Centralized School Management System
Using Distributed Database

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**

Introduction .................................................................................................................. 1

1.1 Objectives .............................................................................................................. 2

1.2 Motivation ............................................................................................................. 3

1.3 Proposing Distributed Database ......................................................................... 3

**Chapter 2**

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**

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**

System Architecture ................................................................................................ 17
<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.1 Use case diagram</td>
<td>17</td>
</tr>
<tr>
<td>4.2 ER model</td>
<td>18</td>
</tr>
<tr>
<td>Chapter 5</td>
<td>20</td>
</tr>
<tr>
<td>Implementation</td>
<td>20</td>
</tr>
<tr>
<td>5.1 Database</td>
<td>20</td>
</tr>
<tr>
<td>5.2 Generating Unique ID</td>
<td>24</td>
</tr>
<tr>
<td>5.3 SEMIJOIN Query processing</td>
<td>25</td>
</tr>
<tr>
<td>5.4 HTML</td>
<td>27</td>
</tr>
<tr>
<td>5.5 CSS</td>
<td>27</td>
</tr>
<tr>
<td>5.6 JavaScript</td>
<td>27</td>
</tr>
<tr>
<td>5.7 Implemented Pages</td>
<td>28</td>
</tr>
<tr>
<td>Chapter 6</td>
<td>34</td>
</tr>
<tr>
<td>Result and Analysis</td>
<td>34</td>
</tr>
<tr>
<td>6.1 Login</td>
<td>34</td>
</tr>
<tr>
<td>6.2 Result Entry</td>
<td>36</td>
</tr>
<tr>
<td>6.3 Adding new teacher and student</td>
<td>37</td>
</tr>
<tr>
<td>6.4 Updating Notice and student’s Achievements</td>
<td>39</td>
</tr>
<tr>
<td>6.5 Updating TSR</td>
<td>41</td>
</tr>
<tr>
<td>Chapter 7</td>
<td>45</td>
</tr>
<tr>
<td>Conclusion and Future Work</td>
<td>45</td>
</tr>
<tr>
<td>7.1 Conclusion</td>
<td>45</td>
</tr>
<tr>
<td>7.2 Future Work</td>
<td>45</td>
</tr>
<tr>
<td>References</td>
<td>46</td>
</tr>
</tbody>
</table>
# Table of Figures

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.1</td>
<td>Top down Approach</td>
<td>8</td>
</tr>
<tr>
<td>3.2</td>
<td>Logo of XAMPP</td>
<td>11</td>
</tr>
<tr>
<td>3.3</td>
<td>Logo of PHP</td>
<td>12</td>
</tr>
<tr>
<td>3.4</td>
<td>Logo of HTML5</td>
<td>13</td>
</tr>
<tr>
<td>3.5</td>
<td>Logo of MySQL</td>
<td>14</td>
</tr>
<tr>
<td>3.6</td>
<td>Logo of Mozilla Firefox</td>
<td>15</td>
</tr>
<tr>
<td>3.7</td>
<td>Logo of Chrome</td>
<td>16</td>
</tr>
<tr>
<td>4.1</td>
<td>Use Case Diagram</td>
<td>17</td>
</tr>
<tr>
<td>4.2</td>
<td>ER Model</td>
<td>18</td>
</tr>
<tr>
<td>5.1</td>
<td>Student Information Table</td>
<td>20</td>
</tr>
<tr>
<td>5.2</td>
<td>Teacher Table</td>
<td>21</td>
</tr>
<tr>
<td>5.3</td>
<td>Subject Grade Table</td>
<td>22</td>
</tr>
<tr>
<td>5.4</td>
<td>Result Table</td>
<td>23</td>
</tr>
<tr>
<td>5.5</td>
<td>Student achievement table</td>
<td>24</td>
</tr>
<tr>
<td>5.6</td>
<td>Code for Unique ID generation</td>
<td>25</td>
</tr>
<tr>
<td>5.7</td>
<td>SEMIJOIN Query processing result</td>
<td>25</td>
</tr>
<tr>
<td>5.8</td>
<td>Implemented Pages</td>
<td>28</td>
</tr>
<tr>
<td>5.9</td>
<td>Student profile</td>
<td>28</td>
</tr>
<tr>
<td>5.10</td>
<td>Educational info</td>
<td>29</td>
</tr>
<tr>
<td>5.11</td>
<td>Class wise mark sheet</td>
<td>30</td>
</tr>
<tr>
<td>5.12</td>
<td>Students Achievements</td>
<td>30</td>
</tr>
<tr>
<td>5.13</td>
<td>Teacher login page</td>
<td>31</td>
</tr>
</tbody>
</table>
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 Rp to Rq, denoted by Rq Rp, is defined as follows: Project Rp, on the join attribute of the join between Rp and Rq first, and then ship this projection to the site of Rq to remove nonmatching tuples from Rq. 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](image-url)
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, $R_{OP} (P\_Name, \text{birthdate, admitDate, Discharge, Dept})$, $R_{NU} (P\_Name, \text{birthdate, GP})$ and $R_{ICU} (P\_Name, \text{birthdate, Weight, workType})$ respectively. Now, we want to collect data of $P\_Name, \text{admitDate, workType and Weight}$ 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: Select $P\_Name, \text{admitDate, workType, Weight}$ FROM $R_{OP}, R_{ICU}$ WHERE ($R_{OP}.P\_Name= R_{ICU}.P\_Name$) and (Dept=Orthopedics) and (admitDate> 1st July) and (Weight>80).

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

Step-1:
Restrict $R_{OP}$ (Dept= Orthopedics, admitDate> 1st July).
Project $P\_Name$ from restricted $R_{OP}$.

Step-2: Transmit the result $R_1$ (from Step-1) to ICU site.

Step-3:
Restrict $R_{ICU}$ (Weight > 80) (say N2 tuples).
Join $R_1$ and Restricted $R_{ICU}$ (say N3 tuples).
PROJECT these tuples over the required attributes.

Step-4: Move the Result $R_2$ (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 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:
\[ N1 \times T_{cost} - a \]

Processing cost at Step-3:
\[ N1 \times N2 \times C_{comp} - tuple + N3 \times C_{cn} - tuple \]

Processing cost at Step-4:
\[ N3 \times 3 \times T_{cost} - a \] (There are three attributes Weight, workType and P_Name)

Processing cost at Step-5:
\[ N1 \times N2 \times C_{comp} - tuple + N3 \times C_{cn} - tuple \]

Hence the total processing cost (TPC semi – join) for Semijoin strategy =
\[
N1 \times T_{cost} - a + N1 \times N2 \times C_{comp} - tuple + N3 \times C_{cn} - tuple \\
+ N3 \times 3 \times T_{cost} - a + N1 \times N3 \times C_{comp} - tuple + N3 \times C_{cn} - tuple \\
= (N1 + 3 \times N3) \times T_{cost} - a + 2 \times N3 \times C_{cn} - tuple (N1 \times N2 + N1 \times N3) \times C_{comp} - tuple .............. (1) [25] 
\]

However, on the same process if JOIN is applied total cost is:
\[
3 \times N1 \times T_{cost} - a + N1 \times N2 \times C_{comp} - tuple + N3 \times C_{cn} - tuple \\
+ (5 \times N3 + 3 \times N1) \times T_{cost} - a + N1 \times N2 \times C_{comp} - tuple + N3 \times C_{cn} - tuple .............. (2) [25] 
\]
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](image)

**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].
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 shapes 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 refers 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].
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.
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].
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].

![Use Case Diagram](image)

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](image)
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].

<table>
<thead>
<tr>
<th>id</th>
<th>student_id</th>
<th>password</th>
<th>name</th>
<th>eim</th>
<th>birth_date</th>
<th>gender</th>
<th>email</th>
<th>mobile</th>
<th>religion</th>
<th>nationality</th>
<th>district</th>
<th>board</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>170191618001</td>
<td>4268009d4d4b86e16440b0390054095</td>
<td>Anurag Alam Chowdhury</td>
<td>13388</td>
<td>01/23/2017</td>
<td>male</td>
<td><a href="mailto:anuragalim@gmail.com">anuragalim@gmail.com</a></td>
<td>1825471906</td>
<td>Islam</td>
<td>Bangladesh</td>
<td>Gazipur</td>
<td>Chittagong</td>
</tr>
<tr>
<td>2</td>
<td>90101234556</td>
<td>2b1c0000000000000000000000000000</td>
<td>Mohsin Alam</td>
<td>13382</td>
<td>01/20/1990</td>
<td>female</td>
<td><a href="mailto:zeus@navo.in">zeus@navo.in</a></td>
<td>01734336023</td>
<td>Illuminated</td>
<td>Mars</td>
<td>Noakhali</td>
<td>Comilla</td>
</tr>
<tr>
<td>3</td>
<td>91019222111</td>
<td>91940457900900000000000000000000</td>
<td>Ko KO</td>
<td>13382</td>
<td>01/20/1991</td>
<td>male</td>
<td><a href="mailto:koko@gmail.com">koko@gmail.com</a></td>
<td>0167545432</td>
<td>Buddhist</td>
<td>Mymensingh</td>
<td>Dhaka</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>905092721131</td>
<td>09000878786000000000000000000000</td>
<td>Shahidul Alam</td>
<td>13382</td>
<td>01/10/1993</td>
<td>male</td>
<td><a href="mailto:arodko@gmail.com">arodko@gmail.com</a></td>
<td>01549735453</td>
<td>Arab</td>
<td>Bangladesh</td>
<td>Chittagong</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>904092888526</td>
<td>d322080b462752e04e56a3e2e387b5d9</td>
<td>Tanvir Nur</td>
<td>13382</td>
<td>06/30/1994</td>
<td>female</td>
<td><a href="mailto:nuranurim@gmail.com">nuranurim@gmail.com</a></td>
<td>01837430396</td>
<td>Islam</td>
<td>Bangladesh</td>
<td>Cox's Bazar</td>
<td>Chittagong</td>
</tr>
<tr>
<td>6</td>
<td>904050741425</td>
<td>54505046052012542502576257625920</td>
<td>Tariq</td>
<td>13382</td>
<td>06/30/1994</td>
<td>female</td>
<td><a href="mailto:nuranurim@gmail.com">nuranurim@gmail.com</a></td>
<td>01837430396</td>
<td>Islam</td>
<td>Bangladesh</td>
<td>Cox's Bazar</td>
<td>Chittagong</td>
</tr>
<tr>
<td>7</td>
<td>901032048599</td>
<td>1337e76404f0072600000000000000000</td>
<td>Samad</td>
<td>13383</td>
<td>05/10/1993</td>
<td>male</td>
<td><a href="mailto:mdndana@gmail.com">mdndana@gmail.com</a></td>
<td>01649735453</td>
<td>Arab</td>
<td>Bangladesh</td>
<td>Chittagong</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>901092721130</td>
<td>3337e76404f0072600000000000000000</td>
<td>Sabir</td>
<td>13383</td>
<td>05/10/1993</td>
<td>male</td>
<td><a href="mailto:mdndana@gmail.com">mdndana@gmail.com</a></td>
<td>01649735453</td>
<td>Arab</td>
<td>Bangladesh</td>
<td>Chittagong</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>904050989526</td>
<td>4506505010c272063469a8477a4021</td>
<td>Rakib</td>
<td>13383</td>
<td>06/30/1994</td>
<td>female</td>
<td><a href="mailto:nurebnurim@gmail.com">nurebnurim@gmail.com</a></td>
<td>01837430396</td>
<td>Islam</td>
<td>Bangladesh</td>
<td>Cox's Bazar</td>
<td>Chittagong</td>
</tr>
<tr>
<td>10</td>
<td>904050974143</td>
<td>54505046052012542502576257625920</td>
<td>Tanvir Nur</td>
<td>13383</td>
<td>06/30/1994</td>
<td>female</td>
<td><a href="mailto:nurebnurim@gmail.com">nurebnurim@gmail.com</a></td>
<td>01837430396</td>
<td>Islam</td>
<td>Bangladesh</td>
<td>Cox's Bazar</td>
<td>Chittagong</td>
</tr>
<tr>
<td>11</td>
<td>901092151985</td>
<td>040250840122231342247142427142602679</td>
<td>Noorul</td>
<td>13383</td>
<td>01/10/1996</td>
<td>male</td>
<td><a href="mailto:noorulba@gmail.com">noorulba@gmail.com</a></td>
<td>01845101545</td>
<td>Islam</td>
<td>Bangladesh</td>
<td>Khulna</td>
<td>Chittagong</td>
</tr>
<tr>
<td>12</td>
<td>123995057901</td>
<td>2220155012626669799509392052</td>
<td>Md Arif</td>
<td>13382</td>
<td>06/23/2013</td>
<td>male</td>
<td><a href="mailto:noz@gmail.com">noz@gmail.com</a></td>
<td>01777777777</td>
<td>Hindu</td>
<td>Bangladesh</td>
<td>Joypurhat</td>
<td>Comilla</td>
</tr>
</tbody>
</table>

Figure 5.1: Student Information 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.
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.
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.

<table>
<thead>
<tr>
<th>id</th>
<th>std_id</th>
<th>class</th>
<th>gpa</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>179501010001</td>
<td>5</td>
<td>2.85</td>
</tr>
<tr>
<td>3</td>
<td>179501010001</td>
<td>2</td>
<td>3.625</td>
</tr>
<tr>
<td>4</td>
<td>16222221111</td>
<td>1</td>
<td>3.525</td>
</tr>
<tr>
<td>7</td>
<td>16222221111</td>
<td>2</td>
<td>3.55</td>
</tr>
<tr>
<td>8</td>
<td>16222221111</td>
<td>3</td>
<td>4.2</td>
</tr>
<tr>
<td>9</td>
<td>16222221111</td>
<td>1</td>
<td>3.85</td>
</tr>
<tr>
<td>10</td>
<td>179501010001</td>
<td>3</td>
<td>3.625</td>
</tr>
<tr>
<td>11</td>
<td>16222221111</td>
<td>4</td>
<td>4.2</td>
</tr>
<tr>
<td>12</td>
<td>16222221111</td>
<td>2</td>
<td>3.85</td>
</tr>
</tbody>
</table>
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.
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)

<table>
<thead>
<tr>
<th>std_id</th>
<th>class</th>
<th>gpa</th>
</tr>
</thead>
<tbody>
<tr>
<td>179501010001</td>
<td>5</td>
<td>2.85</td>
</tr>
</tbody>
</table>

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 Rs₁ 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:
$N_3 * 3 * T_{cost} - a$ (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 =

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

+ $N_3 * 3 * T_{cost} - a + N_1 * N_3 * C_{comp} - tuple + N_3 * C_{cn} - tuple$

= $(N_1 + 3 * N_3) * T_{cost} - a + 2 * N_3 * C_{cn} - tuple (N_1 * N_2 + N_1 * N_3) * C_{comp} - tuple............. (1)$

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

$3 * N_1 * T_{cost} - a + N_1 * N_2 * C_{comp} - tuple + N_3 * C_{cn} - tuple$

+ $(5 * N_3 + 3 * N_1) * T_{cost} - a + N_1 * N_2 * C_{comp} - tuple + N_3 * C_{cn} - tuple............. (2)$
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.

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

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.
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

![Image of result entry form]

**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:

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.

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 fills up all the information required and submits 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.
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

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

6.5 Updating 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.

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
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:


book/200909/20090918.pdf


