prepared Statement	Yes	No
Performance	Fast	Fast
Stored procedure	Yes	Yes
Transaction support	Yes	Yes
Multiple statement support	Yes	Yes

Back end:

- **PHP:** PHP is a popular general purpose scripting language that is especially suited to Web development. PHP code is usually processed on a web server by a PHP interpreter implemented as a module,

Architecture:

- **MVC:** The Model-View-Controller (MVC) is an architectural pattern that separates an application into three main logical components: the model, the view, and the controller. Each of these components are built to handle specific development aspects of an application.

4.4 Implementation Details

4.4.1 Admin Part

The application will get a direct admin panel and access all the features which are provided.

4.4.1.1 Admin Login

Admin will login into the system with the valid email address and password, otherwise he/she can't access to the system. After successfully logged in the admin will get a dashboard like this:

4.4.1.2 Administration

- **Students:**
    - Admin can add a new student and delete it

- Can access all the student’s information.

- **Result:**

  - Admin can add/update/delete result of the students
  - Can access all the student’s result.

- **Admin:**

  - Admin can add a new admin and delete it
  - Can access all the admin’s information.

4.4.2 Student Part

4.4.2.1 Student Login

Student can login with their ID and password in order to view their result according to their department and level/term.

4.5 System Design

4.5.1 Functional Requirements

BAIUST Result system aims to improve the efficiency of information management, and the main function is managing and maintaining information. The administrator and students are two major functional requirements in the system. The Administrator will be given more powers

(enable/disable/ update) than other users. It will be ensured that the information entered is of the correct format. For example: name cannot contain numbers. In case if incorrect form of information is added, the user will be asked to fill the information again. Students use the system to query and enter their information only.

4.5.2 Non-Functional Requirements

- **Performance Requirements:** The proposed system that we are going to develop will be used as the chief performance system for helping the organization in managing the whole database of the student studying in the organization. Therefore, it is expected that the database would perform functionally all the requirements that are specified.
- **Safety Requirements:** The database may get crashed at any certain time due to virus or operating system failure. Therefore, it is required to take the database backup.
- **Security Requirements:** We are going to develop a secured database. There are various categories of people namely Administrator, Student who will be viewing either all or some specific information from the database. Depending upon the category of user the access rights are decided. It means if the user is an administrator then he can be able to modify the data, append etc. All other users only have the rights to retrieve the information from the database.

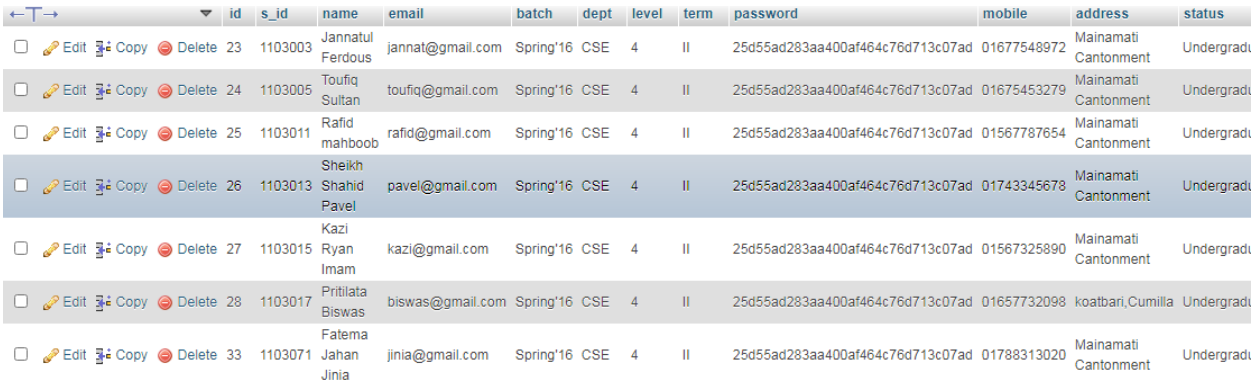
4.5.3 Database Design

It is fair to say that database play a critical role in almost all areas where computers are used, including business, electronic commerce, engineering, medicine, law, education, and library science. A database is collection of a related data. A database has the following implicit properties:

- A database represents some aspect of the real world, sometimes called the mini-world or the Universe of Discourse (UOD) changes to the mini world are reflected in the database.
- A database is a logically coherent collection of data with some inherent meaning. A random assortment of data cannot correctly be referred to as a database.

- A database is designed, built, and populated with data for a specific purpose. It is an intended group of users and some preconceived application which these users are interested.

Database Management System (DBMS) is a collection of programs that enables users to create and maintain a database. DBMS is a general –purpose software system that facilitates the process of defining, constructing, manipulating, and sharing database among various users and applications. Defining a database involves the specifying the data types, structures, and constraints of the data to be stored in the database. The database definition or descriptive information is also stored in the database in the form of dictionary; it is called Meta data constructing the database is the process of storing the data on the storage medium that is controlled by the DBMS. Manipulating a database includes functions such as querying the database to retrieve specific data, updating the database to reflect in the mini-world, and generating reports from the data. Sharing a database allows a multiple users and programs to access the database simultaneously. Application program accesses the database by sending queries or request for data to the DBMS. A query typically causes some data to be retrieved; a transaction may cause some data to be read and some data to be written into the database.



←T→	id	s_id	name	email	batch	dept	level	term	password	mobile	address	status
Edit Copy Delete	23	1103003	Jannatul Ferdous	jannat@gmail.com	Spring'16	CSE	4	II	25d55ad283aa400af464c76d713c07ad	01677548972	Mainamati Cantonment	Undergradu
Edit Copy Delete	24	1103005	Toufiq Sultan	toufiq@gmail.com	Spring'16	CSE	4	II	25d55ad283aa400af464c76d713c07ad	01675453279	Mainamati Cantonment	Undergradu
Edit Copy Delete	25	1103011	Rafid mahboob	rafid@gmail.com	Spring'16	CSE	4	II	25d55ad283aa400af464c76d713c07ad	01567787654	Mainamati Cantonment	Undergradu
Edit Copy Delete	26	1103013	Sheikh Shahid Pavel	pavel@gmail.com	Spring'16	CSE	4	II	25d55ad283aa400af464c76d713c07ad	01743345678	Mainamati Cantonment	Undergradu
Edit Copy Delete	27	1103015	Kazi Ryan Imam	kazi@gmail.com	Spring'16	CSE	4	II	25d55ad283aa400af464c76d713c07ad	01567325890	Mainamati Cantonment	Undergradu
Edit Copy Delete	28	1103017	Pritilata Biswas	biswas@gmail.com	Spring'16	CSE	4	II	25d55ad283aa400af464c76d713c07ad	01657732098	koatbari,Cumilla	Undergradu
Edit Copy Delete	33	1103071	Fatema Jahan Jinia	jinia@gmail.com	Spring'16	CSE	4	II	25d55ad283aa400af464c76d713c07ad	01788313020	Mainamati Cantonment	Undergradu

Fig 3: Student DB

Edit Copy Delete	73	1103005	CSE-411	VLSI DESIGN		II	.75	3.5
Edit Copy Delete	78	1103011	CSE-411	VLSI DESIGN	4	II	3.00	4.00
Edit Copy Delete	79	1103011	CSE-412	VLSI DESIGN SESSIONAL	4	II	.75	3.75
Edit Copy Delete	80	1103011	CSE-413	COMPUTER GRAPHICS	4	II	3.00	4.00
Edit Copy Delete	81	1103011	CSE-414	COMPUTER GRAPHICS SESSIONAL	4	II	.75	3.05
Edit Copy Delete	82	1103011	HUM-415	FINANCIAL & MANAGERIAL ACCOUNTING	4	II	2.00	4.00
Edit Copy Delete	83	1103011	CSE-417	ENGINEERING ETHICS	4	II	3.00	3.00
Edit Copy Delete	84	1103011	CSE-455	DATA MINING	4	II	3.00	4.00
Edit Copy Delete	85	1103011	CSE-456	DATA MINING SESSIONAL	4	II	1.5	3.75
Edit Copy Delete	86	1103013	CSE-411	VLSI DESIGN	4	II	3.00	3.75
Edit Copy Delete	87	1103013	CSE-412	VLSI DESIGN SESSIONAL	4	II	.75	3.75
Edit Copy Delete	88	1103013	CSE-413	COMPUTER GRAPHICS	4	II	3.00	3.00
Edit Copy Delete	89	1103013	CSE-414	COMPUTER GRAPHICS SESSIONAL	4	II	.75	3.05
Edit Copy Delete	90	1103013	HUM-415	FINANCIAL & MANAGERIAL ACCOUNTING	4	II	2.00	4.00
Edit Copy Delete	91	1103013	CSE-417	ENGINEERING ETHICS	4	II	3.00	3.05
Edit Copy Delete	92	1103013	CSE-455	DATA MINING	4	II	3.00	3.5
Edit Copy Delete	93	1103013	CSE-456	DATA MINING SESSIONAL	4	II	1.5	3.75
Console t Copy Delete	94	1103015	CSE-411	VLSI DESIGN	4	II	3.00	3.75

Fig 4: Result DB

Options

	id	s_id	c_code
Edit Copy Delete	1	1103011	CSE-411
Edit Copy Delete	2	1103071	CSE-403
Edit Copy Delete	3	1103071	CSE-315
Edit Copy Delete	4	1103071	CSE-317
Edit Copy Delete	5	1103071	CSE-319
Edit Copy Delete	6	1103071	CSE-305

Fig 5: Backlog DB

4.5.4 Use Case Diagram

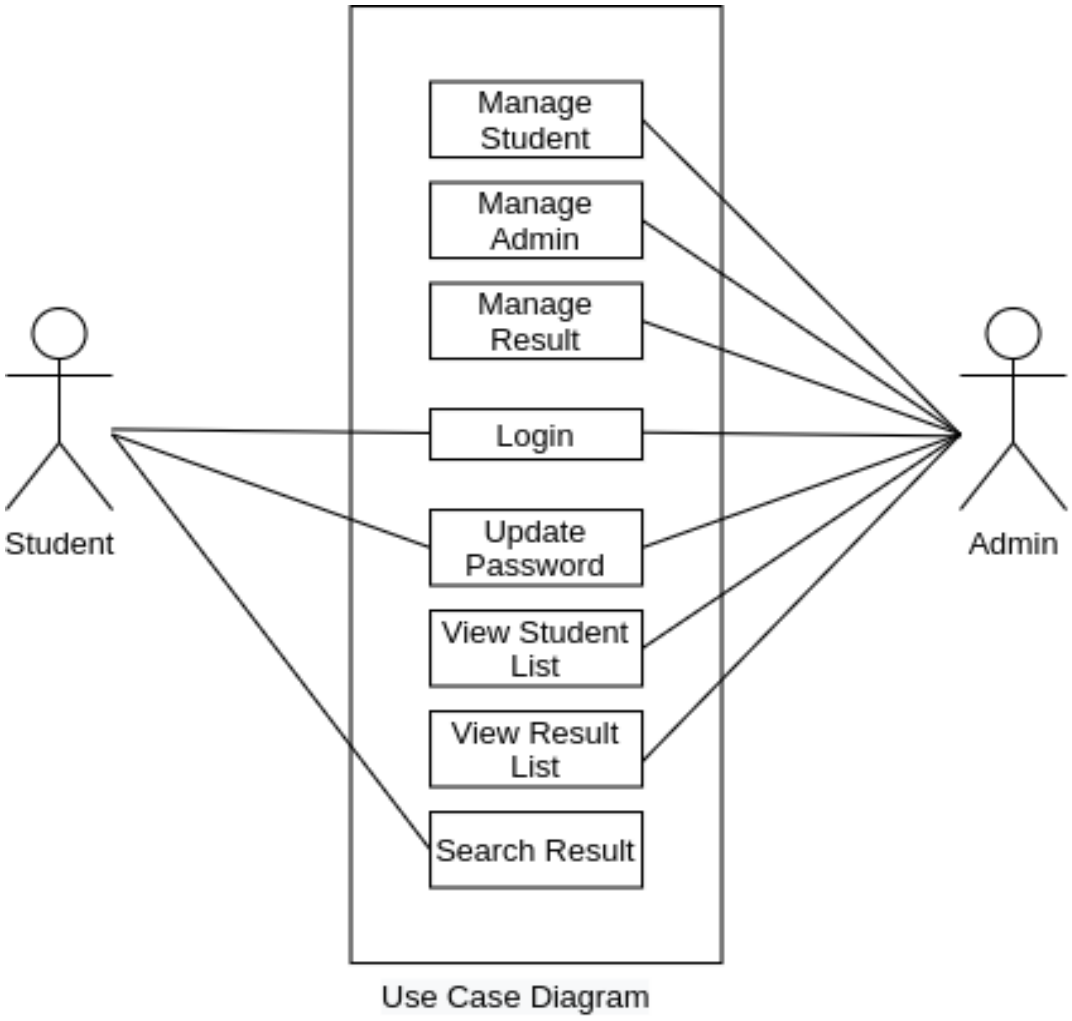


Fig 6: Use Case Diagram of the Project

4.5.5 Entity Relationship Diagram

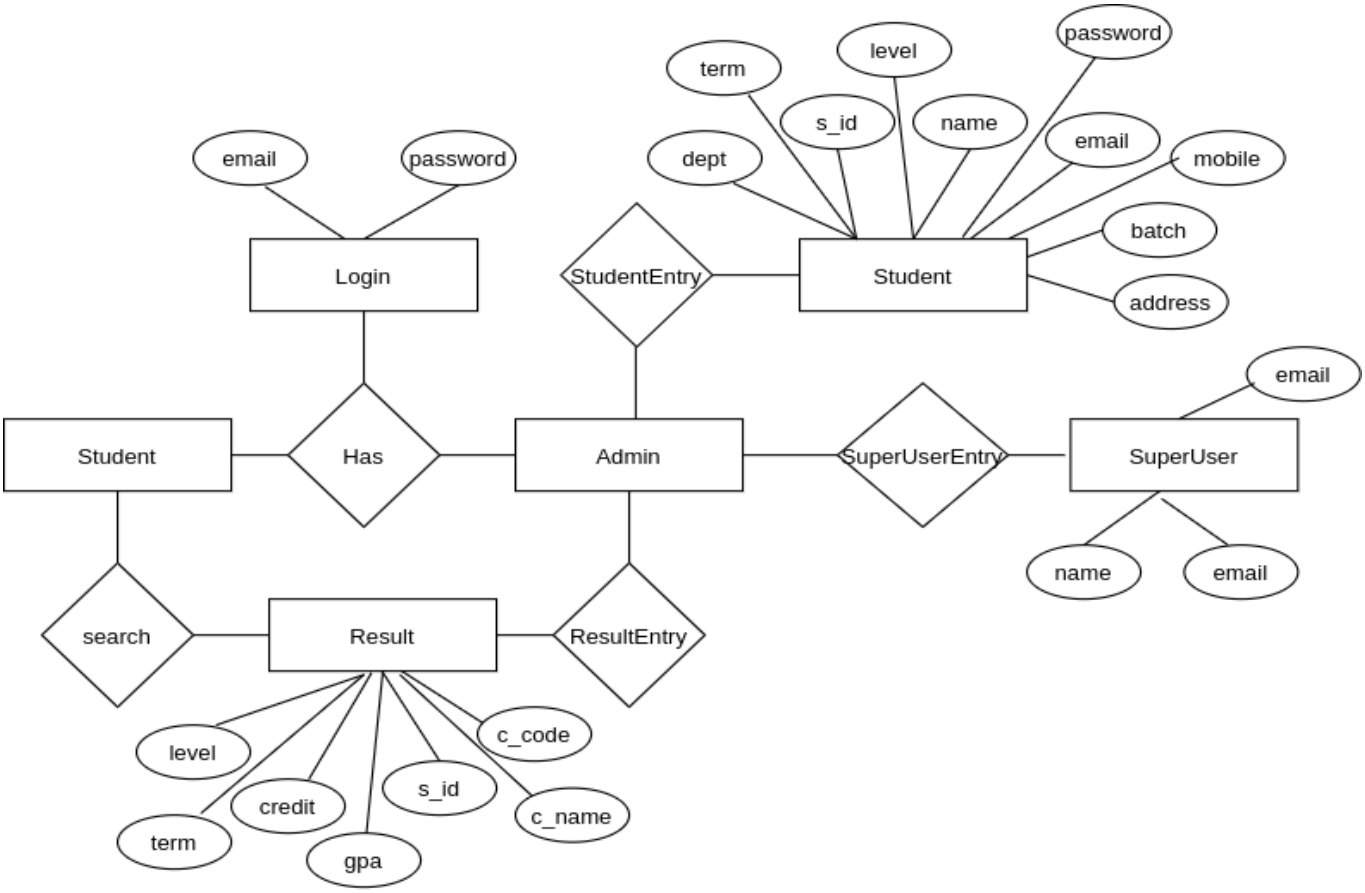


Fig 7: ER Diagram of the Project

4.5.7 Context Diagram

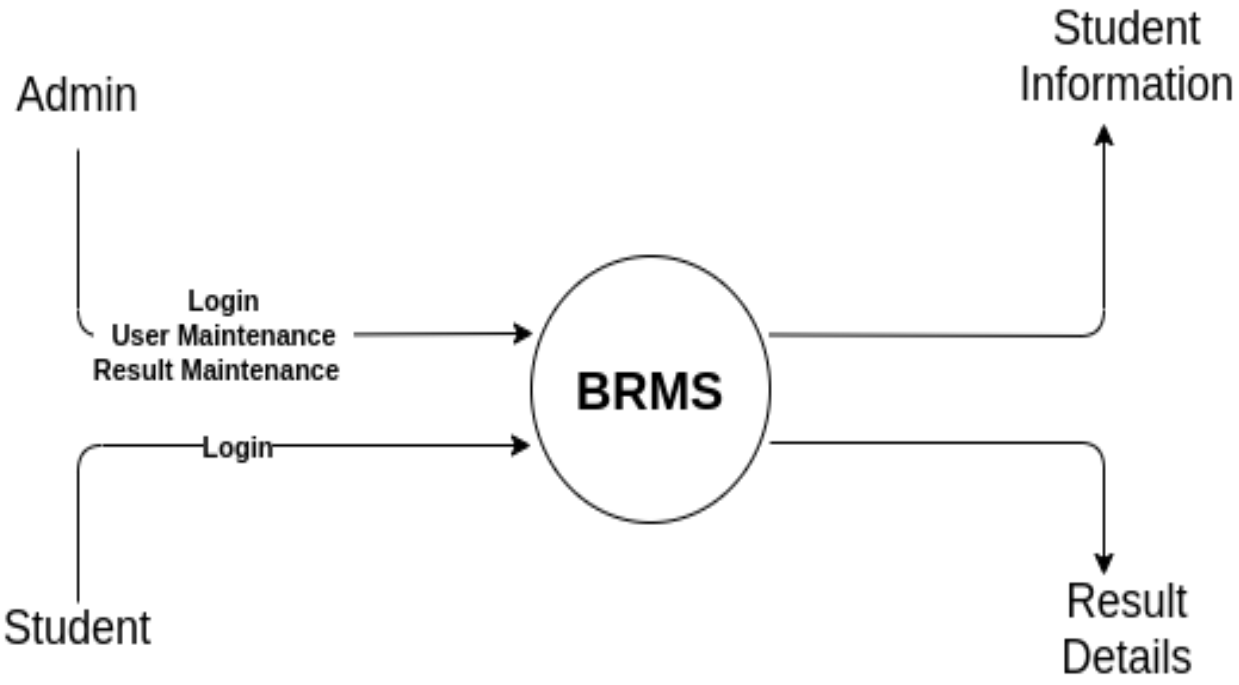
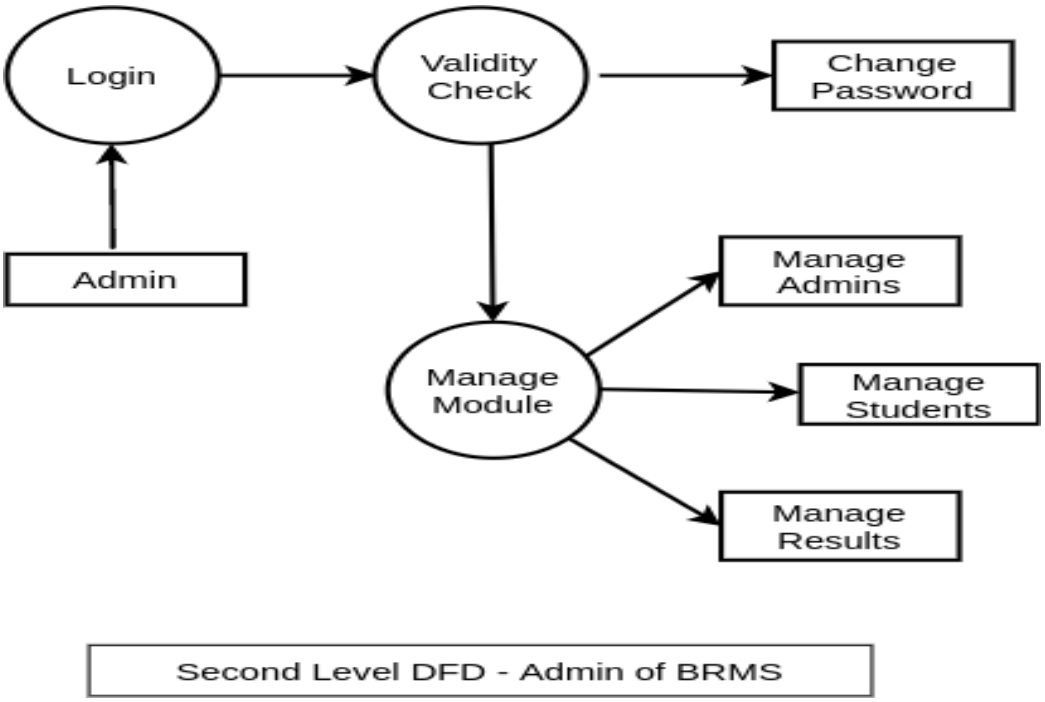
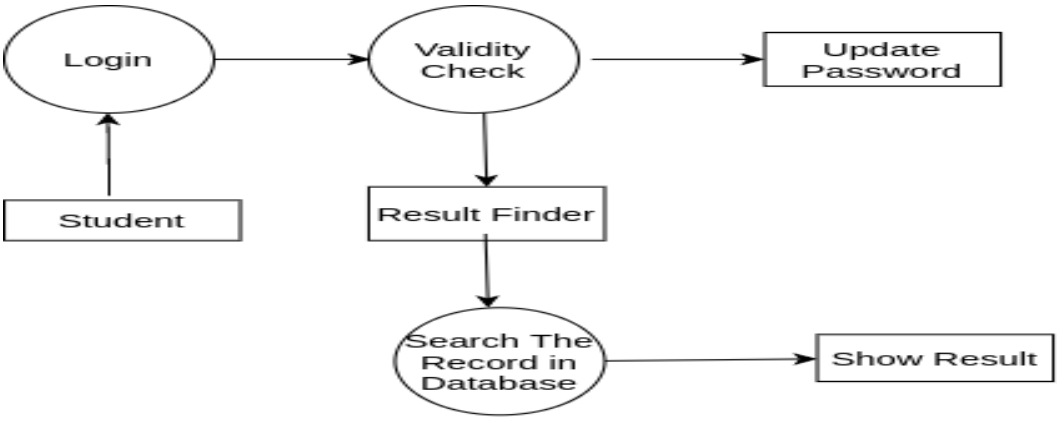


Fig 8: Context diagram

4.5.8 Data Flow Diagram



Second Level DFD - Admin of BRMS



Second Level DFD - BRMS(Student)

Fig 9: Data flow diagram

Chapter 5: Experimental Results & Discussion

5.1 Introduction

In this chapter, we will describe the experimental results of the project. Here, we have also evaluated the extent that this project is successful in achieving the objectives that are defined at the beginning of this project. In this chapter, we will show some real life experimental results from our proposed system. Here, to verify the system we have talked to several students, faculties and fulfill their requirements through our system. Later in this chapter, we will evaluate the system.

5.2 Experimental Data

We collected the information of which functionalities or which features are needed most at the Result system. In this university an online Result system is needed most. We talked with many students about our system and 90% students was satisfied with the idea.

We needed to collect the data of each and every student of the university manually as we need to input them into database.

5.3 Evaluation

Evaluation refers to a periodic process of gathering data and then analyzing or ordering it in such a way that the resulting information can be used to determine whether a organization or program is effectively carrying out planned activities, and the extent to which it is achieving it's stated objectives and anticipated results. So, it is important to evaluate our system for its further development.

5.3.1 Evaluation Method

There are two different distinct ways to measure the effectiveness. Subjective and objective. Subjective performance can't be measured or quantified while objective evaluation can only be evaluated by measurable qualities.

While quality of work can be easily measured, how well a certain student get benefited through the system is the main concern of this project.

5.4 Results and Analysis

In this section, we will show the entire output of the project as screenshots are attached in the document below:

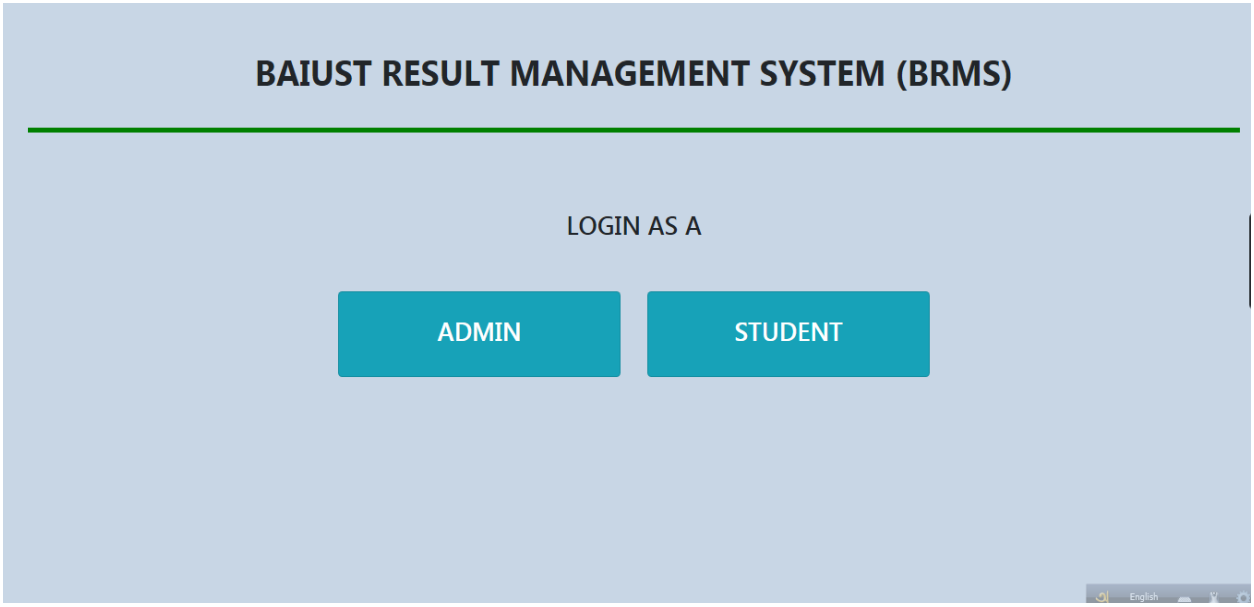


Fig 10: Home page

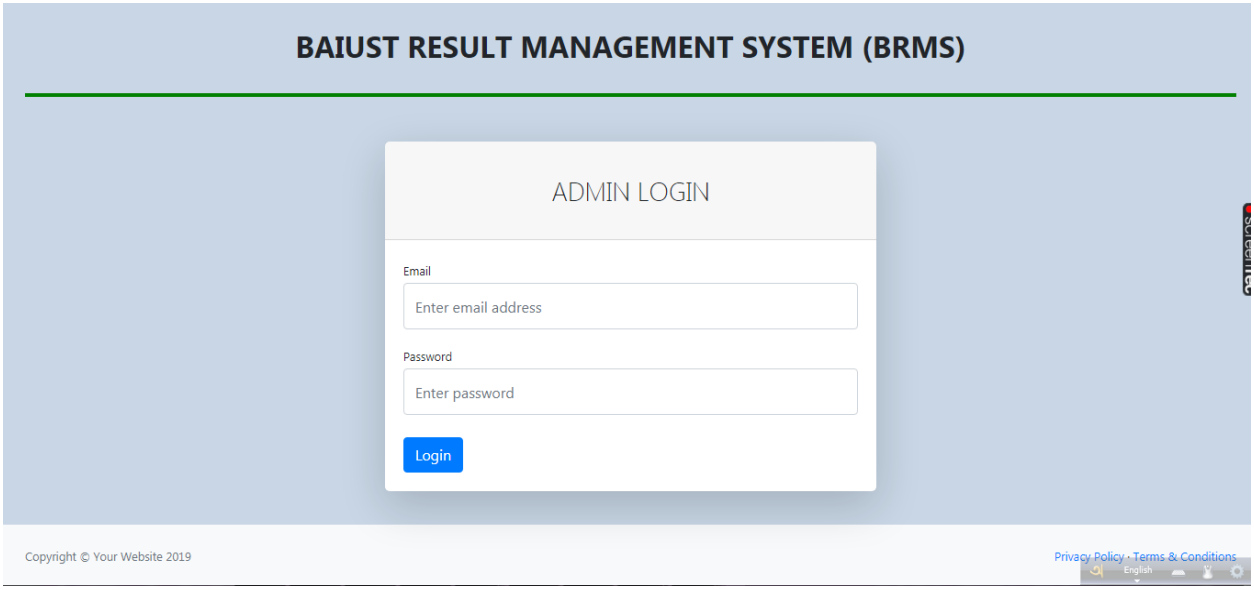


Fig 11: Admin login

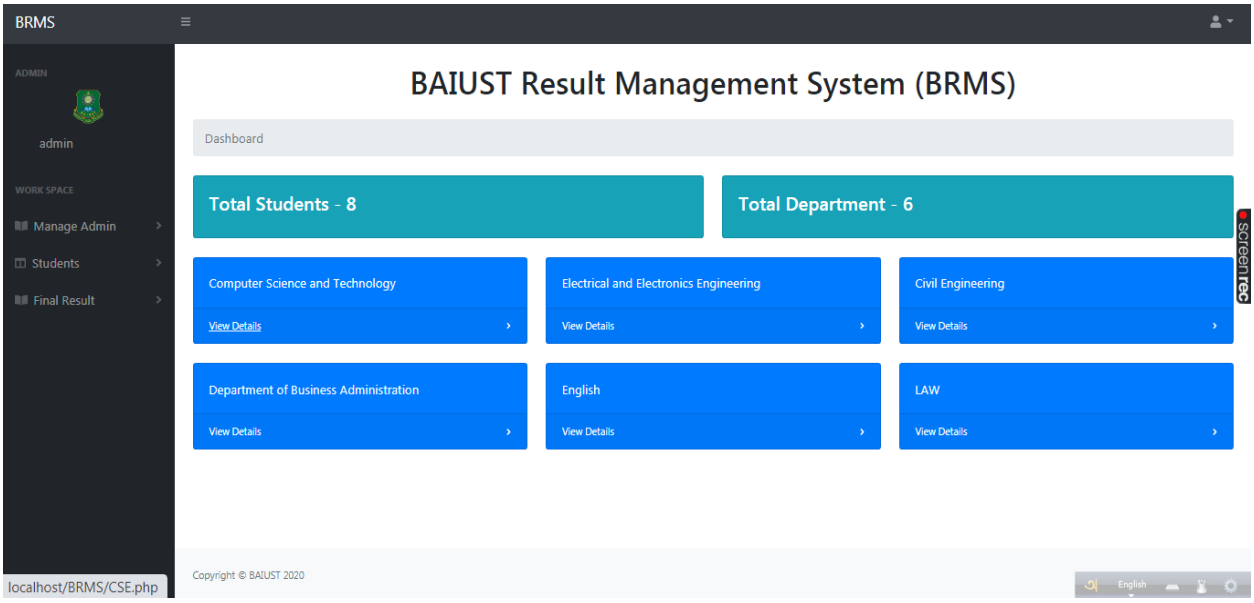


Fig 12: Admin Dashboard

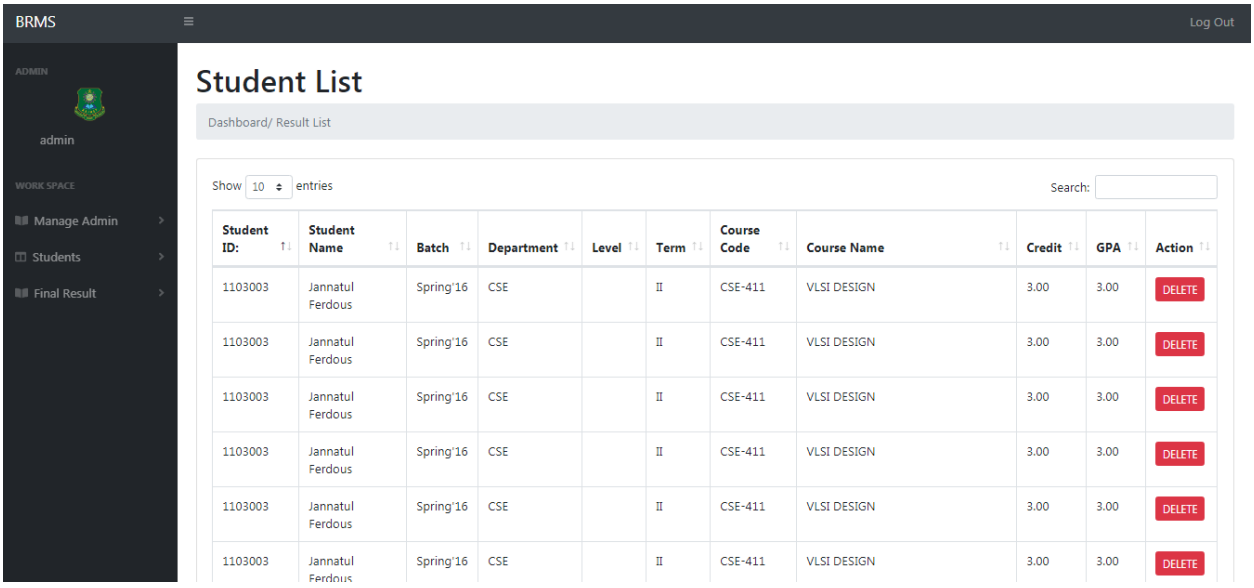


Fig 13: Department wise Student result List (Admin view)

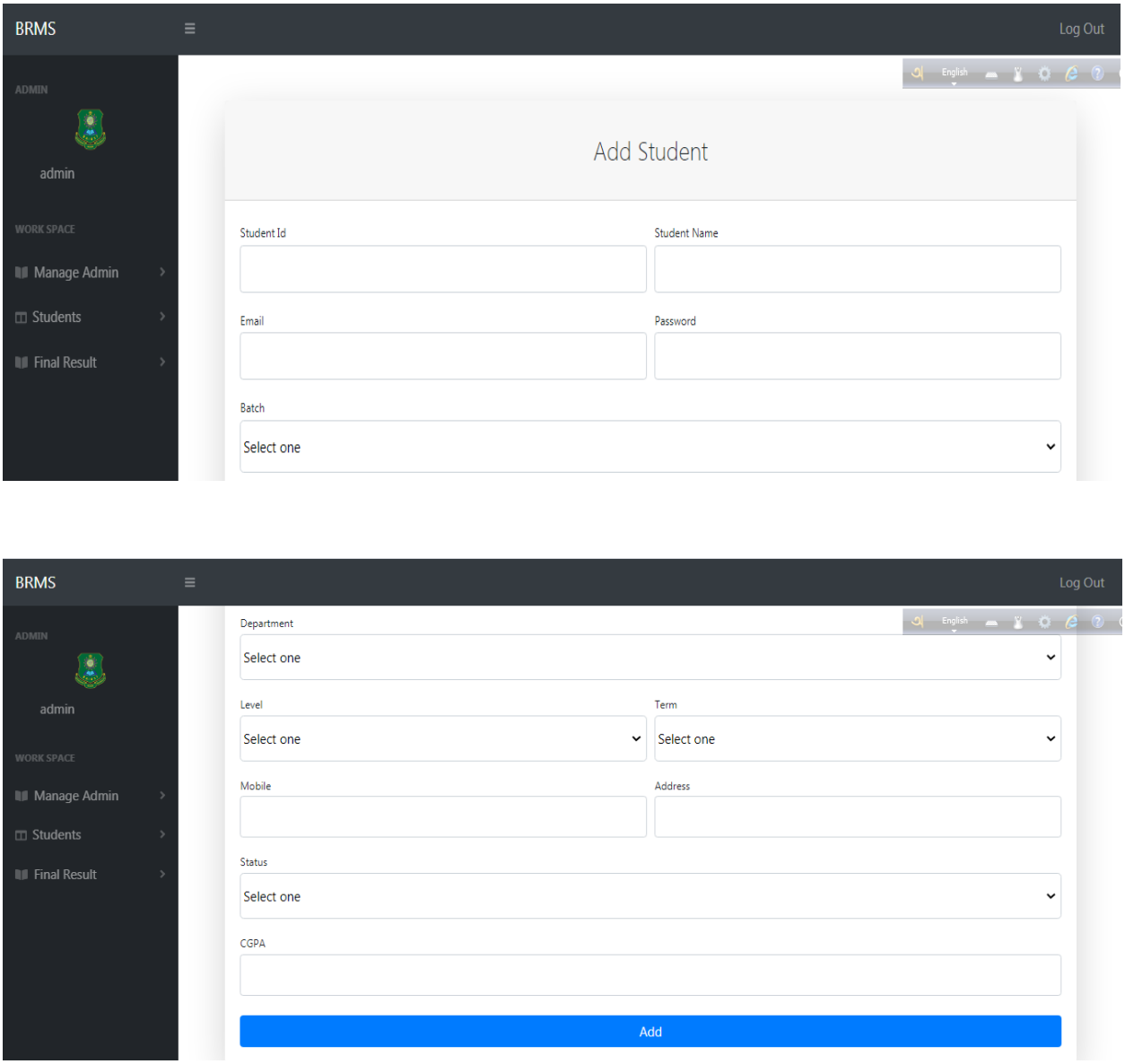


Fig 14: Add Student

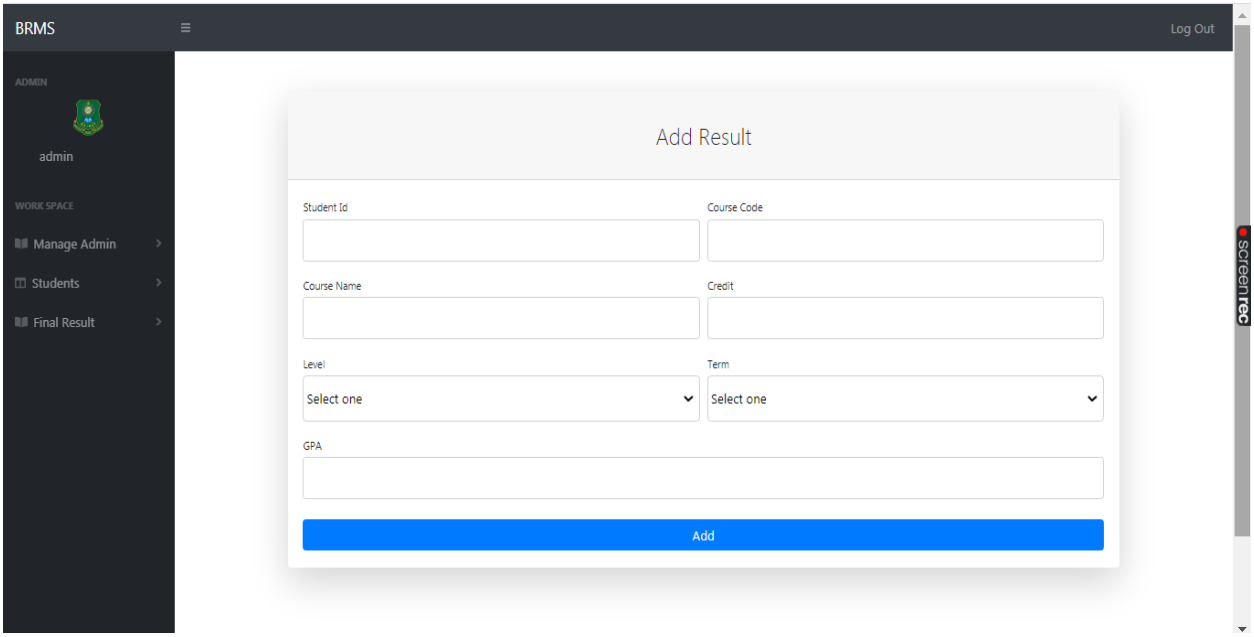


Fig 15: Add Result

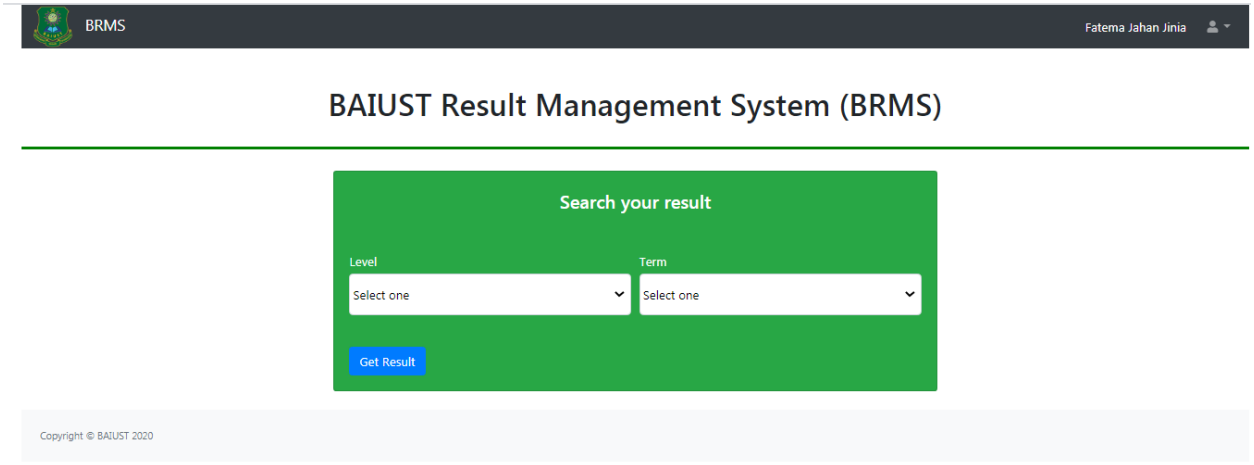


Fig 16: Search Result

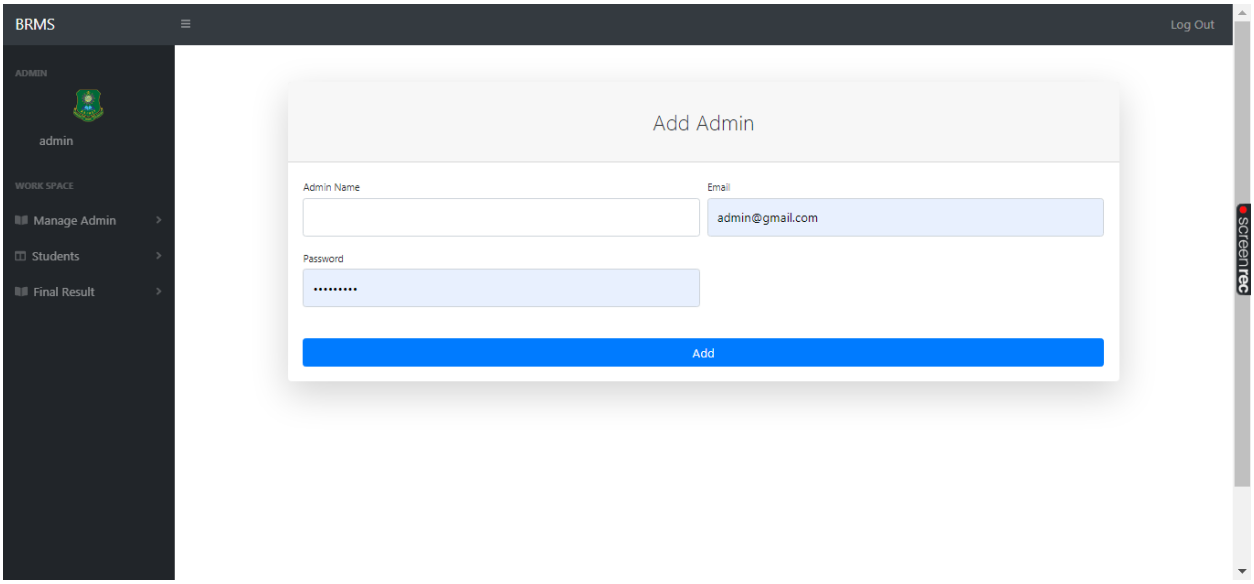


Fig 17: Add Admin

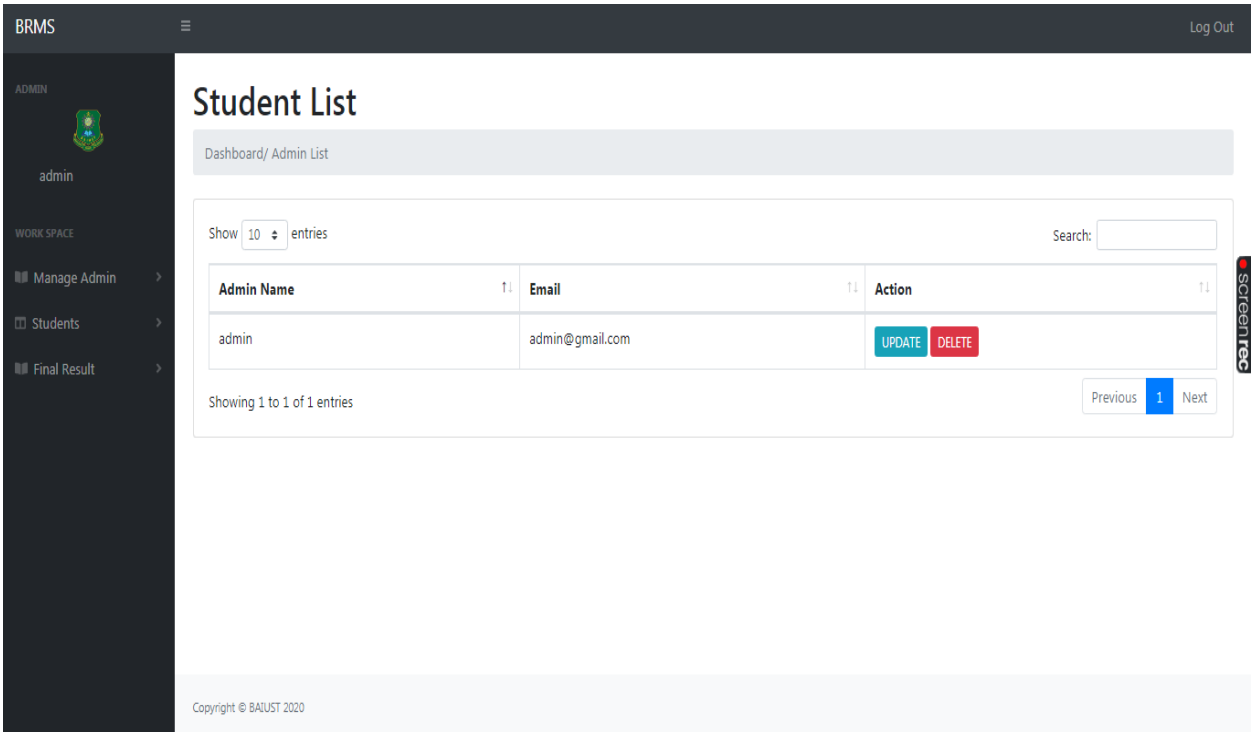


Fig 18: Admin List

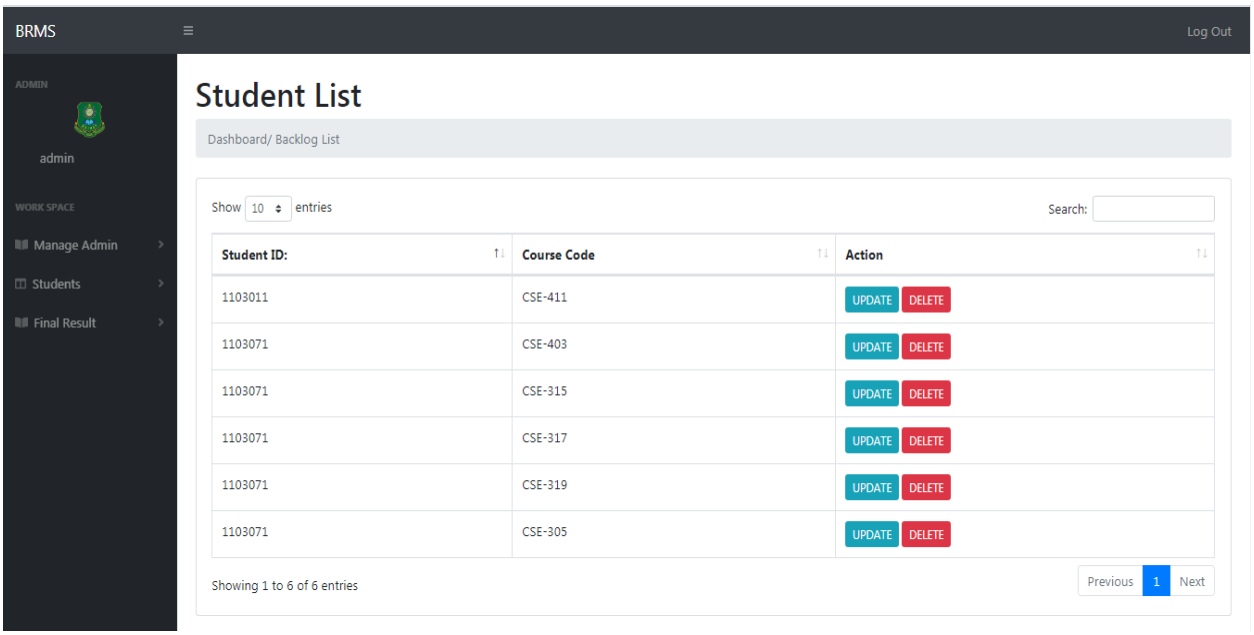


Fig 19: CSE student list (Admin View)

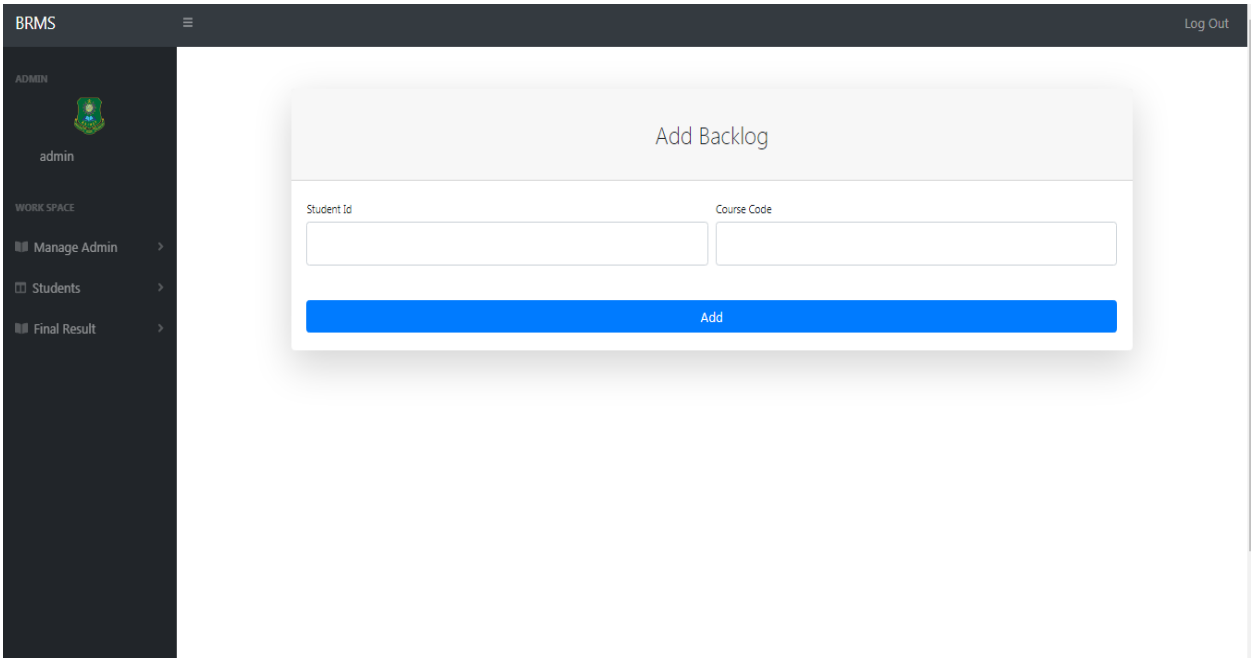


Fig 20: Add Backlog

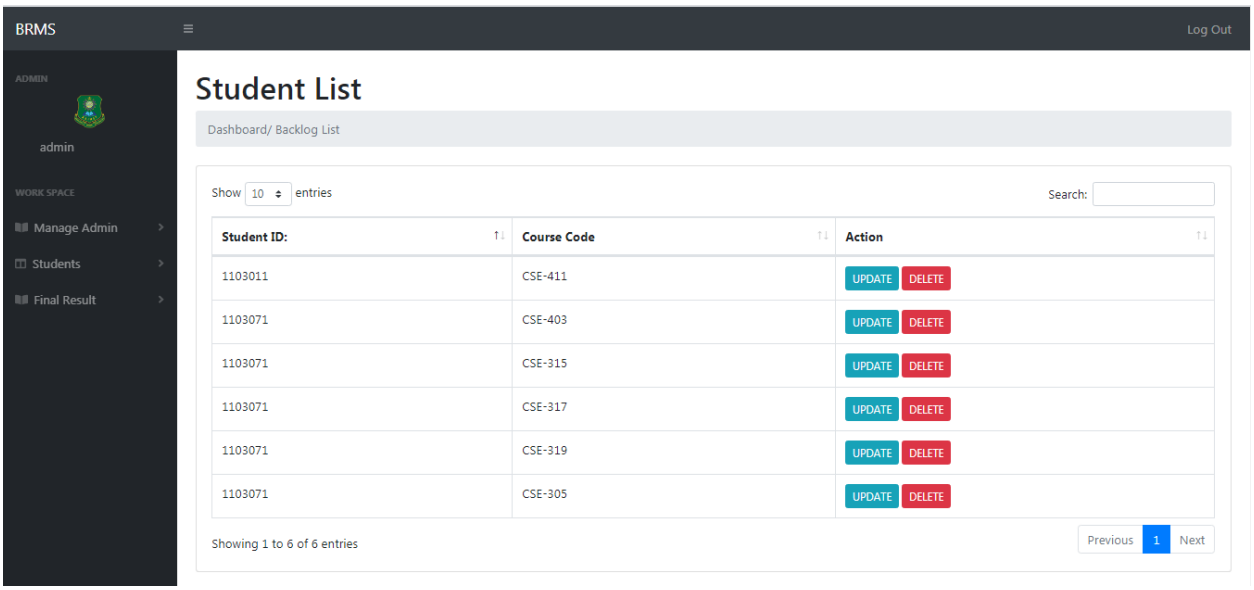


Fig 21: Backlog List

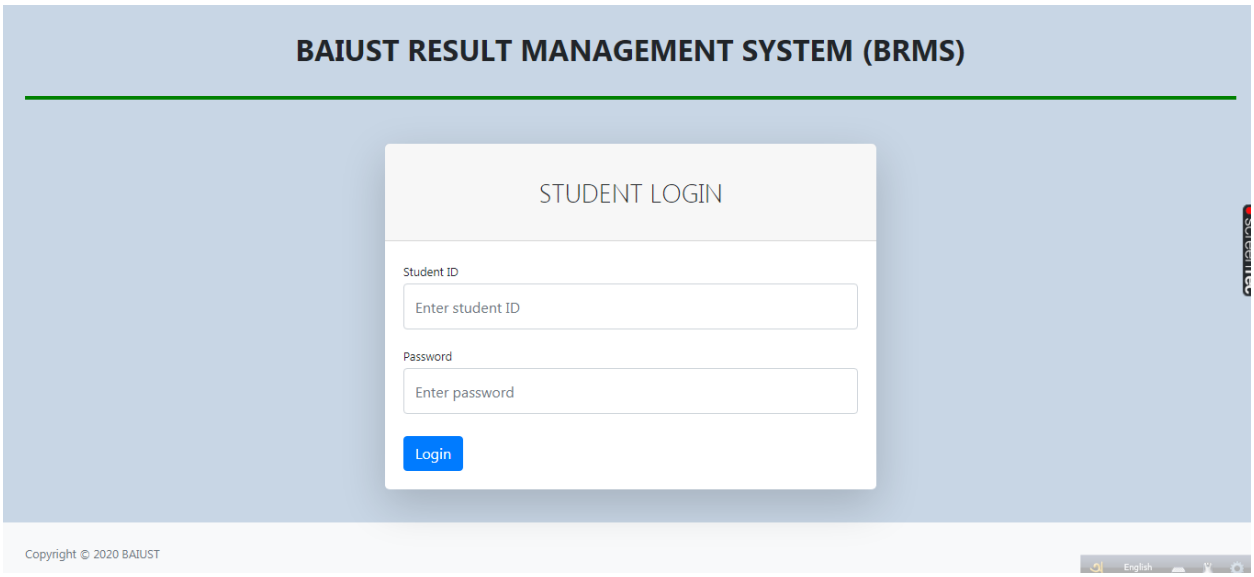


Fig 22: Student Login

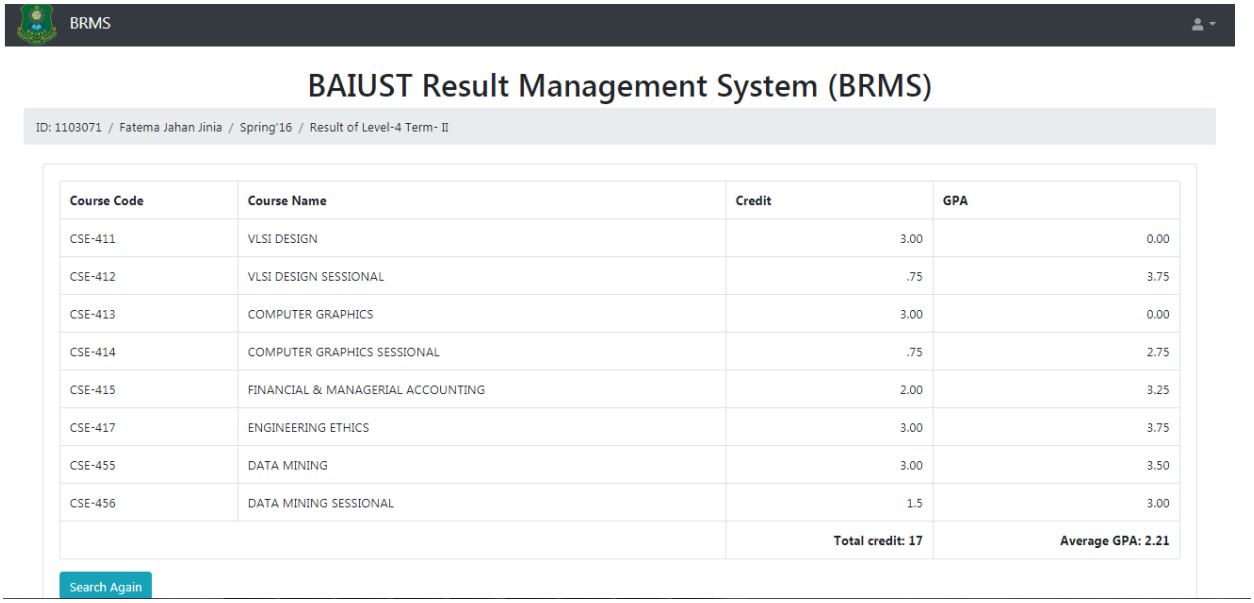


Fig 23: Student Result View

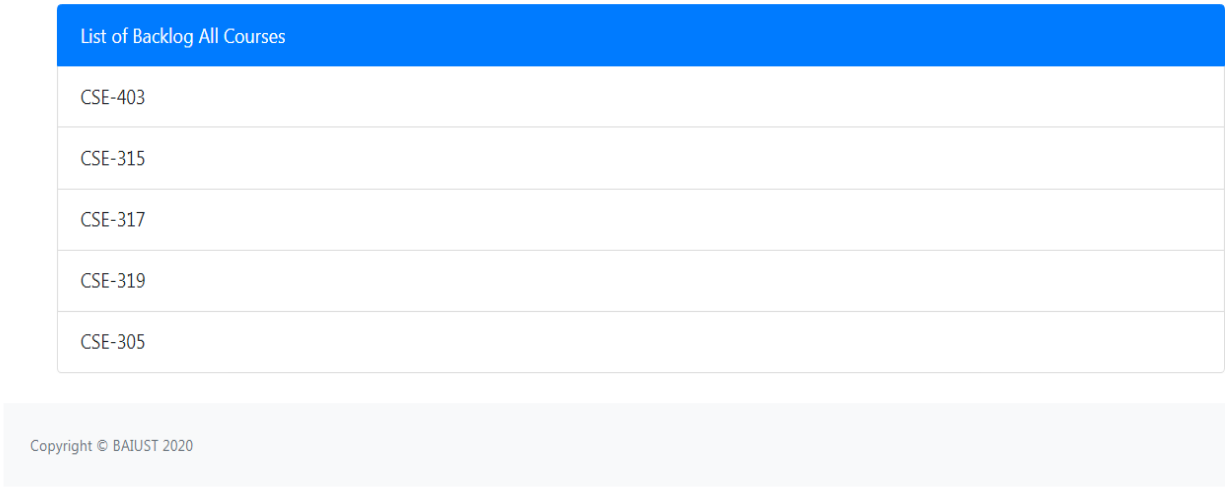


Fig 24: Backlog List (Student view)

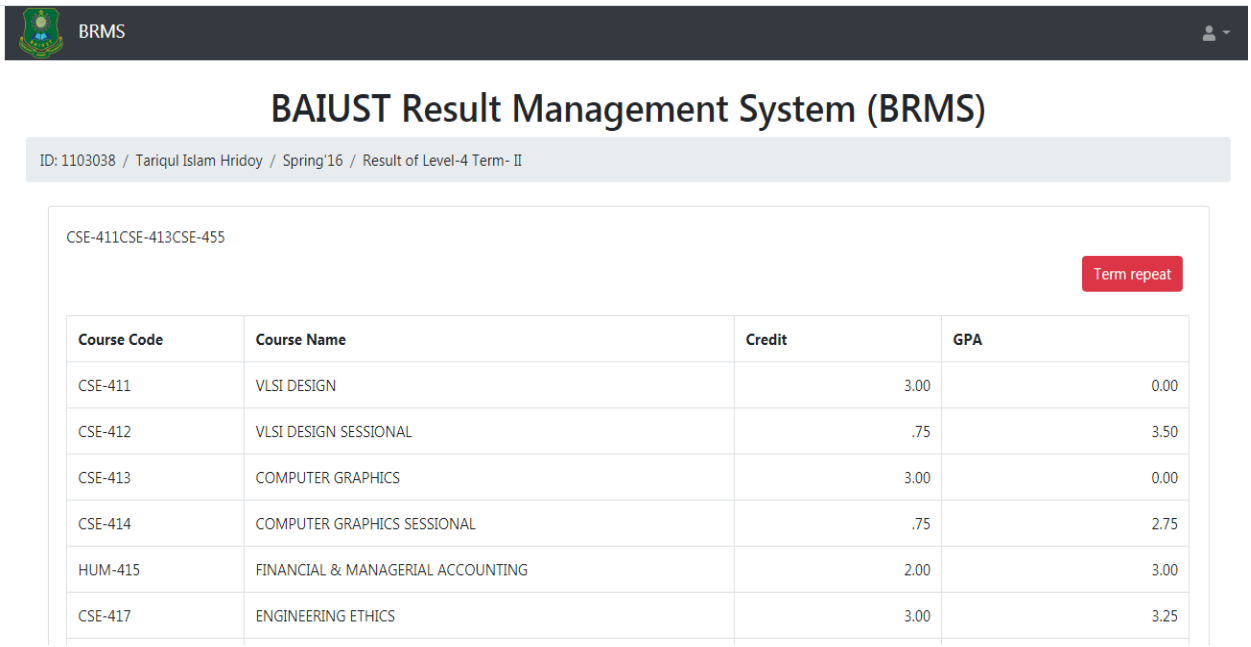


Fig 25: Student Result View with Term repeat

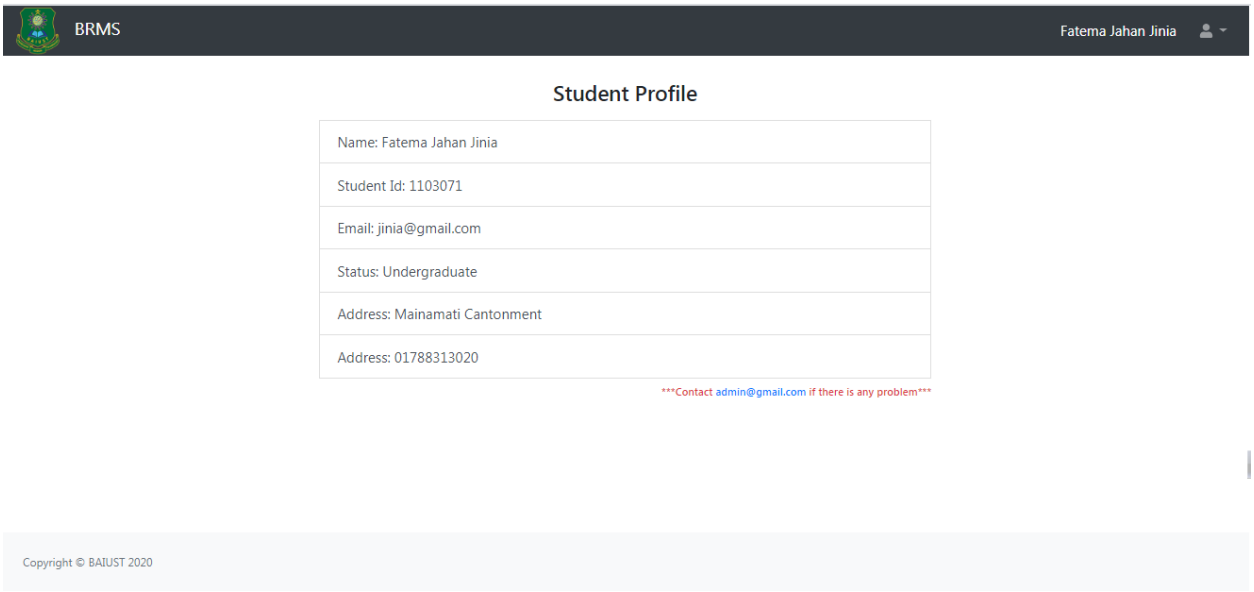


Fig 26: Student profile

Chapter 6: Conclusion and Future Recommendation

In this chapter we will overview the whole project in a nut shell. In our system there are some limitations which we will try to overcome in the near future. So, we also try to explain our work in this section.

6.1 Conclusion

Proposed application is a development for the students of the university. Our application provides all results to the students. It keeps track of the result from the very beginning to the end of the university life

6.2 Future Recommendations

The proposed framework is a development of BAIUST Result system. There are some ways to improve our application. The following issues may be taken into consideration to make the application more reliable and user friendly in the future.

- We can create a more nice and powerful dashboard if we can use Laravel Framework
- We wish to use rest API in our system.
- The system is currently works only in windows. In near future, we will try to make it an android version.

References:

1. A Comprehensive Guide for Beginners interested in learning JavaScript, Bootstrap & PHP https://www.udemy.com/course/javascript-bootstrap-php-certification-for-beginners/
2. A platform to help people learn code for free https://www.freecodecamp.org/learn
3. Bootstrap Documentation: https://getbootstrap.com/
4. The World's largest Web development site https://www.w3schools.com/
5. https://www.tutorialspoint.com/
6. https://stackoverflow.com/
7. https://developer.mozilla.org/en-US/docs/Web/JavaScript/
8. https://www.academia.edu/37780084/_Examination_Management_System_Project_Stage_1_Examination_Management_System
9. https://www.baiust.edu.bd/

Appendix:

1. Preventing unauthorized entry:

```
session_start();
if(empty($_SESSION['user_id']) && $_SESSION['email']==''){
        header("Location:../login.php");
        $_SESSION['warning'] = "<div class='alert alert-danger'>Please Login First!!</div>";
        exit();
}
```

2. Login verification (controller):

```
case "student_login_process":
        $getUser = $admin->getUserByEmail($_POST['email']);
        if(count($getUser) > 0 && $getUser['email']!='')
        {
                if($getUser['password']==md5($_POST['password']))
                {
                $_SESSION['user_id'] = $getUser['id'];
                $_SESSION['email'] = $getUser['email'];
                $_SESSION['name'] = $getUser['name'];
                header("Location:../index.php");
                }
                else{
                $_SESSION['message'] = "<div class='alert alert-danger'>Invalid Password!!</div>";
                header("Location:../login.php");
                }
        }
        else{
                $_SESSION['message'] = "<div class='alert alert-danger'>Invalid Email!</div>";
                header("Location:../login.php");
```

```
        }
break;
```

3. Logout (controller):

```
case "logout_process":
        session_destroy();
        header("Location:../index.html");
break;
```

4.Database Connection:

```
class Dbconnection{
        protected $db;
        public function __construct(){
                try{
                $this->db = new PDO('mysql:host=localhost;dbname=BRMS','root',");
                }
                catch(PDOExection $e){
                echo "Error Found ..!";
                }
        }
}
```

5. Fetching individual result (model):

```
public function all(){
        $sql = "SELECT final_result.id, student.s_id, final_result.gpa, student.name, student.dept
        ,final_result.level, final_result.term, final_result.c_code, final_result.c_name, final_result.credit,
        student.batch FROM student INNER JOIN final_result ON student.s_id=final_result.s_id";



        $query = $this->db->query($sql);
        $data = $query->fetchAll(PDO::FETCH_ASSOC);
        return $data ? $data : [];
}
```

6.GPA calculation (view):

```
<?php if(!empty($status)){ ?>
        <table class="table table-bordered" id="dataTable" width="100%" cellspacing="0">
                <thead>
                        <tr>
                                <th>Course Code</th>
                                <th>Course Name</th>
                                <th>Credit</th>
                                <th>GPA</th>
                        </tr>
                </thead>
                <tbody>
                        <?php foreach($status as $data) { ?>
                        <tr>
                                <td><?php echo $data['c_code'] ?></td>
                                <td><?php echo $data['c_name'] ?></td>
```

```
                    <td class="text-right">
                        <?php
                            echo $data['credit'];
                            $credit[]=$data['credit'];
                            $res[]= $data['credit']*$data['gpa']."<br>";
                        ?>
                    </td>
                    <td class="text-right"><?php echo $data['gpa'] ?></td>
                </tr>
            </tbody>
            <?php } ?>
            <tfoot>
                <tr>
                <th colspan="2"></th>
                    //-----------Calculating total credit----------
                    <th class="text-right"> Total credit: <?php echo array_sum($credit); ?></th>
                    //----------Calculating GPA----------
                    <th class="text-right">GPA: <?php echo round((array_sum($res)/array_sum($credit)),2); ?></th>
                </tr>
            </tfoot>
        </table>
<?php
}
else{
    echo '<h3 class="text-danger">No result Found</h3>';
}?>
```