**INTRODUCTION**

Civil Registry is the online system or agency to help the Indian citizens to apply for there government records like passport, driving license, voter’s ID card, PAN card etc... And register certificates like birth, death, marriage etc.

The primary objective of this web site is to give awareness about the government or legal documents and its registration details as well as to help to register or apply for those documents. This also acts as a consultancy agency to assist the public. The main purpose of the web site is to reduce the effort by the candidate and save his time and avoid unwanted rushes at the government offices and assure a smooth working schedule at government offices.

The project Civil Registry still requires more development of IT solutions and its applications to improve the issuance of copies of government certificates and legal documents. Civil registry team trying to get more affiliation to government offices and departments. Now civil registry team acting as a agency to help the public.

**SYSTEM SPECIFICATIONS**

**HARDWARE DESCRIPTION**

The selection of hardware is very important in the existence and proper working of any software. When selecting hardware, the size and requirements are also important.

Minimum Requirements:

Processor : Pentium II class, 450MHz

RAM : 128MB

Hard Disk Drive : 3GB

Video : 800X600, 256 colors

CD-ROM : Required

The proposed System is developed on:

Processor : INTEL Pentium 4

RAM : 512MB

Hard Disk Drive : 40GB

Key Board : Standard 101/102 or Digi Sync Family

Monitor : Display Panel (1024 X 764)

Display Adapter : Trident Super VGA

Network Adapter : SMC Ethernet Card Elite 16 Ultra

Mouse : Logitech Serial Mouse

**SOFTWARE DESCRIPTION**

Operating System : Windows XP

Front- End : C#. NET with ASP. NET

Back- End : MS SQL SERVER 2005 EXPRESS

Project will be done in ASP. NET with C# 2.0 as front end and SQL Server 2005 as back end. Microsoft .NET is software that connects information, people, systems and devices. It spans clients, servers and developer tools and consists of:

* The .NET Framework programming model that enables developers to build Web-based applications which expose their functionality programmatically over a network using standard protocols such as SOAP and HTTP.
* Developer tools such as Microsoft Visual Studio .NET, which provide a rapid application integrated development environment for programming with the .NET Framework.
* A set of servers including Microsoft Windows 2000, Microsoft SQL, Server and Microsoft BizTalk Server that integrates, runs, operates and manages XML Web services and applications.
* Client software such as Windows XP, Windows CE and Microsoft Office XP that helps developers deliver a deep and compelling user experience across a family of devices and existing products.

The .NET Framework is the programming model of the .NET environment for building, deploying and running Web- based applications, smart client applications and XML Web services. It manages much of the plumbing, enabling developers to focus on writing the business logic code for their applications. The .NET Framework includes the common language runtime and class libraries.

**ORGANIZATIONAL PROFILE**

**Sristhi** Systems is a Software and Web Development Company which provides technology solutions and professional software for clients worldwide. Srishti focus on quality, innovations and leverage deep industry and functional expertise to help customers to transform their highest value business processes.

Our in-depth technical knowledge coupled with industry experience and the unique methodology enables us to successfully deliver the services to the clients worldwide. We are driven to attain our customer’s highest satisfaction by providing business experts with underlying delivery expertise in Enterprise Technology, Methodologies and Implementations. Our core competencies are designed to effectively deliver results to our customers.   
  
  Sristhi Training Center provides premier instructor-led IT training through a carefully balanced blend of hands-on lab exercises and lecture.

Training is conducted in the comfortable classrooms at our training facility and client locations. Experienced instructors make training productive and enjoyable. Customizable courses and computer lab offer additional options fro your training needs. We also provide project guidance for all academic students which help them to build a successful portfolio and start their IT journey successfully.

**OVERVIEW OF THE LANGUAGE USED**

**MICROSOFT VISUAL STUDIO**

Microsoft Visual Studio is Microsoft’s flagship software development product for computer programmers. It centers on an integrated distribution environment which has programmers create stand alone, and web services that run on any platforms supported by Microsoft’s .Net Framework (for all versions after 6). Supported platforms include Microsoft windows, servers and workstations, Pocket PC, Smart Phones and World Wide Web browsers not the Java Virtual Machine that all other java tools target.

**VISUAL STUDIO 2005**

The most important language added in this version was the introduction of generics, which are similar in many aspects to C++ templates. This potentially increases the number of bugs caught at compile- time instead of run- time by encouraging the use of strict type checking in areas where it was not possible before, C++ also got a similar upgrade with the addition of C++/CLI which is slated to replace the use of Managed C++. Other new features of Visual Studio 2005 include the “Development Designer” which allows application designs to be validated before deployments, an improved environment for web publishing when combined with ASP.NET 2.0 and load testing to see application performance under various sorts of user loads. Visual Studio 2005 also added extensive 64- bit support. While the development environment itself only available as a 32- bit application, visual C++ 2005 supports compiling for (x64AMD64 and EM64T) as well as IA- 64 (Itanium). The platforms SDK include 64- bit and 64-bit versions of the libraries.

**NET FRAMEWORK SDK**

The .NET framework is an integral windows component that supports building and running the next generation of applications and XML web services. The key component of the .NET frame work are the common language run time and the . NET frame work class library, which includes ADO.NET, ASP.NET and windows forms. The .NET framework provides a managed execution environment simplified development and deployment and integration with a wide variety of programming languages.

This framework is made up of the following parts:

* The common language runtime(CLR)
* The base class libraries.
* Object oriented internet development with ASP.NET
* Rich client user interface using windows forms
* RAD for the internet using web forms

**OVERVIEW OF THE .NET FRAME WORK**

The .NET framework is a new computing platform that simplifies application development in the highly distributed environment of the internet. The .NET framework is designed to fulfill following objectives:

* To provide a consistent object oriented programming environment whether object code is stored and executed locally but internet- distributed or executed remotely.
* To provide a code execution environment that minimizes software deployment and versioning conflicts.
* To provide a code execution environment that guarantees safe execution of code, including code created by an unknown or semi trusted third party.
* To provide a code execution environment that eliminates the performance problem of scripted or interpreted environments.
* To make the developer experience consistent across widely types of application, such as windows based applications and web based applications.
* To build all communication on industry standards to ensure that code based on the .NET framework can integrate with any other code.

The .NET framework has two main components: the common language runtime and the .Net framework class library. The common language runtime is the foundation of the .NET framework. You can think of the runtime as an agent that manages code at execution time, and remoting while also enforcing strict type safely and other forms of code accuracy that ensure security and robustness in fact the concept of code management is a fundamental principle of the runtime.

Code that targets the runtime is known as managed code, while code that does not target the runtime is known as un managed code. The class library, the other main component of the .NET frameworks is a comprehensive, object-oriented collection reusable types that you can use to develop applications ranging from traditional command line or graphical user interface (FGUI) applications to application base d on the latest innovations provided by ASP.NET, such as web forms and XML web services.

The .NET framework can be hosted by unmanaged component that load the common language runtime into their processes and initiate the execution of managed code. ASP.NET works directly with the runtime to enable ASP.NET application and XML web services, both of which are discussed later in this topic, Internet explorer is an example of unmanaged application that hosts the runtime (in the form of a MIME type extension). Using internet explorer to the host runtime enables you to embed managed components or windows forms controls in HTML documents. Hosting the runtime in this way makes mobile code 9similar to Microsoft Active Xr controls) possible, but with significant improvement that only managed code can offer, such as semi-trusted execution and secure isolated file storage.

The following illustration shows the relationship of the common language runtime and the class library to your application and to the over all system. The illustration also shows how managed code operated with in a larger architecture.

We can use the .NET framework to develop the following types of application and services:

* Console applications
* Window GUI application (Windows Forms) ASP.NET applications
* XML Web services
* Windows services

**COMMON LANGUAGE RUNTIME (CLR)**

The common language runtime (CLR) is responsible for runt-time services such as language integration; security enforcement; and memory, process and thread management. In addition, it has a roll at development time when features such as life cycle management strong type naming, cross-language exception handling, dynamic binding and so on, reduce the amount of code that a developer must write to turn the business logic the reusable component. The runtime can be hosted by high performance, server-side applications, such a s Microsoft Internet Information Services (IIS) for building web applications with ASP.NE and the next release of Microsoft SQL Server. This infrastructure enables you to use code “managed “ by the .NET framework to write your business logic, while still enjoying the superior performance of the industry’s best enterprises servers that support runtime hosting.

**ASP.NET**

ASP.NET is a set of Microsoft.NET framework technologies used for building web applications and XML Web services. ASP.NET page execute on the server and generate mark up such as HTML, WML or XML that is sent to a desktop or mobile browser. ASP.NET pages use a compiled, event-driven programming model that improves performance and enables the separation of application logic and user interface. Both ASP.NET pages and ASP.NET web services files contain server-side (as opposed to client side logic) written in Visual basic .NET, C#.NET or any .NET compatible language, Web applications and XML Web Services take advantage of the features of the common language runtime, such as type safety, inheritance, language, interoperability, versioning, and integrated security.

**IIS**

Microsoft Internet Information S4ervices (IIS; formerly called Server is a set of Internet-based services for servers using Microsoft Windows. It is the world’s second most popular web server in terms of overall websites. As of September 2007 it served 34.94% of all websites and 36.63% of all active websites according to Net craft. The servers currently include FTP, SMTP, NNTOP, and HTTP/HTTPS.

**ADO.NET**

ADO.NET provides consistent access to data sources such a Microsoft SQL Server and XML, as well as to data sources exposed through OLE DB and ODBC. Data sharing consumer applications can use ADO.NET to connect to these data sources and retrieve, manipulate and update the data that they contain.

ADO.NET seperates data access from data manipulation into discrete components that can be used separately or in tandem. ADO.NET includes .NET Frame work data providers for connecting to a database, executing commands and retrieving results. Those results are either processed directly, placed in and ADO.NET Dataset objects in order to be exposed to the used in an ad hoc manner, combined with data from multiple sources or remoted between tiers. The ADO.NET Dataset object can also be used independently of a .NET Framework data provider to manage data local to the application or sourced from XML.

The ADO.NET classes are found in System.Data.dll and are integrated with the XML classes found in System.Xml.dll. When compiling code that uses the System. Data, namespace reference both System.Data.dll and System.Xml.dll.

ADO.NET provided functionality to developers writing managed code similar to the functionality provided to native component object model (COM) developers by ActiveX Data Objects (ADO).

**ADO.NET COMPONENTS**

There are two components of ADO.NET that you can use to access and manipulate data:

* .NET Framework data providers.
* The DataSet

**.NET FRAMEWORK DATA PROVIDERS**

The .NET Framework Data providers are components that have been explicitly designed for data manipulation and fast, forward-only, read-only access to data. The connection object provides connectivity to a data source. The command object enables access to database commands to return data, modify data, run stored procedures and send or retrieve parameter information. The Data Adapter provides a high-performance stream of data from the data source. Finally, the Data Adapter provides the bridge between the DataSet object and the data source. The Data Adapter uses command object to execute SQL commands at the data source to both load the DataSet with data and reconcile changes made to the data in the DataSet back to the data source.

**THE DATASET**

The ADO.NET DataSet is explicitly designed for data access independent of any data source. As a result, it can be used with multiple and differing data sources used with XML data or used to manage data local to the application. The DataSet contains a collection n of one or more DataTable objects made up to rows and columns of data as well as primary key, foreign key, constraint and relation information about the data in the DataTable objects.

**BENEFITS OF ADO.NET**

ADO.NET offers several advantages over previous versions of ADO and over other data access components. These benefits fall into the following categories:

1. Interoperability
2. Maintainability
3. Programmability
4. Salability

**MICROSOFT DATA ACCSS COMPONENTS (MDAC)**

Microsoft Data Access Components (MDAC) is a collection of core files provided to help applications by providing a means of accessing data. MDAC includes core files for Open Database Connectivity (ODBC), ActiveX Data Objects (ADO), OLEDB, Network libraries and client configuration tool for SQL Server. Depending on your data access strategy, you may need to ensure that MDAC is installed on the client computers, the business servers, the Web servers or the database servers. MDAC 2.6 or later is required by the .NET Framework and at least MDAC 2.7 SP1 is recommended.

**SYSTEM ANALYSIS**

**DEFINITION**

System Analysis is the detailed study of the various operations performed by the system and their relationships within and outside the system. Analysis is the process of breaking something into its parts so that the whole may be understood. System analysis is concerned with becoming aware of the problem, identifying the relevant and most decisional variables, analyzing and synthesizing the various factors and determining an optimal or at least a satisfactory solution. During this a problem is identified, alternate system solutions are studied and recommendations are made about committing the resources used to design the system

**DESCRIPTION OF PRESENT SYSTEM**

Existing system refers to the system that is being follow till now. Presently all the registrations are done manually. If a person wants to make registrations like birth, death, marriage etc he should directly contact the corresponding office. The main disadvantage is that there will be lot of difficulties for the citizens. So, all these procedures will be a time consuming one.

**LIMITATIONS OF PRESENT SYSTEM**

* Difficult for persons.
* Time consuming.

To avoid all these limitations and make the working more accurately the system needs to be computerized.

**PROPOSED SYSTEM**

Civil Registry is aimed at developing a web-based system. In this system the person can register online and do many things. The details of all the things are made available to them through the website.

**ADVANTAGES**

* This website provides online help for legal queries.
* This website helps all the users to view the registration.
* The user can post thread in the forum.
* The system is user friendly.

**FEASIBILTY STUDY**

A feasibility analysis usually involves a through assessment of the operational(need), financial and technical aspects of a proposal. Feasibility study is the test of the system proposal made to identify whether the user needs may be satisfied using the current software and hardware technologies, whether the system will be cost effective from a business point of view and whether it can be developed with the given budgetary constraints. A feasibility study should be relatively cheap and done at the earliest possible time. Depending on the study, the decision is made whether to go ahead with a more detailed analysis.

When a new project is proposed, it normally goes through feasibility assessment. Feasibility study is carried out to determine whether the proposed system is possible to develop with available resources and what should be the cost consideration. Facts considered in the feasibility analysis were.

* Technical Feasibility
* Economic Feasibility
* Behavioral Feasibility

**Technical Feasibility**

Technical Feasibility deals with the hardware as well as software requirements. Technology is not a constraint to type system development. We have to find out whether the necessary technology, the proposed equipments have the capacity to hold the data, which is used in the project, should be checked to carryout this technical feasibility.

The technical feasibility issues usually raised during the feasibility stage of investigation includes these

* This software is running in windows 2000 Operating System, which can be easily installed.
* The hardware required is Pentium based server.
* The system can be expanded.

**Economical Feasibility**

This feasibility study present tangible and intangible benefits from the prefect by comparing the development and operational cost. The technique of cost benefit analysis is often used as a basis for assessing economic feasibility. This system needs some more initial investment than the existing system, but it can be justifiable that it will improve quality of service.

Thus feasibility study should center along the following points:

* Improvement resulting over the existing method in terms of accuracy, timeliness.
* Cost comparison
* Estimate on the life expectancy of the hardware
* Overall objective

Our project is economically feasible. It does not require much cost to be involved in the overall process. The overall objectives are in easing out the requirement processes.

**Behavioral/ Operational Feasibility**

This analysis involves how it will work when it is installed and the assessment of political and managerial environment in which it is implemented. People are inherently resistant to change and computers have been known to facilitate change. The new proposed system is very much useful to the useful to the users and there for it will accept broad audience from around the world.

**SYSTEM DESIGN**

**DEFINITION**

The most creative and challenging face of the system development is System Design. It provides the understanding and procedural details necessary for implementing the system recommended in the feasibility study. Design goes through the logical and physical stages of development.

In designing a new system, the system analyst must have a clear understanding of the objectives, which the design is aiming to fulfill. The first step is to determine how the output is to be produced and in what format. Second, input data and master files have to be designed to meet the requirements of the proposed output. The operational phases are handled through program construction and testing.

Design of a system can be defined as a process of applying various techniques and principles for the purpose of defining a device, a process or a system in sufficient detail to permit its physical realization. Thus system design is a solution to “how to” approach to the creation of a new system. Thus important phase provides the understanding and the procedural details necessary for implementing the system recommended in the feasibility study. The design step provides a data design, architectural design, and a procedural design.

**OUTPUT DESIGN**

In the output design, the emphasis is on producing a hard copy of the information requested or displaying the output on the CRT screen in a predetermined format. Two of the most output media today are printers and the screen. Most users now access their reports from a hard copy or screen display. Computer’s output is the most important and direct source of information to the user, efficient, logical, output design should improve the systems relations with the user and help in decision-making.

As the outputs are the most important source of information to the user, better design should improve the system’s relation and also should help in decision-making. The output device’s capability, print capability, print capability, response time requirements etc should also be considered form design elaborates the way output is presented and layout available for capturing information. It’s very helpful to produce the clear, accurate and speedy information for end users.

**INPUT DESIGN**

In the input design, user-oriented inputs are converted into a computer based system format. It also includes determining the record media, method of input, speed of capture and entry on to the screen. Online data entry accepts commands and data through a keyboard. The major approach to input design is the menu and the prompt design. In each alternative, the user’s options are predefined. The data flow diagram indicates logical data flow, data stores, source and destination. Input data are collected and organized into a group of similar data. Once identified input media are selected for processing.

In this software, importance is given to develop Graphical User Interface (GUI), which is an important factor in developing efficient and user-friendly software. For inputting user data, attractive forms are designed. User can also select desired options from the menu, which provides all possible facilities.

Also the important input format is designed in such a way that accidental errors are avoided. The user has to input only just the minimum data required, which also helps in avoiding the errors that the users may make. Accurate designing of the input format is very important in developing efficient software. The goal or input design is to make entry as easy, logical and free from errors.

**LOGICAL DESIGN**

Logical data design is about the logically implied data. Each and every data in the form can be designed in such a manner to understand the meaning. Logical data designing should give a clear understanding and idea about the related data used to construct a form.

**DATA FLOW DIAGRAM**

A Data Flow Diagram (DFD) is a diagram that describes the flow of data and the processes that change data throughout a system. It’s a structured analysis and design tool that can be used for flowcharting in place of or in association with information. Oriented and process oriented system flowcharts. When analysts prepare the Data Flow Diagram, they specify the user needs at a level of detail that virtually determines the information flow into and out of the system and the required data resources. This network is constructed by using a set of symbols that do not imply physical implementations. The Data Flow Diagram reviews the current physical system, prepares input and output specification, specifies the implementation plan etc.

Four basic symbols are used to construct data flow diagrams. They are symbols that represent data source, data flows, and data transformations and data storage. The points at which data are transformed are represented by enclosed figures, usually circles, which are called nodes.

**DATA FLOW DIAGRAM SYMBOLS:-**

* **Source or Destination of data**
* **Data Flow**
* **Process**
* **Storage**

**Steps to Construct Data Flow Diagrams**

Four steps are commonly used to construct a DFD

* Process should be named and numbered for easy reference. Each name should be representative of the process.
* The destination of flow is from top to bottom and from left to right.
* When a process is exploded in to lower level details they are numbered.
* The names of data stores, sources and destinations are written in capital letters.

**Rules for constructing a Data Flow Diagram**

* Arrows should not cross each other.
* Squares, circles and files must bear names.
* Decomposed data flow squares and circles can have same names.
* Draw all data flow around the outside of the diagram.

**CONTEXT DIAGRAM**

View Details

View Details

Manage

View Details

User Details

User

Administrator

Visitor

**Level 1 DFD- User**

Updating

Add Details

User Details

Details

Confirmation

Validation

User name, Password

User

Registration

Login

Office Files

**Level 1 DFD- Visitor**

View Details

Enquires

Details

Visitor

Database

**Level 1 DFD- Administrator**

Managing

Verifications & Approvals

Login Details

Administrator

Login

User

Visitor

Office Files

**Level 2 DFD- User**

Name, Place

Reg: Details

Time

Add Details

Add Details

Login Details

User

Login

Birth Files

Marriage Files

Death File

Voters Files

Help Desk Files

**Level 2 DFD- Visitor**

Feedback

Enquires

Reg: Details

Marriage Registration, Death Registration

Ads Details

Adding Details

View Details

Visitor

Advertisements

Office Files

Help Desk Files

**TABLE SPECIFICATIONS**

**Table Name: Birthreg**

|  |  |  |
| --- | --- | --- |
| **Field Name** | **Data Type** | **Size** |
| Dob | Date/Time | 50 |
| Sex | Varchar | 50 |
| Name of child | Varchar | 50 |
| Fathers name | Varchar | 50 |
| Mothers name | Varchar | 50 |
| Place of birth | Varchar | 50 |
| Informant name | Varchar | 50 |
| Address of info | Varchar | 50 |
| Date of info | Date/Time | 50 |
| Name of town/village | Varchar | 50 |
| District | Varchar | 50 |
| State | Varchar | 50 |
| Religion | Varchar | 50 |
| Fathersoccu | Varchar | 50 |
| Momoccu | Varchar | 50 |
| Moms age at marriage | Varchar | 50 |
| Moms age at delivery | Varchar | 50 |
| Method of delivery | Varchar | 50 |
| Birth weight | Varchar | 50 |
| Pregnancy duration | Varchar | 50 |

**Table Name: Death**

**Primary Key: S NO**

|  |  |  |
| --- | --- | --- |
| **Field Name** | **Data Type** | **Size** |
| S No | Int |  |
| Applicant | Varchar | 50 |
| Sep date | Date/Time | 10 |
| Place marriage | Varchar | 50 |
| Children | Varchar | 50 |
| Children under | Varchar | 50 |
| Number of child | Varchar | 50 |
| Child name | Varchar | 50 |
| Time comm. With child | Varchar | 50 |
| Financial support | Varchar | 50 |
| Health | Varchar | 50 |
| Education | Varchar | 50 |

**Table Name: Divorce common**

|  |  |  |
| --- | --- | --- |
| **Field Name** | **Data Type** | **Size** |
| S No | Int |  |
| Applicant | Varchar | 50 |
| Sep date | Date/Time |  |
| Place marriage | Varchar | 50 |
| Children | Varchar | 50 |
| Children under | Varchar | 50 |
| Number of child | Varchar | 50 |
| Child name | Varchar | 50 |
| Time comm. With child | Varchar | 50 |
| Financial support | Varchar | 50 |
| Health | Varchar | 50 |
| Education | Varchar | 50 |

**Table Name:Electricity**

|  |  |  |
| --- | --- | --- |
| **Fiel dName** | **Data Type** | **Size** |
| Name of applicant | Varchar | 50 |
| Occupation | Varchar | 50 |
| Per addr | Varchar | 50 |
| Location of supply | Varchar | 50 |
| Purpose | Varchar | 50 |
| Covered area | Varchar | 50 |
| Load details | Varchar | 50 |
| KNO | Varchar | 50 |

**Table Name:Feedback**

|  |  |  |
| --- | --- | --- |
| **Fiel dName** | **Data Type** | **Size** |
| Name | Varchar | 50 |
| Designation | Varchar | 50 |
| Company | Varchar | 50 |
| Area of work | Varchar | 100 |
| Address | Varchar | 50 |
| City | Varchar | 50 |
| State | Varchar | 50 |
| Country | Varchar | 50 |
| Pin | Varchar | 50 |
| Email | Varchar | 50 |
| Telephone | Varchar | 50 |
| Fax | Varchar | 50 |
| Queries | Varchar | 50 |

**Table Name: Forum thread**

|  |  |  |
| --- | --- | --- |
| **Fiel dName** | **Data Type** | **Size** |
| Post ID | Varchar | 50 |
| Name | Varchar | 50 |
| Email | Varchar | 100 |
| Description | Varchar | 50 |
| Post date | Date/Time | 50 |

**Table Name:Login**

**Primary Key: Username**

|  |  |  |
| --- | --- | --- |
| Field Name | Data Type | Size |
| Username | varchar | 50 |
| Password | Varchar | 100 |

**Table Name:Marriage**

|  |  |  |
| --- | --- | --- |
| Field Name | Data Type | Size |
| Date of registration | Date/Time | 50 |
| Bridegroomname | Varchar | 50 |
| Bride name | Varchar | 100 |
| Bridegroom father name | Varchar | 50 |
| Brides father name | Varchar | 50 |
| Bridegroom date of birth | Date/Time | 50 |
| Bride date of birth | Date/Time | 50 |
| Bridegroom present addr | Varchar | 50 |
| Bride present addr | Varchar | 50 |
| Bridegroom permanent addr | Varchar | 50 |
| Bride permanent addr | Varchar | 50 |
| Bridegroom voter no | Varchar | 50 |
| Bride voter no | Varchar | 50 |
| Bridegroom marital status | Varchar | 50 |
| Bride marital status | Varchar | 50 |
| Date of marriage | Date/Time | 50 |
| Place of marriage | Varchar | 50 |

**Table Name:Name Change**

|  |  |  |
| --- | --- | --- |
| **Fiel dName** | **Data Type** | **Size** |
| Title | Varchar | 50 |
| PAN No | int | 50 |
| Name | Varchar | 50 |
| New title | Varchar | 200 |
| New name | Varchar | 50 |
| New nationality | Varchar | 100 |
| New status | Varchar | 100 |
| Permanent Account | Varchar | 100 |
| New Address | Varchar | 100 |
| State | Varchar | 50 |
| Country | Varchar | 50 |
| Proof | Varchar | 50 |

**Table Name:PAN**

|  |  |  |
| --- | --- | --- |
| **Field Name** | **Data Type** | **Size** |
| Name | Varchar | 50 |
| Card name | Varchar | 50 |
| Father name | Varchar | 50 |
| Address | Varchar | 50 |
| Offaddress | Varchar | 50 |
| Sex | Varchar | 50 |
| App stat | Varchar | 50 |
| Dob | Date/Time | 100 |
| Regno | Varchar | 50 |
| Citizen | Varchar | 50 |
| Address rep | Varchar | 50 |

**Table Name: Passport**

|  |  |  |
| --- | --- | --- |
| **Field Name** | **Data Type** | **Size** |
| Fee | Varchar | 50 |
| Cash /DD | Varchar | 50 |
| Bank code | Varchar | 50 |
| DD No | Varchar | 50 |
| Date of issue of DD | Date/Time | 50 |
| Name of applicant | Varchar | 50 |
| Changed name | Varchar | 50 |
| Sex | Varchar | 50 |
| Date of birth | Date/Time | 50 |
| Place of birth | Varchar | 50 |
| Father name | Varchar | 50 |
| Mother name | Varchar | 50 |
| Spouse name | Varchar | 50 |
| Present addr | Varchar | 50 |
| Date since residing | Date/Time | 50 |
| Per addr | Varchar | 50 |
| Email | Varchar | 50 |
| File no | Varchar | 50 |
| Date of application | Date/Time | 50 |
| Place of application | Varchar | 50 |
| Education qualification | Varchar | 50 |
| Profession | Varchar | 50 |
| Visible mark | Varchar | 50 |
| Height | Varchar | 50 |
| Working in central or state govt | Varchar | 50 |
| Citizen of india | Varchar | 50 |
| Emigration checkout | Varchar | 50 |
| Elligible category | Varchar | 50 |
| Mother file no | Varchar | 50 |
| Mother date of issue | Date/Time | 50 |
| Mother place of issue | Varchar | 50 |
| Father file no | Varchar | 50 |
| Father date of issue | Date/Time | 50 |
| Father place of issue | Varchar | 50 |

**Table Name: Post Reply**

|  |  |  |
| --- | --- | --- |
| **Field name** | **Data type** | **Description** |
| Post id | Int |  |
| Name | Varchar | 50 |
| Description | Varchar | 50 |
| Email | Varchar | 50 |
| Reply date | Date/Time |  |

**Table Name:Ration Card**

|  |  |  |
| --- | --- | --- |
| **Field name** | **Data type** | **Description** |
| S No | int |  |
| Name of applicant | Varchar | 50 |
| Fathers name | Varchar | 50 |
| Present addr | Varchar | 50 |
| Permanent addr | Varchar | 50 |
| Occupation | Varchar | 50 |
| Name of employer | Varchar | 50 |
| Adults | Varchar | 50 |
| Minor | Varchar | 50 |
| Electrolno | Varchar | 50 |
| Trading liscence | Varchar | 50 |
| Lpg consumer | Varchar | 50 |

**Table Name: Ration Relation**

|  |  |  |
| --- | --- | --- |
| **Field name** | **Data type** | **Description** |
| S No | Int |  |
| Name 1 | Varchar | 50 |
| Age 1 | Int |  |
| Sex 1 | Varchar | 50 |
| Relationship1 | Varchar | 50 |
| Name2 | Varchar | 50 |
| Age2 | Int |  |
| Sex2 | Varchar | 50 |
| Relationship2 | Varchar | 50 |
| Name3 | Varchar | 50 |
| Age3 | Int |  |
| Sex3 | Varchar | 50 |
| Relationship3 | Varchar | 50 |
| Name4 | Varchar | 50 |
| Age4 | Int |  |
| Sex4 | Varchar | 50 |
| Relationship4 | Varchar | 50 |
| Name5 | Varchar | 50 |
| Age5 | Int |  |
| Sex | Varchar | 50 |
| Relationship5 | Varchar | 50 |

**Table Name: Registration**

**Primary Key: Username**

|  |  |  |
| --- | --- | --- |
| **Field name** | **Data type** | **Description** |
| First name | Varchar | 50 |
| Last name | Varchar | 50 |
| User name | Varchar | 50 |
| Password | Varchar | 50 |
| Email | Varchar | 50 |
| location | Varchar | 50 |

**Table Name: Telapply**

|  |  |  |
| --- | --- | --- |
| Field name | Data type | Description |
| Company/organization | Varchar | 50 |
| Name of customer/comp | Varchar | 50 |
| Name of joint application | Varchar | 50 |
| Name of nominee | Varchar | 50 |
| Relationship | Varchar | 50 |
| Name of father/hus/partner | Varchar | 50 |
| PAN No | Int |  |
| Working telephone No | Int |  |
| Nearest teleph No | Int |  |
| Postal addr | Varchar | 50 |
| Pin code | Int |  |
| Billing addr | Varchar | 50 |
| Email | Varchar | 50 |
| Facilities required | Varchar | 50 |
| Telephone instrument | Varchar | 50 |
| Payment mode | Varchar | 50 |
| Payment details | Varchar | 50 |

**Table Name:Water1**

|  |  |  |
| --- | --- | --- |
| **Field name** | **Data type** | **Description** |
| Name of applicant | Varchar | 50 |
| Father/hus name | Varchar | 50 |
| Postal addr | Varchar | 50 |
| Survey/plot no | Int |  |
| Building owned by applicant | Varchar | 50 |
| Residential/commercial | Varchar | 50 |
| Internal installation | Varchar | 50 |
| Road cutting required | Varchar | 50 |
| Length of pipe | Varchar | 50 |

**CODING**

First phase of implementation is coding. Coding can be done in two ways. One by automatic program code and other by programmer’s manually written code. A code generator is a suite of programs that matches the input to an appropriate code template and from these produces modules of code. The code is made simple in such a way that another programmer can easily understand and work on that in future. The crucial phase in the system development life cycle is the successful implementation of the new system design. The process of converting as new or revised system into an operational one is known as system implementation. This includes all those activities that take place to convert from an old system to a new system. The system can be implemented only after a through testing is done and if it is found to work according to the specifications. The most crucial stage in achieving a new successful system and giving confident on the new system for the users is that it will work effectively and efficiently. If involves careful planning, investigation of the current system and its constraints on implementation, design of methods to achieve the change over.

**TESTING**

Testing is a process to show the correctness of the program. Testing is needed to show completeness, it improve the quality of the software and to provide the maintenance aid. Some testing standards are therefore necessary reduce the testing costs and operation time. Testing software extends throughout the coding phase and it represents the ultimate review of configurations, design and coding. Based on the way the software reacts to these testing, we can decide whether the configuration that has been built is study or not. All components of an application are tested, as the failure to do so many results in a series of bugs after the software is put to use.

**Blackbox Testing**

Blackbox testing, also called behavioral testing, focuses on the functional requirements of software. This testing approach enables the software engineer to derive the input conditions that will fully exercise all requirements for a program. Blackbox testing attempts to find the errors like

* Incorrect or missing functions
* Interface errors
* Errors in datastructures or external database access
* Behavior or performance errors
* Initialization and termination errors

In Blackbox testing software is exercised over a full range of inputs and outputs are observed for correctness.

**Whitebox Testing**

Whitebox testing is also called Glassbox testing is a test case design control; structure of the procedural design to derive test cases using Whitebox testing method, the software engineer can derive the test cases that guarantee that all independent paths within the module have been exercised at least once. Exercise all logic decisions on their true or false sides. Execute all loops at their boundaries and within their operational bounds. Exercise internal datastructure to ensure their validity.

**Software Testing Strategies**

Testing involves

* Unit testing
* Integration testing
* Acceptance testing

The first level of test is unit testing. The purpose of unit testing is to ensure that each program is fully tested.

The second step is integration testing. In this individual program units or programs are integrated and tested as a complete system to ensure that the software requirements are met.

Acceptance Testing involves planning and the execution of various types of tests in order to demonstrate that the implemented software system satisfies the requirements. Finally our project meets the requirements after going through all the levels of testing.

**SECURITY**

The system security problem can be divided into four related issues: security, integrity, privacy and confidentiality. They determine the file structure, data structure and access procedures.

System security refers to the technical innovations and procedures applied to the hardware and operating systems to protect against deliberate or accidental damage from a defined threat. In contrast, data security is the protection of data from loss, disclosure, modifications and destruction.

System integrity refers to the proper functioning of programs, appropriate physical security and safety against external threats such as eavesdropping and wiretapping. In comparison, data integrity makes sure that do not differ from original from others and how the organization can be protected against unwelcome, unfair or excessive dissemination of information about it.

The term confidentiality is a special status given to sensitive information in a data base to minimize the possible invasion of privacy. It is an attribute of information that characterizes its need for protection. System security is the technical means of providing such protection. In contrast privacy is largely a procedural matter of how information is used.

**SOURCE CODE**

**Account.aspx.cs**

using System;

using System.Data;

using System.Configuration;

using System.Collections;

using System.Web;

using System.Web.Security;

using System.Web.UI;

using System.Web.UI.WebControls;

using System.Web.UI.WebControls.WebParts;

using System.Web.UI.HtmlControls;

using System.Web.Configuration;

using System.Data.SqlClient;

public partial class Default2 : System.Web.UI.Page

{

SqlConnection con = new SqlConnection(WebConfigurationManager.ConnectionStrings["civilRegDB"].ConnectionString);

SqlCommand cmd = new SqlCommand();

SqlDataReader dr;

int x;

protected void Page\_Load(object sender, EventArgs e)

{

con.Open();

cmd.Connection = con;

}

protected void btnReg\_Click(object sender, EventArgs e)

{

if (txtConfirmPass.Text == "" || txtEmail.Text == "" || txtFname.Text =="" || txtLname.Text == "" || txtLocation.Text == "" || txtLoginName.Text == "" || txtPass.Text == "")

{

lblMsg.Text = "Missing Fields! Please fill up all";

lblMsg.Visible = true;

lblAvail.Visible = false;

}

else

{

cmd.CommandText = "select max(SlNo) from registration";

dr = cmd.ExecuteReader();

if (dr.Read())

{

x = dr.GetInt32(0);

x = x + 1}

dr.Close();

cmd.CommandText = "INSERT INTO REGISTRATION VALUES (" + x + ",'" + txtFname.Text +

"','" + txtLname.Text + "','" + txtLoginName.Text +

"','" + txtPass.Text + "','" + txtEmail.Text +

"','" + txtLocation.Text + "')";

cmd.ExecuteNonQuery();

cmd.CommandText = "insert into login values('" + txtLoginName.Text + "','" + txtPass.Text + "')";

cmd.ExecuteNonQuery();

lblMsg.Text = "You have successfully Registered";

lblMsg.Visible = true;

lblAvail.Visible = false;

}

}

protected void btnReset\_Click(object sender, EventArgs e)

{

txtConfirmPass.Text = "";

txtEmail.Text = "";

txtFname.Text = "";

txtLname.Text = "";

txtLocation.Text = "";

txtLoginName.Text = "";

txtPass.Text = "";

lblMsg.Visible = false;

btnReg.Enabled = false;

}

protected void btnChkAvailable\_Click(object sender, EventArgs e)

{

cmd.CommandText = "SELECT \* FROM REGISTRATION WHERE USERNAME='" + txtLoginName.Text + "'";

dr=cmd.ExecuteReader();

if (dr.Read())

{

lblAvail.Text = "Please Select another name!";

lblAvail.Visible = true;

}

else

{

lblAvail.Text = "Available, You can continue with this name";

lblAvail.Visible = true;

btnReg.Enabled = true;

} }}

**Adminbirth.aspx.cs**

using System;

using System.Data;

using System.Configuration;

using System.Collections;

using System.Web;

using System.Web.Security;

using System.Web.UI;

using System.Web.UI.WebControls;

using System.Web.UI.WebControls.WebParts;

using System.Web.UI.HtmlControls;

public partial class Default2 : System.Web.UI.Page

{

protected void Page\_Load(object sender, EventArgs e)

{

}

protected void GridView1\_RowEditing(object sender, GridViewEditEventArgs e)

{

Application.Set("oldvalue", GridView1.Rows[e.NewEditIndex].Cells[0].Text);

}

protected void GridView1\_RowUpdating(object sender, GridViewUpdateEventArgs e)

{

String old = Application.Get("oldvalue").ToString();

String dob = e.NewValues[0].ToString();

String sex= e.NewValues[1].ToString();

String Childname = e.NewValues[2].ToString();

String fathersname = e.NewValues[3].ToString();

String mothersname = e.NewValues[4].ToString();

String sql = "Update birthreg set dob='" + dob + "',sex='" + sex +"',nameofchild='" + Childname + "',fatherssname='" + fathersname + "' where SlNo=" + old + "";

SqlDataSource1.UpdateCommand = sql;

GridView1.DataBind();

lblmsg.Text = "A Birth Registration Successfully Updated!!";

}

protected void GridView1\_RowDeleting(object sender, GridViewDeleteEventArgs e)

{

/\* string x;

x = GridView1.Rows[e.RowIndex].Cells[0].Text;

SqlDataSource1.DeleteCommand = "delete from birthreg where SlNo='" + x + "'";

GridView1.DataBind();

lblmsg.Text = "BirthRegistration Successfully Deleted!!";\*/ }}

**Adminlog.aspx.cs**

using System;

using System.Data;

using System.Configuration;

using System.Collections;

using System.Web;

using System.Web.Security;

using System.Web.UI;

using System.Web.UI.WebControls;

using System.Web.UI.WebControls.WebParts;

using System.Web.UI.HtmlControls;

public partial class Default2 : System.Web.UI.Page

{

protected void Page\_Load(object sender, EventArgs e)

{

}

protected void Login1\_Authenticate(object sender, AuthenticateEventArgs e)

{

if (Login1.UserName == "admin" && Login1.Password == "registry")

{

Response.Redirect("admin2.aspx");

}

else

{

Response.Write("Invalid ID");

}

}

}

**Birth.aspx.cs**

using System;

using System.Data;

using System.Configuration;

using System.Collections;

using System.Web;

using System.Web.Security;

using System.Web.UI;

using System.Web.UI.WebControls;

using System.Web.UI.WebControls.WebParts;

using System.Web.UI.HtmlControls;

using System.Data.SqlClient;

public partial class Default3 : System.Web.UI.Page

{

SqlConnection con;

SqlCommand cmd;

SqlDataReader dr;

int xx;

protected void Page\_Load(object sender, EventArgs e)

{

con = new SqlConnection(ConfigurationManager.ConnectionStrings["civilRegDB"].ConnectionString);

con.Open();

cmd= new SqlCommand();

cmd.Connection = con;

}

protected void btnsubmit\_Click(object sender, EventArgs e)

{

string dob = Request.Form.Get("dob");

string tdate = Request.Form.Get("date");

int x;

if (dob == "" || drop\_sex.Text == "" || txtnameofchild.Text == "" ||

txtnameoffather.Text == "" || txtnameofmother.Text == "" || txtplaceofbirth.Text == "" ||

txtinformantsname.Text == "" || txtaddressofinformant.Text == "" || tdate == "" ||

txtnameofvillage.Text == "" || txtnameofdistrict.Text == "" || txtnameofstate.Text == "" ||

txtreligion.Text == "" || txtofccupation.Text == "" || txtmoccupation.Text == "" ||

txtmage.Text == "" || txtageatbirthtime.Text == "" || txtmethod.Text == "" || txtweight.Text == "" ||

txtduration.Text == "")

{

lbl\_msg.Text = "Missing Fields";

lbl\_msg.Visible = true;

}

else

{

cmd.CommandText = "select max(SlNo) from birthreg";

dr = cmd.ExecuteReader();

if (dr.Read())

{

xx = dr.GetInt32(0);

xx = xx + 1;

}

dr.Close();

cmd.CommandText = "insert into birthreg values(" + xx + ",'" + dob + "','" + drop\_sex.SelectedValue + "','" + txtnameofchild.Text + "','" + txtnameoffather.Text + "','" + txtnameofmother.Text + "','" + txtplaceofbirth.Text + "','" + txtinformantsname.Text + "','" + txtaddressofinformant.Text + "','" + tdate + "','" + txtnameofvillage + "','" + txtnameofdistrict.Text + "','" + txtnameofstate.Text + "','" + txtreligion.Text + "','" + txtofccupation.Text + "','" + txtmoccupation.Text + "','" + txtageatbirthtime.Text + "','" + txtmage.Text + "','" + txtmethod.Text + "','" + txtweight.Text + "','" + txtduration.Text + "')";

x=cmd.ExecuteNonQuery();

if (x > 0)

{

lbl\_msg.Text = "Registration Successful";

lbl\_msg.Visible = true;

}

}

}

protected void btnreset\_Click(object sender, EventArgs e)

{

txtaddressofinformant.Text = "";

txtageatbirthtime.Text = "";

txtduration.Text = "";

txtinformantsname.Text = "";

txtmage.Text = "";

txtmethod.Text = "";

txtmoccupation.Text = "";

txtnameofchild.Text = "";

txtnameofdistrict.Text = "";

txtnameoffather.Text = "";

txtnameofmother.Text = "";

txtnameofstate.Text = "";

txtnameofvillage.Text = "";

txtofccupation.Text = "";

txtplaceofbirth.Text = "";

txtreligion.Text = "";

txtweight.Text = "";

}

}

**Death.aspx.cs**

using System;

using System.Data;

using System.Configuration;

using System.Collections;

using System.Web;

using System.Web.Security;

using System.Web.UI;

using System.Web.UI.WebControls;

using System.Web.UI.WebControls.WebParts;

using System.Web.UI.HtmlControls;

using System.Web.Configuration;

using System.Data.SqlClient;

public partial class Default4 : System.Web.UI.Page

{

SqlConnection con = new SqlConnection(WebConfigurationManager.ConnectionStrings["civilRegDB"].ConnectionString);

SqlCommand cmd = new SqlCommand();

SqlDataReader dr;

int x;

protected void Page\_Load(object sender, EventArgs e)

{

con.Open();

cmd.Connection = con;

}

protected void btnsubmit\_Click(object sender, EventArgs e)

{

string dob = Request.Form.Get("dob");

String sex;

String status;

if(rtbnyes.Checked==true)

{

status="Yes";

}

else

{

status="No";

}

if(rtbnmale.Checked==true)

{

sex="Male";

}

else

{

sex="Female";

}

if(txtage.Text=="" || txtattention.Text=="" || txtdeathplace.Text=="" || txtdeceased.Text=="" ||

txtdesease.Text=="" || txtdistrict.Text=="" || txtinformantadd.Text=="" || txtinformantname.Text=="" ||

txtoccupation.Text=="" || txtrelativename.Text=="" || txtreligion.Text=="" || txtstate.Text=="" ||

txttown.Text=="")

{

lblmsg.Text = "Missing Fields! Please fill all the mandatory fields";

lblmsg.Visible = true;

}

else

{

cmd.CommandText = "select max(SlNo) from death";

dr = cmd.ExecuteReader();

if (dr.Read())

{

x = dr.GetInt32(0);

x = x + 1;

}

dr.Close();

cmd.CommandText = "INSERT INTO death VALUES(" + x + ",'" + dob + "','" + txtdeceased.Text +

"','" + txtrelativename.Text + "','" + sex + "','" + txtage.Text + "','" + txtdeathplace.Text +

"','" + txtinformantname.Text + "','" + txtinformantadd.Text + "','" + txttown.Text +

"','" + txtdistrict.Text + "','" + txtstate.Text +

"','" + txtreligion.Text + "','" + txtoccupation.Text + "','" + status + "','" + txtdesease.Text + "')";

x = cmd.ExecuteNonQuery();

if (x > 0)

{

lblmsg.Text = "Death Registration Completed Successfully";

lblmsg.Visible = true;

}

else

{

lblmsg.Text = "Death Registration Failed! Pleae try again";

lblmsg.Visible = true;

}

}

}

protected void btnreset\_Click(object sender, EventArgs e)

{

lblmsg.Visible = false;

txtage.Text = "";

txtattention.Text = "";

txtdeathplace.Text = "";

txtdeceased.Text = "";

txtdesease.Text = "";

txtdistrict.Text = "";

txtinformantadd.Text = "";

txtinformantname.Text = "";

txtoccupation.Text = "";

txtrelativename.Text = "";

txtreligion.Text = "";

txtstate.Text = "";

txttown.Text = "";

rtbnmale.Checked = true;

rbtnfemale.Checked = false;

rtbnyes.Checked = true;

rtbnno.Checked = false;

}

}

**Feedback.aspx.cs**

using System;

using System.Data;

using System.Configuration;

using System.Collections;

using System.Web;

using System.Web.Security;

using System.Web.UI;

using System.Web.UI.WebControls;

using System.Web.UI.WebControls.WebParts;

using System.Web.UI.HtmlControls;

using System.Web.Configuration;

using System.Data.SqlClient;

public partial class Default3 : System.Web.UI.Page

{

SqlConnection con = new SqlConnection(WebConfigurationManager.ConnectionStrings["civilRegDB"].ConnectionString);

SqlCommand cmd = new SqlCommand();

protected void Page\_Load(object sender, EventArgs e)

{

con.Open();

cmd.Connection = con;

}

protected void btnSubmit\_Click(object sender, EventArgs e)

{

int x;

if(txtadr.Text=="" || txtarea.Text=="" || txtcity.Text=="" || txtcomp.Text=="" ||

txtcon.Text=="" || txtdes.Text=="" || txtdes.Text=="" || txtmail.Text=="" ||

txtnafed.Text=="" || txtpin.Text=="" || txtquery.Text=="" || txtstate.Text=="")

{

lblmsg.Text="Missing Fields! Please fill all the mandatory fields";

lblmsg.Visible=true;

}

else

{

cmd.CommandText = "INSERT INTO Feedback VALUES('" + txtnafed.Text + "','" + txtdes.Text + "','" + txtcomp.Text + "','" + txtarea.Text + "','" + txtadr.Text + "','" + txtcity.Text + "','" + txtstate.Text + "','" + txtcon.Text + "','" + txtpin.Text + "','" + txtmail.Text + "','" + txttel.Text + "','" + txtfax.Text + "','" + txtquery.Text + "')";

x = cmd.ExecuteNonQuery();

if (x > 0)

{

lblmsg.Text="Feedback posted Successfully!";

lblmsg.Visible=true;

}

else

{

lblmsg.Text="Failed to post Feedback! Please try again";

lblmsg.Visible=true;

}

}

}

protected void btnClear\_Click(object sender, EventArgs e)

{

lblmsg.Visible = false;

txtadr.Text = "";

txtarea.Text = "";

txtcity.Text = "";

txtcomp.Text = "";

txtcon.Text = "";

txtdes.Text = "";

txtfax.Text = "";

txtmail.Text = "";

txtnafed.Text = "";

txtpin.Text="";

txtquery.Text="";

txtstate.Text="";

txttel.Text="";

txtnafed.Focus();

}

}

**Divorce.aspx.cs**

using System;

using System.Data;

using System.Configuration;

using System.Collections;

using System.Web;

using System.Web.Security;

using System.Web.UI;

using System.Web.UI.WebControls;

using System.Web.UI.WebControls.WebParts;

using System.Web.UI.HtmlControls;

using System.Web.Configuration;

using System.Data.SqlClient;

public partial class \_Default : System.Web.UI.Page

{

SqlConnection con = new SqlConnection(WebConfigurationManager.ConnectionStrings["civilRegDB"].ConnectionString);

SqlCommand cmd = new SqlCommand();

SqlDataReader dr;

int xx;

int x;

string who;

string w;

protected void Page\_Load(object sender, EventArgs e)

{

con.Open();

cmd.Connection = con;

}

protected void txtphone\_TextChanged(object sender, EventArgs e)

{

}

protected void btnsub\_Click(object sender, EventArgs e)

{

string dob=Request.Form.Get("dob");

string dob1=Request.Form.Get("dob1");;

string dob2=Request.Form.Get("dob2");;

if (rdbtnhus.Checked == true)

{

who = "Husband";

}

else if (rdbtnwife.Checked == true)

{

who = "Wife";

}

else

{

who = "Both";

}

if(rdbtn18yes.Checked==true)

{

w="Yes";

}

else

{

w="No";

}

if (txtchlidname.Text == "" || txtmarplace.Text == "" || txtchlidname.Text == "" || txttime.Text == "" ||

txtfin.Text == "" || txthealth.Text == "" || txtedu.Text == "")

{

lblmsg.Text = "Missing Fields! Please fill all mandatory fields";

lblmsg.Visible = true;

}

else {

cmd.CommandText = "select max(SlNo) from divorce";

dr = cmd.ExecuteReader();

if (dr.Read())

{

x = dr.GetInt32(0);

x = x + 1;

}

dr.Close();

cmd.CommandText = "insert into divorcecommon values(" + xx + ",'" + who + "','" + dob +

"','" + txtmarplace.Text + "','" + dpdnchild.SelectedValue + "','" + w +

"','" + DropDownList1.SelectedValue + "','" + txtchlidname.Text +

"','" + txttime.Text + "','" + txtfin.Text + "','" + txthealth.Text +

"','" + txtedu.Text + "')";

x = cmd.ExecuteNonQuery();

cmd.CommandText = "insert into divorcehusband values(" + xx + ",'" + txt\_husfamily.Text +

"','" + txt\_husname.Text + "','" + dob1 + "','" + txt\_country.Text +

"','" + txt\_occupation.Text + "','" + txt\_addr.Text +

"','" + dpdnservice.SelectedValue + "','" + txtlawyername.Text +

"','" + txtfirm.Text + "','" + txtcode.Text + "','" + txtphone.Text +

"','" + txtfax.Text + "','" + txtemail.Text + "')";

cmd.ExecuteNonQuery();

cmd.CommandText = "insert into divorcewife values(" + xx + ",'" + txt\_wifefamily.Text +

"','" + txt\_wifename + "','" + dob2 + "','" + txt\_countrywife.Text +

"','" + txt\_occupationwife.Text + "','" + txt\_addrwife + "','" + dpdnwifeser.SelectedValue +

"','" + txtwifeslawyer + "','" + txtfirmwife + "','" + txtcodewife +

"','" + txtphonewife.Text + "','" + txtfaxwife.Text + "','" + txtemailwife.Text + "')";

if (x > 0)

{

lblmsg.Text = "Registration Completed";

lblmsg.Visible = true;

}

else

{

lblmsg.Text = "Registration Failed";

lblmsg.Visible = true;

}

}

}

}

**Electricity.aspx.cs**

using System;

using System.Data;

using System.Configuration;

using System.Collections;

using System.Web;

using System.Web.Security;

using System.Web.UI;

using System.Web.UI.WebControls;

using System.Web.UI.WebControls.WebParts;

using System.Web.UI.HtmlControls;

using System.Web.Configuration;

using System.Data.SqlClient;

public partial class Default3 : System.Web.UI.Page

{

SqlConnection con = new SqlConnection(WebConfigurationManager.ConnectionStrings["civilRegDB"].ConnectionString);

SqlCommand cmd = new SqlCommand();

SqlDataReader dr;

int total;

protected void Page\_Load(object sender, EventArgs e)

{

con.Open();

cmd.Connection = con;

}

protected void btnsub\_Click(object sender, EventArgs e)

{

if (txtelnam.Text == "" || txtkno.Text == "" || txtloc.Text == "" || txtlocal.Text == "" || txtoccup.Text == "" || txtperad.Text == "" || txttot1.Text == "" || txttot2.Text == "" || txttot3.Text == "" || txttot4.Text == "" || txttot5.Text == "" || txttot6.Text == "" || txttotal.Text == "" || txtwat1.Text == "" || txtwat2.Text == "" || txtwat3.Text == "" || txtwat4.Text == "" || txtwat5.Text == "" || txtwat6.Text == "" || txtcov.Text == "")

{

lbl\_msg.Text = "Missing Fields! Please fill up all";

lbl\_msg.Visible = true;

}

else

{

cmd.CommandText = "insert into electricity values('" + txtelnam.Text + "','" + txtoccup.Text + "','" + txtperad.Text + "','" + txtloc.Text + "','" + radio\_purpose.SelectedItem.Value + "','" + txtcov.Text + "','" + txtlocal.Text + "','" + txtkno.Text + "')";

cmd.ExecuteNonQuery();

cmd.CommandText = "insert into electricity1 values('" + txtelnam.Text + "','" + txtwat1.Text + "','" + txttot1.Text + "','" + txtwat2.Text + "','" + txttot2.Text + "','" + txtwat3.Text + "','" + txttot3.Text + "','" + txtwat4.Text + "','" + txttot4.Text + "','" + txtwat5.Text + "','" + txttot5.Text + "','" + txtwat6.Text + "','" + txttot6.Text + "','" + txttotal.Text + "')";

cmd.ExecuteNonQuery();

lbl\_msg.Text="Successfully Registered";

lbl\_msg.Visible = true;

}

}

protected void bt\_reset\_Click(object sender, EventArgs e)

{

txtcov.Text = "";

txtelnam.Text = "";

txtkno.Text = "";

txtloc.Text = "";

txtlocal.Text = "";

txtoccup.Text = "";

txtperad.Text = "";

txttot1.Text = "";

txttot2.Text = "";

txttot3.Text = "";

txttot4.Text = "";

txttot5.Text = "";

txttot6.Text = "";

txttotal.Text = "";

txtwat1.Text = "";

txtwat2.Text = "";

txtwat3.Text = "";

txtwat4.Text = "";

txtwat5.Text = "";

txtwat6.Text = "";

}

protected void btnCalc\_Click(object sender, EventArgs e)

{

int y=0;

if (txttot1.Text == "" || txttot2.Text == "" || txttot3.Text == "" || txttot4.Text == "" || txttot5.Text == "" || txttot6.Text == "")

{

txttot1.Text = y.ToString();

txttot2.Text = y.ToString();

txttot3.Text = y.ToString();

txttot4.Text = y.ToString();

txttot5.Text = y.ToString();

txttot6.Text = y.ToString();

txttotal.Text = total.ToString();

}

else

{

total = Int32.Parse(txttot1.Text) + Int32.Parse(txttot2.Text) + Int32.Parse(txttot3.Text) + Int32.Parse(txttot4.Text) + Int32.Parse(txttot5.Text) + Int32.Parse(txttot6.Text);

txttotal.Text = total.ToString();

rdbtncert.Enabled = true;

}

}

protected void rdbtncert\_CheckedChanged(object sender, EventArgs e)

{

if (rdbtncert.Checked == true)

{

btnsub.Enabled = true;

}

}

}

**Feedback1.aspx.cs**

using System;

using System.Data;

using System.Configuration;

using System.Collections;

using System.Web;

using System.Web.Security;

using System.Web.UI;

using System.Web.UI.WebControls;

using System.Web.UI.WebControls.WebParts;

using System.Web.UI.HtmlControls;

using System.Web.Configuration;

using System.Data.SqlClient;

public partial class Default3 : System.Web.UI.Page

{

SqlConnection con = new SqlConnection(WebConfigurationManager.ConnectionStrings["civilRegDB"].ConnectionString);

SqlCommand cmd = new SqlCommand();

protected void Page\_Load(object sender, EventArgs e)

{

con.Open();

cmd.Connection = con;

}

protected void btnSubmit\_Click(object sender, EventArgs e)

{

int x;

if(txtadr.Text=="" || txtarea.Text=="" || txtcity.Text=="" || txtcomp.Text=="" ||

txtcon.Text=="" || txtdes.Text=="" || txtdes.Text=="" || txtmail.Text=="" ||

txtnafed.Text=="" || txtpin.Text=="" || txtquery.Text=="" || txtstate.Text=="")

{

lblmsg.Text="Missing Fields! Please fill all the mandatory fields";

lblmsg.Visible=true;

}

else

{

cmd.CommandText = "INSERT INTO Feedback VALUES('" + txtnafed.Text + "','" + txtdes.Text + "','" + txtcomp.Text + "','" + txtarea.Text + "','" + txtadr.Text + "','" + txtcity.Text + "','" + txtstate.Text + "','" + txtcon.Text + "','" + txtpin.Text + "','" + txtmail.Text + "','" + txttel.Text + "','" + txtfax.Text + "','" + txtquery.Text + "')";

x = cmd.ExecuteNonQuery();

if (x > 0)

{

lblmsg.Text="Feedback posted Successfully!";

lblmsg.Visible=true;

}

else

{

lblmsg.Text="Failed to post Feedback! Please try again";

lblmsg.Visible=true;

}

}

}

protected void btnClear\_Click(object sender, EventArgs e)

{

lblmsg.Visible = false;

txtadr.Text = "";

txtarea.Text = "";

txtcity.Text = "";

txtcomp.Text = "";

txtcon.Text = "";

txtdes.Text = "";

txtfax.Text = "";

txtmail.Text = "";

txtnafed.Text = "";

txtpin.Text="";

txtquery.Text="";

txtstate.Text="";

txttel.Text="";

txtnafed.Focus();

}

}

**Forumadmin.aspx.cs**

using System;

using System.Data;

using System.Configuration;

using System.Collections;

using System.Web;

using System.Web.Security;

using System.Web.UI;

using System.Web.UI.WebControls;

using System.Web.UI.WebControls.WebParts;

using System.Web.UI.HtmlControls;

using System.Data.SqlClient;

public partial class forum : System.Web.UI.Page

{

SqlDataAdapter da;

SqlConnection con;

DataSet ds = new DataSet();

protected void Page\_Load(object sender, EventArgs e)

{

con = new SqlConnection(ConfigurationManager.ConnectionStrings["civilRegDB"].ConnectionString);

con.Open();

}

}

**Login.aspx.cs**

using System;

using System.Data;

using System.Configuration;

using System.Collections;

using System.Web;

using System.Web.Security;

using System.Web.UI;

using System.Web.UI.WebControls;

using System.Web.UI.WebControls.WebParts;

using System.Web.UI.HtmlControls;

using System.Web.Configuration;

using System.Data.SqlClient;

public partial class Default2 : System.Web.UI.Page

{

SqlConnection con = new SqlConnection(WebConfigurationManager.ConnectionStrings["civilRegDB"].ConnectionString);

SqlCommand cmd = new SqlCommand();

SqlDataReader dr;

protected void Page\_Load(object sender, EventArgs e)

{

con.Open();

cmd.Connection = con;

}

protected void bt\_submit\_Click(object sender, EventArgs e)

{

if (txtpassword.Text == "" || txtuser.Text == "")

{

lbl\_msg.Text = "Missing Fields!";

lbl\_msg.Visible = true;

}

else

{

cmd.CommandText = "select \* from login where Username='" + txtuser.Text + "' and Password='" + txtpassword.Text + "'";

dr = cmd.ExecuteReader();

if (dr.Read())

{

Session["user"] = txtuser.Text;

Response.Redirect("index.aspx");

}

else

{

lbl\_msg.Text = "Login Failed! Invalid User name or Password";

lbl\_msg.Visible = true;

}

}

}

protected void bt\_reset\_Click(object sender, EventArgs e)

{

txtpassword.Text = "";

txtuser.Text = "";

}

}

**Mailer.aspx.cs**

using System;

using System.Data;

using System.Configuration;

using System.Collections;

using System.Web;

using System.Web.Security;

using System.Web.UI;

using System.Web.UI.WebControls;

using System.Web.UI.WebControls.WebParts;

using System.Web.UI.HtmlControls;

using System.Net.Mail;

using System.Data.SqlClient;

public partial class mailer : System.Web.UI.Page

{

MailMessage mailmsg;

SqlConnection con;

SqlCommand cmd;

protected void Page\_Load(object sender, EventArgs e)

{

if (Session["admin"] == null)

{

Response.Redirect("index.aspx");

}

//if (Request.QueryString["id"] != null)

//{

// txtToAddress.Text = Request.QueryString["id"];

//}

if(!IsPostBack)

{

String user = Session["admin"].ToString();

con = new SqlConnection(ConfigurationManager.ConnectionStrings["civilRegDB"].ConnectionString);

con.Open();

cmd = new SqlCommand("select email from registration where username='" + user + "'",con );

SqlDataReader dr = cmd.ExecuteReader();

if(dr.Read())

{

txtFromAddress.Text = dr.GetString(0);

}

dr.Close();

}

}

protected void btnSendmail\_Click(object sender, ImageClickEventArgs e)

{

try

{

String to = txtToAddress.Text;

String from = txtFromAddress.Text;

String cc = txtCCAddress.Text;

String bcc = txtBCC.Text;

String subject = txtSubject.Text;

String body = txtMessage.Text;

mailmsg = new MailMessage();

mailmsg.From =new MailAddress( from );

mailmsg.To.Add ( new MailAddress(to));

if ((cc != null) && (cc != string.Empty))

{

mailmsg.CC.Add(new MailAddress(cc));

}

if ((bcc != null) && (bcc != string.Empty))

{

mailmsg.Bcc.Add(new MailAddress(bcc));

}

mailmsg.Subject = subject;

mailmsg.Body = body;

if (fileAttachment.HasFile)

{

mailmsg.Attachments.Add(new Attachment(fileAttachment.PostedFile.FileName));

}

// Set the format of the mail message body as HTML

mailmsg.IsBodyHtml = true;

// Set the priority of the mail message to normal

mailmsg.Priority = MailPriority.Normal;

SmtpClient smtpclient = new SmtpClient();

smtpclient.Host = ConfigurationManager.AppSettings["hostname"];

smtpclient.Send(mailmsg);

}

catch (Exception ex)

{

lblmsg.Text = "Your mail has been sent..";

}

}

}

**Marriage.aspx.cs**

using System;

using System.Data;

using System.Configuration;

using System.Collections;

using System.Web;

using System.Web.Security;

using System.Web.UI;

using System.Web.UI.WebControls;

using System.Web.UI.WebControls.WebParts;

using System.Web.UI.HtmlControls;

using System.Web.Configuration;

using System.Data.SqlClient;

public partial class \_Default : System.Web.UI.Page

{

SqlConnection con = new SqlConnection(WebConfigurationManager.ConnectionStrings["civilRegDB"].ConnectionString);

SqlCommand cmd = new SqlCommand();

int x;

protected void Page\_Load(object sender, EventArgs e)

{

con.Open();

cmd.Connection = con;

}

protected void btnapp\_Click(object sender, EventArgs e)

{

String bdob=Request.Form.Get("txtBDOB");

String dob1=Request.Form.Get("txtBDOB1");

String dob2=Request.Form.Get("txtBDOB2");

string test=Request.Form.Get("txtMDate");

if (txtfath.Text == "" || txtfathwife.Text == "" || txtnam.Text == "" ||

txtnamewife.Text == "" || txtper.Text == "" || txtperwife.Text == "" ||

txtpre.Text == "" || txtprewife.Text == "" || txtrel.Text == "" ||

txtsol.Text == "" || txtvote.Text == "" || txtvotewife.Text == "")

{

lblmsg.Text = "Missing Fieldds! Please fill all mandatory fields";

lblmsg.Visible = true;

}

else

{

cmd.CommandText="insert into marriage3 values('" + bdob + "','" + txtnam.Text +

"','" + txtnamewife.Text + "','" + txtfath.Text + "','" + txtfathwife.Text +

"','" + dob1 + "','" + dob2 + "','" + txtpre.Text + "','" + txtprewife.Text +

"','" + txtper.Text + "','" + txtperwife.Text + "','" + txtvote.Text +

"','" + txtvotewife.Text + "','" + dpdnmar.SelectedValue + "','" + dpdnmarwife.SelectedValue + "','" + test + "','" + txtsol.Text + "','" + txtrel.Text + "')";

x = cmd.ExecuteNonQuery();

if (x > 0)

{

lblmsg.Text = "Record Added";

lblmsg.Visible = true;

}

else

{

lblmsg.Text = "Record failed to Add";

lblmsg.Visible = true;

}

}

}

}

**Namechange.aspx.cs**

using System;

using System.Data;

using System.Configuration;

using System.Collections;

using System.Web;

using System.Web.Security;

using System.Web.UI;

using System.Web.UI.WebControls;

using System.Web.UI.WebControls.WebParts;

using System.Web.UI.HtmlControls;

using System.Data.SqlClient;

public partial class postreplay : System.Web.UI.Page

{

String postid;

SqlConnection con;

SqlCommand cmd;

protected void Page\_Load(object sender, EventArgs e)

{

con = new SqlConnection(ConfigurationManager.ConnectionStrings["civilRegDB"].ConnectionString);

con.Open();

cmd = new SqlCommand();

cmd.Connection = con;

if (Request.QueryString["id"]!=null)

{

postid = Request.QueryString["id"];

}

}

protected void btnPostReply\_Click(object sender, ImageClickEventArgs e)

{

String name = txtName.Text;

String comment = txtComment.Text;

String email = txtEmail.Text;

cmd.CommandText = "insert into PostReply values(" + postid + ",'" + name + "','" + comment +

"','" + email + "','" + DateTime.Today + "')";

if (cmd.ExecuteNonQuery() > 0)

{

Response.Redirect("forum.aspx?newpost=" + postid);

}

else

{

lblmsg.Text = "Your comment submission failed";

}

}

}

**Pan.aspx.cs**

using System;

using System.Data;

using System.Configuration;

using System.Collections;

using System.Web;

using System.Web.Security;

using System.Web.UI;

using System.Web.UI.WebControls;

using System.Web.UI.WebControls.WebParts;

using System.Web.UI.HtmlControls;

using System.Data.SqlClient;

public partial class postreplay : System.Web.UI.Page

{

String postid;

SqlConnection con;

SqlCommand cmd;

protected void Page\_Load(object sender, EventArgs e)

{

con = new SqlConnection(ConfigurationManager.ConnectionStrings["civilRegDB"].ConnectionString);

con.Open();

cmd = new SqlCommand();

cmd.Connection = con;

if (Request.QueryString["id"]!=null)

{

postid = Request.QueryString["id"];

}

}

protected void btnPostReply\_Click(object sender, ImageClickEventArgs e)

{

String name = txtName.Text;

String comment = txtComment.Text;

String email = txtEmail.Text;

cmd.CommandText = "insert into PostReply values(" + postid + ",'" + name + "','" + comment +

"','" + email + "','" + DateTime.Today + "')";

if (cmd.ExecuteNonQuery() > 0)

{

Response.Redirect("forum.aspx?newpost=" + postid);

}

else

{

lblmsg.Text = "Your comment submission failed";

}

}

}

**Pasport.aspx.cs**

using System;

using System.Data;

using System.Configuration;

using System.Collections;

using System.Web;

using System.Web.Security;

using System.Web.UI;

using System.Web.UI.WebControls;

using System.Web.UI.WebControls.WebParts;

using System.Web.UI.HtmlControls;

using System.Data.SqlClient;

public partial class postreplay : System.Web.UI.Page

{

String postid;

SqlConnection con;

SqlCommand cmd;

protected void Page\_Load(object sender, EventArgs e)

{

con = new SqlConnection(ConfigurationManager.ConnectionStrings["civilRegDB"].ConnectionString);

con.Open();

cmd = new SqlCommand();

cmd.Connection = con;

if (Request.QueryString["id"]!=null)

{

postid = Request.QueryString["id"];

}

}

protected void btnPostReply\_Click(object sender, ImageClickEventArgs e)

{

String name = txtName.Text;

String comment = txtComment.Text;

String email = txtEmail.Text;

cmd.CommandText = "insert into PostReply values(" + postid + ",'" + name + "','" + comment +

"','" + email + "','" + DateTime.Today + "')";

if (cmd.ExecuteNonQuery() > 0)

{

Response.Redirect("forum.aspx?newpost=" + postid);

}

else

{

lblmsg.Text = "Your comment submission failed";

}

}

}

**Postthread.aspx.cs**

using System;

using System.Data;

using System.Configuration;

using System.Collections;

using System.Web;

using System.Web.Security;

using System.Web.UI;

using System.Web.UI.WebControls;

using System.Web.UI.WebControls.WebParts;

using System.Web.UI.HtmlControls;

using System.Data.SqlClient;

public partial class postthread : System.Web.UI.Page

{

String \_sql;

SqlConnection \_con = null;

SqlCommand \_cmd = null;

SqlDataReader \_dr;

protected void Page\_Load(object sender, EventArgs e)

{

\_con = new SqlConnection(ConfigurationManager.ConnectionStrings["civilRegDB"].ConnectionString);

\_con.Open();

\_cmd = new SqlCommand();

\_cmd.Connection = \_con;

}

protected void btnPostThread\_Click(object sender, ImageClickEventArgs e)

{

int threadid = 0;

try

{

\_cmd.CommandText = "select count(\*) from Forumthread";

\_dr = \_cmd.ExecuteReader();

if (\_dr.Read())

{

threadid=\_dr.GetInt32(0) + 1;

}

\_dr.Close();

String \_name = txtName.Text;

String \_email = txtEmail.Text;

String \_desc = txtDesc.Text;

\_sql = "Insert into Forumthread values(@threadid,@name,@email,@desc,@date)";

\_cmd.CommandText = \_sql;

\_cmd.Parameters.AddWithValue("@threadid", threadid );

\_cmd.Parameters.AddWithValue("@name", \_name);

\_cmd.Parameters.AddWithValue("@email", \_email);

\_cmd.Parameters.AddWithValue("@desc", \_desc);

\_cmd.Parameters.AddWithValue("@date", DateTime.Now);

\_cmd.ExecuteNonQuery();

lblmsg.Text="New thread posted (" + DateTime.Now + ") by " + \_name;

txtName.Text=null;

txtEmail.Text=null;

txtDesc.Text=null;

}

catch (Exception ex)

{

Response.Redirect("errorpage.aspx?id=db&desc=" + ex.Message);

}

}

}

**Postthreadadmin.aspx.cs**

using System;

using System.Data;

using System.Configuration;

using System.Collections;

using System.Web;

using System.Web.Security;

using System.Web.UI;

using System.Web.UI.WebControls;

using System.Web.UI.WebControls.WebParts;

using System.Web.UI.HtmlControls;

using System.Data.SqlClient;

public partial class postthread : System.Web.UI.Page

{

String \_sql;

SqlConnection \_con = null;

SqlCommand \_cmd = null;

SqlDataReader \_dr;

protected void Page\_Load(object sender, EventArgs e)

{

\_con = new SqlConnection(ConfigurationManager.ConnectionStrings["civilRegDB"].ConnectionString);

\_con.Open();

\_cmd = new SqlCommand();

\_cmd.Connection = \_con;

}

protected void btnPostThread\_Click(object sender, ImageClickEventArgs e)

{

int threadid = 0;

try

{

\_cmd.CommandText = "select count(\*) from Forumthread";

\_dr = \_cmd.ExecuteReader();

if (\_dr.Read())

{

threadid=\_dr.GetInt32(0) + 1;

}

\_dr.Close();

String \_name = txtName.Text;

String \_email = txtEmail.Text;

String \_desc = txtDesc.Text;

\_sql = "Insert into Forumthread values(@threadid,@name,@email,@desc,@date)";

\_cmd.CommandText = \_sql;

\_cmd.Parameters.AddWithValue("@threadid", threadid );

\_cmd.Parameters.AddWithValue("@name", \_name);

\_cmd.Parameters.AddWithValue("@email", \_email);

\_cmd.Parameters.AddWithValue("@desc", \_desc);

\_cmd.Parameters.AddWithValue("@date", DateTime.Now);

\_cmd.ExecuteNonQuery();

lblmsg.Text="New thread posted (" + DateTime.Now + ") by " + \_name;

txtName.Text=null;

txtEmail.Text=null;

txtDesc.Text=null;

}

catch (Exception ex)

{

Response.Redirect("errorpage.aspx?id=db&desc=" + ex.Message);

}

}

}

**Ration.aspx.cs**

using System;

using System.Data;

using System.Configuration;

using System.Collections;

using System.Web;

using System.Web.Security;

using System.Web.UI;

using System.Web.UI.WebControls;

using System.Web.UI.WebControls.WebParts;

using System.Web.UI.HtmlControls;

using System.Web.Configuration;

using System.Data.SqlClient;

public partial class \_Default : System.Web.UI.Page

{

SqlConnection con = new SqlConnection(WebConfigurationManager.ConnectionStrings["civilRegDB"].ConnectionString);

SqlCommand cmd = new SqlCommand();

SqlDataReader dr;

int xx;

int x;

protected void Page\_Load(object sender, EventArgs e)

{

con.Open();

cmd.Connection = con;

}

protected void RadioButton1\_CheckedChanged(object sender, EventArgs e)

{

if (RadioButton1.Checked == true)

{

btn\_submit.Enabled = true;

}

}

protected void btn\_submit\_Click(object sender, EventArgs e)

{

cmd.CommandText="select max(SNo) from rationcard";

dr=cmd.ExecuteReader();

if(dr.Read())

{

xx=dr.GetInt32(0);

xx=xx+1;

}

dr.Close();

if (txt\_nameappln.Text == "" || txt\_fathersname.Text == "" || txt\_prstaddr.Text == "" ||

txt\_pertaddr.Text == "" || txt\_occptn.Text == "" || txt\_nameEmpl.Text == "" ||

txt\_adult.Text == "" || txt\_minor.Text == "")

{

lblmsg.Text = "Missing Fields! Please fill all the mandatory Fields";

lblmsg.Visible = true;

}

else

{

cmd.CommandText = "insert into rationcard values(" + xx + ",'" + txt\_nameappln.Text +

"','" + txt\_fathersname.Text + "','" + txt\_prstaddr.Text + "','" + txt\_pertaddr.Text +

"','" + txt\_occptn.Text + "','" + txt\_nameEmpl.Text + "','" + txt\_adult.Text +

"','" + txt\_minor.Text + "','" + txt\_elecno.Text + "','" + txt\_trano.Text + "','" + txt\_lpgno.Text + "')";

x = cmd.ExecuteNonQuery();

cmd.CommandText = "insert into rationRelation values(" + xx + ",'" + txtfull1.Text + "','" + txtage1.Text + "','" + dp1.SelectedItem.Value + "','" + txtrel1.Text + "','" + txtfull2.Text + "','" + txtage2.Text + "','" + dp2.SelectedItem.Value + "','" + txtrel2.Text + "','" + txtfull3.Text + "','" + txtage3.Text + "','" + dp3.SelectedItem.Value + "','" + txtrel3.Text + "','" + txtfull4.Text + "','" + txtage4.Text + "','" + dp4.SelectedItem.Value + "','" + txtrel4.Text + "','" + txtfull5.Text + "','" + txtage5.Text + "','" + dp5.SelectedItem.Value + "','" + txtrel5.Text + "')";

x = cmd.ExecuteNonQuery();

if (x > 0)

{

lblmsg.Text = "Registered Successfully";

lblmsg.Visible = true;

}

else

{

lblmsg.Text = "Registeration Failed";

lblmsg.Visible = true;

}

}

}

protected void btnre\_Click(object sender, EventArgs e)

{

lblmsg.Visible = false;

txt\_nameappln.Text = "";

txt\_fathersname.Text = "";

txt\_prstaddr.Text = "";

txt\_pertaddr.Text = "";

txt\_occptn.Text = "";

txt\_nameEmpl.Text = "";

txt\_adult.Text = "";

txt\_minor.Text = "";

txt\_elecno.Text = "";

txt\_trano.Text = "";

txt\_lpgno.Text = "";

txtfull1.Text = "";

txtfull2.Text = "";

txtfull3.Text = "";

txtfull4.Text = "";

txtfull5.Text = "";

txtage1.Text = "";

txtage2.Text = "";

txtage3.Text = "";

txtage4.Text = "";

txtage5.Text = "";

txtrel1.Text = "";

txtrel2.Text = "";

txtrel3.Text="";

txtrel4.Text="";

txtrel5.Text="";

}

}

**Reply.aspx.cs**

using System;

using System.Data;

using System.Configuration;

using System.Collections;

using System.Web;

using System.Web.Security;

using System.Web.UI;

using System.Web.UI.WebControls;

using System.Web.UI.WebControls.WebParts;

using System.Web.UI.HtmlControls;

using System.Data.SqlClient;

public partial class postreplay : System.Web.UI.Page

{

String postid;

SqlConnection con;

SqlCommand cmd;

protected void Page\_Load(object sender, EventArgs e)

{

con = new SqlConnection(ConfigurationManager.ConnectionStrings["civilRegDB"].ConnectionString);

con.Open();

cmd = new SqlCommand();

cmd.Connection = con;

if (Request.QueryString["id"]!=null)

{

postid = Request.QueryString["id"];

}

}

protected void btnPostReply\_Click(object sender, ImageClickEventArgs e)

{

String name = txtName.Text;

String comment = txtComment.Text;

String email = txtEmail.Text;

cmd.CommandText = "insert into PostReply values(" + postid + ",'" + name + "','" + comment +

"','" + email + "','" + DateTime.Today + "')";

if (cmd.ExecuteNonQuery() > 0)

{

Response.Redirect("forum.aspx?newpost=" + postid);

}

else

{

lblmsg.Text = "Your comment submission failed";

}

}

}

**Replyadmin.aspx.cs**

using System;

using System.Data;

using System.Configuration;

using System.Collections;

using System.Web;

using System.Web.Security;

using System.Web.UI;

using System.Web.UI.WebControls;

using System.Web.UI.WebControls.WebParts;

using System.Web.UI.HtmlControls;

using System.Data.SqlClient;

public partial class postreplay : System.Web.UI.Page

{

String postid;

SqlConnection con;

SqlCommand cmd;

protected void Page\_Load(object sender, EventArgs e)

{

con = new SqlConnection(ConfigurationManager.ConnectionStrings["civilRegDB"].ConnectionString);

con.Open();

cmd = new SqlCommand();

cmd.Connection = con;

if (Request.QueryString["id"]!=null)

{

postid = Request.QueryString["id"];

}

}

protected void btnPostReply\_Click(object sender, ImageClickEventArgs e)

{

String name = txtName.Text;

String comment = txtComment.Text;

String email = txtEmail.Text;

cmd.CommandText = "insert into PostReply values(" + postid + ",'" + name + "','" + comment +

"','" + email + "','" + DateTime.Today + "')";

if (cmd.ExecuteNonQuery() > 0)

{

Response.Redirect("forumadmin.aspx?newpost=" + postid);

}

else

{

lblmsg.Text = "Your comment submission failed";

}

}

}

**Telapply.aspx.cs**

using System;

using System.Data;

using System.Configuration;

using System.Collections;

using System.Web;

using System.Web.Security;

using System.Web.UI;

using System.Web.UI.WebControls;

using System.Web.UI.WebControls.WebParts;

using System.Web.UI.HtmlControls;

using System.Web.Configuration;

using System.Data.SqlClient;

public partial class Default2 : System.Web.UI.Page

{

SqlConnection con = new SqlConnection(WebConfigurationManager.ConnectionStrings["civilRegDB"].ConnectionString);

SqlCommand cmd = new SqlCommand();

String fac;

protected void Page\_Load(object sender, EventArgs e)

{

con.Open();

cmd.Connection = con;

}

protected void btnsubmit\_Click(object sender, EventArgs e)

{

int x;

fac = "";

if (chkabbreviated.Checked == true)

{

fac += "Abbreviated Dialing" + ",";

}

if (chkcallforwarding.Checked == true)

{

fac += "Call Forwarding" + ",";

}

if (chkcli.Checked == true)

{

fac += "CLI" + ",";

}

if (chkconference.Checked == true)

{

fac += "Conferencing" + ",";

}

if (chkhotline.Checked == true)

{

fac += "Hotline" + ",";

}

if (chkisd.Checked == true)

{

fac += "ISD" + ",";

}

if (chkstd.Checked == true)

{

fac += "STD" + ",";

}

if (fac == "")

{

}

else

{

fac = fac.Substring(0, fac.Length - 1);

}

if (txtaddress.Text == "" || txtbilling.Text == "" || txtcomp.Text == "" || txtdetails.Text == "" ||

txtemail.Text == "" || txtgphf.Text == "" || txtinstr.Text == "" || txtjoint.Text == "" ||

txtname.Text == "" || txtnear.Text == "" || txtnominee.Text == "" || txtpan.Text == "" || txtpay.Text == "" ||

txtpin.Text == "" || txtrelation.Text == "" || txttele.Text == "")

{

lblmsg.Text = "Missing Fields! Please fill all the mandatory fields";

lblmsg.Visible = true;

}

else

{

cmd.CommandText = "insert into telapply values('" + txtcomp.Text +

"','" + txtname.Text + "','" + txtjoint.Text + "','" + txtnominee.Text +

"','" + txtrelation.Text + "','" + txtgphf.Text + "','" + txtpan.Text +

"','" + txttele.Text + "','" + txtnear.Text + "','" + txtaddress.Text +

"','" + txtpin.Text + "','" + txtbilling.Text + "','" + txtemail.Text +

"','" + fac + "','" + txtinstr.Text + "','" + txtpay.Text + "','" + txtdetails.Text + "')";

x = cmd.ExecuteNonQuery();

if (x > 0)

{

lblmsg.Text = "Registration Completed successfully";

lblmsg.Visible = true;

}

else

{

lblmsg.Text = "Registration Failed";

lblmsg.Visible = true;

}

}

}

protected void btnreset\_Click(object sender, EventArgs e)

{

lblmsg.Visible = false;

txtaddress.Text = "";

txtbilling.Text = "";

txtcomp.Text = "";

txtdetails.Text = "";

txtemail.Text = "";

txtgphf.Text = "";

txtinstr.Text = "";

txtjoint.Text = "";

txtname.Text = "";

txtnear.Text = "";

txtnominee.Text = "";

txtpan.Text = "";

txtpay.Text = "";

txtpin.Text = "";

txtrelation.Text = "";

txttele.Text = "";

}

}

**Water1.aspx.cs**

using System;

using System.Data;

using System.Configuration;

using System.Collections;

using System.Web;

using System.Web.Security;

using System.Web.UI;

using System.Web.UI.WebControls;

using System.Web.UI.WebControls.WebParts;

using System.Web.UI.HtmlControls;

using System.Web.Configuration;

using System.Data.SqlClient;

public partial class \_Default : System.Web.UI.Page

{

SqlConnection con = new SqlConnection(WebConfigurationManager.ConnectionStrings["civilRegDB"].ConnectionString);

SqlCommand cmd = new SqlCommand();

protected void Page\_Load(object sender, EventArgs e)

{

con.Open();

cmd.Connection = con;

}

protected void btncont\_Click(object sender, EventArgs e)

{

int x;

if (txtapp.Text == "" || txthus.Text == "" || txtmemo.Text == "" || txtplot.Text == "" || txtpost.Text == "")

{

lblmsg.Text = "Missing Fields! Please fill all the mandatory fields";

lblmsg.Visible = true;

}

else

{

cmd.CommandText = "insert into water1 values('" + txtapp.Text + "','" + txthus.Text +

"','" + txtpost.Text + "','" + txtplot.Text + "','" + dpdlown.SelectedValue +

"','" + dpdlres.SelectedValue + "','" + dpdlinter.Text + "','" + dpdlroad.SelectedValue +

"','" + txtmemo.Text + "')";

x = cmd.ExecuteNonQuery();

if (x > 0)

{

lblmsg.Text = "You have regitered successfully";

lblmsg.Visible = true;

}

else

{

lblmsg.Text = "Registration Failed";

lblmsg.Visible = true;

}

}

}

protected void btnreset\_Click(object sender, EventArgs e)

{

txtapp.Text = "";

txthus.Text = "";

txtmemo.Text = "";

txtplot.Text = "";

txtpost.Text = "";

txtapp.Focus();

lblmsg.Visible = false;

}

}

**SCREEN SHOTS**

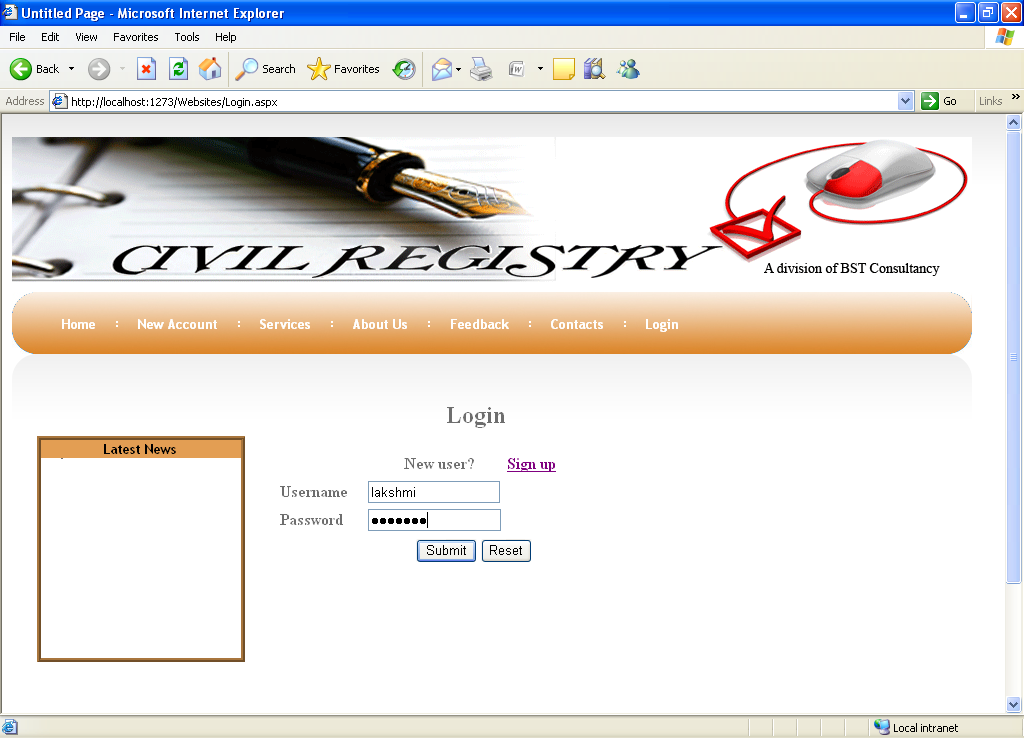
**Home**

****

**Services**

****

**Login**

****

**Birth**

****

**Death**

****

**Divorce**

****

**Electricity application**

****

**Mail**

****

**Marriage Registration**

****

**Pan Card Registration**

****

**Passport Application**

****

**Ration Card Application**

****

**Telephone Application**

****

**Voters ID Card**

****

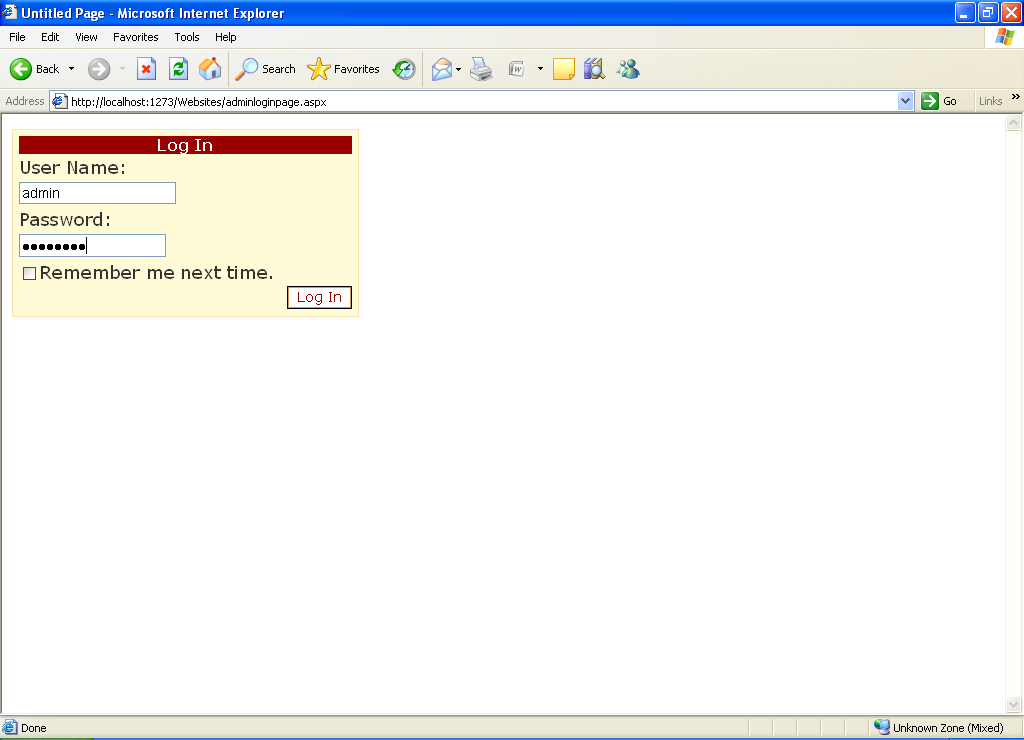
**Water Tap Connection**

****

**Feedback**

****

**Admin Login**

****

**FUTURE ENHANCEMENT**

Now the users can only do registrations through online. All the remaining procedures are done manually. In future we can do full process through online. Civil Registry team can apply for the tie up or authorization from all the Government offices like Passport Office, Register Office, RTO, and Cooperation Office etc.

**CONCLUSION**

Now a day’s manual process for the citizens to apply for their government records like passport, driving license, voter’s id, pan card etc… has become a huge task. The main object of the website is to reduce the effort by the candidate and save his time and avoid unwanted rushes at the government offices and assure a smooth working schedule at government offices. The main features of this site includes flexibility, reduce manual work in an efficient manner, a quick, convenient, reliable and effective way to apply for their government records. The project could very well be enhanced further as per the requirements.

**BIBILIOGRAPHY**

* **Windows programming Charles Petzoid**
* **ASP. NET Unleashed Stephen Walther**
* **Visual Basic .NET Black Book Steven Holzer**
* **Professional ASP. NET Web services Alex Homer, Dave Sussman**
* **SQL Server Essential Reference Sharon Dooley**