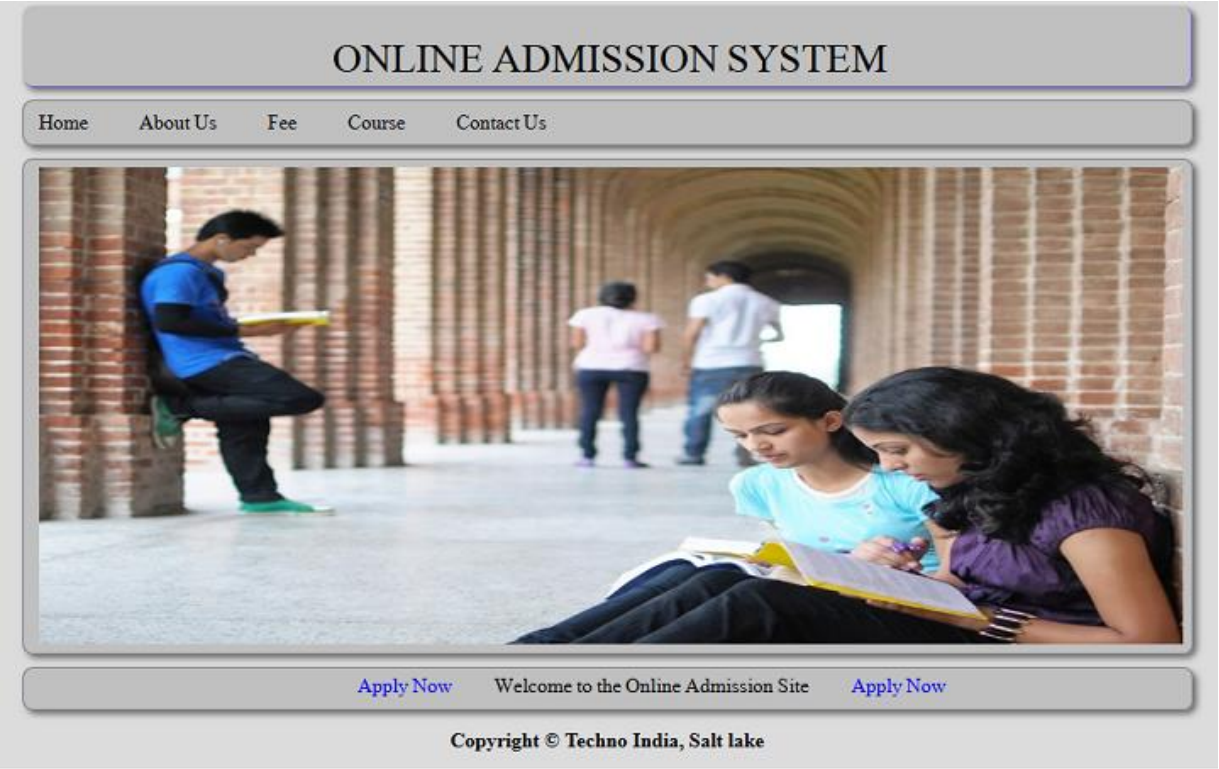
PROJECT REPORT ON
ONLINE ADMISSION SYSTEM

Paper-Design Lab Paper Code- IT891



SUBMITTED BY

- RITU RAJ (13000212033)
- KUMAR JYOTI SAHA (13000212014)
- MONIKA SUMAN (13000212017)
- ANANYA KUMARI (13000212063)

ACKNOWLEDGEMENT

The satisfaction that accompanies the successful completion of any task would be incomplete without the mention of people whose ceaseless cooperation made it possible, whose constant guidance and encouragement crown all efforts with success. We are grateful to our project mentor, for his guidance, inspiration and constructive suggestions that helped us in the preparation of this project. I will like to give a special mention to my colleagues. Last but not the least I am grateful to all the faculty members of Department of Information Technology for their support.

ABSTRACT

Today all the work at the time of admission of the students is done manually by ink and paper, which is very slow and consuming much efforts and time. It is required to Design of a Computerized Automated Student Admission System, to speed up and make it easy to use system. Student admissions are a vital part of any university's running because students are what keep a University alive. The student admission is one of the most important activities within a university as one cannot survive without students. A poor admissions system can mean fewer students being admitted into a university because of mistakes or an overly slow response time. The process begins with a potential student completing an application form through the Universities and Colleges Admissions Service, the first step for students is to apply directly to the university through a custom online form. This project's aim is to automate the system, pre- checking the inclusion of all required material and automatically ranking each student's application based on a number of criteria. These criteria include the ranking of their university, their grade at said university and their language grade Certificate. The data used by the system is stored in a database that will be the centre of all information held about students and the base for the remainder of the process after the initial application has been made. This enables things to be simplified and considerably quickened, making the jobs of the people involved easier. It supports the current process but centralizes it and makes it possible for decisions to be made earlier and easier way.

TABLE OF CONTENT

INTRODUCTION

OVERVIEW OF THE PROJECT

1. USERS' AREA:

i) User have to sign up here first. As a viewer, anyone can see the Public website. Only being a member of the website, they can access their own account, and able to apply for the desired course.

ii) The total facility that a user gets is listed below:

- User can register by giving their appropriate 10^{th} Percentage and Board's name, 12^{th} Percentage and Board's name, and last but not the least JEE rank.
- Next time for entering to profile they have to enter the same ID and Password.
- User can able to choose desired course.
- After all that User can check result just by login with their ID and Password.

2. ADMIN' AREA:

i) Admin should be the representative of college only he/she can able to approve student for admission.

ii) The total facility that a Admin gets is listed below:

- Admin don't have to register.
- There is no restriction of number of registration.
- Admin can view the registered student by course wise.

PROBLEM DEFINITION

Managing the large crowd that assembles for admission. Making space arrangements for the assembled parents. Employing staff to coordinate the crowd. Employing staff at multiple counters to issue admission forms. Employing staff to sort out the collected fees, the admission forms and the collected admission forms. Intimating the eligible candidates for a face to face interview. The whole admission process turning out to be a year on year painful activity for Parents & Institutions. Employee time & effort being utilized in activities which does not add any value to the institution and results in additional cost to the institution. Need to employ staff to generate reports, identify the number of admission forms submitted to each division and so on.

METHODOLOGY USED TO DEVELOP THIS PROJECT

J2EE is basically used to develop this project. The abbreviation of j2ee is JAVA 2 PLATFORM ENTERPRISE EDITION. It is platform independent, java centric environment from Sun Microsystem (together with industry partners such as IBM) for developing, building and deploying Web based enterprise applications online. The j2ee platform consists of a set of services, APIs and PROTOCOLS that provide the functionality for developing multitier, Web based applications. J2ee simplifies application development and decreases the need for programming and programmer training by creating standardized, reusable modular components and by enabling the tier to handle many aspects of programming automatically.

Some of the key features and services of j2ee

- At the client tier, j2ee supports pure HTML, as well as Java applets or applications. It relies on Java Server Pages and servlet code to create HTML or other formatted data for the client.
- Enterprise JavaBeans (EJBs) provide another layer where the platform's logic is stored. An EJB server provides functions
  Such as threading, concurrency, security, and memory management. There services are transparent to the author.
- Java Database Connectivity (JDBC) which is the java equivalent to Open Database Connectivity (ODBC), is the standard interface for java databases.
- The java servlet API enhances consistency for developers without requiring a graphical user interface.
- The Java Development Kit (JDK) is included as the core language package

- Write Once Run Anywhere technology is included to ensure portability.
- A security model is included to protect data both locally and in web based applications.

J2EE also includes a number of components added to J2SE model, such as the following:

- Full support is included for Enterprise JavaBeans. EJB is a server-based technology for delivery of program components in an enterprise environment. It supports the Extensible Mark-up language (XML) and has enhanced deployment and security features.
- The Java servlet API (Application Programming Interface) enhances consistency for developers without requiring a graphical user interface (GUI).
- Java Server pages (JSP) is the java equivalent to Microsoft's Active Server Pages (ASP) and is used for dynamic Web-enables data access and manipulation.

The J2EE architecture consists of four major elements:

- The J2EE Application programming model is standard programming model used to facilitate the development of multi-tier, thin client applications.
- The J2EE Platform includes necessary policies and APIs such as the Java servlets as Java Message Services (JMS).
- The J2EE compatibility test suite ensures that J2EE products are compatible with the platform standards.
- The J2EE reference implementation explains J2EE capabilities and provides its operational definitions.

JAVA SERVER PAGES (JSP) is a technology that helps software developers create dynamically generated web pages based on HTML, XML or other document types. Released in 1999 by Sun Microsystems JSP is similar to PHP, but it uses the Java programming language.

To deploy and run Java Server pages, a compatible web server with a servlet container, such as Apache Tomcat or Jetty, is required.

Architecturally JSP may be viewed as a high-level abstraction of Java Servlets. JSPs are translated into servlets at runtime, each JSP servlet is cached and re-used until the original JSP is modified. JSP can be independently or as the view component of a server side model-view-controller design, normally with JavaBeans as the model and Java Servlets (or a framework such as Apache Struts) as the controller. This is a type of Model 2 architecture.

JSP allows Java code and certain pre-defined actions to be interleaved with static web mark-up content, either the resulting page being compiled and executed on the server to deliver document. The compiled pages, as well as any dependent

Java libraries, use Java byte code rather than a native
Software format. Like any other java program, they must be executed within a Java virtual machine (JVM) that integrates with the server's host operating system to provide an abstract platform-neutral environment.

JSP are usually used to deliver HTML and XML documents, but through the use of output stream, they can deliver other types of data as well. The Web container creates JSP implicit object like page Context, servlet Context, session, request & response.

Now public interest **Servlet** defines methods that all servlets must implement. A servlet is a small java program that runs within a Web server. Servlets receive and respond to requests from Web clients, usually across HTTP, the hypertext transfer protocol.

- To implement this interface you can write a generic servlet that extends javax.servlet.GenericServlet or an HTTP servlet that extends javax.servlet.http.httpServlet. This interface defines methods to initialize a servlet, to service request, and to remove servlet from server.
- These are known as life cycle methods and are called in the following sequences:
- The servlet is constructed, then initialized with the init () method.
- Any calls from the clients to the service method are handled.

The servlet is taken out of service, then destroyed with the destroy method, then garbage collected and finalized.

In addition to the life cycle methods, this interface provides the getServletConfig method, which the servlet can use to get any start up information and the GetServletInfo method, which allows the servlet to return basic information about itself, such as author, version and copyright.

Oracle Database (commonly referred to as oracle RDBMS or simply Oracle) is an object-relational database management system produced and marketed by Oracle Corporation.

Larry Ellison and two friends and former co-workers, Bob Miner and Ed Oates. Started a consultancy called Software development Laboratories (SDL) in 1977. SDL developed the original version of the oracle software. The name Oracle comes from the code name of a CIA funded project Ellison had worked on while previously employed by Ampex. If the oracle database administrator has implemented oracle RAC (Real Application Clusters), then multiple instances, usually on different

servers, attach to a central storage array. This scenario offers advantages such as better performance, scalability and redundancy. However, support becomes more complex and many sites do not use RAC. In version 10g grid computing introduced shared resources where an instance can use (for example) CPU resources from another node (computer) in the grid. The oracle DBMS can store and execute stored procedures and functions within itself. PL/SQL (Oracle Corporation's proprietary extension to SQL) or the object oriented language Java can invoke such code objects and/or provide the programming structures for writing them.

Apache Tomcat is an open source web server and servlet container developed by the Apache Software Foundation (ASF). Tomcat implements several Java EE specifications for Java code to run in.

Apache is developed and maintained by an open community of developers under the auspices of the Apache Software Foundation, released under the Apache License 2.0 license, and is open-source software.

Tomcat 7.x implements the Servlet 3.0 and JSP 2.2 specifications. It requires Java version 1.6, although previous versions have run on Java 1.1 through 1.5. Version 5 through 6 saw improvements in garbage collection, JSP parsing, performance and scalability. Native wrappers, known as "Tomcat Native", are available for Microsoft Windows and UNIX for platform integration.

HARDWARE & SOFTWARE SPECIFICATIONS

HARDWARE PECIFICATIONS FOR SERVER:

Computer : - PC, Laptop

Processor : - Intel Pentium dual core

RAM : - 2GB

Input Device : - Mouse, keyboard

Output Device : - Printer, monitor

SOFTWARE SPECIFICATIONS FOR SERVER:

OPERATING SYSTEM : Windows 7 and above

APPLICATION SOFTWARE : Eclipse, JDK 1.7.

SERVER : Apache tomcat 7.0

DATABASE : Oracle database 10g

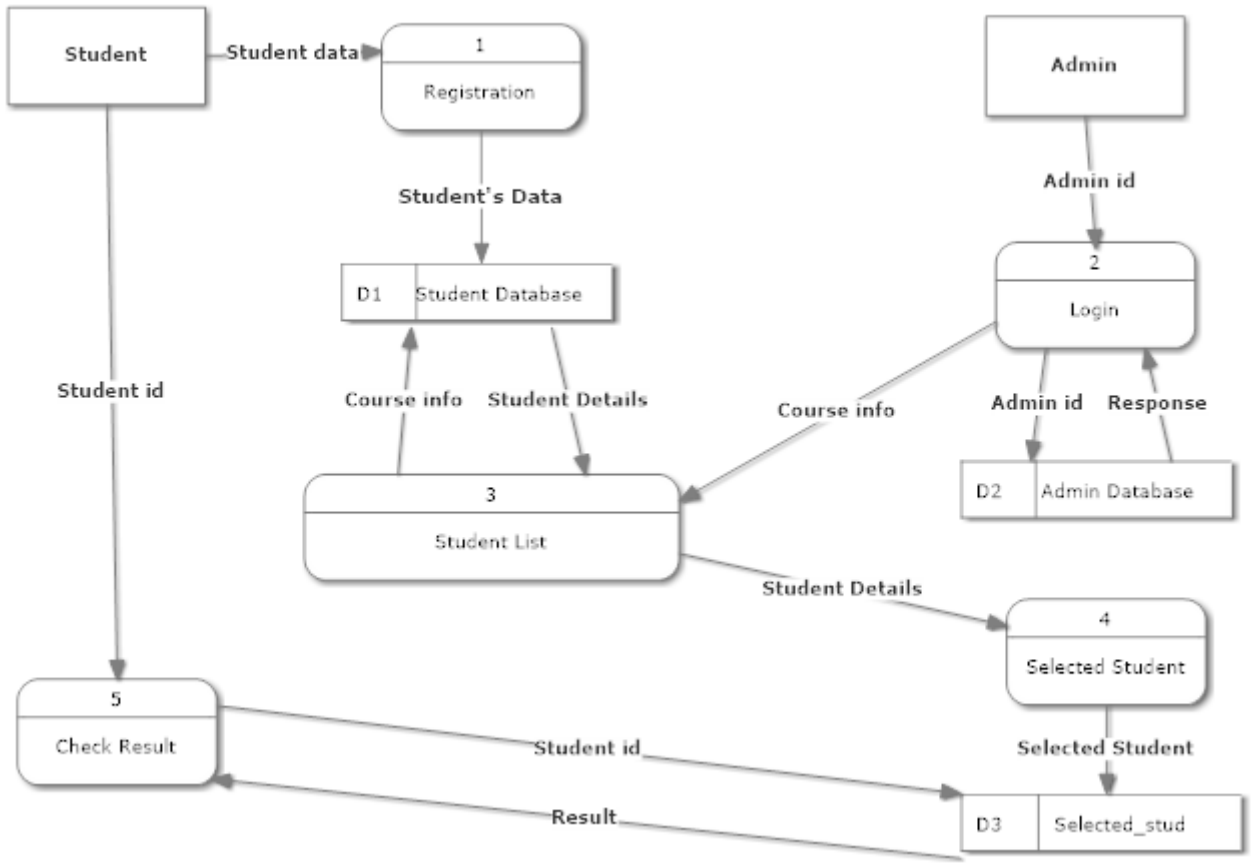
TECHNOLOGY : J2EE

SOFTWARE SPECIFICATION FOR CLIENT:

Browsers like internet explorer, google chrome, Mozilla Firefox.

SYSTEM DESIGN

FLOW DIAGRAM



TABLES STRUCTURES

u_id	pass

Admin Table

Dept	u_id

Selected Student Table

name	email	u_id	pass	byr	bmnth	bday	sex	btenth	pertenth	btwlv	pertwlv	rank	course	fname

Student Table

TABLE: 1 STUDENT DATABASE

	NAME	EMAIL	U_ID	PASS	BYR	BMTH	BDAY	SEX	FNAME	BTENTH	PERTENTH	BTWLV	PERTWLV	RANK	COURSE
1	Ankit Kumar	akumar123@gma...	anki...	1234	1991	03	02	Male	Amod	CBSE	80	ICSE	78	4563	it
2	Nehal Gupta	ng@gmail.com	ng	ng	1993	02	02	Male	Ashok Gupta	CBSE	70	CBSE	71	3654	it
3	Vivek Kumar	vk@gmail.com	vk123	vk123	1993	04	04	Male	Raju Kumar	CBSE	88	CBSE	88	30043	it
4	Pankaj K...	pk90@gmial.com	pk	pk123	1993	07	16	Male	Prakash K...	Biha...	69	CBSE	78	5539	it
5	Shubham	shubham@gmail...	shub	shub	1992	05	07	Male	Kumar Sinha	ICSE	89	CBSE	74	3429	it
6	Abhishek	ab@gmail.com	abk	abk	1994	02	03	Male	Ramesh Kumar	CBSE	79	CBSE	72	8992	it

TABLE 2:- ADMIN DATABASE

	U_ID	PASS
1	Admin	12345
2	ritu	raj123

TABLE 3:- SELECTED STUDENT DATABASE

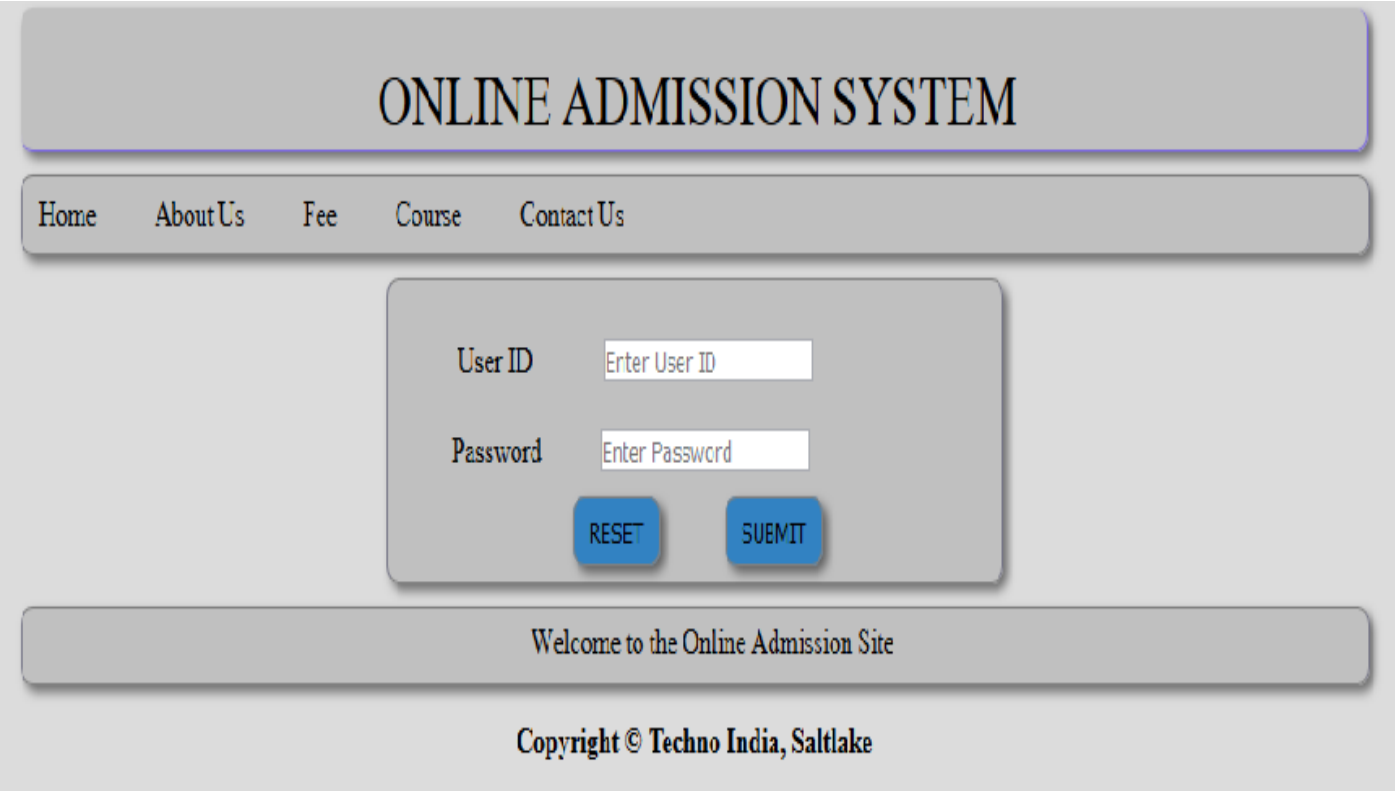
	DEPT	U_ID
1	it	shub
2	it	ng
3	it	ankit123

6. SNAPSHOTS OF THE SCREEN

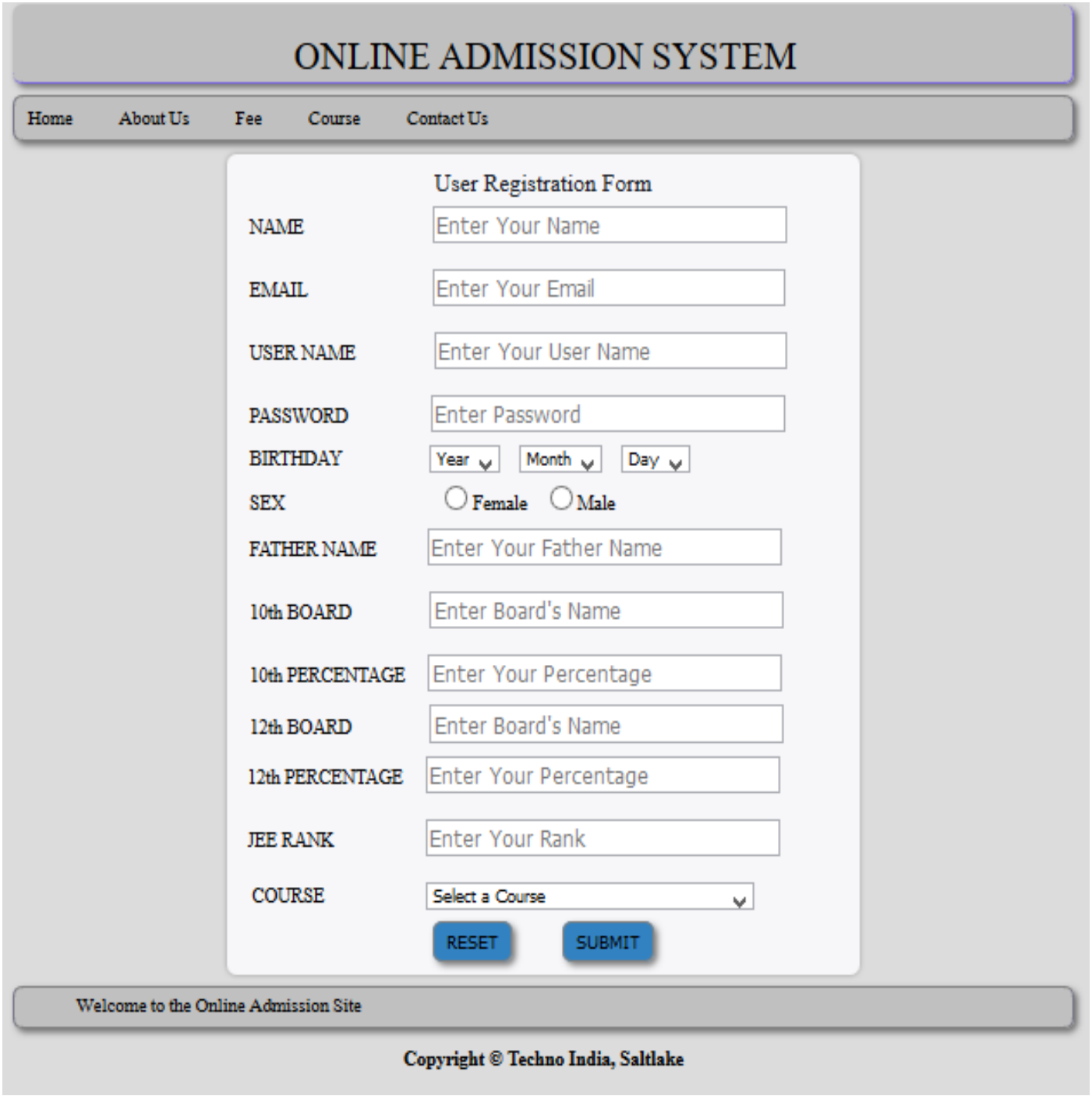
HOME PAGE:-


ONLINE ADMISSION SYSTEM
Home
About Us
Fee
Course
Contact Us
Apply Now
Welcome to the Online Admission Site
Apply Now
Copyright © Techno India, Salt lake

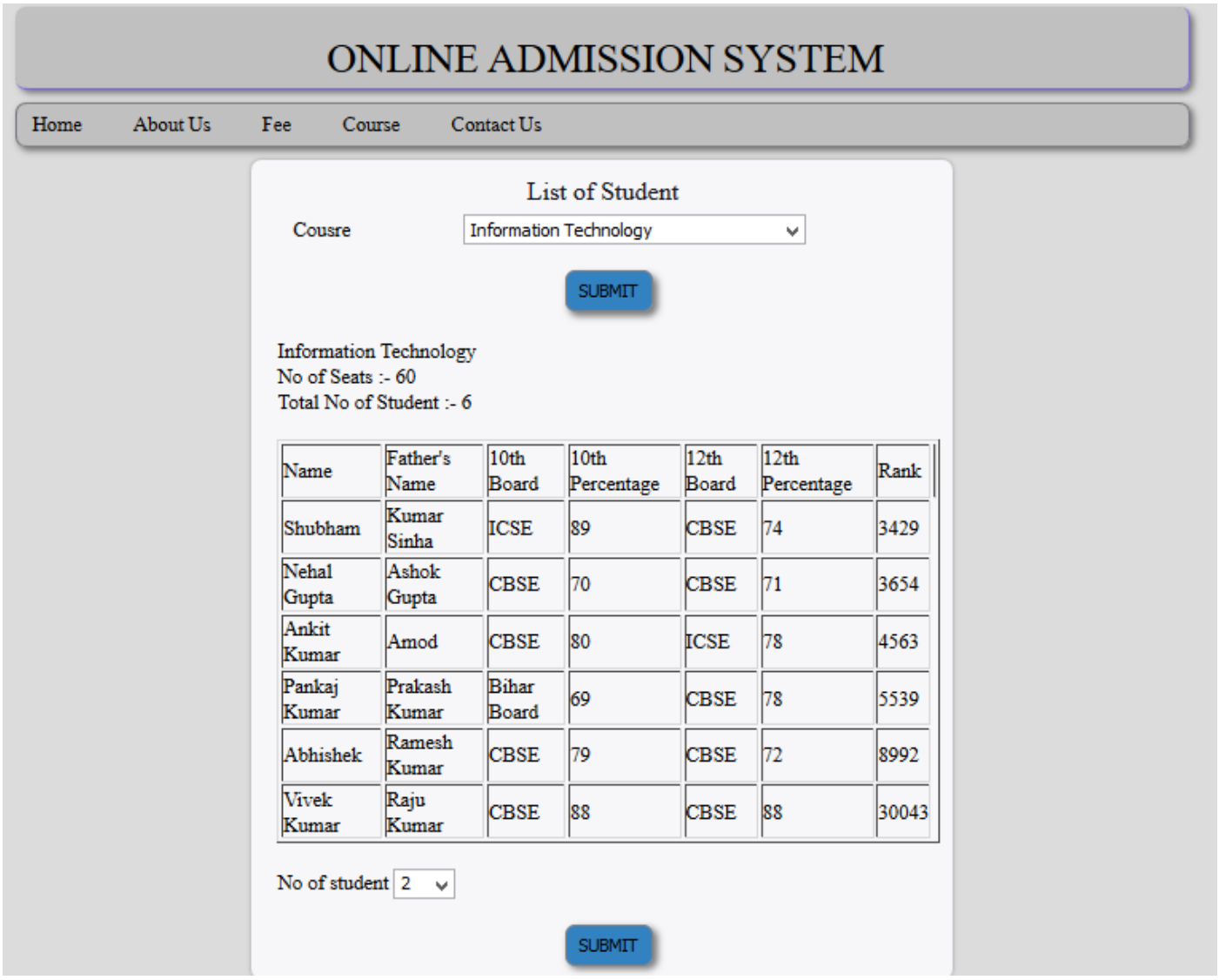
LOGIN: -



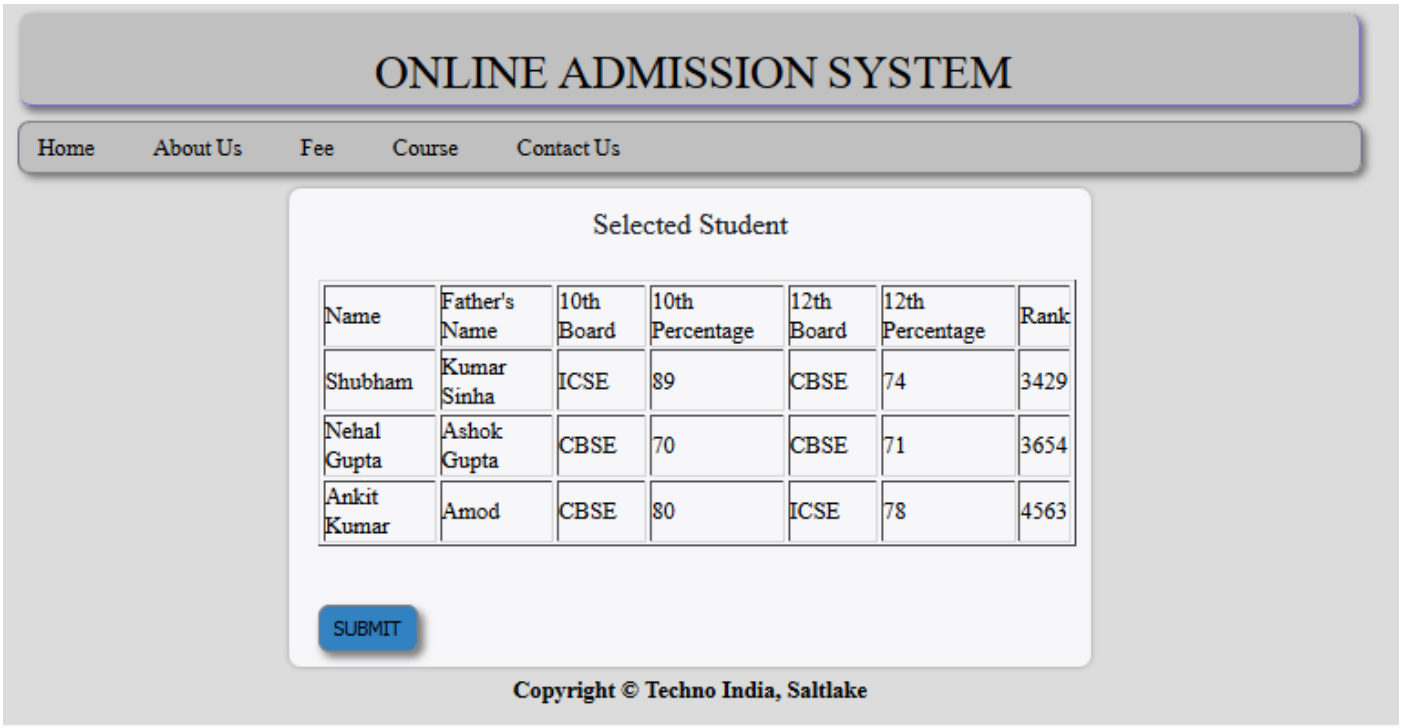
USER REGISTRATION:-



List of Student:-

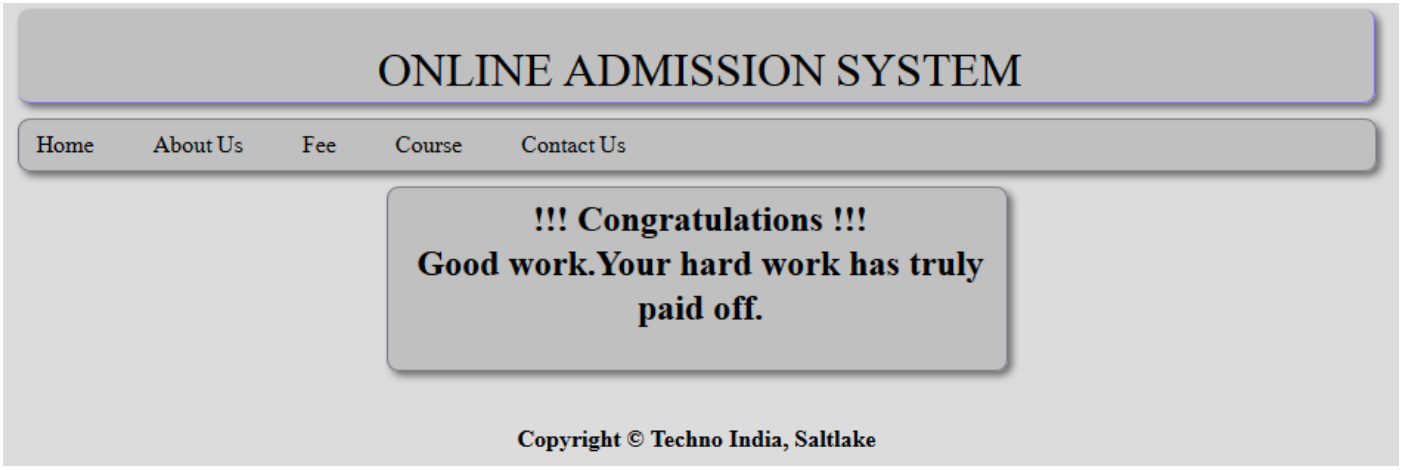


Selected Student:-



Check Result:


ONLINE ADMISSION SYSTEM
Home
About Us
Fee
Course
Contact Us
Welcomens10!!!!!!
Check Your Result
Welcome to the Online Admission Site
Copyright © Techno India, Saltlake


ONLINE ADMISSION SYSTEM
Home
About Us
Fee
Course
Contact Us
!!! Congratulations !!!
Good work.Your hard work has truly paid off.
Copyright © Techno India, Saltlake

LIMITATIONS AND FUTURE SCOPE OF THE PROJECT

LIMITATIONS

Although, the approach of this project is small, we have tried to have minimal limitations and also make it bug and error free.

The limitations possible with this project may be:

Only logged in Student can able to apply.

Only one course per Student is provided, so that the Student is confident about course that is he/she applying for.

Once Student register himself/herself, he/she cannot able to change the course, so selection of course is very important.

The website can easily get affected by virus because it is connected with Internet.

FUTURE SCOPE OF STUDY

Scope of the project is very broad in terms of other manually taking exams. This project's aim is to automate the system, pre-checking the inclusion of all required material and automatically ranking each student's application based on a number of criteria. These criteria include the ranking of their JEE exam, their grade at said university. The data used by the system is stored in a database that will be the centre of all information held about students and the base for the remainder of the process after the initial application has been made. This enables things to be simplified and considerably quickened, making the jobs of the people involved easier. It supports the current process but centralizes it and makes it possible for decisions to be made earlier and easier way.

This project will have a very bright future if the following can be incorporated:

Few of them are:-

- There was a lot of thinking's to increment the project, but for lack of time, many of them are still undone.
- There is a lot of scope to develop this project in future according to the requirements of particular interested people.

CONCLUSION

- The web-based admission system is a web application that allows a graduate program to collect the admission applications electronically. It reduces the time of sending application materials and time of typing data to nearly zero

- In brief, this web-based admission system is a user-friendly, cost effective web application which resolves rather complicated tasks of a graduate admission office.

BIBLIOGRAPHY

During the development of the project, we have used many resources and for that we are grateful to all the people concerned.

Given below are the names of some, which we have used during development and Documentation of the project.

"Head First Java" by Kathy Sierra

Useful sites for this project are as follows:

- www.w3school.com
- www.stackoverflow.com