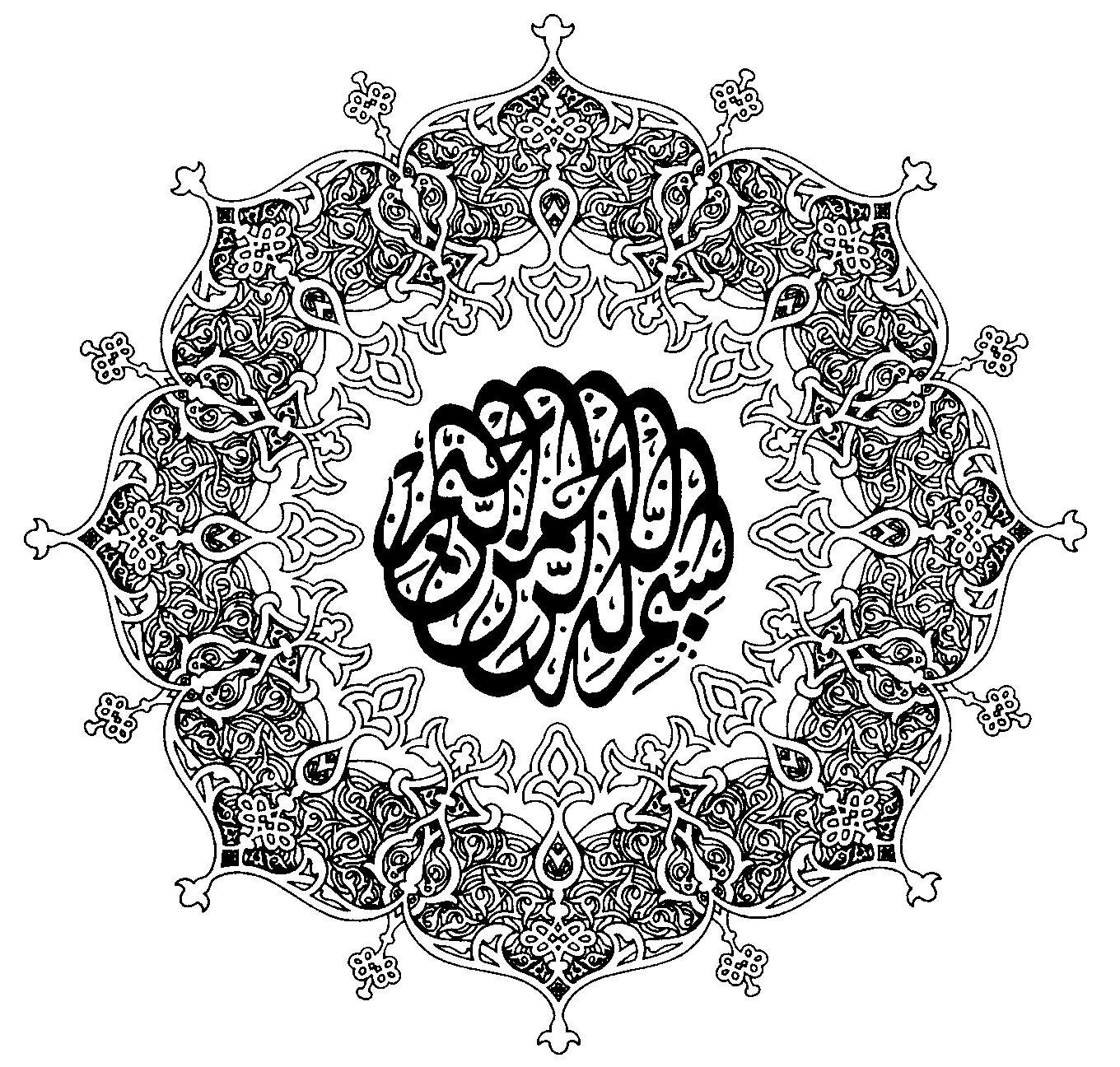
1. 
2. IN THE NAME OF ALLAH, THE MOST GRASIOUS, THE MOST MERCIFUL, AND THE MOST COMPASSIONATE.
3. 

***Junaid Ahmed***

***Roll No: 211***

***Bs (IT) (2011-2015)***

2. ***Muhammad Attique***
3. ***Roll No: 226***
4. ***Bs (IT) (2011-2015)***

**Department Of Computer Science & IT**

**The Islamia University of Bahawalpur**

1. **Final Approval**
2. It is certified that project submitted by **Junaid Ahmed Roll No.211 and** **Muhammad Attique Roll No.226 Session 2011-2015** is up to the standard. It is accepted by The Islamia University of Bahawalpur for the BSIT Degree in Computer Science.
3. **Supervisor:**
4. ***Dr. Nadeem Akhtar*** \_\_\_\_\_\_\_\_\_\_\_\_\_\_
5. ***Assistant Professor***
6. ***Computer Science and Information Technology,***
7. *The Islamia University of Bahawalpur.*
8. ***Mr. Muhammad Moosa*** \_\_\_\_\_\_\_\_\_\_\_\_\_\_
9. ***Lecturer***
10. ***Computer Science and Information Technology,***
11. *The Islamia University of Bahawalpur*.
12. **External Examiner**: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. **Head of Department: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**
2. ***Dr. Dost Muhammad Khan***
3. ***Computer Science and Information Technology***
4. *The Islamia University of Bahawalpur*
5. **Project Brief**
6. **Project Name : *Online swift info***
   1. **Undertaken BY :**
   2. **Junaid Ahmed**
   3. **and**
   5. **Muhammad Attique**
      * + 1. **Supervised By :**
          3. **Dr. Nadeem Akhtar**
          4. **and**
          5. **Mr. Muhammad Moosa**

**Operating System Used : Window 8.1/Windows 7(32bit) Ultimate**

**Project Stared : July-01-2015**

**Project Finished : Aug-12-2015**

**Source Language : PHP/MySQL Database**

1. **DEDICATION**
2. ***Our Allah Almighty***
3. ***Who Gave us Strength***
4. ***And determination***
5. ***To Accomplish our Goal.***
6. **Acknowledgement**
7. First of all, our deepest gratitude to **Almighty ALLAH**: the most beneficent, compassionate, most merciful and most gracious whose favor and kindness made it possible for me to complete this project work.
8. A very special thanks and appreciation goes to our parents and other family members for always encouraging us, whose endless prayers are a source of determination for us.
9. We wish to thank to ***Mr. Dr. Dost Muhammad*** Head of Department for providing us the facility and best environment to complete our task.
10. We are also greatly indebted to our respectable teacher ***Mr. Dr. Nadeem Akhtar and Mr. Muhammad Moosa*** for their supervision, kind support, unforgettable devotion and encouraging behavior.
11. We are also thankful to all staff members of the department of computer science &IT for their coordination.
12. ***Junaid Ahmad***
13. ***Muhammad Attique***

# **Abstract**

1. Swift info can be used by education institutes to manage and maintain the records of user events and alerts by displaying on online at website. Achieving this objective is difficult using a manual system as the information is scattered, can be redundant and collecting relevant information may be very time consuming. All these problems are solved using this project.
2. This project has the information needed for any department and faculty profiles as well. Here different Notices also will be displayed about results or time table.
3. Project is used mostly in management of students in different education institutes. Here activities and grades information system is controlled by administrator. Activities are displayed to those users who are registered and to the administrators as well.
4. The creation and management of accurate, up-to-date information regarding a students’ academic career is critically important in the university as well as colleges. Swift Info system deals with all kind of student details.

**Table of Contents**

1. **Project overview**
   1. Introduction……………………………………………………………………………………………….... 4
   2. Background…..……………………………………………………………..……………………….………5
   3. Goals and Objectives…...……...……………………………………………………………………....8
   4. Scope of Project……..……….…………………………………………………………………………...8
2. **Analysis and Design**
   1. **Rational Unified Process**

2.1.0 Inception Phase………………………………………………………..…………………………………10

2.1.1 Project Description………………………………….......................................10

2.1.2 Global Factors…………..……………….…………………………………………………….…….12

2.2.0 Elaboration Phase………………………………………..................................................13

2.2.1 Actors.……….………….………………………………………………………….……..13

2.2.2 Actors Description….…………………....……………………………….…….....13

2.3.0 Use Case Diagram for User and Administrator ……………………….….14

**2.2 User Diagrams**

2.2.1 Use Case, Sequence and Activity Diagram for Home Page……………………….16

2.2.2 Use Case, Sequence and Activity Diagram for about Page………………….…..19

2.2.3 Use Case, Sequence and Activity Diagram for Contact Page……………….….22

2.2.4 Use Case, Sequence and Activity Diagram for Register Page………………....25

2.2.5 Use Case, Sequence and Activity Diagram for Login Page…………………..……28

**2.3 Administrator Diagrams**

2.3.1 Use Case, Sequence and Activity Diagram for Login Page………………………...31

2.3.2 Use Case, Sequence and Activity Diagram for User Profiles………………………35

2.3.3 Use Case, Sequence and Activity Diagram for Students……………………….…..39

**3. Internal Technical Design**

3.1 Database Management…………………………………………………………………………...44

3.2 Database Design………………………………………………………………………………….…..45

3.3 Final Database Tables………………………………………………………………………….….47

**4. Interface and Design**

4.1 Admin interface and Design……………………………………………………………………..…...51

4.2 User Interface and Design……………………………………………………………………..……54

**5. Website Development and Design**

5.1 Development Approach ………………………………………………………………………………..57

5.2 Conceptual Design ……………………………………………………………………………………….58

5.3 Top Level Information Architecture……………………………………………………………..59

5.4 Tools and Technology Used With Description………………………………………….…..61

**5.1 Appendices**

5.1.0 References Web and Book…………………………………………………………………..….64

#### Chapter No

01

**Project overview**

***Introduction***

Department Information System is a Web based management and information system for education establishments to manage student data. Online swift info provide you facilities of adding a department and program of that department. Also Provides Adding Students to be enrolled and manage them, adding instructors and adding activities.

It provide capabilities for entering marks of student in the different subjects and automatically calculates marks then enable administrator (Teacher, Registrar) to send them via email or to show on user side and other student-related data needs in a Department.

It provides Maintenance and reporting of student data, enrolling new students, Displaying class and teacher schedules.

It provides information about what kind of services provided by them is and how this work is done. In what kind of services done area a user can get information about different fields in department and how user can get information about activities, grades, results etc.

The main aim of the project is to develop a complete web application for online Information system of an educational institute or organization which will facilitate the user to manage all the information regarding education of the respective Institute.

***Background***

The Internet is changing the life and the world web site is a key factor in this revolution. Establishing a Web Site can be one of the most important thing moves you make and when web site is educational it should be according to the need. Because of the enormous potential and significant information involved, it is essential that you make this move carefully. This part of document provides project specifications and discusses the feasibility study of educational site of Swift info.

Web site is one of the main sources of knowledge these days for any

Person to familiar with his need and also finds the way to fulfill his needs .The input of this document is the result of survey, based on people views & education trends.

**What is web site?**

A web site is a place on the internet where we can get the information and knowledge about any topic of our own desire or choice related to site which we are visiting. Or one can say that a person who is visiting a site is logically present on the spot.

**What information does it provide?**

A web site should provide each and everything for which purpose it is being developed. Through a web site it has become easier for us to get the information of any kind and also to communicate with concerned person.

For a common person (non-IT professional) a web site is most comprehensive reliable media by which he/she can access easily everywhere in the world. As for related to our own requirement, nothing is out of reach. We should say a web site is a media on the net by which a world is become as small place (global village) as if we are present in the same place at the same time. Not only for a professional computer user but for a non-IT professional a web site is more useful and easiest way to get in touch over the world. Just by putting the site of our own choice can get all the information related to that particular field.

For example for a doctor, it has become easier to discuss different problems and share his/her knowledge with others.

**What information does an educational site provide?**

As mentioned a web site should provide all information related to organization it belong. Same is the case for educational site. An educational site provides information about what kind of services provided by them is and how this work is done. In what kind of services done area a user can get information about different fields in department and how user can get information about activities, grades, results etc. Now days, when the world is global village everything is in access of user’s single click on the net then why should a student or user physically go to department for information about his desire thing or information. Student just sit back in front of computer screen and browse education sites and see the information and finally get the information. Also Swift info facility is very much popular these days. People do not have much time to physically reach university they just use this medium which not only save time but also money. You need not to be worry to go anywhere but in front of your computer screen.

In the beginning it was consider that computer and its fields are mean for those who are related to it , means either they are business man or any person of this profession (Any student, engineer, IT Professional, any Hardware or Software programmer).

These educational sites also provide information about different universities. And it has become easier for any person to get and to collect all information where they want to take admissions etc.

### 

### Goals and Objectives

Following key goals and objectives have been identified for web site.

* Site has to easy and simple to navigate.
* One click experience for user to get to the information they want.
* Quick response on most interaction on the website.
* Allow users to quickly find the info they want.

### Scope of Project

This document describes:

1. Setting up the website
2. Setting up a process to maintain and support the website in educational environment.
3. Provide an interface for user to interact easily.
4. Provide an interface for user to view online grades & other information.
5. Administrator interface through which new information like grades, instructors & student’s information etc. can be added easily.
6. Swift info is designed for Educational Institutes (like schools, universities).
7. The system handles all the operations, and automatically generates grades of student.
8. Allow students to see activities.
9. Allow management to display activities and grades.

#### Chapter No

02

**Analysis and Design**

# **Rational Unified Process**

The Iterative Process works well with Object Oriented methodologies. It helps to identify and address risks early in the process leading to more robust and better quality systems. Here is a brief description of the process,

This Iterative, risk driven process is divided into.

1. Inception
2. Elaboration

## **Inception Phase**

### Project Description

In Swift Info project there are two fields of interest

1. User
2. Administrator Interface

**1. User:**

Here we will describe how user interacts with Web Site.

When user enters URL in browser then first of all Index page is opened. From this page user can move to any other link. It contains different links for navigation of site. Now we describe the links one by one.

First is the **Home** Link from which the user will start the navigation of site, and user can reach the home page from anywhere of the site

Second is the **ABOUT US** which gives the introduction of Swift Info.

In introduction, User can get the information, when Swift info was established and its working.

Third is **Gallery** in which a user watch the pictures of the different events latest.

Fourth is **Contact page** in which a user can contact in any problem that he faced during the process of getting information.

Fifth is **Register page**, which allow user to get registered by the use of register form.

Sixth is **login page**, which allow user to get registered by the use of sign in form.

Every page in website has **FOOTER** and **HEADER.**

**Administration Interface:**

Here we will describe how Administrator interacts with Swift Info Web site.

For the security reason I have hide the Administration interface and only Administrator knows how he reached on this.

When Administrator enters, he will have to enter on web space allowed which is specific for Swift info, where he will provide his Log In ID and Password, On verification Administration area on web will open where he can administrate the info like view student and edit, add new student, add new instructor etc.

* He can add/view/update/delete new students.
* He can add activities and view activities.
* He can add description of photos.
* Add new activities, update and delete each information
* Add new Departments update, delete each information
* It has a user management system
* He can view and delete registered users

### Global Factors

* In most part of the world, all adults spent most of their time on internet. They have limited time available for checking results physically, so are usually willing to check their result, activities online.
* Online educational swift info is very much popular these days.
* Other educational sites provide 24- hour access, response time ranging from overnight to very few days.

## **Elaboration Phase**

In this phase I shall identify system boundaries, by identifying actors and use cases, and I shall break the system into sub systems.

### Actors

There are two actors that interact with Swift info web site.

* User
* Administrator

**Figure**

Administrator

*(from Actors)*

user

*(from Actors)*

### Actors Description:

**USER**

User can view general as well as specific information about Swift info and its work.

**Administrator**

This is administrator user who has maximum privileges. He can add new students, departments etc. He will authenticate before accessing the web site.

## **Use Case Diagrams For SWIFT INFO**

View about Us

View contact us

View Register

View Login

User

(From Logical View)

View Home

View Services

View Gallery

##### Administration of department

##### Administration of student

##### Administration of student

##### Administration of Course

##### Administration of Subject

##### Administration of Session

##### View Home

Administrator

(From Logical View)

##### Administration Login

l

r

##### Administration of Instructor

##### Administration of Manage Users

Administration of Notices

## **Package: User**

### Use Case Name: Home

**Description:**

* This use case describes the process to check the home page of Swift Info.

**Actors:**

* User

**Use Case Diagram: HOME**

Home

*(From User)*

User

*(From Actors)*

**Pre-Conditions:**

* User must view the Swift info site.

**Flow of Events:**

**Primary Scenarios:**

* Use case begins when user enter URL in browser.
* Home page will be shown to user.

**Secondary Scenarios:**

* User does not request for Swift info website.

**Post Conditions:**

* User closes the browser.

**Sequence Diagram:**

*Home Page*

*: User*

*Browser*

*Enter SI URL Page*

*Return Home Page*

*View Home Page*

*Return to Home*

*1:*

*2:*

*3:*

*:*

**Activity Diagram:**

*Enter the SI site URL*

*End*

*Start*

*Home Page*

## **Use Case Name: About**

**Description:**

* This use case describes the process to check about us page of Swift info site.

**Actors:**

* User

**Use Case Diagram:**

About

*(From User)*

User

*(From Actors)*

**Pre-Conditions:**

* User must view the Swift Info site.

**Flow of Events:**

**Primary Scenarios:**

* Use case begins when user clicks About Us link.
* About page will be shown to user.

**Secondary Scenarios:**

* User does not request for about page and “cancel” the activity.

**Post Conditions:**

* User closes the browser.

**Sequence Diagram:**

*Home Page*

*: User*

*Browser*

*About Page*

*Enter SI Url Page*

*Return Home Page*

*Select About Link*

*Required About page is Returned*

*Return to Home*

*1:*

*2:*

*3:*

*4:*

**Activity Diagram:**

*Enter The SI site URL*

*End*

*Start*

*Home Page*

*Select About*

*View about Page*

## **Use Case Name: Contact Us**

**Description:**

* This use case describes the process to Contact about any problem about information through Swift info.

**Actors:**

* User

**Use Case Diagram:**

Contact us

*(From User)*

User

*(From Actors)*

**Pre-Conditions:**

* User must view the Swift Info site.

**Flow of Events:**

**Primary Scenarios:**

* Use case begins when user clicks Contact us link.
* Contact page will be shown to user.

**Secondary Scenarios:**

* User does not request for contact page and “cancel” the activity.

**Post Conditions:**

* User closes the browser.

**Sequence Diagram:**

*Home Page*

*: User*

*Browser*

*Contact us Page*

*Enter SI URL Page*

*Return Home Page*

*Select contact us Link*

*Required contact page is returned*

*Return to Home*

*1:*

*2:*

*3:*

*4:*

**Activity Diagram:**

*Enter The SI site URL*

*End*

*Start*

*Home Page*

*Select contact page*

*View contact page*

### Use Case Name: Register

**Description:**

* This use case describes the process to get registered to Swift Info.

**Actors:**

* User

**Use Case Diagram:**

Log in

*(From User)*

User

*(From Actors)*

**Pre-Conditions:**

* User must view the Swift Info.

**Flow of Events:**

**Primary Scenarios:**

* Use case begins when user clicks Register link at Header.
* Register page will be shown to user.

**Secondary Scenarios:**

* User does not request for Register page and “cancel” the activity.

**Post Conditions:**

* User closes the browser.

**Sequence Diagram:**

*Home Page*

*:User*

*Browser*

*Register*

*Enter SI URL Page*

*Return Home Page*

*Select Register Link*

*Required Register page is returned*

*Return to Home*

*1:*

*2:*

*3:*

*4:*

**Activity Diagram:**

*Enter The SI site URL*

*End*

*Start*

*Home Page*

*Select Register*

*View Register Page*

### Use Case Name: Log In

**Description:**

* This use case describes the process to Log In to Swift Info.

**Actors:**

* User

**Use Case Diagram:**

Login In

*(From User)*

User

*(From Actors)*

**Pre-Conditions:**

* User must view the Swift Info.

**Flow of Events:**

**Primary Scenarios:**

* Use case begins when user clicks Log In link.
* Log In page will be shown to user.

**Secondary Scenarios:**

* User does not request for Log In page and “cancel” the activity.

**Post Conditions:**

* User closes the browser.

**Sequence Diagram:**

*Home Page*

*: User*

*Browser*

*Sign In*

*Enter SI URL Page*

*Return Home Page*

*Select Log in Link*

*Required Log In page is returned*

*Return to Home*

*1:*

*2:*

*3:*

*4:*

**Activity Diagram**:

*Enter The SI site URL*

*End*

*Start*

*Home Page*

*Select Log in*

*View Log in Page*

## **Package: Administrator**

### Use Case Name: Administrator Log In

**Description:**

* This use case describes the process by which administrator will log in on the Swift Info web site

**Actors:**

* Administrator.

**Use Case Diagram:**

Administrator

*(From Actors)*

*Administrator Log In*

*(From Administration)*

**Pre-Conditions:**

* Administrator must view the Swift Info site.

**Flow of Events:**

**Primary Scenarios:**

* The use case starts when the administrator starts application.
* The Administrator enters secret URL.
* The system will display login screen.
* Administrator will enter login name and password.
* The system will verify information.
* Main menu will be shown to the system.

**Secondary Scenarios:**

* Information not valid.
* Administrator closes the browser

**Post Conditions:**

* Administrator will log in.

**Sequence Diagram:**

*: Administrator*

*Browser*

*Admin Login*

Page

*Storage*

*1:*

*Admin Page URL*

*2:*

*Return Admin Page*

*3:*

*Enter User Name And Password*

*Get Data*

*After Verification Return Admin Page*

*If Not Verified*

*Verification*

*4:*

*5:*

*6:*

*7:*

**Activity Diagram:**

****

### Use Case Name: User Profile

**Description:**

* This use case describes the process by which Administrator will view and delete information about Registered Users.

**Actors:**

* Administrator.

**Context:**

* System use case.

**“Used” Use cases:**

* Login.

**Use case diagrams:**

Administrator

(From Actors)

User Profile

**Associated Requirements:**

* Profile information about particular User.

**Precondition:**

Administrator must have logged in through Administrator Login (Admin Login Use case).

**Flow of events:**

**Basic Path:**

* Use case starts when user clicks view profile.

**Scenario variations:**

* User clicks close button of browser.

**Post conditions:**

* Return back to the menu page.

**Sequence Diagram:**

*Admin Page*

*: Administrator*

*Browser*

*View Page*

*Storage*

*1:*

*2:*

*3:*

*4:*

*Enter Admin URL Page*

*Return Admin Page*

*Select View Profile Link*

*Return view Page*

*Click User name to delete Or Update particular User*

*Required Page is returned*

*Updated the Record*

*Processing*

*Record is updated*

*Return To Administrator*

*5:*

*6:*

*7:*

*8:*

*9:*

*If the Values are Already*

Present or Wrong

**Activity Diagram:**

*Enter the Admin*

Page URL

*End*

*Start*

*Admin Page*

*If Log In*

*Select The*

View Profile

*View Page*

*View The*

record

*Update/Delete the record*

*If Not LogIn*

### Use Case Name: Students

**Description:**

* This use case describes the process by which administrator will Add/view/Update/Delete Students on website.

**Actors:**

* Administrator

**Use Case Diagram:**

Administrator

*(From Actors)*

*Administration of Students*

*(From Administration)*

**Pre-Conditions:**

* Administrator must log in to the Swift Info site.

**“Used” Use Cases:**

* Log In

**Flow of Events:**

**Primary Scenarios:**

* Use case starts when Administrator clicks link “Add New Student”.
* An input form is displayed for administrator through which he can add Student.
* Administrator click on project ID to View/Update/Delete particular Student.

**Secondary Scenarios:**

* Administrator closes the browser without any change.

**Post Conditions:**

* Administrator will add the new Student.

**Sequence Diagram:**

*Admin Page*

*: Administrator*

*Browser*

*ADD Student Page*

*Storage*

*Enter Admin Url Page*

*Return Admin Page*

*Select Add New Student Link*

*Return ADD Student Page*

*ADD new student and Select Option to view/ Update or delete students*

*Required Page is returned*

*Updated or delete the Student*

*Processing*

*Record is Updated or deleted*

*Return To Administrator*

*If the Values are Already*

Present or Wrong

*1:*

*2:*

*3:*

*4:*

*5:*

*6:*

*7:*

*8:*

*9:*

**Activity Diagram:**

*Enter the Admin*

Page URL

*End*

*Start*

*Admin Page*

*If Log In*

*Click The*

ADD New Student

*ADD student Page*

*View/update/delete the*

Student

*Add the Student*

*If Not LogIn*

## 

#### Chapter No

## 03

## **Internal Technical Design**

**Database Management:**

A shared collection of logically related data, design to meet the information needs of multiple users in an organization. The term database is often erroneously referred to as a synonym for a “database management system (DBMS)”. This provides powerful query management facility to enable the user to access data and use it for his needs. A DBMS is a system software package that helps the use of integrated collection of data records and files known as databases. It allows different user application programs to easily access the same database. DBMSs may use any of a variety of database models, such as the network model or relational model. In large systems, a DBMS allows users and other software to store and retrieve data in a structured way. This means that the user need not learn the SQL language commands to formulate his query and get the information he wants. It provides ease of use by the user.

**Further Advantage of Database System:**

Database systems are very much beneficent to enterprises and businesses, some of the advantages are listed below:

* Data Consistency
* Better Data Security
* Faster Development of new Applications
* Economy of sale
* Better Concurrency Control
* Better Backup and Recovery Procedures
* Reduced Data Redundancy
* Reduced Updating Errors and Increased Consistency
* Greater Data Integrity and Independence From Applications Programs
* Improved Data Access to Users Through Use of Host and Query Languages
* Improved Data Security
* Reduced Data Entry, Storage, and Retrieval Costs
* Facilitated Development of New Applications Program

## **Data Base Design**

These activities shall consist of:

* Designing appropriate database structures to store those identified persistent classes which identified during database design
* Defining techniques and strategies for storing and retrieving persistent data in such a way that the performance criteria for the project are met.

**Entity**:

An **entity** is a thing or object of importance about which data must be captured. All things aren't entities—only those about which information should be captured. Information about an entity is captured in the form of attributes and/or relationships. If something is a candidate for being an entity and it has no attributes or relationships, it isn't an entity. Database entities appear in a data model as a box with a title. The title is the name of the entity.

**Attributes**:

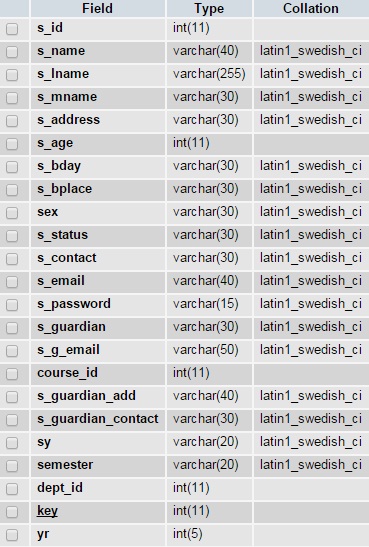
An attribute of an object usually consists of a name and a value; of an element, a type or class name; of a file, a name and extension. An **attribute** is a specification that defines a property of an object, element, or file. It may also refer to or set the specific for a given instance of such. For clarity, attributes should more correctly be considered metadata. An attribute is frequently and generally a property of a property. However, in actual usage, the term attribute can and is often treated as equivalent to a property depending on the technology being discussed.

## **Database Tables**

Under are the database tables and their relationships and constraints in SQL server view.

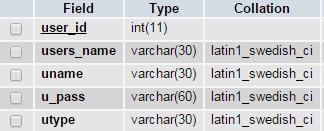
**User Table:**

This table has data for Students of Swift info.

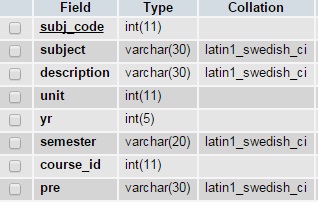
****

**Admin Table:**

This table has data for Sign In form of Admin.

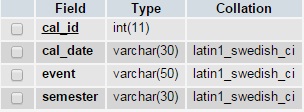


**Subjects Table:**

This table has data for product categories of Swift Info.

**Activity Table:**

This table has data of activities to send registered students.

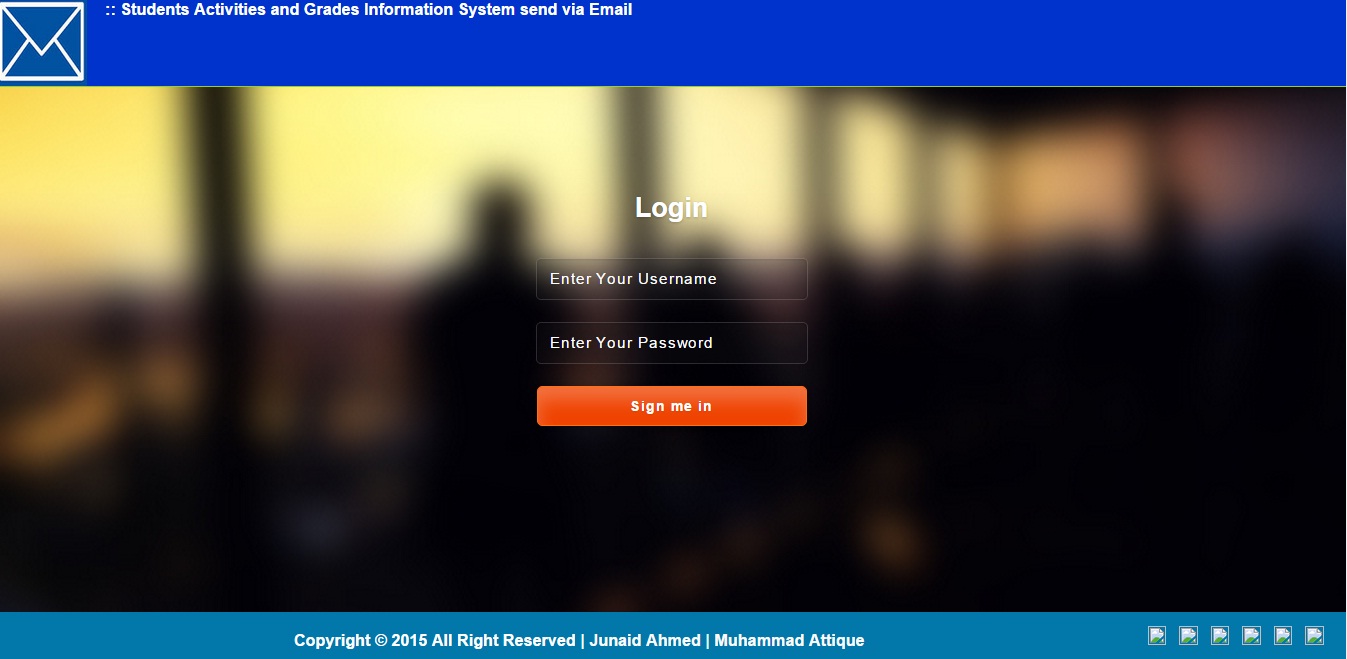


#### Chapter No

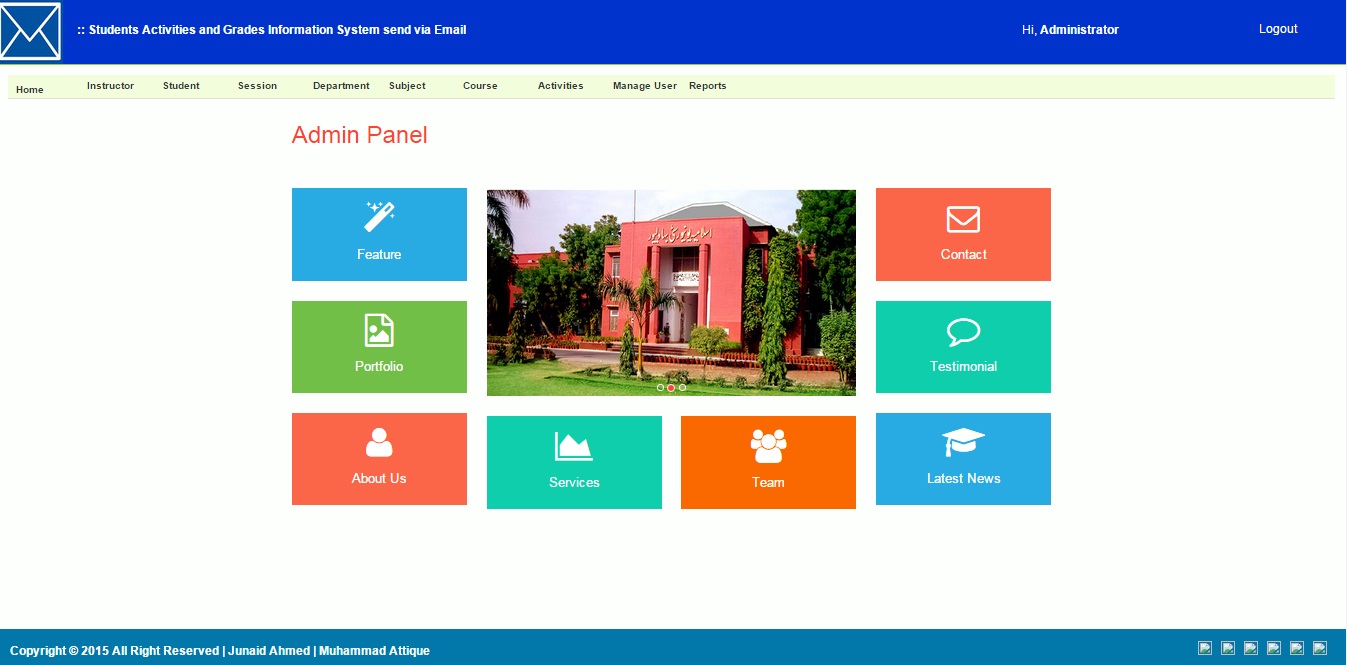
04

# **Interface and Design**

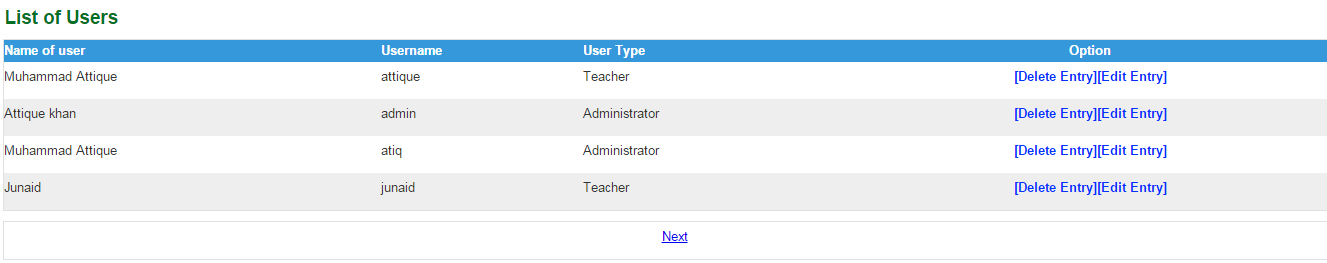
## **Admin Login:**



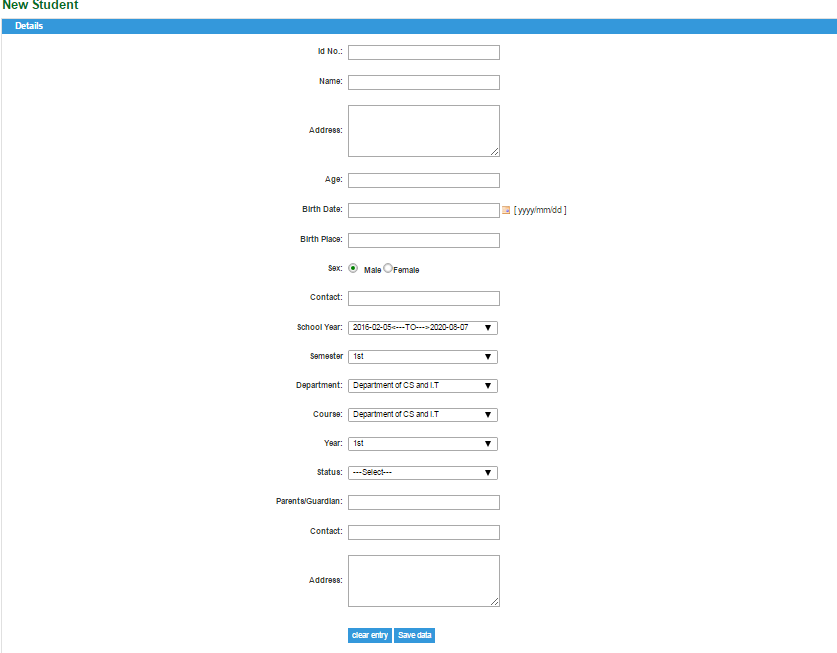
## **Admin Home:**



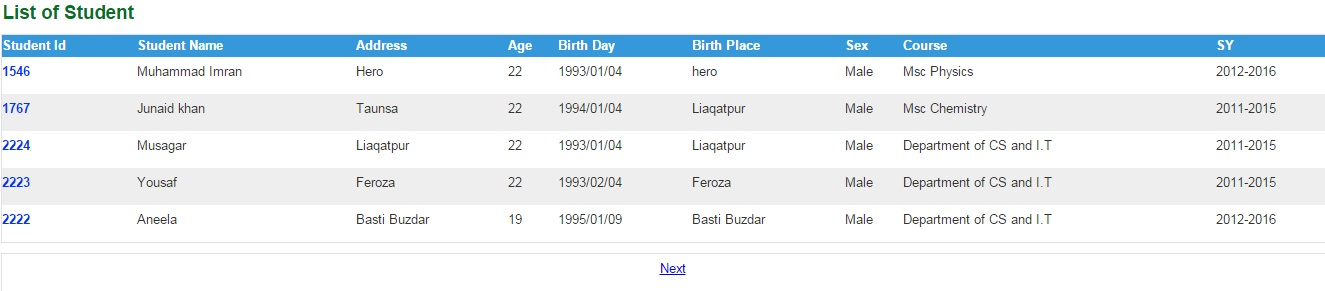
## **View Registered Users:**



## **Add New Student:**



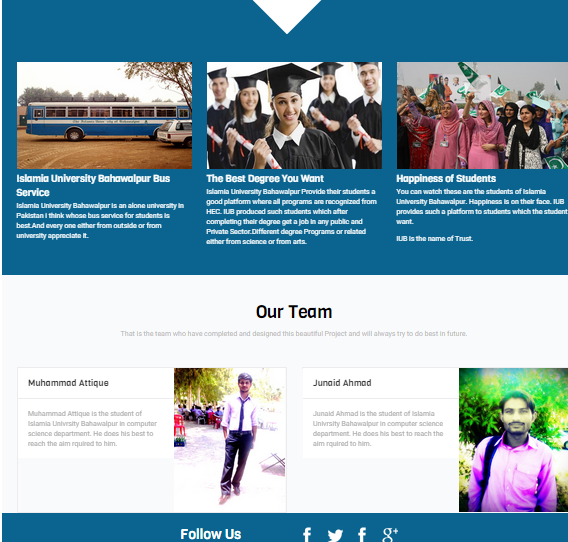
## **View Students:**



# **Client Side Home:**

# C:\Users\Ateeq\Desktop\Front end Pic.png

# **Client Side Home 1:**



#### Chapter No

05

**Website Development and Design**

### Development Approach

During we shall use RUP (Rational Unified Process) modeling to:

* Identity deliverables for all the modules.
* Develop and interaction matrix to specify mapping between actors and sub-systems.
* Verify that modeling meets the requirements of SWIFT INFO.
* Identity Educational Objects and develop Object Interaction, Sequence, Collaboration and Activity Diagrams.
* Develop high-level information architecture.

### 

**Conceptual Design**

**Screen Resolution:**

This web site can be best viewable on 800X 600 resolutions.

**Page Margins:**

By default all web pages exhibit margins around all sides that can be eliminated to provide extra space and /or enhance visual appeal. Hot wear shop website will not have any page margins.

**Horizontal Scrolling:**

Horizontal scroll is eliminated almost from all pages.

**Browsers:**

It will have all the tags compatible with both major browsers, IE 4.0 or higher and Netscape 3.75 or higher.

**Display Speed:**

The web site will contain lighter graphics, so that it may be loaded easily even if ISP’s bandwidth is not up to the mark.

## 

## **Top Level Information Architecture:**

**USER SIDE:**

Visitor / User

Visitor / User

General View

General View

View projects

User Log In

Maintain Profile

**Administration Site:**

Administrator

Log In

Information Administration

Profile Administration Pages

## **Tools and Technology:**

**Tools Required**

* Apache Web server Version
* PHP Script Language Version 5.1.1
* MYSQL Database Manager Version 2.6.4-pl4
* Adobe Dreamweaver Version CS 5
* Mozilla Firefox and Google Chrome
* XAMP and WAMP server

**PHP:**

**What is PHP?**

PHP (PHP Hypertext Preprocessor) is a reflective programming language originally designed for producing dynamic web pages. PHP is used mainly in server-side scripting, but can use for a command line interface or in standalone graphical application.

The main implementation is produced by “The PHP Group” and released under the PHP License. It is considered to be free software by the free Software Foundation. This implementation serves to define a de facto standard for PHP, as there is no formal specification.

On July 13, 2004, PHP 5 was released powered by the new Zend Engine ||.PHP 5 included new features such as:

**Feature of PHP**

Robust support for object-Oriented Programming

The PHP Data Objects extension, which defines a lightweight and consistent interface for accessing databases.

Performance enhancements taking advantage of the new engine.

Better support for MYSQL through a completely rewritten extension.

Embedded support for SQLite.

Integrated SOAP support.

Data integrators.

Error handling through exception.

**Server-side Scripting**

Originally designed to create dynamic web pages, PHP’s principal focus is server-side scripting. While running the PHP parser with a web server and web browser, the PHP model can be compared to other server-side scripting languages such as Microsoft’s ASP.NET system, Sun Microsystems’s Java Server Pages, mod pearl and the Ruby on Rails framework, as they all provide dynamic content to the client from a web server. To more directly compete with the “Framework” approach taken by these systems, Zend is working on the Zend Framework- emerging (as of Junen 2006) set of PHP building blocks and best practices; other PHP frameworks along the same lines include CakePHP, PRADO and Symfony.

The LAMP architecture has become popular in the web industry as a way of deploying inexpensive, reliable, scalable, secure web applications. PHP is commonly used as the in this bundle alongside Linux, Apache and MySQL. PHP can be used with a large number of relational database management systems, runs on all of the most popular web servers and is available for many different operating systems. This flexibility means that PHP has a wide installation base across the internet; over 19 million internet domains are currently hosted on servers PHP installed.

**MySQL**

**What is MySQL?**

MySQL is an open source relational database management system (RDBMS) that uses structured Query Language (SQL), the most popular language for adding, accessing and processing the data in database. Because it is open source, anyone can download MySQL and tailor it to their needs in accordance with the general public license. MySQL is noted mainly for its speed, reliability and flexibility.

### 

### Appendices

### Web References

[PHP] http://www.php.net

[UML] <http://www.rational.com>

[General] http://stackoverflow.com

<http://www.microsoft.com>

[Search]

<http://www.google.com>

<http://www.yahoo.com>

<http://www.msn.com>

<http://www.ask.com>

# **Books References**

* Learning PHP, MySQL and JavaScript: A step by step guide for creating dynamic websites

By Robin Nixon

* Object Oriented Modeling And Design

James Rumbaugh

Michael Blaha

* Database management system

By korth