

**Automation of Course Registration System for  
Chittagong University of Engineering & Technology**

**Mohammad Armaan-Ul-Islam**  
**0904093**

November 2014

# **Automation of Course Registration System for Chittagong University of Engineering & Technology**



This thesis is submitted in partial fulfillment of the requirement for the degree of  
Bachelor of Science in Computer Science & Engineering.

**Mohammad Armaan-Ul-Islam**

**0904093**

Supervised by

**Dr. Asaduzzaman**

**Associate Professor**

**Department of Computer Science & Engineering (CSE)**

**Chittagong University of Engineering & Technology (CUET)**

**Department of Computer Science & Engineering**

**Chittagong University of Engineering & Technology**

**Chittagong-4349, Bangladesh.**

The thesis titled “**Automation of Course Registration System for Chittagong University of Engineering & Technology**” submitted by Roll No. 0904093, Session 2009-2010 has been accepted as satisfactory in fulfillment of the requirement for the degree of Bachelor of Science in Computer Science & Engineering (CSE) abbreviated B.Sc. Engineering awarded by the Chittagong University of Engineering & Technology (CUET).

## **Board of Examiners**

1. \_\_\_\_\_

Chairman

Dr. Asaduzzaman

Associate Professor

Department of Computer Science & Engineering (CSE)

Chittagong University of Engineering & Technology (CUET)

2. \_\_\_\_\_

Member

Dr. Kaushik Deb

(Ex-official)

Professor

Department of Computer Science & Engineering (CSE)

Chittagong University of Engineering & Technology (CUET)

3. \_\_\_\_\_

Member

Department of Computer Science & Engineering (CSE)

(External)

Chittagong University of Engineering & Technology (CUET)

## **Statement of Originality**

It is hereby declared that the contents of this project are original and any part of it has not been submitted elsewhere for the award of any degree or diploma.

-----  
**Signature of the Supervisor**

**Date:**

-----  
**Signature of the Candidate**

**Date:**

## **Acknowledgment**

First of all Glory to the Almighty Allah. All praises & gratitude to him for giving me capability to successfully complete this project and theses work. Then I would like to convey my sincerest thanks and gratitude to my honorable project supervisor Dr.Asaduzzaman, Associate Professor of Department of Computer Science and Engineering, Chittagong University of Engineering and Technology, for his valuable suggestion, constructive advice, encouragement and sincere guidance and care throughout my project work. I also convey special thanks and gratitude to all of my respected teachers of the department and special thanks all of my friends and the staffs of the department for their valuable suggestion and assistance. Finally, I would like to thank my parents especially to my mom for their constant love and support during my study period without whom anything would be possible.

## **Abstract**

Automation of Course Registration System (ACRS) aims to introduce automated student's courses registration using Web based System. The number of students joining undergraduate studies is increasing fast through most universities. Manual registration results in crowding a huge number of students inside the registration halls. Registration employees are suffering a lot as well as the students waiting for hours to accomplish their registration. Even the Professors, Doctors of Philosophy are also ought to play role in this massacre which results in tremendous wastage of manpower and valuable time and the registration process becomes a tedious process which is not a tolerable event in this era of technology. ACRS enlightens on solving these problems enabling the students to register their courses online while skipping hours of waiting and sufferings. Web based system is developed to control the registration process. Online mobile banking system is applied to achieve the payment module. Mobile banking enables students to access accounts and accomplish transactions through a Mobile telephone device. The payment verification is facilitated by Android OS based SMS gateway on API approach which will forward the notifications of payment to the system server as XML data will be analyzed and processed to retrieve information of payment. Then it will validate a student and register him to database. The design, analysis, implementation, and test of the designed system are included. The proposed framework in this project may serve as a good starting point for online registration system in Bangladesh and will bring symphony to this massacre.

# Table of Contents

<b>Acknowledgment.....</b>	<b>iii</b>
<b>Abstract.....</b>	<b>iv</b>
<b>Chapter 1 .....</b>	<b>1</b>
<b>Introduction.....</b>	<b>1</b>
1.1 Introduction.....	1
1.2 Related Works.....	2
1.2.1 Automated Student’s Courses Registration Using CTI .....	2
1.2.2 Online Registration System .....	4
1.3 Motivation.....	5
1.4 Specified Objective .....	7
1.5 Organization of the project paper .....	7
<b>Chapter 2 .....</b>	<b>8</b>
<b>Literature Review .....</b>	<b>8</b>
2.1 Server .....	8
2.2 Front and back end.....	9
2.2.1 Front-end.....	10
2.2.2 Back-end .....	11
2.3 Database Management System .....	11
2.4 SMS/USSD-based transactional payments .....	13
2.4.1 Strengths .....	14
2.4.2 Weaknesses .....	15
2.4.3 Challenges.....	15
2.4.4 Current Situation.....	15
2.5 Transportation of data .....	16
2.5.1 Extensible Markup Language (XML).....	16
2.6 Application Programming Interface .....	17
2.6.1 SMS gateway REST API.....	19
<b>Chapter 3 .....</b>	<b>21</b>
<b>Methodology .....</b>	<b>21</b>

3.1 System Modules.....	21
3.2 Design of modules .....	23
3.3 Software Development Tools .....	27
3.3.1 Construction of User Interface (UI):.....	27
3.3.2 Client Side Programming.....	27
3.3.3 Server Side Programming .....	28
<b>Chapter 4 .....</b>	<b>30</b>
<b>Implementation .....</b>	<b>30</b>
4.1 User Interface.....	30
4.2 Register an account .....	31
4.3 Log In.....	33
4.4 Registration .....	34
4.5 Payment.....	36
4.6 Registration .....	42
4.7 Administration Panel .....	43
4.8 Security .....	46
4.8.1 SQL injection .....	46
4.8.2 MD5 encryption .....	48
4.8.3 Transaction Source Verification .....	49
4.8.4 Institution Code Verification .....	49
4.9 Analysis.....	50
<b>Chapter 5 .....</b>	<b>53</b>
<b>Conclusion .....</b>	<b>53</b>
5.1 Overview.....	53
5.2 Future Recommendation.....	53
<b>Bibliography .....</b>	<b>54</b>
<b>Appendix A.....</b>	<b>56</b>
Source Code .....	56



# List of figures

Figure 1.1: Section code generation (unique number) for CTI [1] .....	4
Figure 2.1: Web Server .....	9
Figure 2.2: Front end and Back end.....	10
Figure 4.1: System modules of ACRS .....	22
Figure 3.2: Flowchart of ACRS .....	26
Figure 4.1: User Interface .....	30
Figure 4.2: Account page .....	31
Figure 4.3: Registering an account .....	32
Figure 4.4: Successful Registration .....	32
Figure 4.5: Unsuccessful account Registration.....	32
Figure 4.6: Unsuccessful Login .....	33
Figure 4.7: Registration application form.....	34
Figure 4.8: Registration type selection.....	34
Figure 4.9: Regular term registration.....	34
Figure 4.10: Short term registration.....	34
Figure 4.11: Unsuccessful application form fill-up .....	35
Figure 4.12: Successful application form fill-up .....	35
Figure 4.13: Post application page.....	35
Figure 4.14: Status page.....	36
Figure 4.15: Payment via bKash.....	37
Figure 4.16: Successful payment .....	38
Figure 4.17: Insufficient amount.....	38
Figure 4.18: Incorrect ID .....	39
Figure 4.19: Payment via Bank.....	40
Figure 4.20: Bank receipt.....	40
Figure 4.21: Unsuccessful payment.....	41
Figure 4.22: Successful payment .....	41
Figure 4.23: Downloaded Registration form .....	42
Figure 4.24: List of successful payments.....	43
Figure 4.25: Hall administration login.....	44

Figure 4.26: Bank administrator login .....	45
Figure 4.27: Super administrator login .....	45
Figure4.28: Fake payment SMS detection.....	49
Figure 4.30: Google Developers PageSpeed Insights.....	51
Figure 4.31: Google Developers PageSpeed Insights report on cuet.ac.bd .....	52
Figure 4.32: Google Developers PageSpeed Insights report on ACRS.....	52

# Chapter 1

## Introduction

### 1.1 Introduction

The number of students joining undergraduate studies is increasing fast through most universities. The Students have to register themselves in each semester in order carry on their course .The process also includes payment to the accounts under the organization. Manual registration process includes involvement of tons of papers, lots of documents. A non-permissible amount of manpower is required to carry out the system while they are given the task of keeping track of the record of each student's hall fees, registration fees; providing necessary application forms to the applicants, validating the forms, ordering and delivering the papers from desk to desk and at last storing the documents in files and databases. Even the Doctors of Philosophy are also ought to play role in this massacre, Instead of providing their valuable lectures to the students they are found giving signatures to these application forms. Again the students have to wait for hours for the signatures and teachers are also wasting their valuable time in such act instead of giving attention to their researches and providing lectures. Such tremendous wastage of manpower and precious time is a curse to the system. Student have to be present physically in the banks and registration halls on the date specified. A huge number of students are present at a time inside the registration halls which results in chaos. In such circumstances employees are suffering a lot as well as the students waiting for hours to accomplish their registration. Even if a student is not able to be present physically in the institution in particular time interval, he or she has to wait until it is possible for him or her to be physically present in the institution and in many circumstances he/she has to pay extra charge or fine he/she doesn't have the opportunity to register from remote location or their home. Thus the registration process existing is a very slow and tedious process.

Automation of Course Registration System (ACRS) offers remedy to such chaos. Automation of Course Registration System (ACRS) is an application that would let the registration system to be led automatically. The system proposed is a web based system. The web server would keep track of records of the students regarding their hall fees, due fine and registration fees.

When a student enters the site using any computer or mobile phone web browser has to enter his/her ID and the necessary information required including the semester he/she wants to be registered in. Then the server would automatically acknowledge him of his registration status, registration and hall fees and fines due. It will also provide the account number where he/she has to pay. Now students are facilitated with the option of using mobile banking systems available and pay through his own mobile phone account (if he/she has any) or from any booths nearby his/her current location. Thus the system offers the student the independency from being physically present. The payment verification is facilitated by Android OS based SMS gateway on API approach. The android application will forward the notifications of payment through Short message service (SMS) to the system server as XML data. The System server will analyze and process the data and retrieve information of payment. Then it will validate a student and register him to database. Otherwise he/she can also reach the bank physically and pay thereby also.

## **1.2 Related Works**

Previous work covers a wide variety of topics such as Automated Student's Course Registration Using Computer Telephony Integration, Online Registration system for university of Jordan. They are described below.

### **1.2.1 Automated Student's Courses Registration Using CTI**

Automation of registration process is common since rapid development of network. There have been some Automated Registration Systems operating in various technologies. Such as M. Fahmy [1] developed an Automated Student's Courses Registration System Using CTI. This technology stands for Computer-Telephony Integration. This research project aims to introduce automated student's courses registration using computer- telephony integration. Technology Application Programming Interface (TAPI) controls are used to develop a CTI application for accessing and updating registration. Voice processing and Computer Telephony Integration (CTI) has important applications in education. An overview of ideas to help design and build new computer telephony and voice-processing application for automated student's courses registration is discussed in this research. This technique allows

students to register for courses without waiting in long queues. The main features of such system are discussed in the following. Students call the system and identify themselves by entering their student registration number. Then they select the course they wish to attend. The system automatically checks if this course is required and if it is opened, whether the student has already taken it, and whether there is room still available in the course. Also the system checks to see if the course has prerequisite or not. If so, the system checks if the student has already taken the prerequisite. The system repeats each choice for verification after a selection is made. If positive, the student is automatically registered in the course. When completed, a copy of his/her completed course schedule is faxed or sent by mail to him/her. Different priority levels can be given to students if necessary, depending on whether they are graduating or not. A system, running on a 12-port platform, can substantially reduce waiting time and frustration for more than 7,000 students who can commonly use it for the sake of course registration. Moreover, if registration periods overlap with courses, this system will substantially reduce absences and late arrivals in class. The system can also handle the different "drop- and -add" changes in student's courses. These requirements include system hardware, system software, and pre-recorded prompts.

The system hardware of ARSR is divided into three include system hardware, system software, and pre-recorded prompts that are used to pay attentions to the caller. Hardware categories are personal computer based, telephony interface (Voice card), and telephone lines. Each one of these categories is discussed in the following. Voice cards require computer hardware resources in order to perform voice processing. Cards by different manufacturers have different resource requirements. These cards include Input/ Output (I/O) port addresses, hardware interrupts (IRQ's) and dual port (shared) memory. PC-Based could be either ISA-16 bit or PCI-32 bit .An dialogic voice card is installed in 16 bit ISA slots. Telephony Interface represents the interface between the telephone lines that the caller uses, and the computer hardware and software components providing Integrated Services Digital Network (ISDN) primary rate interface service termination and call processing for up to 30 voice channels in a single PC slot with rate of 2.148 Mbps. It handles all telephony signaling and performs all tone detection, and Audio/Voice signal processing tasks. Software requirements are Windows NT or above as environment, Voice Driver as an interface between the two components (Computer and Telephone), and telephony toolkit as Telephony Application Programming Interface (TAPI). Voice Driver is a software module that provides a defined interface between a computer telephony application program and the firmware interface.

The firmware is a set of program instructions that reside on an expansion board. Pre-recorded prompts refers to a voice response application that allows users to enter and retrieve information over the telephone . One technique of prompts is text-to-speech, which is converting text into voice output using speech synthesis techniques. Although initially used by the blind to listen to written material, it is used extensively to convey information via telephone for everyone. Another technique implies recording from personal voice in a professional studio which is more expensive than the former.

When the student calls the system, the computer will answer him after two rings. Then the system will ask the student enter his/her identification number (ID). When the student finishes entering ID , the system will receive necessary information via numbers pressed by the student. As example given 2002 2 222 102 1 refers to year 2002, 2<sup>nd</sup> semester, department code 222, course code 102, and section id 1.

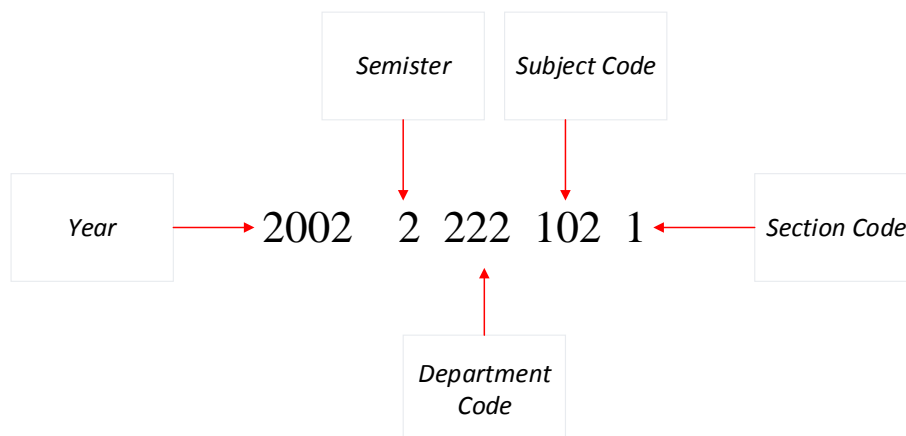


Figure 1.1: Section code generation (unique number) for CTI [1]

## 1.2.2 Online Registration System

A. M. Al-Shaikh [2] developed an Online Registration System. The system is a 3-Tier web-based system. 3-Tier Architecture is a Client/Server Architecture in which the user interface, functional process logic (business rules), computer data storage, and data access are developed and maintained as independent modules, most often in different platform. The system's database is implemented using Microsoft SQL Server. This layer provides high connectivity

and availability, plus, it provides system developers with the ability to manage and administer their databases easily, especially using the Graphical User Interface (GUI) of its Management Studio. Using MS-SQL Server as a Relational Database Management System (RDBMS) of the entire solution gives the user the ability to create Server-Side Cursors to iterate programmatically through different table records and manipulate them row by row. At development time, developers may need to process resulting records at the server without the need to use another programming language, i.e. by means of the built-in functionality of the RDBMS. Never forgetting the use of triggers to perform actions on data upon insertion, deletion, or updating. The Application Layer contains the User Interface (UI), Business Rules, and the Data-access Components. By using the system, most problems used to be faced by the UEGE's administration and college registrars were now eliminated by the means of the Online Registration Subsystem, which allows students to enter to the system immediately once they fill the required application form. Now, there's no need to the coordinator to make long calls to get the number of students currently enrolled into the exam. Plus, by monitoring the instantaneous insert/update/delete operations done by the system, UEGE's administration can detect any type of errors that may enter the database immediately once they occur. Also, there's no need now for other activities to wait the end of the registration duration, since the Reporting Subsystem give the administration the necessary let them predict approximate student numbers, specializations, and colleges they came from. Finally, using paper and fax correspondence have been deducted for the Repository Subsystem which allows System Administrator to upload the necessary files immediately to the system and announce their upload to the users by the news bar associated with this is an efficient system.

### **1.3 Motivation**

A huge number of students are present at a time inside the registration halls which results in chaos. The Students have to register themselves in each semester in order carry on their course. Student have to be present physically in the banks and registration halls on the date specified. Manual registration results in crowding a huge number of students inside the registration halls. Even if a student is not able to be present physically in the institution in particular time interval, he or she has to wait until it is possible for him or her to be physically present in the institution and in many circumstances he/she has to pay extra charge or fine. He/she doesn't have the opportunity to register from remote location or their home. The number of students joining

undergraduate studies is also increasing fast through most universities. The process also includes payment to the accounts under the organization. Manual registration process includes involvement of tons of papers, lots of documents and store house of records. A non-permissible amount of manpower is required to carry out the system while they are given the task of keeping track of the record of each student's hall fees, registration fees; providing necessary application forms to the applicants, validating the forms, ordering and delivering the papers from desk to desk and at last storing the documents in files and databases. Even the PhD degree holders are also ought to play role in this massacre, they are ought to verify the students identity and give signatures in the forms. Again the students have to wait for hours for the signatures and teachers are also wasting their valuable time in such act instead of giving attention to their researches and providing lectures. Such tremendous wastage of manpower and precious time is a curse to the system. In such circumstances employees are suffering a lot as well as the students waiting for hours to accomplish their registration. Thus the registration process existing is an age old process in this era of technology. Thus the system is proposed to enlighten on solving these situation and offering remedy to such chaos.

When starting to implement a CTI system [1], it is important to notice that the student will not see a system interface. He/ She will hear only comments through the phone line and response through telephone digits only. The system should be as simple as possible and be well organized to let the student navigate through the system and inform about the stage. System does not provide any process for payment, it only register the students into database. Thus Student has to pay manually One technique of prompts is text-to-speech, which is converting text to speech using synthesis techniques, which is also a lengthy process. Thus the System becomes inefficient in some cases.

Online Registration System [2] designed does not support AJAX (Asynchronous JavaScript and XML) thus the system takes a long term to load, which makes the system slow and inefficient to interact and the System does not provide any process for payment, it only register the students into database. Thus Student has to pay manually.



## 1.4 Specified Objective

The objective of the system proposed is to facilitate the administrative and academic activities. To obtain the task we have to accomplish objectives below:

- To develop a Web based application.
- To Configure a Web Server to serve the web application.
- To define a money transaction process.
- And finally to develop a complete system to Automate Course Registration process.

## 1.5 Organization of the project paper

Chapters following emphasis on the description of the project ACRS subsequently.

**Literature review:** The chapter enlightens on the fundamental Concepts of Server, Front end, Back end, Database Management system (DBMS), SMS/USSD based transaction system, Application programming interface (API).

**Methodology:** The chapter contains the detailed description on designing the modules and sketching them out which includes the map design from scratch to the top.

**Implementation:** Implementation chapter is about the detailed description of the system procedures of the work. This chapter contains step by step screenshots which will illustrate the project.

**Conclusion and future recommendations:** The chapter concludes our project paper and some recommendations given here for the better hope in future.

## Chapter 2

### Literature Review

#### 2.1 Server

A web server is a computer system that processes requests via HTTP, the basic network protocol named Hypertext Transfer protocol used to distribute information on the World Wide Web. The term can refer either to the entire system, or specifically to the software that accepts and supervises the HTTP requests [6].

- 1) In information technology, a server is a computer program that provides services to other computer programs (and their users) in the same or other computers.
- 2) The computer that a server program runs in is also frequently referred to as a server (though it may be used for other purposes as well).
- 3) In the client/server programming model, a server is a program that awaits and fulfills requests from client programs in the same or other computers. A given application in a computer may function as a client with requests for services from other programs and also as a server of requests from other programs.

Specific to the Web, a Web server is the computer program (housed in a computer) that serves requested HTML pages or files. A Web client is the requesting program associated with the user. The Web browser in your computer is a client that requests HTML files from Web servers. The primary function of a web server is to store, process and deliver web pages to clients. The communication between client and server takes place using the Hypertext Transfer Protocol (HTTP). Pages delivered are most frequently HTML documents, which may include images, style sheets and scripts in addition to text content.

A user agent, commonly a web browser or web crawler, initiates communication by making a request for a specific resource using HTTP and the server responds with the content of that resource or an error message if unable to do so. The resource is typically a real file on the server's secondary storage, but this is not necessarily the case and depends on how the web server is implemented.

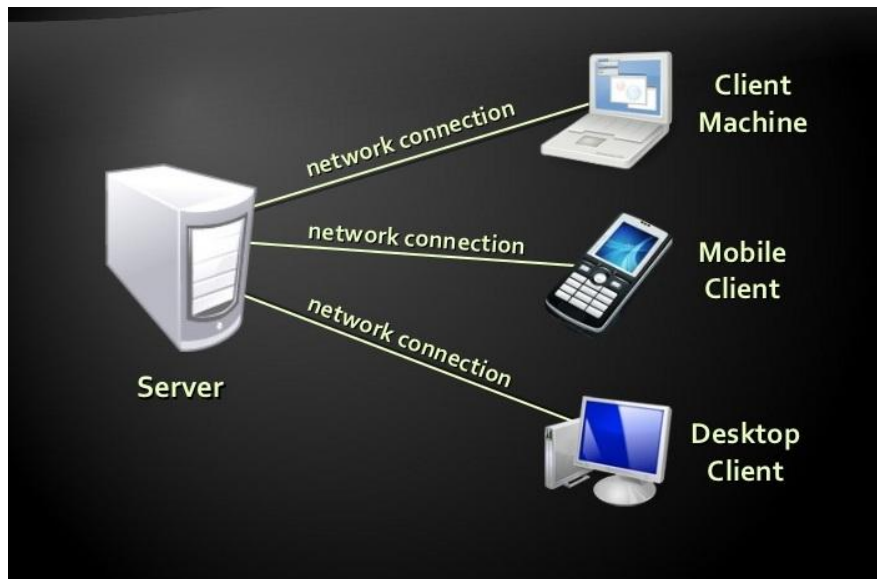


Figure 2.1: Web Server

While the primary function is to serve content, a full implementation of HTTP also includes ways of receiving content from clients. This feature is used for submitting web forms, including uploading of files.

Many generic web servers also support server-side scripting using Active Server Pages (ASP), PHP, or other scripting languages. This means that the behavior of the web server can be scripted in separate files, while the actual server software remains unchanged. Usually, this function is used to create HTML documents dynamically ("on-the-fly") as opposed to returning static documents. The former is primarily used for retrieving and/or modifying information from databases. The latter is typically much faster and more easily cached but cannot deliver dynamic content.

## 2.2 Front and back end

Front-end" typically means the parts of the project a user interacts with--such as the graphical user interface or command line. Back-end means the parts that do the work, but the user is unaware of or cannot see. Databases, services, etc.

We can think of it like a restaurant where you can't see the kitchen. As a customer you see the front-end the decorations, menus, staff. Meanwhile the kitchen and stockroom are out of view.

The easiest way to differentiate front-end and back-end development is to say that web designers are more focused on the front-end (design, typography, colors, etc.), and web programmers are more concerned about back-end development using specific languages (MySQL, PHP, other database languages, etc). Front-end is what you see; back-end is how it works.

### 2.2.1 Front-end

These days, front-end development refers to the part of the web users interact with. In the past, web development consisted of people who worked with Photoshop and those who could code HTML and CSS. Now, developers need a handle of programs like Photoshop and be able to code not only in HTML and CSS, but also JavaScript or jQuery, which is a compiled library of JavaScript.

Most of everything you see on any website is a mixture of HTML, CSS, and JavaScript, which are all controlled by the browser. For example, if you're using Google Chrome or Firefox, the browser is what translates all of the code in a manner for you to see and with which to interact, such as fonts, colors, drop-down menus, sliders, forms, etc. In order for all of this to work, though, there has to be something to support the front-end; this is where the backend comes into play.

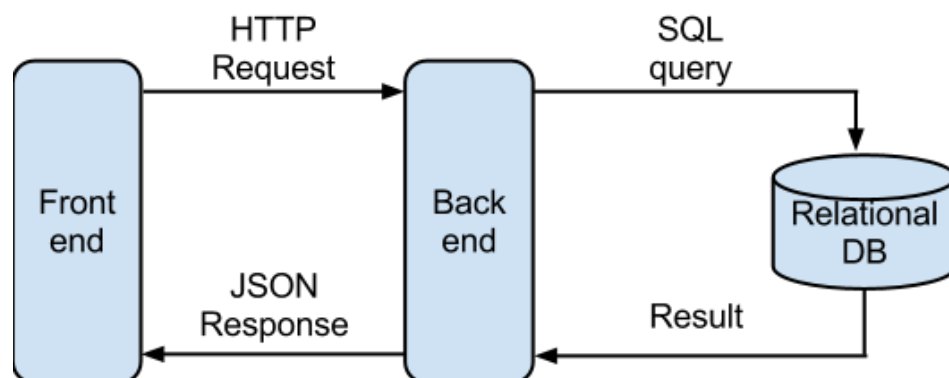


Figure 2.2: Front end and Back end

### **2.2.2 Back-end**

Web programmers are concerned with launching websites, updates, and maintenance, among other things. All of that works to support the front-end of the website. The back-end has three parts to it: server, application, and database.

To better explain how all of this works, let's use the example of a customer trying to purchase a plane ticket using a website. Everything that the customer sees on the webpage is the front-end, as we have explained before, but once that customer enters all of his or her information, such as their name, billing address, destination etc. the web application stores the information in a database that was created previously on the server in which the website is calling for information.

The web application creates, deletes, changes, renames, etc. items in the database. For example, when a customer purchases a ticket, that creates an item in the database, but when they have a change in their order or they wish to cancel, the item in the database is changed.

A server, in the simplest form, is a computer accessed remotely that runs software to fulfill requests from clients. In our example, the server the customer's browser is communicating with is where the database is stored and modified.

In short, when a customer wants to buy a ticket, the backend operation is the web application communicating with the server to make a change in a database stored on said server. Technologies like PHP, Ruby, Python, and others are the ones backend programmers use to make this communication work smoothly, allowing the customer to purchase his or her ticket with ease.

## **2.3 Database Management System**

A database management system (DBMS) is a collection of programs that enables you to store, modify, and extract information from a database. There are many different types of DBMSs, ranging from small systems that run on personal computers to huge systems that run on mainframes.

From a technical standpoint, DBMSs can differ widely. The terms relational, network, flat, and hierarchical all refer to the way a DBMS organizes information internally. The internal organization can affect how quickly and flexibly you can extract information.

Stands for "Database Management System." In short, a DBMS is a database program. Technically speaking, it is a software system that uses a standard method of cataloging, retrieving, and running queries on data. The DBMS manages incoming data, organizes it, and provides ways for the data to be modified or extracted by users or other programs.

Some DBMS examples include MySQL, Microsoft SQL Server, Oracle, Clipper, and FoxPro. Since there are so many database management systems available, it is important for there to be a way for them to communicate with each other. For this reason, most database software comes with an Open Database Connectivity (ODBC) driver that allows the database to integrate with other databases. For example, common SQL statements such as SELECT and INSERT are translated from a program's proprietary syntax into a syntax other databases can understand.

In case of working with MySQL and Oracle technologies, both of them are great solutions. The point is to find the right environment to leverage what they do well for the best price/performance/scalability you are looking for.

- MySQL is relatively light-weight, can be extremely fast when applications leverage architecture. Lots of features and free to use.
- Oracle offers lots of feature/functionality for solving complex problems. Supports large OLTP environments as well as VLDBs.

Here are some of the feature comparisons from a DBA perspective.

MySQL is powerful enough for most things. If looking for data storage, then MySQL will do the job. If one is going to be doing a lot of clustering, replication, stored procedures, triggers, and integration with enterprise software such as PeopleSoft, then Oracle may be a better solution. [7]

It depends on how the database will be utilized.

### **When is MySQL the Better Choice?**

Many content management systems use MySQL databases. Whenever the plan to use one of these CMS choices, you must use MySQL. Often, these projects are on a smaller scale and MySQL provides the better choice. It's free to use, provides plenty of speed and is very user-friendly.

Startup companies and smaller companies fit best with MySQL. This can also include blogs and those planning to build many niche sites.

Licensing: Open Source.

### **When is Oracle the Better Choice?**

Oracle is a very popular choice with the Fortune 100 list of companies and for larger enterprises. They excel most with larger business applications and large data warehouses. Those looking for the largest amount of features, will choose Oracle. It's packed with features and minimizes the need for third party software.

Licensing: Proprietary

## **2.4 SMS/USSD-based transactional payments**

USSD refers to Unstructured Supplementary Service Data. In the predominant model for SMS payments, the consumer sends a payment request via an SMS text message or an USSD to a short code and a premium charge is applied to their phone bill or their online wallet. The merchant involved is informed of the payment success and can then release the paid for goods.

Earlier this century, the mobile phone became the first communications technology to have more users in developing countries like Bangladesh. With the combination of two most recent technological advancements – internet and mobile phone, a new service (mobile data service) is thus enabled and the first such wireless internet commercial transaction is performed by the banking industry. Banking through mobile phone has been common in developed countries for

years. The real potential of “m-banking” may be to make basic financial services more accessible to millions of poor people. An appropriate banking environment is considered a key pillar as well as an enabler of economic growth. With the continuously emerging wave of information driven economy, the banking industry in Bangladesh has inevitably found itself unable to resist technological indulgence. The need for convenient ways of accessing financial resources beyond the conventional norms has seen the recurrent expansion and modernization of banking patterns. And given the huge demand for finance oriented services, institutions beside the historical banks have joined the fray in an attempt to grab a piece of the perceived cake of opportunity within the banking industry.

Mobile Banking [3] is a process of no branch banking which provides financial services to unbanked communities in both urban and rural area at affordable cost. The aim of the service is not to destroy branch banking but to bring those people under the umbrella of banking service that are away from banking facilities. Government thinks it has a great prospect as it is a new technology in digital Bangladesh. Through M-banking one can avail various services i.e.; utility bill payment, Fund Transfer, Shopping, Cash Withdrawn from selected ATM or Cash point and many more exciting facilities. It is cheap both for the banks and the customers. The bank will be able to lower down the overhead costs and make more profit out of it. The customers will be able to save time as well as money for their transaction needs.

### **2.4.1 Strengths**

Mobile Banking is new in our economy. Only a few banks are now offering this service. Through Internet Banking Most of the banks are offering only balance information. Actual fund transfer and fund disbursement is not possible in all the banks that are offering internet banking services. So this product will enjoy the benefit of a first mover. It is cheap both for the banks and the customers. The bank will be able to lower down the overhead costs and make more profit out of it. The customers will be able to save time as well as money for their transaction needs.

Mobile banking, a real time on-line banking service is available anytime, anywhere throughout the country. So it can save one's time. People have not to wait by standing in a long line so it



takes less time than traditional banking. One can trust mobile banking as traditional banking system. It has secured PIN (Personal Identification Number) code which is known by the user, and also has a check digit

### **2.4.2 Weaknesses**

The system will have a problem with the identification of the individual who is initiating the transaction. In Bangladesh, the identification of an individual is not yet supported digitally. So there may be a problem in moving toward mobile banking era.

### **2.4.3 Challenges**

People have concern about security and privacy. They like to feel their money with their hand. They actually don't believe in virtual money transfer. Furthermore a mobile handset may not easily be operated to handle banking transactions as most of our people do not have sound technological knowledge. In the field of IT new technology is coming every day. One which is very popular today might get obsolete tomorrow. So, to have a competitive edge over the competitors the banks must always update their services. But the challenge is to create awareness and make it believable to potential clients.

### **2.4.4 Current Situation**

Recently Dutch Bangla Bank Ltd. (DBBL) and Trust Bank Ltd. (TBL) introduced mobile banking. On the other side Brac Bank Ltd. introduced another mobile banking product named bkaash. But till now no interbank transaction guideline or regulation policy established among those. So, it's not possible to avail the m-banking facilities of bkaash through DBBL or TBL. As no supportive m-transaction policy is available Mbanking has a positive impact on transfers, payments, deposits and withdrawals in financial transactions. It is cost effective, reliable, speedy and simple way of conducting business and reduces the instances of human error that is characteristic during human interaction in traditional banking.

## 2.5 Transportation of data

Transmission of data between server programs may be achieved by Extensible Markup Language (XML), JavaScript object notation (JSON).

### 2.5.1 Extensible Markup Language (XML)

XML is a markup language that defines a set of rules for encoding documents in a format which is both human-readable and machine-readable. It is defined by the W3C's XML 1.0 Specification and by several other related specifications,<sup>[3]</sup> all of which are free open standards. The design goals of XML emphasize simplicity, generality and usability across the Internet. It is a textual data format with strong support via Unicode for different human languages. Although the design of XML focuses on documents, it is widely used for the representation of arbitrary data structures. Such as those used in web services.

Several schema systems exist to aid in the definition of XML-based languages, while many application programming interfaces (APIs) have been developed to aid the processing of XML data.

XML was designed to describe data.

- XML stands for Extensible Markup Language
- XML is a markup language much like HTML
- XML was designed to describe data, not to display data
- XML tags are not predefined. You must define your own tags
- XML is designed to be self-descriptive
- XML is a W3C Recommendation

XML is not a replacement for HTML. XML and HTML were designed with different goals:

- XML was designed to describe data, with focus on what data is
- HTML was designed to display data, with focus on how data looks

HTML is about displaying information, while XML is about carrying information. Maybe it is a little hard to understand, but XML does not DO anything. The following example is a note to Tove, from Jani, stored as XML:

```
<note>
  <to>Tove</to>
  <from>Jani</from>
  <heading>Reminder</heading>
  <body>Don't forget me this weekend!</body>
</note>
```

It has sender and receiver information, it also has a heading and a message body. But still, this XML document does not DO anything. It is just information wrapped in tags. Someone must write a piece of software to send, receive or display it. The tags in the example above (like <to> and <from>) are not defined in any XML standard. These tags are "invented" by the author of the XML document. That is because the XML language has no predefined tags. The tags used in HTML are predefined. HTML documents can only use tags defined in the HTML standard (like <p>, <h1>, etc.).

XML allows the author to define his/her own tags and his/her own document structure. XML is a complement to HTML. It is important to understand that XML is not a replacement for HTML. In most web applications, XML is used to describe data, while HTML is used to format and display the data. So, XML stands for Extensible Markup Language. XML was designed to describe data. XML is a software- and hardware-independent tool for carrying information. XML is easy to learn. XML is a software- and hardware-independent tool for carrying information.

## 2.6 Application Programming Interface

An application-programming interface (API) is a set of programming instructions and standards for accessing a Web-based software application or Web tool. A software company releases its API to the public so that other software developers can design products that are powered by its service.

For example, Amazon.com released its API so that Web site developers could more easily access Amazon's product information. Using the Amazon API, a third party Web site can post direct links to Amazon products with updated prices and an option to "buy now."

An API is a software-to-software interface, not a user interface. With APIs, applications talk to each other without any user knowledge or intervention. When you buy movie tickets online and enter your credit card information, the movie ticket Web site uses an API to send your credit card information to a remote application that verifies whether your information is correct. Once payment is confirmed, the remote application sends a response back to the movie ticket Web site saying it's OK to issue the tickets. The API itself is a chunk of software code written as a series of XML messages. There are several advantages of leveraging a conferencing API for developers, businesses and the software providers themselves:

- Developers can speed up the application development process by easily integrating remote tools and systems.
- Companies don't have to pay for several different software applications and the hardware to make them all work.
- By integrating conferencing functionality into existing applications, companies don't have to train IT staff and employees on how to administer and use new software.
- The company who releases the API allows its customers to access their conferencing services in new, more efficient ways, increasing brand recognition and customer loyalty.

REST (Representational State Transfer) is a simple stateless architecture that generally runs over HTTP. REST involves reading a designated Web page that contains an XML file. The XML file describes and includes the desired content.

REST is often used in mobile applications, social networking Web sites, mash up tools and automated business processes. The REST style emphasizes that interactions between clients and services is enhanced by having a limited number of operations (verbs). Flexibility is provided by assigning resources (nouns) their own unique universal resource indicators (URIs). Because each verb has a specific meaning (GET, POST, PUT and DELETE), REST avoids ambiguity.

## 2.6.1 SMS gateway REST API

An SMS gateway allows a computer to send or receive Short Message Service (SMS) transmissions to or from a telecommunications network. Most messages are eventually routed into the mobile phone networks. Many SMS gateways support media conversion from email and other formats. This application turns your android phone into powerful SMS gateway. Features provided:

1. Periodically checks for e-mails via POP3 and send them via SMS.
2. Send SMS by HTTP GET request on its internal webserver.
3. Forward received SMS to desired e-mail via SMTP.
4. Forward received SMS calling remote webserver via HTTP GET.

Android's source code is released by Google under open source licenses, although most Android devices ultimately ship with a combination of open source and proprietary software. Initially developed by Android, Inc. Android is popular with technology companies which require a ready-made, low-cost and customizable operating system for high-tech devices. Android's open nature has encouraged a large community of developers and enthusiasts to use the open-source code as a foundation for community-driven projects, which add new features for advanced users or bring Android to devices which were officially released running other operating systems.

Java programming language using the Android software development kit (SDK). The SDK includes a comprehensive set of development tools including a debugger, software libraries, a handset emulator, documentation, sample code, and tutorials. The officially supported integrated development environment (IDE) is Eclipse using the Android Development Tools (ADT) plugin.

Android has a growing selection of third-party applications, which can be acquired by users by downloading and installing the application's APK file, or by downloading them using an application store program that allows users to install, update, and remove applications from their devices. Google Play Store is the primary application store installed on Android devices that comply with Google's compatibility requirements and license the Google Mobile Services

software. Google Play Store allows users to browse, download and update applications published by Google and third-party developers.

Android is an open-source software stack for a wide array of mobile devices with different form factors. Android's purpose is to establish an open platform for developers to build innovative apps. The Android Compatibility program defines the technical details of the Android platform and provides tools for device manufacturers to ensure developers' apps run on a variety of devices. Thus Android platform is preferred in the proposed system.

## **Chapter 3**

### **Methodology**

Automation of Course Registration System (ACRS) is an application that would let the registration system to be led automatically. The system proposed is a web based system.

#### **3.1 System Modules**

The system is modularized in three modules: Student Module, System server Module and Payment Module. Administration module provides Registration and Payment regarded Information to Student module and Payment module. Student logs in into the system. Student is provided an application form with fields for his/her name, father's name, mother's name, level and term he/she wants to register in. He/she has to choice either regular term or short term to register. If he/she picks regular term he/she would be asked to provide level and term he/she wants to register in. Otherwise if he/she picks short term he/she would be provided with a form where he/she can enter the subjects he/she wants to attend. Then necessary payments to be done to the hall office and registration office by the particular student is provided by the system server called payment information.

A successful payment is that which is notified to Payment module. Payment module verifies the payment. If payment is valid system server module stores the payment record. Student completing a successful payment receives a transaction Id. Then he/she has to enter the transaction Id to the server. Server compares the amount and checks the payment receipt. If everything is correct then the server validates the student and validation passed to Student module. Student receiving payment receipt is eligible to download the forms named 'Registration form'. Students receives the forms including necessary information along. Following figure shows the modules and interactions between the modules.

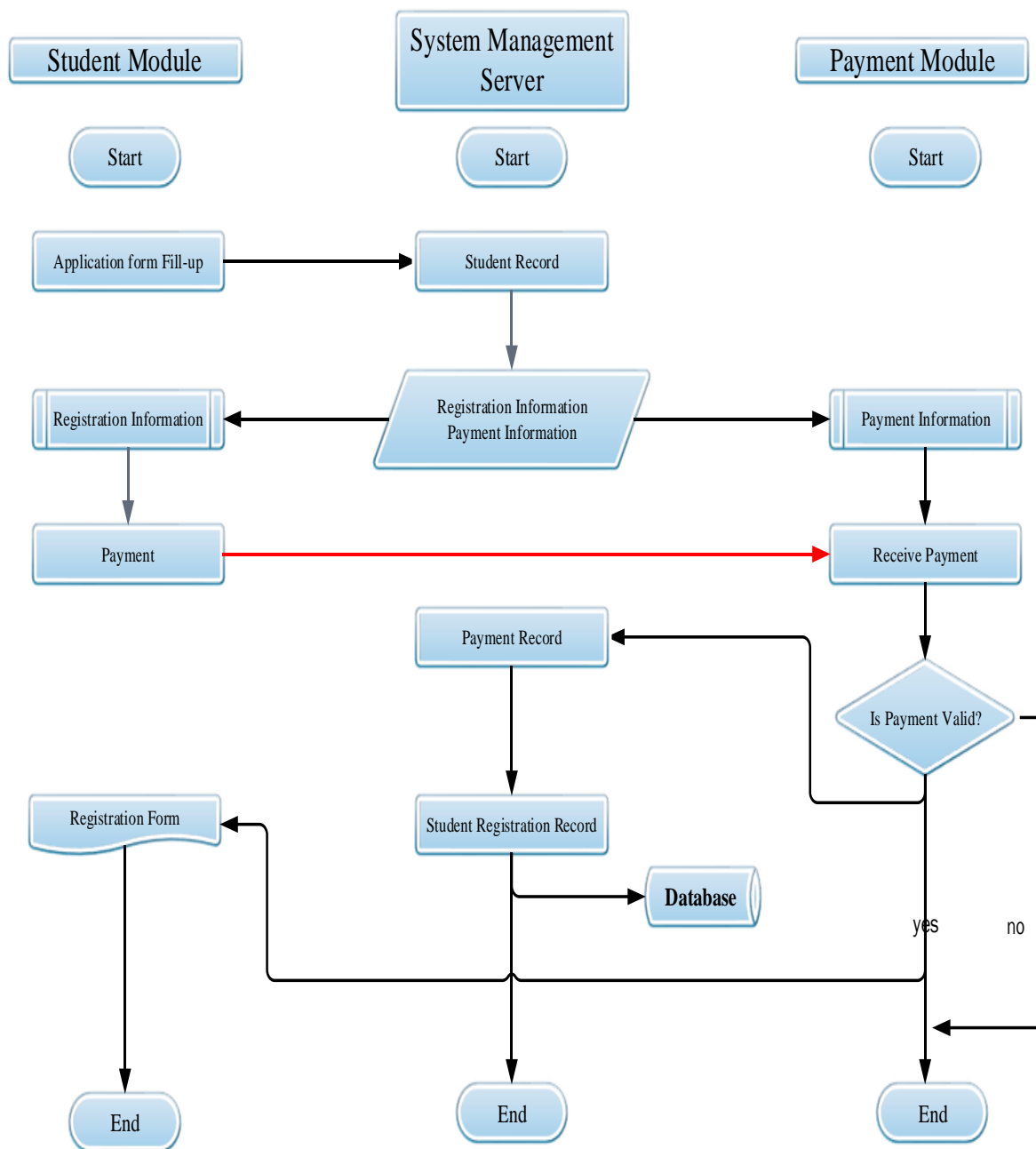


Figure 4.1: System modules of ACRS



**Application form:** Student logged in the system is provided an application form fields for his/her name, father's name, mother's name, level and term he/she wants to register in. He/she has to choice either regular term or short term to register. If he/she picks regular term he/she would be asked to provide level and term he/she wants to register in. Otherwise if he/she picks short term he/she would be provided with a form where he/she can enter the subjects he/she wants to attend. Information provided is stored as the **student record**.

**Payment information:** Then necessary payments to be done to the hall office and registration office by the particular student is provided by the system server called payment information.

**Payment record:** A successful payment is that which is notified to Payment module. Payment module verifies the payment. If payment is valid system server module stores the student id, transaction id and amount received as **student registration record**.

**Registration form:** Student completing a successful payment receives a transaction Id. Then he/she has to enter the transaction Id to the server. Server compares the amount and checks the payment receipt. If everything is correct then the server validates the student and validation passed to Student module. Student receiving payment receipt is eligible to download the forms named 'Registration form'.

## 3.2 Design of modules

JavaScript is one of the most versatile and effective languages used to extend functionality in websites. JavaScript is executed on the client side which means that the code is executed on the user's processor instead of the web server thus saving bandwidth and strain on the web server. JavaScript is a relatively easy language and relatively fast to the end user and has extended functionality to web pages.

PHP is one of the most popular server side scripting languages running today. It is used for creating dynamic webpages that interact with the user offering customized information. PHP offers many advantages; it is fast, stable, secure, easy to use and open source .The scripting language PHP and JavaScript would be used for simplicity and functionality.

As database management system (DBMS) MySQL is chosen. The relational database management system (RDBMS) officially called Oracle Database (and commonly referred to as Oracle RDBMS or simply as Oracle) has become a major presence in database computing

which is highly enriched. MySQL is characterized as a fast, robust database with a good feature that's designed to be portable it's a pretty good choice. All in all, a pretty good choice for many "middle of the road" requirements. MySQL is a free, fast, reliable open source relational database. It. At certain times there will be a trade-off between speed and capabilities, and the MySQL team intend to keep their database engine fast and reliable. The criteria where Oracle rules over MySQL is 'big data challenge'. But in this case this would not be a matter of consideration. Thus MySQL is preferred.

The web server would keep track of records of the students regarding their hall fees, due fine and registration fees. When a student enters the site using any computer or mobile phone has to enter his/her ID and the semester he/she wants to be registered in if he/she wants to register himself/herself in regular term. If he/she chose to register for short term he/she has to enter the number of subjects, subject codes. Then the server would automatically acknowledge him of his registration status, registration and hall fees and fines due. It will also provide the account number where he/she has to pay for each individual accounts. Now students are facilitated with the option of using mobile banking systems available and pay through his own mobile phone account (if he/she has any) or from any booths nearby his/her current location. Thus the system offers the student the independency from being physically present. Mobile banking systems are now available in our country and safety is ensured also. bKash, Dutch Bangla mobile banking, Ok banking, MobiCash are the leading companies and most of the students has their personal accounts there. Even if the student have no personal account he would be easily able to complete transaction.

Each accounts of the system would be associated with distinct subscriber identity modules (SIM). Now the common way of establishing connection between these SIMs and application server is attaching a SIM Port machine with the server pc. To program this SIM port AT style command is used. Almost all of the Hayes modem commands start with the two letter sequence AT - for getting the modem's attention. Because of this, modem commands are often called AT Commands. This still holds for many of the manufacturer specific command set extensions. Most of them also start with AT, and are called AT Commands, too. The exact usage of the term AT command set slightly varies from manufacturer to manufacturer. In general, it can be assumed that a modem with an AT command set uses commands mostly starting with AT, uses

the original Hayes way of separating data and commands and supports the original Hayes commands and register settings as a subset. This traditional system costs a lot. SIM port (64 SIM) costs around \$1750. Furthermore deep knowledge in AT command set is also required. Thus the system proposes a cost efficient, easy to use, portable alternative.

The payment verification is facilitated by Android OS based SMS gateway server on API approach. The android application would be installed in the android phones which are part of the Automation of Course Registration System (ACRS). These phones will have the specific subscriber identity modules (SIM) which holds the accounts for each distinct departments. The android application will forward the notifications of payment through Short message service (SMS) which will be forwarded to the system server.

System server will receive the XML data forwarded by the SMS gateway android application. Server will explode the XML data and convert it into a SQL query. The query will enter the phone number, sms center and text in UTF-8 format into the 'bKash' table in database.

UTF-8 (Universal Character Set Transformation Format—8-bit) is a character encoding capable of encoding all possible characters (called code points) in Unicode. The encoding is variable-length and uses 8-bit code units. It was designed for backward compatibility with ASCII and to avoid the complications and byte order marks in UTF-16 and UTF-32.

Then the server explodes the text message and cut in into pieces to process data form it. The ACRS server process out the reference ID, transaction ID and amount from the text. Student is asked to enter his/her student id as reference id while payment. Then student is asked to enter the transaction ID he receives by the sms and feed it to the system. System then matches the transaction id given by the student with the traction ids explode. If they matches then system checks for matching of student Id and corresponding amount. If everything matches correctly then the student is considered as an eligible candidate and he/she is then given the permission to download the registration form including the information given by themselves.

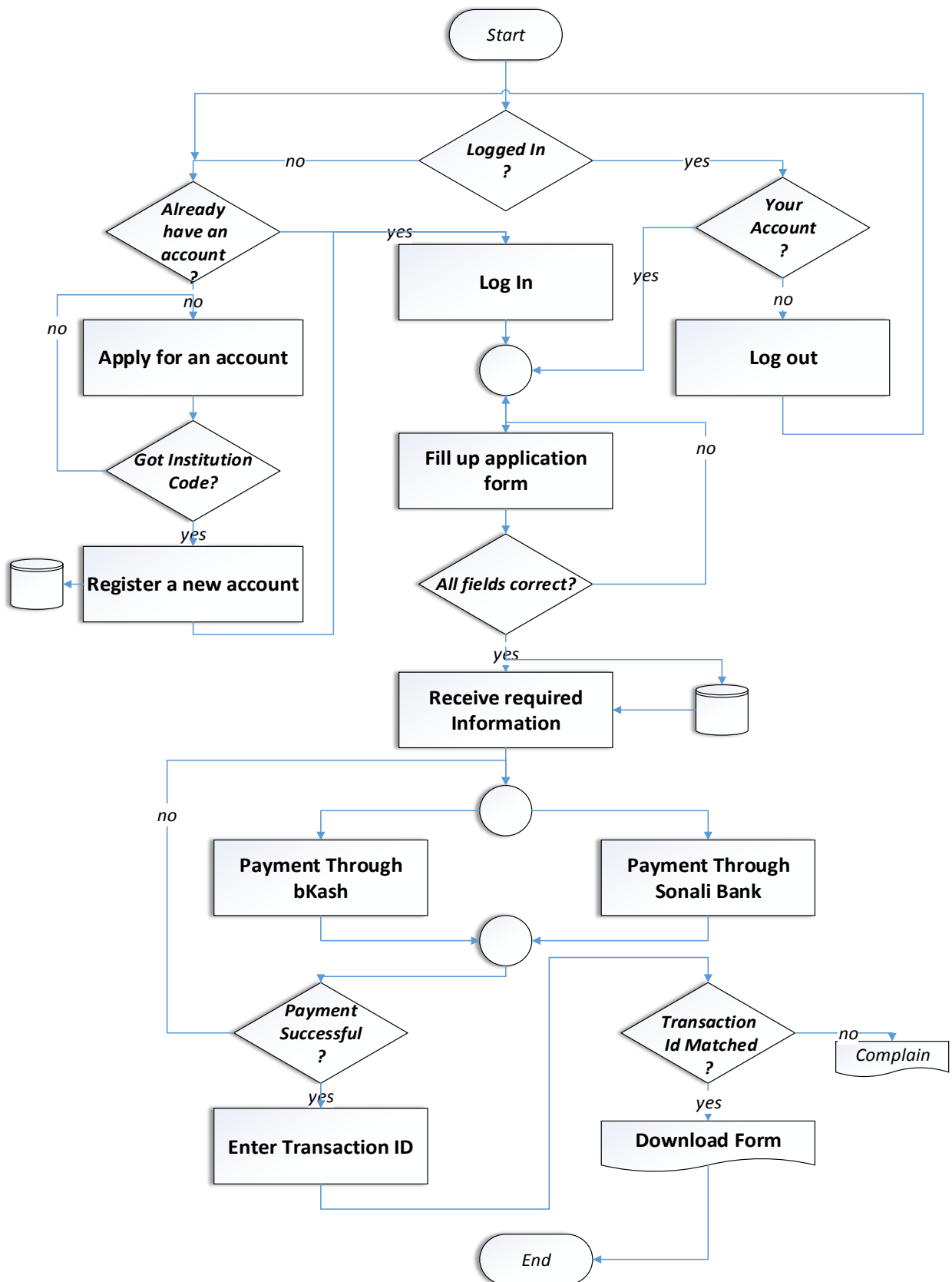


Figure 3.2: Flowchart of a registration via ACRS

## **3.3 Software Development Tools**

Software developments tools used to develop the Automation of Course Registration System (ACRS):

### **3.3.1 Construction of User Interface (UI):**

Success of a software depends hugely in the user friendly User Interface. The user interface is the space where interactions between humans and machines occur. Generally, the goal of human-machine interaction engineering is to produce a user interface which makes it easy, self-explanatory, efficient, and enjoyable (user friendly) to operate a machine in the way which produces the desired result.

To acquire a user friendly interface HTML along with CSS is used by us.

HTML is a markup language for describing web documents (web pages). HTML stands for Hyper Text Markup Language. A markup language is a set of markup tags. HTML documents are described by HTML tags. Each HTML tag describes different document content.

Cascading Style Sheets (CSS) is a style sheet language used for describing the look and formatting of a document written in a markup language. While most often used to change the style of web pages and user interfaces written in HTML and XHTML, the language can be applied to any kind of XML document, including plain XML, SVG and XUL. Along with HTML and JavaScript, CSS is a cornerstone technology used by most websites to create visually engaging webpages, user interfaces for web applications, and user interfaces for many mobile applications

### **3.3.2 Client Side Programming**

Client Side Scripting refers to the program running on the browser. Client side scripting or front end programming reduces transmission of data and workload of server. For client side programming JavaScript is used to develop ACRS. The most common use of JavaScript is to add client-side behavior to HTML pages, a.k.a. Dynamic HTML (DHTML).

Scripts are embedded in or included from HTML pages and interact with the Document Object Model (DOM) of the page.

Some simple examples of this usage are:

- Loading new page content or submitting data to the server via AJAX without reloading the page (for example, a social network might allow the user to post status updates without leaving the page)
- Animation of page elements, fading them in and out, resizing them, moving them, etc.
- Interactive content, for example games, and playing audio and video
- Validating input values of a web form to make sure that they are acceptable before being submitted to the server.
- Transmitting information about the user's reading habits and browsing activities to various websites. Web pages frequently do this for web analytics, ad tracking, personalization or other purposes.

Because JavaScript code can run locally in a user's browser (rather than on a remote server), the browser can respond to user actions quickly, making an application more responsive. Furthermore, JavaScript code can detect user actions which HTML alone cannot, such as individual keystrokes.

### **3.3.3 Server Side Programming**

Server side Scripting or back end programming is done to achieve functional tasks of a web based system. For server side scripting PHP is used. PHP is an acronym for "PHP: Hypertext Preprocessor". PHP is a widely-used, open source scripting language PHP scripts are executed on the server. The reason behind PHP to be chosen:

- Runs efficiently on the server side
- Runs on various platforms (Windows, Linux, Unix, Mac OS X, etc.)
- Compatible with almost all servers used today (Apache, IIS, etc.)
- Supports a wide range of databases
- Able to create, open, read, write, delete, and close files on the server

- Can collect form data and add, delete, modify data in your database
- Used to control user-access can encrypt data

With PHP you are not limited to output HTML. You can output images, PDF files, and even flash movies. You can also output any text, such as XHTML and XML.

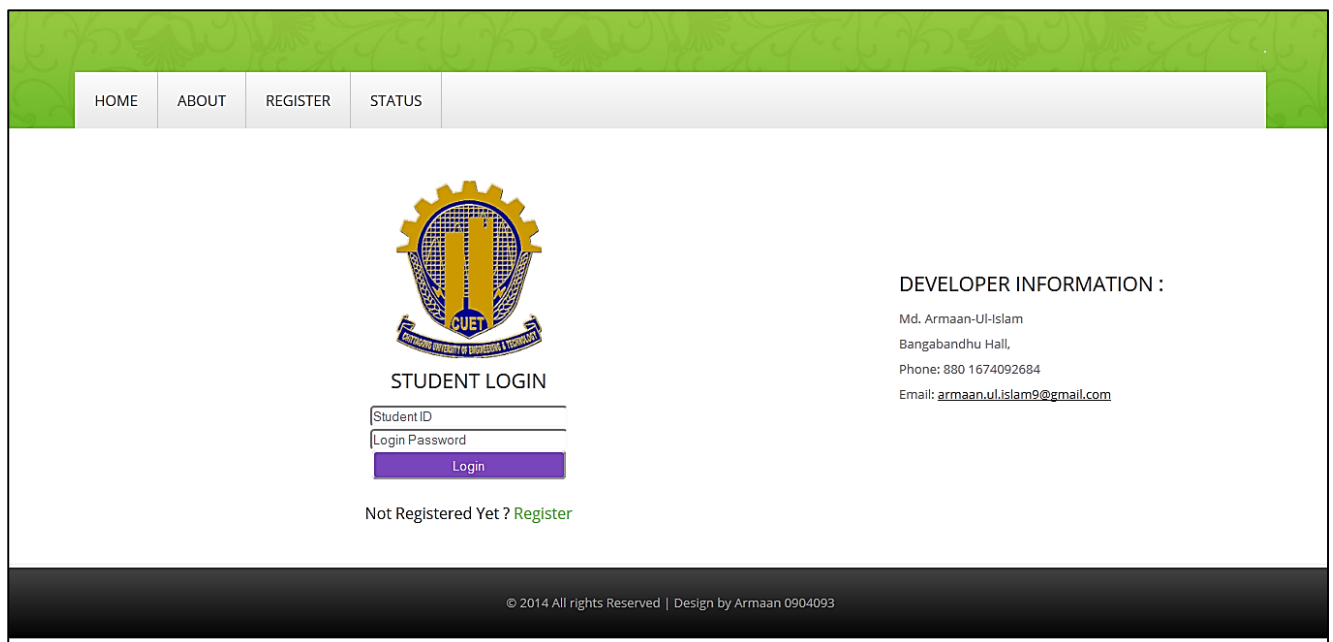
# Chapter 4

## Implementation

### 4.1 User Interface

To acquire a user friendly interface HTML along with CSS is used by us. HTML is a markup language for describing web documents (web pages). HTML stands for Hyper Text Markup Language. A markup language is a set of markup tags. HTML documents are described by HTML tags. Each HTML tag describes different document content.

Cascading Style Sheets (CSS) is a style sheet language used for describing the look and formatting of a document written in a markup language. While most often used to change the style of web pages and user interfaces written in HTML. The User Interface of ACRS. The tabs available are Home, About, Register, Status.



The screenshot displays the user interface of the ACRS system. At the top, there is a green navigation bar with four tabs: HOME, ABOUT, REGISTER, and STATUS. Below the navigation bar, the main content area is white. On the left side, there is a logo of Chittagong University of Engineering & Technology (CUET) featuring a gear and a book. Below the logo, the text "STUDENT LOGIN" is displayed. Underneath, there are two input fields: "Student ID" and "Login Password". A purple "Login" button is positioned below the password field. Below the login button, there is a link that says "Not Registered Yet ? Register". On the right side of the main content area, there is a section titled "DEVELOPER INFORMATION :". Below this title, the following information is listed: "Md. Armaan-Ul-Islam", "Bangabandhu Hall,", "Phone: 880 1674092684", and "Email: armaan.ul.islam9@gmail.com". At the bottom of the page, there is a black footer bar with the text "© 2014 All rights Reserved | Design by Armaan 0904093".

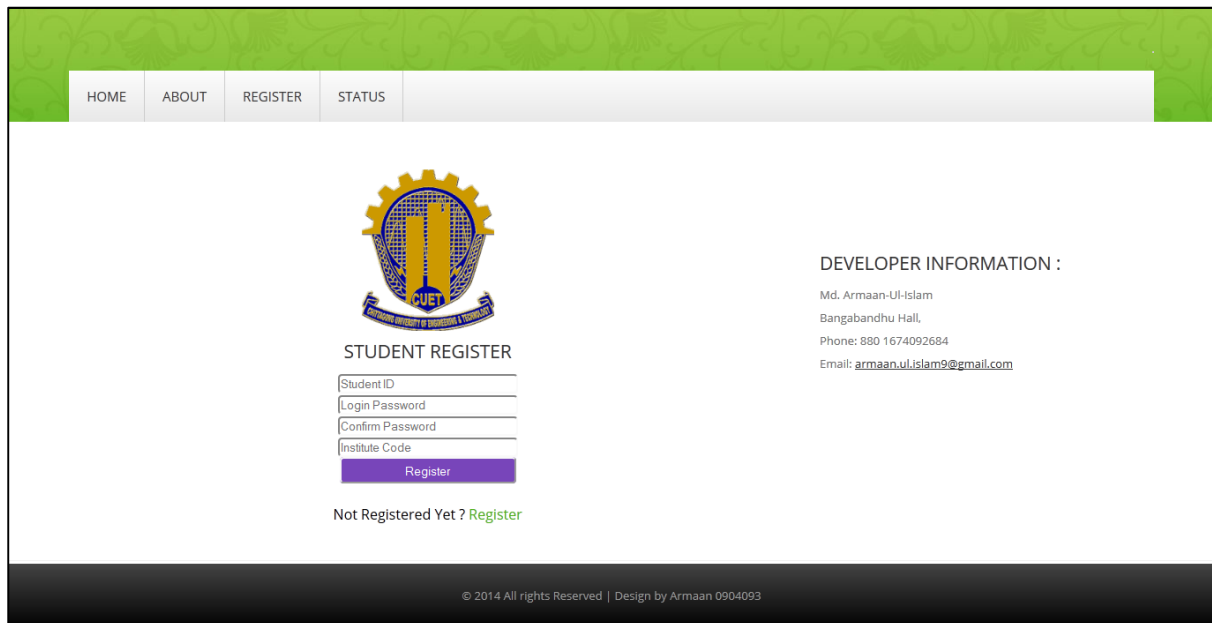
Figure 4.1: User Interface

ABOUT tab holds information about Chittagong University of Engineering & Technology. User cannot gain access to REGISTER, STATUS until he/she is logged in.



## 4.2 Register an account

To begin with a new user has to register an account. To register a new account user have to click the link: “[Not Registered Yet? Register](#)”. User will be directed to the account registration page.



The screenshot shows a web page for student registration. At the top, there is a green header with a navigation menu containing links for HOME, ABOUT, REGISTER, and STATUS. The main content area features the logo of the Bangladesh University of Textiles (BUT) on the left. To the right of the logo is the text 'STUDENT REGISTER' followed by four input fields: 'Student ID', 'Login Password', 'Confirm Password', and 'Institute Code'. Below these fields is a purple 'Register' button. To the right of the registration form, there is a section titled 'DEVELOPER INFORMATION :' which lists the developer's name (Md. Armaan-Ul-Islam), location (Bangabandhu Hall), phone number (880 1674092684), and email address (armaan.ul.islam9@gmail.com). At the bottom of the page, there is a footer with the text '© 2014 All rights Reserved | Design by Armaan 0904093'. Below the registration form, there is a link that says 'Not Registered Yet ? [Register](#)'.

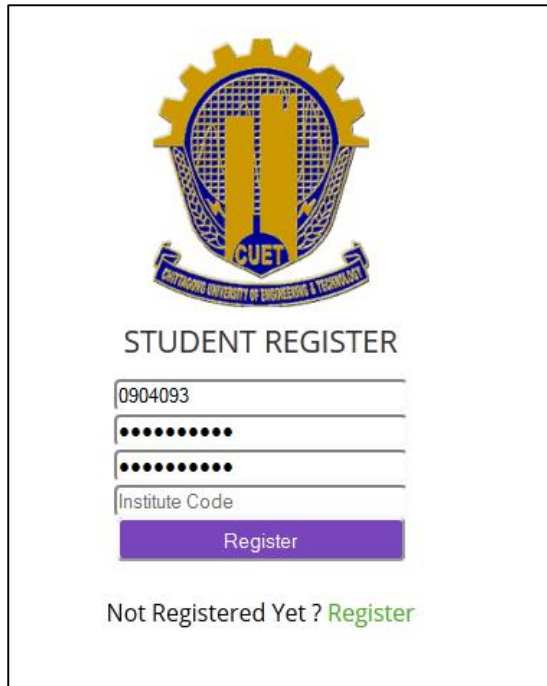
Figure 4.2: Account page

Then User has to enter a valid ‘User Id’. And enter a password for login and confirm the same password again. Password are written in unrecognizable format to be seen. Then the student has to enter the ‘Institution Code’.

Institution Code would be a code provided by the Institution to the students who wants to open an account. This measure is taken to prevent account registration from out of the institution to maintain the privacy of information among the members of the Institution.

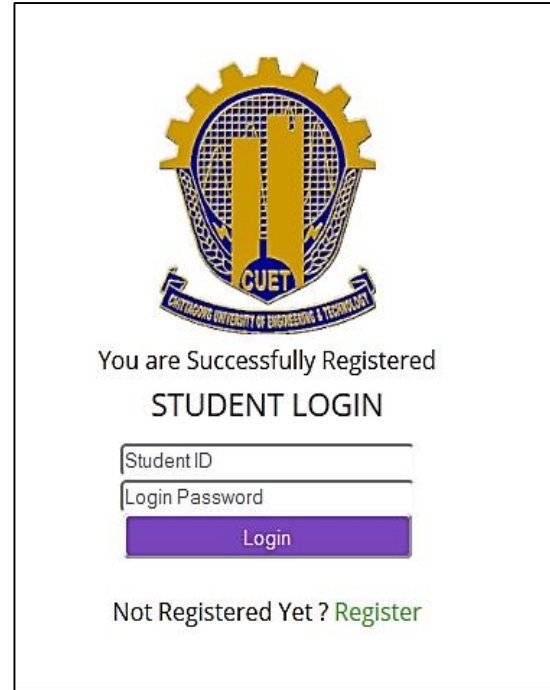
The User has to collect the ‘Institution Code’ in order to register an account. Once he has entered valid information in the fields, his/her initial and conformational password matches. And he/she enters the correct ‘Institution Code’, the account will be registered

“You are Successfully Registered” message would be shown after a successful registration and user would be redirected to home page. Otherwise user would see the message “please fill all the fields and check password/Inscore.”



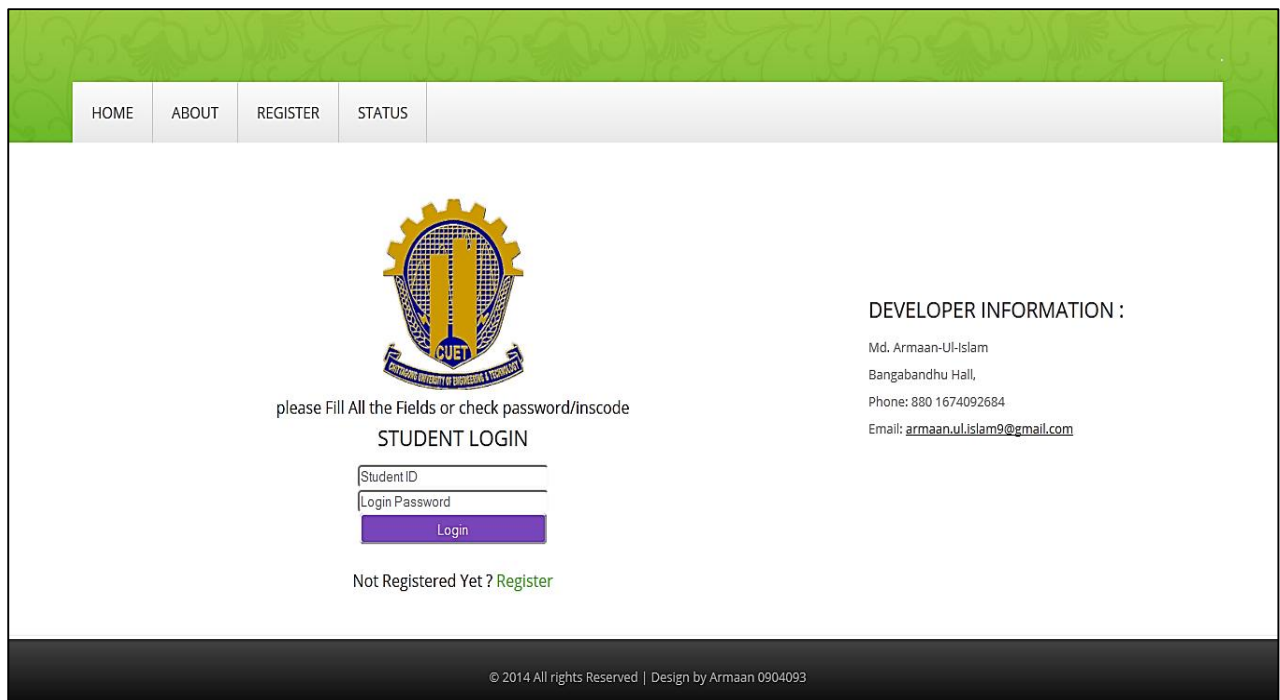
The image shows a web form titled "STUDENT REGISTER" for Chittagong University of Engineering & Technology (CUET). At the top is the CUET logo. Below it, the form has four input fields: a text field containing "0904093", two password fields represented by dots, and a text field labeled "Institute Code". A purple "Register" button is at the bottom. Below the button, it says "Not Registered Yet ? [Register](#)".

Figure 4.3: Registering an account



The image shows a web form titled "STUDENT LOGIN" for Chittagong University of Engineering & Technology (CUET). At the top is the CUET logo. Below it, the text "You are Successfully Registered" is displayed. The form has two input fields: "Student ID" and "Login Password". A purple "Login" button is at the bottom. Below the button, it says "Not Registered Yet ? [Register](#)".

Figure 4.4: Successful Registration



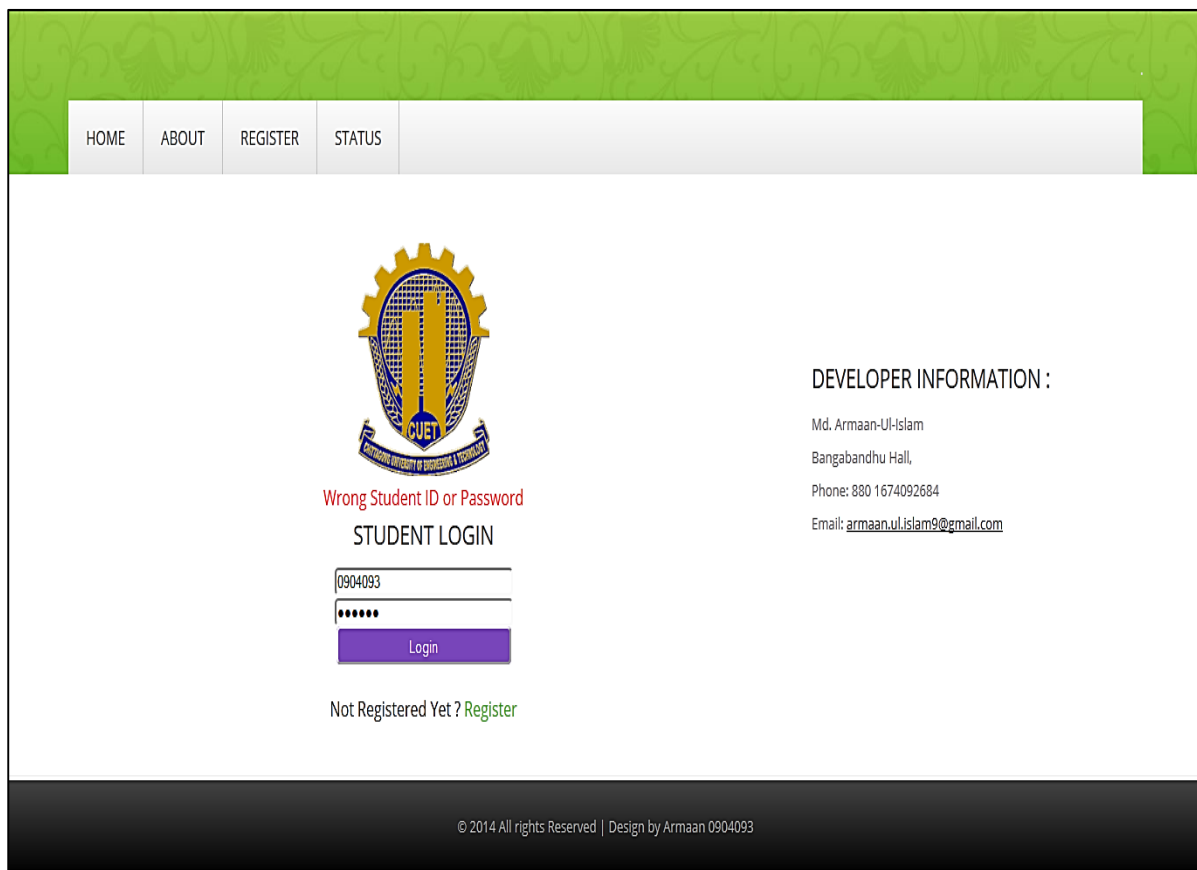
The image shows a web page with a green header containing navigation links: HOME, ABOUT, REGISTER, and STATUS. The main content area has the CUET logo at the top. Below the logo, the text "please Fill All the Fields or check password/inscode" is displayed. Underneath is a "STUDENT LOGIN" form with "Student ID" and "Login Password" fields and a "Login" button. Below the button, it says "Not Registered Yet ? [Register](#)". To the right of the login form, under the heading "DEVELOPER INFORMATION :", the following details are listed: Md. Armaan-UI-Islam, Bangabandhu Hall, Phone: 880 1674092684, and Email: [armaan.ul.islam9@gmail.com](mailto:armaan.ul.islam9@gmail.com). The footer of the page contains the text "© 2014 All rights Reserved | Design by Armaan 0904093".

Figure 4.5: Unsuccessful account Registration

## 4.3 Log In

After the User has registered a valid account he/she can now log in the system by entering the student ID and password. If the Student Id and password matches will the entry in database he/she would be logged in.

Otherwise, page below would be shown with the message “Wrong Student ID or Password”.



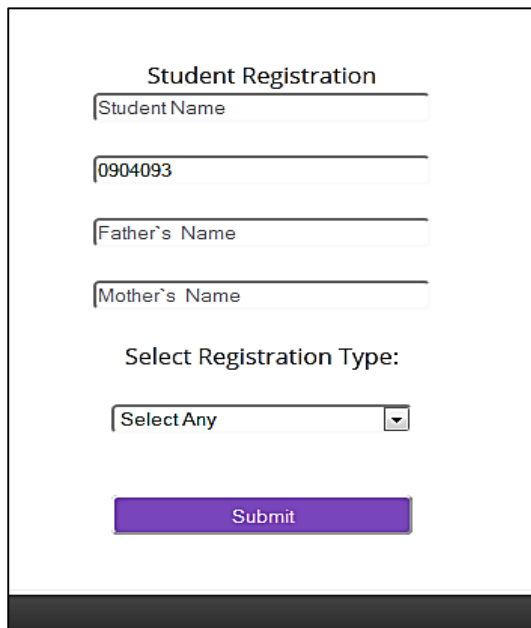
The screenshot displays a web application interface for a student login system. At the top, there is a green header bar with a navigation menu containing links for HOME, ABOUT, REGISTER, and STATUS. Below the header, the main content area is white. On the left side, there is a logo for CUET (Central University of Engineering & Technology) featuring a gear and a book. Below the logo, the text "Wrong Student ID or Password" is displayed in red. Underneath this message, the text "STUDENT LOGIN" is shown in black. There are two input fields: the first one contains the text "0904093" and the second one contains six dots, representing a password field. Below these fields is a purple button labeled "Login". At the bottom of the login section, there is a link that says "Not Registered Yet ? Register" in green. On the right side of the page, under the heading "DEVELOPER INFORMATION :", the following details are listed: "Md. Armaan-Ul-Islam", "Bangabandhu Hall,", "Phone: 880 1674092684", and "Email: [armaan.ul.islam9@gmail.com](mailto:armaan.ul.islam9@gmail.com)". At the very bottom of the page, there is a black footer bar with the text "© 2014 All rights Reserved | Design by Armaan 0904093".

Figure 4.6: Unsuccessful Login

Once the user is logged in, he/she is provided with the log out tab. By accessing the Log Out tab user can sign themselves off so that no other can access to his personal information.

## 4.4 Registration

Once the student has registered an account and logged into the system, he/she is provided the registration application form. The student then has to fill the form up providing necessary information regarding his/her name, student id, father's name, mother's name. After that he/she has to complete the selection of registration type either regular term registration or



Student Registration

Student Name

0904093

Father's Name

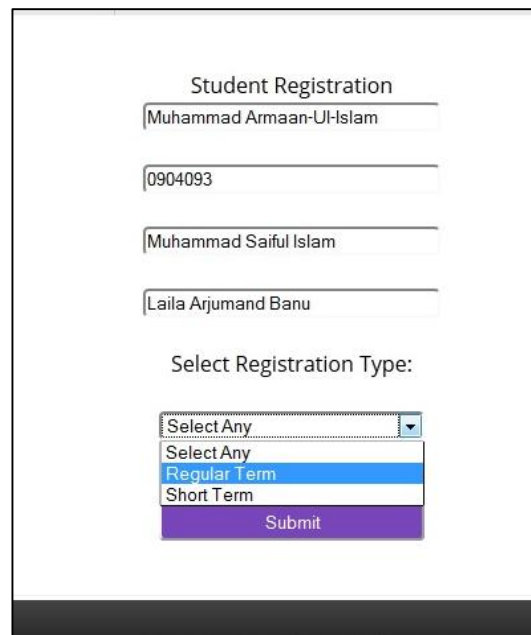
Mother's Name

Select Registration Type:

Select Any

Submit

Figure 4.7: Registration application form



Student Registration

Muhammad Armaan-Ul-Islam

0904093

Muhammad Saiful Islam

Laila Arjumand Banu

Select Registration Type:

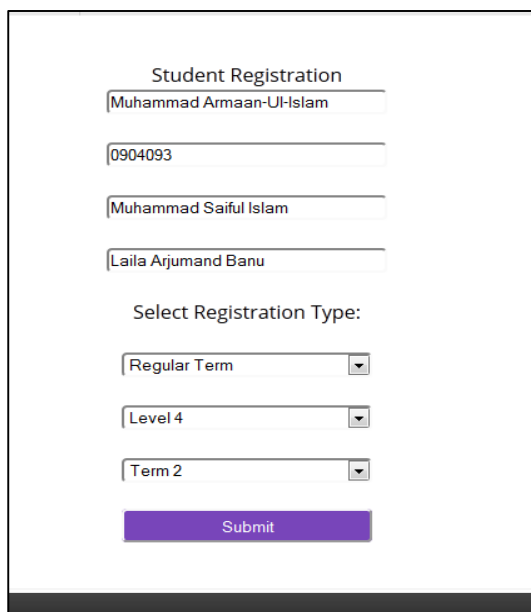
Select Any

Regular Term

Short Term

Submit

Figure 4.8: Registration type selection



Student Registration

Muhammad Armaan-Ul-Islam

0904093

Muhammad Saiful Islam

Laila Arjumand Banu

Select Registration Type:

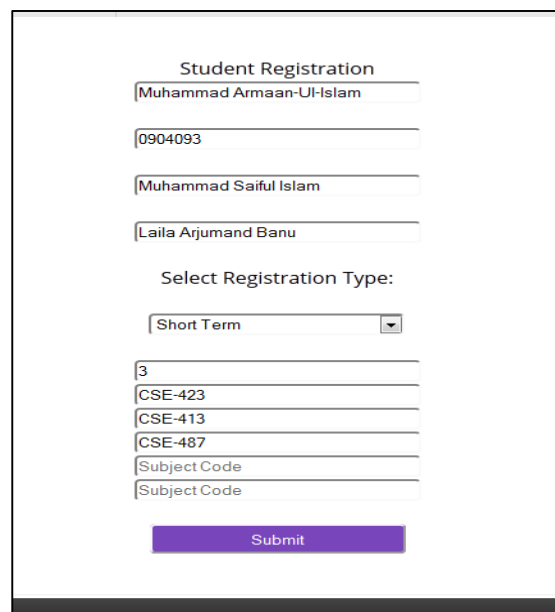
Regular Term

Level 4

Term 2

Submit

Figure 4.9: Regular term registration



Student Registration

Muhammad Armaan-Ul-Islam

0904093

Muhammad Saiful Islam

Laila Arjumand Banu

Select Registration Type:

Short Term

3

CSE-423

CSE-413

CSE-487

Subject Code

Subject Code

Submit

Figure 4.10: Short term registration

short term registration. If the student selects regular term he/she then has to select the level and term he/she wants to register himself/herself in.

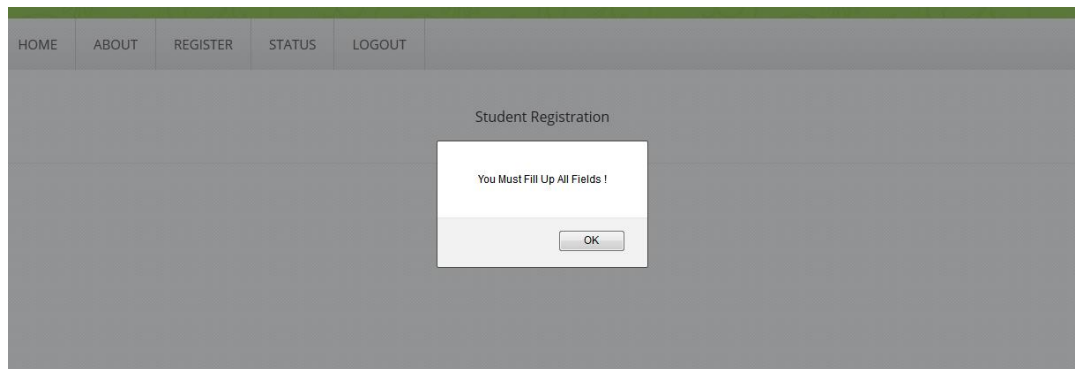


Figure 4.11: Unsuccessful application form fill-up

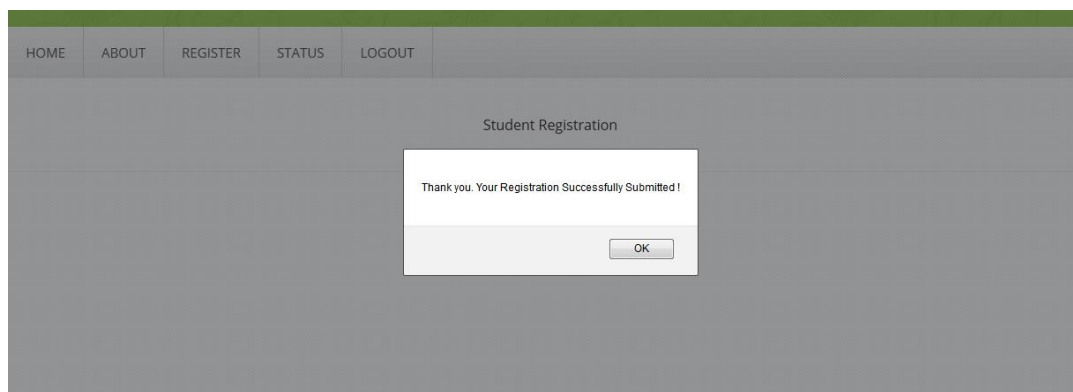


Figure 4.12: Successful application form fill-up

If the student selects short term registration then he/she is provided fields to enter the name of courses he/she wants to register to attend. If the application for registration is accepted the student is notified by pop up menu. Otherwise he/she is notified to fill all the fields correctly. After a successful application he/she is granted access to the payment status.

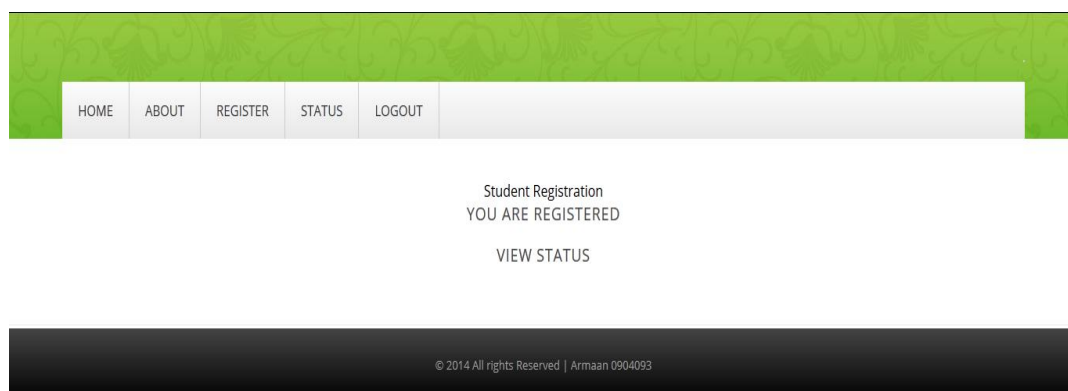


Figure 4.13: Post application page

## 4.5 Payment

The student being in a state where he/she has completed the registration application form successfully is provided with a list of payments due to complete a successful registration. The user is provided information regarding the name accounts, fees to be paid, fine charged. He/she will also be given an option to complete the payment in this stage.

STUDENT ID :0904093  
STUDENTS NAME :MUHAMMAD ARMAAN-UL-ISLAM  
FATHERS NAME :MUHAMMAD SAIFUL ISLAM  
MOTHERS NAME :LAILA ARJUMAND BANU  
LEVEL :4 TERM :2

All_Section	Name	Due_fee	Fine	Total	Payment
Register Office	CSE	1100	100	1200	<a href="#">Complete Payment</a>
Hall Office	BBHALL	700	200	900	<a href="#">Complete Payment</a>

© 2014 All rights Reserved | Design by Armaan 0904093


Figure 4.14: Status page


When the student chose an entry to complete that particular payment he/she is given option to pay trough bKash or Sonali bank manually.

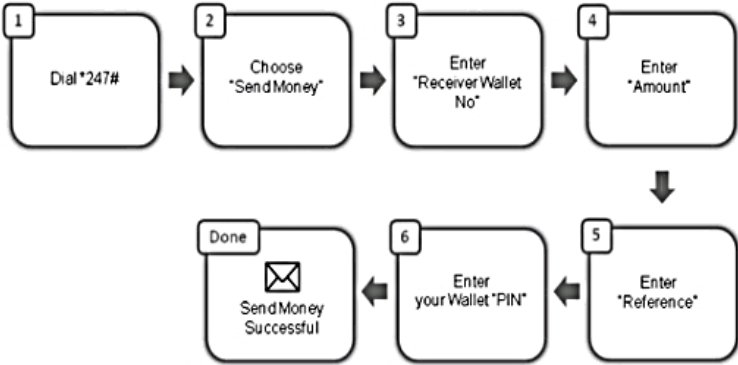
### 4.5.1 Payment through bKash

If the user choses to pay through bKash he/she will be given a particular SIM number regarding to that account and amount is notified. Then the student has to make a real time money transaction via bKash. If the user has his/her own bKash wallet/account in his mobile that would make things easy for him. Otherwise the user have to reach a bKash booth which is quite available in our country. The user will receive necessary instructions by the system in order to complete the payment successfully. The user must enter his/her student ID in the field of reference to get the response back from the system.

HOME	ABOUT	REGISTER	STATUS	LOGOUT
<p>STUDENT ID :0904093</p> <p>STUDENTS NAME :MUHAMMAD ARMAAN-UL-ISLAM</p> <p>FATHERS NAME :MUHAMMAD SAIFUL ISLAM</p> <p>MOTHERS NAME :LAILA ARJUMAND BANU</p> <p>LEVEL :4 TERM :2</p> <p><b>MAKE PAYMENT TO +8801674092684</b></p> <p><b>AMOUNT 1200 BDT</b></p>				



a  **BRAC BANK** company



```

graph TD
    1[1. Dial *247#] --> 2[2. Choose "Send Money"]
    2 --> 3[3. Enter "Receiver Wallet No"]
    3 --> 4[4. Enter "Amount"]
    4 --> 5[5. Enter "Reference"]
    5 --> 6[6. Enter your Wallet "PIN"]
    6 --> Done[Done  
Send Money Successful]
  
```

N.B : 3.Enter the number given as Reciever wallet number.

4.Must enter the amount specified,neither more nor less.

5.Enter your Stduent ID as Reference.

After Successful Tarnsaction a SMS would be recieved.

Enter the Transaction Id (TrxlD) mentioned below.(Example:'1419875104')

Helpline:8801674092684

Figure 4.15: Payment via bKash

The payment that successfully reaches the bKash wallet number provided by the system will pass through Android SMS gateway to the Server. The server will then explode the XML data received and process them into reference Id, amount and transaction id. Then the system will store those processed data into table.

The user accomplishing a successful payment will receive a transaction id. The next thing user has to do is to put that transaction id in the verify transaction field. The system will then match that transaction id with the transaction ids it has in the tables. If transaction id matches then the student id and amount is checked for the corresponding transaction. If all the parameters matches accurately then the student is considered eligible.

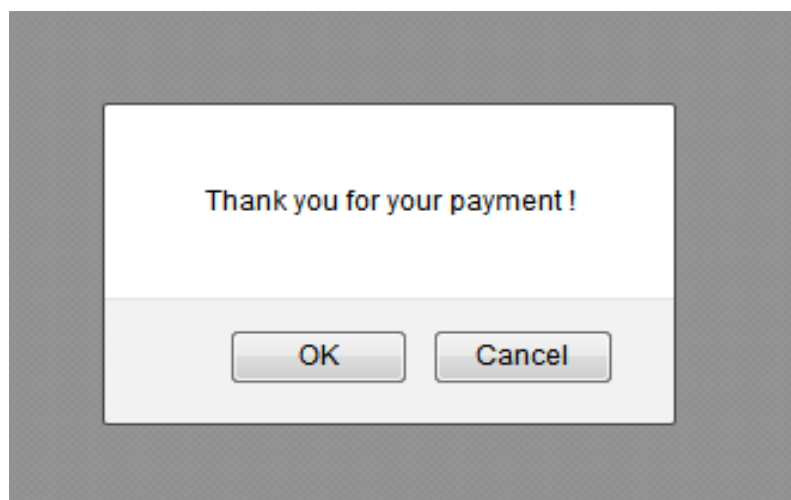


Figure 4.16: Successful payment

When the particular transaction entered exists in the list but the amount is less than the amount required then this pop up window appears:

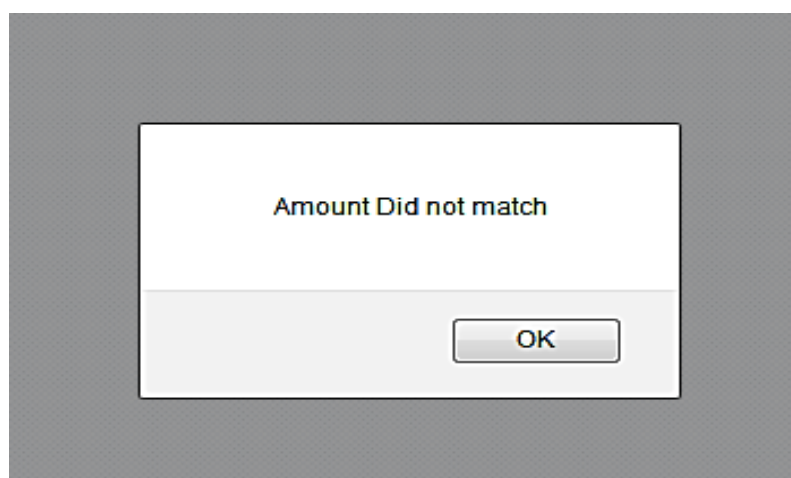


Figure 4.17: Insufficient amount



When the particular transaction entered exists in the list but the student id does not matches with the then this pop up window appears:

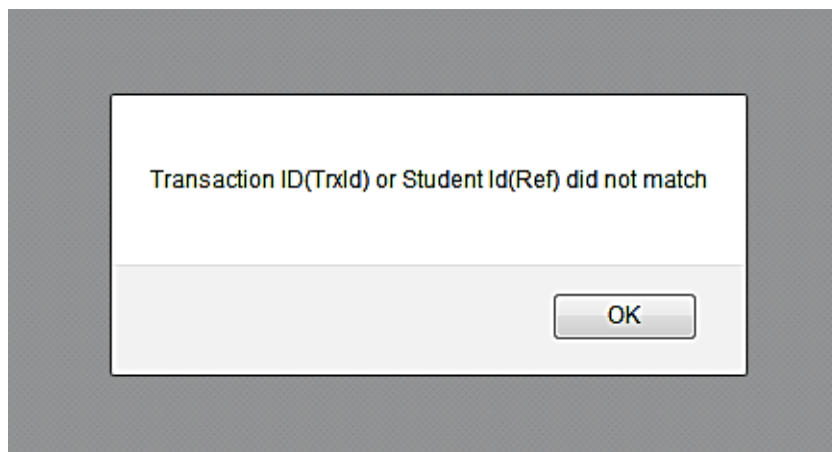


Figure 4.18: Incorrect ID

#### **4.5.2 Payment through Sonali Bank**

If the user choses to pay manually by going to the Sonali bak, CUET branch which is in a legal agreement with university authority, he/she has to collect the payment slip from the register office or hall office (depends on the account name hw/she wants to pay in). Then the particular amount has to be transferred to the bank. Once the Bank authority verifies the transaction, the student has to verify the transaction from the bank administrator of ACRS.

Bank administrator of ACRS will then enter the transaction into ‘Sbank’ table of databse of ACRS. Then the student has to enter the transaction id to the system which is mentioned in his/her receipt from the bank.



Figure 4.19: Payment via Bank

চট্টগ্রাম প্রকৌশল ও প্রযুক্তি বিশ্ববিদ্যালয়  
ছাত্র / ছাত্রীর বেতনাদির রশিদ  
চতুর্থ খন্ড (ছাত্র / ছাত্রীর কপি)

নাম ..... আই ডি নং .....  
বিভাগ ..... লেভেল ..... টার্ম .....  
সেশন ..... জমার তারিখ .....  
হিসাব নম্বর : চুয়েট, ছাত্র ফিস কালেকশন - ৩৪১২৯৩০২

কোড	বিবরণ	টাকা	পয়সা
০১	ছাত্র বেতন		
০২	ভর্তি ফিস		
০৩	সিট ভাড়া		
০৪	পরীক্ষার ফিস	২০০০	
০৫	বার্ষিকী ফিস		
০৬	জরিমানা / বিলম্ব ফিস		
০৭	ছাত্র সংসদ ফিস		
০৮	খেলাধুলা ফিস		
০৯	কমনরুম ফিস		
১০	ম্যাগাজিন ফিস		
১১	ছাত্র কল্যাণ ফিস		
১২	ধর্মীয় অনুষ্ঠান ফিস		
১৩	কশনমানি		
১৪	মূল / সাময়িক সনদপত্র / ডুপ্লিকেট		
১৫	গ্রেড শিট / ডুপ্লিকেট		
১৬	ট্রান্সক্রিপ্ট / ডুপ্লিকেট		
১৭	ডকুমেন্ট সত্যায়ন ফিস		
১৮	বইয়ের রেন্ট / লাইব্রেরী ফিস		
১৯	কনভোকেশন		
২০	মাইগ্রেশন ফিস		
২১	চিকিৎসা ফিস		
২২	রেজিস্ট্রেশন ফিস / কোর্স রেজিঃ		
২৩	উন্নয়ন ফিস		
২৪	পরীক্ষাগার ফিস		
২৫	যানবাহন ফিস		

SONALI BANK LTD.  
CHUET BRANCH, CTG.  
11 NOV 2014  
CASH RECEIVED

Figure 4.20: Bank receipt

The user accomplishing a successful payment will receive a transaction id. The next thing user has to do is to put that transaction id in the verify transaction field. The system will then match that transaction id with the transaction ids it has in the tables. If transaction id matches then the student id and amount is checked for the corresponding transaction. If all the parameters matches accurately then the student is considered eligible. When the particular transaction entered exists in the list but the amount is less than the amount required then pop up will appear and notify unsuccessful entry. And when the particular transaction entered does not exist in the database then also the pop up will appear and notify unsuccessful entry.

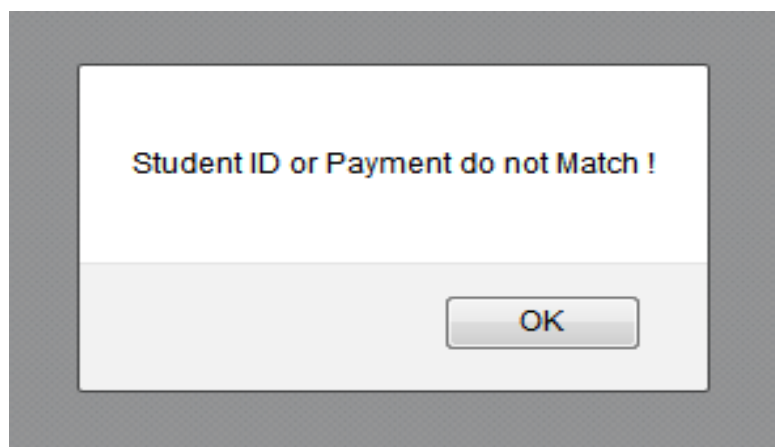


Figure 4.21: Unsuccessful payment

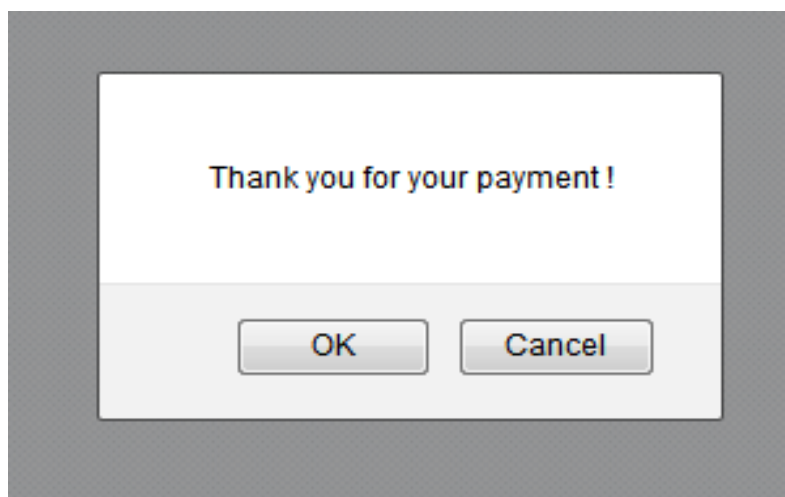


Figure 4.22: Successful payment

## 4.6 Registration

Once the student successfully completes a transaction he/she will then be given to option to download the registration form. Next step the student has to take is to print the document and put his/her signature on it and forward it to the register office.



---

Student ID : 0904093  
Payment For :Hall Office  
Total Amount in BDT :900  
Students Name :Muhammad Armaan-Ul-Islam  
Fathers Name :Muhammad Saiful Islam  
Mothers Name :Laila Arjumand Banu  
Level :4 Term :2

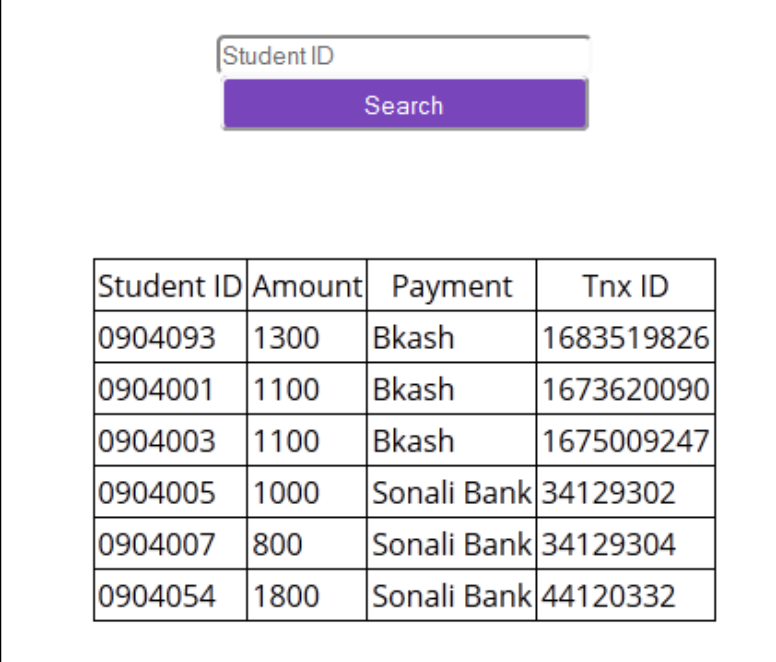
Course No	Course Title	Total Credit
CSE-419	VLSI Design	3
CSE-421	Digital System Design	3
CSE-422	Digital System Design (Sessional)	1.5
CSE-433	Software Engineering	3
CSE-434	Software Engineering (Sessional)	0.75
CSE-457	Computer Graphics	3
CSE-458	Computer Graphics(Sessional)	1.5
CSE-451	Digital Image Processing	3
CSE-400	Project and Thesis	3
	Total	21.75

-----  
Students Signature

-----  
Office Signature

Figure 4.23: Downloaded Registration form

Register office would have list of all successful transactions through ACRS. The students will deliver the registration forms to the register office. The administrator will then check the forms and match the Ids and payments with the forms in order to make sure the form is result of a successful payment.



The screenshot displays a web interface for viewing successful payments. At the top, there is a search bar labeled 'Student ID' with a purple 'Search' button. Below the search bar is a table with four columns: 'Student ID', 'Amount', 'Payment', and 'Tnx ID'. The table contains six rows of data representing successful transactions.

Student ID	Amount	Payment	Tnx ID
0904093	1300	Bkash	1683519826
0904001	1100	Bkash	1673620090
0904003	1100	Bkash	1675009247
0904005	1000	Sonali Bank	34129302
0904007	800	Sonali Bank	34129304
0904054	1800	Sonali Bank	44120332

Figure 4.24: List of successful payments

## 4.7 Administration Panel

To administrate the ACRS three administration panel is designed.

### Hall Admin

Hall administrator can log in to to the system by his unique user id and password through login page of ACRS. The task of hall administrator is to entry into the system the fees and fines of distinct students of each hall. Administrator can also edit or delete an entry.

### Hall Information

New Entry

Student_ID	Hall	Fine	Fee	Operation	Operation
0904123	SKHALL	600	700	EDIT	Delete
0904093	BBHALL	200	700	EDIT	Delete
0704001	BBHALL	1420	1200	EDIT	Delete

Figure 4.25: Hall administration login

### Bank Admin

Bank administrator can also log in to the system by his unique user id and password through login page of ACRS.. The task of bank administrator is to entry into the system the student id, amount and transaction id of each transactions. Bank administrator can be any staff of the bank or someone from the university authority.

### Super Admin

Super administrator is the one who is in charge of the ACRS. He/she is the one with flesh and bones within the system. Super administrator has a different panel to log in from.

Super administrator is given the authority to edit, delete or enter an entry in the database. He/she can also create and drop tables from database. The role of Super administrator is to

- Delete all data in the table after a registration session is expired to make sure no conflict occurs.
- Receive complains from Students, Bank admin or Hall admin and revise the entry in database.
- Contact with developer if any fatal error occurs.

### Bank payment Information

New Entry

Student_ID	Amount	Transaction Id
0904005	1000	34129302
0904007	800	34129304
0904054	1800	44120332

Figure 4.26: Bank administrator login

Overview Query Import Export
Logout

> Home

> Users

> Query

> Import

> Export

DATABASES

▶ information\_schema

▼ loveurln\_cuet

bkash (4)

hall (3)

office (6)

reg (6)

sbank (3)

student (9)

Select: All None With selected: Empty Drop Optimize

Table	Rows	Charset	Overhead
<input type="checkbox"/> bkash	4	utf8	704 B
<input type="checkbox"/> hall	3	utf8	
<input type="checkbox"/> office	6	utf8	
<input type="checkbox"/> reg	6	utf8	256 B
<input type="checkbox"/> sbank	3	utf8	
<input type="checkbox"/> student	9	utf8	160 B

OPTIONS

Drop the 'loveurln\_cuet' database

EDIT DATABASE

Charset: utf8

CREATE A NEW TABLE

Name:

Charset: utf8

Setup the fields for the table below:

<New field>

Name:

Size:

Default:

Other: ☐ Unsigned ☐ Binary ☐ Not Null ☐ Auto Increment

Type: varchar

Key:

Charset:

Add field

Figure 4.27: Super administrator login

## 4.8 Security

### 4.8.1 SQL injection

SQL injection is a code injection technique, used to attack data-driven applications, in which malicious SQL statements are inserted into an entry field for execution. SQL injection must exploit a security vulnerability in an application's software, for example, when user input is either incorrectly filtered for string literal escape characters embedded in SQL statements or user input is not strongly typed and unexpectedly executed. SQL injection is mostly known as an attack vector for websites but can be used to attack any type of SQL database. SQL Injection

SQL injection is a technique where malicious users can inject SQL commands into an SQL statement, via web page input.

Injected SQL commands can alter SQL statement and compromise the security of a web application.

#### SQL Injection Based on 1=1 is Always True

Let's say that the original purpose of the code was to create an SQL statement to select a user with a given user id. If there is nothing to prevent a user from entering "wrong" input, the user can enter some "smart" input like this:

User Id:

Server Result

```
SELECT * FROM Users WHERE UserId = 105 or 1=1
```

The SQL above is valid. It will return all rows from the table Users, since WHERE 1=1 is always true. The example above seem dangerous. What if the Users table contains names and passwords?

The SQL statement above is much the same as this:

```
SELECT UserId, Name, Password FROM Users WHERE UserId = 105 or 1=1
```

A hacker might get access to all the user names and passwords in a database by simply inserting 105 or 1=1 into the input box.



Here is a common construction, used to verify user login to a web site:

User Name:

" or ""="

Password:

" or ""="

Server Code

A hacker might get access to user names and passwords in a database by simply inserting " or ""=" into the user name or password text box. The code at the server will create a valid SQL statement like this:

```
SELECT * FROM Users WHERE Name ="" or ""="" AND Pass ="" or ""=""
```

The result SQL is valid. It will return all rows from the table Users, since WHERE ""="" is always true. So anyone can gain access in the system!

Such case may arise in ACRS while entering transaction Id, password. Thus a student will gain access to registration form without paying any fee!

## Prevention

To prevent such cases from arising we have used “`mysqli_real_escape_string('database connection', 'input string')`” function wherever a input field is given to the user. `mysqli_real_escape_string('database connection', 'input string')` requires the string to be escaped. Characters encoded are NUL (ASCII 0), `\n`, `\r`, `\`, `'`, `"`, and Control-Z. Thus the user don't have the chance to input SQL statement into the system. The system will strip special characters before feeding it into the system. As an example the entry of transaction id is shown below:

```
$tnx=mysqli_real_escape_string($con, $_POST['tnx']);
```

```
$tc=mysqli_real_escape_string($con, $_POST['tc']);
```

```
$studentid=mysqli_real_escape_string($con, $_POST['studentid']);
```

```
$total=mysqli_real_escape_string($con, $_POST['total']);
```

### 4.8.2 MD5 encryption

The **MD5** message-digest algorithm is a widely used cryptographic hash function producing a 128-bit (16-byte) hash value, typically expressed in text format as a 32 digit hexadecimal number. MD5 has been utilized in a wide variety of cryptographic applications, and is also commonly used to verify data integrity.

The 128-bit (16-byte) MD5 hashes (also termed *message digests*) are typically represented as a sequence of 32 hexadecimal digits. The following demonstrates a 43-byte ASCII input and the corresponding MD5 hash:

MD5 ("The quick brown fox jumps over the lazy dog") =  
9e107d9d372bb6826bd81d3542a419d6

Even a small change in the message will (with overwhelming probability) result in a mostly different hash, due to the avalanche effect. For example, adding a period to the end of the sentence:

MD5 ("The quick brown fox jumps over the lazy dog.") =  
e4d909c290d0fb1ca068ffaddf22cbd0

The hash of the zero-length string is:

MD5 ("") =  
d41d8cd98f00b204e9800998ecf8427e

MD5 encryption is introduced in ACRS while storing the passwords of the users. Now even if the hacker reached the database he/she won't be able to recognize the passwords in the system, which makes ACRS more secured system.

### 4.8.3 Transaction Source Verification

The situation may arise when a student may make a carbon copy of payment verification message of bKash and forward it to the SIM number provided by the system. Therefore he/she will expect that their payment status may be updated.

In reality ACRS has another security module designed just to handle this kind of intruder. Whenever any message is received in the message format of bKash money transaction but the sender is not 'bKash' which is a unique number that can only be used by the Brac bank server, even the computerized bulk SMS systems can't access using the number 'bKash' then the system detects an illegal attempt and pop up a window notifying.

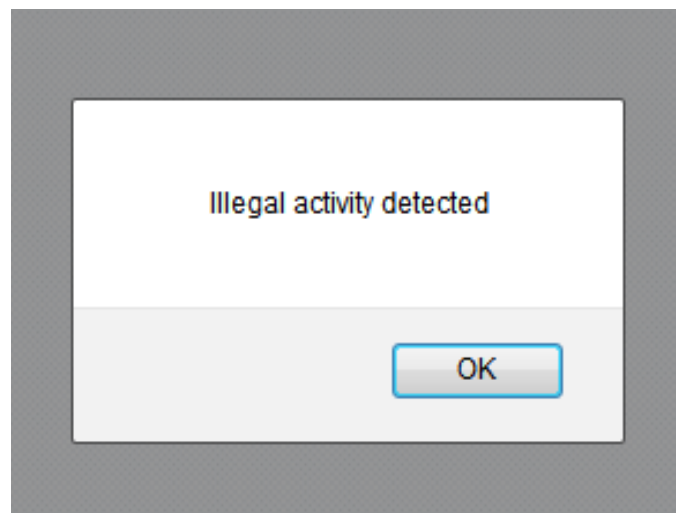


Figure 4.28: Fake payment SMS detection

### 4.8.4 Institution Code Verification

Institution Code would be a code provided by the Institution to the students who wants to open an account. This measure is taken to prevent account registration from out of the institution to maintain the privacy of information among the members of the Institution.

The User has to collect the ‘Institution Code’ in order to register an account. Once he has entered valid information in the fields, his/her initial and conformational password matches. And he/she enters the correct ‘Institution Code’, then account will be registered.

## **4.9 Analysis**

To analyze the system we are using an web system analyzing tool ‘Google developer web analysis tool’. .Page Speed Insights measures the performance of a web system for mobile devices and desktop devices. It fetches the url twice, once with a mobile user-agent, and once with a desktop-user agent.

The PageSpeed Score ranges from 0 to 100 points. A higher score is better and a score of 85 or above indicates that the page is performing well. PageSpeed Insights measures how the page can improve its performance also.

However, since the performance of a network connection varies considerably, PageSpeed Insights only considers the network-independent aspects of page performance: the server configuration, the HTML structure of a page, and its use of external resources such as images, JavaScript, and CSS. Implementing the suggestions should improve the relative performance of the page. However, the absolute performance of the page will still be dependent upon a user’s network connection. Each suggestion is rated with a priority indicator to indicate its importance.




Icon	Name	Description
	red exclamation point	Fixing this would have a measurable impact on page performance.
	yellow exclamation point	Consider fixing this if it is not a lot of work.
	green check mark	No significant issues found. Good job!

Fig 4.29 : Result indicator from google

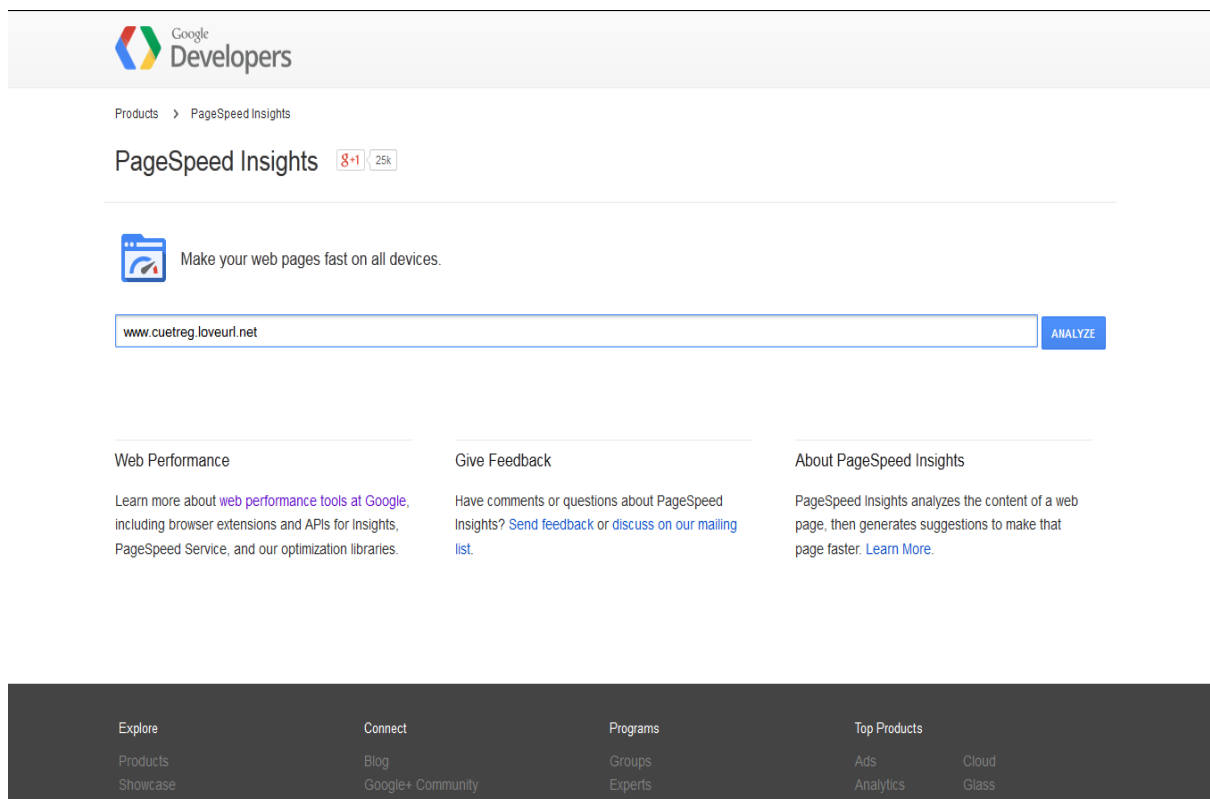


Figure 4.30: Google Developers PageSpeed Insights

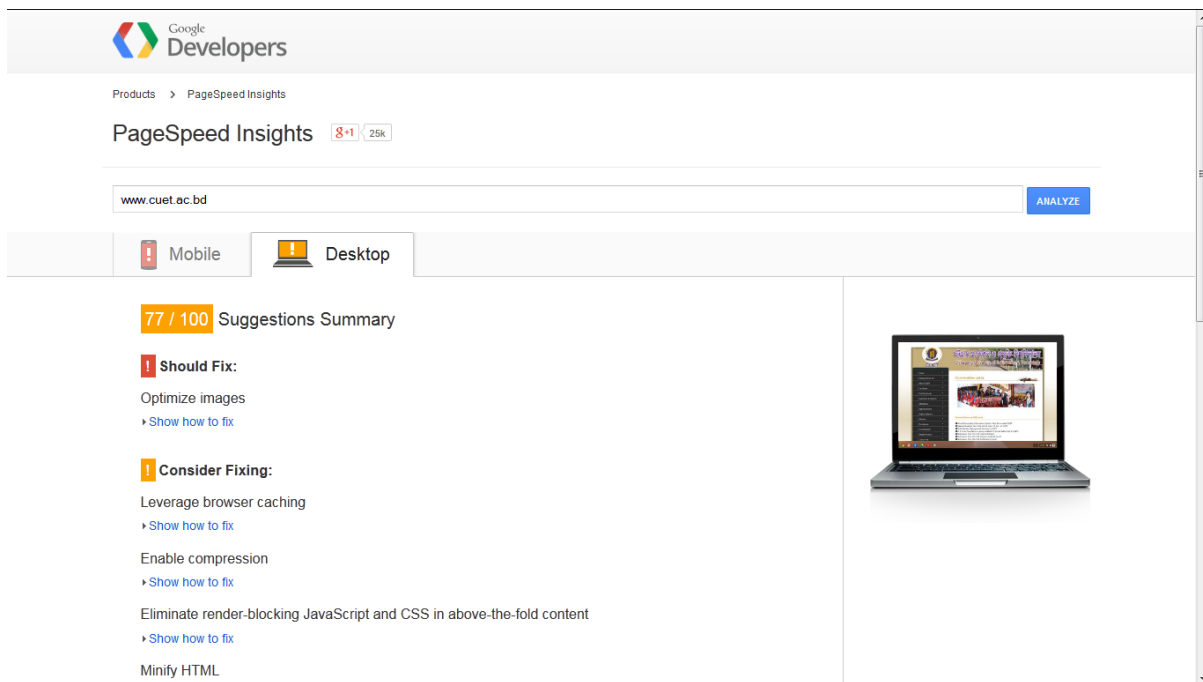


Figure 4.31: Google Developers PageSpeed Insights report on cuet.ac.bd

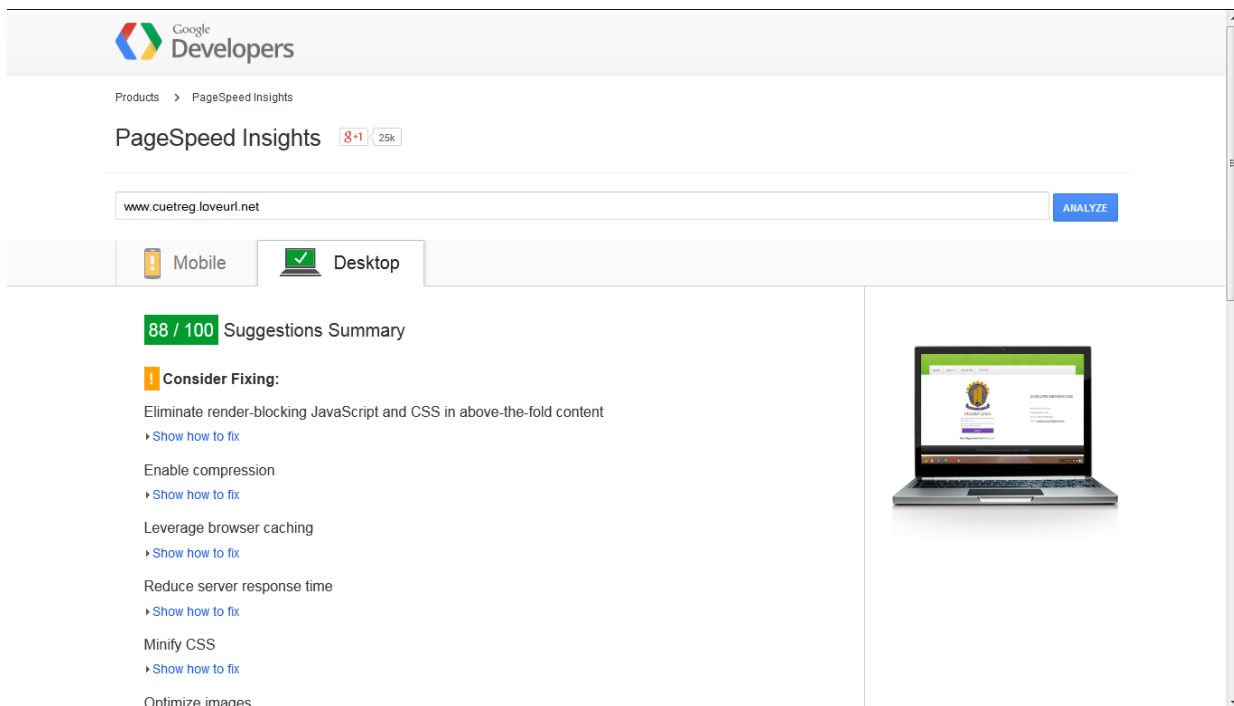


Figure 4.32: Google Developers PageSpeed Insights report on ACRS

# **Chapter 5**

## **Conclusion**

### **5.1 Overview**

Automation of Course Registration System (ACRS) will introduce automated student's courses registration in Chittagong University of Engineering & Technology and provide remedy to the tremendous massacre. The rapid increment of number of students in undergraduate studies and availability of technology demands the abolition of manual registration system which is the motto of ACRS. We strongly hope and believe that in this particular situation the ACRS will help out the students, the staff as well as the university up to a great extent to avoid the intolerable wastage of time and manpower.

### **5.2 Future Recommendation**

The Automation of Course Registration System is an application that would let the registration system to be led automatically. The vision of the work is to ease the task of registration. To fulfill the vision one may go to further extend to facilitate the university authority. The following issues may be taken into consideration

- The system still depends on massive human interactions which could be taken to the next step by avoiding the use of paper form and use online database to track the registrations.
- Our focus was solely on functionality, we would strongly recommend to design it with a legitimate look and appearance.

## Bibliography

- [1] M. Fahmy, “*Automated Student’s Courses Registration Using Computer-Telephony Integration*”, In Proc. Of The International Arab Journal of Information Technology, V4, No. 4, October, (2007) pp. 353-358.
- [2] A. M. Al-Shaikh, “*Online Registration System*”, In Proc. of International Journal of Computer Science and Security (IJCSS), Volume (4), Issue: (3),(2010), pp.331-345.
- [3] Md. Shahanur Islam, “*Mobile Banking: An Emerging Issue in Bangladesh*”, In Proc. Of ASA University Review, Vol. 7 No. 1, January–June, 2013, pp.1-8
- [4] Oracle.Database.11g.The.Complete.Reference by Kevin Loney
- [5] Learning PHP, MySQL, JavaScript & CSS 2nd Ed. by Robin Nixon  
(O'Reilly 2012, ISBN 978-1449319267)
- [6] Web Server  
[http://en.wikipedia.org/wiki/Web\\_server](http://en.wikipedia.org/wiki/Web_server)
- [7] MySQL vs. Oracle – The Complete Comparison  
<http://itxdesign.com/mysql-vs-oracle>
- [8] UTF-8  
<http://en.wikipedia.org/wiki/UTF-8>
- [9] Application Programming Interface  
<http://www.webopedia.com/TERM/A/API.html>
- [10] SMS gateway  
[http://en.wikipedia.org/wiki/SMS\\_gateway](http://en.wikipedia.org/wiki/SMS_gateway)



[11] ER model: Aggregation and Specialization

<http://understandingrdbms.blogspot.com/2013/04/data-model-part-2-specialization.html>

[12] SQL injection

<http://php.net/manual/en/security.database.sql-injection.php>

[13] MD5 encryption

<http://en.wikipedia.org/wiki/MD5>

[12] Google Developer PageSpeed Insight

<https://developers.google.com/speed/pagespeed/insights>

[13] bKash User terms and Conditions, cited on November (2014)

<http://www.bkash.com/terms-and-conditions>

# Appendix A

## Source Code

### payment.php

```
<?php

include"conX.php";

$tnx=mysqli_real_escape_string($con, $_POST['tnx']);

$tc=mysqli_real_escape_string($con, $_POST['tc']);

$studentid=mysqli_real_escape_string($con, $_POST['studentid']);

$total=mysqli_real_escape_string($con, $_POST['total']);

$u="http://cuereg.loveurl.net/paymentdone.php?studentid=".$studentid."&total=".$total;
```

```
include"conX.php";

$result = mysqli_query($con,"SELECT * FROM sbank");

$a=0;
```

```
while($row = mysqli_fetch_array($result)) {

    $amount=$row['amount'];

    $studentidb=$row['studentid'];

    $tnxid=$row['tnxid'];

    $amount=intval($amount); $tnxid=intval($tnxid);

    if($tnxid==$tnx)

    { if($amount>=$total){

        if($studentidb==$studentid){
```

```

$a=1;}

}

}

}

if($a==1)

{?><script>

if (window.confirm("Thank you for your payment !"))

{function Redirect(url)

{location.href = url; }

Redirect ("<?php echo $u; ?>");

}

</script>

<?php

} else {echo "<script>alert('Student ID or Payment do not Match !');</script>";}

?>

```

**/////MD5 encryption/////**

**md5()**

**{**

**var int[64] s, K**

```

s[ 0..15] := { 7, 12, 17, 22,  7, 12, 17, 22,  7, 12, 17, 22,  7, 12, 17,
22 }
s[16..31] := { 5,  9, 14, 20,  5,  9, 14, 20,  5,  9, 14, 20,  5,  9, 14,
20 }
s[32..47] := { 4, 11, 16, 23,  4, 11, 16, 23,  4, 11, 16, 23,  4, 11, 16,
23 }
s[48..63] := { 6, 10, 15, 21,  6, 10, 15, 21,  6, 10, 15, 21,  6, 10, 15,
21 }

```

```

for i from 0 to 63
    K[i] := floor(abs(sin(i + 1)) × (2 pow 32))
end for

```

```

var int a0 := 0x67452301  //A
var int b0 := 0xefcdab89  //B
var int c0 := 0x98badcfe  //C
var int d0 := 0x10325476  //D

append "1" bit to message

append "0" bit until message length in bits  $\equiv 448 \pmod{512}$ 
append original length in bits mod (2 pow 64) to message

for each 512-bit chunk of message
  break chunk into sixteen 32-bit words  $M[j]$ ,  $0 \leq j \leq 15$ 
  var int A := a0
  var int B := b0
  var int C := c0
  var int D := d0

  for i from 0 to 63
    if  $0 \leq i \leq 15$  then
      F := (B and C) or ((not B) and D)
      g := i
    else if  $16 \leq i \leq 31$ 
      F := (D and B) or ((not D) and C)
      g := (5i + 1) mod 16
    else if  $32 \leq i \leq 47$ 
      F := B xor C xor D
      g := (3i + 5) mod 16
    else if  $48 \leq i \leq 63$ 
      F := C xor (B or (not D))
      g := (7i) mod 16
    dTemp := D
    D := C
    C := B
    B := B + leftrotate((A + F + K[i] + M[g]), s[i])
    A := dTemp
  end for

  a0 := a0 + A
  b0 := b0 + B
  c0 := c0 + C
  d0 := d0 + D
end for

var char digest[16] := a0 append b0 append c0 append d0

leftrotate (x, c)
  return (x << c) binary or (x >> (32-c));}

```