**TenderOnline**

**A Project Report**

Submitted in partial fulfillment of the

Requirements for the award of the Degree of

**BACHELOR OF SCIENCE (INFORMATION TECHNOLOGY)**

**BY**

**ADITYA GUDDAD & SNEHIT MHATRE**

Seat Number -

**Under the esteemed guidance of**

**Prof Prajakta Patil**



**DEPARTMENT OF INFORMATION TECHNOLOGY**

**F. G. NAIK COLLEGE OF ARTS, COMMERCE**

**AND**

**SCIENCE (IT)**

***(Affiliated to University of Mumbai)***

**NAVI MUMBAI, 400709**

**MAHARASHTRA**

**2021-2022**

**PERFORMA FOR THE APPROVAL PROJECT PROPOSAL**

***(Note: All entries of the Performa of approval should be filled up with appropriate and complete information. Incomplete Performa of approval in any respect will be summarily rejected.)***

PNR No : - Roll No **:-** 16 & 19

1. Name of the Student: -

Guddad Aditya Shivling & Mhatre Snehit Ramesh

2. Title of the Project: -

Tender Online

3. Name of the Guide: -

Prof. Mrs. Prajakta Patil

4. Teaching experience of the Guide \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

5. Is this your first submission? Yes No

Signature of the Student Signature of the Guide

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

Signature of the Coordinator

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

**F. G. NAIK COLLEGE OF ARTS, COMMERCE AND SCIENCE (IT)**

***(Affiliated to University of Mumbai)***

**NAVI MUMBAI- MAHARASHTRA-400709**

**DEPARTMENT OF INFORMATION TECHNOLOGY**

****

**CERTIFICATE**

This is to certify that the project entitled, **"COURIER COVERAGE”, is** bonafied work of **GUPTA CHANCHAL SUSHILKUMAR & GHALME PRANALI BHASKAR** bearing seat number **– 4024025 & 4024016** Submitted in partial fulfillment of the requirements for the award of degree of BACHELOR OF SCIENCE in INFORMATION TECHNOLOGY from University of Mumbai.

**Internal Guide Coordinator**

**External Examiner**

**Date: College Seal**

**ABSTRACT**

This project is developed with an aim to provide an improved Information and Management for Tender Services.

The Software provides help in effective management of customers ,annual Tender plans opted by the customers ,service plans ,accessory details ,staff details ,etc.

The special features are the generation of tender bill ,Search criteria and the generation of Reports etc.

The following documentation provides an insight of existing system ,its limitations ,its scope of improvement and logic with its intricate details to achieve those improvements.

**ACKNOWLEDGEMENT**

The success and final outcome of this project required a lot of guidance and assistance from many people and I am extremely privileged to have got this all along the completion of my project .All that I have done is only due to such supervision and assistance and I would not forget to think them.

I Respect and thank to our **Principal MR.PRATAP MAHADK**. For providing me an opportunity to do the project work in F.G.NAIK COLLEGE and giving us all the support and guidance ,which made me complete the project duty ,I am extremely thankful to her for providing such a nice support and guidance ,although she had busy schedule managing the corporate affairs.

I owe my deep gratitude to our project guide **PROF. Mrs .PRAJAKTA PATIL & PROF . Mrs .SMRITIGANDHA BIDKAR** ,who took keen interest on our project work and guided us all along ,till the completion of our project work by providing all the necessary information for developing a good system.

I am thankful to and fortunate enough to get constant encouragement .support and guidance from all Teaching staffs of Technology department which helped us in successfully completing our project work .Also ,I would like to extend our sincere esteems to all staff in laboratory for their timely support.

**DECLARATION**

I hereby declare that the project entitled , “TENDERONLINE” done at F.G.NAIK COLLEGE OF ARTS,COMMERCE,SCIENCE(IT), has not been in any case duplicated to submit to any other university for the award of any Degree .To the best of my knowledge other than me ,no one has submitted to any other university.

The project is done in done in partial fulfillment of the requirements for the award of degree of BACHLOR OF SCIENCE (INFORMATION TECHNOLOGY) to be submitted as final semester project as part of our curriculum.

Aditya Shivling Guddad &

Snehit Mhatre

**Name and signature of student**

**TABLE OF CONTENTS**

**Chapter 1:Introduction**

**1.1 About Project**

**1.2 Objectives**

**1.3 Scope Of the System &Purpose Of the System.**

**Chapter 2:Survey Of Technologies**

**2.1 Technology Used**

**2.2 System Requirements**

**Chapter 3:Requirement And Analysis**

**3.1 Limitations Of The System**

**3.2 Software Requirement Analysis**

**3.3 Requirement Specification**

**3.4 Planning And Scheduling**

**3.5 Software And Hardware Requirements**

**3.6 Conceptual Model**

**Chapter 4:System Design**

**4.1 Module Division**

**4.2 Data Dictionary**

**4.3 ER Diagram**

**4.4 DFD/UML Diagrams**

**CHAPTER 1: INTRODUCTION**

**1.1: ABOUT PROJECT**

The Online tender handling system is a web based project .The main purpose of this application will manage Tenders through online .Tenders will be available for customers .Daily many tenders will be released and the new tenders updated so that the customers can view them and if they interested they can download the tender form .customers will have to register themselves and will have to register themselves and will get a permanent user id and password ,by using id and password he can download the tender forms in future also .All the transactions will be made through credit card.

If he is an already registered user he has to give out the user id and password and if he gives wrong user id or password he cannot enter into the site .After the user logged in he can view all the available tenders on the site .On clicking a particular tender he can view that tender details .The administrator software stores the tender details .whenever new tenders are released the administrator software will upload that tender and make it available to the customers.

After viewing the tender details if the customer is interested in coding to that tender he has to initially give out his credit card details .The card details given by the customer will be stored in the database .The card details given by him will be verified and then only he can download the tender application form.

**1.2:** **OBJECTIVES**

**The Objectives of this project are as follows:**

1. To make available the tender documents online to the customers and download the tender application form.

2. To give full online handling system for the tender documents.

3. To provide registration and feedback of tenders for the organization and customers.

**1.3: SCOPE OF THE SYSTEM:**

This website is mainly designed for

1. The industrial area in which daily many tenders are released.

2. It will be useful for the organization who has the tenders and customers from different region.

3. In this system there are registration process for customers,

Administration, tender issuer, authentication processes.

**PURPOSE OF THE PROJECT:**

The basic purpose of this system is to make available the customers and download the tender application form. Daily many tenders will be released.

**CHAPTER 2: SURVEY OF TECHNOLOGIES**

**2.1:** **TECHNOLOGY USED:**

**Front-End:**

**1. HTML:**

**HTML** or **Hypertext Mark-up Language** is the standard mark-up language used to create web pages.

HTML is written in the form of HTML elements consisting of *tags* enclosed in angle brackets (like <html>). HTML tags most commonly come in pairs like <h1> and </h1>, although some tags represent *empty elements* and so are unpaired, for example <imp>. The first tag in a pair is the *start tag*, and the second tag is the *end tag* (they are also called *opening tags* and *closing tags*). Though not always necessary, it is best practice to append a slash to tags which are not paired with a closing tag.

The purpose of a web browser is to read HTML documents and compose them into visible or audible web pages. The browser does not display the HTML tags, but uses the tags to interpret the content of the page. HTML describes the structure of a website semantically along with cues for presentation, making it a mark-up language rather than a programming language.

HTML elements form the building blocks of all websites. HTML allows images and objects to be embedded and can be used to create interactive forms. It provides a means to create structured documents by denoting structural semantics for text such as headings, paragraphs, lists, links, quotes and other items. It can embed scripts written in languages such as JavaScript which affect the behaviour of HTML web pages.

**2. CASCADING STYLE SHEETS (CSS):**

It is a style sheet language used for describing the look and formatting of a document written in a mark-up language. While most often used to style web pages and interfaces written in HTML and XHTML, the language can be applied to any kind of XML document, including plain XML, SVG and XUL. CSS is a cornerstone specification of the web and almost all web pages use CSS style sheets to describe their presentation.

CSS is designed primarily to enable the separation of document content from document presentation, including elements such as the layout, colours, and fonts. [1] This separation can improve content accessibility, provide more flexibility and control in the specification of presentation characteristics, enable multiple pages to share formatting, and reduce complexity and repetition in the structural content.

CSS can also allow the same mark-up page to be presented in different styles for different rendering methods, such as on-screen, in print, by voice (when read out by a speech-based browser or screen reader) and on Braille-based, tactile devices. It can also be used to allow the web page to display differently depending on the screen size or device on which it is being viewed. While the author of a document typically links that document to a CSS file, readers can use a different style sheet, perhaps one on their own computer, to override the one the author has specified. However if the author or the reader did not link the document to a specific style sheet the default style of the browser will be applied.

**3. JAVASCRIPT:**

JavaScript is the scripting language of the Web. All modern HTML pages are using JavaScript. A scripting language is a lightweight programming language .JavaScript code can be inserted into any HTML page, and it can be executed by all types of web browsers. JavaScript is easy to learn.

**WHY TO USE JAVASCRIPT:**

JavaScript is one of the 3 languages all web developers must learn:

1. HTML to define the content of web pages

2. CSS to specify the layout of web pages

3. JavaScript to specify the behaviour of web pages

Example

x =document.getElementById("demo"); //Find the HTML element with id="demo" x.innerHTML = "Hello JavaScript"; //Change the content of the HTML element

**document. getElementById()** is one of the most commonly used HTML DOM methods.

**OTHER USES OF JAVASCRIPT**:

* Delete HTML elements
* Create new HTML elements
* Copy HTML elements
* In HTML, JavaScript is a sequence of statements that can be executed by the web browser.

**JAVASCRIPT STATEMENTS :**

* JavaScript statements are "commands" to the browser.
* The purpose of the statements is to tell the browser what to do.
* This JavaScript statement tells the browser to write "Hello Dolly" inside an HTML element with id="demo”: Semicolon;
* Semicolon separates JavaScript statements.
* Normally you add a semicolon at the end of each executable statement.
* Using semicolons also makes it possible to write many statements on one line.

**JAVASCRIPT CODE:**

* JavaScript code (or just JavaScript) is a sequence of JavaScript statements.
* Each statement is executed by the browser in the sequence they are written.
* This example will manipulate two HTML elements:

• Example

* document.getElementById("demo").innerHTML="HelloDolly"; document.getElementById("myDIV").innerHTML="How are you?";

**JAVASCRIPT PROPERTIES:**

* Properties are the values associated with a JavaScript object.
* A JavaScript object is a collection of unordered properties.
* Properties can usually be changed, added, and deleted, but some are read only.

**DOT.NET:**

.NET is framework an API for programming on the windows platform. Along the .NET ,VB is a language that has been designed from scratch to work with .NET as well as to take advantage of all the progress in developer environments and in our understanding of objects oriented programming principles that have taken over the past 20 years.

We Should make it clear that backward compatibility has not been lost in the process .Existing programs will continue to work ,and .NET was designed with the ability to work with existing software. Presently ,communication between software components on windows almost entirely takes place using COM. Taking account of this;.NET does have the ability to provide wrappers around existing COM components.

**Advantages of .NET:**

**1)Object-Oriented Programming: -**

Both the .NET framework and VB are entirely based on object-oriented principles right from the start.

**2)Good Design: -**

A base class library which is designed from the ground up in a highly intuitive way.

**3) Language Independence :-**

With .NET all the languages visual basic .NET ,C#,J# and managed C++ compiled to a common intermediate Language.This means that languages are interoperable in a way that has not been seen before.

**4) Better Support for Dynamic web pages:**

While ASP offered a lot of flexibility ,it was also inefficient because if its use of interpreted scripting language,and the lack of object oriented design resulted in messy ASP code .NET offers an integrated support for web pages.

**5) Efficient Data Access: -**

A set of NET components, collectively known as ADO.NET, provides access to relational databases and a variety of data sources.

**6) Code Sharing: -**

NET has completely revamped the way that code is shared between applications, introducing the concept of the assembly, which replaces the traditional DLL. Assemblies have formal facilities for versioning and different versions of assemblies can exist side by side.

**7) Improved Security: -**

Each assembly can also contain built-in security information.

**8) Zero Impact Installation: -**

There are two types of assembly shared and private. Shared assemblies are common libraries available to all software. Private assemblies are intended only for use with particular software.

**9) Support for Web Services: -**

NET has fully integrated support for developing Web services as easily as you'd develop any other type of application.

**10) Visual Studio 2008:**

.NET comes with a developer environment, Visual Studio 2008, which can cope equally well with C++, C#, J#, and Visual Basic 2008 as well as with ASP NET code. Visual Studio 2008 integrates all the best features of the respective language specific environments of visual Studio.NET 2005 and Visual Studio 6.

**VB.NET:**

**Features:**

1) Most suitable building robust, n-tier client server applications

2] VB with ADO.NET gives quick access to data stores like SQL server and oracle.

3) It offers XML web services and server side controls.

4] Makes it easy writing windows forms user Interface

5] Component conflicts become infrequent.

6) Deployment is easier.

**ADO.NET:**

ADO.NET is a large set of .NET classes that enable us to retrieve and manipulate data and update data. As an Integral part of the NET framework, it shares many of its features:

Features such as multi-language support, garbage collection, just-in-time compilation, object-oriented design and dynamic caching, and is far more than an upgrade of previous versions of ADO

ADO.NET is set to become a core component of any data-driven NET application or Web Service, and understanding its power will be essential to anyone wishing to utilize NET data support to maximum effect.

**BACK END:**

**MySQL:**

MySQL is developed, distributed, and supported by Oracle Corporation. MySQL is a database system used on the web it runs on a server. MySQL is ideal for both small and large applications. It is very fast, reliable, and easy to use. It supports standard SQL. MySQL can be compiled on a number of platforms.

The data in MySQL is stored in tables. A table is a collection of related data, and it consists of columns and rows. Databases are useful when storing information categorically.

**FEATURES OF MySQL:**

Internals and portability:

* Written in C and C++.
* Tested with a broad range of different compilers.
* Works on many different platforms.
* Tested with Purify (a commercial memory leakage detector) as well as with Val grind, a GPL tool.
* Uses multi-layered server design with independent modules.

**Security:**

* A privilege and password system that is very flexible and secure, and that enables host-based verification.
* Password security by encryption of all password traffic when you connect to a server.

Scalability and Limits:

* Support for large databases. We use MySQL Server with databases that contain 50 million records. We also know of users who use MySQL Server with 200,000 tables and about 5,000,000,000 rows.
* Support for up to 64 indexes per table (32 before MySQL 4.1.2). Each index may consist of 1 to 16 columns or parts of columns. The maximum index width is 767 bytes for **InnoDB** tables, or 1000 for **MyISAM** ; before MySQL 4.1.2, the limit is 500 bytes. An index may use a prefix of a column for **CHAR**, **VARCHAR**, **BLOB**, or **TEXT** column types

**CONNECTIVITY:**

Clients can connect to MySQL Server using several protocols:

* Clients can connect using TCP/IP sockets on any platform.
* On Windows systems in the NT family (NT, 2000, XP, 2003, or Vista), clients can connect using named pipes if the server is started with the --enable-named-pipe option. In MySQL 4.1 and higher, Windows servers also support shared-memory connections if started with the --shared-memory option. Clients can connect through shared memory by using the --protocol=memory option.
* On UNIX systems, clients can connect using Unix domain socket files.

**LOCALIZATION:**

* The server can provide error messages to clients in many languages.
* All data is saved in the chosen character set.

**CLIENTS AND TOOLS:**

• MySQL includes several client and utility programs. These include both command-line programs such as **mysqldump** and **mysqladmin**, and graphical programs such as MySQL Workbench.

* • MySQL Server has built-in support for SQL statements to check, optimize, and repair tables. These statements are available from the command line through the **mysqlcheck** client. MySQL also includes **myisamchk** , a very fast command-line utility for performing these operations on **MyISAM** tables.
* • MySQL programs can be invoked with the --help or -? option to obtain online assistance.

**WHY TO USE MySQL:**

• Leading open source RDBMS

• Ease of use – No frills

• Fast

• Robust

• Security

• Multiple OS support

• Free

• Technical support

• Support large database– up to 50 million rows, file size limit up to 8 Million TB

**2.2 SYSTEM REQUIRMENT :**

**Hardware and Software Requirements :**

In the development of the system, a major element is the selection of the compatible hardware and software. Here the system is built in client-server architecture. So that user can use ordinary PC as client to access the system. As per developers concern, the developer requires Pentium IV with installed n/w card.

VB.net gives most powerful API, which enables the developer to build the system, which

fulfills the above requirements. As per Back-End

**System Requirement Specifications :**

**Hardware Requirements:**

. PIV 2.8 GHz Processor and Above

. RAM 512MB and Above

HDD 40 GB Hard Disk Space and Above

**Software Requirements:**

* WINDOWS OS (XP/7/8.1/10)
* Visual Studio Net 2019 Enterprise Edition
* Internet information Server 5.0 (IIS)
* Visual Studio Net Framework (Minimal for Deployment) version 4.2.1
* SQL Server 2005 Enterprise Edition

**CHAPTER 3: REQUIREMENT & ANALYSIS**

**3.1 LIMITATIONS OF SYSTEM :**

The development of this new system contains the following activities, which try to automate the entire process keeping in the view of database integration approach. User Friendliness is provided in the application with various controls provided by system Rich User Interface. The system makes the overall project management much easier and flexible. It can be accessed over the Intranet. Various classes have been used for file uploading and down loading. The user information files can be stored in centralized database which can be maintained by the system. This can give the good security for user information because data is not in client machine. Authentication is provided for this application only registered users can access. User can share is data to others, and also he can get data from others. There is no risk of data management at any level while the project development is under process. Report generation features is provided using Data reports to generate different kind of reports.

**3.1.1 : EXISTING SYSTEM :**

Generally tender notifications are released through newspapers and the customers read them. If the customer has to code to a particular tender he has take a DD or Cheque in favor of the company name and then post it to the company so that they can get a tender from through which they can quote the tender. This involves a tedious and time-consuming manual procedure. This process may take no. Of days time delay occurs.

**3.2 : PROPOSED SYSTEM :**

Full on-line tender handling system with online tender documents helps the customer to reduce his manual work. He can just sit in front of a computer and following the instructions he can download the tender forms. The goal of the work is to bring down the work of the customer with the increased efficiency and to speed up his activities. Availability of Detailed Tender Documents. An integrated normalized relational database is maintained. Pre defined queries for generation of any specific enquiry purposes.

**3.1.3: PLATFORMS :**

**1.Google Chrome** is a [cross-platform](https://en.wikipedia.org/wiki/Cross-platform) [web browser](https://en.wikipedia.org/wiki/Web_browser) developed by [Google](https://en.wikipedia.org/wiki/Google). It was first released in 2008 for [Microsoft Windows](https://en.wikipedia.org/wiki/Microsoft_Windows), built with [free software](https://en.wikipedia.org/wiki/Free_software) components from [Apple WebKit](https://en.wikipedia.org/wiki/Apple_WebKit) and [Mozilla Firefox](https://en.wikipedia.org/wiki/Mozilla_Firefox).[[13]](https://en.wikipedia.org/wiki/Google_Chrome#cite_note-14) It was later [ported](https://en.wikipedia.org/wiki/Ported) to [Linux](https://en.wikipedia.org/wiki/Linux), [macOS](https://en.wikipedia.org/wiki/MacOS), [iOS](https://en.wikipedia.org/wiki/IOS), and [Android](https://en.wikipedia.org/wiki/Android_(operating_system)), where it is the default browser.[[14]](https://en.wikipedia.org/wiki/Google_Chrome#cite_note-15) The browser is also the main component of [Chrome OS](https://en.wikipedia.org/wiki/Chrome_OS), where it serves as the platform for [web applications](https://en.wikipedia.org/wiki/Web_application).

**2. Microsoft Edge** is a [cross-platform](https://en.wikipedia.org/wiki/Cross-platform_software) [web browser](https://en.wikipedia.org/wiki/Web_browser) created and developed by [Microsoft](https://en.wikipedia.org/wiki/Microsoft). It was first released for [Windows 10](https://en.wikipedia.org/wiki/Windows_10) and [Xbox One](https://en.wikipedia.org/wiki/Xbox_One) in 2015, for [Android](https://en.wikipedia.org/wiki/Android_(operating_system)) and [iOS](https://en.wikipedia.org/wiki/IOS) in 2017,[[4]](https://en.wikipedia.org/wiki/Microsoft_Edge#cite_note-5)[[5]](https://en.wikipedia.org/wiki/Microsoft_Edge#cite_note-6) for [macOS](https://en.wikipedia.org/wiki/MacOS) in 2019,[[6]](https://en.wikipedia.org/wiki/Microsoft_Edge#cite_note-7) and as a preview for [Linux](https://en.wikipedia.org/wiki/Linux) in October 2020,[[7]](https://en.wikipedia.org/wiki/Microsoft_Edge#cite_note-Linux_Preview_Announced-8)[[8]](https://en.wikipedia.org/wiki/Microsoft_Edge#cite_note-Linux_Preview_Released-9) and can replace [Internet Explorer](https://en.wikipedia.org/wiki/Internet_Explorer) on [Windows 7](https://en.wikipedia.org/wiki/Windows_7), [Windows 8](https://en.wikipedia.org/wiki/Windows_8), [Windows 8.1](https://en.wikipedia.org/wiki/Windows_8.1), [Windows Server 2008 R2](https://en.wikipedia.org/wiki/Windows_Server_2008_R2), [Windows Server 2012](https://en.wikipedia.org/wiki/Windows_Server_2012), [Windows Server 2012 R2](https://en.wikipedia.org/wiki/Windows_Server_2012_R2), [Windows Server 2016](https://en.wikipedia.org/wiki/Windows_Server_2016) and [Windows Server 2019](https://en.wikipedia.org/wiki/Windows_Server_2019) but unlike IE, this browser does not support [Windows Vista](https://en.wikipedia.org/wiki/Windows_Vista) or an earlier version.

**3. Mozilla Firefox** or simply **Firefox**, is a [free and open-source](https://en.wikipedia.org/wiki/Free_and_open-source_software)[[20]](https://en.wikipedia.org/wiki/Firefox#cite_note-LWNtrademark-20) [web browser](https://en.wikipedia.org/wiki/Web_browser) developed by the [Mozilla Foundation](https://en.wikipedia.org/wiki/Mozilla_Foundation) and its subsidiary, the [Mozilla Corporation](https://en.wikipedia.org/wiki/Mozilla_Corporation). Firefox uses the [Gecko](https://en.wikipedia.org/wiki/Gecko_(software)) [rendering engine](https://en.wikipedia.org/wiki/Browser_engine) to display web pages, which implements current and anticipated web standards.[[21]](https://en.wikipedia.org/wiki/Firefox#cite_note-21) In 2017, Firefox began incorporating new technology under the code name [Quantum](https://en.wikipedia.org/wiki/Quantum_(Mozilla)) to promote [parallelism](https://en.wikipedia.org/wiki/Parallel_computing) and a more intuitive [user interface](https://en.wikipedia.org/wiki/User_interface).[[22]](https://en.wikipedia.org/wiki/Firefox#cite_note-22) Firefox is available for [Windows 7](https://en.wikipedia.org/wiki/Windows_7) and later versions, [macOS](https://en.wikipedia.org/wiki/MacOS), and [Linux](https://en.wikipedia.org/wiki/Linux). [Its unofficial ports](https://en.wikipedia.org/wiki/Firefox#Unofficial_ports) are available for various [Unix](https://en.wikipedia.org/wiki/Unix) and [Unix-like](https://en.wikipedia.org/wiki/Unix-like) operating systems, including [FreeBSD](https://en.wikipedia.org/wiki/FreeBSD),[[9]](https://en.wikipedia.org/wiki/Firefox#cite_note-FreeBSD_ports-9) [OpenBSD](https://en.wikipedia.org/wiki/OpenBSD),[[10]](https://en.wikipedia.org/wiki/Firefox#cite_note-OpenBSD_ports-10) [NetBSD](https://en.wikipedia.org/wiki/NetBSD),[[11]](https://en.wikipedia.org/wiki/Firefox#cite_note-NetBSD_pkgsrc-11) [illumos](https://en.wikipedia.org/wiki/Illumos),[[12]](https://en.wikipedia.org/wiki/Firefox#cite_note-OpenIndiana_Wiki-12) and [Solaris Unix](https://en.wikipedia.org/wiki/Solaris_Unix).[[14]](https://en.wikipedia.org/wiki/Firefox#cite_note-Open_Source_software_in_Solaris,_Github-14) Firefox is also available for [Android](https://en.wikipedia.org/wiki/Firefox_for_Android) and [iOS](https://en.wikipedia.org/wiki/Firefox_for_iOS). However, the iOS version uses the [WebKit](https://en.wikipedia.org/wiki/WebKit) layout engine instead of Gecko due to platform requirements, as with all other iOS web browsers. An optimized version of Firefox is also available on the [Amazon Fire TV](https://en.wikipedia.org/wiki/Amazon_Fire_TV), as one of the two main browsers available with [Amazon's Silk Browser](https://en.wikipedia.org/wiki/Amazon_Silk).

**3.2 Software Requirement Analysis:**

**3.2.1 Role of SRS**

The purpose of the software requirement specification is to establish a learning platform for the users in order to be aware of the basic words being used in our daily lives. Software Requirement Specification is the medium through which the user needs are accurately specified. It forms the basis of software development.

**3.3: Requirement Specification**

**3.3.1 : Feasibility Study :**

• Feasibility study is totally depending upon the preliminary investigation & requirements of the system. Hence we have to determine the system requested is feasible or not. This helps us to check Technical, Economical & Operational feasibility of requested system against the current system. The data collection done at preliminary stage examines that the system, which we are developing, will be beneficial to understand the pronunciation of words in a better fashion.

• The project helps to understand and learn pronunciations of basic words for new English learner and kids. Categorization of these words also helps to understand usage of a particular word and what context these words can be used. The app also will be helpful to learn different accent of English for travellers.

**3.3.2 :Technical Feasibility**

* The technical issue usually raised during the feasibility stage of the investigation includes the following:
* Does the necessary technology exist to do what is suggested?
* Do the proposed equipments have the technical capacity to hold the data required to use
* the new system?
* Will the proposed system provide adequate response to inquiries, regardless of the number or location of users?
* Can the system be upgraded if developed?
* Are there technical guarantees of accuracy, reliability, ease of access and data security?

Earlier no system existed to cater to the needs of 'Secure Infrastructure Implementation System. The current system developed is technically feasible. It is a web based user interface for audit workflow at NIC CSD. Thus it provides an easy access to the users. The database's purpose is to create, establish and maintain a workflow among various entities in order to facilitate all concerned users in their various capacities or roles. Permission to the users would be granted based on the roles specified. Therefore, it provides the technical guarantee of accuracy, reliability and security. The software and hard requirements for the development of this project are not many and are already available in-house at NIC or are available as free as open source. The work for the project is done with the current equipment and existing software technology Necessary bandwidth exists for providing a fast feedback to the users irrespective of the number of users using the system.

**3.3.3 Operational Feasibility**

Proposed projects are beneficial only if they can be turned out into information system. That will meet the organization's operating requirements. Operational feasibility aspects of the project are to be taken as an important part of the project implementation. Some of the important issues raised are to test the operational feasibility of a project includes the following:

* Is there sufficient support for the management from the users?
* Will the system be used and work properly if it is being developed and implemented?
* Will there be any resistance from the user that will undermine the possible application benefits?

This system is targeted to be in accordance with the above-mentioned issues. Beforehand, the management issues and user requirements have been taken into consideration. So there is no question of resistance from the users that can undermine the possible application benefits.

The well-planned design would ensure the optimal utilization of the computer resources and would help in the improvement of performance status.

**3.3.4 : Economics Feasibility**

A system can be developed technically and that will be used if installed must still be a good investment for the organization. In the economical feasibility, the development cost in creating the system is evaluated against the ultimate benefit derived from the new systems. Financial benefits must equal or exceed the costs.

The system is economically feasible. It does not require any addition hardware or software. Since the interface for this system is developed using the existing resources and technologies available at NIC, There is nominal expenditure and economical feasibility for certain.

**PLANNING AND SCHEDULING :**

**Gantt Chart:**

A Gantt chart could be a kind of bar graph that illustrates a project schedule. Gantt chart illustrate the beginning and end dates of the terminal parts and outline parts of a project.

A Gantt chart, or harmonogram, is a type of bar chart that illustrates a project schedule. This chart lists the tasks to be performed on the vertical axis, and time intervals on the horizontal axis. The width of the horizontal bars in the graph shows the duration of each activity. Gantt charts illustrate the start and finish dates of the terminal elements and summary elements of a project. Terminal elements and summary elements constitute the work breakdown structure of the project. Modern Gantt charts also show the dependency (i.e., precedence network) relationships between activities. Gantt charts can be used to show current schedule status using percent-complete shadings and a vertical "TODAY" line as shown here.

Gantt charts are sometimes equated with bar charts.]

Gantt charts are usually created initially using an *early start time approach*, where each task is scheduled to start immediately when its prerequisites are complete. This method maximizes the float time available for all tasks.

The following phases are covered in Gantt chart:

**1. Planning Phase**

1.1 Define the problem

1.2 Produce Project Schedule

1.3 Launch the Project

**2. Analysis Phase**

2.1 Gather Information

2.2 Define System Requirements

**3. Design Phase**

3.1 Complete Application Design

3.2 Design User Interfaces

3.3 Design & Integrate Database & System Controls4. Coding phase

4.1 Writing code for modules &integrating modules

4.2 Testing & Implementation



**PERT CHART**



A PERT chart is a project management tool that provides a graphical representation of a project's timeline. The Program Evaluation Review Technique (PERT) breaks down the individual tasks of a project for analysis. PERT charts are considered preferable to Gantt Charts because they identify task dependencies, but they're often more difficult to interpret.

**How Do PERT Charts Work?**

A PERT chart uses circles or rectangles called nodes to represent project events or milestones. These nodes are linked by vectors or lines that represent various tasks.

Dependent tasks are items that must be performed in a specific manner. For example, if an arrow is drawn from Task No. 1 to Task No. 2 on a PERT chart, Task No. 1 must be completed before work on Task No. 2 begins.

Items at the same stage of production but on different task lines within a project are referred to as parallel tasks. They're independent of each other, but they're planned to occur at the same time.

**Interpreting PERT Charts**

A PERT chart is a visual representation of a series of events that must occur within a project’s lifetime. The direction of arrows indicates the flow and sequence of events required for project completion. Dotted activity lines represent dummy activities—items that are located on another PERT path. Numbers and time allotments are assigned and shown inside each vector.

These charts have their distinct , the most important of which anticipate how long it will take to finalize a project. "Optimistic time" refers to the shortest duration, and "pessimistic time" is logically the longest it might take. The "most likely time" indicates a reasonable estimate of the best-case scenario, whereas "expected time" accounts for problems and obstacles.

**The Benefits of PERT Charts**

A PERT chart allows managers to evaluate the time and resources necessary to manage a project. This evaluation includes the ability to track required assets during any stage of production in the course of the entire project.

PERT analysis incorporates data and information from multiple departments. This combining of information encourages department responsibility, and it identifies all responsible parties across the organization. It also improves communication during the project, and it allows an organization to commit to projects that are relevant to its strategic positioning.

Finally, PERT charts are useful for what-if analyses. Understanding the possibilities concerning the flow of project resources and milestones allows management to achieve the most efficient and useful project path.

**CONCEPTUAL MODEL :**

**DFD:**

A Data multidimensional language (DFD) may be a graphical illustration of the “flow” of information through Associate in Nursing data system, modelling its method aspects. A DFD is usually used as a preliminary step to form an outline of the system while not going into nice detail, which may later be elaborate. DFDs can even be used for the image of information process.

A DFD shows what quite info are going to be input to and output from the system, however the info can advance through the system, and wherever the info are going to be keep. It doesn't show info regarding method temporal order or whether or not processes can operate in sequence or in parallel, in contrast to a conventional structured multidimensional language that focuses on management flow , or a UML activity workflow diagram, which presents both control and data flows as a unified model.

**DATA FLOW DIAGRAM**

**Context Level(0 th level DFD)**

DATABASE

USER

RESPONSE STORAGE

REQUEST RESPONSE

TENDERONLINE

WEBSITE

**Level 0 DFD Explanation:**

This is level 0 of DFD. It only contains one process node that generalizes the function of the entire system in relationship of external entities. User sent request to the application, application check the database storage, and send response to the application and that response is then visible to the user on user’s screen.

**DFD SYMBOLS:**

In the DFD, there are four symbols

1. A square defines a source(originator) or destination of system data

2. An arrow identifies data flow. It is the pipeline through which the information flows

3. A circle or a bubble represents a process that transforms incoming data flow into outgoing data flows.

4. An open rectangle is a data store, data at rest or a temporary repository of data

 Process that transforms data flow.

Source or Destination of data

 Data flow

Data Store

**CONSTRUCTING A DFD:**

**Several rules of thumb are used in drawing DFD’S:**

1. Process should be named and numbered for an easy reference. Each name should be representative of the process.

2. The direction of flow is from top to bottom and from left to right. Data traditionally flow from source to the destination although they may flow back to the source. One way to indicate this is to draw long flow line back to a source. An alternative way is to repeat the source symbol as a destination. Since it is used more than once in the DFD it is marked with a short diagonal.

3. When a process is exploded into lower level details, they are numbered.4.

4. The names of data stores and destinations are written in capital letters. Process and dataflow names have the first letter of each work capitalized A DFD typically shows the minimum contents of data store. Each data store should contain all the data elements that flow in and out. Questionnaires should contain all the data elements that flow in and out. Missing interfaces redundancies and like is then accounted for often through interviews.

**SAILENT FEATURES OF DFD’S:**

1. The DFD shows flow of data, not of control loops and decision are controlled considerations do not appear on a DFD.

2. The DFD does not indicate the time factor involved in any process whether the dataflow take place daily, weekly, monthly or yearly.

3. The sequence of events is not brought out on the DFD

**TYPES OF DATA FLOW DIAGRAMS :**

1.Current Physical

2.Current Logical

3.New Logical

4.New Physical

**CURRENT PHYSICAL:**

In Current Physical DFD proecess label include the name of people or their positions or the names of computer systems that might provide some of the overall system-processing label includes an identification of the technology used to process the data. Similarly data flows and data stores are often labels with the names of the actual physical media on which data are stored such as file folders, computer files, business forms or computer tapes.

**CURRENT LOGICAL:**

The physical aspects at the system are removed as mush as possible so that the current system is reduced to its essence to the data and the processors that transform them regardless of actual physical form.

**NEW LOGICAL:**

This is exactly like a current logical model if the user were completely happy with he user were completely happy with the functionality of the current system but had problems with how it was implemented typically through the new logical model will differ from current logical model while having additional functions, absolute function removal and inefficient flows recognized.

**NEW PHYSICAL:**

The new physical represents only the physical implementation of the new system.

**RULES GOVERNING THE DFD’S**

**PROCESS**

1) No process can have only outputs.

2) No process can have only inputs. If an object has only inputs than it must be a sink.

3) A process has a verb phrase label.

**DATA STORE:**

1) Data cannot move directly from one data store to another data store, a process must move data.

2) Data cannot move directly from an outside source to a data store, a process, which receives, must move data from the source and place the data into data store

3) A data store has a noun phrase label .

**SOURCE OR SINK:**

The origin and /or destination of data.

* Data cannot move direly from a source to sink it must be moved by a process
* A source and /or sink has a noun phrase land

1) A Data Flow has only one direction of flow between symbols. It may flow in both directions between a process and a data store to show a read before an update. The later is usually indicated however by two separate arrows since these happen at different type.

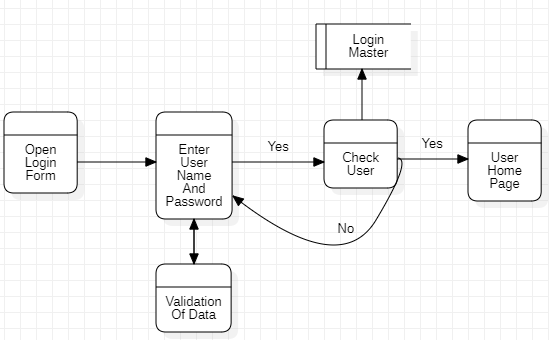
2) A join in DFD means that exactly the same data comes from any of two or more different processes data store or sink to a common location.

3) A data flow cannot go directly back to the same process it leads. There must be atleast one other process that handles the data flow produce some other data flow returns the original data into the beginning process.

4) A Data flow to a data store means update (delete or change).

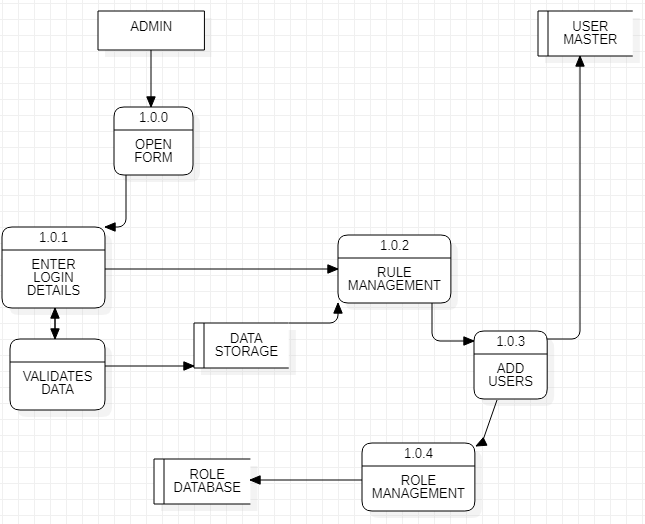
5) A data Flow from a data store means retrieve or use.

**LOGIN DFD :**

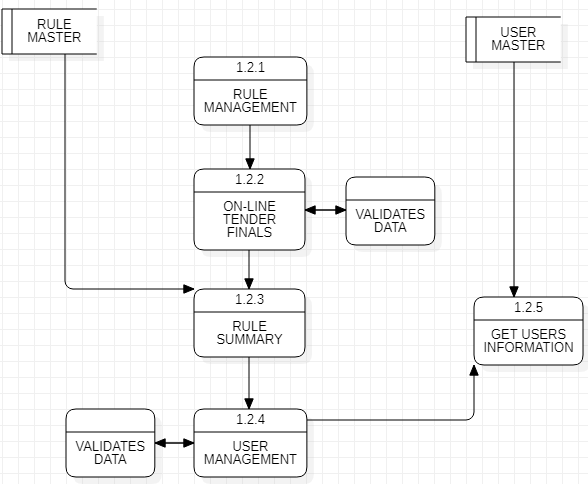


**Admin Details Data Flow**

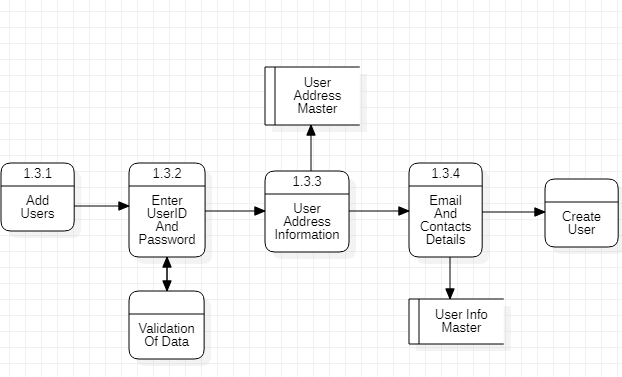
**1st level DFD**

****

**2ND level DFD(Admin)**

****

**Add User Information**

****

**ERD :**

The entity-relationship (ER) data model allows us to describe the data involved in a

real world enterprise in terms of object and their relationships and is widely used to

develop an initial database design.

The ER model is important primarily for its role in database design. It provides useful concepts that allow us to move from an informal description of what users want from their database to a more detailed and precise description that can be implemented in a DBMS. The ER model is used in a phase called ―Conceptual Database Design‖. It should be noted that many variations of ER diagrams are in use and no widely accepted standards prevail.

ER modelling is something regarded as a complete approach to design a logical database scheme. This is incorrect because the ER diagram is just an approximate description of data, constructed through a very subjective evaluation of the information collected during requirements analysis.

**Entity**

ER modelling is something regarded as a complete approach to design a logical database schema. This is incorrect because the ER diagram is just an approximate description of data, constructed through a very subjective evaluation of the information collected during requirements analysis. An entity is an object in the real world that is distinguishable from other objects. Examples include the following: The address of the manager of the institution, a Person with unique name etc.

It is often useful to identify a collection of similar entities. Such a collection is called as ―Entity set. Note that entity set need not be disjoint.

**Attributes**

An entity is described using a set of attributes. All entities in a given entity set have the same attributes; this essentially what we mean by similar. Our choice of attributed reflects the level of detail at which we wish to represent information in crisis.

For e.g. The Admission entity set would use the name, age, and qualification of the students as the attributes. In this case we will store the name, the registry no, the course enrolled of the student and not his/her address or the gender.

**Domain :**

For each attribute associated with an entity set, we must identify a domain of possible values.

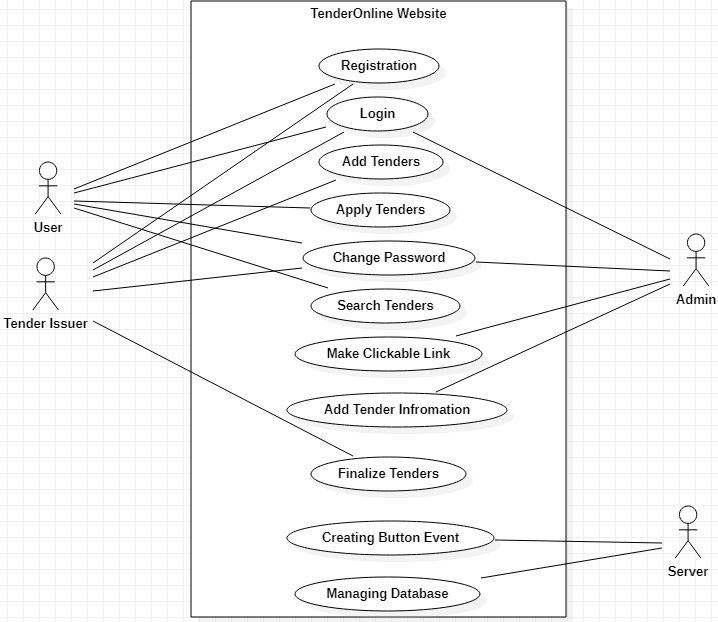
For e.g. the domain associated with the attribute name of the student might be of the set of 20-character string.

Another example would be the ranking of the students in the institute would be on the scale of 1-6, the associated domain consists of integers 1 through 6.

**Key :**

Further, for each entity set we choose a key. A key is a minimal set of attributed whose values uniquely identify an entity in the set. There could be more than one candidate; if so we designate one of them as primary key. For now we will assume that each entity set contains at least one set of attributes that uniquely identify an entity in the entity set; that is the set of attributes contains a key.

**ERD :**

****

**CHAPTER 4: SYSTEM DESIGN**

**SYSTEM DESIGN**

**4.1 :Introduction to UML:**

The Unified Modeling Language (UML) is a standard language for indicating, picturing, building, and reporting the product framework and its parts. It is a graphical language , which gives a jargon and set of semantics and standards. The UML centers around the calculated and physical portrayal of the framework. It catches the choices and understandings about frameworks that must be built. It is utilized to get, plan, arrange, keep up, and control data about the frameworks.

The UML is a language for:

• Visualizing

• Specifying

• Constructing

• Documenting

**Visualizing**

Through UML we see or picture a current framework and eventually we envision how the framework will be after usage. Except if we figure, we can't execute. UML pictures, how the parts of the framework impart and connect with one another.

**Specifying**

Indicating means building, models that are exact, unambiguous and complete UML addresses the determination of all the significant examination plan, usage choices that must be made in creating and sending a product framework.

**Constructing**

UML models can be legitimately associated with an assortment of programming language through mapping a model from UML to a programming language like JAVA or C++ or VB. Forward Engineering and Reverse Engineering is conceivable through UML.

**Documenting**

The Deliverables of a task separated from coding are a few Artifacts, which are basic in controlling, estimating and conveying about a framework during its creating necessities, engineering, want, source code, venture plans, tests, models releasers, and so forth...

**UML Approach:**

**UML Diagram**

An diagram is the graphical introduction of a lot of components, frequently rendered as an associated chart of vertices and curves you attract chart to picture a framework from alternate point of view, so an outline is a projection into a framework. For everything except most unimportant frameworks, a graph speaks to an omitted perspective on the components that make up a framework. A similar component may show up in all outlines, just a couple of charts, or in no graphs by any means. In principle, a graph may contain any blend of things and connections. Practically speaking, nonetheless, few regular mixes emerge, which are reliable with the five most valuable perspectives that involve the engineering of a product concentrated framework. Thus, the UML incorporates nine such charts:

1. Use Case Diagram.

2. Activity Diagram.

3. Deployment Diagram.

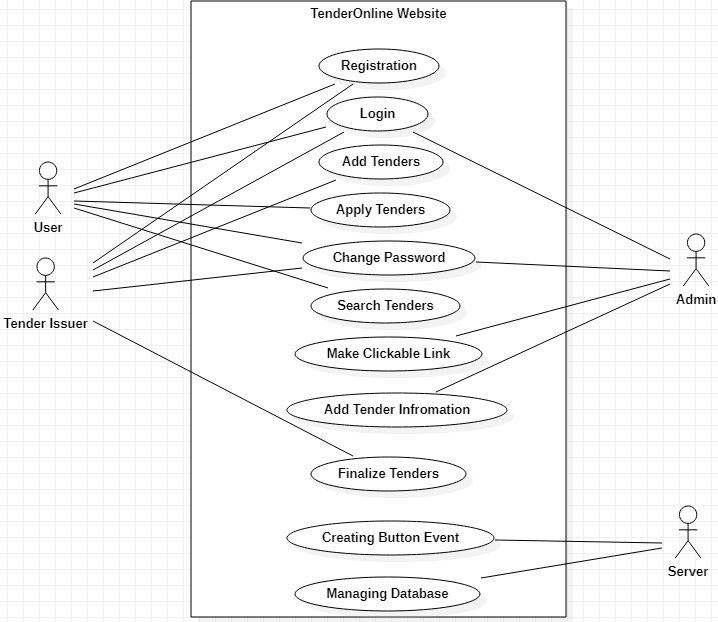
4. State Chart Diagram.

5.Flow Chart Diagram.

* **USE CASE DIAGRAM**

A Use case diagram in the Unified Modeling Language (UML) is a kind of creating by using something called use case analysis which gives out behavioral feel of the system. The outcome is a graphical overview of all the functionality of the project which is depicted in the overall project. This graphical overview incorporates models such as actors and cases (functions) and their co-operation. We have used the UML language for the creation of all the diagrams and using the software called Star UML.

**Use case Diagram of our project:**

****

* **Activity Diagram:**

**Activity diagrams** are graphical representations of workflows of stepwise activities and actions with support for choice, iteration and concurrency. In the Unified Modeling Language, activity diagrams are intended to model both computational and organizational processes (i.e., workflows), as well as the data flows intersecting with the related activities. Although activity diagrams primarily show the overall flow of control, they can also include elements showing the flow of data between activities through one or more data stores.

Activity diagram is another important behavioral diagram in UML diagram to describe dynamic aspects of the system. Activity diagram is essentially an advanced version of flow chart that modeling the flow from one activity to another activity.

Activity Diagrams describe how activities are coordinated to provide a service which can be at different levels of abstraction. Typically, an event needs to be achieved by some operations, particularly where the operation is intended to achieve a number of different things that require coordination, or how the events in a single use case relate to one another, in particular, use cases where activities may overlap and require coordination. It is also suitable for modeling how a collection of use cases coordinate to represent business workflows

1. Identify candidate use cases, through the examination of business workflows

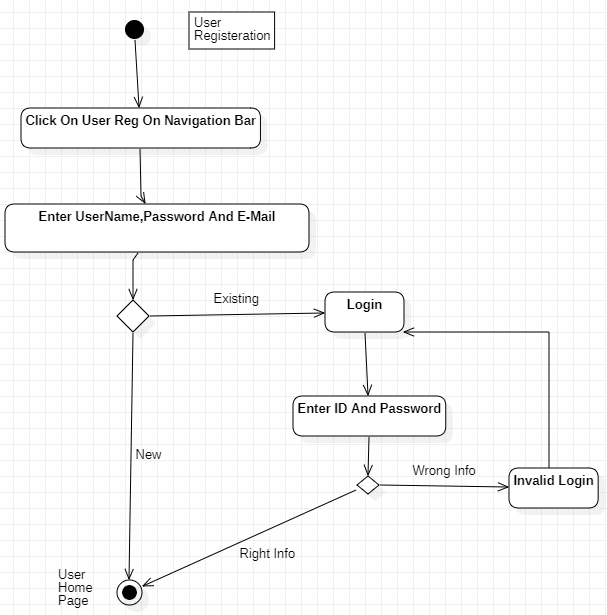
2. Identify pre- and post-conditions (the context) for use cases

3. Model workflows between/within use cases

4. Model complex workflows in operations on objects

5. Model in detail complex activities in a high level activity Diagram.

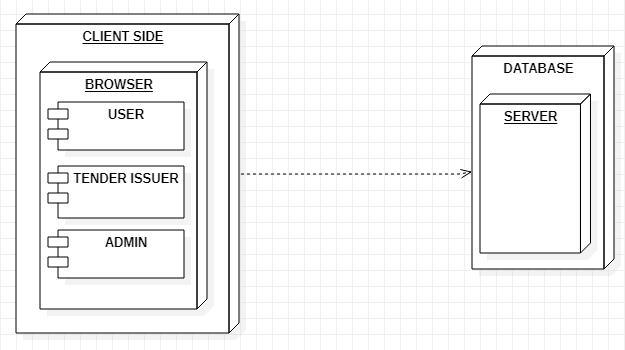
**ACTIVITY DIAGRAM :**

****

* **Deployment Diagram:**

This is the diagram which lets us inspect the run-time view of the nodes which are involved in the processing of the components that live on them. The diagram acknowledges the architecture of the static deployment view of the project. The diagram follows as components like a node typically related to or enclosed with one or more components.

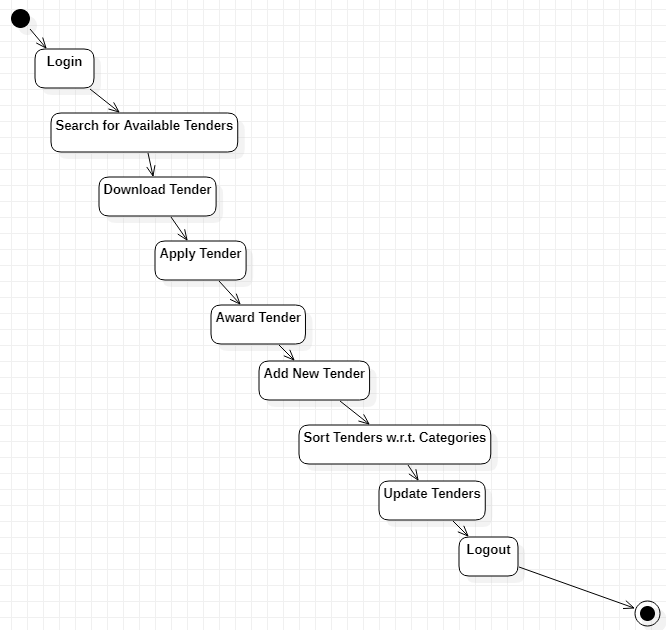
**Deployment diagram of our project:**

****

* **State Chart Diagram:**

This diagram depicts the state of the object and represents their activities as arrows which joins all the states. We will highlight the activities. Each activity will be represented by a rounded rectangle-narrowed and oval-shaped state icon. The transition from one state to another is represented by an arrow which connects one activity to the next. The starting point of the diagram is represented by a filled-circle and then the end point of the diagram is represented by a bull’s eye.

**State chart diagram of our project:**

****

* **Flow Diagram:**

**Flow diagram** is a collective term for a diagram representing a flow or set of dynamic relationships in a system. The term flow diagram is also used as a synonym for flowchart and sometimes as a counterpart of the flowchart.

Flow diagrams are used to structure and order a complex system or to reveal the underlying structure of the elements and their interaction.

The term flow diagram is used in theory and practice in different meanings. Most commonly the flow chart and flow diagram are used in an interchangeable way in the meaning of a representation of a process. For example, the *Information Graphics: A Comprehensive Illustrated Reference* by Harris (1999) gives two separate definitions:

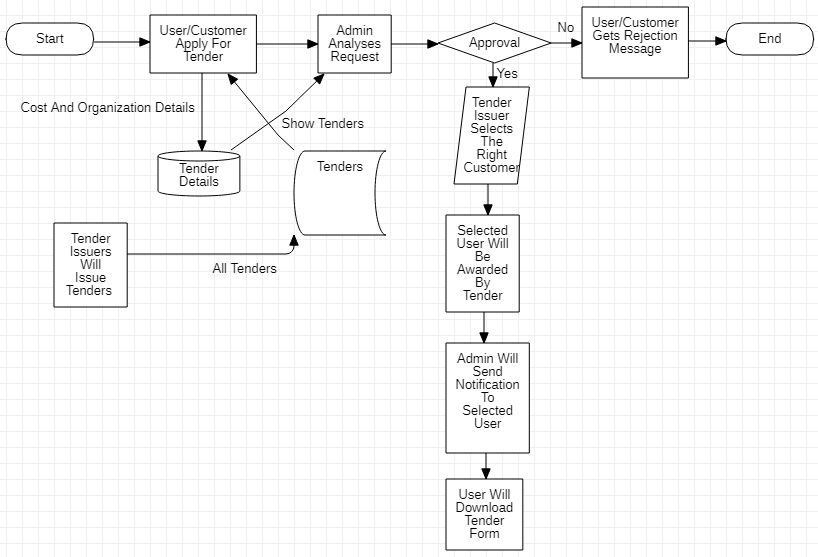
**Flow chart or flow diagram** is a diagram that visually displays interrelated information such as events, steps in a process, functions, etc., in an organized fashion, such as sequentially or chronologically.

**Flow diagram** is a graphic representation of the physical route or flow of people, materials, paper works, vehicles, or communication associated with a process, procedure plan, or investigation*.*

In the second definition the meaning is limited to the representation of the physical route or flow. An example of such a diagram is the illustration of the flows in a nuclear submarine propulsion system, which shows different streams back and forth in the system. The representation of such a system can be supplemented by one or more flowcharts, which show the sequence of one of the flows in one direction, or any of the control flows to manage the system.

The physical movement of objects from one location to another can also be visualized in a mix of maps and flowchart or Sankey diagram, which are called flow maps. Flow maps can indicate on a map, what flows, moves or migrates, in which direction, and in which quantities etc.

**FlowChart Diagram :**

****

**Test case design**

**•**Test is the process carried out on software to detect the differences between its behavior and the desired behavior as stipulated by the requirements specifications.

**•Test is advantageous in the several ways**

**•**Firstly, the defects found help in the process of making the software reliable

**•**Secondly, even if the defects found are not corrected, testing gives an idea as to how reliable the software is.

**•**Thirdly, over time the record of defects found reveals the most common kind of defects, which can be used for developing appropriate preventive majors such as training prover design and reviewing.

**4.3 Test plan**

The testing sub-process includes the following activities in a phase dependent manner:

a) Create test plans

b) Create test specification

e) Review test plans and test specification

d) Conduct test according to the test specification and log the defects

e) Fix defects, if any

f) When defects are fixed continue from activity

**CHAPTER 5: IMPLEMENTATION & TESTING**

**Security Testing of the Project**

**Testing**

System testing is designed to uncover the weaknesses that were not found in earlier test. In the testing phase, the program is executed with the explicit intention of finding errors.

This includes forced system failures and validation of the system, as its user in the operational environment will implement it. For this purpose test cases are developed. When a new system replaces the old one, such as in the present case, the organization can extract data from the old system to test them on the new. Such data usually exist in sufficient volume to provide sample listings and they can create a realistic environment that ensures eventual system success. Regardless of the source of test data, the programmers and analyst will eventually conduct four different types of tests.

**The steps in the software testing**

The steps involved during Unit testing are as follows:

a. Preparation of the test cases.

b. Preparation of the possible test data with all the validation checks.

c. Complete code review of the module.

d. Actual testing done manually.

e. Modifications done for the errors found during testing.

f. Prepared the test result scripts.

**The unit testing done included the testing of the following items:**

1. Functionality of the entire module/forms.

2. Validations for user input.

3. Checking of the Coding standards to be maintained during coding.

4. Testing the module with all the possible test data.

5. Testing of the functionality involving all type of calculations etc.

6. Commenting standard in the source files.

After completing the Unit testing of all the modules, the whole system is integrated

with all its dependencies in that module. While System Integration, We integrated

the modules one by one and tested the system at each step. This helped in

reduction of errors at the time of the system testing.

**The steps involved during System testing are as follows:**

Integration of all the modules/forms in the system.

Preparation of the test cases.

Preparation of the possible test data with all the validation checks.

Actual testing done manually.

Recording of all the reproduced errors.

Modifications done for the errors found during testing.

Prepared the test result scripts after rectification of the errors.

**The System Testing done included the testing of the following items:**

1. Functionality of the entire system as a whole.

2. User Interface of the system.

3. Testing the dependent modules together with all the possible test data

scripts.

4. Verification and Validation testing.

5. Testing the reports with all its functionality.

After the completion of system testing, the next following phase was the Acceptance Testing. Clients at their end did this and accepted the system with appreciation. Thus, we reached the final phase of the project delivery.

**There are other six tests, which fall under special category. They are described below:**

Peak Load Test:

It determines whether the system will handle the volume of activities that occur when the system is at the peak of its processing demand. For example, test the system by activating all terminals at the same time.

Storage Testing:

It determines the capacity of the system to store transaction data on a disk or in other files.

Performance Time Testing:

it determines the length of time system used by the system to process transaction data. This test is conducted prior to implementation to determine how long it takes to get a response to an inquiry, make a backup copy of a file, or send a transmission and get a response.

Recovery Testing:

This testing determines the ability of user to recover data or re-start system after failure. For example, load backup copy of data and resume processing without data or integrity loss.

Procedure Testing:

It determines the clarity of documentation on operation and uses of system by having users do exactly what manuals request. For example, powering down system at the end of week or responding to paper-out light on printer.

Human Factors Testing:

It determines how users will use the system when processing data or preparing reports.

**Levels of Testing:**

 **Unit Testing:**

In unit testing the analyst tests the programs making up a system. For this reason, unit testing is sometimes called program testing. Unit testing gives stress on the modules independently of one another, to find errors. This helps the tester in detecting errors in coding and logic that are contained within that module alone. The errors resulting from the interaction between modules are initially avoided. The test cases needed for unit

testing should exercise each condition and option. Unit testing can be performed from the bottom up, starting with smallest and lowest-level modules and proceeding one at a time. For each module in bottom-up testing a short program is used to execute the module and provides the needed data, so that the module is asked to perform the way it will when embedded within the larger system.

 **System Testing:**

**System testing consists of the following five steps:**

**1) Program Testing**

A program represents the logical elements of a system. For a program to run satisfactorily, it must compile and test data correctly and tie in properly with other programs. it is the responsibility of a programmer to have an error free program. At

the time of testing the system, there exists two types of errors that should be checked. These errors are syntax and logic. A syntax error is a program statement that violates one or more rules of the language in which it is written. An improperly defined field dimension or omitted key words are common syntax errors. These errors are shown through error messages generated by the computer. A logic error, on the other hand, deals with incorrect data fields out of range items, and invalid combinations. Since the logical errors are not detected by compiler, the programmer

must examine the output carefully to detect them. When a program is tested, the actual output is compared with the expected output. When there is a discrepancy, the sequence of the instructions, must be traced to determine the problem. The process is facilitated by breaking the program down into selfcontained portions, each of which can be checked at certain key points.

**2) String Testing**

Programs are invariably related to one another and interact in a

total system. Each program is tested to see whether it conforms

to related programs in the system. Each part of the system is

tested against the entire module with both test and live data

before the whole system is ready to be tested.

**4) System Documentation**

All design and test documentation should be well prepared and kept in the library for future reference. The library is the central location for maintenance of the new system**.**

**5) User Acceptance Testing**

An acceptance test has the objective of selling the user on the validity and reliability of the system. It verifies that the system's procedures operate to system specifications and that the integrity of important data is maintained. Performance of an acceptance test is actually the user's show. User motivation is very important

for the successful performance of the system. After that a comprehensive test report is prepared. This report shows the system's tolerance, performance range, error rate and accuracy. User acceptance of a system is a key factor for the success of any system.

** Special Systems Tests:**

There are other six tests which fall under special category. They are described below:

**Peak Load Test** :

1. It determines whether the system will handle the volume of activities that occur when the system is at the peak of its processing demand. For example, test the system by activating all terminals at the same time.
2. **Storage Testing:** It determines the capacity of the system to store transaction data on a disk or in other files. For example, verify documentation statements that the system will store 10,000 records of 400 bytes length on a single flexible disk.
3. **Performance Time Testing:** it determines the length of time system used by the system to process transaction data. This test is conducted prior to implementation to determine how long it takes to get a response to an inquiry, make a backup copy of a file, or send a transmission and get a response.
4. **Recovery Testing:** This testing determines the ability of user to recover data or re-start system after failure. For example, load backup copy of data and resume processing without data or integrity loss.
5. **Procedure Testing:** It determines the clarity of documentation on operation and use of system by having users do exactly what manuals request. For example, powering down system at the end of week or responding to paper-out light on printer.
6. **Integration Testing :** The integration is the next important concept that highlights in the testing scenario. Integration testing can be performed in different strategies. One of them is the Big Bang testing in which one could first test all of a system’s modules separately and then whole systems at once. But here we proceed abruptly from the module testing and the integration testing disappears. Another alternative is the Incremental Testing.With the Incremental testing there are many advantages. We can start the integration as soon reasonable subsets of modules have been developed. It is easier to localize errors incrementally. The partial aggressions of modules often constitute important subsystems that can have autonomy with these testing. The need for stubs and drivers can be reduced. There are two approaches to the Incremental Testing. They include Bottom-up and Top-down aggregations. The former means starting aggregation and testing from leaves of the module charts. The l tter means starting from the top-level modules and

substitute for higher-level modules. In our project we have used the top-down approach of incremental the ting. Top-down integration is an incremental approach to the construction of programs structure. Modules are integrated by moving downward through the control hierarchy, beginning with the main control module that is the basic connectivity module in our project. Test is done on each module. The top down integration strategy verifies major control or decision points. In the beginning of the integration phase dummy frames were selected as stubs to ensure that the data flow occurred through the correct hierarchical structure. Later the actual module replaces these stubs.

** Output Testing**

After performing the validation testing the next step is output testing of the proposed system since no system is useful if it does not produce the required output in the specific format. The outputs generated or displayed by the system under consideration are tested by asking the users about the formats required by them.

 **Quality Assurance Methodologies**

Quality assurance is a planned and systematic of all actions necessary to provide adequate confidence that the item or product confirms to established technical requirements. The purpose of software quality assurance group is to provide assurances

that the procedures, tools and techniques used during product development and modification and adequate to provide desired level of confidence in the work products. Often, software quality assurance personnel are organizationally distinct from software development group. Preparation of a Software Quality Assurance Plan for each software products is primary responsibility of software quality assurance group. Quality assurances personnel are sometimes are charge of arrangements for walkthroughs, inspections and major milestones reviews In addition, quality assurance personnel often conduct the project post mortem, write project legacy document and provide long term retention of the project records. Typically the quality assurance group will work with the development group to derive Source Code Test Plan*.* A test plan for the source code specifies the objectives of testing; the test plan for source code specifies the objectives of testing, the test completion criteria, the system integration plan, and methods to be used on particular test inputs expected outcomes. There are four types of tests that the source code must satisfy: function tests, performance tests, stress test and structural test. Functional test cases specify typical operating conditions, typical input values and typical expected values. Function tests are also tests that are performed on the inside and just beyond the functional boundaries. Examples of functional test include testing a real-valued square route routine with small positive numbers, zero and negative numbers; or testing a matrix version of the inversion routine on a one-by-one matrix and a singular matrix. Performance tests are also designed to verify response time under varying loads, percent execution time spent in various segments of the program, throughput, primary and secondary memory utilization and traffic rates on the data channels and communication links. Stress tests are designed in such a way that to overload a system in various ways. Examples of stress tests include attempting to sign on more than the maximum number allowed terminal, processing more than the allowed number of identifiers or static levels or disconnecting a communication link. Structure test are concerned with examining of the internal processing logic of the software system. The particular routines called and the logic paths traversed through the routines are object of interest.

 **System verification and validation**

System testing is designed to uncover weaknesses that were not found in earlier tests. This includes forced system failure and validation of total system as it will be implemented by its user in the operational environment. Under this testing, generally we take low volumes of transactions based on live data. This volume is increased until the maximum level for each transaction type is reached. The total system is also tested for recovery and fallback after various major failures to ensure that no data are lost during the emergency. All this is done with the old system still in operation. When we see that the proposed system is successful in the test, the old system is continued. It is executing programs to check logical changes made in it with intention of finding errors. a system is tested for online response, volume of transaction, recovery from failure etc. System testing is done to ensure that the system satisfies all the user requirements.

The important and essential part of the system development phase, after designing and developing the software is system testing. We cannot say that every program or system design is perfect and because of lack of communication between the user

and the designer, some error is there in the software development. The number and nature of errors in a newly designed system depend on some usual factors like

communication between the user and the designer; the programmer's ability to generate a code that reflects exactly the systems specifications and the time frame for the design. Theoretically, a newly designed system should have all the parts

or sub-systems are in working order, but in reality, each subsystem works independently. This is the time to gather all the subsystem into one pool and test the whole system to determine whether it meets the user requirements. This is the last change to detect and correct errors before the system is installed for user acceptance testing. The purpose of system testing is to consider all the likely variations to which it will be subjected and then push the system to its limits. Testing is an important function to the success of the system. System testing makes a logical assumption that if all the parts of the system are correct, the goal will be successfully activated.

Moving through each module from top to bottom tests the entire system. The verification and validation process are then carried out. The errors that occur the testing phase are eliminated and a well functioning system is developed. Test case design focuses on a set of techniques, which meets all testing objectives, which are mentioned below. Testing is a process of executing a program with the intent of finding an error.

Another reason for system testing is its utility as a user-oriented vehicle before implementation. A number of different transactions are used to perform verification. Validation is the process of demonstrating that the implemented software does satisfy the system requirements. One aspect of software validation is to statistically analyse the program without resorting to actual execution. The system validation done in such-a-way that the system response time will not cause any hardship to the user. The system testing deals with the process of testing the system as a whole. This is done after the integration process.

** White Box Testing**

White box testing is a test case design method that uses the control

structure of the procedural design to derive test cases. Using white box testing

methods, we can derive test cases that in this technique, the close examination of the logical parts through the software are tested by cases that exercise species sets of conditions or loops. all logical parts of the software checked once. errors that can be corrected using this technique are typographical errors, logical expressions which should be executed once may be

getting executed more than once and error resulting by using wrong controls and loops. When the box testing tests all the independent part within a module a logical decision on their true and the false side are exercised, all loops and bounds within their operational bounds were exercised and internal data structure to ensure their validity were exercised once.

1. Guarantee that all independent paths within a module have been exercised at least once.
2. Exercise all logical decisions on their true and false sides
3. Execute all loops at their boundaries and within their operational bounds
4. Exercise internal data structures to ensure their validity

** Black Box Testing**

Black box testing methods focus on the functional requirements if the software. That is, black box testing enables us to derive sets of input conditions that will fully exercise all functional requirements of the program. This method enables the software engineer to device sets of input techniques that fully exercise all functional requirements for a program. black box testing tests the input, the output and the external data. it checks whether the input data is correct and whether we are getting the desired output

Black box testing attempts to find errors in following categories:

1. Incorrect or missing functions
2. Interface errors
3. Errors in data structures or external database access
4. Performance errors Initialization and termination errors.

**CHAPTER 6: RESULTS AND DISCUSSION**

**RESULTS AND DISCUSSION**

### Admin

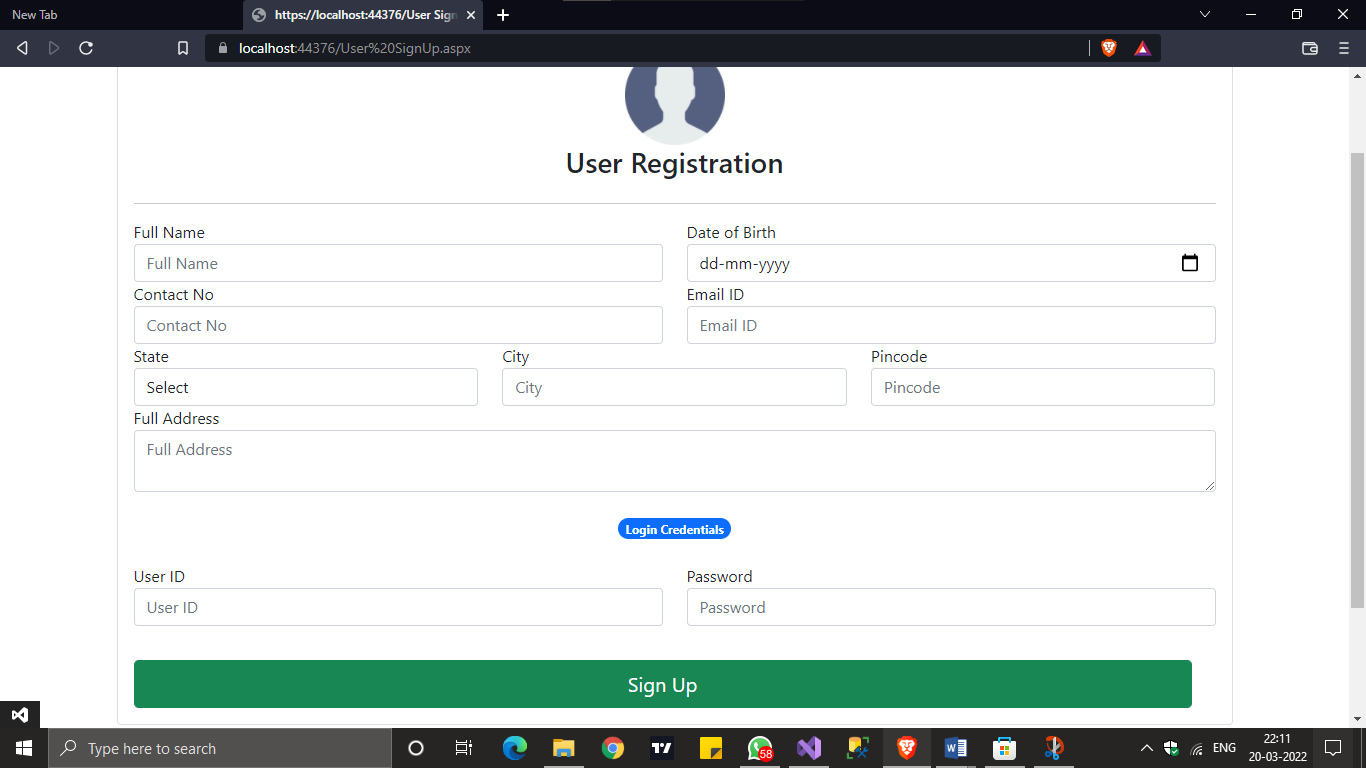
### In This Field, We Tested Every Thing About Administrator Roles, Like Maintaining The Users And Issuers As Per The Convenience of this Project. Admin Have Authority To Delete Issuer or User If Any Malfunction Occurs Or Illegal Activity.

### Admin Page –

### 

So, The Admin Have **Admin ID** & **Password** To Login, Managing Users And Issuers.

**User SignUp –**

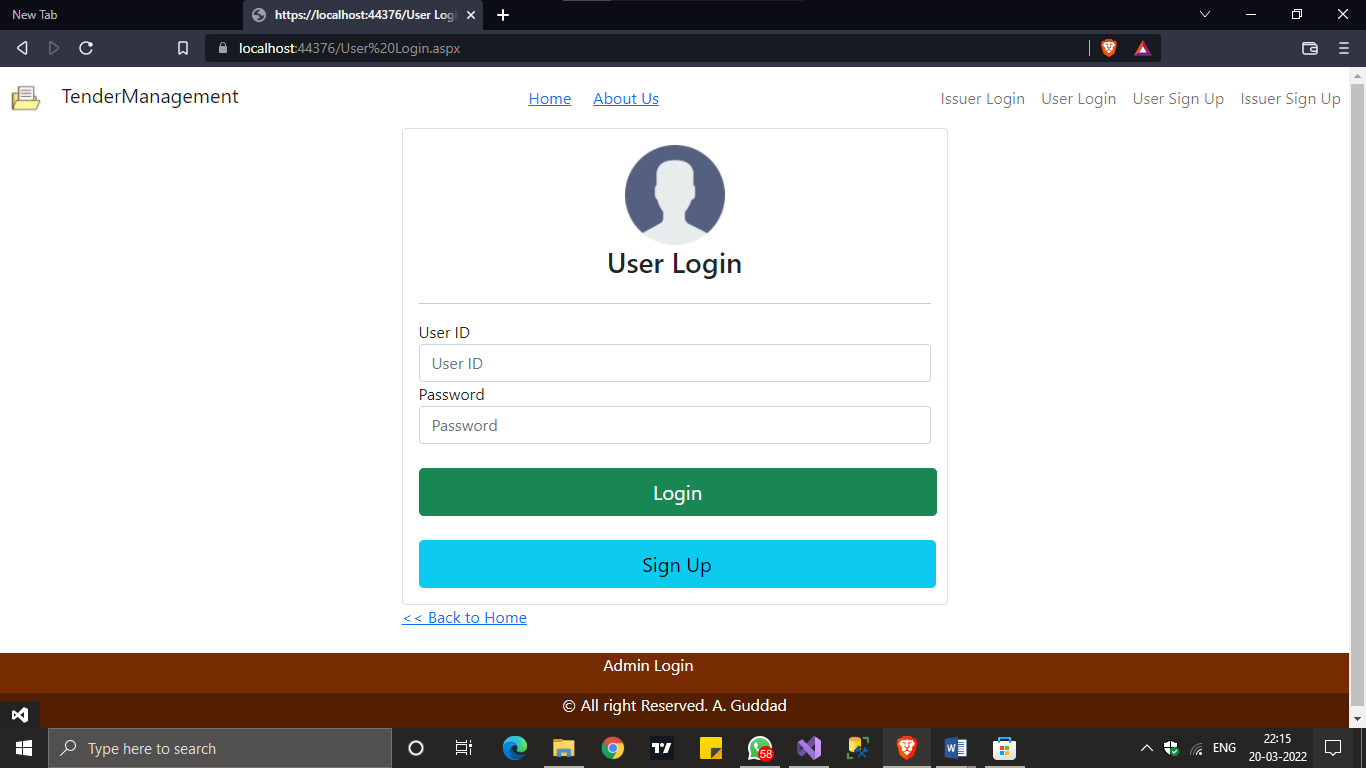


If New Member/Contractor Want to check out the Tenders So He Need To Be User,

So He Need To SignUp First. Fill Out All The Field And Decide **User ID & Password** For Further

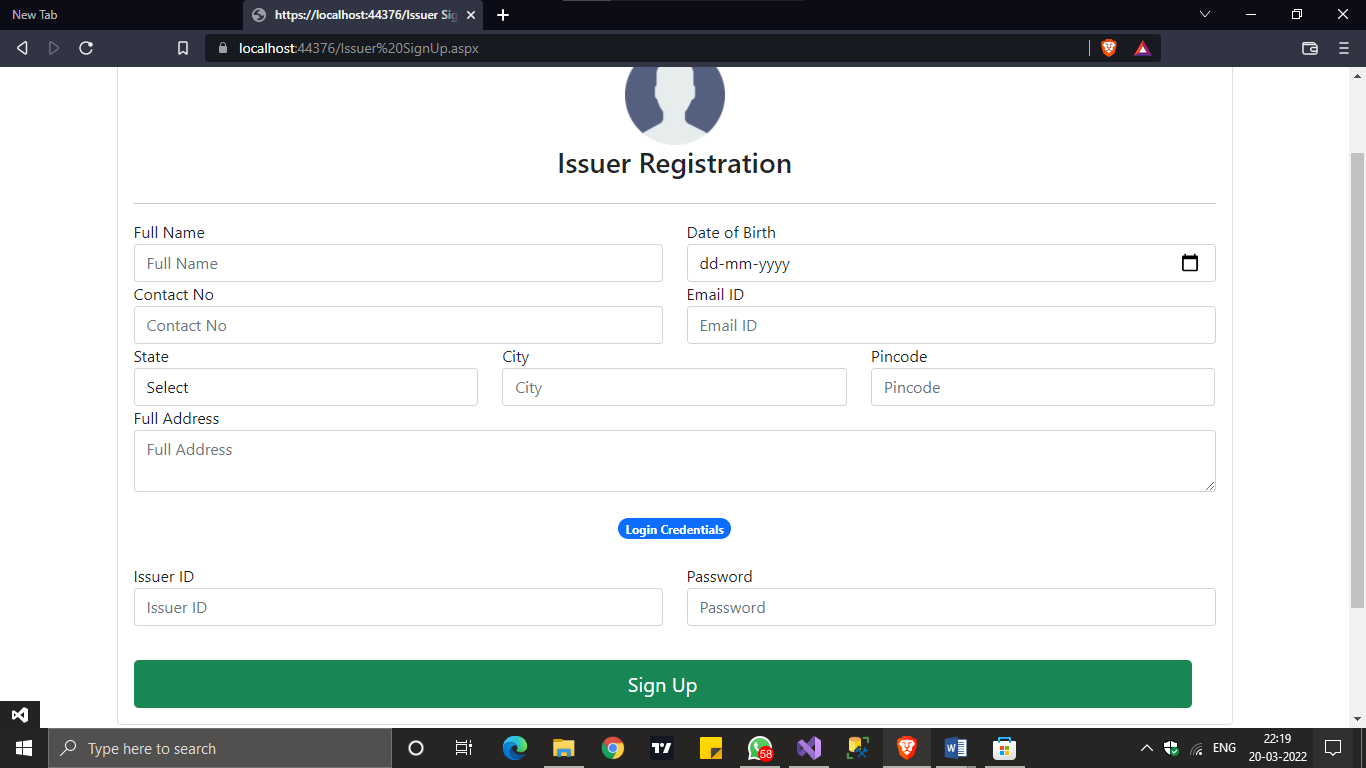
Login sessions.

**User Login –**

****

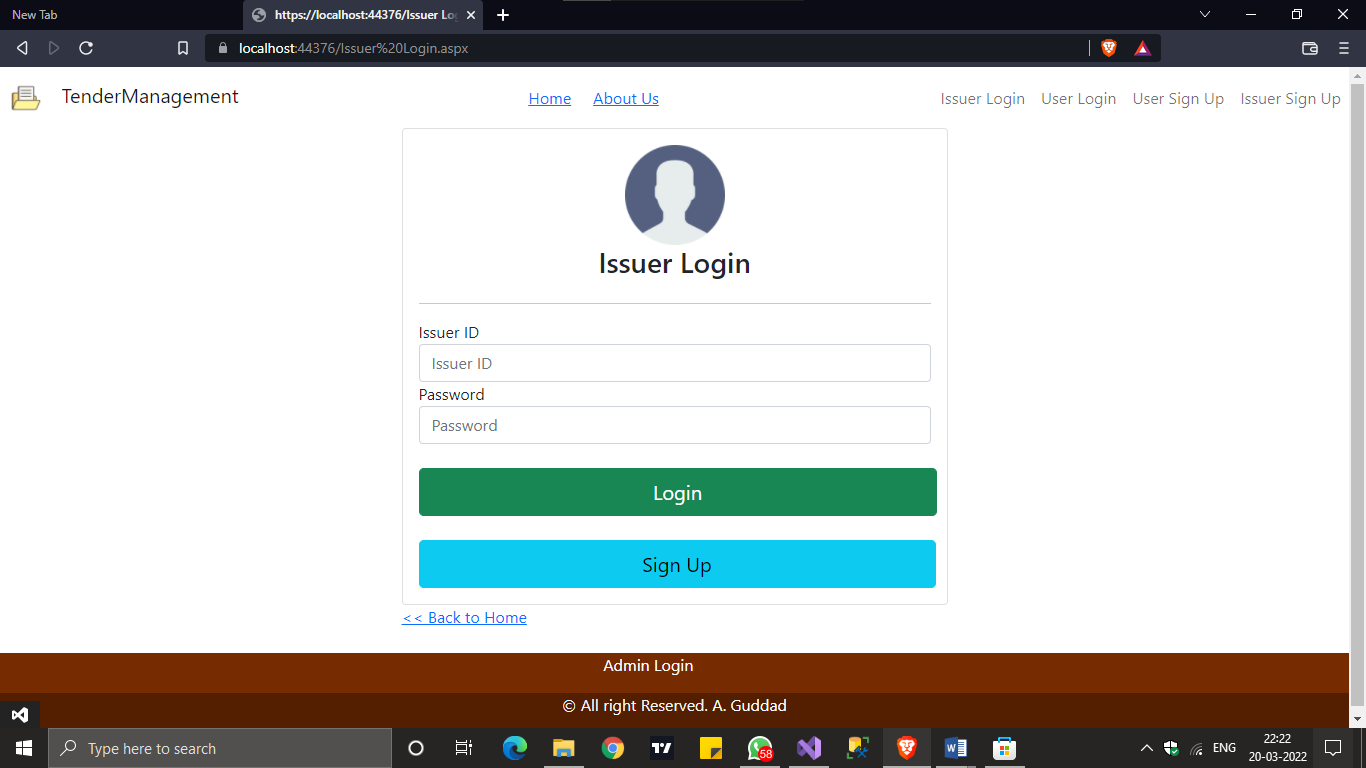
When Contractor Registered As A User, Then He Should Use **User ID & Password** For Login He Provided While Registering As User.

**Issuer SignUp –**

****

If Someone Wants to Only Issue Tenders For Their Work Related Things, Then He Should Register As Issuer By Filling All The Fields And Deciding **Issuer ID & Password** For Further Login Sessions.

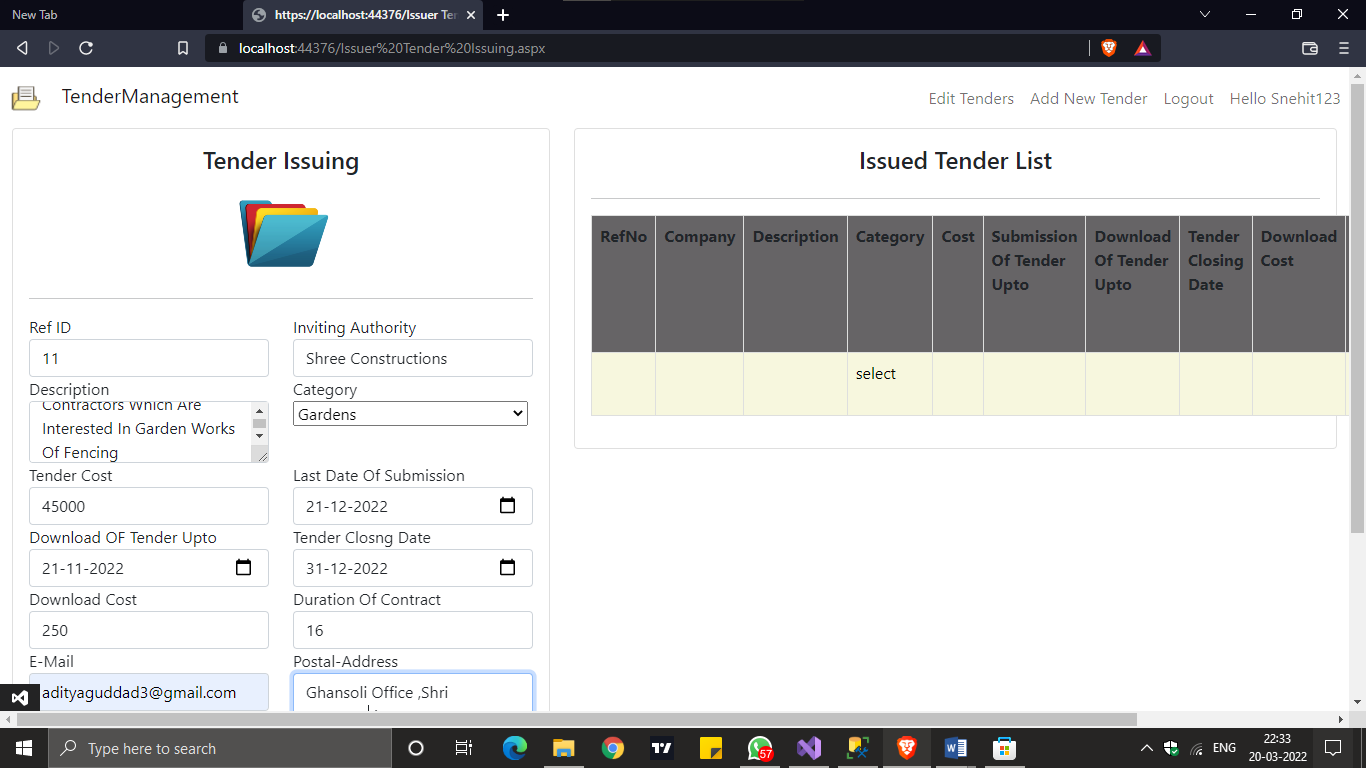
**Issuer Login –**

****

The Issuer Should Use **Issuer ID & Password** for logging in To His Profile.

**ID & Password** Which is Provided While Registering As Issuer.

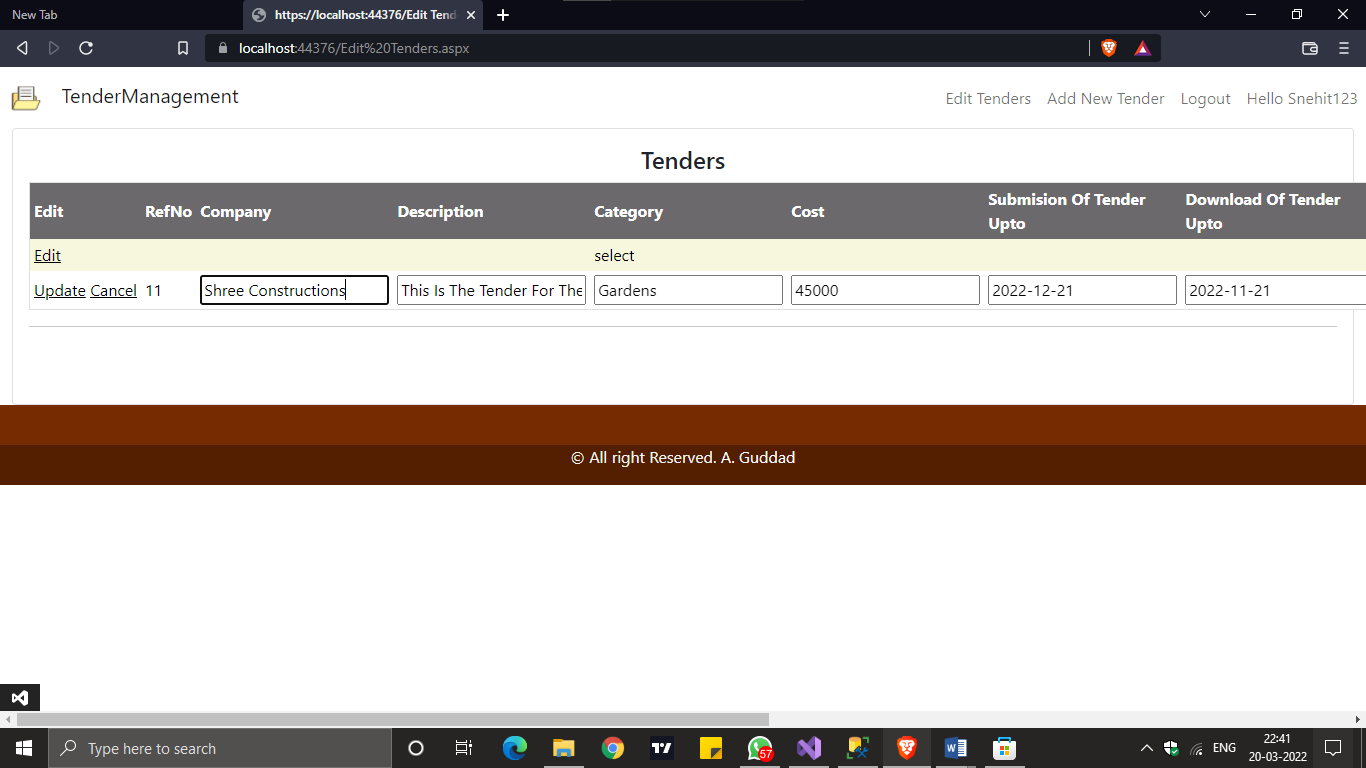
**Adding New Tender –**



The Issuer/Company Will Seek Contractors To Take A Look At The Tender And The Cost , Due Date ,E-Mail & Address To Send Application To Apply for This Tender.

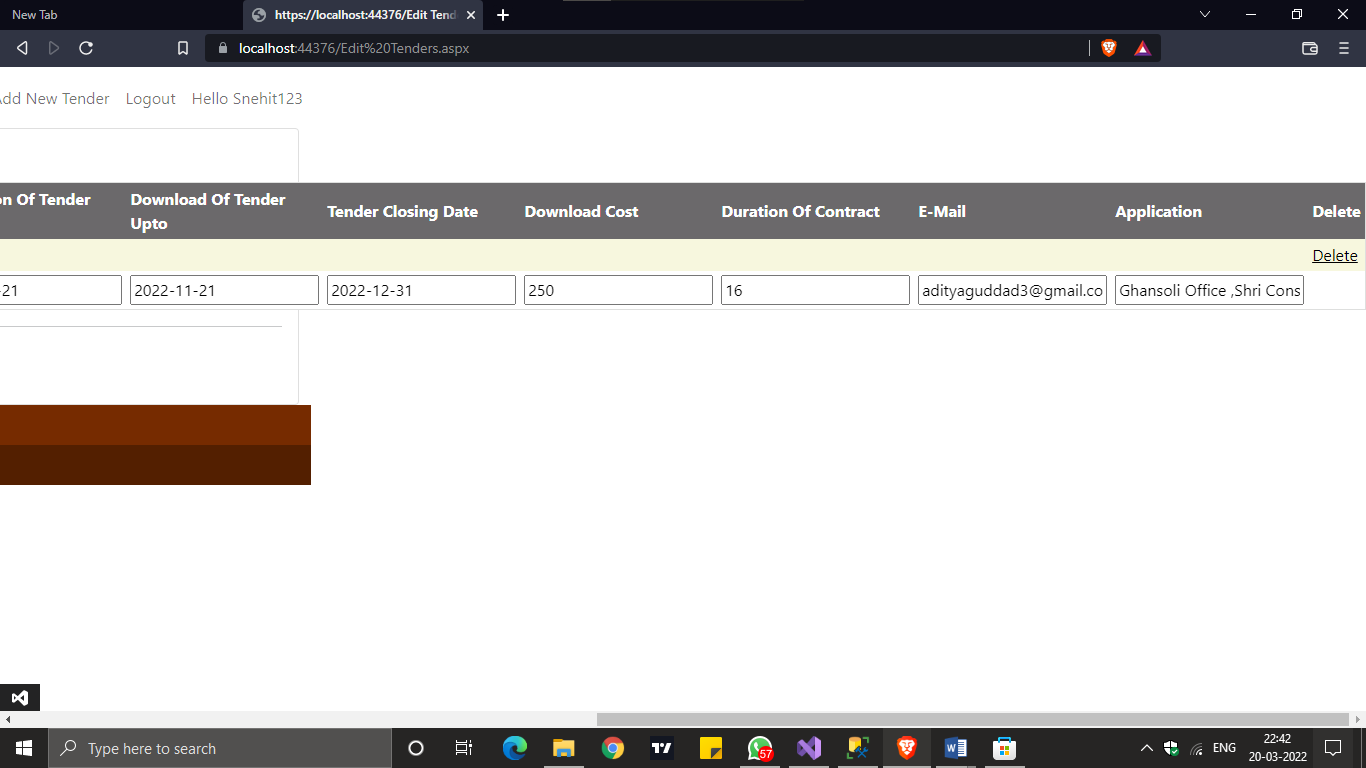
After Issuing This Tender, The Users Will Get To Know About This Project Just By Logging In On Their Profile.

**Editing Tender –**

****

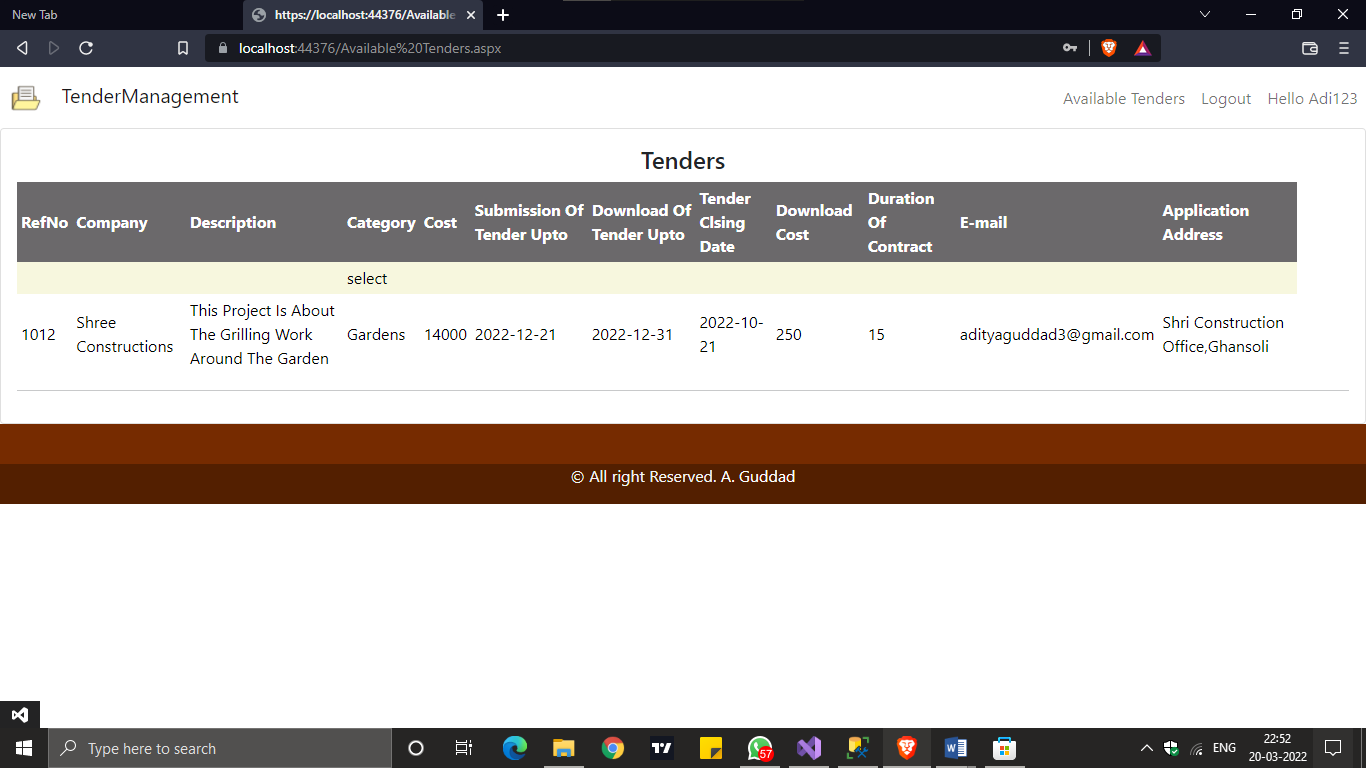
If Issuer Wants To Change The Info About The Tender or The Address Of Application ,Then He Can Use **Edit Tenders** Feature To Edit And Update The Tender.

**Delete The Tender –**



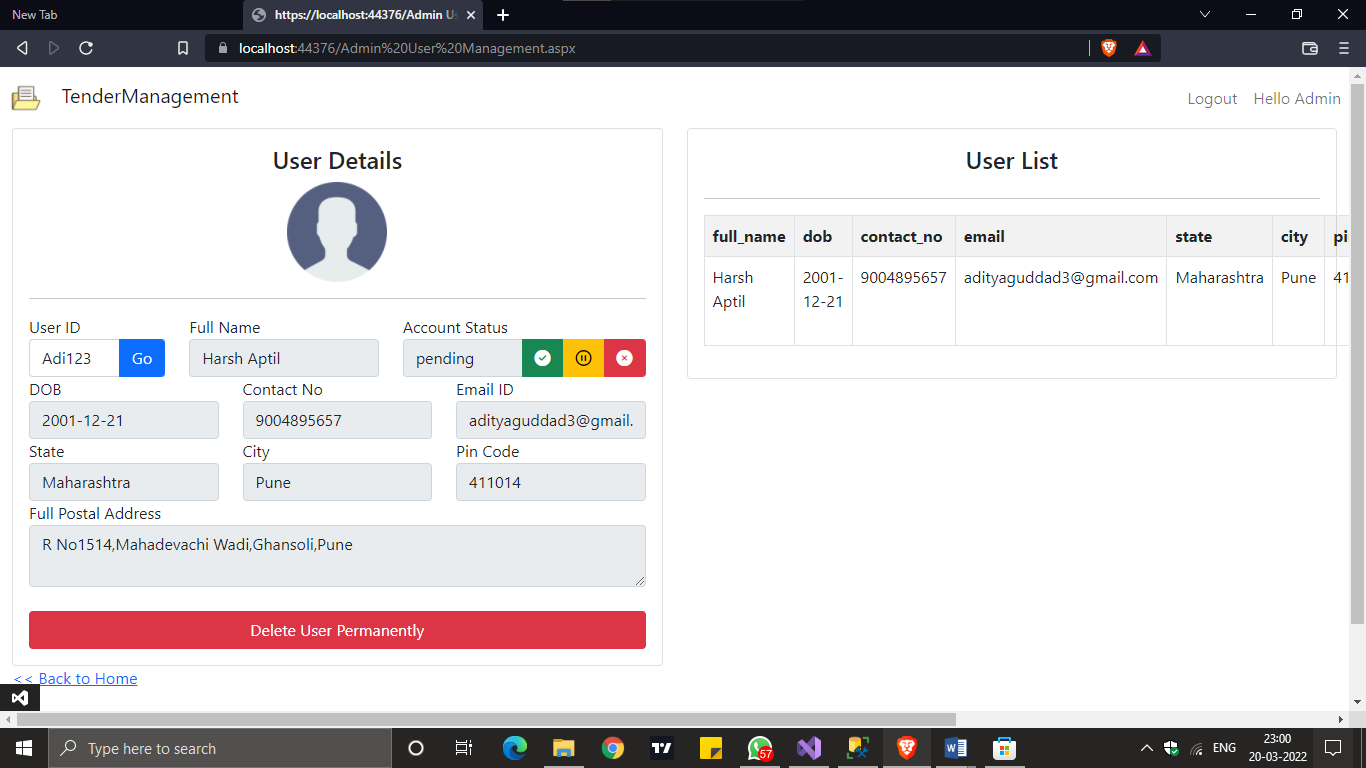
If The Purpose Of The Tender Is Fulfilled, Then Issuer Can Delete That Useless Tender By Clicking **Delete** Button On That Table.

**Available Tenders –**

****

The User Need To Login To His Account, And Click **Available Tenders** Navigation Button To Check Out The Latest Tender/Project And If He’s Interested He Can Contact And Send The Application to the Provided Address.

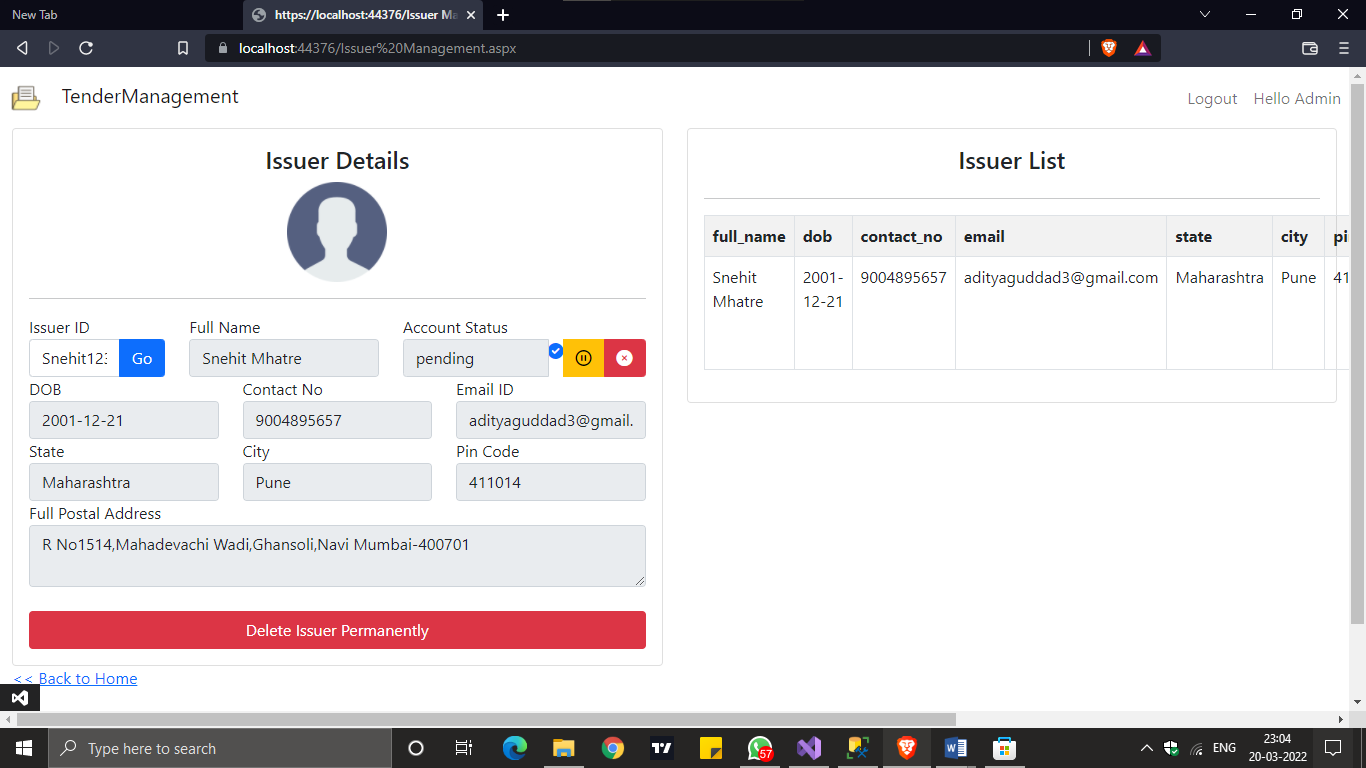
**User Management –**



So, The Admin Can Login And Manage The Users. By Using The **User ID** Admin Can See The Detail Of The User, Update The User Status And Delete The User Permanantely.

But Admin Can’t Edit The Personal Information About User.

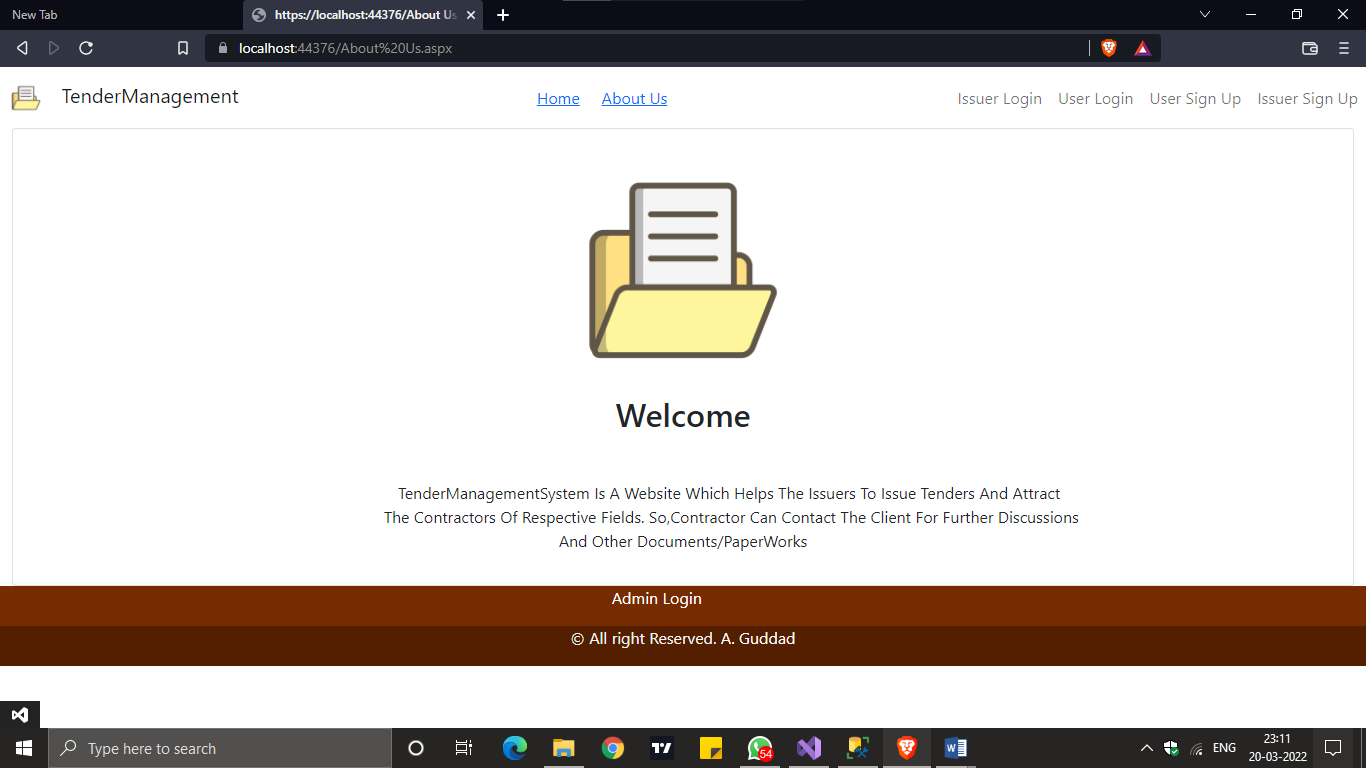
**Issuer Management –**



So, The Admin Can Login And Manage The Users. By Using The **Issuer ID** Admin Can See The Detail Of The Issuer, Update The Issuer Status And Delete The Issuer Permanantely.

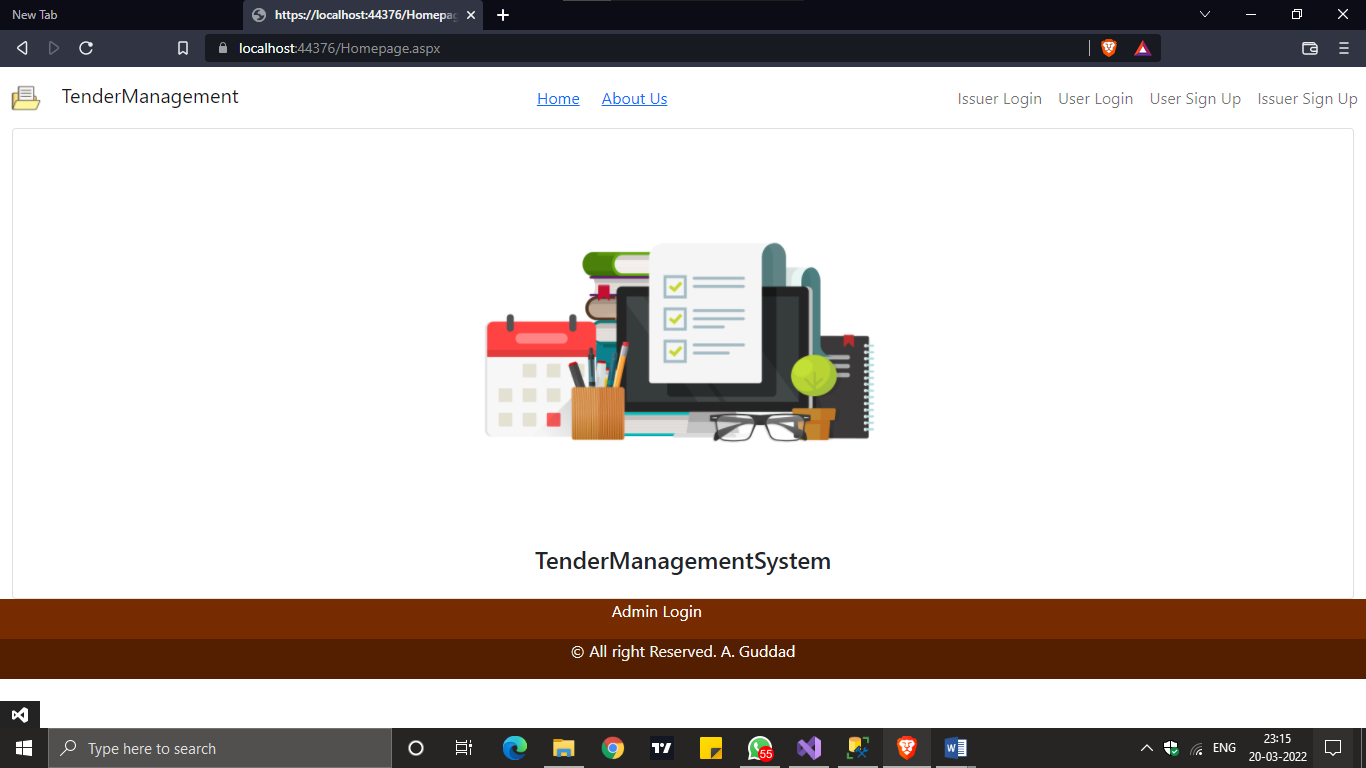
But Admin Can’t Edit The Personal Information About Issuer.

**About Us –**



This Page Contains The Goal About This Website In Short.

**Homepage –**



This Is Homepage Of The Website, It Contains The **Navigation Bar & Footer Bar.**

**The Road to All Registeration , Login And Admin Login** Pages.

**CHAPTER 7: CONCLUSION & FUTURE ENHANCEMENTS**

**Conclusion**

* Being a computerized system it is accurate, time saving and beneficial with paperless documents. It minimizes error at greater extent.
* Easy to release the tenders for an Organization.
* Easy to get the customers from different region because of online
* tenders
* Easy to register the customers and get the feedback from the customers.

At the end it is concluded that we have made effort on following points...

* A description of the background and context of the project and its relation to work

already done in the area.

* Made statement of the aims and objectives of the project.
* The description of Purpose, Scope, and applicability.
* We define the problem on which we are working in the project.
* We describe the requirement Specifications of the system and the actions that

can be done on these things.

* We understand the problem domain and produce a model of the system, which

describes operations that can be performed on the system.

* We included features and operations in detail, including screen layouts.
* We designed user interface and security issues related to system.
* Finally the system is implemented and tested according to test cases.

**FUTURE ENHANCEMENTS –**

* It can be implemented to upload files with an huge amount of size with the support of various file formats.
* This System being web-based and an undertaking of Cyber Security Division, needs to be thoroughly tested to find out any security gaps.
* A console for the data centre may be made available to allow the personnel to monitor on sites which were cleared for hosting during a particular period.
* Moreover, it is just a beginning; further the system may be utilized in various other types of auditing operation viz. Network auditing or similar process/workflow based applications...