

Centralized School Management System Using Distributed Database



Inspiring Excellence

Sabid Samayel (13101062)

Ashikul Alam Chowdhury (12201084)

Md. Shahidul Alam (13101038)

Department of Computer Science and Engineering

Declaration

We, hereby declare that this thesis is based on results we have found ourselves. Materials of work from researches conducted by others are mentioned in the reference. This thesis, neither in whole nor in part, has been previously submitted for any other degree or any other publication. All the implementation has been and functionalities been used are done by ourselves.

Signature of Authors

Signature of Supervisor

Sabid Samayel (13101062)

Dr. Amitabha Chakrabarty

Assistant Professor

BRAC University

Dhaka, Bangladesh

Ashikul Alam Chowdhury (12201084)

Md. Shahidul Alam (13101038)

Acknowledgement

This idea was suggested to us by Dr. Amitabha Chakrabarty (Assistant Professor), Department of Computer Science and engineering, BRAC University, Dhaka as our graduation thesis. We shared our idea with him and our desire to work with him on this topic. This project thesis is the work of Sabid Samayel, Ashikul Alam Chowdhury and MD. Shahidul Alam, student of Computer Science and Engineering Department, BRAC University.

We would like to express our gratitude to Almighty Allah who gave us the opportunity, determination, courage, patience in hard times and strength to complete our work.

A very big thank you to Dr. Amitabha Chakrabarty who was kind enough to let us do work with him and shared his idea how to bring more completeness in the work and constantly reviewing our work. He was always there for us to guide us. We are extremely fortunate to have a supervisor like him.

Our family members have been a great support throughout our work. We would like to also thank them for having faith in us, without their support we could not have achieved this success.

Lastly, we would like to thank all our friends and classmates for supporting us and helping us by providing moral support and thank all other faculty members of Computer Science and Engineering Department from whom we gained our knowledge and helped throughout the thesis work.

Abstract

Focusing on the digitalization of different sectors by Bangladesh government, it came in accordance that education system of the country is lagging behind on the track. Considering an individual's educational information, people are still using papers or documented sheets and there are no structured or centered information stored of an individual starting from his/her beginning of educational life. A centralized database having information stored and updated frequently can overcome this issue and help an individual to access his information, share them with national and international educational institutions for higher studies. However, a centralized information system can help monitor the educational progress of the country under digital methods. Our proposition is to break this barrier and bring all the student information under one roof where we can easily manipulate these information to bring important changes needed. On the contrary, it will help to make the work much easier and help the education sector grow much better than before.

Table of Content

Contents

Chapter 1.....	1
Introduction.....	1
1.1 Objectives.....	2
1.2 Motivation.....	3
1.3 Proposing Distributed Database.....	3
Chapter 2.....	5
Literature Review.....	5
2.1 Education Board Bangladesh Website.....	6
2.2 Top down and bottom up approach.....	6
2.3 Query processes.....	7
Chapter 3.....	8
System Specification.....	8
3.1 Top down Approach.....	8
3.2 Semi join query processing.....	9
3.3 XAMPP.....	10
3.4 PHP.....	11
3.5 HTML 5.....	12
3.6 MySQL.....	13
3.7 Mozilla Firefox.....	14
3.8 Google Chrome.....	15
Chapter 4.....	17
System Architecture.....	17

4.1 Use case diagram.....	17
4.2 ER model.....	18
Chapter 5.....	20
Implementation.....	20
5.1 Database.....	20
5.2 Generating Unique ID.....	24
5.3 SEMIJOIN Query processing.....	25
5.4 HTML.....	27
5.5 CSS.....	27
5.6 JavaScript.....	27
5.7 Implemented Pages.....	28
Chapter 6.....	34
Result and Analysis.....	34
6.1 Login.....	34
6.2 Result Entry.....	36
6.3 Adding new teacher and student.....	37
6.4 Updating Notice and student's Achievements.....	39
6.5 Updating TSR.....	41
Chapter 7.....	45
Conclusion and Future Work.....	45
7.1 Conclusion.....	45
7.2 Future Work.....	45
References.....	46

Table of Figures

Figure 3.1: Top down Approach.....	8
Figure 3.2 Logo of XAMPP.....	11
Figure 3.3 Logo of PHP.....	12
Figure 3.4 Logo of HTML5.....	13
Figure 3.5 Logo of MySQL.....	14
Figure 3.6 Logo of Mozilla Firefox.....	15
Figure 3.7 Logo of Chrome.....	16
Figure 4.1 Use Case Diagram.....	17
Figure 4.2 ER Model.....	18
Figure 5.1 Student Information Table.....	20
Figure 5.2 Teacher Table.....	21
Figure 5.3 Subject Grade Table.....	22
Figure 5.4 Result Table.....	23
Figure 5.5 Student achievement table.....	24
Figure 5.6 Code for Unique ID generation.....	25
Figure 5.7 SEMIJOIN Query processing result.....	25
Figure 5.8 Implemented Pages.....	28
Figure 5.9 Student profile.....	28
Figure 5.10 Educational info.....	29
Figure 5.11 Class wise mark sheet.....	30
Figure 5.12 Students Achievements.....	30
Figure 5.13 Teacher login page.....	31

Figure 5.14 Teacher profile.....	31
Figure 5.15 Notice Board.....	32
Figure 5.16 TSR Home page.....	32
Figure 5.17 Class wise lectures.....	33
Figure 6.1 Student Login Page.....	34
Figure 6.2 Student Dashboard.....	35
Figure 6.3 Result entry form.....	36
Figure 6.4 Student mark sheet.....	37
Figure 6.5 Adding teacher form.....	37
Figure 6.6 New teacher added in the database.....	38
Figure 6.7 Adding student form.....	38
Figure 6.8 New teacher added in the database.....	39
Figure 6.9 Student's achievement entry form.....	39
Figure 6.10 Achievement enters the database.....	40
Figure 6.11 Notice entry form.....	40
Figure 6.12 Published Notice.....	41
Figure 6.13 Teacher Logging in TSR.....	41
Figure 6.14 Uploading files.....	42
Figure 6.15 TSR files table.....	42
Figure 6.16 Fetched information from the table.....	43
Figure 6.17 Admin logs into TSR.....	43
Figure 6.18 Adding a new class.....	43
Figure 6.19 Adding a new subject.....	44

Chapter 1

Introduction

For the past few years digital Bangladesh has been a major term focusing on the technological development of the nation. Emphasizing on the major aspects of digital development throughout the country, Bangladesh has achieved humongous success creating immense number of job opportunities in the ICT sector. In this situation, the country is moving forward towards the technological development in almost every sector. On the education sector, huge numbers of digital classes have been setup throughout the country, but the whole school management system is still out of the technological touch. For an individual, there are no computerized profiling based on his/her educational expertise. This section of education sector is still under papers and files. On the other hand, there are no centralized systems to maintain all the information like student details, school profile, lecture distribution, exam scheduling, etc. To solve these certain issues we came with the concept named ‘iSkool’.

‘iSkool’ is a web based centralized school management system using distributed database which stores all the school management information in a distributed manner, but executes centrally. Here, a student is able to view all his educational information under a personal profile and is be able to analyze results based on different classes he/she attended. Students get education contents through this system and all the necessary information needed from school or board. On the other hand, a shared space for all the teachers throughout country enables to share important lectures and help the school teachers of under privileged zones in conducting the same lectures. A system to generate question papers help the system prevents question leak issues which has been a major problem nowadays. However, payment procedures, admission tests, board exams, etc. can be controlled through the ‘iSkool’ platform. ‘iSkool’ helps to bring the whole educational system under one platform and ensures equal system of educational standards throughout the nation. An individual’s all sorts of education information stored in a platform can help a student in many ways. For applying abroad for higher studies, one can easily extract information needed other than moving from one place to another. However, if a student changes his/her institution based

on certain priority conditions, the student can share all his information with the new institution and that institution can analyze the student's previous records.

So, focusing on all the problems related to the educational system in Bangladesh 'iSkool' can solve most of them and create a centralized system where information accessibility and data extraction can make the system much more efficient.

1.1 Objectives:

- Consisting of separate student and teacher profile the system will enable schools and the systems prioritized users to access information based on different conditions.
- Students get the accessibility to review their profile, see results & analyze, visit school website for information needed, etc.
- A teacher has the accessibility to a file sharing space to share the lectures for the students, schedule exams and review all the contents needed for the lectures.
- Teachers throughout the country will be able to share all their important educational contents with other teachers which can help distribute equal standard of education all over the country.
- The system will be able to prepare exam schedule for schools. However, class routines and calendar can also be made using this system.
- After the data submission of results, school is able to generate mark sheets and send texts to the desired phone numbers of students in case of publishing. Common syllabus and lecture patterns can be provided centrally by the education board and it can ensure same pattern of education throughout the country.
- Every student will have a unique ID generated using their birthdate, board and school ID no. respectively. This ID cannot be changed and remains the same if a student changes his or her institution. Each student gets his or her unique ID and password to log into the system for viewing their profile and updates related to their schools.

- Three different level of admins will control the whole system based on their priorities. The three different levels are head admin, board and school admins. School admins will control the information related to their respective schools, board admins controls everything under the board and head admins has the accessibility over everything on the system.
- The system will be able to handle a huge number of users at the same time.

1.2 Motivation:

Students in Bangladesh are not in availability of their educational information. Here, they cannot access their previous educational profile or see ECA achievements. However, there are no centralized system that can manipulate data like student profile, results every year, changes in pass rate, etc. On the other hand, an analysis of one's educational information can drive them to a certain strategy for his upcoming step. So, iSkool provides a platform where student's information are stored centrally and can be accessible by admins and the student himself based on priorities.

There has always been a difference of education standards based on zones or divisions. A student in Dhaka may get better facilities than a student in a rural areas of Bangladesh. So, to minimize these differences, iSkool provides a file storage page where teachers upload their valuable lectures and students all over the country can access those for enriching their knowledge. This feature can help reduce the gaps between students all over the country and create an equal standard of education throughout the country. iSkool's purpose is to help Bangladeshi education get digitalized and help students to reduce their problems.

1.3 Proposing Distributed Database:

As Bangladesh education board has no centralized system that can control all the information gathering in a distributed manner, we are proposing the system to be implemented using distributed database. Reasons for using distributed database is the availability of the information.

The data that can be replicated into multiple sites and in case if any sites falls down, another site can be used for request processing. For this, failure in one system doesn't result in failure of the whole. Again, data at different sites can be used by users from different locations. As many machines are involved in the system, the load and managing of data is distributed. Thus it helps solving local queries locally and can be targeted into the central database [26].

Considering these massive advantages in distributed database, we came across the idea to implement and manage the system in a distributed design. Previously, the education board needed to collect data from each and every school to store into the only central database. However, our system can improve this time and resource killing procedure. As the system provides access to schools and boards to connect into the central system, data can be collected without hassle and any sort of paper works. This brings a huge flexible option to work under the centralized and distributed option.

CHAPTER 2

Literature Review

In the era of modernization people are tend have information on the blink of an eye. An individual's education information is not out of that need. A centralized system of Bangladesh's educational system having student's information stored with all of his performance ratings, academic management information, server space for sharing lectures, statistical analysis, etc. can be an effective solution which can help the educational system execute fast and be more efficient. The system that we are developing is basically a centralized school management system using distributed database. The whole system will be divided based on boards and then based on schools. During searching information and viewing board exam results, distributed system will reduce the server jams and execute efficiently.

Considering the recent scenario, education boards have their own websites and government is on the process of forcing schools to open their websites & execute all the digital system to run school management processes. To make the school-going generation competent for the 21st century, government has installed modern laboratories in 3,500 schools across the country. In public schools, contents for classrooms are being accessed on computers that the government has already placed in more than 5,500 digital classrooms across the country as part of its Digital Bangladesh initiative. But, there are rarely few schools that run the total administration process through any sort of specialized digital system. On the other hand, few schools are trying to setup websites and develop the management which is enabling a scattered scenario for storing data. These schools are under their individual network, not under any centralized system. So, data accessibility and analyzing in any sort of educational project is very challenging. 'iSkool' ensures a centralized system where all the educational institution's database is under one platform and they can be monitored centrally.

2.1 Education Board Bangladesh Website

Bangladesh has a central website that is used for the purpose only to provide information related to examination dates, result publishing, checking results, education board related information, etc. However, a student is not able to review anything other than his/her JSC, SSC and HSC results. On the other hand, every year thousands of student gets enrolled and this information is under papers and documents which are stored on the server later on. This website often crashes and are not feasible for the users for collecting any sort of information required. The main problem with this site is the server jam during exam results publishing. We have considered few of these issues from the education board website and figured out that a student should be able to view his/her educational history and keep a profile for further use. However, our system distributes priority based work for schools and central site. Schools can enroll the student information and upload them which is sent to the central server. In this way, every school uploads information of their students and edits the previous ones. This is how information is stored with low labor and in a limited amount of time. On the contrary, we have tried to make the whole system using distributed database strategies, so that information handling gets easier, faster and much more effective. Here, a school updates his every year's final year result and a student is able to view all these information. This helps an institution to analyze the student each year and help them improve gradually.

2.2 Top down and bottom up approach

For developing any database two approaches that are used, the top-down and the bottom-up. Sharing the common goal of uniting a system by describing all of the interaction between the processes these approaches appear radically different.

The top-down method starts from the general and moves to the specific. Starting with a general idea of the system this method asks the end users which pattern of information needs to be stored in the database. Then analyst works with the end users to determine the type of data that should be kept in the database. The analyst should have a detailed understanding of the system for using the top-down method. The top-down method may also have shortcomings. In some cases, the analyst and end users might miss something that is important for the system which may lead to unsatisfactory results [27].

Beginning with the specific details the bottom-up approach moves up to the general. The system analyst will inspect all the interfaces that the system has, screens, checking reports and forms. The analyst will work backwards through the system to determine what data should be stored in the database [27].

We consider top down approach to be feasible for our system. Structuring the creation process of a distributed database is the main goal of top down approach. Defining and separating the construction stages in a correct manner, the database architects and other people involved in the construction of a distributed database will have more chances of achieving success in a given project. [24]

2.3 Query processes

Semijoin and join sequence [28] these two approaches have been used to reduce the amount of data transmission required for the stages of distributed query processing. The semijoin operation from R_p to R_q , denoted by $R_q \bowtie R_p$, is defined as follows: Project R_p on the join attribute of the join between R_p and R_q first, and then ship this projection to the site of R_q to remove nonmatching tuples from R_q . In addition to semijoins, join operations can also be used as reducers in processing distributed queries [29, 30]. Using join reducers, a query is translated into a sequence of joins, and each join is implemented locally by shipping one of the operand relations to the site of the other operand so as to exploit parallelism and minimize the processing overhead. Moreover, joins and semijoins can be combined to form an integrated scheme to further improve distributed query processing [29]. Semijoin strategy is about half of the cost of any JOIN strategies [25]. So, we have considered focusing on the implementation of semijoin query processing throughout the database.

Chapter 3

System Specification

In this chapter, we will go through the system structure and procedures of developing the system. We will cover how the system will be built and develop with the necessary strategies and software implementation.

3.1 Top down Approach:

The top-down approach is employed in different computer areas. In distributed databases it correlates a series of stages to the construction of a distributed databases project beginning from the ground and is employed in homogeneous systems [16].

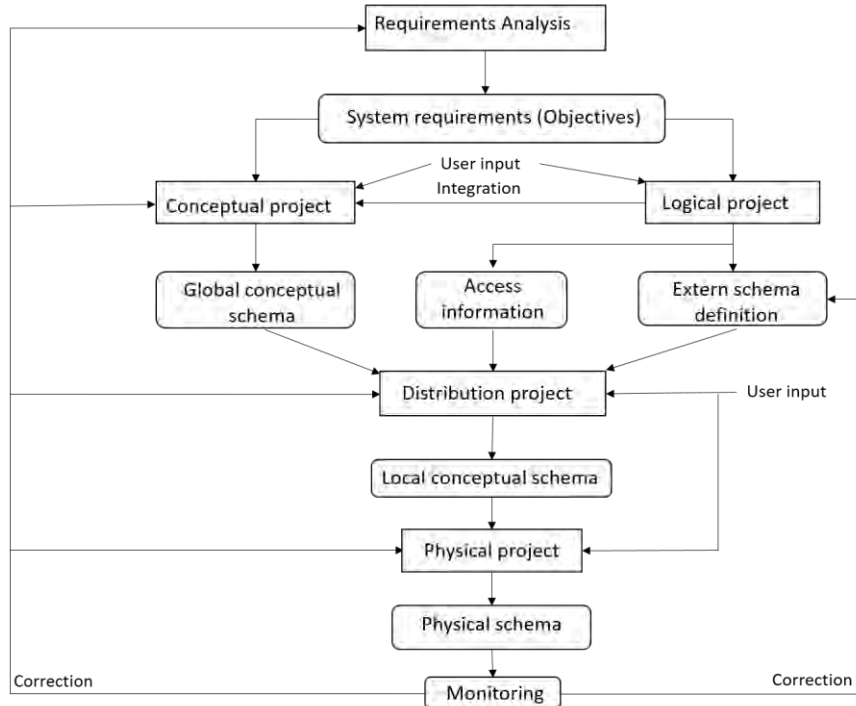


Figure 3.1: Top down Approach

The top down model is divided into certain stages: requirement analysis, conceptual project, logical project, distribution project and physical project. The requirement analysis phase collects the information about the data and the relationships within the system which results in creating a document with requirement specifications. Then, through analyzing requirement specification the conceptual project a conceptual schema diagram with correct data integration [24]. After that, distribution project decides the data allocation and fragmentation. At the physical project, the logical schema is defined into distributed database data model.

3.2 Semi join query processing

For the query processing methods, we have considered semi join query processing to be the best one for our distributed database. If we are having a database distributed into three different sites; operation, ICU and nursing. The schema refers as, ROP (P_Name, birthdate, admitDate, Discharge, Dept), RNU (P_Name, birthdate, GP) and RICU (P_Name, birthdate, Weight, workType) respectively. Now, we want to collect data of P_Name, admitDate, workType and Weight from Operation site. However, the department should be Orthopedics, admission must be before 1st July, P_Name should be common between between Operation and ICU sites and weights must be greater than 80. The query will be: Slect P_Name, admitDate, workType, Weight FROM ROP, RICU WHERE (ROP.P_Name= RICU.P_Name) and (Dept=Orthopedics) and (admitDate> 1st July) and (Weight>80).

According to the **Semijoin** strategy the query processing is as follows:

Step-1:

Restrict ROP (Dept= Orthopedics, admitDate> 1st July).

Project P_Name from restricted ROP.

Step-2: Transmit the result R1 (from Step-1) to ICU site.

Step-3:

Restrict RICU (Weight > 80) (say N2 tuples).

Join R1 and Restricted RICU (say N3 tuples).

PROJECT these tuples over the required attributes.

Step-4: Move the Result R2 (From step-3) to Operation site.

Step-5: JOIN result (R2) with restricted ROP at Operation site.

Cost analysis of the Semijoin Strategy:

Let the cardinality of the Relation R1 be N_1 , transmission cost/attribute be T_{cost-a} , cost per tuple comparison be $C_{comp-tuple}$, and cost per tuple concatenation is $C_{cn-tuple}$.

Processing cost at Step-2:

$$N_1 * T_{cost-a}$$

Processing cost at Step-3:

$$N_1 * N_2 * C_{comp-tuple} + N_3 * C_{cn-tuple}$$

Processing cost at Step-4:

$$N_3 * 3 * T_{cost-a} \text{ (There are three attributes Weight, workType and P_Name)}$$

Processing cost at Step-5:

$$N_1 * N_2 * C_{comp-tuple} + N_3 * C_{cn-tuple}$$

Hence the total processing cost (TPC_{semi-join}) for

Semijoin strategy =

$$\begin{aligned} & N_1 * T_{cost-a} + N_1 * N_2 * C_{comp-tuple} + N_3 * C_{cn-tuple} \\ & + N_3 * 3 * T_{cost-a} + N_1 * N_3 * C_{comp-tuple} + N_3 * C_{cn-tuple} \\ & = (N_1 + 3 * N_3) * T_{cost-a} + 2 * N_3 * C_{cn-tuple} (N_1 * N_2 + N_1 * N_3) * C_{comp-tuple} \dots \dots \dots (1) [25] \end{aligned}$$

However, on the same process if JOIN is applied total cost is:

$$\begin{aligned} & 3 * N_1 * T_{cost-a} + N_1 * N_2 * C_{comp-tuple} + N_3 * C_{cn-tuple} \\ & + (5 * N_3 + 3 * N_1) * T_{cost-a} + N_1 * N_2 * C_{comp-tuple} + N_3 * C_{cn-tuple} \dots \dots \dots (2) [25] \end{aligned}$$

Comparing (1) and (2) we can see that, SEMIJOIN strategy is about half of the cost of any JOIN strategies [25]. In our database, we implemented the semijoin query processing strategies to make it more efficient.

3.3 XAMPP:

XAMPP is an open source cross-platform web server that has been developed by “Apache Friends”. It consists of “Apache HTTP Server, MariaDB database” and PHP and Perl are used as interpreters for scripts. XAMPP stands for Cross-Platform (X), Apache (A), MariaDB (M), PHP (P) and Perl (P). It is easy to create a local web server for testing and deployment purpose due to its lightweight advantage. For setting up a web server it has everything required. Being a cross-

platform XAMPP equally works on Linux, Mac, and Windows. Transitioning local test server to live server is made easy by XAMPP. [2]



Figure 3.2: Logo of XAMPP

3.4 PHP:

PHP is an intuitive server side scripting language. It allows to build logic into the web page content and manipulate data returned from the web browser. It consists of a scripting language and an interpreter. It helps developers defining the behavior and logic needed in a web page. The scripts of PHP are embedded into HTML documents that are served by the web server. The form of a module taken by the interpreter integrates into the web server [3]. After the conversion of scripts into commands the computer then executes to provide the result defined in the script by the web developer. PHP stands for Personal Home Page. It was developed by Rasmus Lerdorf in 1994 and later on created by The PHP Development Team. It is to a great extent been utilized dialect to create sites. Inserting into HTML code it can be utilized in a blend of various types of web layouts and system [14].



Figure 3.3: Logo of PHP.

3.5 HTML 5:

The most standard markup dialect for marking pages and sites, HTML stands for Hypertext Markup Language. The scenario of World Wide Web is shaped into a new dimension by HTML through the utilization of CSS and JavaScript. Consideration of CSS controls the standpoint and design of the web content and the involvement of JavaScript controls the conduct of the substance. HTML components refer to building squares of HTML pages. HTML directions are implemented using HTML labels, for example, `<head/>` and `<input/>`. The most recent form of HTML (HTML5) was introduced on 28 October 2014 by the World Wide Web Consortium (W3C) [5].



Figure 3.4: Logo of HTML5

3.6 MySQL:

MySQL is an open source relational database management system (RDBMS) based on Structured Query Language [13]. It was established by Widenius and Axmark in 1994 and created by a Swedish organization MySQL AB. For its superb execution, demonstrated unwavering quality and convenience, MySQL has turned into the world's driving database decision to web designers because of its superb execution, quality and convenience. Famous tech companies including Google, Facebook, Twitter, Yahoo, etc. consider MySQL database framework [1]. MySQL can run on more than 20 operating systems [4], such as Windows (including Windows Server) or Linux based systems [23]. So, we have considered MySQL to be the best choice for our system.



Figure 3.5: Logo of MySQL

3.7 Mozilla Firefox:

Firefox is a Web program that is simple, fast and in some ways more secure than the Mozilla program from which a lot of its code was initially implemented. Arguably the most well-known web program, Firefox serves its clients with a smoother interface and flexible download speeds. Firefox incorporates the vast majority of the elements with which clients of different programs are recognizable. The primary focus of the Firefox was to incorporate a few new elements other programs did not have at the time, for example, a bookmarks toolbar and selected perusing that allowed clients to rapidly switch forward and backward between a few Web destinations. Offering expanded convenience and usefulness Firefox's significant engineer base keeps on making new features and modules. Since scripting controls, for example, Java and ActiveX can without much of a stretch be selected amid establishment, there is the likelihood of better security [15].



Figure 3.6: Logo of Mozilla Firefox

3.8 Google Chrome:

Google Chrome program is an open source program for diving into the World Wide Web and running Web-based applications. It depends on the open source Chromium extend. Google ejected Chrome in 2008 and issues a few overhauls a year. For Windows, Mac OS X, Linux, Android and iOS working frameworks Google Chrome is accessible. For web security sandboxing-based strategy has been adopted by The Google Chrome program. Every open site keeps running as its own particular procedure, which prevents pernicious code on one page from influencing others. The program additionally boosts Web models, for example, HTML5 and falling templates (CSS) [12].



Figure 3.7: Logo of Chrome

Chapter 4

System Architecture

In this section, we discuss system architecture which will describe our system flow, actors, entities, coordination. ER diagram will describe the full design of the system and flow of the system.

4.1 Use case diagram:

A use case diagram is a graphic depiction of the interactions among the elements of a system. A use case is a methodology used in system analysis to identify, clarify, and organize system requirements [17].

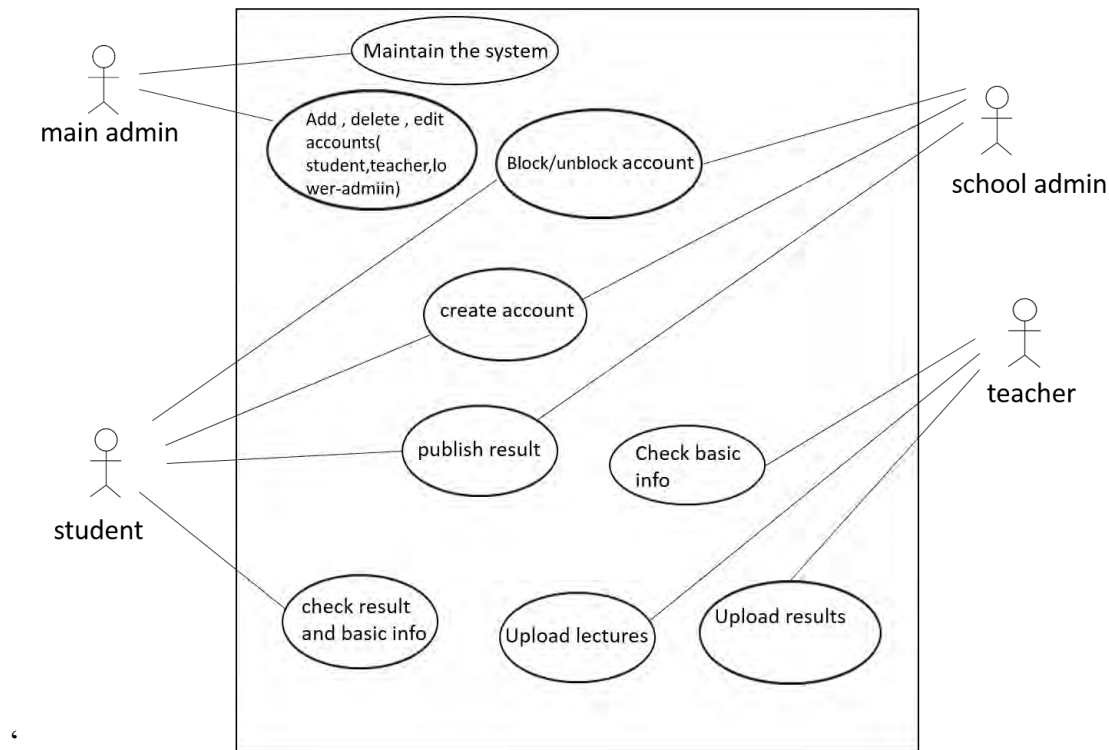


Figure 4.1: Use Case Diagram

Here in our system, there are four actors who will be able to access the system. They are main admin, student, school admin and teacher. All of them will have to log in with authentic

information. Admin will be able to have an overview of the whole system. The admin will also be able to add, delete, edit accounts, etc. without any notice and has the full access to manipulate the whole database. School admins will be able to publish notices, handle the student information and update results for grade sheet. Teachers will upload important lectures, check basic information of individual schools and upload their respective results based on subjects. On the other hand, students will be able to see their results, notices from schools, view their previous records, etc.

4.2 ER model:

ER-modelling is a data modeling technique used in software engineering to produce a conceptual data model of an information system. Diagrams created using this ER-modelling technique are called Entity-Relationship Diagrams, or ER diagrams or ERDs. So we can say that Entity Relationship Diagrams illustrate the logical structure of databases [20].

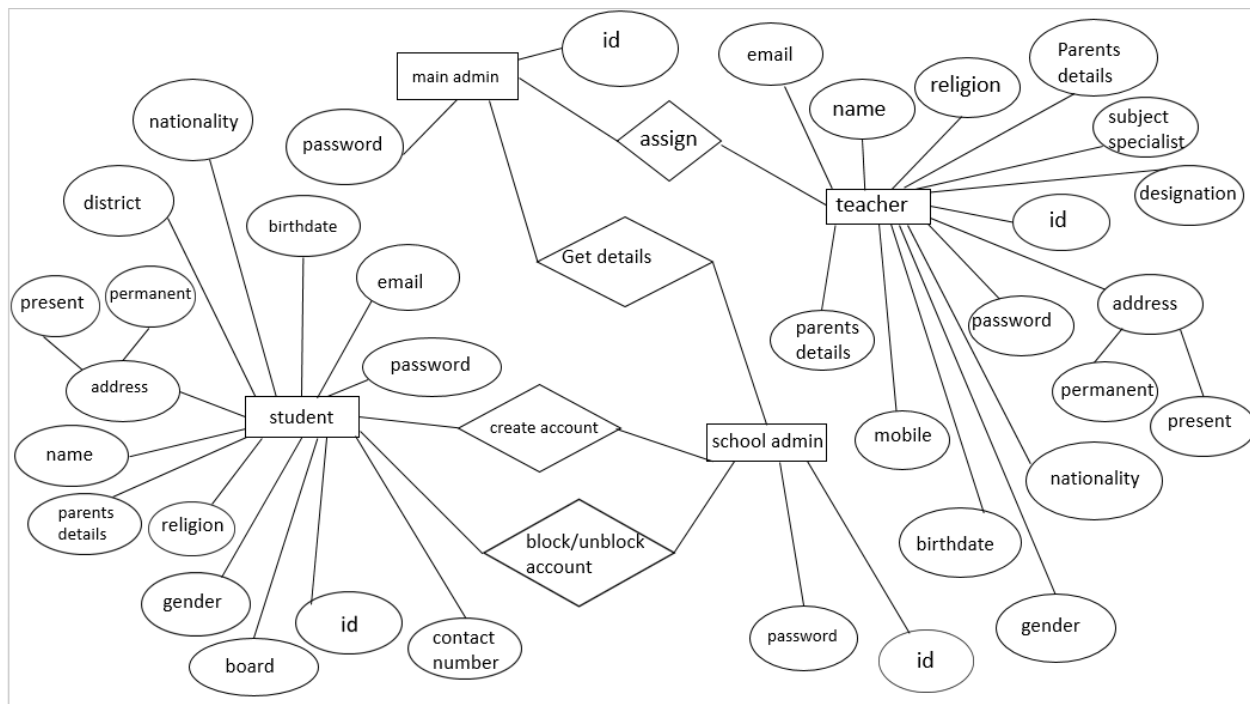


Figure 4.2: ER Model

In the system we developed, we used this database model shown above. There are four main tables in our database. These are a main admin, student, school admin and teacher. Teacher and student table contains all the information about teachers and students respectively. As the information is enrolled in the database a new student or a teacher is updated and a unique ID is generated from the system. Teachers will be able to upload lecture on a platform where all the students throughout the country can view all the uploaded lectures set according to classes and subjects. On the other hand, school admins will maintain the details of all the students under their respective schools, publish results, add or delete students and publish notices. However, main admin has access over all the information in the database.

Chapter 5

Implementation

The system we are trying to implement is based on distributed database. We have used top down approach, SEMIJOIN strategies and used PHP for merging the web languages. The system consists of information related to students, teachers, school admin and main admin. The database maintains a huge list of student and teacher information. However, the database also holds the list of results and achievements for a specific student and shows whenever required. On the other hand, a teacher holds his information for students, updates results and uploads necessary results that can be viewed by any student enlisted in the database.

5.1 Database:

As a new student gets enrolled into a school, admin (school or main) enters all the information necessary like name, date of birth, phone numbers, etc. and the student is added in the database with a unique ID which is generated following certain procedures. For keeping the database lighter all large file like images or patient reports are to be kept on the server or cloud storage [19].

id	student_id	password	name	eiin	birth_date	gender	email	mobile	religion	nationality	district	board
1	170101010001	42b600ffad4a84ee1d468b9309205486	Ashikul Alam Chowdhury	133982	01/23/2017	male	ashikbracu@gmail.com	1829647199	Islam	Bangladesh	Gazipur	Chittagong
2	900102123456	2bc240bb2fa2d1038ef0e9e76a7625a1	Mohidul Alam	133982	01/20/1990	female	zeus@heaven.com	01738413602	illuminate	Mars	Noakhali	Comilla
3	910102222111	91640e6dfef1a9a6ed006bf675cbaa05	Ko KO	133982	01/20/1991	male	koko@gmail.com	016754548455	buddist	Bangladesh	Manikganj	Dhaka
4	930502721131	09efc01a71076b1c95c8ce0bde8ffc2b	Shahidul Alam	133982	05/10/1993	male	asdsadsad@adfasd.com	05499735455	asdasdas	asdsad	Kishoreganj	Chittagong
5	940602085625	c322db98b62752e40e5a81e62c387d95	Tasnim Nur	133982	06/30/1994	female	nurtasnim@gmail.com	01837483094	Islam	Bangladesh	Coxs Bazar	Chittagong
6	940602741422	54800c4f6381d225422cb575c794aed8	Tanzil	133982	06/30/1994	female	nurtasnim@gmail.com	01837483094	Islam	Bangladesh	Coxs Bazar	Chittagong
7	960103246899	153e766c46fb5b2edea9d33de45809b7	Alam	133983	01/20/1996	female	zeus@heaven.com	01738413602	illuminate	Mars	Noakhali	Comilla
8	930502721130	33d3e764e679bf446271e29e51d4628c	Sabid Samayel	133983	05/10/1993	male	asdsadsad@adfasd.com	05499735455	asdasdas	asdsad	Kishoreganj	Chittagong
9	940602085626	4f5fc65b1bc48270634b98a46f7a4021	Rokib	133983	06/30/1994	female	nurtasnim@gmail.com	01837483094	Islam	Bangladesh	Coxs Bazar	Chittagong
10	940602741423	0581d2b1e27f2b9683bcd34fd5893de2	Tasnim Nur	133983	06/30/1994	female	nurtasnim@gmail.com	01837483094	Islam	Bangladesh	Coxs Bazar	Chittagong
11	960102161196	04e250a49ce272df28304c27f614e20d	Nokibul Alam Chowdhury	133983	01/01/1996	male	nokibaust@gmail.com	01845101545	Islam	Bangladesh	Khulna	Chittagong
12	130904547591	2250150b1b2e6da87709ec43fa9e35f2	Md Arif	133983	09/03/2013	male	xyz@gmail.com	01777777777	Hindu	Bangladesh	Joypurhat	Comilla

Figure 5.1: Student Information Table

id	teacher_id	teacher_password	name	des	birth_date	gender	email	mobile	religion	nationality	district	present_add	permanent_add	f_name
1	121212	121212	shahid	Lecturur	01/01/1993	male	shahid@gmail.com	01821309906	Islam	Bangladeshi	Chittagong	Mohakhali	Chittagong	n alam
5	121213	121213	shahid	Lecturur	01/01/1993	male	shahid@gmail.com	01821309906	Islam	Bangladeshi	Chittagong	Mohakhali	Chittagong	n alam
2	1213112	1213112	sabid	professor	02/23/1993	male	sabid@gmail.com	01738413602	Islam	bangladeshi	Dhaka	banani	coxsbazar	K Alam
6	12131133	12131133	sabid	professor	02/23/1993	male	sabid@gmail.com	01738413602	Islam	bangladeshi	Dhaka	banani	coxsbazar	K Alam
10	12131135	12131135	sabid	professor	02/23/1993	male	sabid@gmail.com	01738413602	Islam	bangladeshi	Dhaka	banani	coxsbazar	K Alam
9	1213115	1213115	sabid	professor	02/23/1993	male	sabid@gmail.com	01738413602	Islam	bangladeshi	Dhaka	banani	coxsbazar	K Alam
4	1313131313	1313131313	Sabid Samayel	Assistant Headmaster	06/25/1984	male	sabid@gmail.com	017333333333	Islam	Bangladesh	Tangail	Banani	Banani	Khurshed Alam
7	1313131314	1313131314	Sabid Samayel	Assistant Headmaster	06/25/1984	male	sabid@gmail.com	017333333333	Islam	Bangladesh	Tangail	Banani	Banani	Khurshed Alam
12	13131313145	13131313145	Sabid Samayel	Assistant Headmaster	06/25/1984	male	sabid@gmail.com	017333333333	Islam	Bangladesh	Tangail	Banani	Banani	Khurshed Alam
11	1313131315	1313131315	Sabid Samayel	Assistant Headmaster	06/25/1984	male	sabid@gmail.com	017333333333	Islam	Bangladesh	Tangail	Banani	Banani	Khurshed Alam
8	9090901	9090901	asssss	asdkigas	07/05/2017	male	asdsa@jhasgd	sda	sdif	saa	Rajbari	ser	asdsa	SDFSD
3	9090909	9090909	asssss	asdkigas	07/05/2017	male	asdsa@jhasgd	sda	sdif	saa	Rajbari	ser	asdsa	SDFSD

Figure 5.2: Teacher Table

This is a basic teacher table containing all the basic information of the teachers. A teacher is enrolled by main admin by entering the values in a basic HTML form. As the form is submitted it directly enters the main database. Teacher gets access to upload results of the students and upload lectures of the specific courses necessary. It helps students throughout the country to avail the same resources required.

id	stu_id	class	subject	point
1	179501010001	5	Bangla	3.5
2	179501010001	5	English	4.5
3	179501010001	5	Math	3.3
4	179501010001	5	GEO	4
5	179501010001	2	Bangla	1
6	179501010001	2	English	2
7	179501010001	2	Math	3
8	179501010001	2	GEO	4
9	16222221111	5	Bangla	1
10	16222221111	5	English	4
11	16222221111	5	Math	3
12	16222221111	5	ICT	4
13	930502721130	5	Bangla	1
14	930502721130	5	English	4
15	930502721130	5	Math	3
16	930502721130	5	ICT	4
17	940602085625	5	Bangla	1
18	940602085625	5	English	4
19	940602085625	5	Math	3
20	940602085625	5	ICT	4

Figure 5.3: Subject Grades Table

From this table, for the corresponding student ID, class and subject the GPA of that specific course is provided. This table contains all the GPAs for all the students under the main database.

id	std_id	class	gpa
1	179501010001	5	2.85
3	179501010001	2	3.625
4	16222221111	1	3.525
7	16222221111	2	3.55
8	16222221111	3	4.2
9	960102161196	1	3.85
10	179501010001	3	3.625
11	16222221111	4	4.2
12	960102161196	2	3.85

Figure 5.4 Result Table

This table extracts information from the subject grades table and enters the total GPA of the corresponding student. Upon query the result is shown on the student profile.

id	std_id	date	achiev
2	179501010001	2017-08-05 14:16:20	Got junior scholarship in school
3	16222221111	2017-08-05 14:16:13	Got junior scholarship in school
4	16222221111	2017-08-05 14:15:22	Got scholarship in school
5	16222221111	2017-08-05 14:16:00	became 1st in 100 meter race
6	179501010001	2017-08-09 23:38:59	Got junior scholarship in school
7	179501010001	2017-08-09 23:38:59	<p>Got junior scholarship in school. In has laskdn...
8	179501010001	2017-08-09 23:39:03	Got junior scholarship in school
9	179501010001	2017-08-09 23:39:03	<p>Got junior scholarship in school. In has laskdn...
10	179501010001	2017-08-09 23:39:03	<p>Got junior scholarship in school. In has laskdn...
11	179501010001	2017-08-09 23:39:03	<p>Got junior scholarship in school. In has laskdn...
12	16222221111	2017-08-05 14:15:22	Got scholarship in school
13	960102161196	2017-08-10 02:01:55	Got Sclarship in class 5.
14	960102161196	2017-08-10 02:01:43	Got Junior Scholarship

Figure 5.5: Students achievement table

This table contains all the achievements related to the corresponding student ID. The information is extracted from the table to show on the student profile as his/her achievements. The information is updated by the school admins upon the achievement of any specific student.

5.2 Generating Unique ID

For a new student or a new teacher we selected a certain pattern to be followed in generating a unique ID for the student. The ID is consisted of 12 Digits. First two digit comes from the last two digit of his/her birth year. The next two digit comes from the number of month and the next two digit comes from the corresponding board ID. After this 6 digit generation, the rest of the 6 digit is generated automatically maintain a serial right after the previous student. For example, if a student is born in June 1993, getting enrolled in Chittagong board (Boar ID-02) and is 398756th student, therefore his/her unique ID would be 930602398756.

```

$sel_board = "SELECT * FROM boards";
$run_board = mysqli_query($conn,$sel_board);
while ($rows = mysqli_fetch_assoc($run_board)){
    if ($rows['name'] == $_POST['board']) {
        $value = $rows['value'];
    }
}

$inputstring = $_POST['date'];
list($string1, $string2, $string3) = explode('/', $inputstring);
$arr = str_split($string3, strlen($string3)/2);
$num_str = sprintf("%06d", mt_rand(000001, 999999));
$st_id = $arr[1].$string1.$value.$num_str; //student unique id
$password = md5($st_id);

```

Figure 5.6: Code for Unique ID generation

5.3 SEMIJOIN Query processing

SEMIJOIN strategy is about half of the cost of any JOIN strategies [25]. So, for our database we implemented SEMIJOIN queries wherever we needed to join tables. Here is an example of joining the result table and student information table. For collecting information of a student's GPA for his corresponding classes let's consider that we need the corresponding GPA of a specific student and in the query student_id should be common between them. The query in SQL as follows: SELECT class, gpa FROM Rs, Rr WHERE (Rs.student_id, Rr.student_id) and (class = 5)

std_id	class	gpa
179501010001	5	2.85

Figure 5.7: SEMIJOIN Query processing result

According to the **Semijoin** strategy the query processing is as follows:

Step-1:

Restrict Rs (Rs.student_id, Rr.student_id)

Project class, gpa from restricted Rs.

Step-2: Transmit the result R₁ (from Step-1) to ICU site.

Step-3:

Restrict R_r (class = 5) (say N₂ tuples).

Join R₁ and Restricted R_r (say N₃ tuples).

PROJECT these tuples over the required attributes.

Step-4: Move the Result R₂ (From step-3) to Operation site.

Step-5: JOIN result (R₂) with restricted Rs} at Operation site.

Cost analysis of the Semijoin Strategy:

Let the cardinality of the Relation R₁ be N₁, transmission cost/attribute be T_{cost-a} , cost per tuple comparison be $C_{comp-tuple}$, and cost per tuple concatenation is $C_{cn-tuple}$.

Processing cost at Step-2:

$$N_1 * T_{cost-a}$$

Processing cost at Step-3:

$$N_1 * N_2 * C_{comp-tuple} + N_3 * C_{cn-tuple}$$

Processing cost at Step-4:

$$N_3 * 3 * T_{cost-a} \text{ (There are three attributes Weight, workType and P_Name)}$$

Processing cost at Step-5:

$$N_1 * N_2 * C_{comp-tuple} + N_3 * C_{cn-tuple}$$

Hence the total processing cost (TPC_{semi-join}) for

Semijoin strategy =

$$\begin{aligned} & N_1 * T_{cost-a} + N_1 * N_2 * C_{comp-tuple} + N_3 * C_{cn-tuple} \\ & + N_3 * 3 * T_{cost-a} + N_1 * N_3 * C_{comp-tuple} + N_3 * C_{cn-tuple} \\ & = (N_1 + 3 * N_3) * T_{cost-a} + 2 * N_3 * C_{cn-tuple} (N_1 * N_2 + N_1 * N_3) * C_{comp-tuple} \dots \dots \dots (1) \end{aligned}$$

However, on the same process if JOIN is applied total cost is:

$$\begin{aligned} & 3 * N_1 * T_{cost-a} + N_1 * N_2 * C_{comp-tuple} + N_3 * C_{cn-tuple} \\ & + (5 * N_3 + 3 * N_1) * T_{cost-a} + N_1 * N_2 * C_{comp-tuple} + N_3 * C_{cn-tuple} \dots \dots \dots (2) \end{aligned}$$

Implementation of User Interface:

To develop a standard user interface we compute HTML, CSS and JavaScript code and merged them.

5.4 HTML:

Hypertext Markup Language (HTML) markup language used for creating hypertext documents that are platform independent. HTML documents are SGML documents with generic semantics that are appropriate for representing information from a wide range of domains [7]. HTML markup can represent hypertext news, mail, documentation, and hypermedia, menus of options, database query results and simple structured documents with in-line graphics and hypertext views of existing bodies of information [6].

5.5 CSS:

Cascading Style Sheets (CSS) is the language for designing Web pages, including colors, layout, and fonts. It allows one to adapt the presentation to different types of devices, such as large screens, small screens, or printers. CSS can be used with any XML-based markup language and independent of HTML. The separation of HTML from CSS makes it easier to maintain sites, share style sheets across pages, and tailor pages to different environments [8].

5.6 JavaScript:

JavaScript is Netscape's cross-platform, object-oriented scripting language. Core JavaScript contains a core set of objects, such as Array, Date, and Math, and a core set of language elements such as operators, control structures, and statements. Core JavaScript's extension can be for a variety of purposes by supplementing it with additional objects [11]. Client-side JavaScript extends the core language by supplying objects to control a browser and. Through supplying objects relevant to running JavaScript on a server its Document Object Model and Server-side JavaScript extends the core language.

5.7 Implemented Pages:

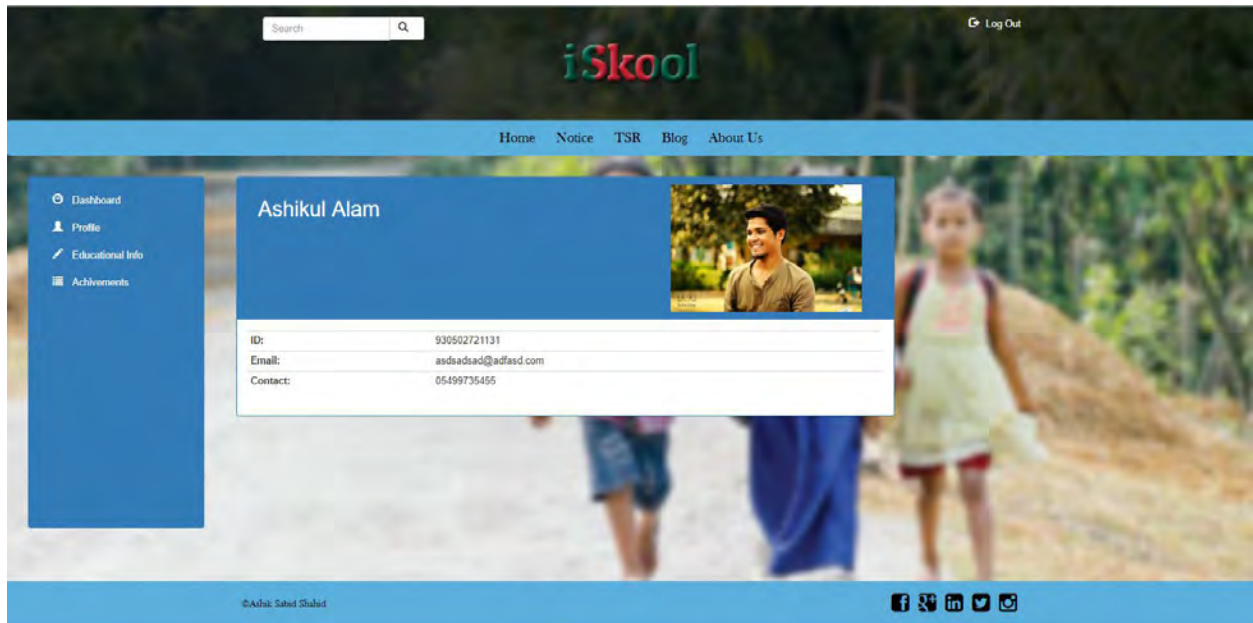


Figure 5.8: Student Main Page

Students Main Page is opened as the student logs in with his/her unique ID and password. There is 'Dashboard' button which shows basic details of the student.

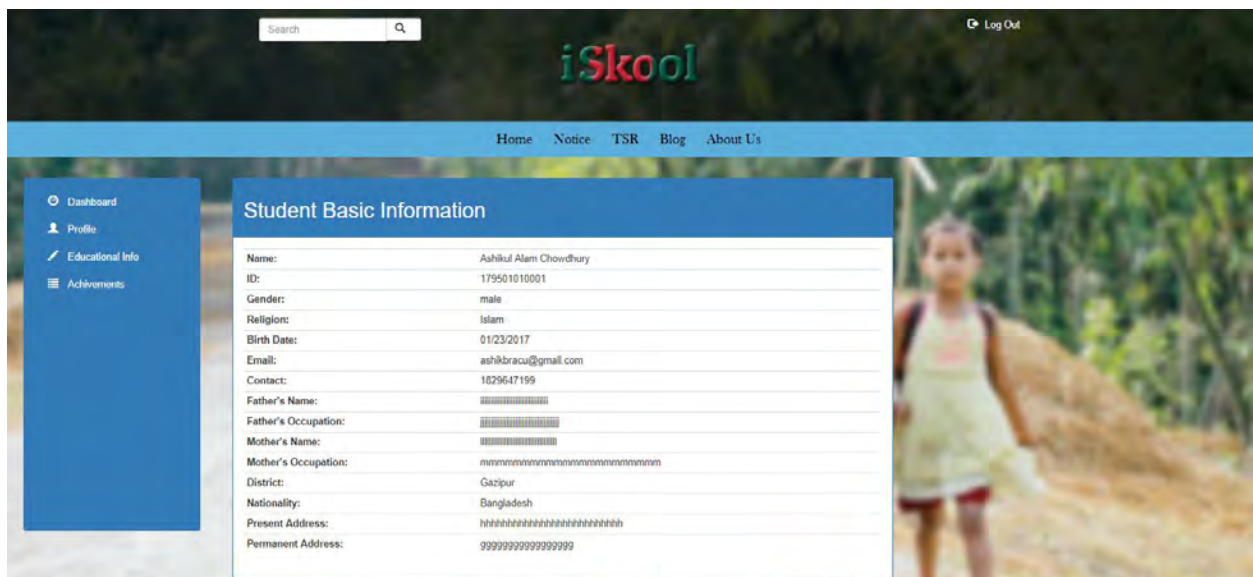


Figure 5.9: Student profile

After logging in, a student is able to view his basic profile by clicking the ‘profile’ button which shows his/her name, ID, gender, date of birth, email, contact, parent’s details, nationality, present address, permanent address, etc.



The screenshot shows the iSkool website interface. At the top, there is a search bar and a 'Log Out' button. Below the header, a navigation bar includes links for Home, Notice, TSR, Blog, and About Us. A sidebar on the left lists menu items: Dashboard, Profile, Educational Info, and Achievements. The main content area features a 'Results' section with a table showing examination results for two classes.

Class	Examination	Grade Point
2	Final	3.625
5	Final	2.85

Figure 5.10: Educational info

Another button is the ‘Educational Info’ that gives access to the academic result of the respective student based on the classes he completed. The results are the total GPA calculated for each class a student completed.



Figure 5.11: Class wise mark sheet

From the educational info page, by clicking anyone of the class a student is able to view his/her mark sheet of that respective class. The list of marks which are viewed on the mark sheet panel gives the total GPA calculating on the educational info page.

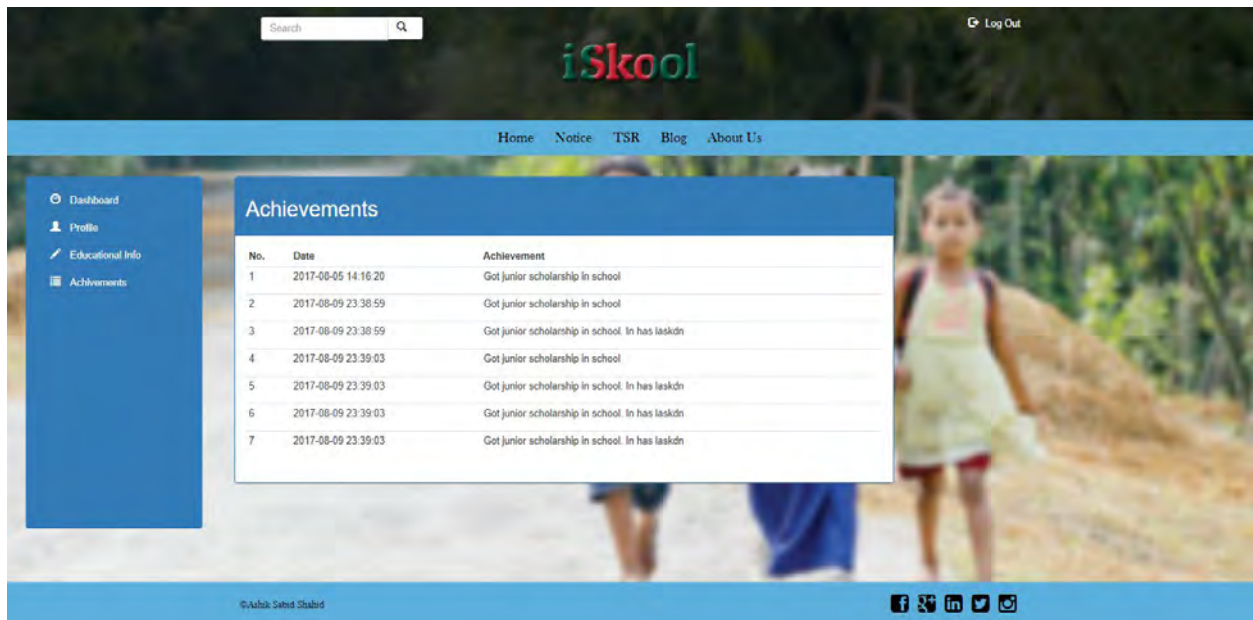


Figure 5.12: Students Achievements

At the ‘Achievements’ section a student is able to view all his/her academic achievements that are updated by the school admin.

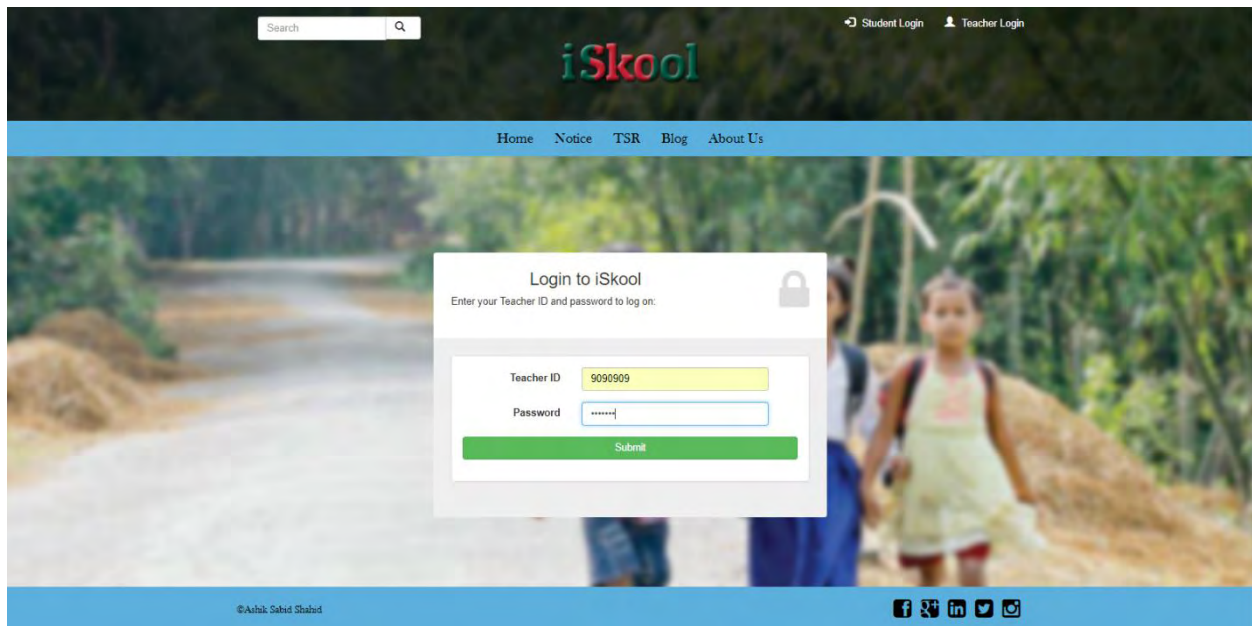


Figure 5.13: Teacher login page

Through entering Teacher ID and password, a teacher can log in and view his details.

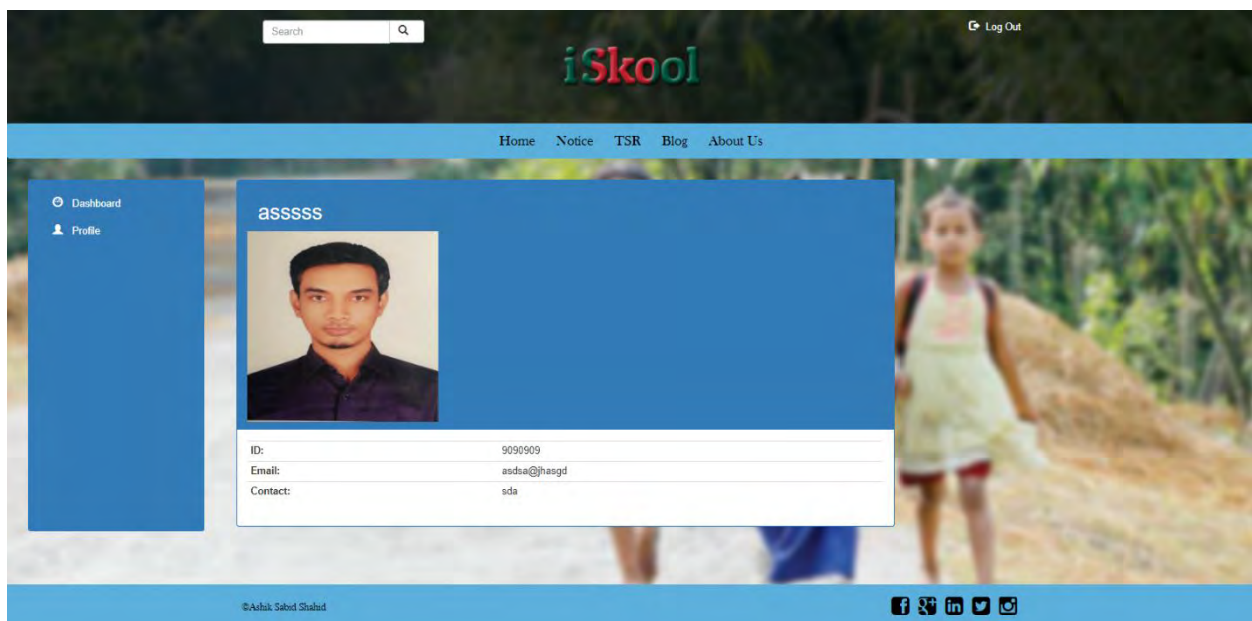


Figure 5.14: Teacher profile

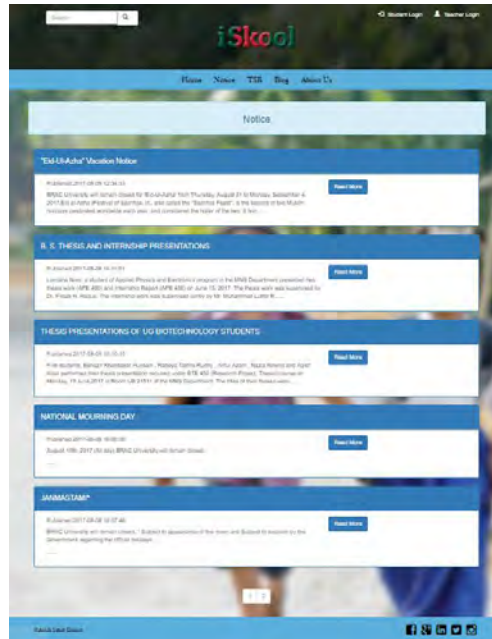


Figure 5.15: Notice Board

Clicking the ‘Notice’ section, students can view the latest notices uploaded by their respective schools.

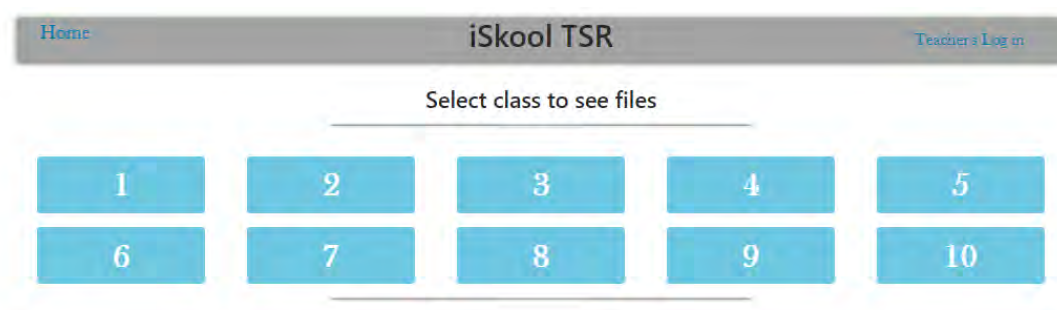


Figure 5.16: TSR Home page

As the user clicks the TSR button, they get access to the TSR Home page which show list of buttons starting from class 1 to class 10.

[Home](#)

iSkool TSR

[Teacher's Login](#)

Select class to see files

1

2

3

4

5

6

7

8

9

10

Class : 4

Select a subject ▼

Calibration between Self - Copy.docx
csaca

Download

eco.docx
csaca

Download

SAUVIK KUNDU.pdf
vfdvfd

Download

course Contents Mat- 215- Sum-17.docx
vfdvfdvfd

Download

Figure 5.17: Class wise lectures

Through clicking any of the class number, TSR shows all the lectures available for the corresponding class. On the right side of the page, 'Select a subject' button enables a student to click on the subject to view lectures for the subject he/she wants to.

Chapter 6

Result and Analysis

The main goal of our system is to provide a centralized database that enable students to view their educational info whenever they need. However, bringing all information under one roof and distributing it to different sites needed is also one of our biggest agenda. The system that we have tried to implement fulfills all the criteria needed for the system. The system enables student view their basic education profile, see their results, achievements, admins can control all the information and students located in different locations are able to access same lectures through the TSR option of our system.

6.1 Login:

This is the initial page where either student or teacher can login to the system with their unique ID and passwords.



Figure 6.1: Student Login Page

The login page contains a 'Home' button to access the homepage of the system. There is a 'Notice' button where notices related to the education board is available. However, 'TSR' button gives you the access to all the files and contents related to class subjects and their lectures.

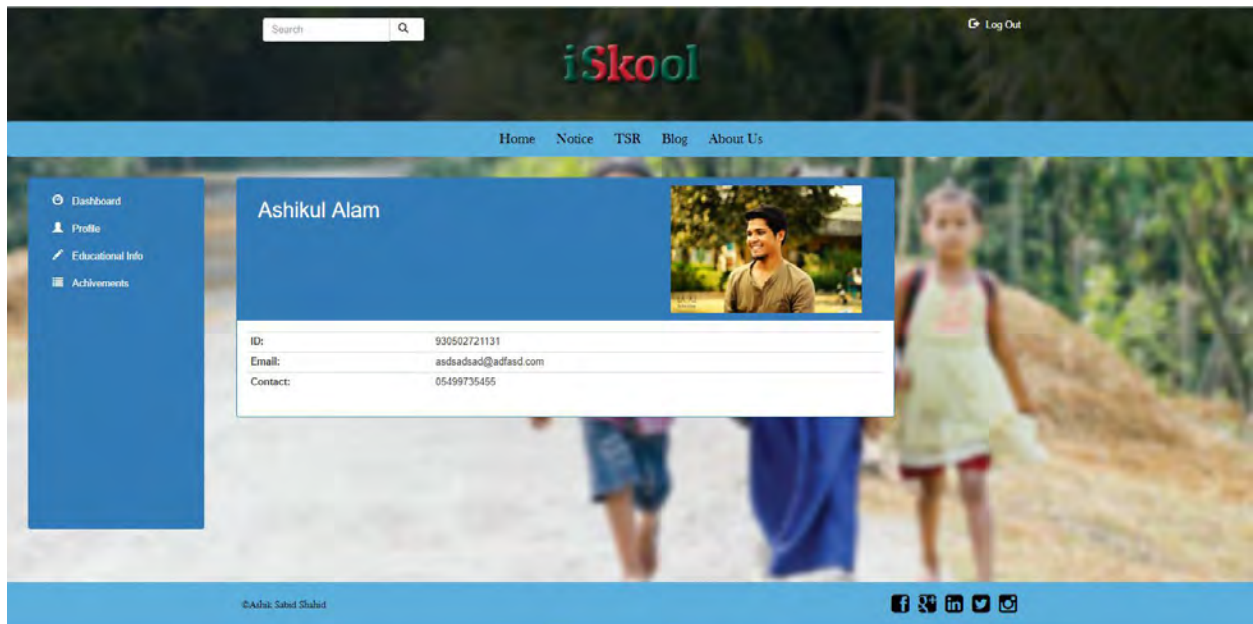


Figure 6.2: Student Dashboard

6.2 Result Entry



The screenshot shows the iSkool website's 'Student Result Entry Form'. The header includes a search bar, a 'Log Out' link, and the iSkool logo. A navigation bar contains links for Home, Notice, TSR, Blog, and About Us. The form itself is titled 'Student Result Entry Form' and is set against a background image of three children. It contains the following fields: a text input for 'ID*' with the value '960102161196'; a dropdown for 'Class*' set to 'One'; a section titled 'Choose Subject and Grade' with four rows. Each row has a subject dropdown and a grade input field. The entries are: Subject 1 (Bangla, 4.5), Subject 2 (English, 4.8), Subject 2 (GEO, 3.9), and Subject 4 (Math, 2.2). A red 'Submit Student Result' button is at the bottom of the form. The footer shows the copyright notice '©Aulizi School Shabab' and social media icons for Facebook, YouTube, LinkedIn, Twitter, and Instagram.

Choose Subject and Grade	
Subject 1	Bangla 4.5
Subject 2	English 4.8
Subject 2	GEO 3.9
Subject 4	Math 2.2

Figure 6.3: Result entry form

By selecting student ID, class, subject and mention the grades a student's mark sheet can be updated from this section. A school admin or teacher can upload a student's result. This result takes entry into the database table of the result table.



Figure 6.4: Student mark sheet

From the information stored in the database, the system calculates the total GPA and displays it on the education info of a student as their mark sheet.

6.3 Adding new teacher and student:

Teacher Profile Entry Form

Teacher ID*	1313131313
Teacher Password*	1313131313
Full Name*	Sabid Samayel
Designation*	Assistant Headmaster
Date Of Birth*	06/25/1984
Gender*	Male
Email Address*	sabid@gmail.com
Mobile Number*	01733333333
Religion*	Islam
Nationality*	Bangladesh

Figure 6.5: Adding teacher form

Adding a teacher can be done by the school admin or the main admin. By filling up all the details of the respective teacher a new teacher can be added to the system.

  	4	1313131313	1313131313	Sabid Assistant Samayel Headmaster	06/25/1984	male	sabid@gmail.com	01733333333	Islam	Bangladesh	Tangail	Banani	Banani	Khurshed Alam
---	---	------------	------------	---------------------------------------	------------	------	-----------------	-------------	-------	------------	---------	--------	--------	------------------

Figure 6.6: New teacher added in the database

The information enters the database teacher table and gets a new entry.

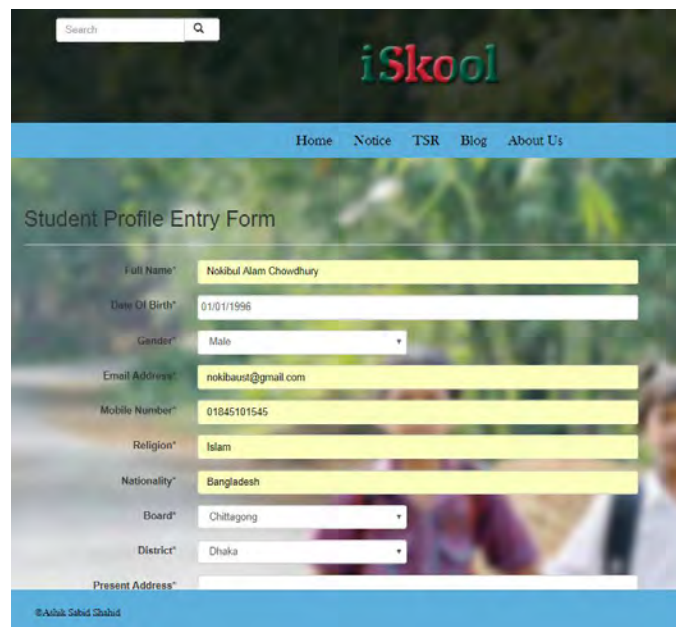


Figure 6.7: Adding student form

Same procedure is followed for adding a new student as well. Admins fill up all the information required and submit to put an entry into the student table of the student. However, on the student registering system, a unique ID is generated using a dynamic strategy that is discussed on section 5.3 Generating Unique ID.

			10	940602741423	0581d2b1e27f2b9683bcd34fd5893de2	Tasnim Nur	06/30/1994	female	nurtasnim@gmail.com	01837483094	Islam	Bangladesh	Coxs Bazar	Chittagong
			11	960102161196	04e250a49ce272df28304c2f7614e20d	Nokibul Alam Chowdhury	01/01/1996	male	nokibaust@gmail.com	01845101545	Islam	Bangladesh	Khulna	Chittagong

Figure 6.8: New teacher added in the database

The information enters the database student table and gets a new entry along with the unique ID.

6.4 Updating Notice and student's Achievements

Search

Log Out

Home Notice TSR Blog About Us

New achievement added successfully and your Student ID is 960102161196

Student Achievements Entry

ID*

Date*

Achievement

File Edit View Format

Formats B I

Got Junior Scholarship

Powered by TinyMCE

Submit Student Achievement

©Ashik Sakib Shabir

f t in t i

Figure 6.9: Student's achievements entry form

The school admins through filling up the corresponding student ID, date of achieving and details of the achievement, they can update any student's achievement history.

10	179501010001	2017-08-09 23:39:03	<p>Got junior scholarship in school. In has laskdn...
11	179501010001	2017-08-09 23:39:03	<p>Got junior scholarship in school. In has laskdn...
12	16222221111	2017-08-05 14:15:22	Got scholarship in school
13	960102161196	2017-08-10 02:01:55	Got Scholarship in class 5.
14	960102161196	2017-08-10 02:01:43	Got Junior Scholarship

Figure 6.10: Achievement enters the database

Collecting the information for the corresponding student and the information enters into the database table.

Figure 6.11: Notice entry form

A school admin enters the notices needed to publish and fills up the form to publish them.

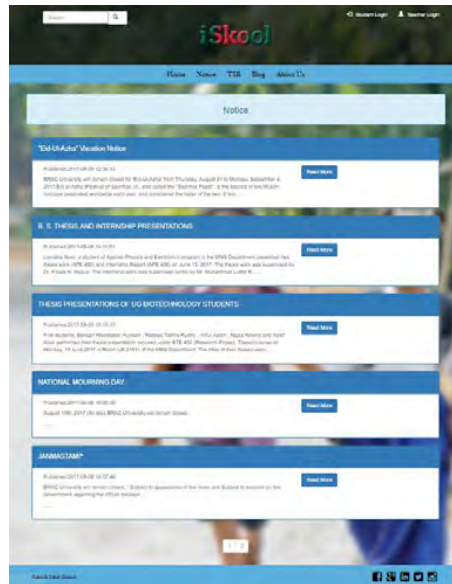


Figure 6.12: Published Notice

Collecting the information from the notice table, the notice gets published with the appropriate details and headlines.

6.5 Updating TSR

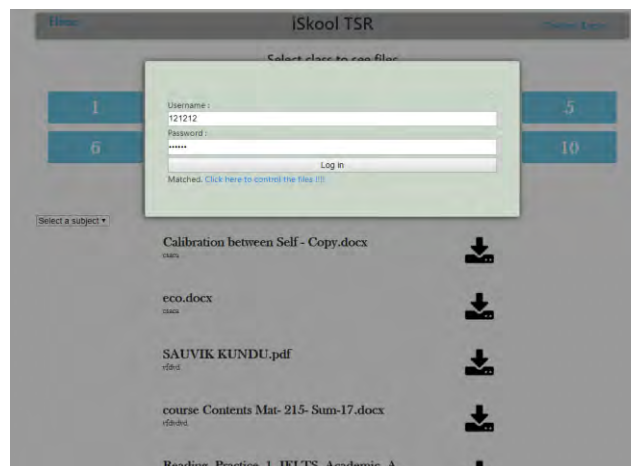


Figure 6.13: Teacher Logging in TSR

From this panel teacher logs into the TSR.

Figure 6.14: Uploading files

Selecting the corresponding class a teacher is able to select course, upload files and put some information about the file to upload it on the TSR.

id_auto	name	info	class	subject	upload_time
24	Calibration between Self - Copy.docx	csaca	10	Bang	05.29.17
25	eco.docx	csaca	10	Bang	05.29.17
26	SAUVIK KUNDU.pdf	vfdvd	10	English	05.29.17
27	course Contents Mat- 215- Sum-17.docx	vfdvdvd	10	ICT	05.29.17
28	Admission_Post_April_2017.pdf	Bangla	1	Bangla	08.11.17
29	01-handout.ddm.pdf	Bangla	4	Bangla	08.12.17
30	Reading_Practice_1_IELTS_Academic_Answers.pdf	English	10	ENGLISH	08.12.17
31	distributed.pdf	Bangla	8	BANGLA	08.12.17
32	Ashikul Alam Chowdhury.pdf	MATH	5	MATH	08.12.17
33	4. 15-18.pdf	MATH	11	MATH	08.12.17
34	Arduino_Bootloader_Shield.pdf	English Lecture	1	ENGLISH	08.13.17

Figure 6.15: TSR files table

After uploading, files enters the database which is fetched to display the files to students enrolled inside the database.

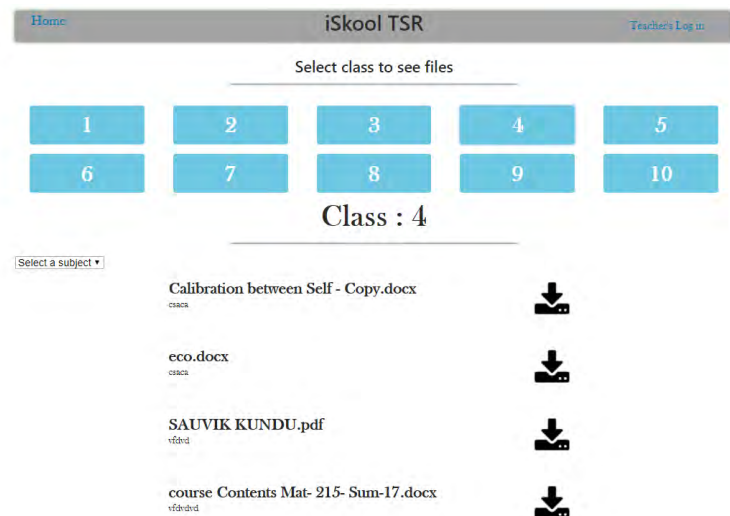


Figure 6.16: Fetched information from the table

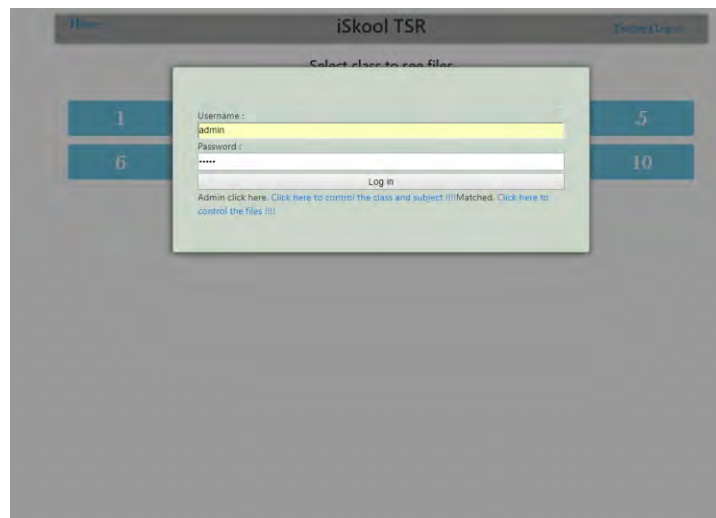


Figure 6.17: Admin logs into TSR



Figure 6.18: Adding a new class

The screenshot shows the 'Add Subject' form on the TSR.COM website. At the top, there is a navigation bar with 'Home', 'TSR.COM', and 'Upload Log out' links. Below this, there are two tabs: 'Class Panel' and 'Subject Panel'. The 'Add Subject' section includes a 'Select Class' dropdown menu currently set to '10', a text input field containing 'General Science', and an 'Add Subject' button. Below the 'Select Class' dropdown, there are two checkboxes: 'BANGLA' and 'ENGLISH', both of which are checked, indicated by an 'X' in a small box next to each label.

Figure 6.19: Adding a new subject

As admin logs into the TSR, admin is able to add a new class or a new subject for any corresponding class. However, there are options to delete or update any class or subject related contents whenever necessary.

Chapter 7

Conclusion and Future Work

7.1 Conclusion:

Considering the digitalization goals of the country, it's high time we should take necessary steps to overcome all the complexities related to the education sector of Bangladesh. Government is trying to enable all the educational institution to digitalize their work scenarios and management system. In the situation, there should be a system where students have feasibility to access their study progress, government can centrally access data needed for education based researches and an individual's education profiling is properly organized. These objectives can easily be filled up by the system we are trying to build. Flexibility and efficiency of 'iSkool' is immense. We just need to put an emphasis on it to develop and grow the system better than before. As we have proposed a system where we can centrally manage all the information of the students of Bangladesh, this system can be replicated dividing into different boards and managing them centrally.

7.2 Future Work:

The system will not be indulged into just what we have implemented, but there are a lot of scope and opportunities to bring in a better developed system. In future, we can setup online classrooms for conducting lecture all together throughout the nation. On the other hand, an app based solution can be developed that can incorporate with 'iSkool' and enable users for one tap based services. We are planning on developing online paying procedures to reduce another hassle that goes in intuitions. Payment through cards, bKash or mobile banking systems can be added for this purpose.

References:

- [1.] Urlocker, M. Zack (13 December 2005). "Google Runs MySQL". *The Open Force*. M. Zack Urlocker. Retrieved 3 August 2010. AdWords was built using the MySQL database
- [2.] MANFIELD, A. (2016). JOOMLA FOR DEVELOPERS. S.l.: LULU COM.
- [3.] N. S. (2010). PHP Essentials. EBookFrenzy.
- [4.] *Miguel, Ian (July 2001). Dynamic Flexible Constraint Satisfaction and its Application to AI Planning (Ph.D. thesis). University of Edinburgh School of Informatics. CiteSeerX 10.1.1.9.6733. hdl:1842/326.*
- [5.] Berners-Lee, Tim; Connelly, Daniel (November 1995). "RFC 1866 – Hypertext Markup Language – 2.0
- [6.] Manu Sporny (2013-01-26). "DRM in HTML5". *The Beautiful, Tormented Machine*. Manu Sporny. Archived from the original on 2014-04-25. Retrieved 2014-05-16.
- [7.] *Raggett, Dave (28 March 1995). "HyperText Markup Language Specification Version 3.0". HTML 3.0 Internet Draft Expires in six months. World Wide Web Consortium. Retrieved 17 June 2010.*
- [8.] Meyer, Eric A. (2006). *Cascading Style Sheets: The Definitive Guide* (3rd ed.). O'Reilly Media, Inc. ISBN 0-596-52733-0.
- [9.] Lie, Hakon W (10 Oct 1994). "Cascading HTML style sheets - a proposal" (Proposal)(0.92). CERN. Retrieved 25 May 2014.
- [10.] Meyer, Eric A. (2006). *Cascading Style Sheets: The Definitive Guide* (3rd ed.). O'Reilly Media, Inc. ISBN 0-596-52733-0.
- [11.] Haverbeke, Marijn (2011). *Eloquent JavaScript*. No Starch Press. pp. 139–149. ISBN 978-1-59327-282-1.
- [12.] Mediati, Nick (July 7, 2009). "Google Announces Chrome OS". PC World. Retrieved July 8, 2009.

- [13.] What is MySQL? - Definition from WhatIs.com. (n.d.). Retrieved August 10, 2017, from <http://searchoracle.techtarget.com/definition/MySQL>
- [14.] Golemon, Sara (2006). *Extending and Embedding PHP*. ISBN 978-0-672-32704-9.
- [15.] Festa, Paul (May 6, 2003). "Mozilla's Firebird gets wings clipped". CNET. Retrieved January 30, 2007.
- [16.] Ozsu, M. Tamer e Valduriez, Patrick. *Principles of Distributed Database Systems*. 2nd Edition. Upper Saddle River: Prentice Hall, 1999.
- [17.] Jacobson Ivar, Christerson Magnus, Jonsson Patrik, Övergaard Gunnar, *Object-Oriented Software Engineering - A Use Case Driven Approach*, Addison-Wesley, 1992.
- [18.] Suzanne Robertson. *Scenarios in Requirements Discovery*. Chapter 3 in Alexander and Maiden, 2004. Pages 39-59.
- [19.] Beynon-Davies, Paul (2004). *Database Systems*. Basingstoke, UK: Palgrave: Houndmills. ISBN 1403916012.
- [20.] A.P.G. Brown, "Modelling a Real-World System and Designing a Schema to Represent It", in Douque and Nijssen (eds.), *Data Base Description*, North-Holland, 1975, ISBN 0-7204-2833-5.
- [21.] Samek, Miro (2008). *Practical UML Statecharts in C/C++, Second Edition: Event-Driven Programming for Embedded Systems*. Newnes. p. 728. ISBN 978-0-7506-8706-5.
- [22.] Taylor Booth (1967) *Sequential Machines and Automata Theory*, John Wiley and Sons, New York.
- [23.] Rittinghouse, J. W., & Ransome, J. F. (2010). *Cloud computing: implementation, management, and security*. Boca Raton: CRC Press.
- [24.] Leniel Macaferi's blog. (n.d.). Retrieved August 13, 2017, from <http://www.leniel.net/2007/12/top-down-approach-in-distributed.html>

- [25.] Alom, B. M., Henskens, F., & Hannaford, M. (2009). Query Processing and Optimization in Distributed Database Systems. IJCSNC International Journal of Computer Science and Network Security, 9, 143-152. Retrieved from <http://paper.ijcsns.org/07book/200909/20090918.pdf>
- [26.] Hiremath, D. S., & Kishor, S.B. (n.d.). Distributed Database Problem areas and Approaches. IOSR Journal of the Computer Engineering (IOSR-JCE) ,15-18. Retrieved from <http://www.iosrjournals.org/IOSR-JCE.html>
- [27] Top-down vs. Bottom-Up Object Database Design. (n.d.). Retrieved August 18, 2017, from http://www.dba-oracle.com/t_object_top_down_bottom_up.htm
- [28] J. S. J. Chen and V. O. K. Li, "Optimizing Joins in Fragmented Database Systems on a Broadcast Local Network," IEEE Trans. & Software Engineering, vol. 15:1, pp. 26-38, 1989.
- [29] M. S. Chen and P. S. Yu, "Combining Join and Semi Operations for Distributed Query Processing," IEEE Trans. of Knowledge & Data Engineering, vol. 5:3, pp. 534-542, 1993.
- [30] M. S. Chen and P. S. Yu, "A Graph Theoretical Approach to Determine a Join Reducer Sequences in Distributed Query Processing," IEEE Trans. Of Knowledge & Data Engineering, vol. 6:1, pp. 152-165,1994.