**PROJECT REPORT**

(SEMESTER TRAINING)

## (KnolDevelopers.com)

Submitted by

**(Jaskirat Singh Sandhu)**

**Roll No.100903038**

Under the Guidance of

# Mr.Harsimranjeet Singh

(Director-Knol Software Solutions Pvt. Ltd.)

# Department Of Computer Science and Engineering

**THAPAR UNIVERSITY, PATIALA**

**(Deemed University)**

**Jan-May/Jun 2012**

**ANNEXURE – VIII**

# DECLARATION

I hereby declare that the project work entitled (“KnolDevelopers.com”) is an authentic record of my own work carried out at (Knol Software Solutions Pvt. Ltd.) as requirements of semester project term for the award of degree of B.E. (Computer Science & Engineering), Thapar University, Patiala, under the guidance of (Mr.Harsimranjeet Singh, Director-Knol Software Solutions Pvt. Ltd.), during 2 Jan to 16 June, 2012).

(Signature of student)

( Jaskirat Singh Sandhu )

(100903038)

Date: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**ACKNOWLEDGEMENT**

This project in itself is an acknowledgement to the inspiration, drive and technical assistance contributed to it by many individuals.

I extend our heartfelt thanks to **KNOL SOFTWARE SOLUTIONS PVT LTD. Patiala** for giving me an opportunity to work in such good professional environment and have a good experience databases.

I would like to extend my profound thanks to **Mr. Harsimranjeet Singh**, Project Manager and Director , for providing all the support in the lifetime of the project, his valuable motivation and time to time discussions which kept a constant vigil on the progress of the project.

**Contents**

1. **The Project Summary**
2. **The Programming Or Development Environment**
3. **Introduction To The Project**
4. **System Requirements Specifications**
5. **System Analysis**
6. **System Design**
7. **Implementation And Testing**
8. **Maintenance**
9. **Project Legacy And Future Scope**

**10. References**

**Introduction To The Project**

This project was basically aimed at providing the admin as well as the students with easy accessibility to all information regarding the ongoing developments in the knol community.

Each student has to get himself registered with the site providing all the information necessary such as the course he wants to join , where he is currently studying , other essentials and also his motivation to work for the project he has/she has chosen .This registration has to be approved by the administrator and once he is registered his profile is created and he becomes a member of the community.

This site is mainly about sharing of information and thus provides a simple way for the students to share notes and important documents and also to interact each other with messages and can also view the status of various projects and courses that are ongoing.

The admin can instantly view and change the status of any event or project that is scheduled to happen at the company .He can post new messages to different students and their batches and can also view what a particular student is working on at the moment and what all he has accomplished during his internship.

An important feature of this website is that the admin or the concerned authority can interact with the students through sms sending by way of smsgateway.

Also, the website issues login and password combinations to parents and college authorities of students so that they can review their progress as well as their attendance.

**Some Features Of Our Website**

* Secure and Authorized Access.
* Admin can fill the student admission form and print the form.
* Instant change of passwords.
* User can add, view and update student biodata.
* Admin can search the record of student.
* Admin can send messages via sms gateway.

**2. [ORGANISATIONAL PROFILE]**

**2.1 Introduction**

**2.2** **Vision**

**2.3** **Mission**

**2.4 Professional Services**

**2.5 Our Values**

### 2.1 INTRODUCTION

"As a software consultancy and R&D firm, offering top notch solutions to all business and customer needs delivering quality product/services. In the fields of software design and development, content management e-commerce solutions, web based applications using high end open source technologies."

SlickSoft Technologies R&D Pvt. Ltd. is committed to provide world class softwares in the region. With the time it has become a well known IT Development and Learning Center.

**2.2 VISION**

“To be the leaders in providing turnkey consulting and solutions implementing suitable high end technologies, focusing the customer business needs, dealing in best of breed product engineering life cycle solutions to the global software development and IT industry.”

“At SlickSoft Technologies R&D Pvt. Limited, we measure our amplitude with your success. Projecting us as forward looking company and helping to pave the way offering 360 degree wide spectrum solutions in the global market especially contributing to our Indian industry. Among the rising IT and R&D software services company combining trusted cutting edge technology with business acumen to deliver with quality , innovative web solutions , engineering consultancy and in the fields of software design and development ,content management solutions , e-commerce solutions and web application development also providing web hosting and domain . . “

**2.3 MISSON STATEMENT**

“To achieve our objective in an environment that fosters honesty in reliability, fairness and integrity and a single minded focus on the success of our customers and our enterprise .“

### *2.4 Professional Services*

* SMS Gateway and Bulk SMS Services
* Software & Web Development
* Software Testing
* Domains and Hosting
* Linux Administration
* AMC Services
* NLP services
* Embedded Software Soluions

***Society and Us***

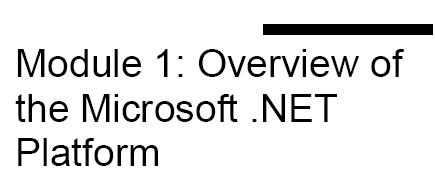
SlickSoft has enabled young and old global citizens to become familiar with the key tools of the emerging IT and Internet age. Our initiatives, aimed at narrowing the digital divide across the world, have helped people gain access to IT knowledge and enjoy the benefits that technology brings to ordinary lives.

### 1.5 Our Values

As a company, and as individuals, we value integrity, honesty, openness, personal excellence, constructive self-criticism, continual self-improvement, and mutual respect. We are committed to our clients & employees and have a passion for technology. We take on big challenges, and pride ourselves on seeing them through..Every successful corporation has a responsibility to use its resources and influence to make a positive impact on the world and its people. SlickSoft is focused on mobilizing our resources across the country, to create opportunities in the world, and to fulfill our commitment to serving the public good through innovative technologies. SlickSoft provides training that creates the social and economic opportunities that can transform company and help people realize their potential. We want to continually evolve our company to be in the best position to accelerate new technologies as they emerge and to better serve our students.

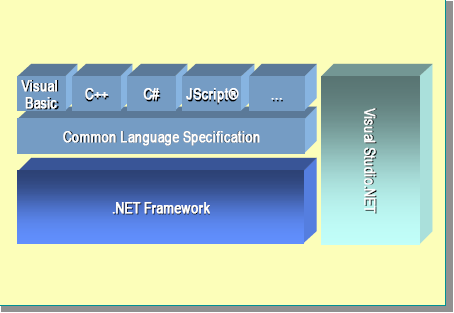
The Environment-

ASP.NET

****

# 

# 1.1- Overview of the Microsoft .NET Platform

****

Before COM, applications were completely separate entities with little or no integration. By using COM, you can integrate components within and across applications by exposing common interfaces. However, as a developer, you must still write the code to wrap, manage, and clean up after components and objects.

# 1.1.1 - Building Components in the .NET Framework

In the .NET Framework, components are built on a common foundation. You no longer need to write the code to allow objects to interact directly with each other. In addition, you no longer need to write component wrappers in the .NET environment, because components do not use wrappers. The .NET Framework can interpret the constructs that developers are accustomed to using in object oriented languages. The .NET Framework fully supports class, inheritance, methods, properties, events, polymorphism, constructors, and other object oriented constructs.

# 1.1.2 - The Common Language Specification

The Common Language Specification (CLS) defines the common standards to which languages and developers must adhere if they want their components and applications to be widely useable by other .NET languages.

# 1.2 - Benefits of the .NET Framework

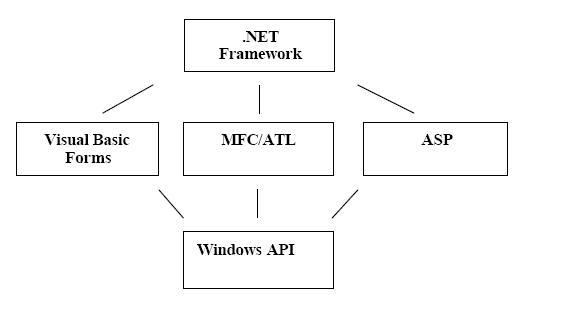
The NET Framework was designed to meet the following goals.

* **Based on Web standards and practices**

The .NET Framework fully supports the existing Internet technologies including Hypertext Markup Language (HTML), XML, SOAP, Extensible Stylesheet Language for Transformations (XSLT), Xpath, and other Web standards. The .NET Framework favors loosely connected, stateless Web services.

* **Designed using unified application models**

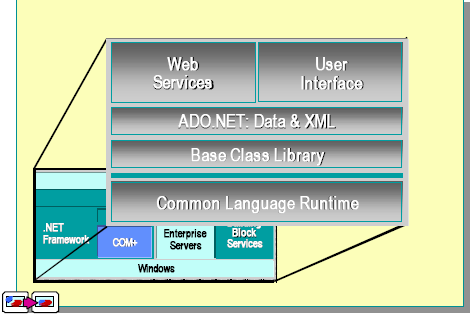
A .NET class’s functionality is available from any .NET language or programming model.

****

* **Easy for developers to use**

In the .NET Framework, code is organized into hierarchical namespaces and classes. The Framework provides a common type system, referred to as the unified type system, that is used by any .NET language. In the unified type system, all languages elements are objects. There are no variant types, there is only one string type, and all string data is Unicode. The unified type system is described in more detail in later modules.

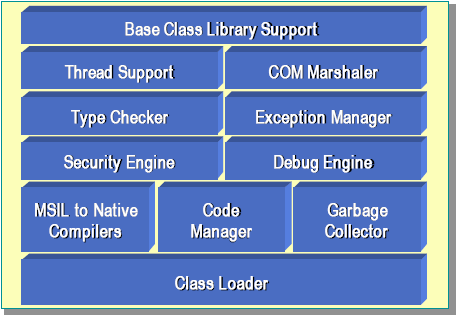
1.3 - The .NET Framework Components

****

The .NET Framework is a set of technologies that form an integral part of the Microsoft .NET platform. It provides the basic building blocks for developing Web applications and Web services. This section includes the following topics:

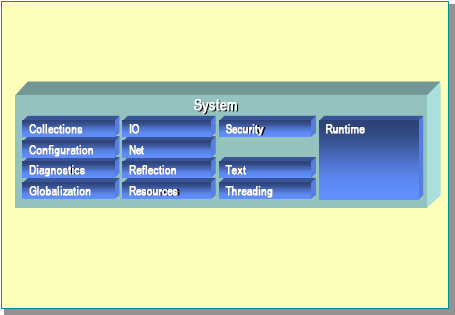
* Common Language Runtime
* Base Class Library
* ADO.NET: Data and XML
* Web Forms and Services
* User Interface

# 1.3.1 - Common Language Runtime

****

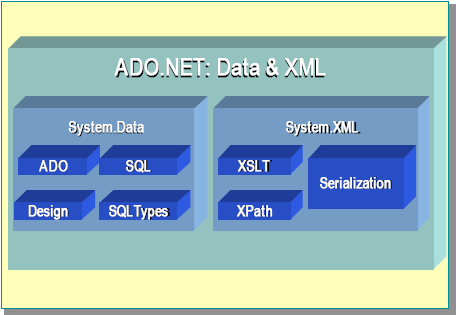
The Common Language Runtime simplifies application development, provides a robust and secure execution environment, supports multiple languages, and simplifies application deployment and management. The environment is also referred to as a managed environment, one in which common services, such as garbage collection and security, are automatically provided.

# 1.3.2 - Base Class Library

****

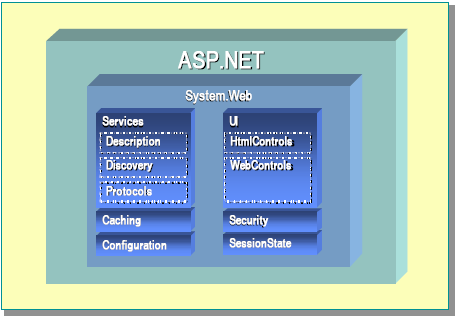
The Base Class Library (BCL) exposes features of the runtime and provides other high-level services that every programmer needs through namespaces. For example, the **System.IO** namespace contains input/output (I/O) services. In the **System.IO** namespace, all of the base data types, such as **int** and **float**, are defined for the platform. Inside the **System.IO** namespace, there are other namespaces that provide various runtime features. The **Collections** namespace provides sorted lists, hash tables, and other ways to group data. The **IO** namespace provides file I/O, streams, and so on. The **Net** namespace provides Transmission Control Protoc ol/Internet Protocol (TCP/IP) and sockets support. For more information about namespaces, search for “namespaces” in the .NET Framework SDK Help documents.

# 1.3.3 - ADO.NET: Data and XML

****

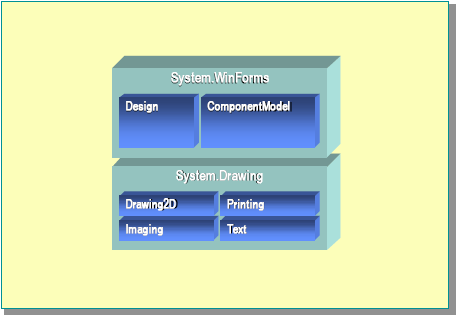
ADO.NET is the next generation of ActiveX® Data Object (ADO) technology. ADO.NET provides improved support for the disconnected programming model. It also provides rich XML support.The **System.Data** namespace consists of classes that constitute the ADO.NET object model. At a high level, the ADO.NET object model is divided into two layers: the connected layer and the disconnected layer. The **System.Data** namespace includes the **DataSet** class, which represents multiple tables and their relations. These **DataSets** are completely self contained data structures that can be populated from a variety of data sources. One data source could be XML, another could be OLEDB, and a third data source could be the direct adapter for SQL Server. The **System.Xml** namespace provides support for XML. It includes an XML parser and a writer, which are both W3C-compliant. The Extensible Stylesheet Language (XSL) transformation is provided by the **XSLT** namespace. The implementation of XPath allows data graph navigation in XML. The **Serialization** namespace provides the entire core infrastructure for Web Services, including features such as moving back and forth from objects to an XML representation.

1.3.4 - Web Forms and Services

****

Microsoft ASP.NET is a programming framework built on the Common Language Runtime that can be used on a server to build powerful Web Applications. ASP.NET Web Forms provide an easy and powerful way to build dynamic Web user interfaces (UIs). ASP.NET Web Services provide the building blocks for constructing distributed Web-based applications. Web Services are based on open Internet standards, such as HTTP and XML. The Common Language Runtime provides built-in support for creating and exposing Web Services by using a programming abstraction that is consistent and familiar to both ASP Web Forms and Visual Basic developers. The resulting model is both scalable and extensible. This model is based on open Internet standards (HTTP, XML, SOAP, SDL) so that it can be accessed and interpreted by any client or Internet-enabled device

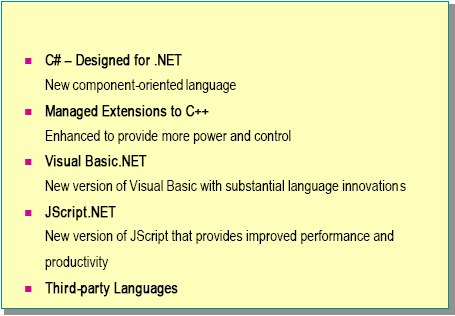
# 1.3.5 - User Interface for Windows

****

You can use the **System.WinForms** classes to build the client user interface (UI). This class lets you implement the standards Windows UI in your .NET applications.

You can use the **System.Drawing** class to access the new GDI+ features. This class provides support for the next generation of Graphics Device Interface (GDI) two-dimensional graphics. It also provides native support for Graphics Interchange Format (GIF), Tagged Image File Format (TIFF), and other formats.

# 1.4 - Languages in the .NET Framework

****

The .NET Framework provides support for several programming languages. C# is the programming language specifically designed for the .NET platform, but C++ and Visual Basic have also been upgraded to fully support the .NET Framework.

**Abstract**

The project "Knol Developers" is a web application to automate and log on to various day-to-day activities and to provide instantaneous information required in effective management of any given institute. The automation part of the software would involve data capture and maintenance of details on the institute, teachers, students and tasks in the overall management of the institute.

Institute is the place to learn about new things and to build his carrier. So to maintain the institute lot of things are needed. In the existing system all the data is maintained in the records manually. 

Even more number of staff is required to maintain all the records. The manual maintenance may lead to many faults, so there should be a automated system to over come the defects of the existing system. 

In the proposed system we should develop an automated system that over come the drawbacks of the existing system. In this system all t he management is done on the system. So the time taken to perform a task is reduced, there is no need of maintaining large books and no need of any place to store them.

**Modules:**

The project “knol institute management” can be viewed as three modules:

1) Adding record’s module

2) Updating and deleting module

3) Searching and printing module

4) Messaging and viewing module

**User Adding Detail Module:**

Before entering to user module user have to write two things 1st user id and 2nd password. If both information are in correct order an user account will appear. You right side of page at top the login type. A user can do the following things.

* User can also change own password by clicking the password change button. It has to write three things (old password, new password and confirm the new password) if all are in correct order your password will change.
* User can provide access to another person by providing password and user id.
* User can add student detail, personal detail and course detail.
* 1st User has to add himself general detail then click next button add course detail then click next button add personal detail then finally click submit button .
* User can upload the snap.
* User can manage the account.

**Administrator Update And Delete Detail Module:**

* Admin can update the detail of student firstly admin has to search the record .they can search according to course , reg no, name when the record is found they can click over the record and then click update button separate window will appear and you can edit the detail and click submitted button for storing update data.
* Admin can delete the detail of student firstly admin has to search the record .they can search according to course , reg no, name when the record is found they can click over the record and then click delete button for deleting record.

**Administrator Search And Print Detail Module:**

* Admin can print the detail of student firstly admin has to search the record .they can search according to course , reg no, name when the record is found they can click over the record and then click print button separate window will appear and if you have printer attach with your computer you can command your printer and the form will printed

**Authority or parent search student Module:**

The college authority and the parents can search for a particular student using their id and can then view their attendance and performance.

**Administrator messaging students module:**

* Admin can also notify the students regarding any important information or any updates in the courses. He can also notify them about their result and attendance information by using the pre-prepared message templates.

**REQUIREMENT ANALYSIS**

* **Hardware Requirements**

|  |  |
| --- | --- |
| **Number** | **Description** |
| 1 | Pentium 4 ,WIN xp/7 |
| 2 | 256 MB RAM |
| 3 | HARD DISK DRIVE: 20 GB |
| 4 | CD ROM: 42X |
| 5 | Floppy Disk Drive: 1.44” |
| 6 | Monitor: 15” color |
| 7 | Keyboard: Standard 110 keys |
| 8 | Mouse: Scroll |
| 9 | Network Adapter: Internal Ethernet Card |

* **Software Requirements**

|  |  |
| --- | --- |
| **Number** | **Description** |
| 1 | Windows XP –SP2 |
| 2 | Visual studio 2008,telerik rad ajax controls |
| 3 | SQL 2005 |

**Introduction TO Tools**

**SOFTWARE REQUIREMENTS**

The lists of the software used currently for the project are:

* **Platform:** Microsoft Windows 7
* **Front End:**

Development:

* + Environment:- **Microsoft .Net Framework**
  + Application:- **ASP.Net 3.5**
  + **Back End**

Database

* + SQL **2008**

Connection

* + Sql Connection

**PROBLEM ANALYSIS**

**Applications**

The main applications of the Knol Institute Management is the ability of the website to properly show and manage information about students courses offered by institute .The administrator has the ability to change ,modify, view and delete the various details regarding the users and arts. The users have the ability to log in and post their queries and download arts.

**Challenges**

The challenges mainly lie in detecting attacks like viruses, hacking and also in the implementation of firewall. A virus can enter the system and can disrupt the working of the website. Hacking can be done by some people who want to access some restricted sections of the website (e.g. administrator’s area) and to modify or taper some aspects of the website.

Scanning attacks may yield:

(i) The method used by viruses to enter the system.

(ii) The types of database allowed through a firewall.

(iii) The paths or ways used by hackers to enter the system

(iv) The loopholes remaining in the system (or website) which are used by attackers.

(v) The server from where the viruses or hackers are gaining access to the system.

(vi) The types of viruses able to affect the website.

And with the implementation of firewall and other security mechanisms that are designed for it, the On Line Placement System Website safe and secure.

**REQUIREMENT ANALYSIS**

**Goal of Thesis**

The goal of our thesis is to develop a website that can be used as an institute management application with the features of keeping and manage the records of students. The whole project will be based on CSharp.Net with SQL as the database with certain security constraints added to it.

Our aim is also to implement the Administrator part in to the project so that the server or administrator himself can view, add, delete and modify.

1. **Administrator:**

Administrator is owner of the institute or the authentic person who manage the record of student regarding the courses they pursuing, regarding their fee structure and teacher allotted to them to their respective course and time slot allotted to them.

He has to see whether the website is working properly and whether the details available in the system are relevant and correct. He can view, add, modify, delete details.

1. **Database**

The database keeps all the records of all the users i.e. name, course, phone no., dob, city, country, etc. For creating such records it takes the help of tables which is created in the SQL. The tables can have entries of all the registered students as entered by administrators.

1. **Clients**

Our aim will also to provide efficient way through menu driven by which students can enter to see various queries, ask questions to their respective teacher and play games.

1. **Security Constraints**

There need to be certain constraints which have to be implemented on the database as well as on the administrator in order to work properly the whole system, such as declaring the primary key, or such constraints in order to keep the database work properly.

**FEASIBILTY STUDY**

The software is said to have life cycle composed of several phases.

At the feasibility stage, it is desirable that two or three different configuration will be pursed that satisfy the key technical requirement but which represent different level of ambition and cost.

Feasibility is the determination of whether or not a project is worth doing. A feasibility study is carried out select a best system that mate performance requirements.

The data collected during primary investigation examines system feasibilities that is likelihood that the system will be beneficial to the organization. Four tests for feasibility study are as follows:-

* **Technical Feasibility:** This is concerned with specifying equipment and software that w ill successfully satisfy the use considerably, but might include
  + The feasibility to produce output in a given time because system is fast enough to handle multiple users.
  + Response time under certain circumstances and ability to process a certain volume of transaction of a particular speed.
  + Feasibility to communicate data to distant location.
* **Economical Feasibility:** Economic analysis is the most frequently used technique used for evaluating the effectiveness of a proposed system. More commonly known as cost/benefit analysis the procedure is to determine the benefits and savings that are expected from a proposed system and compared them with cost. Though the cost of installing the system may appear high, it is one time investment. The resulting benefits is that automation results in turnaround time. The resulting cost/benefit ratio is favorable.
* **Operational Feasibility:** It is mainly related to human organizational as social aspects. The points to be considered are - The system interface is standard, user friendly and provides extensive help. Hence no special training is not required.
* **Social Feasibility:** Social feasibility is determination of whether a proposed project will be acceptable to people or not, So this project is totally Social and Feasible

**DESIGN PHASE**

The design phase involves converting the informational, functional, and network requirements identified during the initiation and planning phases into unified design specifications that developers use to script programs during the development phase. Program designs are constructed in various ways. Using a top-down approach, designers first identify and link major program components and interfaces, then expand design layouts as they identify and link smaller subsystems and connections. Using a bottom-up approach, designers first identify and link minor program components and interfaces, then expand design layouts as they identify and link larger systems and connections.  
Contemporary design techniques often use prototyping tools that build mock-up designs of items such as application screens, database layouts, and system architectures. End users, designers, developers, database managers, and network administrators should review and refine the prototyped designs in an iterative process until they agree on an acceptable design.

Designers should carefully document completed designs. Detailed documentation enhances a programmer’s ability to develop programs and modify them after they are placed in production. The documentation also helps management ensure final programs are consistent with original goals and specifications. Organizations should create initial testing, conversion, implementation, and training plans during the design phase. Additionally, they should draft user, operator, and maintenance manuals.

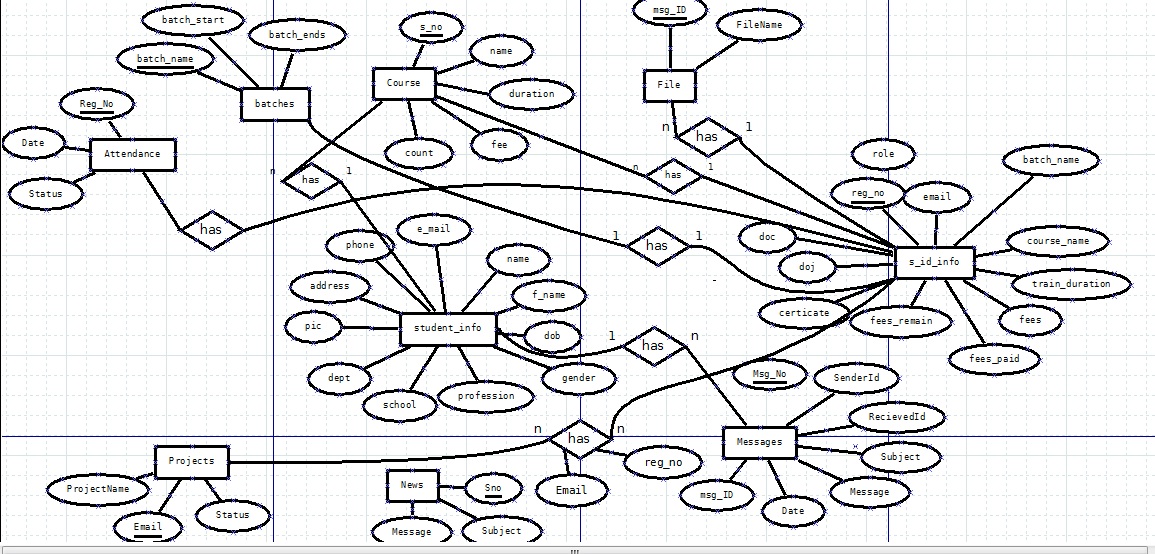
For design of the website project:

First Database has to be designed which can be used to handle all the requirements of the users.

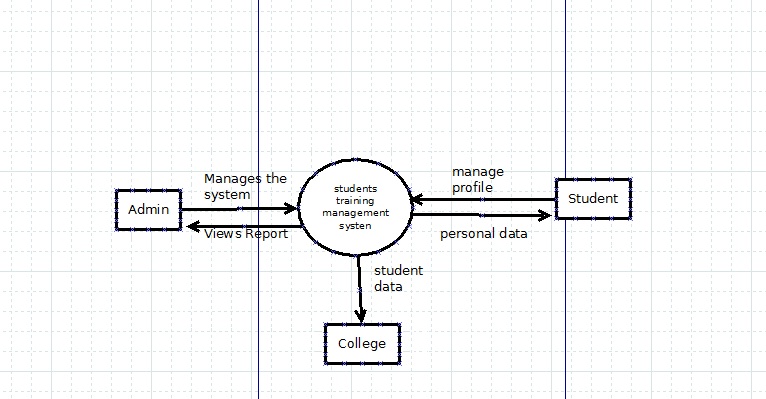
The basic structure of the website has to be designed.

The main template to be used for the website is designed.

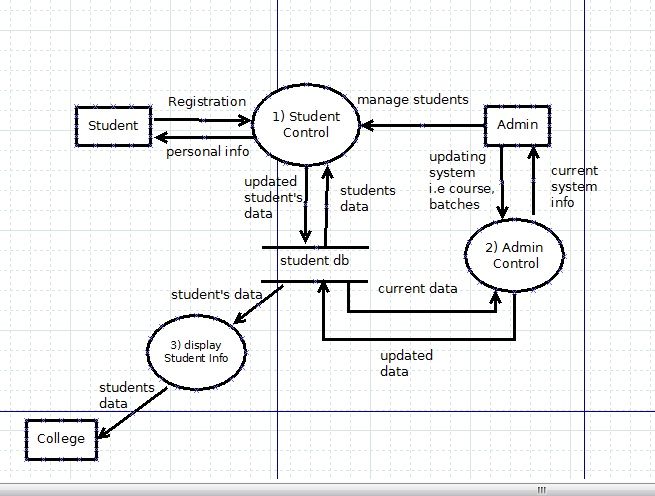
**ER Diagram**

****

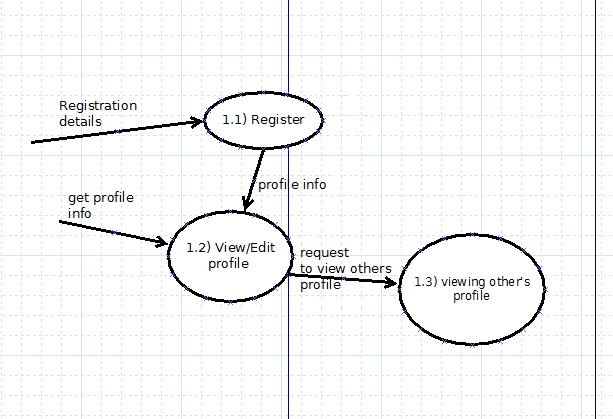
**DFD LEVEL 0**

****

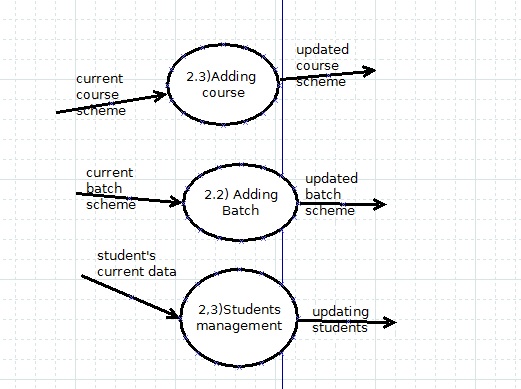
**DFD LEVEL 1**

****

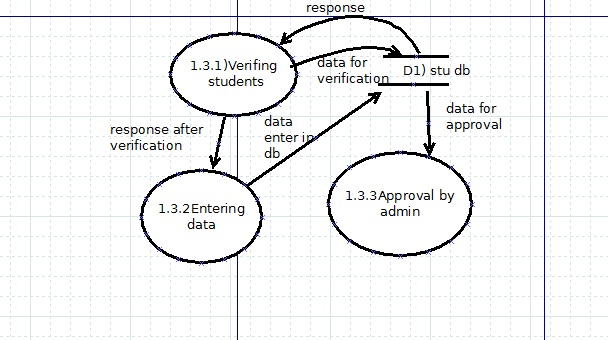
**DFD LEVEL 2**

****

**DFD LEVEL 2**

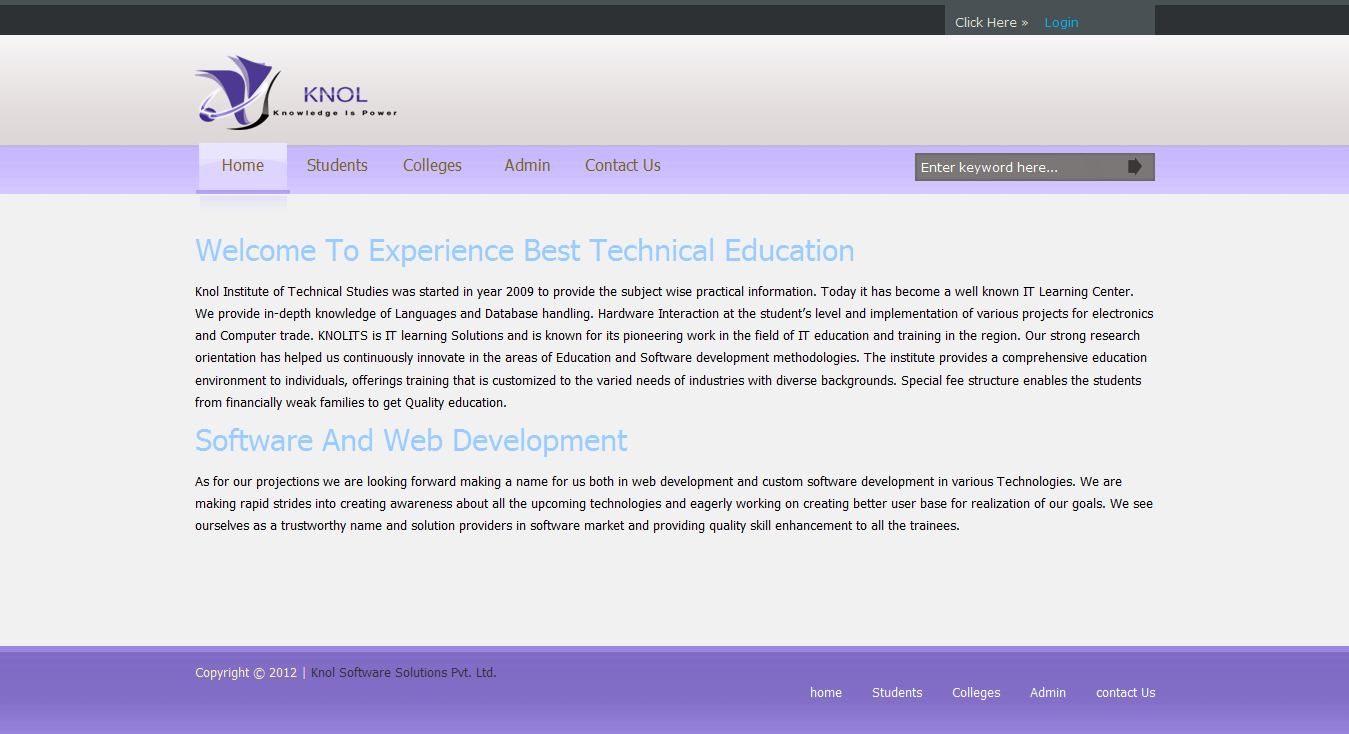
****

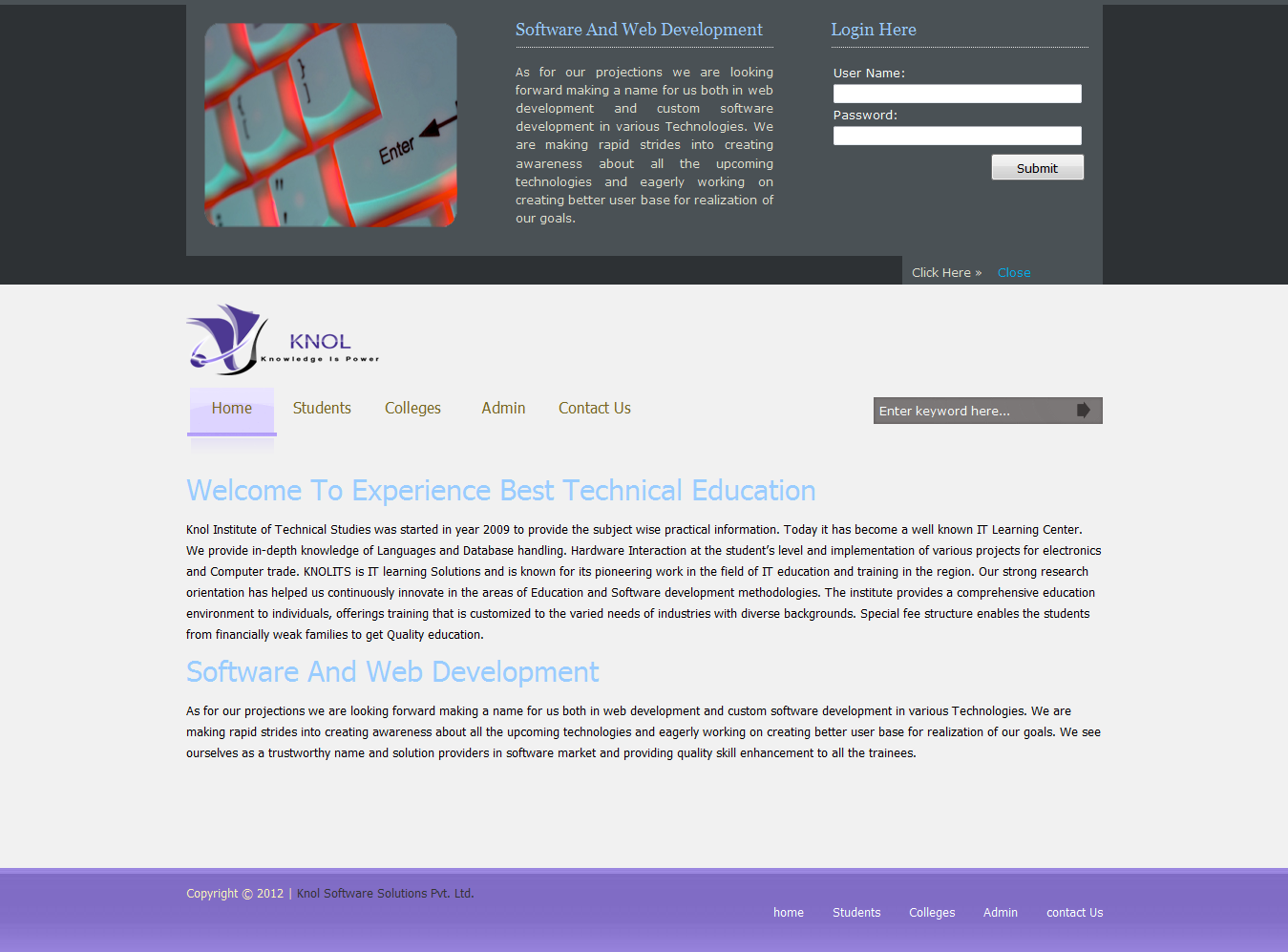
**DFD LEVEL 3**

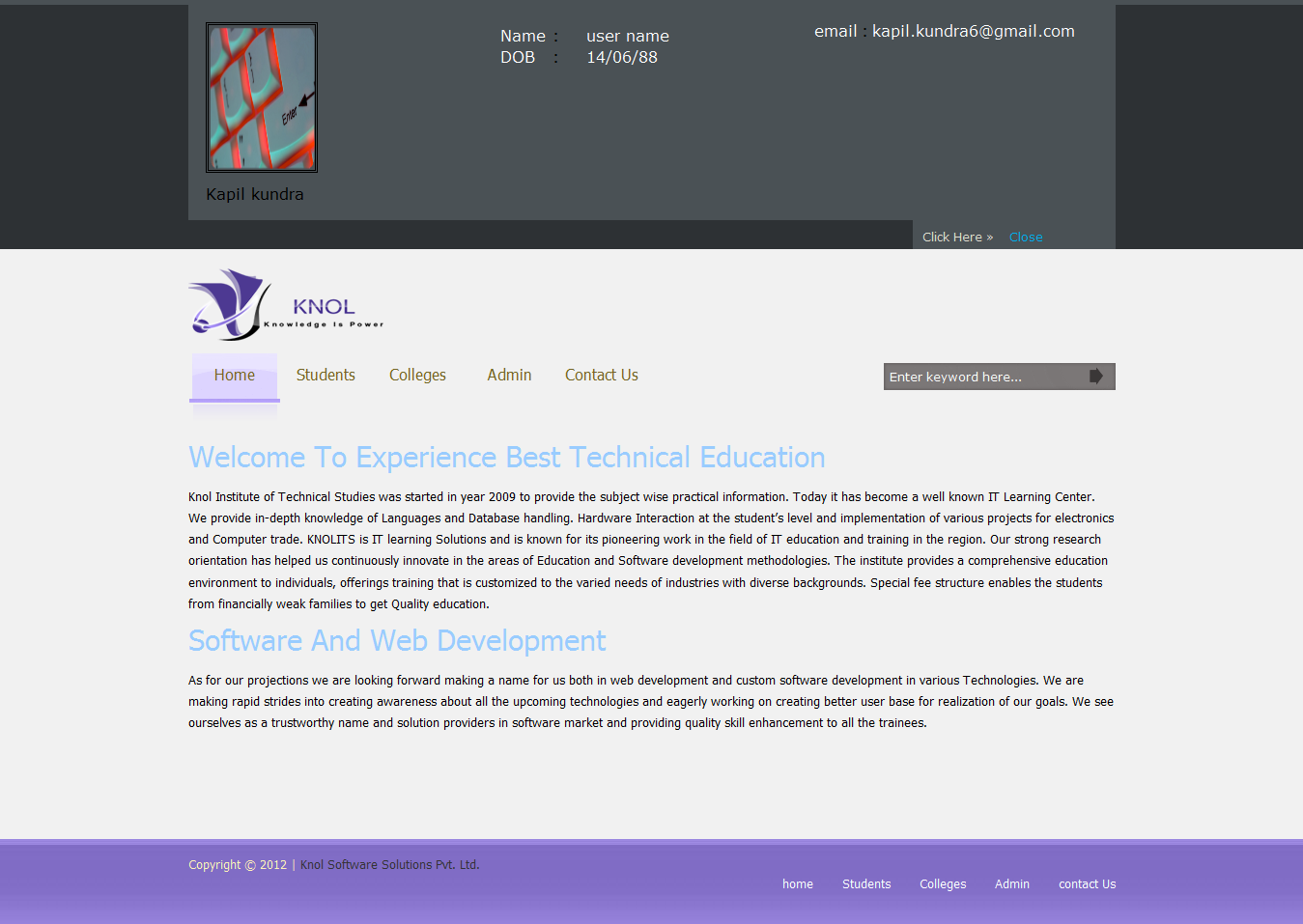
****

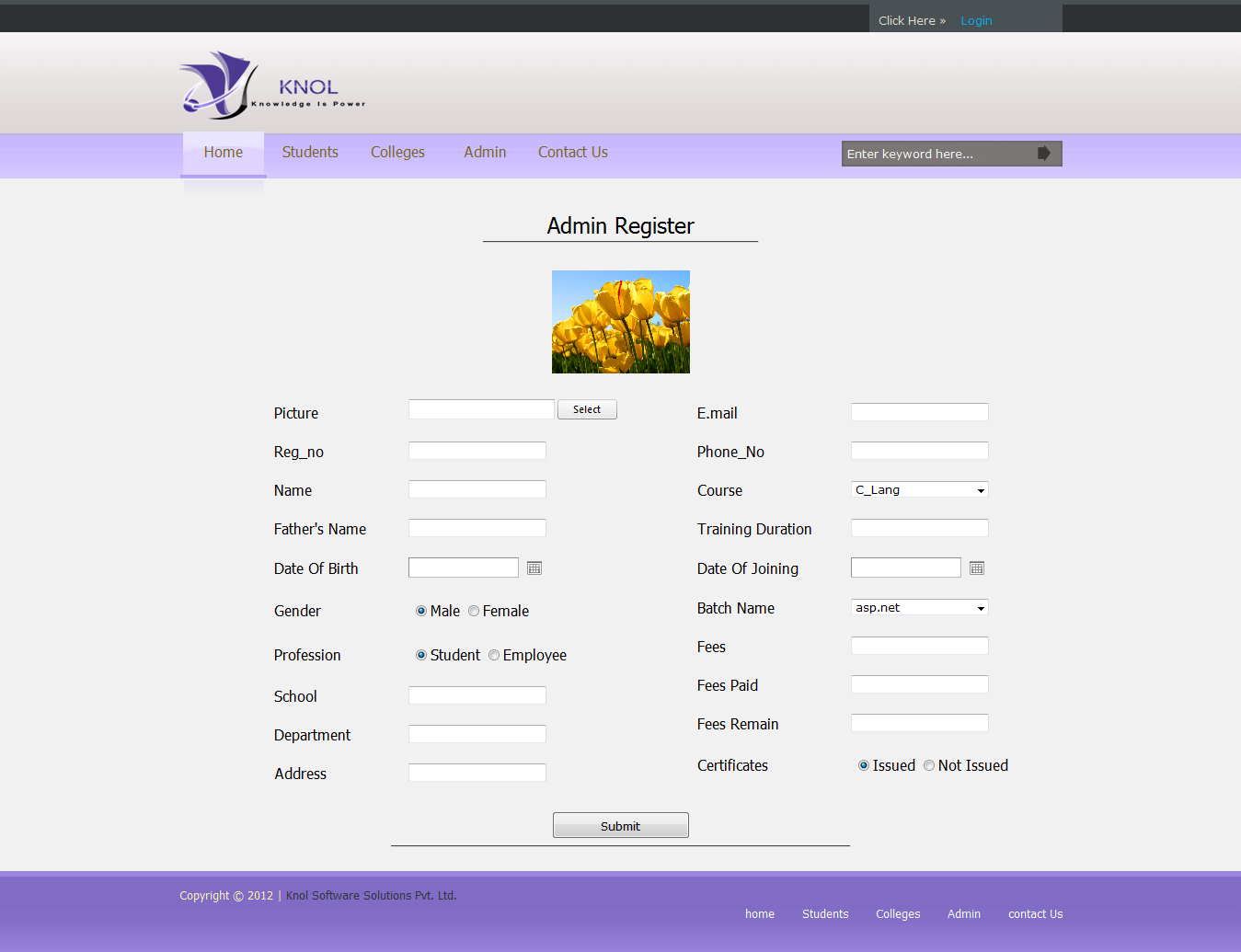
**IMPLEMENTATION**

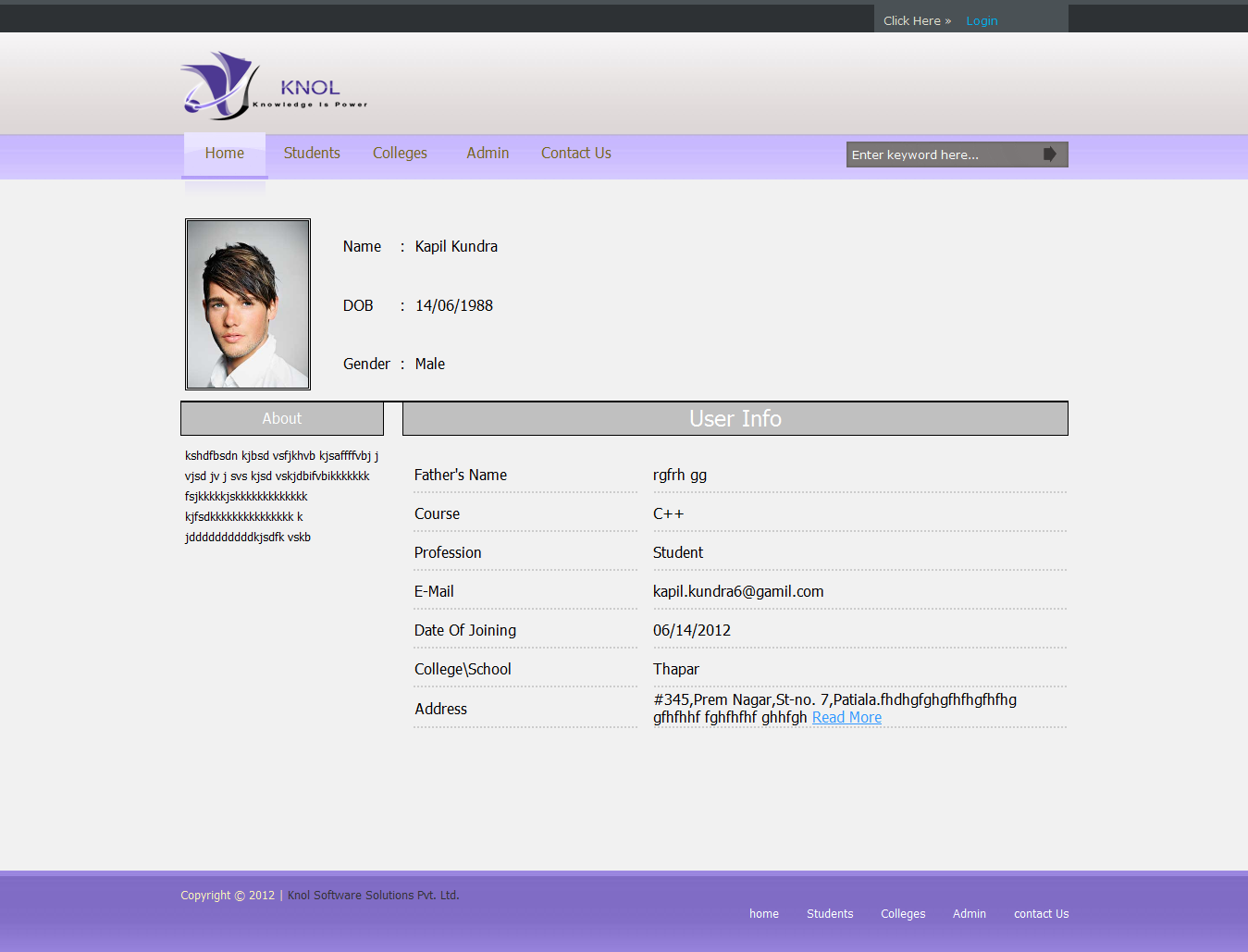
**HOME SCREEN**

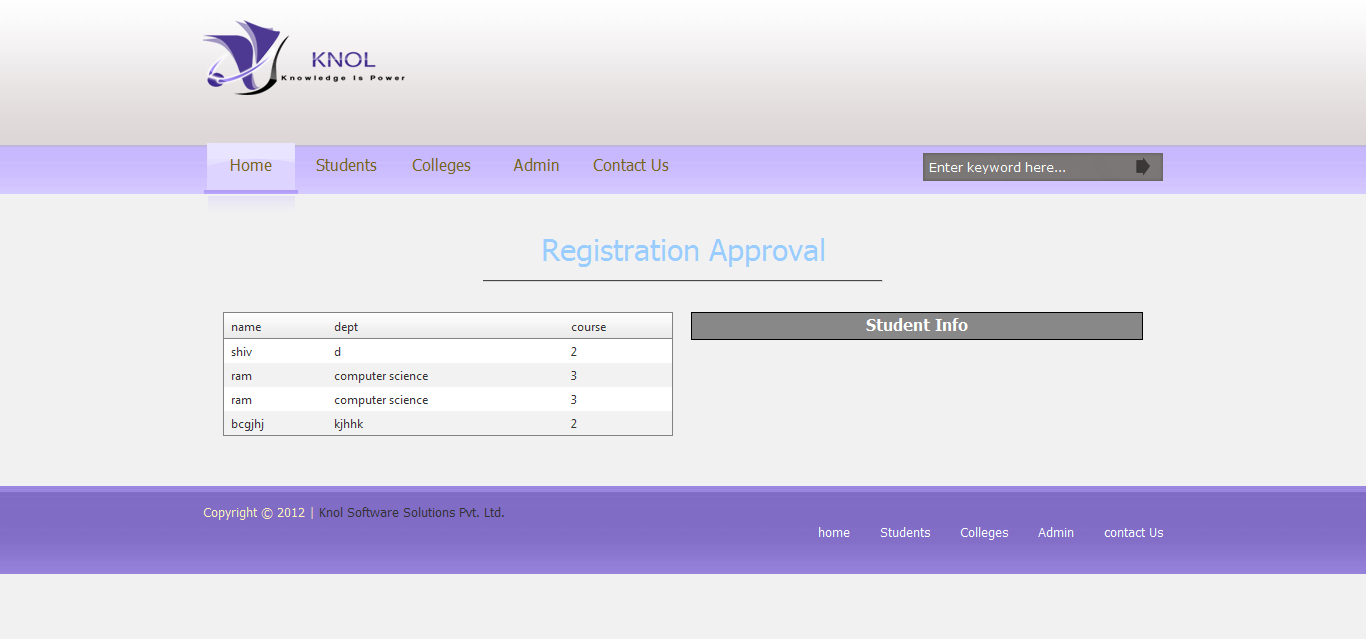
****

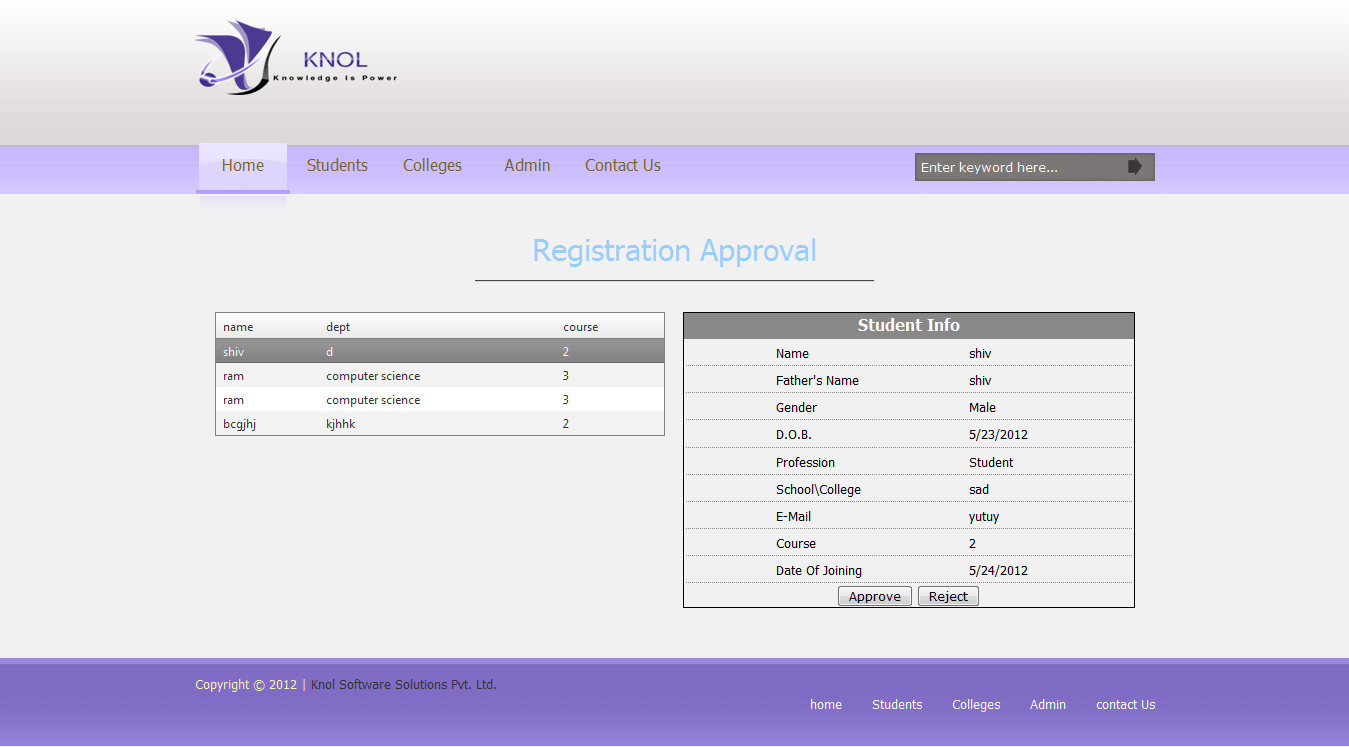
****

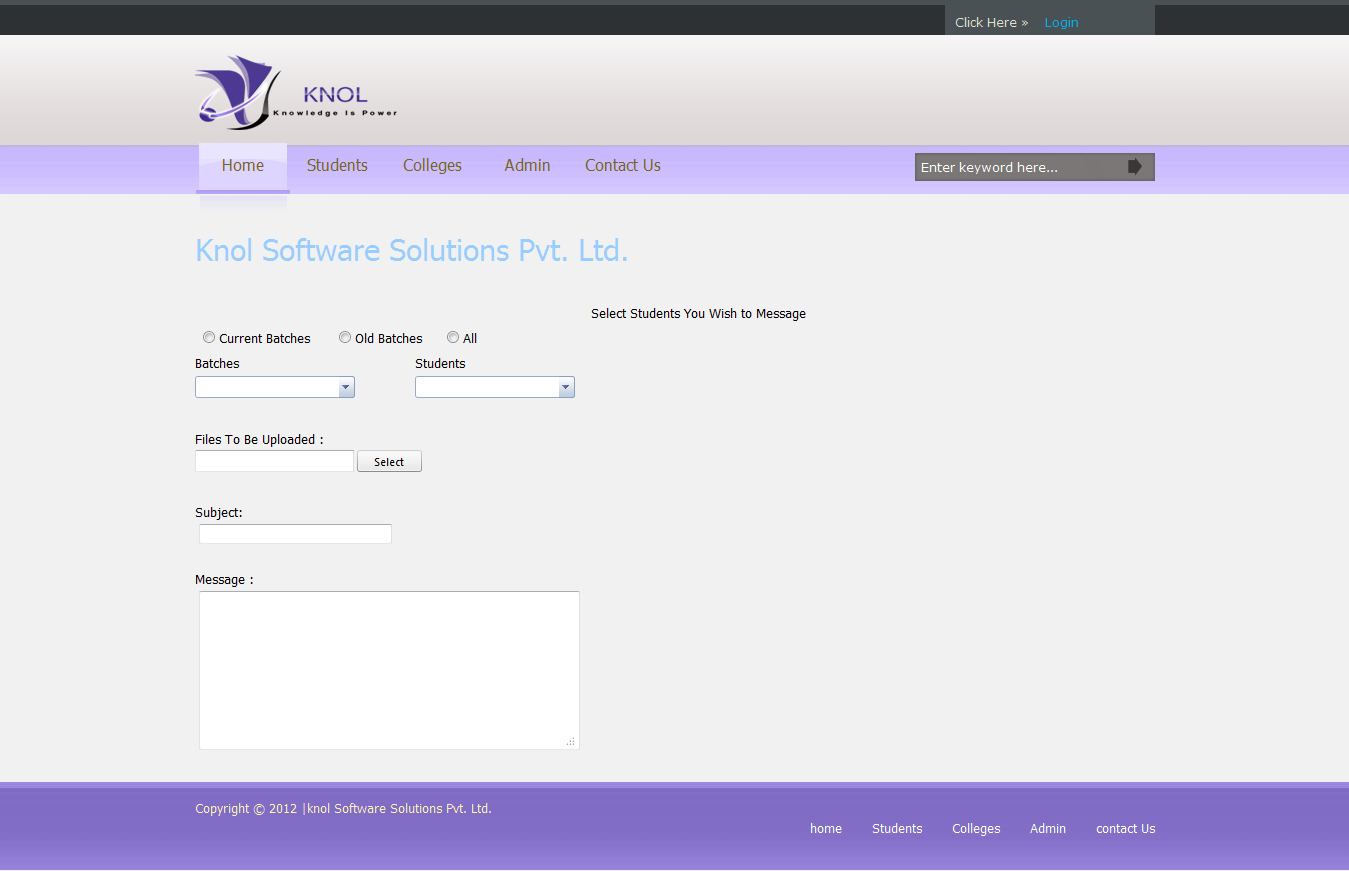
****

****

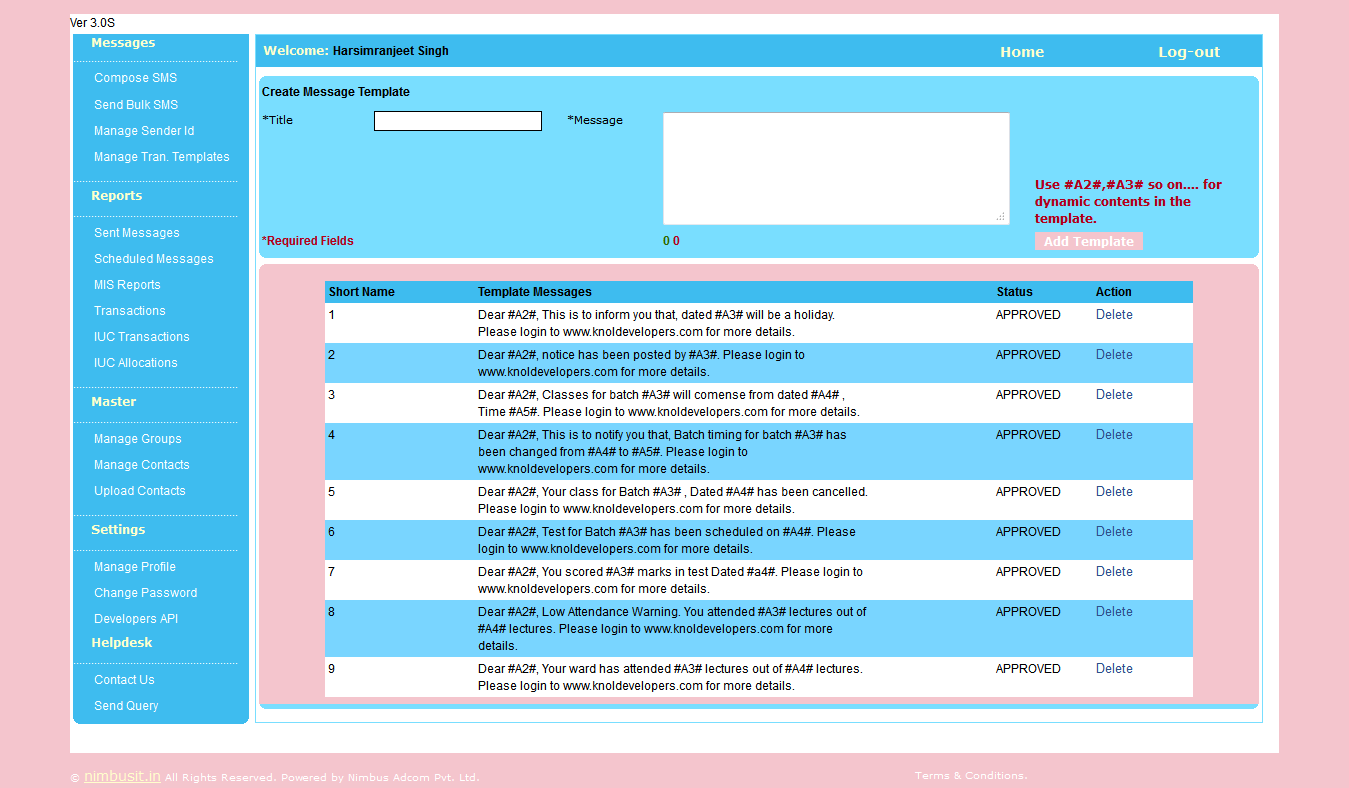
****

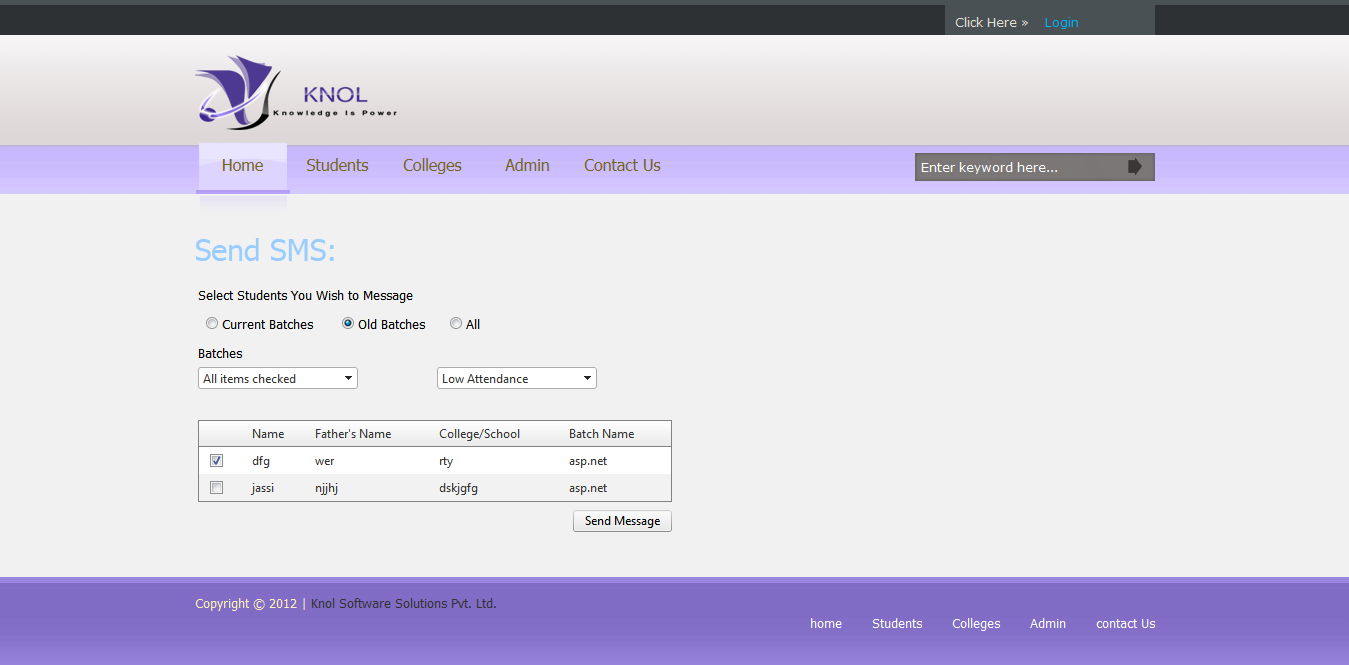
****

****

****

****

****

****

**Components to be tested:**

**REGISTRATION:**

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **TEST CASE** | **USER INPUT** | **PASS CRITERIA** |
| **U\_R\_1** | **User Registration** | **User selects already**  **existing email** | **Display message to**  **choose different email** |
| **U\_R\_2** | **User Registration** | **User enters different**  **password in password**  **confirm field** | **Display message that**  **Password and Confirm**  **Password fields don't**  **Match** |
| **U\_R\_3** | **User Registration** | **User forgets to enter a**  **particular required fields** | **Display message The**  **value in field is**  **required** |
| **U\_R\_4** | **User Registration** | **User enters all the details**  **Successfully** | **User account created** |

**LOGIN:**

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **TEST CASE** | **USER INPUT** | **PASS CRITERIA** |
| **U\_L\_1** | **User Login** | **User enters a wrong**  **email** | **Display message Login**  **or Password is**  **incorrect** |
| **U\_L\_2** | **User Login** | **User enters a wrong**  **password** | **Display message Login**  **or Password is**  **incorrect.** |
| **U\_L\_3** | **User Login** | **User enters correct**  **Email and password** | **User logs in**  **successfully** |

**Consult DETAILS:**

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **TEST CASE** | **USER INPUT** | **PASS CRITERIA** |
| **U\_ADD\_1** | **Leave query or leave reply** | **User opens text box and submit text** | **User has to first logon to the system**  **And data added to database** |
| **U\_ADD\_2** | **View Query or view reply** | **Registered clicks the view button related to subsequent query corresponding** | **The item details are added to the database** |

**Message DETAILS**

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **TEST CASE** | **USER INPUT** | **PASS CRITERIA** |
| **U\_ADD\_1** | **Leave message** | **User opens text box and submit text** | **User has to first logon to the system**  **And data added to database** |
| **U\_ADD\_2** | **Reply message** | **Registered clicks the view button related to subsequent message and add text to text box corresponding** | **The item details are added to the database** |

# 

# TESTING

**Software testing**

Software testing is the process used to measure the [quality](http://en.wikipedia.org/wiki/Software_quality) of developed [computer software](http://en.wikipedia.org/wiki/Computer_software). Usually, quality is constrained to such topics as [correctness](http://en.wikipedia.org/wiki/Correctness), completeness, [security](http://en.wikipedia.org/wiki/Computer_security_audit), but can also include more technical requirements as described under the [ISO](http://en.wikipedia.org/wiki/International_Organization_for_Standardization) standard [ISO 9126](http://en.wikipedia.org/wiki/ISO_9126), such as capability, [reliability](http://en.wikipedia.org/wiki/Reliability), [efficiency](http://en.wikipedia.org/wiki/Algorithmic_efficiency), [portability](http://en.wikipedia.org/wiki/Porting), [maintainability](http://en.wikipedia.org/wiki/Maintainability), compatibility, and [usability](http://en.wikipedia.org/wiki/Usability). Testing is a process of technical investigation, performed on behalf of stakeholders, that is intended to reveal quality-related information about the product with respect to the context in which it is intended to operate.

**White box, black box, and grey box testing**

[White box](http://en.wikipedia.org/wiki/White_box_testing) and [black box testing](http://en.wikipedia.org/wiki/Black_box_testing) are terms used to describe the point of view that a test engineer takes when designing test cases. **Black box** testing treats the software as a black-box without any understanding as to how the internals behave. Thus, the tester inputs data and only sees the output from the test object. This level of testing usually requires thorough test cases to be provided to the tester who then can simply verify that for a given input, the output value (or behavior), is the same as the expected value specified in the test case.

**White box** testing, however, is when the tester has access to the internal data structures, code, and algorithms. For this reason, [unit testing](http://en.wikipedia.org/wiki/Unit_testing) and debugging can be classified as white-box testing and it usually requires writing code, or at a minimum, stepping through it, and thus requires more skill than the black-box tester. If the software in test is an interface or API of any sort, white-box testing is almost always required.

**Grey box** testing could be used in the context of testing a client-server environment when the tester has control over the input, inspects the value in a SQL database, and the output value, and then compares all three (the input, sql value, and output), to determine if the data got corrupt on the database insertion or retrieval.

**Verification and Validation**

Software testing is used in association with [verification and validation](http://en.wikipedia.org/wiki/Verification_and_Validation) (V&V). *Verification* is the checking of or testing of items, including software, for conformance and consistency with an associated specification. Software testing is just one kind of verification, which also uses techniques such as reviews, inspections, and walkthroughs. *Validation* is the process of checking what has been specified is what the user actually wanted.

* Verification: Have we built the software right? (i.e. does it match the specification).
* Validation: Have we built the right software? (i.e. Is this what the customer wants?)

**Level of testing**

* [**Unit testing**](http://en.wikipedia.org/wiki/Unit_testing) tests the minimal software component, or module. Each unit (basic component) of the software is tested to verify that the detailed design for the unit has been correctly implemented. In an Object-oriented environment, this is usually at the class level, and the minimal unit tests include the constructors and destructors.
* [**Integration testing**](http://en.wikipedia.org/wiki/Integration_testing) exposes defects in the interfaces and interaction between integrated components (modules). Progressively larger groups of tested software components corresponding to elements of the architectural design are integrated and tested until the software works as a system.
* [**Functional testing**](http://en.wikipedia.org/wiki/Functional_testing) tests at any level (class, module, interface, or system) for proper functionality as defined in the specification.
* [**System testing**](http://en.wikipedia.org/wiki/System_testing) tests a completely integrated system to verify that it meets its requirements.
* [**System integration testing**](http://en.wikipedia.org/wiki/System_integration_testing)verifies that a system is integrated to any external or third party systems defined in the system requirements.
* [**Acceptance testing**](http://en.wikipedia.org/wiki/Acceptance_testing)can be conducted by the end-user, customer, or client to validate whether or not to accept the product. Acceptance testing may be performed as part of the hand-off process between any two phases of development.
  + **Alpha testing** is simulated or actual operational testing by potential users/customers or an independent test team at the developers' site. Alpha testing is often employed for off-the-shelf software as a form of internal acceptance testing, before the software goes to beta testing.

**Beta testing** comes after alpha testing. Versions of the software, known as [beta versions](http://en.wikipedia.org/wiki/Beta_version), are released to a limited audience outside of the company. The software is released to groups of people so that further testing can ensure the product has few faults or [bugs](http://en.wikipedia.org/wiki/Computer_bug). Sometimes, beta versions are made available to the open public to increase the [feedback](http://en.wikipedia.org/wiki/Feedback) field to a maximal number of future users.

**TESTING PHASE:**

Testing is the process of examining an application to ensure it fulfills the requirements for which it was designed and meets quality expectations. More importantly, testing ensures the application meets customer expectations.

Testing accomplishes a variety of things, but most importantly it measures the quality of the software you are developing. This view presupposes there are defects in your software waiting to be discovered and this view is rarely disproved or even disputed.

Several factors contribute to the importance of making testing a high priority of any software development effort. These include:

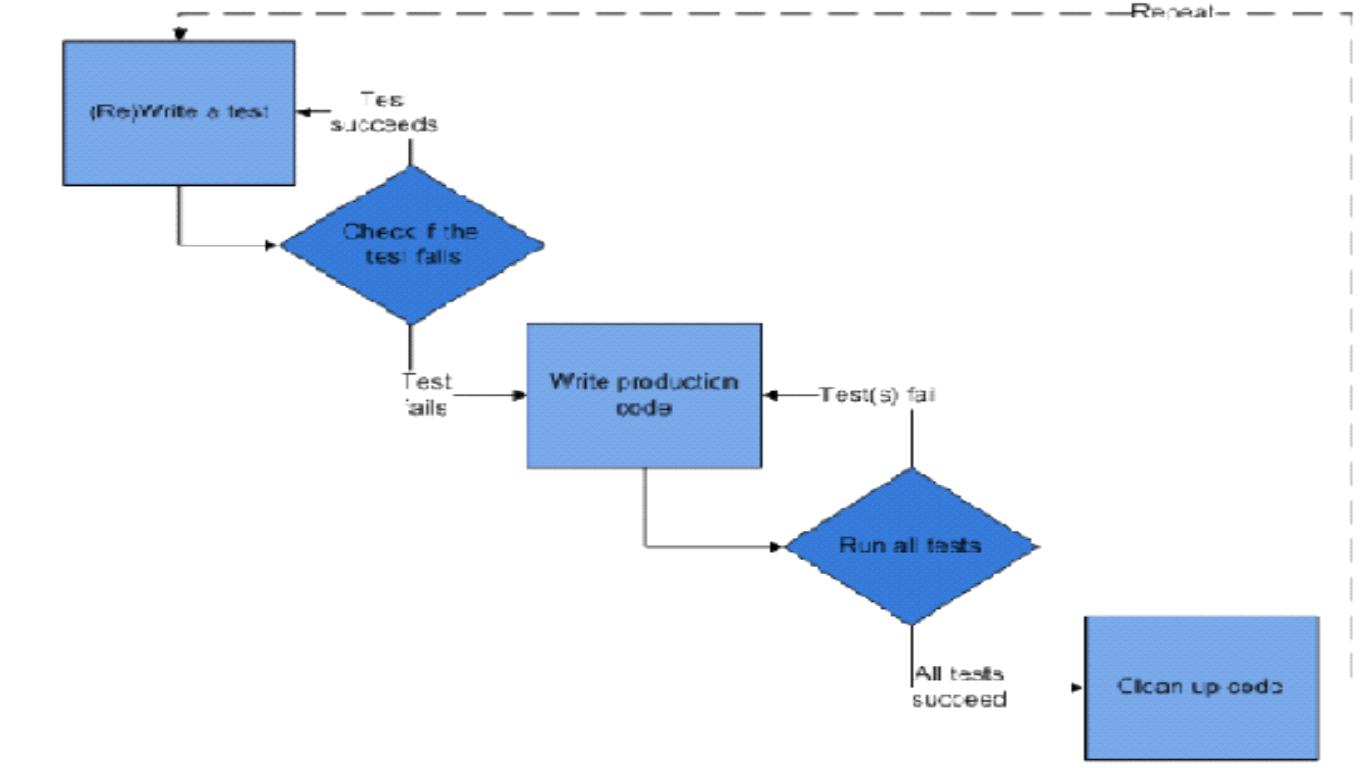
* Reducing the cost of developing the program. Minimal savings that might occur in the early stages of the development cycle by delaying testing efforts are almost certainly bound to increase development costs later. Common estimates indicate that a problem that goes undetected and unfixed until a program is actually in operation can be 40 – 100 times more expensive to resolve than resolving the problem early in the development cycle.
* Ensuring that your application behaves exactly as you explain to the user. For the vast majority of programs, unpredictability is the least desirable consequence of using an application.
* Reducing the total cost of ownership. By providing software that looks and behaves as shown in your documentation, your customers require fewer hours of training and less support from product experts.
* Developing customer loyalty and word-of-mouth market share. Finding success with a program that offers the kind of quality that only thorough testing can provide is much easier than trying to build a customer base on buggy and defect-riddled code.

**Test Driven Development Software Methodology :**

It is a software development practice in which a developer writes unit tests before writing code for a module. It is carried out in a number of iterations. A developer writes tests first. Then he implements the functionality by writing the code. The code should pass the tests. If it doesn’t, then the developer refactors the code or makes changes in the test case. This goes on till the code passes all unit tests. This helps a developer focus more on the behavior and interface of software and each of its modules. In other words, the developer first defines the classes and the class interfaces for the application. After the interfaces are defined, he writes tests for each class to verify the behavior of class methods. The next step involves writing the

The unit tests should be simple,easy to understand and test a small piece of code. The implemented code should pass the unit test. Relevant refactoring in implementation or test case is done during iteration. All tests should pass before you add a new piece of code to an existing code.

TDD is used with a version control system so that if the code fails the unit tests unexpectedly, it’s always best to revert back to the previous version, that had passed all tests, rather than debugging and trying to find the bugs.



Test driven development

**Advantages of using TDD**

* TDD helps a developer in better understanding of the code and its logic.
* Using TDD, the developer becomes more responsible for his piece of code. TDD helps save time in implementation due to better understanding of the business needs. Although, it is true that the amount of code increases.
* TDD increases productivity.
* The bugs and errors are found and fixed early during development. A bug found later in development lifecycle is much more costly compared to the same bug found early in the cycle. This helps in reducing the bugs per lines of code in final product. TDD results in 40% fewer defects in code compared to code produced that uses an ad-hoc testing approach. This also helps in saving the costs involved in maintenance.
* Since, you write tests for every piece of code, whenever the system breaks, you know whether a change in the code is responsible for the error.
* Unit tests form an essential component of design documentation.
* TDD produces better quality of code as the developer is forced to keep testing his code and refactor it although the time taken to write the code increases by a small percentage. TDD gives rapid feedback to a developer about the workings and quality of his code. This also helps in improving the quality of code during development.
* TDD increases the confidence of a developer in the behavior of the system, and ensuring that his code is working as intended and his ability to make changes in code without injecting new bugs increases.
* It helps create a simple design and also achieve continuous integration.
* It increases code coverage as there is a unit test for each piece of code.100 % code coverage is expected although hard to achieve in reality. Thus, TDD improves software quality.

**UNIT TESTING:**

unit testing is a method by which individual units of source code, sets of one or more computer program modules together with associated control data, usage procedures, and operating procedures, are tested to determine if they are fit for use.[1] Intuitively, one can view a unit as the smallest testable part of an application. In procedural programming a unit could be an entire module but is more commonly an individual function or procedure. In object-oriented programming a unit is often an entire interface, such as a class, but could be an individual method. Unit tests are created by programmers or occasionally by white box testers during the development process.

Ideally, each test case is independent from the others: substitutes like method stubs, mock objects,[2] fakes and test harnesses can be used to assist testing a module in isolation. Unit tests are typically written and run by software developers to ensure that code meets its design and behaves as intended.

The primary goal of unit testing is to take the smallest piece of testable software in the application, isolate it from the remainder of the code, and determine whether it behaves exactly as you expect. Each unit is tested separately before integrating them into modules to test the interfaces between modules. Unit testing has proven its value in that a large percentage of defects are identified during its use.

The most common approach to unit testing requires drivers and stubs to be written. The driver simulates a calling unit and the stub simulates a called unit. The investment of developer time in this activity sometimes results in demoting unit testing to a lower level of priority and that is almost always a mistake. Even though the drivers and stubs cost time and money, unit testing provides some undeniable advantages. It allows for automation of the testing process, reduces difficulties of discovering errors contained in more complex pieces of the application, and test coverage is often enhanced because attention is given to each unit.

For example, if you have two units and decide it would be more cost effective to glue them together and initially test them as an integrated unit, an error could occur in a variety of places:

* Is the error due to a defect in unit 1?
* Is the error due to a defect in unit 2?
* Is the error due to defects in both units?
* Is the error due to a defect in the interface between the units?
* Is the error due to a defect in the test?

Finding the error (or errors) in the integrated module is much more complicated than first isolating the units, testing each, then integrating them and testing the whole.

Drivers and stubs can be reused so the constant changes that occur during the development cycle can be retested frequently without writing large amounts of additional test code. In effect, this reduces the cost of writing the drivers and stubs on a per-use basis and the cost of retesting is better controlled.

**INTERGRATION TESTING:**

Integration testing is a logical extension of unit testing. In its simplest form, two units that have already been tested are combined into a component and the interface between them is tested. A component, in this sense, refers to an integrated aggregate of more than one unit. In a realistic scenario, many units are combined into components, which are in turn aggregated into even larger parts of the program. The idea is to test combinations of pieces and eventually expand the process to test your modules with those of other groups. Eventually all the modules making up a process are tested together. Beyond that, if the program is composed of more than one process, they should be tested in pairs rather than all at once.

Integration testing identifies problems that occur when units are combined. By using a test plan that requires you to test each unit and ensure the viability of each before combining units, you know that any errors discovered when combining units are likely related to the interface between units. This method reduces the number of possibilities to a far simpler level of analysis.

You can do integration testing in a variety of ways but the following are three common strategies:

* The top-down approach to integration testing requires the highest-level modules be test and integrated first. This allows high-level logic and data flow to be tested early in the process and it tends to minimize the need for drivers. However, the need for stubs complicates test management and low-level utilities are tested relatively late in the development cycle. Another disadvantage of top-down integration testing is its poor support for early release of limited functionality.
* The bottom-up approach requires the lowest-level units be tested and integrated first. These units are frequently referred to as utility modules. By using this approach, utility modules are tested early in the development process and the need for stubs is minimized. The downside, however, is that the need for drivers complicates test management and high-level logic and data flow are tested late. Like the top-down approach, the bottom-up approach also provides poor support for early release of limited functionality.
* The third approach, sometimes referred to as the umbrella approach, requires testing along functional data and control-flow paths. First, the inputs for functions are integrated in the bottom-up pattern discussed above. The outputs for each function are then integrated in the top-down manner. The primary advantage of this approach is the degree of support for early release of limited functionality. It also helps minimize the need for stubs and drivers. The potential weaknesses of this approach are significant, however, in that it can be less systematic than the other two approaches, leading to the need for more regression testing.

**BIBLIOGRAPHY**

1. **Wikipedia.org**
2. **WROX ASP.NET BOOK.**