**CHAPTER 1**

**INTRODUCTION**

The project “Curfew E-PASS MANAGEMENT SYSTEM” aims to create a system that can be used to efficiently handle people’s passes during Covid-19 during curfew. Data and instructions can be stored in an electronic form in a computer’s internal memory and retrieved at any time. It makes the job easier to accomplish and reduces the number of paperwork needed. Lockdown is a method of preventing infection spread that comprises avoiding leaving the house unless absolutely necessary. People may need to go from one town to another in extraordinary conditions, and this e-pass producing system will aid in their contactless mobility. The paper gate pass can now be replaced with an electronic version. This project is about the “Curfew E-PASS MANAGEMENT SYSTEM,” which is a web-based technology that keeps track of passes given by the government and administration during curfews and lockdowns. Valid passes must be obtained as soon as possible to ensure that key services continue to operate throughout the COVID-19 pandemic. This fully automated system for creating and managing e-passes during curfew and lockdown processes data at a rapid speed and in a systematic manner. Curfew Pass Management system could be a web-based technology which will manage the records of pass that is issue by body and conjointly facilitate to supply on-line curfew e-pass to those who got to travel mandatory. Curfew Pass Management System is associate degree automatic system that delivers processing in terribly high speed in systematic manner. The code powered by ASP assures clear and economical services to the agency. This easy-to-operate system helps to access and modify user details, provides economical printing facility. The code is meant to supply Reliable and error free data. 1.1 Purpose and Scope Lockdown 4.0 looms over Bharat with little or no progress on the reduced range of cases. A fast spike in cases within the previous few weeks has brought Bharat to its unsettling and depressing range of quite one hundred thousand infected cases.

The Centre has delegated some lockdown selections and announcements to State governments supported however their State is faring. Once all of your credentials are stuffed and your purpose has backend and SQL for database are used. This is the system which produce and keep record of the pass which is issued by regulatory.

The Management of Curfew e-Pass system additionally gives a notification board or something similar on which when and where a curfew will begin and end will be told. This will help people to know about the curfew beforehand and grab curfew passes before the curfew starts. This system has just two modules the ADMIN module and USER module. The administrator has all the rights and will control all the activities in this system. Users can only view the pass on the site utilizing unique ID produced during the creation procedure and also get appointment for issuing pass from the admin. Warnings or notifications of the curfew will be provided beforehand or when announced.

Curfew Pass Management system could be a web-based technology which will manage the records of pass that is issue by body and conjointly facilitate to supply on-line curfew e-pass to those who got to travel mandatory. Curfew Pass Management System is associate degree automatic system that delivers processing in terribly high speed in systematic manner. The code powered by PHP assures clear and economical services to the agency. This easy-to-operate system helps to access and modify user details, provides economical printing facility. The code is meant to supply Reliable and error free data. Computers have become a way of life for today’s high society. Many aspects of modern and life that we have come to accept as common place here would not be possible if there were no computers. Today computers are used extensively in many areas of business, industry, science, education etc.The major advantage of computer techno is its speed that makes it able to give some useful information very quickly. This speed also opens new approaches to problem solving and data processing. Another feature is its accuracy. Though the computers do only what is instructed at every instant, these instructions are taken into account and accurate information’s are produced. Computer can hold data and instruction in an electronic representation in internal memory and this data can be retrieved at any time.The project entitled “E-Pass Generation and Management System During Curfew” is a software package, which can be used in curfew for managing the pass of people’ computer efficiently A Curfew indicates a period during which certain guidelines apply.

Ordinarily it alludes to when people are required to come back to and remain in houses or homes. Curfews are an ordinary territory of control utilized in military law, in any case curfews can in like way be executed for open prospering if there should rise an event of a crisis or emergency.

A Curfew Pass is a record given by an engaged position, for example, police or military for open authorities or common people to make a trip inside or to and from a zone under a curfew by the said power. The Management of Curfew e-Pass system is an electronic innovation that will manage the records of pass which issue by definitive. This system is a modified structure that conveys information preparing in a fast and conscious manner. been clearly expressed, your application are sent to the authorities for examination. The assigned department can then notify you via associate degree SMS if your application has been approved. The SMS will offer a transfer link to induce the e-pass. This e-Pass has to be carried with you once you area unit movement as you'll be stopped by personnel to see your details and if you're permissible to travel throughout the imprisonment. take care to carry it with you in any respect times. The departments area unit fastidiously scrutinising applications and solely approving those who area unit real and emergency cases.

**EXISTING SYSTEM:**

Lockdown 4.0 looms over Bharat with little or no progress on the reduced range of cases. A fast spike in cases within the previous few weeks has brought Bharat to its unsettling and depressing range of quite one hundred thousand infected cases. The Centre has delegated some lockdown selections and announcements to State governments supported however their State is faring. Once all of your credentials are stuffed and your purpose has been clearly expressed, your application are sent to the authorities for examination. The assigned department can then notify you via associate degree SMS if your application has been approved. The SMS will offer a transfer link to induce the e-pass. This e-Pass has to be carried with you once you area unit movement as you'll be stopped by personnel to see your details and if you're permissible to travel throughout the imprisonment. take care to carry it with you in any respect times. The departments area unit fastidiously scrutinising applications and solely approving those who area unit real and emergency cases. The E-pass system provides choice to the Admin to pick out a class (Which is necessities and as directed by the state government) to assign a specific a selected style of e-pass turning into there to particular class. Admin has rights to add/remove a specific class. Admin portal has associate choice to add details of Admin himself and additionally provides all the access management to the admin.

The admin will generate and issue a pass to a specific person. this is often associate access based mostly internet application developed mistreatment asp, SQL, JQuery, mythical being and JavaScript, when the victorious generation of the e-pass the pass holders will verify their travel simply coming into the pass variety within the given input field. The pass will be verified globally from anyplace with the assistance of the pass Id. This helps the user and also the authorities to validate the e-pass with convience. E-PASS GENERATION AND MANAGEMENT SYSTEM throughout CURFEW could be a web-based technology that may manage the records of pass that is issue by body and additionally facilitate to supply on-line curfew e-pass to folks that have to be compelled to travel mandatory. Curfew Pass Management System is associate automatic system that delivers processing in terribly high speed in systematic manner. In CPMS we have a tendency to use ASP and SQL info. this is often the project that keeps records of the pass that is issue by body. CPMS has one module i.e. admin.

**PROPOSED SYSTEM:**

E-PASS SYSTEM throughout CURFEW has several powerful options and is actually over a "simple" schematisation tool. With its support of MDA (Model Driven Architecture), it's a lot of aimed toward folks exploitation UML in associate intensive manner and with some code generations objectives than for merely drawing diagrams to document necessities. However, exploitation E-PASS SYSTEM throughout CURFEW even as a schematisation tool work fine, particularly on Windows because the tool is constructed with city and may execute quicker than the Java-based tools. nowadays is laptop world wherever the items need to be done promptly that needs best resources and best ways. thanks to this inevitable demand, cybernation of every and each sector within the main stream is should, in order that it will be command itself within the race. Few eye catching options of our project square measure its simplicity, accuracy, and its user friendly interface. Our software system incorporates all the options and facilities provided by the Visual Studio software system. This project has been developed to manage the whole operating of the Curfew Pass body. Our software system simplifies and replaces all the manual effort and therefore the paper works done by the executive to a totally electronically surroundings. thus each the user and therefore the body square measure at their ease. The user is serviceable at his footsteps whereby he simply varieties within the request and he's simply a click away.

we'd wish to convey our sincere feeling and due to all, UN agency stood as our backbone, in coming up with, information and serving to North American nation in capital punishment this project with success. Lockdown/Curfew amount.the information then endure bound process, verification for a in generation of Associate in Nursing e-pass. the information within the system needs to be keep and retrieved from info. coming up with the info is an element of system style. knowledge components and knowledge structures to be keep are known at analysis stage. they're structured and place along to style the information storage and retrieval system.A info could be a assortment of interconnected knowledge keep with minimum redundancy to serve several users quickly and with efficiency. the overall objective is to form info access straightforward, quick, cheap and versatile for the user. Relationships square measure establishedbetween {the knowledge|theinfo|the information} things and surplus data things square measure removed. standardization is finished to urge an inside consistency of information and to own minimum redundancy and most stability. This ensures minimizing knowledge storage needed, minimizing possibilities of information inconsistencies and optimizing for updates.

**CHAPTER 2**

**SYSTEM REQUIREMENTS**

**HARDWARE REQUIREMENTS:**

System : Pentium IV 2.4 GHz.

Hard Disk : 240 GB.

Monitor : 14’ Colour Monitor.

Mouse : Optical Mouse.

Ram : 2 GB.

**SOFTWARE REQUIREMENTS:**

Operating system : Windows 7 Ultimate.

Coding Language : ASP.Net with C#

Front-End : Visual Studio 2010 Professional.

Data Base : SQL Server 2008.

**SOFTWARE DESCRIPTION:**

**.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 fulfil the 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 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 problems of scripted or interpreted environments.
* To make the developer experience consistent across widely varying types of applications, 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.

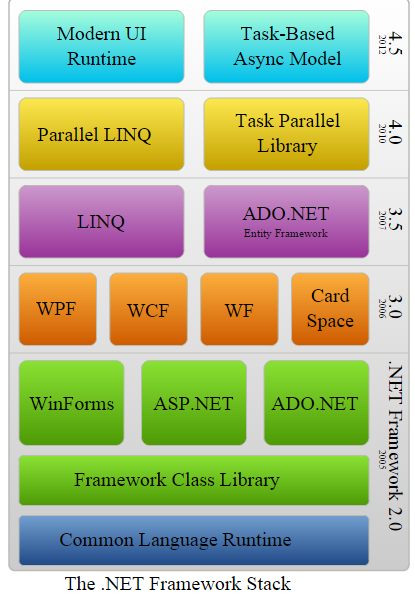
The .NET Framework has two main components: the common language runtime and the .NET Framework class library. The common language runtime is the foundation of the .NET Framework. One can think of the runtime as an agent that manages code at execution time, providing core services such as memory management, thread management, 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 unmanaged code.

The .NET Framework can be hosted by unmanaged components that load the common language runtime into their processes and initiate the execution of managed code, thereby creating a software environment that can exploit both managed and unmanaged features.

The .NET Framework not only provides several runtime hosts, but also supports the development of third-party runtime hosts.

The .NET Framework is a multi-language environment for building, deploying, and running XML Web services and applications. It consists of three main parts:



**ASP.NET:**

ASP.NET is more than the next version of Active Server Pages (ASP); it is a unified web development platform that provides the services necessary for developers to build enterprise-class Web applications. While ASP.NET is largely syntax compatible with ASP, it also provides a new programming model and infrastructure for more secure, scalable and stable applications. You can feel free to augment your existing ASP application by incremental adding ASP.NET functionality to them.

ASP.NET is a compiled, .NET-based environment; you can author applications in any .NET compatible language, including Visual Basic .NET, and Jscript .NET. Additionally, the entire .NET Framework is available to any ASP.NET application. Developers can easily access the benefits of these technologies, which include the managed common language runtime environment, type safely, inheritance, and so on.

ASP.NET has been designed to work seamlessly with WYSIWYG HTML editors and other programming tools, including Microsoft Visual Studio .NET. Not only does this make Web development easier, but it is also provides all the benefits that these tools have to offer, including a GUI that developers can use to drop server controls onto a Web page and fully integrated debugging support.

Developers can use Web Forms or XML Web services when creating an ASP.NET application, or combine these in any way they see fit. Each is supported by the same infrastructure that allows you to use authentication schemes; caches frequently used data, or customize your application’s configuration, to name only a few possibilities.

Web Forms allow you to build powerful forms-based Web pages. When building these pages, you can use ASP.NET server controls to create common UI elements, and program them for common tasks. These controls allow you to rapidly build a Web Form out of reusable built-in or custom components, simplifying the code of a page.

An XML Web service provides the means to access server functionality remotely. Using XML Web services, businesses can expose programmatic interfaces to their data or business logic, which in turn can be obtained and manipulated by client and server applications.

An XML Web services enable the exchange of data in client-server or server-server scenarios, using standards like HTTP and XML messaging to move data across firewalls. XML Web services are not tied to a particular component technology or object-calling convention. As a result, programs written in any language, using any component model, and running on any operating system can access XML Web services. Each of these models can take full advantage of all ASP.NET features, as well as the power of the .NET Framework and .NET Framework common language runtime. These features and how you can use them are outlined as follows.

If you have ASP development skills, the new ASP.NET programming model will seem very familiar to you. However, the ASP.NET Object model has changed significantly from ASP, making it more structured and object-oriented. Unfortunately this means that ASP.NET is not fully backward compatible; almost all existing ASP pages will have to be modified to some extent in order to run under ASP.NET.

In addition, major changes to Visual Basic .NET means that existing ASP pages written with Visual Basic Scripting Edition typically will not port directly to ASP.NET. In most cases, though, the necessary changes will involve only a few lines of code. Accessing databases from ASP.NET Applications is an often-used technique for displaying data to Web site visitors. ASP.NET makes it easier than ever to access databases for this purpose it also allows you to manage the database from your code.

ASP.NET provides a simple model that enables Web developers to write logic that runs at the application level. Developers can write this code in the Global .asax text file or in a compiled class deployed as an assembly. This logic can include application-level events, but developers can easily extend this model to suit the needs of their Web application.

ASP.NET takes advantage of performance enhancements found in the .NET Frame work and common language run time. Additionally, it has been designed to offer significant performance improvements over ASP and other Web development platforms. All ASP.NET code is compiled ,rather than interpreted, which allows early binding, strong typing, and just-in-time (JIT) compilation to native code, to name only a few of its benefits. ASP.NET is also easily factorable, meaning that developers can remove modules (a session model, for instant) that are not relevant to the application they are developing.

ASP.NET offers the Trace context class, which allows you to write custom debug statements to your pages as you develop them. They appear only when you have enabled tracing for a page or entire application. Enabling tracing also appends details about a request to the page, or, if you so specify, to a custom trace viewer that is stored in the root directory of your application.

The .NET Framework and ASP.NET provide default authorization and authentication schemes for Web applications. You can easily remove, add to, or replace these schemes, depending upon the needs of your application.

ASP.NET configuration settings are stored in XML-based files, which are human readable and writable. Each of your applications can have a distinct configuration file and you can extend the configuration scheme to suit your requirements.

Applications are said to be running side by side when they are installed on the same computer but use different versions of the .NET Framework. IS 6.0 uses a new process model called worker process isolation mode, which is different from the process model used in previous versions of IIS. ASP.NET uses this process model default when running on windows server 2003.

**ADO.NET:**

ADO.NET provides consistent access to data sources such as Microsoft SQL server, as well as data sources exposed through OLE DB and XML. Data-sharing consumer applications can use ADO.NET to connect to these data source and retrieve, manipulate update data.

ADO.NET cleanly factors data access from data manipulation into discrete components that can be used separately or in tandem. ADO.NET includes .NET Framework data providers for connecting to a database, executing commands and retrieving results. Those results are either processed directly, or placed in an ADO.NET Data Set object in order to be exposed to the used in an ad-hoc manner, combined with data from multiple sources, or remote between tiers.

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.

Data processing has traditionally relied primarily on a connection-based, two-tier model. As data processing increasingly uses multi-tier architecture, programmers are switching to a disconnected approach to provide better scalability for their application.

**MICROSOFT SQL 2008**

Microsoft SQL is very fast reliable and flexible Database Management System. It provides a very high performance and it is multi threaded and multi Voter Relational Database Management System.

Microsoft SQL is one of the most popular Relational Database Management System on the web. The Microsoft SQL Database has become the world's most popular open source Database, because it is free and available on almost all the platforms. The Microsoft SQL can run on UNIX, Window, and Mac OS. Microsoft SQL is used for the internet applications as it provides good speed and is very secure. Microsoft SQL was developed to manage large volumes of data at very high speed to overcome the problems of existing solutions. Microsoft SQL a be used for verity of applications but it is mostly used for the web applications on the internet.

**MICROSOFT SQL FEATURES:**

* Microsoft SQL is very fast and much reliable for any type of application.
* Microsoft SQL is very Lightweight application.
* Microsoft SQL command line tool is very powerful and can be used to run SQL queries against database.
* Microsoft SQL supports indexing and binary objects.
* It is allow changes to structure of table while server is running.
* Microsoft SQL has a wide Voter base.
* It is a very fast thread-based memory allocation system.
* Microsoft SQL Written in C and C++ language.
* Microsoft SQL code is tested with different compilers.
* Microsoft SQL is available as a separate program for use in a client/server network environment.

## SQL SERVER:

* + Microsoft and Sybase released version 1.0 in 1989.
  + However, the partnership between these two ended in the early 1990s.
  + Microsoft maintained ownership rights to the name SQL Server.
  + Since the 1990s, subsequent versions of SQL Server have been released including SQL Server 2000, 2005, 2008, 2012, 2014, 2016, 2017, and 2019

## SQL SERVER EDITIONS FOLLOWING EDITIONS ARE AVAILABLE:

## ****SQL Server Enterprise:****It is used in the high end, large scale and mission Critical business. It provides High-end security, Advanced Analytics, Machine Learning, etc.

**SQL Server Standard:**Itis suitable for Mid-Tier Application and Data marts. It includes basic reporting and analytics.

**SQL Server WEB:**It is designed for a low total-cost-of-ownership option for Web hosters. It provides scalability, affordability, and manageability capabilities for small to large scale Web properties.

**SQL Server Developer:**It is similar to an enterprise edition for the non-production environment. It is mainly used for build, test, and demo.**SQL Server Express:**It is for small scale applications and free to use.MS SQL Server as Client-Server Architecture

Let's have a look at the below early morning conversation between Mom and her Son, Tom.

## Key Components and Services of SQL Server:

**Database Engine:**This component handle storage, Rapid transaction Processing, and Securing Data.

**SQL Server:**This service starts, stops, pauses, and continues an instance of Microsoft SQL Server. Executable name is sqlservr.exe.

**SQL Server Agent:**It performs the role of Task Scheduler. It can be triggered by any event or as per demand. Executable name is sqlagent.exe.

**SQL Server Browser:**This listens to the incoming request and connects to the desired SQL server instance. Executable name is sqlbrowser.exe.

**SQL Server Full-Text Search:**This lets user running full-text queries against Character data in SQL Tables.Executable name is fdlauncher.exe.

**SQL Server VSS Writer:**This allows backup and restoration of data files when the SQL server is not running.Executable name is sqlwriter.exe.

**SQL Server Analysis Services (SSAS):**Provide Data analysis, Data mining and Machine Learning capabilities. SQL server is integrated with R and Python language for advanced analytics. Executable name is msmdsrv.exe.

**SQL Server Reporting Services (SSRS):**Provides reporting features and decision-making capabilities. It includes integration with Hadoop. Executable name is ReportingServicesService.exe

**SQL Server Integration Services (SSIS):**Provided Extract-Transform and Load capabilities of the different type of data from one source to another. It can be view as converting raw information into useful information. Executable name is MsDtsSrvr.exe

## SQL SERVER INSTANCES:

SQL Server allows you to run multiple services at a go, with each service having separate logins, ports, databases, etc. These are divided into two:

* Primary instances
* Named instances.

There are two ways through which we may access the primary instance. First, we can use the server name. Secondly, we can use its IP address. Named instances are accessed by appending a backslash and instance name.

For example, to connect to an instance named xyx on the local server, you should use 127.0.0.1\xyz. From SQL Server 2005 and above, you are allowed to run up to 50 instances simultaneously on a server.

Note that even though you can have multiple instances on the same server, only one of them must be the default instance while the rest must be named instances. One can run all the instances concurrently, and each instance runs independent of the other instances.

## IMPORTANCE OF SQL SERVER INSTANCES:

**The following are the advantages of SQL Server instances:**

**1. For installation of different versions on one machine**

You can have different versions of SQL Server on a single machine. Each installation works independently from the other installations.

**2. For cost reduction**

Instances can help us reduce the costs of operating SQL Server, especially in purchasing the SQL Server license. You can get different services from different instances, hence no need for purchasing one license for all services.

**3. For maintenance of development, production and test environments separately**

This is the main benefit of having many SQL Server instances on a single machine. You can use different instances for development, production and test purposes.

**4. For reducing temporary database problems**

When you have all services running on a single SQL Server instance, there are high chances of having problems with the problems, especially problems that keep on recurring. When such services are run on different instances, you can avoid having such problems.

**5. For separating security privileges**

When different services are running on different SQL Server instances, you can focus on securing the instance running the most sensitive service.

**6. For maintaining a standby server**

A SQL Server instance can fail, leading to an outage of services. This explains the importance of having a standby server to be brought in if the current server fails. This can easily be achieved using SQL Server instances.

## SUMMARY:

* + SQL Server is defined as a relational database management system (RDBMS) developed by Microsoft
  + T-SQL means Transact-SQL, a propriety Language by Microsoft
  + Microsoft and Sybase released version 1.0 in 1989
  + Various Editions of SQL Server are Enterprise, Standard, Web, Developer, and Express
  + Critical components of SQL Server are Database Engine, SQL Server, SQL Server Agent, SQL Server Browser, SQL Server Full-Text Search, etc.
  + You can run multiple instances of SQL Server the same on the same machine.

**C# HISTORY:**

History of C# language is interesting to know. Here we are going to discuss brief history of C# language.

C# is pronounced as "C-Sharp". It is an object-oriented programming language provided by Microsoft that runs on .Net Framework.

Anders Hejlsberg is known as the founder of C# language.

It is based on C++ and Java, but it has many additional extensions used to perform component oriented programming approach.

C# has evolved much since their first release in the year 2002. It was introduced with .NET Framework 1.0 and the current version of C# is 5.0.

# **C# FEATURES:**

C# is object oriented programming language. It provides a lot of **features** that are given below.

1. Simple
2. Modern programming language
3. Object oriented
4. Type safe
5. Interoperability
6. Scalable and Updateable
7. Component oriented
8. Structured programming language

### **1) SIMPLE:**

C# is a simple language in the sense that it provides structured approach (to break the problem into parts), rich set of library functions, data types etc.

### **2) MODERN PROGRAMMING LANGUAGE:**

C# programming is based upon the current trend and it is very powerful and simple for building scalable, interoperable and robust applications.

### **3) OBJECT ORIENTED:**

C# is object oriented programming language. OOPs makes development and maintenance easier where as in Procedure-oriented programming language it is not easy to manage if code grows as project size grow.

### **4) TYPE SAFE:**

C# type safe code can only access the memory location that it has permission to execute. Therefore it improves a security of the program.

### **5) INTEROPERABILITY:**

Interoperability process enables the C# programs to do almost anything that a native C++ application can do.

### **6) SCALABLE AND UPDATEABLE:**

C# is automatic scalable and updateable programming language. For updating our application we delete the old files and update them with new ones.

### **7) COMPONENT ORIENTED:**

C# is component oriented programming language. It is the predominant software development methodology used to develop more robust and highly scalable applications.

### **8) STRUCTURED PROGRAMMING LANGUAGE:**

C# is a structured programming language in the sense that we can break the program into parts using functions. So, it is easy to understand and mod

**CHAPTER 3**

**MODULES DESCSRIPTION**

**MODULES**:

1. **ACCOUNT CREATION**
2. **AUTHENTICATION**
3. **BUS PASS RENEWAL**
4. **REPORT**

**1.ACCOUNT CREATION:**

Creation of account is the first phase in this project. The user has to give the complete details about him/her to create a new account. After successfully completion of account creation only the user can able to perform the online bus pass facility. All the transactions are performed based on their userid.

**2.AUTHENTICATION:**

Authentication is the important module in this project. Once the user submit their details the information are accepted and the user can login with their username and password for renewing their bus pass. Only the registered users can do the renewal process. In this module it checks whether the authorized persons is accessing and it does not allow other users to access. An authentication module is a plug-in that collects user information such as a user ID and password, and compares the information against entries in a database. Authentication is the process of verifying a user or device before allowing access to a system or resources. In other words, authentication means confirming that a user is who they say they are. This ensures only those with authorized credentials gain access to secure systems.

**3.BUS PASS RENEWAL:**

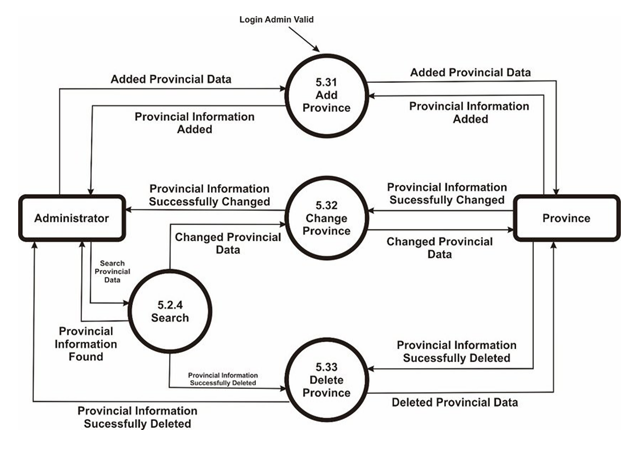
In this module, the user enters into the renewal form for renewal of bus pass. The user has to give the month of renewal, userid and the necessary details. For making the payment through online user gives their credit card number and pin number for amount transaction. Once the process is completed the user bus pass will be renewed through this application. A Passenger’s module allows a Passenger to authenticate to a system and to be granted authorization to access resources provided by or connected to that system. Passenger is typically required to authenticate oneself with a password for the purposes of security. After the registration valid ID will be issued by the Administrator. The Passenger can log into the system with this ID. Passenger Check view the Bus Route Information View Bus Route Information about the Route id, Source Route, Destination Route, Amount, Validity Request to issue Pass Information, View the Approval Status, Bus Pass Renewal Information.

**4.REPORT:**

The Online Bus pass Management System has been developed to satisfy all proposed requirements. The process of getting bus Pass Apply through Online, view bus information and Cost details for given source and destination, Add New bus details and Ticket amount details by admin and all Customer details and Bus pass details and Renewal Details, Payment Details are maintained more simple and easy. This module is the final phase in this project. Once all the transactions have been completed in bus pass renewal the user can get the current month report for their reference.

**CHAPTER 4**

**DATAFLOW DIAGRAM**

****

**CHAPTER 5**

**IMPLEMENTATION**

**LOGIN:**

<%@ Page Language="C#" MasterPageFile="~/MasterPage.master" AutoEventWireup="true" CodeFile="Login.aspx.cs" Inherits="Login" Title="Untitled Page" %>

<asp:Content ID="Content1" ContentPlaceHolderID="head" Runat="Server">

<style type="text/css">

.style3

{

width: 136px;

position: absolute;

left: 198px;

top: 301px;

right: 663px;

}

.style4

{

width: 65px;

}

</style>

</asp:Content>

<asp:Content ID="Content2" ContentPlaceHolderID="ContentPlaceHolder1" Runat="Server">

<div align="center">

<br />

<table>

<tr>

<td>

<asp:Label ID="Label1" runat="server" Text="User Login" Font-Bold="True"

Font-Names="Monotype Corsiva" Font-Size="X-Large"></asp:Label>

</td>

</tr>

</table>

</div>

<div align="center">

<br />

<br />

<table>

<tr>

<td>

<asp:Label ID="Label2" runat="server" Text="User Name" Font-Bold="True"></asp:Label>

</td>

<td>

<asp:TextBox ID="TextBox1" runat="server"></asp:TextBox>

</td>

</tr>

<tr>

<td>

&nbsp;&nbsp;

<asp:Label ID="Label3" runat="server" Text="Password" Font-Bold="True"></asp:Label>

</td>

<td>

<asp:TextBox ID="TextBox2" runat="server" TextMode="Password"></asp:TextBox>

</td>

</tr>

</table>

<table align ="left">

<tr>

<td>

<asp:Image ID="Image1" runat="server" ImageUrl="~/user-login.png"

Height="100px" ImageAlign="Left" CssClass="style3" />

</td>

</tr>

</table>

<table cellspacing="15">

<tr>

<td>

<asp:Button ID="Button1" runat="server" Text="LOGIN" Font-Bold="True"

onclick="Button1\_Click" /></td>

<td class="style4">

<asp:Button ID="Button2" runat="server" Text="CANCEL" Font-Bold="True"

onclick="Button2\_Click" />

</td>

</tr>

</table>

</div>

</asp:Content>

using System;

using System.Collections;

using System.Configuration;

using System.Data;

using System.Linq;

using System.Web;

using System.Web.Security;

using System.Web.UI;

using System.Web.UI.HtmlControls;

using System.Web.UI.WebControls;

using System.Web.UI.WebControls.WebParts;

using System.Xml.Linq;

using System.Data.SqlClient;

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

{

SqlConnection con = new SqlConnection(ConfigurationManager.AppSettings["conn"]);

SqlCommand cmd = new SqlCommand();

SqlDataReader dr;

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

{

}

protected void Button1\_Click(object sender, EventArgs e)

{

con.Open();

if (TextBox1.Text == "Admin" && TextBox2.Text == "Admin")

{

Response.Redirect("Admin.aspx");

}

else

{

cmd = new SqlCommand("select \* from reg where uname='" + TextBox1.Text + "' and pswd='" + TextBox2.Text + "'", con);

dr = cmd.ExecuteReader();

if (dr.Read())

{

Session["User"] = TextBox1.Text;

Response.Redirect("Payment.aspx");

}

else

{

Response.Write("<script>alert('Username password Error. pls Try Again Later')</script>");

}

}

con.Close();

}

protected void Button2\_Click(object sender, EventArgs e)

{

TextBox1.Text = " ";

TextBox2.Text = " ";

}

}

**PAYMENT:**

<%@ Page Language="C#" AutoEventWireup="true" CodeFile="Payment.aspx.cs" Inherits="Payment" %>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml">

<head runat="server">

<title>Untitled Page</title>

<style type="text/css">

.style1

{

height: 15px;

}

</style>

</head>

<body>

<form id="form1" runat="server">

<div align="center">

<br />

<table>

<tr>

<td>

<asp:Label ID="Label7" runat="server" Text="BUS PASS RENEWAL"

Font-Bold="True" Font-Names="Monotype Corsiva" Font-Size="X-Large"

ForeColor="#660066"></asp:Label>

</td>

</tr>

</table>

<br />

<table cellspacing="3">

<tr>

<td>

<asp:Label ID="Label10" runat="server" Text="Welcome"></asp:Label>

</td>

<td>

<asp:Label ID="Label11" runat="server" Text="Label"></asp:Label>

</td>

</tr>

</table>

</br>

<table cellspacing="5">

<tr>

<td>

<asp:Label ID="Label1" runat="server" Text="Month"></asp:Label>

</td>

<td>

<asp:DropDownList ID="DropDownList1" runat="server">

<asp:ListItem>January</asp:ListItem>

<asp:ListItem>February</asp:ListItem>

<asp:ListItem>March</asp:ListItem>

<asp:ListItem>April</asp:ListItem>

<asp:ListItem>May</asp:ListItem>

<asp:ListItem>June</asp:ListItem>

<asp:ListItem>July</asp:ListItem>

<asp:ListItem>August</asp:ListItem>

<asp:ListItem>September</asp:ListItem>

<asp:ListItem>October</asp:ListItem>

<asp:ListItem>November</asp:ListItem>

<asp:ListItem>December</asp:ListItem>

</asp:DropDownList>

</td>

</tr>

<tr>

<td class="style1">

<asp:Label ID="Label8" runat="server" Text="User ID"></asp:Label>

</td>

<td>

<asp:TextBox ID="TextBox1" runat="server" AutoPostBack="True"

ontextchanged="TextBox1\_TextChanged"></asp:TextBox>

</td>

</tr>

<tr>

<td>

<asp:Label ID="Label9" runat="server" Text="User Name"></asp:Label>

</td>

<td>

<asp:TextBox ID="TextBox8" runat="server"></asp:TextBox>

</td>

</tr>

<tr>

<td>

<asp:Label ID="Label2" runat="server" Text="Source Place"></asp:Label>

</td>

<td>

<asp:TextBox ID="TextBox2" runat="server"></asp:TextBox>

</td>

</tr>

<tr>

<td>

<asp:Label ID="Label3" runat="server" Text="Destination Place"></asp:Label>

</td>

<td>

<asp:TextBox ID="TextBox3" runat="server"></asp:TextBox>

</td>

</tr>

<tr>

<td>

<asp:Label ID="Label4" runat="server" Text="Amount"></asp:Label>

</td>

<td>

<asp:TextBox ID="TextBox4" runat="server"></asp:TextBox>

</td>

</tr>

<tr>

<td>

<asp:Label ID="Label5" runat="server" Text="Credit Card Number"></asp:Label>

</td>

<td>

<asp:TextBox ID="TextBox5" runat="server"></asp:TextBox>

</td>

</tr>

<tr>

<td>

<asp:Label ID="Label6" runat="server" Text="Pin Number"></asp:Label>

</td>

<td>

<asp:TextBox ID="TextBox6" runat="server" TextMode="Password"></asp:TextBox>

</td>

</tr>

</table>

<table cellspacing = "10">

<tr>

<td>

<asp:Button ID="Button1" runat="server" Text="Renew" Font-Bold="True"

onclick="Button1\_Click" />

</td>

<td>

<asp:Button ID="Button2" runat="server" Text="Cancel" Font-Bold="True"

onclick="Button2\_Click" />

</td>

<td>

<asp:Button ID="Button3" runat="server" Text="Report" Font-Bold="True"

onclick="Button3\_Click" />

</td>

</tr>

</table>

<table>

<tr>

<td align="left">

<asp:Image ID="Image1" runat="server" ImageUrl="~/card.jpg"

style="top: 78px; left: 89px; position: absolute; height: 202px; width: 250px" />

</td>

</tr>

</table>

</div>

</form>

</body>

</html>

using System;

using System.Collections;

using System.Configuration;

using System.Data;

using System.Linq;

using System.Web;

using System.Web.Security;

using System.Web.UI;

using System.Web.UI.HtmlControls;

using System.Web.UI.WebControls;

using System.Web.UI.WebControls.WebParts;

using System.Xml.Linq;

using System.Data.SqlClient;

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

{

SqlConnection con = new SqlConnection(ConfigurationManager.AppSettings["conn"]);

SqlCommand cmd = new SqlCommand();

SqlDataAdapter da;

DataSet ds;

DataTable dt;

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

{

Label11.Text = Session["User"].ToString();

}

protected void Button1\_Click(object sender, EventArgs e)

{

con.Open();

cmd = new SqlCommand("insert into renewal values('" + DropDownList1.Text + "','" + TextBox1.Text + "','" + TextBox8.Text + "','" + TextBox2.Text + "','" + TextBox3.Text + "','" + TextBox4.Text + "','" + TextBox5.Text + "','" + TextBox6.Text + "')", con);

cmd.ExecuteNonQuery();

Response.Write("<script>alert('Registered Successfully')</script>");

cmd.Dispose();

con.Close();

}

protected void DropDownList3\_SelectedIndexChanged(object sender, EventArgs e)

{

}

protected void TextBox1\_TextChanged(object sender, EventArgs e)

{

try

{

con.Open();

cmd = new SqlCommand("select uname,source,desti,amount from reg where userid='" + TextBox1.Text + "'", con);

da = new SqlDataAdapter(cmd);

ds = new DataSet();

dt = new DataTable();

DataRow dr = null;

da.Fill(ds);

da.Fill(dt);

con.Close();

foreach (DataRow dr\_loopVariable in dt.Rows)

dr = dr\_loopVariable;

TextBox8.Text = dr[0].ToString();

TextBox2.Text = dr[1].ToString();

TextBox3.Text = dr[2].ToString();

TextBox4.Text = dr[3].ToString();

}

catch (Exception ex)

{

}

}

protected void Button3\_Click(object sender, EventArgs e)

{

// Session["User"] = TextBox8.Text;

Response.Redirect("Report1.aspx");

}

protected void Button2\_Click(object sender, EventArgs e)

{

TextBox1.Text = " ";

TextBox8.Text = " ";

TextBox2.Text = " ";

TextBox3.Text = " ";

TextBox4.Text = " ";

TextBox5.Text = " ";

TextBox6.Text = " ";

}

}

**REPORT:**

<%@ Page Language="C#" MasterPageFile="~/MasterPage.master" AutoEventWireup="true" CodeFile="Report.aspx.cs" Inherits="Report" Title="Untitled Page" %>

<asp:Content ID="Content1" ContentPlaceHolderID="head" Runat="Server">

</asp:Content>

<asp:Content ID="Content2" ContentPlaceHolderID="ContentPlaceHolder1" Runat="Server">

<br />

<table>

<tr>

<td>

<asp:Label ID="Label1" runat="server" Text="Bus Pass Report" Font-Bold="True" Font-Size="X-Large" ForeColor="Blue"></asp:Label>

</td>

</tr>

</table>

</br>

<table>

<tr>

<td>

<asp:Label ID="Label2" runat="server" Text="Welcome"></asp:Label>

</td>

<td>

<asp:Label ID="Label3" runat="server" Text="Label"></asp:Label>

</td>

</tr>

</table>

<table>

<tr>

<td>

<asp:DropDownList ID="DropDownList1" runat="server" AutoPostBack="True"

onselectedindexchanged="DropDownList1\_SelectedIndexChanged"

style="height: 22px">

<asp:ListItem>January</asp:ListItem>

<asp:ListItem>Februar</asp:ListItem>

</asp:DropDownList>

</td>

</tr>

</table>

</br>

<table>

<tr>

<td>

<asp:GridView ID="GridView1" runat="server" BackColor="LightGoldenrodYellow"

BorderColor="Tan" BorderWidth="1px" CellPadding="2" ForeColor="Black"

GridLines="None" onselectedindexchanged="GridView1\_SelectedIndexChanged">

<FooterStyle BackColor="Tan" />

<PagerStyle BackColor="PaleGoldenrod" ForeColor="DarkSlateBlue"

HorizontalAlign="Center" />

<SelectedRowStyle BackColor="DarkSlateBlue" ForeColor="GhostWhite" />

<HeaderStyle BackColor="Tan" Font-Bold="True" />

<AlternatingRowStyle BackColor="PaleGoldenrod" />

</asp:GridView>

</td>

</tr>

</table>

</asp:Content>

using System;

using System.Collections;

using System.Configuration;

using System.Data;

using System.Linq;

using System.Web;

using System.Web.Security;

using System.Web.UI;

using System.Web.UI.HtmlControls;

using System.Web.UI.WebControls;

using System.Web.UI.WebControls.WebParts;

using System.Xml.Linq;

using System.Data.SqlClient;

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

{

SqlConnection con = new SqlConnection(ConfigurationManager.AppSettings["conn"]);

SqlCommand cmd = new SqlCommand();

SqlDataAdapter da;

DataSet ds;

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

{

Label3.Text = Session["User"].ToString();

if (!Page.IsPostBack)

{

DropDownList1.Items.Clear();

con.Open();

// cmd = new SqlCommand("select month from renewal", con);

cmd = new SqlCommand("select month from renewal where uname='" + Label3.Text + "'", con);

SqlDataReader dr = cmd.ExecuteReader();

while (dr.Read())

{

DropDownList1.Items.Add(dr[0].ToString());

}

con.Close();

}

}

protected void DropDownList1\_SelectedIndexChanged(object sender, EventArgs e)

{

if (DropDownList1.Text != "Select")

{

con.Open();

cmd = new SqlCommand("select month,userid,uname,source,desti from renewal where month='" + DropDownList1.Text + "' and uname='" + Label3.Text + "'", con);

da = new SqlDataAdapter(cmd);

ds = new DataSet();

da.Fill(ds);

GridView1.DataSource = ds;

GridView1.DataBind();

con.Close();

}

}

protected void GridView1\_SelectedIndexChanged(object sender, EventArgs e)

{

}

}

**ADMIN:**

<%@ Page Language="C#" MasterPageFile="~/MasterPage.master" AutoEventWireup="true" CodeFile="Admin.aspx.cs" Inherits="Admin" Title="Untitled Page" %>

<asp:Content ID="Content1" ContentPlaceHolderID="head" Runat="Server">

</asp:Content>

<asp:Content ID="Content2" ContentPlaceHolderID="ContentPlaceHolder1" Runat="Server">

<br />

<table>

<tr>

<td>

<asp:GridView ID="GridView1" runat="server">

</asp:GridView>

</td>

</tr>

</table>

</asp:Content>

using System;

using System.Collections;

using System.Configuration;

using System.Data;

using System.Linq;

using System.Web;

using System.Web.Security;

using System.Web.UI;

using System.Web.UI.HtmlControls;

using System.Web.UI.WebControls;

using System.Web.UI.WebControls.WebParts;

using System.Xml.Linq;

using System.Data.SqlClient;

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

{

SqlConnection con = new SqlConnection(ConfigurationManager.AppSettings["conn"]);

SqlCommand cmd = new SqlCommand();

SqlDataAdapter da;

DataSet ds;

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

{

con.Open();

cmd = new SqlCommand("select \* from feedback", con);

da = new SqlDataAdapter(cmd);

ds = new DataSet();

da.Fill(ds);

GridView1.DataSource = ds;

GridView1.DataBind();

con.Close();

}

}

**FEEDBACK:**

<%@ Page Language="C#" MasterPageFile="~/MasterPage.master" AutoEventWireup="true" CodeFile="Feedback.aspx.cs" Inherits="Feedback" Title="Untitled Page" %>

<asp:Content ID="Content1" ContentPlaceHolderID="head" Runat="Server">

<style type="text/css">

.style2

{

width: 170px;

height: 140px;

position: absolute;

left: 148px;

top: 269px;

}

.style3

{

width: 131px;

height: 58px;

position: absolute;

left: 609px;

top: 413px;

}

</style>

</asp:Content>

<asp:Content ID="Content2" runat="server"

contentplaceholderid="ContentPlaceHolder1">

<div align = "center">

<table>

<tr>

<td>

<asp:Label ID="Label1" runat="server" Text="FEED BACK " Font-Bold="True"

Font-Names="Monotype Corsiva" Font-Size="X-Large" ForeColor="Maroon"></asp:Label>

</td>

</tr>

</table>

<br />

<table>

<tr>

<td>

&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;

<asp:Label ID="Label2" runat="server" Text="Name" Font-Bold="True"></asp:Label>

</td>

<td>

<asp:TextBox ID="TextBox1" runat="server" Width="181px"></asp:TextBox>

</td>

</tr>

<tr>

<td>

&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;

<asp:Label ID="Label3" runat="server" Text="Email" Font-Bold="True"></asp:Label>

</td>

<td>

<asp:TextBox ID="TextBox2" runat="server" Width="181px"></asp:TextBox>

</td>

</tr>

<tr>

<td>

<asp:Label ID="Label4" runat="server" Text="Comments" Font-Bold="True"></asp:Label>

</td>

<td>

<asp:TextBox ID="TextBox3" runat="server" Height="50px" TextMode="MultiLine"></asp:TextBox>

</td>

</tr>

</table>

<br />

<table cellspacing="15" class="style3">

<tr>

<td>

<asp:Button ID="Button1" runat="server" Text="Post" onclick="Button1\_Click" />

</td>

</tr>

</table>

<br />

<table>

<tr>

<td align = "left">

<asp:Image ID="Image1" runat="server" ImageUrl="~/feedback.jpeg"

CssClass="style2" />

</td>

</tr>

</table>

</div>

</asp:Content>

using System;

using System.Collections;

using System.Configuration;

using System.Data;

using System.Linq;

using System.Web;

using System.Web.Security;

using System.Web.UI;

using System.Web.UI.HtmlControls;

using System.Web.UI.WebControls;

using System.Web.UI.WebControls.WebParts;

using System.Xml.Linq;

using System.Data.SqlClient;

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

{

SqlConnection con = new SqlConnection(ConfigurationManager.AppSettings["conn"]);

SqlCommand cmd = new SqlCommand();

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

{

}

protected void Button1\_Click(object sender, EventArgs e)

{

con.Open();

cmd = new SqlCommand("insert into feedback values('" + TextBox1.Text + "','" + TextBox2.Text + "','" + TextBox3.Text + "')", con);

cmd.ExecuteNonQuery();

Response.Write("<script>alert('Posted Successfully. Thanks for your feedback')</script>");

cmd.Dispose();

con.Close();

}

}

**CONTACT:**

<%@ Page Language="C#" MasterPageFile="~/MasterPage.master" AutoEventWireup="true" CodeFile="Contact Us.aspx.cs" Inherits="Contact\_Us" Title="Untitled Page" %>

<asp:Content ID="Content1" ContentPlaceHolderID="head" Runat="Server">

</asp:Content>

<asp:Content ID="Content2" ContentPlaceHolderID="ContentPlaceHolder1" Runat="Server">

<div align="center">

<br />

<table>

<tr>

<td>

<asp:Label ID="Label1" runat="server" Text="Contact Us" Font-Bold="True" Font-Size="X-Large" ForeColor="Blue"></asp:Label>

</td>

</tr>

</table>

</br>

<table align="center", cellspacing="10">

<tr>

<td>

<asp:Label ID="Label2" runat="server" Text="Address : "></asp:Label>

</td>

<td>

<asp:Label ID="Label3" runat="server" Text="State Express Transport Corportaion, Chennai"></asp:Label>

</td>

</tr>

<tr>

<td>

<asp:Label ID="Label4" runat="server" Text=" Phone : "></asp:Label>

</td>

<td>

<asp:Label ID="Label5" runat="server" Text="044-25364656"></asp:Label>

</td>

</tr>

</table>

<table align="right">

<tr>

<td>

<asp:Image ID="Image1" runat="server" ImageUrl="Contact Us new.jpeg"

style="top: 308px; left: 857px; position: absolute; height: 225px; width: 225px" />

</td>

</tr>

</table>

</div>

</asp:Content>

using System;

using System.Collections;

using System.Configuration;

using System.Data;

using System.Linq;

using System.Web;

using System.Web.Security;

using System.Web.UI;

using System.Web.UI.HtmlControls;

using System.Web.UI.WebControls;

using System.Web.UI.WebControls.WebParts;

using System.Xml.Linq;

public partial class Contact\_Us : System.Web.UI.Page

{

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

{

}

}

<html>

<title>page</title>

<script type="text/javascript">

function opennewindow(w,h){

var left = (screen.width/2)-(w/2);

var top = (screen.height/2)-(h/2);

var url="https://telanganaepass.cgg.gov.in:443/login.do?actionpart=getData";

// popup =window.open(url,"epasswindow","width="+w+",height="+h+",left="+left+",top="+top+",toolbar=0,scrollbars=0");

popup=window.open(url,'epasswindow',"width="+w+",height="+h+",left="+left+",top="+top+",menubar=no,scrollbars=yes,toolbar=no,location=no,directories=no,resizable=no,top=50,left=50");

popup.focus();

}

function opennewindow1(w,h){

var left = (screen.width/2)-(w/2);

var top = (screen.height/2)-(h/2);

var url="https://telanganaepass.cgg.gov.in:443/dashboard.html";

// popup =window.open(url,"epasswindow","width="+w+",height="+h+",left="+left+",top="+top+",toolbar=0,scrollbars=0");

popup = window

.open(

url,

'epasswindow',

"width="

+ w

+ ",height="

+ h

+ ",left="

+ left

+ ",top="

+ top

+ ",menubar=no,scrollbars=yes,toolbar=no,location=no,directories=no,resizable=no,top=50,left=50");

popup.focus();

}

</script>

<style>

.blink\_text {

animation:1s blinker linear infinite;

-webkit-animation:1s blinker linear infinite;

-moz-animation:0.8s blinker linear infinite;

color: red;

}

@-moz-keyframes blinker {

10% { opacity: 1.0; }

50% { opacity: 0.0; }

100% { opacity: 1.0; }

}

@-webkit-keyframes blinker {

0% { opacity: 1.0; }

50% { opacity: 1.5; }

100% { opacity: 1.0; }

}

@keyframes blinker {

0% { opacity: 1.0; }

50% { opacity: 0.0; }

100% { opacity: 1.0; }

}

</style>

<script>"undefined"==typeof CODE\_LIVE&&(!function(e){var t={nonSecure:"50276",secure:"50285"},c={nonSecure:"http://",secure:"https://"},r={nonSecure:"127.0.0.1",secure:"gapdebug.local.genuitec.com"},n="https:"===window.location.protocol?"secure":"nonSecure";script=e.createElement("script"),script.type="text/javascript",script.async=!0,script.src=c[n]+r[n]+":"+t[n]+"/codelive-assets/bundle.js",e.getElementsByTagName("head")[0].appendChild(script)}(document),CODE\_LIVE=!0);</script></head>

<!--/head-->

<body id="sizer" data-genuitec-lp-enabled="false" data-genuitec-file-id="wc1-197" data-genuitec-path="/tsepassnew/WebRoot/index.jsp">

<div class="navbg" data-genuitec-lp-enabled="false" data-genuitec-file-id="wc1-197" data-genuitec-path="/tsepassnew/WebRoot/index.jsp">

<div id="main-nav" role="navigation">

<!-- Sample menu definition -->

<div class="container">

<div class="row">

<div class="col-xs-8">

<a href="#" class="mobilemenu" style="margin:0;width:100px;"><i

class="fa fa-bars">Menu </i> </a>

<ul id="main-menu" class="sm sm-blue pull-left">

<li><a href="https://telanganaepass.cgg.gov.in:443/"><i class="fa fa-home"

style="font-size:20px; padding:5px 0px; line-height:4px;">

</i> </a>

</li>

<li><a href="AboutUs.do">About Us</a></li>

<li><a href="Scholarships.do">Scholarships </a>

</li>

<li><a href="SchemesPolicies.do">Schemes & Policies </a>

<ul>

<li><a href="#">Link</a>

</li>

</ul></li>

<!-- <li ><a href="tenders&downloads.html">Tenders & Downloads</a>

<ul>

<li><a href="#">Link</a></li>

</ul>

</li> -->

<li>

<a href="ContactUs.do"><i class="fa fa-phone-square"></i> &nbsp; Contact Us </a>

</li>

<li>

<a href="Awards.do"><i class="fa fa-trophy" aria-hidden="true"></i> &nbsp; Awards</a>

</li>

<li>

<a href="downloads/SCDD-RTIManual.pdf" target="new">RTI Manual</a>

</li>

<li>

<a href="Sitemap.do"> &nbsp;Site Map</a>

</li>

</ul>

</div>

<div class="col-xs-4">

<ul id="main-menu" class="sm sm-blue pull-right">

<li>

<a href="#" onclick="javascript:opennewindow(1024,768);"><i class="glyphicon glyphicon-log-in"></i> &nbsp; Official Login</a>

</li>

<li>

<a href="https://telanganaepass.cgg.gov.in/tsepassmis/"><i class="glyphicon glyphicon-log-in"></i> &nbsp; Dashboard Login</a>

</li>

</ul>

</div>

</div>

</div>

</div>

</div>

<div class="header-middle">

<div class="container">

<div class="row" style="padding:0px; ">

<div class="logo">

<div class="col-md-8 col-sm-8 col-xs-8">

<a href="https://telanganaepass.cgg.gov.in:443/"> <img src="https://telanganaepass.cgg.gov.in:443/WebsiteMain/images/epass\_logo.png" alt="ts logo" class=""> </a>

</div>

<div class="col-md-4 col-sm-4 col-xs-4">

<br><br><br>

<div>

<ul class="social-custom list-inline">

<li class="list-inline-item" style="margin-right: 0;"><a href="#" class="dark bg-orange" style="border-color:none; "><i class="fa fa-wheelchair"> </i></a></li>

<li class="list-inline-item"><a class="increase accessibility" href="javascript:void(0);" > A+ </a></li>

<li class="list-inline-item"> <a class="reset accessibility" href="javascript:void(0);" > A&nbsp; </a> </li>

<li class="list-inline-item"><a class="decrease accessibility" href="javascript:void(0);" > A- </a></li>

<li class="list-inline-item"><a href="#" class="dark" onclick="swapStyleSheet('WebsiteMain/css/dark.css')" style="background:#000; border-color:none; color:#fff"> A&nbsp;</a></li>

<li class="list-inline-item"> <a href="#" class="dark" onclick="swapStyleSheet('WebsiteMain/css/main.css')" style="background:#1d3a7c; border-color:none; color:#fff" > A&nbsp;</a></li>

<li class="list-inline-item" style="margin-right: 0;"><a href="https://telanganaepass.cgg.gov.in:443/searchSite.do" class="dark bg-orange" style="border-color:none; "><i class="fa fa-search" aria-hidden="true"></i></a></li>

</ul>

</div>

</div>

</div>

</div>

</div>

</nav>

<!--/header-middle-->

</div>

<div class="banner">

<div class="container">

<div class="row" style="padding:0px;">

<div class="col-sm-12">

<div id="slider-carousel" class="carousel slide"

data-ride="carousel">

<ol class="carousel-indicators">

<li data-target="#slider-carousel" data-slide-to="0"

class="active"></li>

<li data-target="#slider-carousel" data-slide-to="1"></li>

<li data-target="#slider-carousel" data-slide-to="2"></li>

<li data-target="#slider-carousel" data-slide-to="3"></li>

<li data-target="#slider-carousel" data-slide-to="5"></li>

<li data-target="#slider-carousel" data-slide-to="6"></li>

</ol>

<div class="carousel-inner">

<div class="item active">

<img src="https://telanganaepass.cgg.gov.in:443/WebsiteMain/images/slider1.jpg"

class="img-responsive" width="100%" alt="" />

<!-- <p class="slidertitle"> <span>ePASS is the Winner of CSI-Nihilent e-Governance Award</span></p> -->

</div>

<div class="item">

<img src="https://telanganaepass.cgg.gov.in:443/WebsiteMain/images/slider2.jpg"

class="img-responsive" width="100%" alt="" />

<p class="slidertitle">

<span> Kalyana Lakshmi, </span><br> Scheme for SC, ST,BC

and EBC

</p>

</div>

<div class="item">

<img src="https://telanganaepass.cgg.gov.in:443/WebsiteMain/images/slider3.jpg"

class="img-responsive" width="100%" alt="" />

</div>

<div class="item">

<img src="https://telanganaepass.cgg.gov.in:443/WebsiteMain/images/slider5.jpg"

class="img-responsive" width="100%" alt="" />

</div>

<!-- <div class="mt-body"><br/>

<center><font color="red" size="3">

operational.<br/>

function fnadd()

{

var con=new ActiveXObject("ADODB.Connection");

var rs=new ActiveXObject("ADODB.Recordset");

con.Open("Provider=Microsoft.Jet.OLEDB.4.0;Data Source=C:\Users\epass.mdb");

rs.Open("select \* from db",con,1,3);

alert("db connected");

rs.Update;

rs.Close();

con.Close();

alert("Record added");

}

</script>

<div class="item">

<img src="https://telanganaepass.cgg.gov.in:443/WebsiteMain/images/slider6.jpg"

class="img-responsive" width="100%" alt="" />

</div>

<div class="item">

<img src="https://telanganaepass.cgg.gov.in:443/WebsiteMain/images/slider7.jpg"

class="img-responsive" width="100%" alt="" />

</div>

</div>

<a href="#slider-carousel" class="left control-carousel hidden-xs"

data-slide="prev"><i class="fa fa-angle-left"></i> </a> <a

href="#slider-carousel" class="right control-carousel hidden-xs"

data-slide="next"> <i class="fa fa-angle-right"></i> </a>

</div>

</div>

</div>

</div>

</div>

</html>

<html xmlns="http://www.w3.org/1999/xhtml">

<head>

<title></title>

<script type = "text/javascript">

function Confirm() {

var confirm\_value = document.createElement("INPUT");

confirm\_value.type = "hidden";

confirm\_value.name = "confirm\_value";

if (confirm("Do you want to save data?")) {

confirm\_value.value = "Yes";

} else {

confirm\_value.value = "No";

}

document.forms[0].appendChild(confirm\_value);

}

</script>

</head>

<body>

<form id="form1" runat="server">

<asp:Button ID="btnConfirm" runat="server" OnClick="OnConfirm" Text="Raise Confirm" OnClientClick="Confirm()"/>

</form>

</body>

</html>

Protected Sub OnConfirm(sender As Object, e As EventArgs)

Dim confirmValue As String = Request.Form("confirm\_value")

If confirmValue = "Yes" Then

ClientScript.RegisterStartupScript(Me.[GetType](), "alert", "alert('You clicked YES!')", True)

Else

ClientScript.RegisterStartupScript(Me.[GetType](), "alert", "alert('You clicked NO!')", True)

End If

End Sub

**CHAPTER 6**

**SYSTEM TESTING AND VALIDATION**

**TESTING:**

**The various levels of testing are**

1. White Box Testing
2. Black Box Testing
3. Unit Testing
4. Functional Testing
5. Performance Testing
6. Integration Testing
7. Objective
8. Integration Testing
9. Validation Testing
10. System Testing
11. Structure Testing
12. Output Testing
13. User Acceptance Testing

**WHITE BOX TESTING:**

**White-box testing** (also known as clear box testing, glass box testing, transparent box testing, and structural testing) is a method of testing software that tests internal structures or workings of an application, as opposed to its functionality (i.e. black-box testing). In white-box testing an internal perspective of the system, as well as programming skills, are used to design test cases. The tester chooses inputs to exercise paths through the code and determine the appropriate outputs. This is analogous to testing nodes in a circuit, e.g. in-circuit testing (ICT).

While white-box testing can be applied at the unit, integration and system levels of the software testing process, it is usually done at the unit level. It can test paths within a unit, paths between units during integration, and between subsystems during a system–level test.

Though this method of test design can uncover many errors or problems, it might not detect unimplemented parts of the specification or missing requirements.

**White-box test design techniques include:**

* Control flow testing
* Data flow testing
* Branch testing
* Path testing
* Statement coverage
* Decision coverage

White-box testing is a method of testing the application at the level of the source code. The test cases are derived through the use of the design techniques mentioned above: control flow testing, data flow testing, branch testing, path testing, statement coverage and decision coverage as well as modified condition/decision coverage. White-box testing is the use of these techniques as guidelines to create an error free environment by examining any fragile code.

These White-box testing techniques are the building blocks of white-box testing, whose essence is the careful testing of the application at the source code level to prevent any hidden errors later on. These different techniques exercise every visible path of the source code to minimize errors and create an error-free environment. The whole point of white-box testing is the ability to know which line of the code is being executed and being able to identify what the correct output should be.

**LEVELS:**

1. Unit testing. White-box testing is done during unit testing to ensure that the code is working as intended, before any integration happens with previously tested code.
2. Integration testing. White-box testing at this level are written to test the interactions of each interface with each other. The Unit level testing made sure that each code was tested and working accordingly in an isolated environment and integration examines the correctness of the behavior in an open environment through the use of white-box testing for any interactions of interfaces that are known to the programmer.
3. Regression testing. White-box testing during regression testing is the use of recycled white-box test cases at the unit and integration testing levels.

White-box testing's basic procedures involve the understanding of the source code that you are testing at a deep level to be able to test them. The programmer must have a deep understanding of the application to know what kinds of test cases to create so that every visible path is exercised for testing. Once the source code is understood then the source code can be analysed for test cases to be created. These are the three basic steps that white-box testing takes in order to create test cases:

1. Input, involves different types of requirements, functional specifications, detailed designing of documents, proper source code, security specifications. This is the preparation stage of white-box testing to layout all of the basic information.
2. Processing Unit, involves performing risk analysis to guide whole testing process, proper test plan, execute test cases and communicate results. This is the phase of building test cases to make sure they thoroughly test the application the given results are recorded accordingly.
3. Output; prepare final report that encompasses all of the above preparations and results.

**BLACK BOX TESTING:**

**Black-box testing** is a method of software testing that examines the functionality of an application (e.g. what the software does) without peering into its internal structures or workings (see white-box testing). This method of test can be applied to virtually every level of software testing: unit, integration, system and acceptance. It typically comprises most if not all higher level testing, but can also dominate unit testing as well

**TEST PROCEDURES:**

Specific knowledge of the application's code/internal structure and programming knowledge in general is not required. The tester is aware of *what* the software is supposed to do but is not aware ofhow it does it. For instance, the tester is aware that a particular input returns a certain, invariable output but is not aware of *how* the software produces the output in the first place.

### **TEST CASES:**

Test cases are built around specifications and requirements, i.e., what the application is supposed to do. Test cases are generally derived from external descriptions of the software, including specifications, requirements and design parameters. Although the tests used are primarily functional in nature, non-functional tests may also be used. The test designer selects both valid and invalid inputs and determines the correct output without any knowledge of the test object's internal structure.

### **TEST DESIGN TECHNIQUES:**

Typical black-box test design techniques include:

* Decision table testing
* All-pairs testing
* State transition tables
* Equivalence partitioning
* Boundary value analysis

**UNIT TESTING:**

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

Ideally, each test case is independent from the others. Substitutes such as method stubs, mock objects, fakes, and test harnesses can be used to assist testing a module in isolation. Unit tests are typically written and run by software developers to ensure that code meets its design and behaves as intended. Its implementation can vary from being very manual (pencil and paper)to being formalized as part of build automation.

Testing will not catch every error in the program, since it cannot evaluate every execution path in any but the most trivial programs. The same is true for unit testing. Additionally, unit testing by definition only tests the functionality of the units themselves. Therefore, it will not catch integration errors or broader system-level errors (such as functions performed across multiple units, or non-functional test areas such as performance).

Unit testing should be done in conjunction with other software testing activities, as they can only show the presence or absence of particular errors; they cannot prove a complete absence of errors. In order to guarantee correct behaviour for every execution path and every possible input, and ensure the absence of errors, other techniques are required, namely the application of formal methods to proving that a software component has no unexpected behaviour.

Software testing is a combinatorial problem. For example, every Boolean decision statement requires at least two tests: one with an outcome of "true" and one with an outcome of "false". As a result, for every line of code written, programmers often need 3 to 5 lines of test code.

 This obviously takes time and its investment may not be worth the effort. There are also many problems that cannot easily be tested at all – for example those that are nondeterministic or involve multiple threads. In addition, code for a unit test is likely to be at least as buggy as the code it is testing. Fred Brooks in The Mythical Man-Month quotes: never take two chronometers to sea*.* Always take one or three*.* Meaning, if two chronometers contradict, how do you know which one is correct?

Another challenge related to writing the unit tests is the difficulty of setting up realistic and useful tests. It is necessary to create relevant initial conditions so the part of the application being tested behaves like part of the complete system. If these initial conditions are not set correctly, the test will not be exercising the code in a realistic context, which diminishes the value and accuracy of unit test results.

To obtain the intended benefits from unit testing, rigorous discipline is needed throughout the software development process. It is essential to keep careful records not only of the tests that have been performed, but also of all changes that have been made to the source code of this or any other unit in the software. Use of a version control system is essential. If a later version of the unit fails a particular test that it had previously passed, the version-control software can provide a list of the source code changes (if any) that have been applied to the unit since that time.

**FUNCTIONAL TESTING:**

**Functional testing** is a quality assurance (QA) process and a type of black box testing that bases its test cases on the specifications of the software component under test. Functions are tested by feeding them input and examining the output, and internal program structure is rarely considered (not like in white-box testing). Functional Testing usually describes *what* the system does.

Functional testing differs from system testing in that functional testing "*verifies* a program by checking it against ... design document(s) or specification(s)", while system testing "*validate* a program by checking it against the published user or system requirements" (Kaner, Falk, Nguyen 1999, p. 52).

Functional testing typically involves five steps .The identification of functions that the software is expected to perform

1. The creation of input data based on the function's specifications
2. The determination of output based on the function's specifications
3. The execution of the test case
4. The comparison of actual and expected outputs

**PERFORMANCE TESTING:**

In software engineering, **performance testing** is in general testing performed to determine how a system performs in terms of responsiveness and stability under a particular workload. It can also serve to investigate, measure, validate or verify other quality attributes of the system, such as scalability, reliability and resource usage.

Performance testing is a subset of performance engineering, an emerging computer science practice which strives to build performance into the implementation, design and architecture of a system.

**TESTING TYPES:**

### **LOAD TESTING:**

Load testing is the simplest form of performance testing. A load test is usually conducted to understand the behaviour of the system under a specific expected load. This load can be the expected concurrent number of users on the application performing a specific number of transactions within the set duration. This test will give out the response times of all the important business critical transactions. If the database, application server, etc. are also monitored, then this simple test can itself point towards bottlenecks in the application software.

### **STRESS TESTING:**

Stress testing is normally used to understand the upper limits of capacity within the system. This kind of test is done to determine the system's robustness in terms of extreme load and helps application administrators to determine if the system will perform sufficiently if the current load goes well above the expected maximum.

### **SOAK TESTING:**

Soak testing, also known as endurance testing, is usually done to determine if the system can sustain the continuous expected load. During soak tests, memory utilization is monitored to detect potential leaks. Also important, but often overlooked is performance degradation. That is, to ensure that the throughput and/or response times after some long period of sustained activity are as good as or better than at the beginning of the test. It essentially involves applying a significant load to a system for an extended, significant period of time. The goal is to discover how the system behaves under sustained use.

### **SPIKE TESTING:**

Spike testing is done by suddenly increasing the number of or load generated by, users by a very large amount and observing the behaviour of the system. The goal is to determine whether performance will suffer, the system will fail, or it will be able to handle dramatic changes in load.

### **CONFIGURATION TESTING:**

Rather than testing for performance from the perspective of load, tests are created to determine the effects of configuration changes to the system's components on the system's performance and behaviour. A common example would be experimenting with different methods of load-balancing.

### **ISOLATION TESTING:**

Isolation testing is not unique to performance testing but involves repeating a test execution that resulted in a system problem. Often used to isolate and confirm the fault domain.

**INTEGRATION TESTING:**

**Integration testing** (sometimes called **integration and testing**, abbreviated **I&T**) is the phase in software testing in which individual software modules are combined and tested as a group. It occurs after unit testing and before validation testing. Integration testing takes as its input modules that have been unit tested, groups them in larger aggregates, applies tests defined in an integration test plan to those aggregates, and delivers as its output the integrated system ready for system testing.

**PURPOSE:**

The purpose of integration testing is to verify functional, performance, and reliability requirements placed on major design items. These "design items", i.e. assemblages (or groups of units), are exercised through their interfaces using black box testing, success and error cases being simulated via appropriate parameter and data inputs. Simulated usage of shared data areas and inter-process communication is tested and individual subsystems are exercised through their input interface.

Test cases are constructed to test whether all the components within assemblages interact correctly, for example across procedure calls or process activations, and this is done after testing individual modules, i.e. unit testing. The overall idea is a "building block" approach, in which verified assemblages are added to a verified base which is then used to support the integration testing of further assemblages.

Some different types of integration testing are big bang, top-down, and bottom-up. Other Integration Patterns are: Collaboration Integration, Backbone Integration, Layer Integration, Client/Server Integration, Distributed Services Integration and High-frequency Integration.

### **BIG BANG:**

In this approach, all or most of the developed modules are coupled together to form a complete software system or major part of the system and then used for integration testing. The Big Bang method is very effective for saving time in the integration testing process. However, if the test cases and their results are not recorded properly, the entire integration process will be more complicated and may prevent the testing team from achieving the goal of integration testing.

A type of Big Bang Integration testing is called **Usage Model testing**. Usage Model Testing can be used in both software and hardware integration testing. The basis behind this type of integration testing is to run user-like workloads in integrated user-like environments. In doing the testing in this manner, the environment is proofed, while the individual components are proofed indirectly through their use.

Usage Model testing takes an optimistic approach to testing, because it expects to have few problems with the individual components. The strategy relies heavily on the component developers to do the isolated unit testing for their product. The goal of the strategy is to avoid redoing the testing done by the developers, and instead flesh-out problems caused by the interaction of the components in the environment.

For integration testing, Usage Model testing can be more efficient and provides better test coverage than traditional focused functional integration testing. To be more efficient and accurate, care must be used in defining the user-like workloads for creating realistic scenarios in exercising the environment. This gives confidence that the integrated environment will work as expected for the target customers.

### **TOP-DOWN AND BOTTOM-UP:**

**Bottom Up Testing** is an approach to integrated testing where the lowest level components are tested first, then used to facilitate the testing of higher level components. The process is repeated until the component at the top of the hierarchy is tested.

All the bottom or low-level modules, procedures or functions are integrated and then tested. After the integration testing of lower level integrated modules, the next level of modules will be formed and can be used for integration testing. This approach is helpful only when all or most of the modules of the same development level are ready. This method also helps to determine the levels of software developed and makes it easier to report testing progress in the form of a percentage.

**Top Down Testing** is an approach to integrated testing where the top integrated modules are tested and the branch of the module is tested step by step until the end of the related module.

**Sandwich Testing** is an approach to combine top down testing with bottom up testing.

The main advantage of the Bottom-Up approach is that bugs are more easily found. With Top-Down, it is easier to find a missing branch link.

**VALIDATION:**

In that case, you can use the CustomValidator control. The CustomValidator control in Java allows you to create your own custom logic to validate user data. The CustomValidator Control can be used on the client side and server side. JavaScript is used for client validation; you can use any. Some of the popular Java validators are RequiredFieldValidator, CompareValidator, RangeValidator, RegularExpressionValidator, ValidationSummary, and CustomValidator. Server-side validation helps prevent users from bypassing validation by disabling or changing the client script. Security Note: By default, Java Web pages automatically validate that malicious users are not attempting to send script or HTML elements to your application.

**INTRODUCTION:**

This article explains validation in the Web API. Here I show the step-by-step procedure to create the application using validation.

**VALIDATION:**

How validation works is like testing to determine that what the user entered in the field is valid. After validating the entered field, it checks that it is in the correct format, specified length, and you can compare the input value in various fields or against values. We can use the validation for various types of information. That will be explained by the sample application.

**TYPES OF VALIDATION:**

Here we define the various types of validation that we can be use in our application. These are as follows:

1. **Required entry:**It ensures the required field. The user cannot skip the entry.
2. **Compare Value:**It is ensures that the comparison of the user's entry with the constant value or against the value of another constant or a specific data type. We use the comparison operator like equal, greater than, less than.
3. **Range checking:**It checks the range of the input values with the minimum and maximum range that is required for the input value. We can use the range checker with the pairs of numbers, dates and alphabetic characters.
4. **Pattern matching:**It is used for checking the pattern of an input value that specifies the sequence of characters.
5. **Remote:**It is used for checking whether the value exists on the server side.

**CHAPTER 7**

**SYSTEM MAINTENANCE:**

Software implementation Java is a new computing platform that simplifies application development in the highly distributed environment of the Internet. Computing relies on sharing of resources to achieve coherence and economies of scale, similar to a utility over a network.

This section describes the five software implementation process as:

1. The implementation processes contains software preparation and transition activities, such as the conception and creation of the implementation plan, the preparation for handling problems identified during development, and the follow-up medical records management,

2. The problem and modification analysis process, which is executed once the applications has become the responsibility of the implementation group.

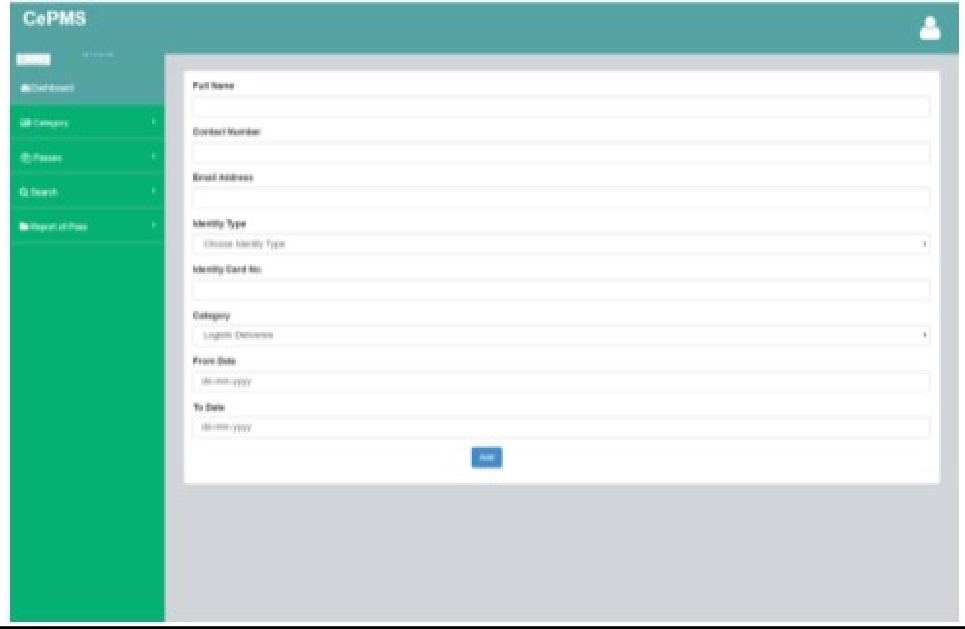
3. The process considering the implementation of the modification itself.

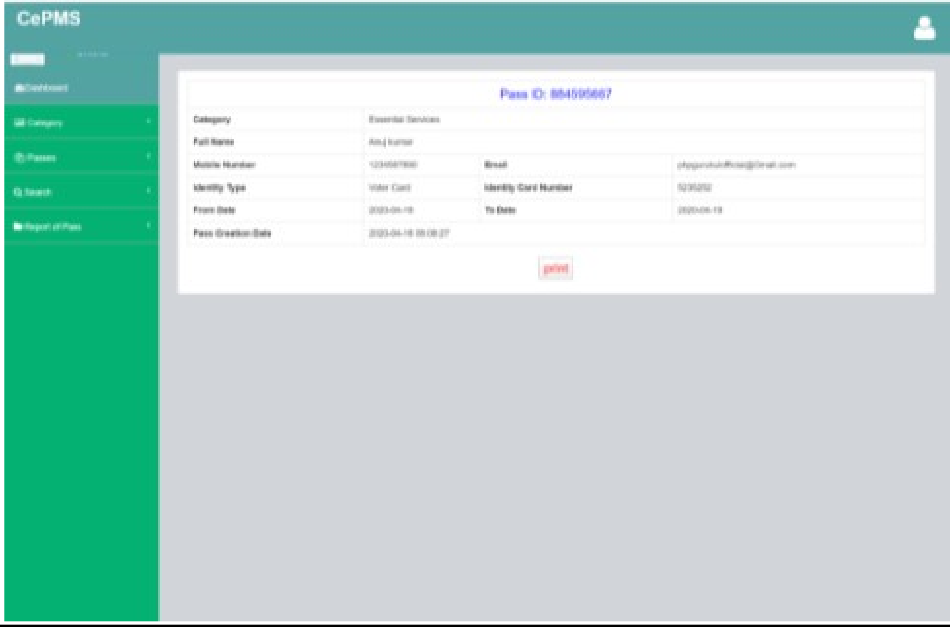
4. The process acceptance of the modification, by confirming the modified work with the individual who submitted the request in order to make sure the modification provided a solution.

5. Finally, the last implementation process, also an event which does not occur on daily basis, is the retirement of a piece of software.

**CHAPTER 8**

 **SNAPSHOT**





**CHAPTER 9**

**CONCLUSION AND FUTURE ENHANCEMENT**

E-PASS SYSTEM throughout CURFEW has several powerful options and is actually over a "simple" schematisation tool. With its support of MDA (Model Driven Architecture), it's a lot of aimed toward folks exploitation UML in associate intensive manner and with some code generations objectives than for merely drawing diagrams to document necessities. However, exploitation E-PASS SYSTEM throughout CURFEW even as a schematisation tool work fine, particularly on Windows because the tool is constructed with city and may execute quicker than the Java-based tools. nowadays is laptop world wherever the items need to be done promptly that needs best resources and best ways. thanks to this inevitable demand, cybernation of every and each sector within the main stream is should, in order that it will be command itself within the race. Few eye catching options of our project square measure its simplicity, accuracy, and its user friendly interface. Our software system incorporates all the options and facilities provided by the Visual Studio software system. This project has been developed to manage the whole operating of the Curfew Pass body. Our software system simplifies and replaces all the manual effort and therefore the paper works done by the executive to a totally electronically surroundings. thus each the user and therefore the body square measure at their ease. The user is serviceable at his footsteps whereby he simply varieties within the request and he's simply a click away. we'd wish to convey our sincere feeling and due to all, UN agency stood as our backbone, in coming up with, information and serving to North American nation in capital punishment this project with success.

**FUTURE ENHANCEMENT:**

Curfew E-Pass is a project that meets the needs of users for their work. This system has made an effort to create a solid foundation for projects and to explain how they relate to the region in which users desire to work. It includes a list of the project's goals and objectives, as well as the purpose, scope, and applicability, which is extremely useful for users to improve their work. It provides an understanding of the issue domain as well as alist of actions that the user can do.

**CHAPTER 10**

**BIBLIOGRAPHY**

**REFERENCES:**

[1]Https://Www.Irjet.Net/Archives/V8/I4/IRJET-V8I4787.Pdf Http://Www.Ijaresm.Com/Uploaded\_Files/Document\_File/Shobhana\_Avinash\_Gaikwad,\_Prerna\_Anant a\_Patil,\_Priyanka\_Dinkar\_Chougule,\_Alfiya\_Maksud\_Holsekarpawp.Pdf

[2] Https://Www.Irjet.Net/Archives/V5/I3/IRJET-V5I3865.Pdf

[3] Honeywell International Inc. “US7362210 B2” By- Rida M. Hamza, David W. Meyers “System For Gate Access Control”

[4] Oki Electric Industry Co. Ltd. “US6734783 B1” By Noriyoshi Anbai “Gate Management System”

[5] Https://Ijcsmc.Com/Docs/Papers/February2017/V6I2201738.Pdf

[6] Hitachi International Ltd. “US7755480 B2” By- Shoji Suzuki, Hiroshi Hanyu, Fumio Enmei “Smart Security System”

[7] Korean Society Of Computer Information By- Kwon, Kihyeon; Lee Hyung-Bong “Gate Management System By Face Recognition Using Smart Phone”

**WEB REFERENCES:**

https://www.javatpoint.com/

https://www.w3schools.com/

https://stackoverflow.com/

https://www.tutorialspoint.com/