****

**academic quality improvement system**

patel hardik p. (120783107018)

ruparel jayesh v. (120783107024)



Department of **c**omputer Engineering

Smt. S. R. Patel Engineering College, Dabhi, Unjha Gujarat Technological University, Gandhinagar

**academic quality improvement system**

A project submitted to

**sMT. s. r. pATEL eNGINEERING cOLLEGE,**

**Gujarat Technological University**

In partial fulfillment of the project assigned to prefinal semester (7th) of

**BACHELOR OF TECHNOLOGY**

In

COMPUTER ENGINEERING

Submitted By

patel hardik p. (120783107018)

ruparel jayesh v. (120783107024)

Under the guidance of

**Prof. a.v.panchal**



Department of Computer Engineering

Smt. S. R. Patel Engineering College, Dabhi, Unjha

Gujarat Technological University, Gandhinagar

|  |  |
| --- | --- |
| **logo** | **DEPARTMENT OF COMPUTER ENGINEERING**  Smt. S. R. PATEL ENGINEERING COLLEGE  At & Po: DABHI, Ta: UNJHA, 384170,  Logo of GTU_ High Resolusion1DIST: MEHSANA[N.G.] |

***Certificate***

**This is to certify that we have examined the project titled**

**“ACADEMIC QUALITY IMPROVEMENT SYSTEM”**

**Being submitted by**

**Patel Hardik P. (120783107018)**

**Ruparel Jayesh V. (120783107024)**

**Are Undergraduate Students of Computer Engineering. We hereby accord our approval of it as a study carried out and presented in manner required for its acceptance in Partial fulfillment of the project assigned to Pre-final Semester (8th) for which it has been submitted. This approval does not necessarily endorse or accept every statement made, opinion expressed or conclusion drawn as recorded in this project. It only signifies the acceptance of the project for the purpose for which it is submitted.**

|  |  |  |
| --- | --- | --- |
| **Internal Guide**  **Mr. A.V.Panchal**  **Assistant Professor,**  **Dept. of Computer Engg.**  **SRPEC-Dabhi, Unjha** |  | **Head of Department**  **Mr. Jignesh Patel**  **Assistant Professor,**  **Dept. of Computer Engg.**  **SRPEC-Dabhi, Unjha** |
| **Principal**  **DR. AMI H. SHAH**  **SRPEC-Dabhi, Unjha** | | |

|  |  |
| --- | --- |
| **logo** | **DEPARTMENT OF COMPUTER ENGINEERING**  Smt. S. R. PATEL ENGINEERING COLLEGE  At & Po: DABHI, Ta: UNJHA, 384170,  DIST: MEHSANA[N.G.] |

***Certificate of Examination***

This is to certify that the work presented in the Project Entitled **ACADEMIC QUALITY IMPROVEMENT SYSTEM**

Has Been Carried Out By

**Patel Hardik P. (120783107018)**

**Ruparel Jayesh V. (120783107018)**

In Partial fulfillment of the project assigned to Pre-final Semester (8th) of

**BACHELLOR OF ENGINEERING**

**In**

**COMPUTER ENGINEERING**

Of

Gujarat Technological University, Gandhinagar during the academic year 2013-14.

**External Examiner Internal Examiner**

**DATE: \_\_\_\_\_\_\_\_\_\_\_\_\_ DATE: \_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**SIGN: \_\_\_\_\_\_\_\_\_\_\_\_\_ SIGN: \_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**ACKNOWLEDGEMENT**

First and foremost let our grateful to almighty God. Our deepest regards and greatest admiration remain to my parents & family, who has taught our how to walk on an untraded path in the quest of knowledge.

I would like to express my deep sense of respect and gratitude my guide **Mr. A. V. Panchal**, Computer Engineering Department, SRPEC, Dabhi, who have never failed to help our to get a grasp on the subject.

Would like to thank our principal **Dr. A. H. Shah** and other faculty members in Computer Engineering Department especially **Prof. Jignesh Patel** (Head of the Department) for their valuable suggestion.

//We Sincerely thanks to **Mr. Manthan Patel** (M.D of e miracle solutions.) and **Mr. Satish //pansuriya**(employee of e miracle solutions).

Very special thanks also to other friends who had guided and helped me a lot in the project. Not to forget, I would also wish to thank all of my lecturers who had given their full co-operation. They had never hesitated to share knowledge and opinions in ensuring the project be completed successfully.

Also, we would like to thanks our librarian, all non-teaching staff and other all.

Patel Hardik P.(120783107018) Sign.: ……………..

Ruparel Jayesh V.(120783107024) Sign.: ……………...

**ABSTRACT**

Academic Quality Improvement system enhance academic quality, share accountability, building trust and confidence between educators and learners. This is the only way to achieve goals and the performance results it obtains from them.

AQIS means that student/learner shows his/her interest, put appropriate effort, appreciate people, learn to adopt things and understand the value of time. The portfolio provides important information for diagnosing students' issues - theirs strengths and weaknesses to help them to improve their performance.

**// Tables**

|  |  |  |
| --- | --- | --- |
| **S.r No.** | **Name Of Table** | **Page No.** |
| **6.1.1** | **LOGIN** | **48** |
| **6.1.2** | **ROLE** | **49** |
| **6.2.1** | **BRANCH** | **51** |
| **6.3.1** | **COLLEGE** | **52** |
| **6.4.1** | **COLLEGE\_BRANCH** | **53** |
| **6.4.2** | **STATE** | **54** |
| **6.4.3** | **CITY** | **55** |
| **6.4.4** | **COUNTRY** | **56** |
| **6.4.5** | **STUDENT** | **57** |
| **6.4.6**  **6.4.7**  **6.4.8**  **6.4.9**  **6.5.0** | **FACULTY**  **EVENT**  **EVENT\_REG**  **SEMINAR**  **SEMINAR\_REG**  **CIRCULAR**  **JOB**  **PAPER\_PRESENTATION**  **BLOG**  **COMMENT**  **QUESTION**  **ANSWER** | **58** |

**List Of Diagrams**

|  |  |  |
| --- | --- | --- |
| **S.r no** | **Name Of Figure** | **Page No.** |
| **4.1** | **Data flow diagram** | **28** |
| **4.2**  **4.3** | **E-R diagram**  **System flow diagram** | **31**  **32** |
| **5.1** | **Class diagram** | **34** |
| **5.2** | **Activity diagram** | **35** |
| **5.3** | **Use case diagram** | **41** |
| **5.4** | **Sequence diagram** | **44** |
| **5.5** | **State transition diagram** | **45** |

**Index:**

**Description Page No.**

**Chepter 1. Introduction**.............................................................................

1.1 Project Definition.................................................................................. 02

1.2 Purpose.................................................................................................. 03

1.3 Scope..................................................................................................... 03

1.4 Technology and literature review…………………………….............. 04

**Chepter 2. System Requirement study....................................................**

2.1 Software process model……………………………………………. 14

2.2 Product Perspective............................................................................... 15

2.3 User characteristics............................................................................... 16

2.4 Software and hardware requirements.................................................... 18

2.5 Interfaces............................................................................................... 19

2.6 Constraints............................................................................................ 19

2.7 Assumption and dependencies............................................................. 20

2.8 User Requirements............................................................................... 20

**Chepter 3. System Analysis.....................................................................**

3.1 Study of current system........................................................................ 22

3.2 Problem and weakness of current system............................................. 22

3.3 Requirement of new system................................................................. 22

3.4 Feasibility study.................................................................................... 23

**Chepter 4. Data Modelling.......................................................................**

4.1 Data flow diagram................................................................................. 28

4.2 ER diagram............................................................................................ 31

4.3 System flow diagram……………………………………………….. 32

**Chepter 5. System Design.........................................................................**

5.1 Class diagram........................................................................................ 34

5.2 Activity diagram................................................................................... 35

5.3 Use-case diagram................................................................................... 41

5.4 Sequence diagram.................................................................................. 44

5.5 State transition diagram........................................................................ . 45

**Chepter 6. Database Design………………………………………**

6.1 Jobseeker………………………………………………………………. 48

6.2 Employee……………………………………………………………… 51

6.3 Admin………………………………………………………………… . 52

6.4 Master Table…………………………………………………………. . 53

**Chepter 9. Conclusion................................................................**94

**Chepter 10. References..............................................................**96

**Chapter 1**

**Introduction**

**1.1 Project Definition: - Academic Quality Improvement System (AQIS)**

­ Academic Quality Improvement system centralizes the authority and responsibility for coordinating and assessing academic advising through a Director of Academic Advising. Common Learning Outcomes focuses on the knowledge, skills, and abilities expected of graduates from all programs.

Describe the processes for determining, communicating, and ensuring the stated common learning outcomes and who is Involved in those processes. System Learning Outcomes focuses on the knowledge, skills, and abilities graduates from particular programs are expected to possess. Describe the processes for determining, communicating, and ensuring the stated program learning outcomes and who is involved in those processes.

Academic System Design focuses on developing and revising programs to meet stakeholder’s needs. Describe the Processes for ensuring new and current programs meet the needs of the institution and its diverse stakeholders.

Academic System Quality focuses on ensuring quality across all programs, modalities, and locations. Describe the processes for ensuring quality academic programming.

Academic Student Support focuses on systems designed to help students be successful. Describe the processes for developing and delivering academic support to students.

Academic Integrity focuses on ethical practices while pursuing knowledge. Describe the processes for supporting ethical scholarly practices by students and faculty. Mission and Vision focuses on how the institution develops, communicates, and reviews its mission and vision. Describe the processes for developing, communicating, and reviewing the institution’s mission, vision, and values and who is involved in those processes. Leadership focuses on governance and leadership of the institution. Describe the processes for ensuring sound and effective leadership of the institution and who is involved in those processes.

## Project Summary

Academic Quality Improvement System centralizes the authority and responsibility for coordinating and assessing academic advising through a Director of Academic Advising.

It puts academic advising on a formal, institution-wide, continuous cycle of assessment that is developed within a teaching/learning pedagogical framework. It creates a university-wide academic advisor for undecided, exploring and re-admitted students. It gives colleges and schools the resources to implement assessment and increase the advising staff so that student/advisor ratios move closer recommended ratios.

## 1.2 Purpose

The purpose of this project is to develop the improvement quality of the students as well as academic. It gives colleges and schools the resources to implement assessment and increase the advising staff so that student/advisor ratios move closer recommended ratios.

In this system is alternative assessment program and it can either include a record of students achievements or simply document their best work and for some the portfolio documents the students learning process and still others use it as a means of promoting learner reflection.

## 1.3 Scope

The aim of the project is to build a simple, effective computerized Recruitment Module, which complements the regular workflow of HR management. It starts from the process of collecting relevant details and requirements from the Users until the candidate becomes a successful employee in the company. It covers the traditional HR functions:

1. Define Recruitment Processes
2. Smoothen the process of recruitment replacing the traditional method of recruitment.
3. Collect and Manage applicants’ profile, including personal detail, education details and work experience (if any).
4. Convert successful Applicants to Employees.
5. Manage the Recruitment Process for each Position through tracking all steps.
6. Enable Users to view the website for various jobs posted on the system
7. Allow Registered Users to add and edit their profile and appear for the evaluation process
8. Provide useful analysis; in the form of Reports and Statistics to support decision-making.
9. Filter worthy candidates based on the company’s requisites

ASP.NET is a new Internet programming technology developed by Microsoft. It applies an object-oriented approach for building dynamic Web application and acts as a Web development platform that provides the features necessary for developers to create enterprise-level Web application.

ASP.NET can be used for managing large-scale business as it is ideally suited for this purpose. These application can either be used for an intranet (with an organization) or can be made accessible on the internet after applying security measures so, that internet users can safely access and update information on the Web site.

**1.4 TECHNOLOGY AND LITERATURE REVIEW**

**1.4.1 ASP.NET**

ASP.NET is a new Internet programming technology developed by Microsoft. It applies an object-oriented approach for building dynamic Web application and acts as a Web development platform that provides the features necessary for developers to create enterprise-level Web application.

ASP.NET can be used for managing large-scale business as it is ideally suited for this purpose. These application can either be used for an intranet (with an organization) or can be made accessible on the internet after applying security measures so, that internet users can safely access and update information on the Web site.

**Features of ASP.NET:**-

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 offers several important advantages over previous Web development models:

**Enhanced Performance:**

* ASP.NET is compiled common language runtime code running on the server. Unlike its interpreted predecessors, ASP.NET can take advantage of early binding, just-in-time compilation, native optimization, and caching services right out of the box. This amounts to dramatically better performance before you ever write a line of code.

**Power and Flexibility:**

* Because ASP.NET is based on the common language runtime, the power and flexibility of that entire platform is available to Web application developers. The .NET Framework class library, Messaging, and Data Access solutions are all seamlessly accessible from the Web. ASP.NET is also language-independent, so you can choose the language that best applies to your application or partition your application across many languages. Further, common language runtime interoperability guarantees that your existing investment in COM-based development is preserved when migrating to ASP.NET.

**Simplicity:**

ASP.NET makes it easy to perform common tasks, from simple form submission and client authentication to deployment and site configuration. For example, the ASP.NET page framework allows you to build user interfaces that cleanly separate application logic from presentation code and to handle events in a simple, Visual Basic - like forms processing model. Additionally, the common language runtime simplifies development, with managed code services such as automatic reference counting and garbage collection.

**Manageability:**

ASP.NET employs a text-based, hierarchical configuration system, which simplifies applying settings to your server environment and Web applications. Because configuration information is stored as plain text, new settings may be applied without the aid of local administration tools. This "zero local administration" philosophy extends to deploying ASP.NET Framework applications as well. An ASP.NET Framework application is deployed to a server simply by copying the necessary files to the server. No server restart is required, even to deploy or replace running compiled code.

**Scalability and Availability:**

ASP.NET has been designed with scalability in mind, with features specifically tailored to improve performance in clustered and multiprocessor environments. Further, processes are closely monitored and managed by the ASP.NET runtime, so that if one misbehaves (leaks, deadlocks), a new process can be created in its place, which helps keep your application constantly available to handle requests.

**Customizability and Extensibility:**

ASP.NET delivers a well-factored architecture that allows developers to "plug-in" their code at the appropriate level. In fact, it is possible to extend or replace any subcomponent of the ASP.NET runtime with your own custom-written component. Implementing custom authentication or state services has never been easier.

**Security:**

With built in Windows authentication and per-application configuration, you can be assured that your applications are secure.

The Microsoft .NET Platform currently offers built-in support for 3 languages.

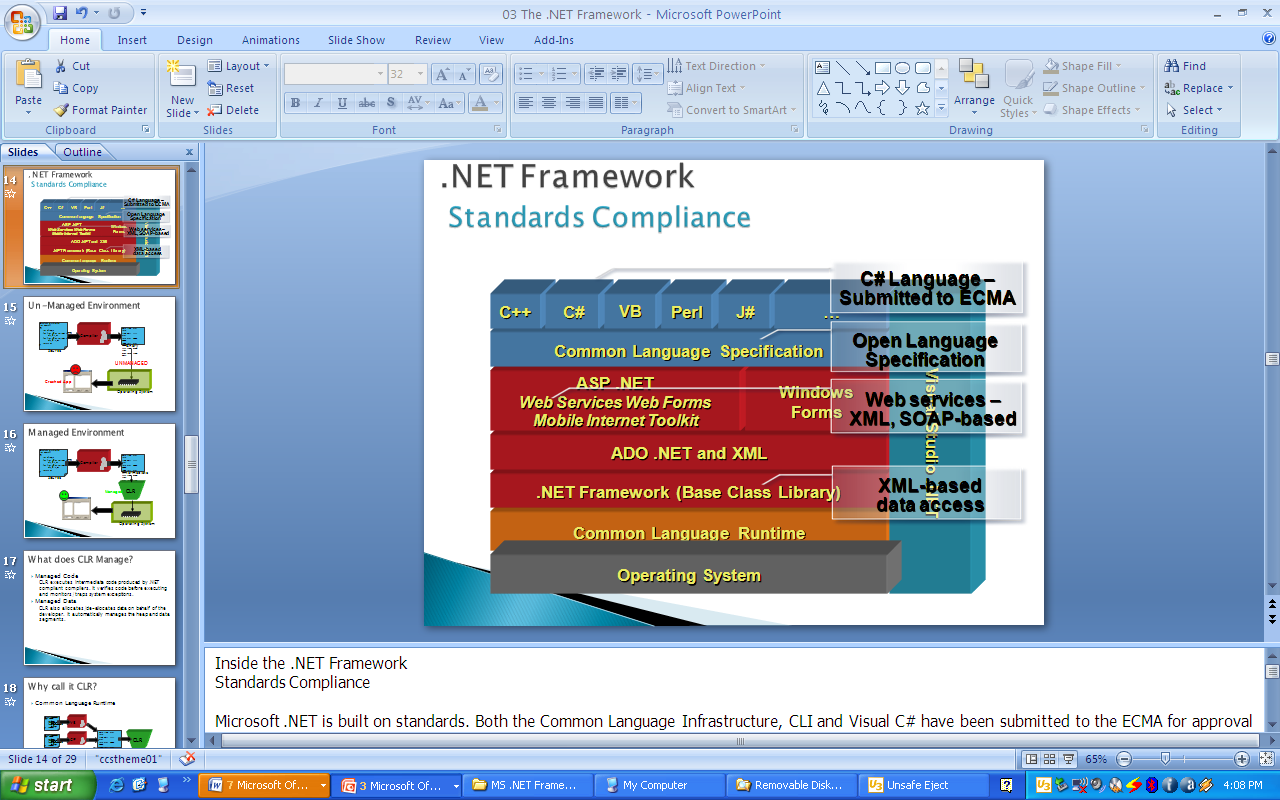
**Why ASP.NET?**

* Since 1995, Microsoft has been constantly working to shift its focus from Windows-based platforms to the Internet. As a result Microsoft introduced ASP (Active Server Pages) in November 1996. ASP offered the efficiency of ISAPI applications along with a new level of simplicity that made it easy to understand and use.
* However, ASP script was an interpreted script and consisted unstructured code and was difficult to debug and maintain. As the web consists of many different technologies, software integration for Web development was complicated and required to understand many different technologies.
* Also, as applications grew bigger in size and became more complex, the number of lines of source code in ASP applications increased dramatically and was hard to maintain. Therefore, an architecture was needed that would allow development of Web applications in a structured and consistent way.
* The .NET Framework was introduced with a vision to create globally distributed software with Internet functionality and interoperability. The .NET Framework consists of many class libraries, includes multiple language support and a common execution platform.
* It's a very flexible foundation on which many different types of top class applications can be developed that do different things. Developing Internet applications with the .NET Framework is very easy.
* ASP.NET is built into this framework; we can create ASP.NET applications using any of the built-in languages.

Unlike ASP, ASP.NET uses the Common Language Runtime (CLR) provided by the .NET Framework. This CLR manages execution of the code we write. ASP.NET code is a compiled CLR code instead of interpreted code (ASP). CLR also allows objects written in different languages to interact with each other. The CLR makes development of Web applications simple.

**What is .NET Framework?**

* .NET is the framework for which we develop applications.  It sits in between our application programs and operating system. Applications developed for .NET run inside .NET and are controlled by .NET. It supports both Windows and web applications.
* Applications developed for .NET make use of the features of .NET. We will see more of features of .NET later in this piece.



* .NET provides an object oriented environment. It ensures safe execution of the code by performing required runtime validations. For example, it is never possible to access an element of an array outside the boundary. Similarly, it is not possible to a program to write into another programs area, etc. The runtime validations performed by .NET makes the entire environment robust.

**Why C#?**

For this project we choose the language C#. First the C# is the most powerful feature of C# it is very powerful and efficiency like C++ and it is many methods in built so by these methods we can use its functionality.

First Microsoft .net framework that is available with several Microsoft operating systems. It includes a large library of pre-coded problem solution to common programming problem and a virtual Machine that manage the execution of program written specifically for the framework.

**C# Advantage**

* Unsafe code. This allows pointer arithmetic etc, and can improve performance in some situations. However, it is not to be used lightly, as a lot of the normal safety of C# is lost (as the name implies). Note that unsafe code is still managed code, i.e. it is compiled to IL, Jitter, and run within the CLR.
* It is a object oriented programming
* Good design
* It is a support for dynamic WebPages.
* Efficient data access.
* Improved security.

**1.4.2 Microsoft SQL Server 2008**

**Database:**

A database management, or DBMS, gives the user access to their data and helps them transform the data into information. Such database management systems include dBase, paradox, IMS, SQL Server and SQL Server. These systems allow users to create, update and extract information from their database. A database is a structured collection of data. Data refers to the characteristics of people, things and events. SQL Server stores each data item in its own fields. In SQL Server, the fields relating to a particular person, thing or event are bundled together to form a single complete unit of data, called a record (it can also be referred to as raw or an occurrence). Each record is made up of a number of fields. No two fields in a record can have the same field name.

During an SQL Server Database design project, the analysis of your business needs identifies all the fields or attributes of interest. If your business needs change over time, you define any additional fields or change the definition of existing fields.

SQL Server Tables. SQL Server stores records relating to each other in a table. Different tables are created for the various groups of information. Related tables are grouped together to form a database.

**Primary Key:**

Every table in SQL Server has a field or a combination of fields that uniquely identifies each record in the table. The Unique identifier is called the Primary Key, or simply the Key. The primary key provides the means to distinguish one record from all other in a table. It allows the user and the database system to identify, locate and refer to one particular record in the database.

**Relational Database:**

Sometimes all the information of interest to a business operation can be stored in one table. SQL Server makes it very easy to link the data in multiple tables. Matching an employee to the department in which they work is one example. This is what makes SQL Server a relational database management system, or RDBMS. It stores data in two or more tables and enables you to define relationships between the tables and enables you to define relationships between the tables.

**Foreign Key:**

When a field is one table matches the primary key of another field is referred to as a foreign key. A foreign key is a field or a group of fields in one table whose values match those of the primary key of another table.

**Referential Integrity:**

Not only does SQL Server allow you to link multiple tables, it also maintains consistency between them. Ensuring that the data among related tables is correctly matched is referred to as maintaining referential integrity.

**Data Abstraction:**

A major purpose of a database system is to provide users with an abstract view of the data. This system hides certain details of how the data is stored and maintained. Data abstraction is divided into three levels

**Physical level:**

This is the lowest level of abstraction at which one describes how the data are actually stored.

**Conceptual Level:**

At this level of database abstraction all the attributed and what data are actually stored is described and entries and relationship among them.

**View level:**

This is the highest level of abstraction at which one describes only part of the database.

**Advantages of RDBMS:**

* Redundancy can be avoided
* Inconsistency can be eliminated
* Data can be Shared
* Standards can be enforced

**Disadvantages of DBMS:**

A significant disadvantage of the DBMS system is cost. In addition to the cost of purchasing of developing the software, the hardware has to be upgraded to allow for the extensive programs and the workspace required for their execution and storage. While centralization reduces duplication, the lack of duplication requires that the database be adequately backed up so that in case of failure the data can be recovered.

**Chapter 2**

**SYSTEM REQUIREMENT STUDY**

**2.1 Software Process Model**

For the development of the project **Academic Quality Improvement System,** SPIRAL MODEL is best suited. The spiral model is a software development process combining elements of both design and prototyping-in-stages, in an effort to combine advantages of top-down and bottom-up concepts.

We have selected this model as it is a risk-reduction oriented model that breaks a software project up into mini-projects, each addressing one or more major risks. After major risks have been addressed, the spiral model terminates as a waterfall model.



**Figure 2.1 Spiral Model**

**//Background**

// Recruitment is an important process which provides the organization with a pool of qualified candidates. The more qualified the pool the better the success rate in selection and staffing of the organization. Recruitment can help the organization meet their affirmative action goals and increases organizational effectiveness. Hence, the importance of a Recruitment Module.

## //Objectives

To develop a Web-based Employee Recruitment System and deploy it onto the Client’s System.

## //Introduction

Employee Recruitment System is a HR module which has to be incorporated on an existing client’s system. The module is built keeping in view the client’s expanding business, annual recruitment channel the company follows and the changing business environment.

**//2.2 Product Perspective**

The Online Job Portal System is a package to be used by agencies to improve the efficiency of business. The Online Job Portal System to be developed benefits greatly the members. The system provides jobs catalog and information to members and helps them decide on the jobs to apply. The Admin can keep the jobs catalog updated all the time so that the members (Job seekers and the agencies) get the updated information all the time.

## 2.3 User Characteristics:

|  | **Description** | **Tasks / Responsibilities** |
| --- | --- | --- |
| Admin | Refers to privileged actors of the System who have control over the whole system. | * Login * Create Event * Create Blog * Event Registration * Create Circular * Create Seminar * Seminar Registration * Logout |
| Student | Refers to all those actors who are creating its own profile and many more in this system. | * Login * Post Question * Post Answer * Create Blog * Upload Resume * Update profile * Change Password * Logout |
| Staff | Refers to this actors share knowledge about this system. | * Login * Post Question * Post Answer * Post Circular * Post Blogs * Logout |

**2.4 Software and Hardware Requirement:**

* **Software Environment:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Tools/Technology** | **Front-end** | **-** | ASP.NET (with C# Language) |
| **Back-end** | **-** | MS-SQL Server 2008. |
| **Browser** | **-** | Internet Explorer 6.0, Firefox, Google Chrome |
|  | **Editor** | **-** | Microsoft Visual Studio 2013. |
| **Operating System** | Windows XP, Windows 7, Windows Vista. | | |

* **Hardware Environment:**

|  |
| --- |
| * Intel Core 2 duo * 20GB Hard Disk Drive * RAM: 2 GB Expected. |

## 2.5 Interfaces

**//2.4.1 User Interface**

The user Interface is provided by the any kind of web browser like Internet Explorer, Mozilla Firefox, msn, safari etc. As the project is the extension of the live project, based on the client’s requirement, the project testing will be done for the above specified four browsers so the project would be portable and will have pluggable look and feel.

**2.4.2 Communication Interface**

This is website so it requires HTTP protocol and Internet connection.

**2.4.3 Software Interface**

The application mainly interacts with the SQL Server database for storing data at the back end. Other than this it does not deal with any software.

## 2.6 Constraints:

1. The information of all the users must be stored in a database that is Accessible by the Online Job Portal System.
2. The Online Job Portal System is connected to the computer and is Running all 24 hours a day.
3. The users access the Online Job Portal System from any computer that has Internet browsing capabilities and an Internet connection.
4. The users must have their correct usernames and passwords to Enter into the Online Job Portal System.

**2.7 Assumptions and Dependencies:**

1. The users have sufficient knowledge of computers.
2. The Computer should have Internet connection and Internet server Capabilities.
3. The users know the English language, as the user interface will be Provided in English
4. The product can access the member’s database.

**2.8 User Requirements:**

**2.8.1 Functional requirements:**

1. Data Security and user authentication.
2. Cost effectiveness of system.
3. Not time consuming.
4. Reduce the complexity of function and work.

**2.8.2 Non Functional requirements:**

1. System should have a graphically friendly user interface such that easy for any user to learn it.
2. Must generate appropriate message.
3. Buttons should be looking effective.
4. Menu items names should be user understandable.

**Chapter 3**

**SYSTEM ANALYSIS**

**3.1 Study of Current System**

The recruitment process is a very important process for all kinds of people as the common man wants the job and the company needs man power. The current system is based only on human beings. The all activities of this process are done on paper work, nothing is system dependent. The candidates apply for job manually to the company by sending their resumes. The company responds them if they are qualified and then the evaluation process takes place which filters the candidates. Filtered candidates are called for the interviews and thus the whole recruitment process takes place.

## 3.2 Problems and Weaknesses of Current System

The recruitment process is a very large and important process which is mostly done by human beings, not by the system. The conventional approach of submitting resumes to the company and then holding the evaluation process is a very tedious task as the company has to give advertisements or has to go to the colleges for recruiting people. This process can be made easy and flexible for the candidates as well as the company managers so that the whole recruitment process is handled automatically.

This conventional approach of recruiting people increases the work of maintaining candidates on paper work which requires a lot of manpower. Other than this, it increases the complexity and also contains less efficiency. The automated system needs to be found which replaces the current conventional system or at least decreases this whole process' complexity.

**3.3 Requirements of New System**

The problems of the conventional recruitment process increases the need of some automated system which handles the process at some extent and can decrease the work of this process. The system can be built which is general and not specific to any company as the evaluation process contains different steps for different companies. Any company can use that system and can decrease complexity of the conventional approach. The new system may not include the phase of evaluation process but other general formalities of saving resumes and other details of the candidates, maintaining their data etc. can be included.

## 3.4 Feasibility Study

The feasibility study is the important step in any software development process. This is because it makes analysis of different aspects like cost required for developing and executing the system, the time required for each phase of the system and so on. If these important factors are not analyzed then definitely it would have impact on the organization and the development and the system would be a total failure. So for running the application and the organization successfully this step is a very important step in a software development life cycle process.

By making analysis with the requirement of the organization it would be possible to make a report of identified area of problem. By making a detailed analysis in this area a detailed document or report is prepared in this phase which has details like project plan or schedule of the project, the cost estimated for developing and executing the system, target dates for each phase of delivery of system developed and so on. This phase is the base of software development process since further steps taken in software development life cycle would be based on the analysis made on this phase and so careful analysis has to be made in this phase.

Though the feasibility study cannot be focused on a single area some of the areas or analysis made in feasibility study is given below. But all the steps given below would not be followed by all system developed. The feasibility study varies based on the system that would be developed.

* Feasibility study is made on the system being developed to analyze whether the system development process require training of personnel. This help in designing training sessions as required in later stage
* Is the system developed has scope for expanding or scope for switching to new technology later if needed in ease. In study is made to find the portability of the system in future.
* The above feasibilities are analysis which helps in development of the system. But the scope of feasibility study does not end with this. Analysis or feasibility study also includes the analysis of maintenance stage. In other words feasibility study is made to analyze how one would maintain the system during maintenance stage. This helps sin planning for this stage and also helps in risk analysis. Also the analysis helps in making analysis about what training must be given and how and what all documents must be prepared to help users and developers to face maintenance phase.

Feasibility study is the likelihood the system will be useful to organization. After studying the requirements, whether the proposed project is feasible or not, is determined by checking the various feasibilities. The three aspects in the feasibility study portion of preliminary investigation are:

**3.4.1 TECHNICAL FEASIBILITY:-**

Technical feasibility corresponds to determination of whether it is technically feasible to develop the software.

* Necessary technology exists to do what is suggested and required by the organization.
* The proposed equipments have the technical capacity to hold the data required to use the new system.
* The proposed system will provide adequate response to inquiries regardless of the location if users.
* The hardware needed to develop and implement the system is adequate.
* The software guarantees accuracy, reliability and ease of access and data security

**3.4.2 ECONOMIC FEASIBILITY:-**

A system that can be developed and that will be used if installed must still be a good investment for the organization. Financial benefits must equal or exceed the costs.

The financial and economical issues raised are as under:

* No extra cost is incurred for developing the system. As required software are already used by the department.
* No extra cost for the modification or addition of software and hardware will require in case of future expansion of the current system.
* As the project is to be developed by trainees the cost incurred by the company is in the form of resource allocation rather than monetary. The cost on the company is indirect in the form of resources utilization.
* The company will be at profit if they implement this system because of the cost of implementation is nominal as compared to the profit they will be earning in terms of efficiency.

Considering above factors project is economically feasible.

**3.4.3 OPERATIONAL FEASIBILITY:**

Operational feasibility focuses on whether the system will work when it is developed and installed. Operationally the system is feasible because:

* There is sufficient support for the project from management and user. The system is well liked and used to the extent that persons will not be able to see reasons for change.
* The current business methods are not acceptable because the manual system is time consuming. The users though initially repressive worked along with the development team once the initial doubts were cleared.
* The users have been involved in the planning and development of the project. This reduces the chances of resistance to the system.
* The proposed system will not cost any harm to the existing system and its users.
* No special training required for the user as it has a self explanatory interface. Validation of data input is taken care of by the system and not by the user.

Since the most trivial of issues assumes a major problematic state later in the development cycle, every possible aspect of operational feasibility was checked. The proposed project passed all the feasibility tests and hence was declared feasible to organization and its functioning

.

**Chapter 4**

**DATA MODELING**

**4.1 Data Flow Diagram**

**Level 0**



**Figure 4.1 Level 0 of DFD**



**Figure 4.2 Levels 1 of DFD**



**Figure 4.3 Level 2 of DFD**



**4.2 ER Diagram**



**Figure 4.4 E-R Diagrams**

* 1. **system flow diagram** 

**Figure 4.5 System Flow Diagrams**

**Chapter 5**

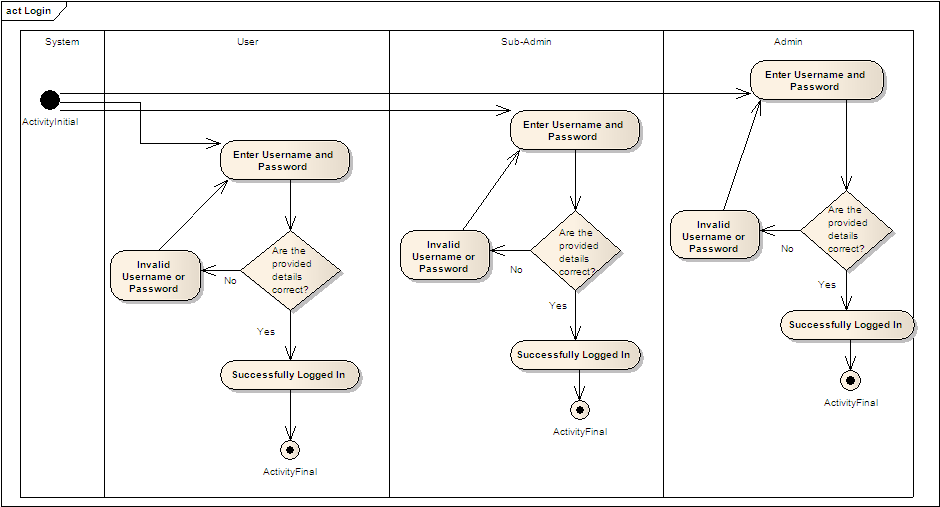
**SYSTEM DESIGN**

**5.1 Class Diagram**

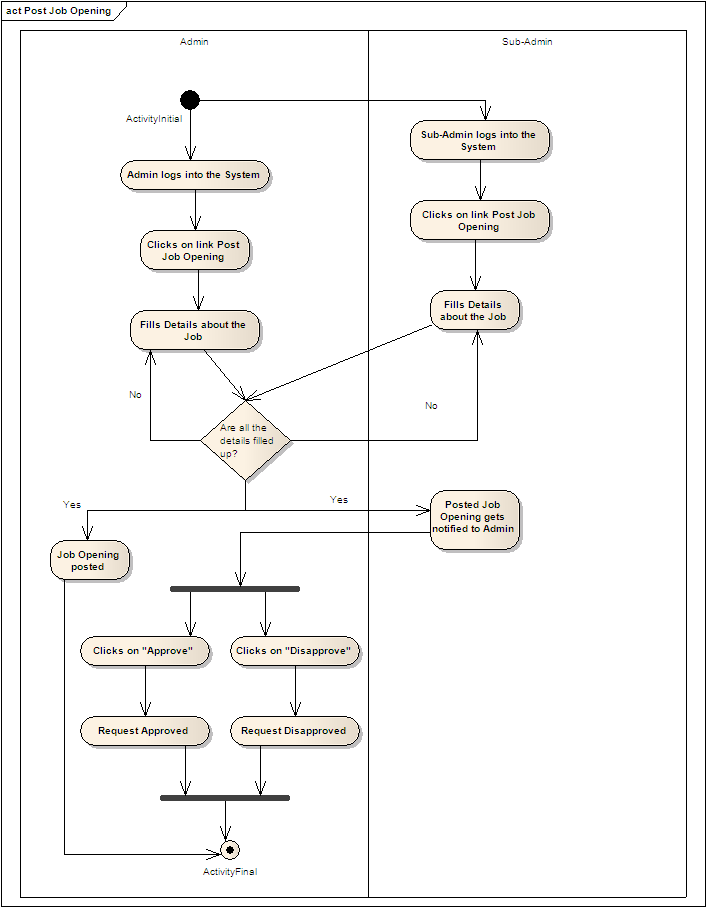
**5.2 Activity Diagram**

Below are shown the activity diagrams of the system. As the organization required the activity diagrams, instead of flow chart I’ve drawn the activity diagrams. The diagrams are shown below very clearly and clear the idea about the activities carried out by the project.

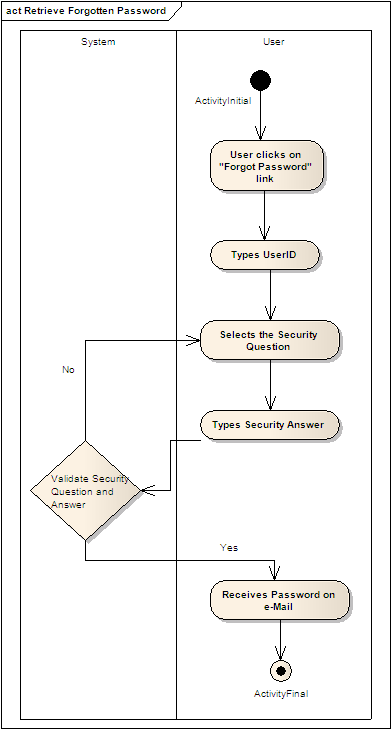
**(1) Activity Diagram for Login:**



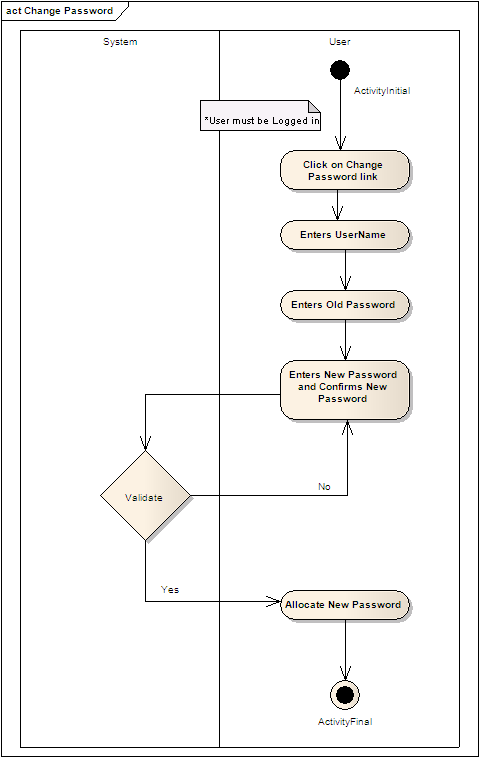
**(2) Activity Diagram for Post Job Openings:**



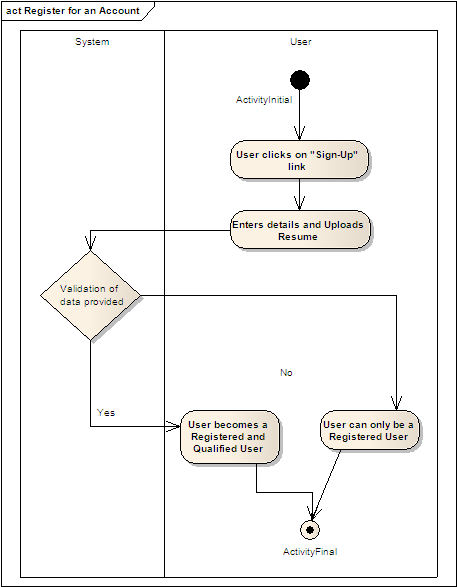
**(3) Activity Diagram for Retrieve Forgotten Password:**

****

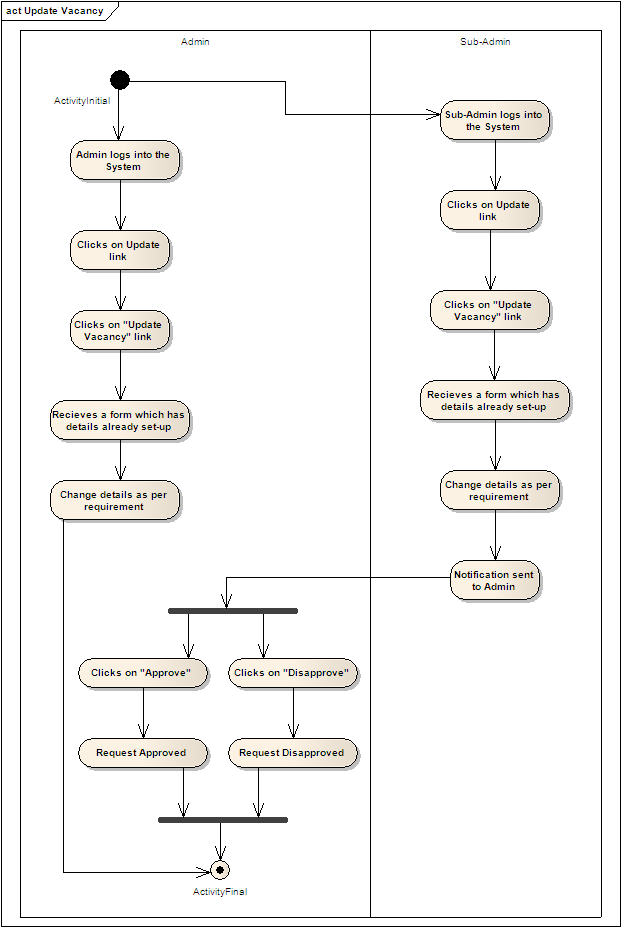
**(4) Activity Diagram for Change Password:**

****

**(5) Activity Diagram for Register for an Account:**

****

**(6) Activity Diagram for Update Vacancy:**

****

**5.3 Use case Diagram**

1. **Use Case Diagram for Employee**



**5.3 Use case Diagram**

**1) Use Case Diagram for Employee**

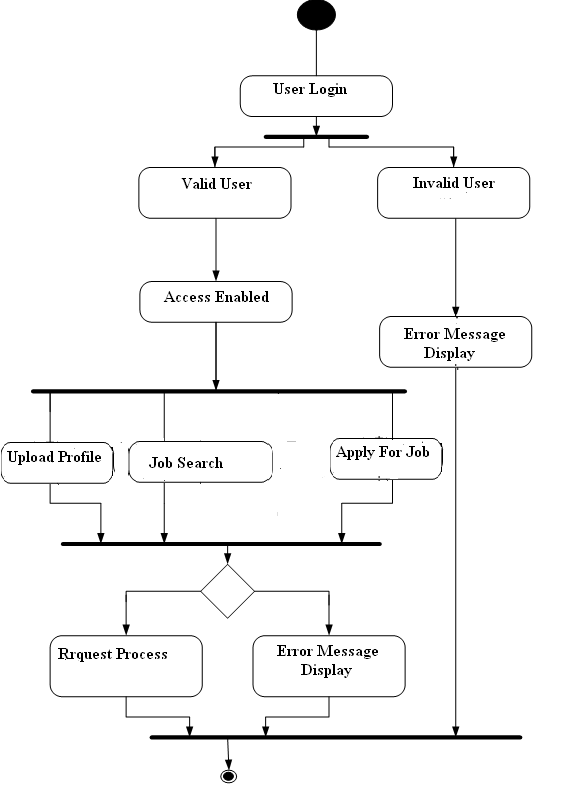
1. **Use Case Diagram for Admin**

**3) Use Case Diagram For Job Seeker**

**5.4 Sequence Diagram**

\\

**5.5 State transition diagram**



Employer Login

Invalid Emp Valid Emp

Access Enabled

Update Job Post List

Update Profile

Job Post

Pending Accept

Done Respective Process

Accept

Done Respective Process

Error Message Display

Job List

**Chapter 6**

**DATABASE DESIGN**

**6.1 Jobseeker:**

**6.1.1 reg\_master**

|  |  |  |  |
| --- | --- | --- | --- |
| Field Name | Data Type | Constraint | Description |
| Uname | Varchar(50) | Not Null | User Name |
| Upassword | varchar(15) | Not Null | User password |
| Umobileno | numeric(15) | Allow Null | User mobile no |
| Dob | nvarchar(10) | Not null | Date of Birth |
| U type | Int | Not Null | User Type |
| Gender | varchar(50) | Not Null | Gender |
| C\_code | Int | Not Null | Country code |
| OC\_code | Int | Allow Null | Other country code |
| S\_code | Int | Not Null | State code |
| OS\_code | Int | Allow Null | Other state code |
| O\_city | nvarchar(50) | Not Null | Other city |
| Texpyr | varchar(10) | Not Null | Total experience year |
| Email\_id | Nvarchar(50) | Not Null | Email Id |
| Uquification | varchar(50) | Not null | User qualification |
| Urecsalary | varchar(50) | Allow null | User respect salary |
| Created by | varchar(50) | Not Null | Created user name |
| Created date | Date & time | Not Null | Created date |
| Modified by | varchar(50) | Not Null | Modified user name |
| Modified date | Date & time | Not Null | Modified date |

**6.1.2. jobpost**

|  |  |  |  |
| --- | --- | --- | --- |
| Field Name | Data Type | Constraint | Description |
| Uname | Varchar(50) | Not Null | User Name |
| Upassword | varchar(15) | Not Null | User password |
| Umobileno | numaeric(15) | Allow null | User mobile no |
| Uaddress | Nvarchar(50) | Allow null | User address |
| Dob | nvarchar(10) | Not null | Date of Birth |
| Gender | varchar(50) | Not Null | Gender |
| C\_code | Int | Not Null | Country code |
| S\_code | Int | Not Null | State code |
| IS\_code | Int | Not Null | Industry code |
| O\_city | nvarchar(50) | Not Null | Other city |
| Utexpyr | varchar(10) | Not Null | User Total experience year |
| Utexpmt | Varchar(10) | Not Null | User Total experience month |
| Email\_id | Nvarchar(50) | Not Null | Email Id |
| Uquification | Varchar(50) | Not null | User qualification |
| J\_title | Varchar(10) | Not null | Job title |
| Last date | Date & time | Not Null | Display last date |
| Status | Nvarchar(15) | Not null | Display status |
| Created By | Varchar(50) | Not null | Created user name |
| Created Date | Date & time | Not null | Created date |
| Modified By | Varchar(50) | Not null | Modified user name |
| Modified Date | Date & time | Not null | Modified date |

**6.2. Employee:**

**6.2.1. emp\_request**

|  |  |  |  |
| --- | --- | --- | --- |
| Field Name | Data Type | Constraint | Description |
| Uname | Varchar(50) | Not Null | User Name |
| Upassword | varchar(15) | Not Null | User password |
| Umobileno | numaeric(15) | Allow null | User mobile no |
| Uaddress | Nvarchar(50) | Allow null | User address |
| Dob | nvarchar(10) | Not null | Date of Birth |
| Gender | varchar(50) | Not Null | Gender |
| Uemailid | nvachar(15) | Not null | User email id |
| C\_name | Nvarchar(15) | Not Null | Company name |
| I\_code | Int | Not null | Industry code |
| C\_code | Int | Not null | City code |
| Created By | varchar(50) | Not null | Created user name |
| Created Date | Date & time | Not null | Created date |
| Modified By | varchar(50) | Not null | Modified user name |
| Modified Date | Date & time | Not null | Modified date |

**6.3. Admin:**

**6.3.1. Admininfo**

|  |  |  |  |
| --- | --- | --- | --- |
| Field Name | Data Type | Constraint | Description |
| Uname | Varchar(50) | Not Null | User Name |
| Upassword | varchar(15) | Not Null | User password |
| Created By | Varchar(50) | Not Null | Created user name |
| Created Date | Date &Time | Not Null | Created date |
| Modified By | Varchar(50) | Not Null | Modified user name |
| Modified Date | Date &Time | Not Null | Modified date |

**6.4. Master Tables:**

**6.4.1. country\_mst**

|  |  |  |  |
| --- | --- | --- | --- |
| Field Name | Data Type | Constraint | Description |
| C\_name | Varchar(50) | Not Null | Country Name |
| C\_code | Int | Not Null | Country code |
| Created By | Varchar(50) | Not Null | Created user name |
| Created Date | Date &Time | Not Null | Created date |
| Modified By | Varchar(50) | Not Null | Modified user name |
| Modified Date | Date &Time | Not Null | Modified date |

**6.4.2. state\_mst**

|  |  |  |  |
| --- | --- | --- | --- |
| Field Name | Data Type | Constraint | Description |
| S\_name | Varchar(50) | Not Null | State Name |
| S\_code | Int | Not Null | State code |
| Created By | Varchar(50) | Not Null | Created user name |
| Created Date | Date &Time | Not Null | Created date |
| Modified By | Varchar(50) | Not Null | Modified user name |
| Modified Date | Date &Time | Not Null | Modified date |

**6.4.3.city\_mst**

|  |  |  |  |
| --- | --- | --- | --- |
| Field Name | Data Type | Constraint | Description |
| Cityname | Varchar(50) | Not Null | City Name |
| Citycode | Int | Not Null | City code |
| Created By | Varchar(50) | Not Null | Created user name |
| Created Date | Date &Time | Not Null | Created date |
| Modified By | Varchar(50) | Not Null | Modified user name |
| Modified Date | Date &Time | Not Null | Modified date |

**6.4.4. area\_mst**

|  |  |  |  |
| --- | --- | --- | --- |
| Field Name | Data Type | Constraint | Description |
| A\_name | Varchar(50) | Not Null | Area Name |
| A\_code | Int | Not Null | Area code |
| Created By | Varchar(50) | Not Null | Created user name |
| Created Date | Date &Time | Not Null | Created date |
| Modified By | Varchar(50) | Not Null | Modified user name |
| Modified Date | Date &Time | Not Null | Modified date |

**6.4.5. user type\_mst**

|  |  |  |  |
| --- | --- | --- | --- |
| Field Name | Data Type | Constraint | Description |
| U type id | Int | Not Null | Type of user id |
| U type name | Varchar(50) | Not Null | Type of user name |
| Created By | Varchar(50) | Not Null | Created user name |
| Created Date | Date &Time | Not Null | Created date |
| Modified By | Varchar(50) | Not Null | Modified user name |
| Modified Date | Date &Time | Not Null | Modified date |

**6.4.6. sub\_category**

|  |  |  |  |
| --- | --- | --- | --- |
| Field Name | Data Type | Constraint | Description |
| S\_cate | Varchar(50) | Not Null | Sub category |
| A\_code | Int | Not Null | Area code |
| Created By | Varchar(50) | Not Null | Created user name |
| Created Date | Date &Time | Not Null | Created date |
| Modified By | Varchar(50) | Not Null | Modified user name |
| Modified Date | Date &Time | Not Null | Modified date |

**Chapter 9**

**CONCLUSION**

* This project is best suited for any company as it handles whole recruitment process itself. The manager can just put merge this project with his company’s website and post the job vacancies on it. The users who visit the website will be able to see the vacancies and if interested then can apply for the job by registering themselves and sending their resumes.
* The special part of this project is at the admin side where the admin are divided in layers so the lower level admin cannot affect database without permission. The layers at admin side is one of the interesting and important thing of this project which shows the security of this system. However, the lower level admin can send the requests to the higher level admin, only after his approval, the lower level admin will be able to make changes in database, till that time, the database will not be reflected for that request.
* The users’ all details are saved here, so it becomes easy for the admin to see the applicants in any way, eg the users applying for particular job can be seen at a time. This adds the flexibility to the system. The user once registers, he will be given his status for the job he has applied that whether he is eligible for that job. If the user is not eligible for the job he has applied then also he will remain registered for that job so that when he updates his profile and adds his new qualification which fulfills the criteria for the job he had applied, he will become qualified for that and he will be given that message that he is now eligible for the job and now has become qualified user for that job.
* Another very good thing noted here is the architecture followed. The architecture is so lengthy and tuff that it becomes difficult to hack any request and so security is at the peak level. There are 7 layers in the architecture which make the whole functionality fully dynamic so flexibility for the developer also increases though the architecture is difficult.

**Chepter 10**

**REFERENCES**

**10.1 BOOKS**

* BLACK BOOK
* MAGIC OF ASP.NET
* ASP.NET PROFFESIONAL (WROX)

**10.2 BIBLIOGRAPHY**

* [**www.99galaxy.com**](http://www.99galaxy.com)
* [**www.w3schools.com**](http://www.w3schools.com)
* **www.sourceforge.net**
* [**www.naukari.com**](http://www.naukari.com)
* **www.timesjob.com**
* **www.ajax.org**