|  |  |
| --- | --- |
| C:\Users\User\AppData\Local\Microsoft\Windows\INetCache\Content.MSO\99A076C5.tmp | **<Tender Management System>**    **Technical Design Document** |
| |  |  |  |  | | --- | --- | --- | --- | |  | **Prepared By / Last Updated By** | **Reviewed By** | **Approved By** | | **Name** | Vinay Yadav,  Devesh Kumawat,  Shaik Mohammed Kashif,  Chandana Satya Sai Phani Kumar P,  Lav Rajput |  |  | | **Role** | Associate-Projects |  |  | | **Signature** |  |  |  | | **Date** | 23/07/2021 |  |  | |
|  |

**Table of Contents**

1. **Introduction**
   1. **Purpose of this Document**
   2. **Project Description**
   3. **Project Overview**
2. **Solution Summary**
   1. **Scope**
   2. **Assumptions**
   3. **Dependencies**
3. **Schematic Diagram**
   1. **Flow Chart**
   2. **Modular Design**
4. **System Design**
   1. **Proposed Design** 
      1. **Login & Registration Module**
      2. **Administration Module**
      3. **Contractor Module**
      4. **Tender Management Module**
      5. **User Services Module**
      6. **Querying & Bidding Module**
   2. **Component Inventory**
      1. **MicroServices**
5. **Data Base Design**
   1. **Data Base Model Diagram**
   2. **Table Structures**
6. **UI Design**
   1. **Registration Page**
   2. **Login Page**
   3. **Contractor View**
   4. **Admin View**
   5. **Logout Page**
   6. **Feedback Page**
7. **Terms & Conditions**
8. **Change Log**

1.0 Introduction

1.1 Purpose of this document

Managing tender is very time-consuming exercise and lot of paper work is required.  Manual work has some merits but are prone to errors and cannot deal with growing volumes of tenders to be managed.

In this Document We will discuss about our Ideas and conceptualities, and describe how we have implemented the same.

1.2 Project description

In modern days, many huge enterprises handle a large number of projects from government and other private enterprises. In all such projects thousands of employees would be working on them. Such an enterprise requires purchases from various vendors for different purposes. The employees would float tenders globally by advertising in reputed newspapers. Advertising in newspapers involves a lot of cost, searching in newspapers is time-consuming, communication gap between buyers and suppliers (only way is through phone or mail), receiving bids and finalizing tenders takes a lot of time. So, we make this whole process of tendering online which is efficient and consumes very less time. And this is now adapted all around the globe ever since the advent of the Web.

This Project revolves around the same conceptuality and idea that Tenders can be managed online. Involving many other revisions about security, feasibility and usability.

In the existing system there were not enough implemented usability features to the end user, so in this project we will be mainly focusing on the User Services like Help, Feedback, Reviews and intuitively understandable designs. And for the security of transactions, Access Control and avoidance of malicious users we have a verification process through submission of Provisional documents.

The proposed System avails Vendors to create tenders, view them, download a report of the tender and from contractors’ side it avails to View tender, bid them, check the bid-status, download a report of the tender.

1.3 Project Overview

Whenever a company requires a service / merchandise, a tender is floated. Company maintains an empaneled list of Vendors. An empaneled vendor can only bid for a tender. Every vendor can bid only once against each tender. Against each tender there may be bids from several vendors. The company will then select the most suitable bid and places the order to that vendor.

* The Administration process of both Evaluating tenders and approving of bids.
* Users Security while creating, bidding, a tender is availed using Document Verification.
* User security, feasibility and usability are the main priority.
* The proposed System avails Vendors to create tenders, view them, download a report of the tender and from contractors’ side it avails to View tender, bid them, check the bid-status, download a report of the tender.

2.0 Solution Summary

* 1. Scope

Tender Management is the process of planning, selecting and publishing bids efficiently, keeping a clear audit trail that can be used to determine best practice for future contracts.

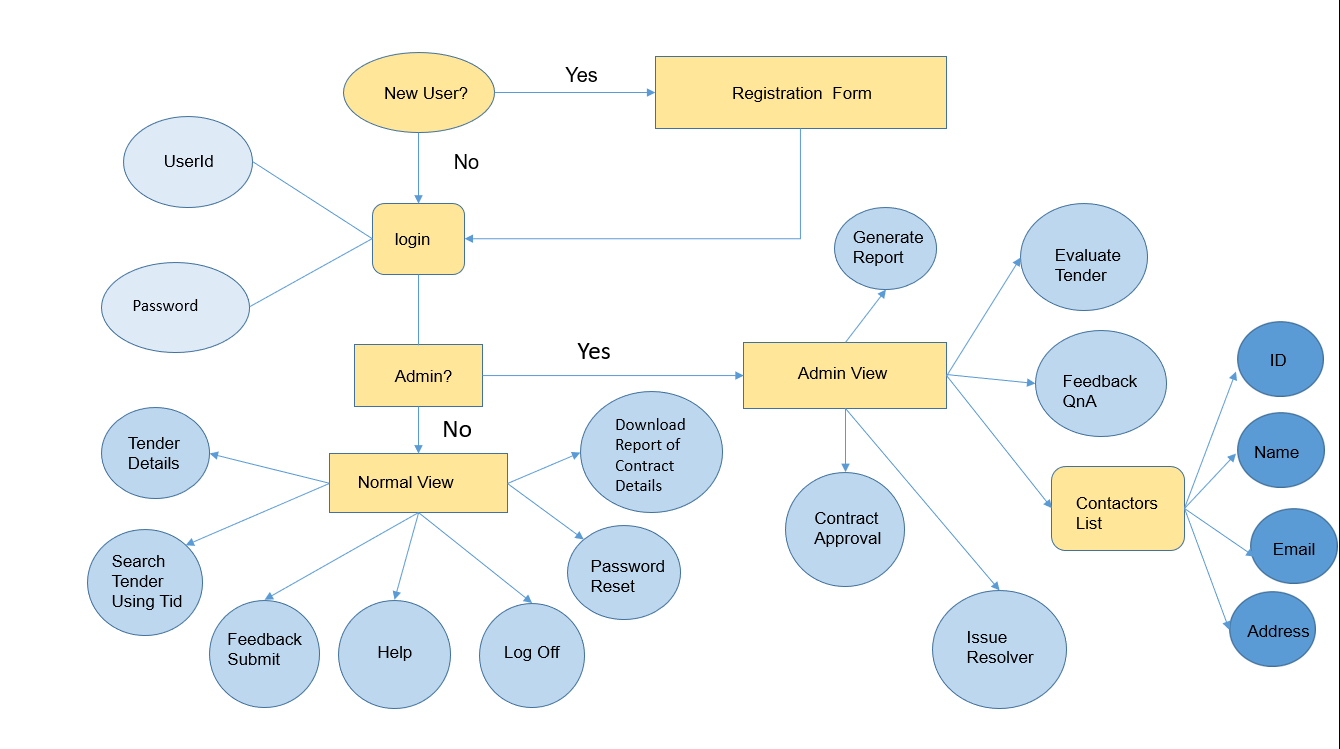
* Clicking on the “Register” button should open the Registration form
* System have Admin Login credentials stored in DB.
* Clicking “Sign-in” should allow user to enter user Id and Password.
* Admin able to display the tender details through online and tender link redirect to tender detail page.
* Admin able to evaluate the tender and able to approve the tender or contract.
* Admin able to create feedback questionnaires and user able to provide feedback.
* User should be able to know the tender details and able to upload documents online and after upload display message upload successfully.

* 1. Assumptions
* When user forgot the User Id then the System ask for the secret question, Email id and Mobile number on answering the questions correctly User Id will be displayed.
* System asks for three secret questions for the recovery of the password in case you forget the password or User Id.
* If User Fails to provide the mandatory fields, then system should show an alert message “Please update the highlighted mandatory fields”.
* Admit can approve the contract and user can give feedback

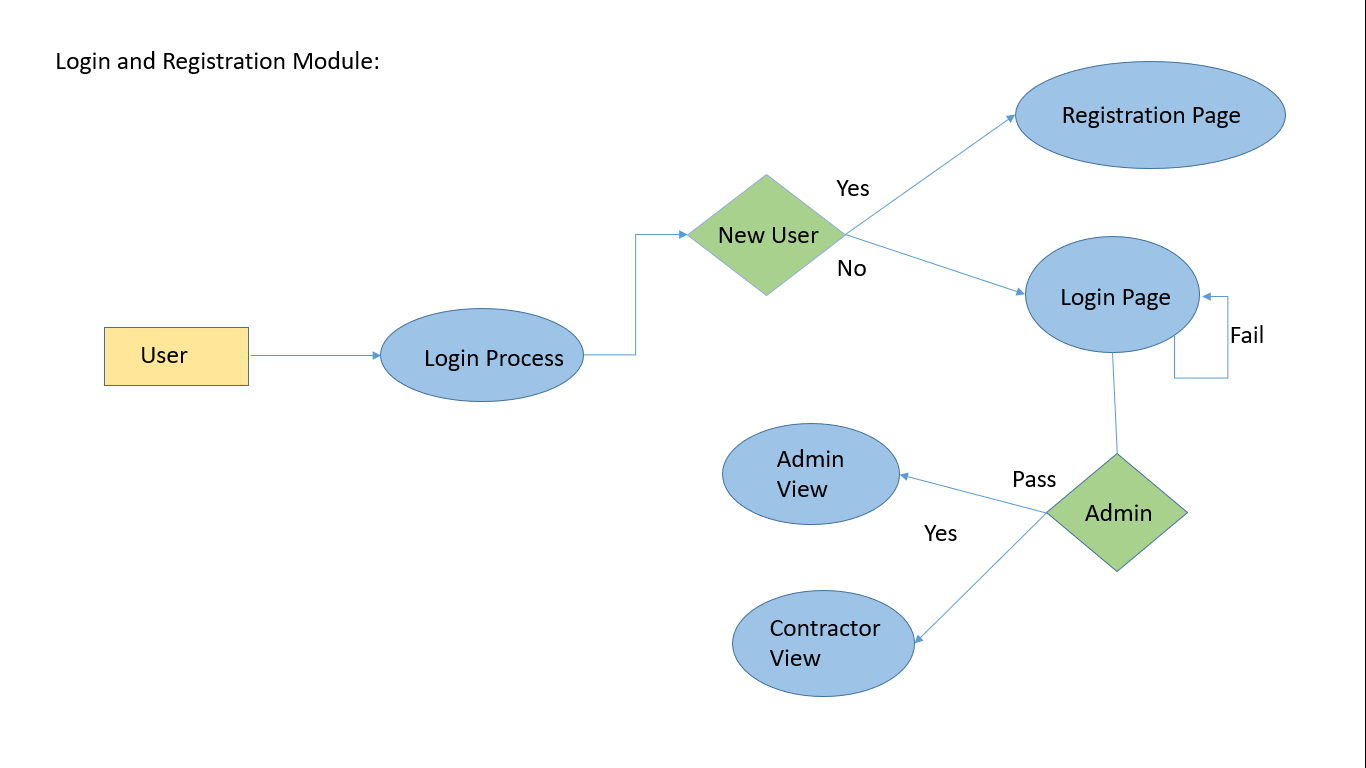
* 1. Dependencies
* MySQL
* Eclipse

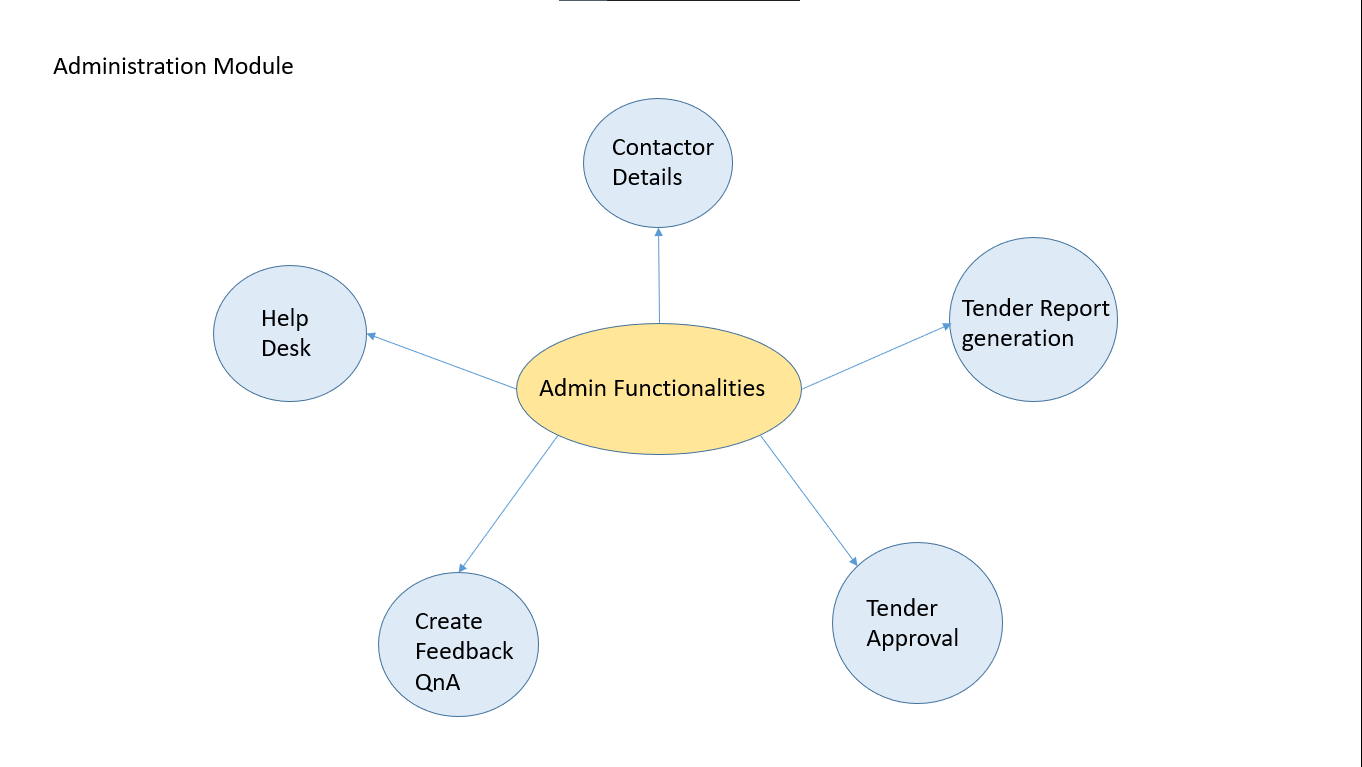
3.0 Schematic Diagram

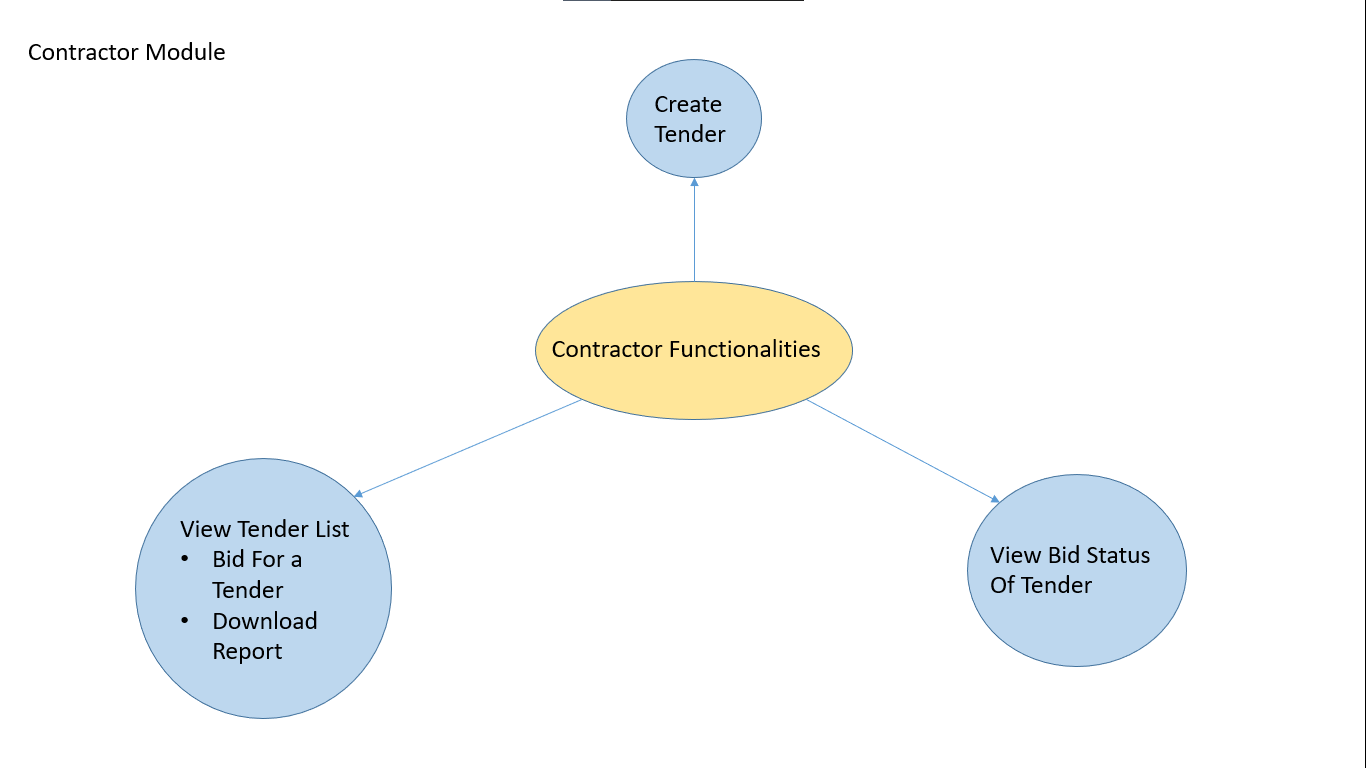
3.1 Flow chart

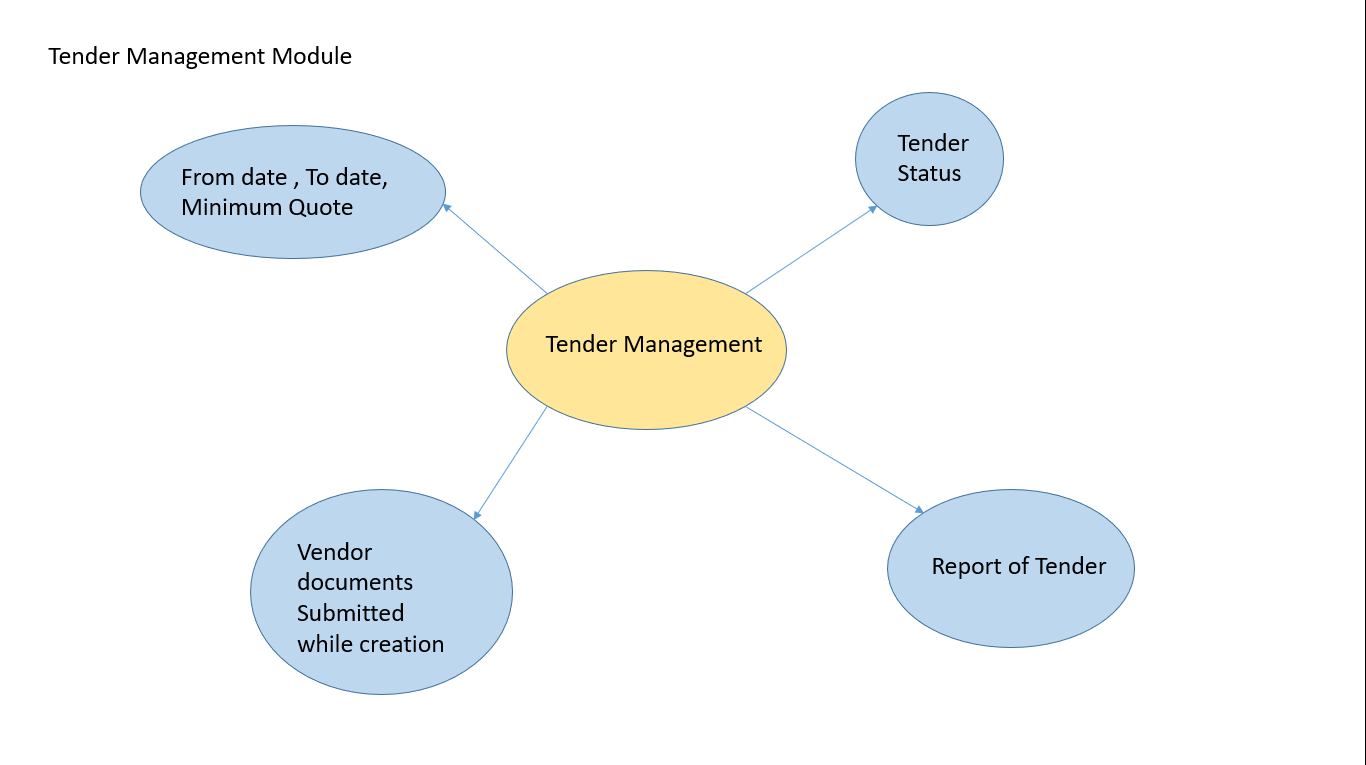


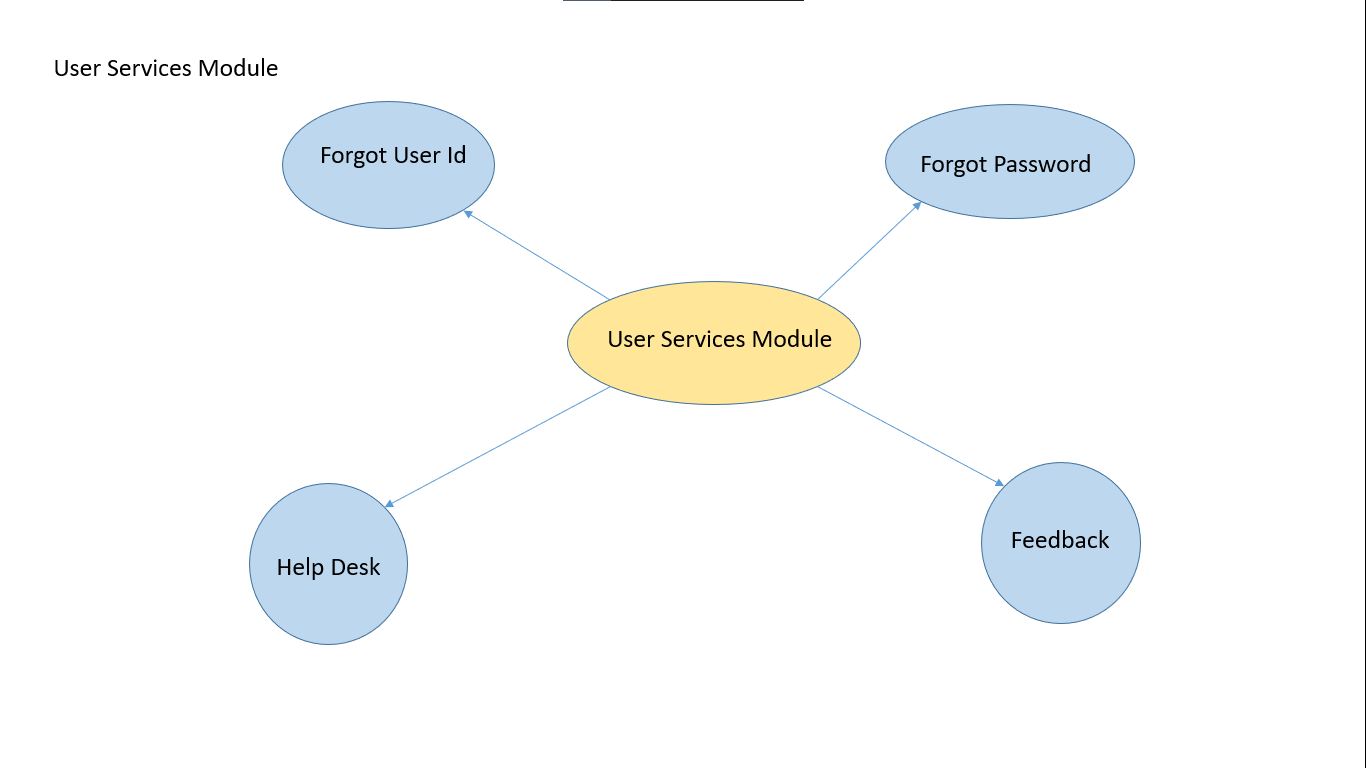
3.2 Modular Design

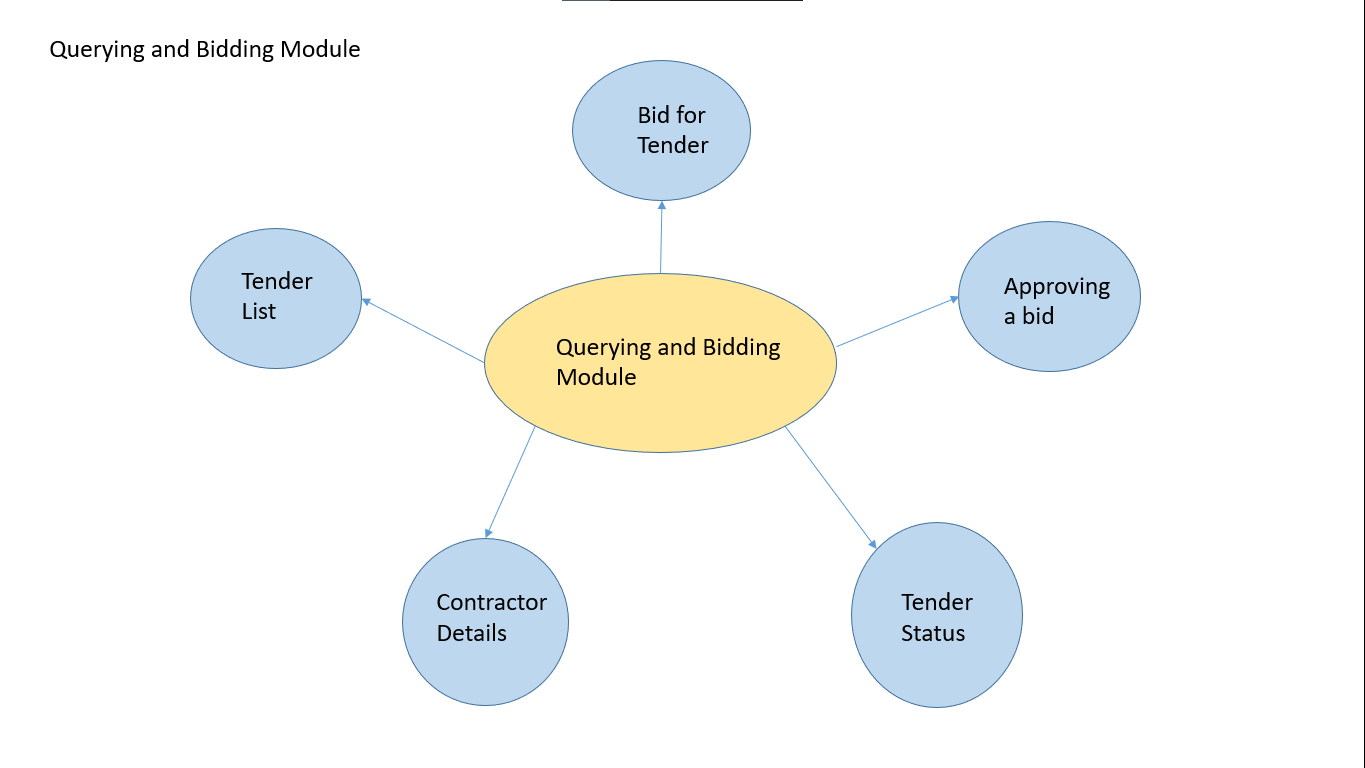












4.0 System Design

4.1 Proposed design

  4.1.1 Login & Registration Module

* A user tries to Login, if not signed up, Register as a new User.
* Admin gets AdminView, whereas Naïve User gets NormalView.

  4.1.2 Administration Module

* An Admin can view all Contractor Details.
* Admin needs to generate the Tender Report
* Admin needs to Approve a Tender by checking the constraints like BidAmount, MinQuote..etc.
* Admin can create new FeedBack QnAs.
* Whenever a user reports some issue in HelpDesk, Admin needs to resolve the issue.

  4.1.3 Contractor Module

* A user can Create a new Tender.
* A user can view all available Tender’s List, Bid for a tender, Download the report of the tender.
* A user can view their BidStatus(Approved/Rejected).

  4.1.4 Tender Management Module

* A user while creating should give the fromDate and ToDate of the tender.
* The TenderStatus needs to reflecting Realtime on Contractors view and database.
* Tender Creation should be Approved by Admin, only if there are valid Documents submitted from User’s Side.
* A Tender Report needs to be generated, downloadable to any Contractor.

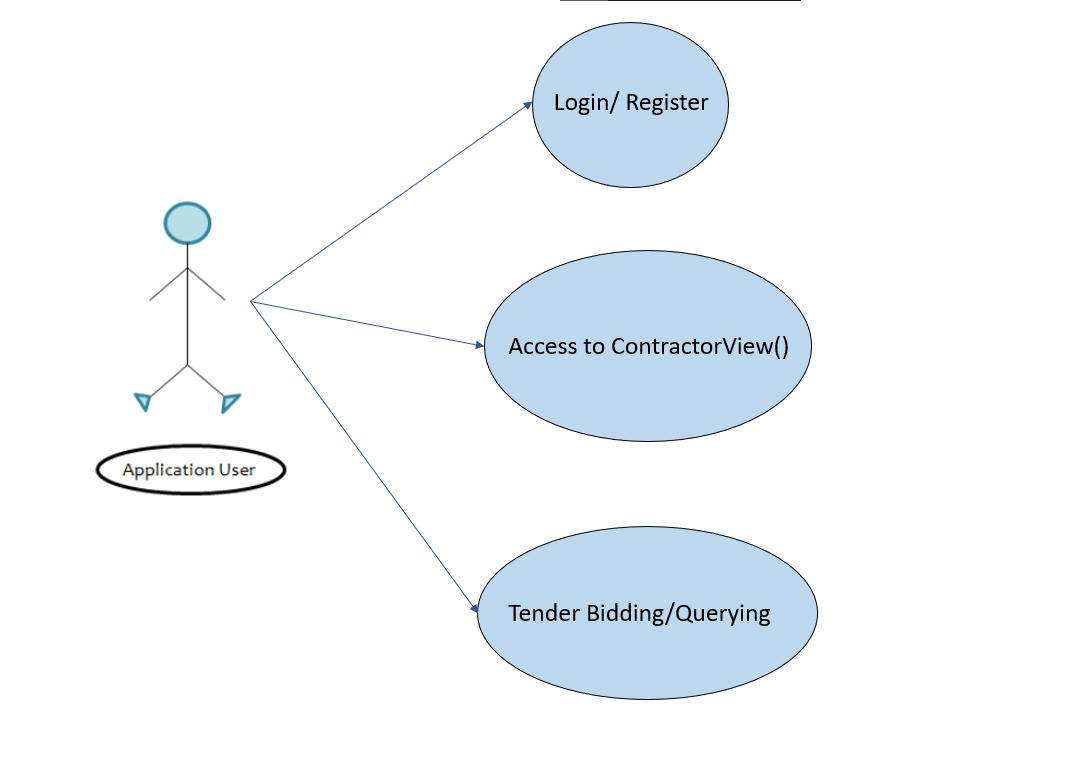
 4.1.5 User Service Module

* A user can Retrieve their UserId or Password using, a Secret Question Validation, submitted by them during registration.
* A user can reset their password.
* A user can report any issue through HelpDesk, they will get a Resolvent from Admin.
* A user can provide FeedBack, Reviews from his User Experience.

 4.1.6 Querying & Bidding Module

* A user can Bid for a tender, through TenderList.
* Admin can Approve a bid, the same needs to be reflected on contractor’s view.
* TenderStatus needs to be Realtime update.
* Admin can view all the Contractors Details.
* Admin and User both can view TenderList.

4.2 Component inventory



4.2.1 Microservices

1. **Login & Registration Microservices:**

**Model Class -**

* VenderBean
* BidderBean

**Dao Class -**

* Bidder Dao
* Vender Dao
* Bidder DaoImpl
* Vender DaoImpl

**Service Class-**

* LoginSrv
* LogoutSrv

1. **Administration Microservices:**

**Service Class-**

* AcceptBidSrv
* RejectBidSrv

1. **Contractor Microservices:**

**Model Class -**

* VenderBean
* BidderBean
* TenderStatusBean

**Dao Class -**

* Bidder Dao
* VenderDao
* BidderDaoImpl
* Vender DaoImpl
* Tender Dao
* Tender DaoImpl

**Service Class-**

* Bid Tender Srv
* Create Tender Srv
* Update Profile Srv
* Change Password Srv

1. **Tender Management Microservices**

**Model Class -**

* Tender Bean
* Tender Status Bean

**Dao Class -**

* Tender Dao
* Tender DaoImpl

**Service Class-**

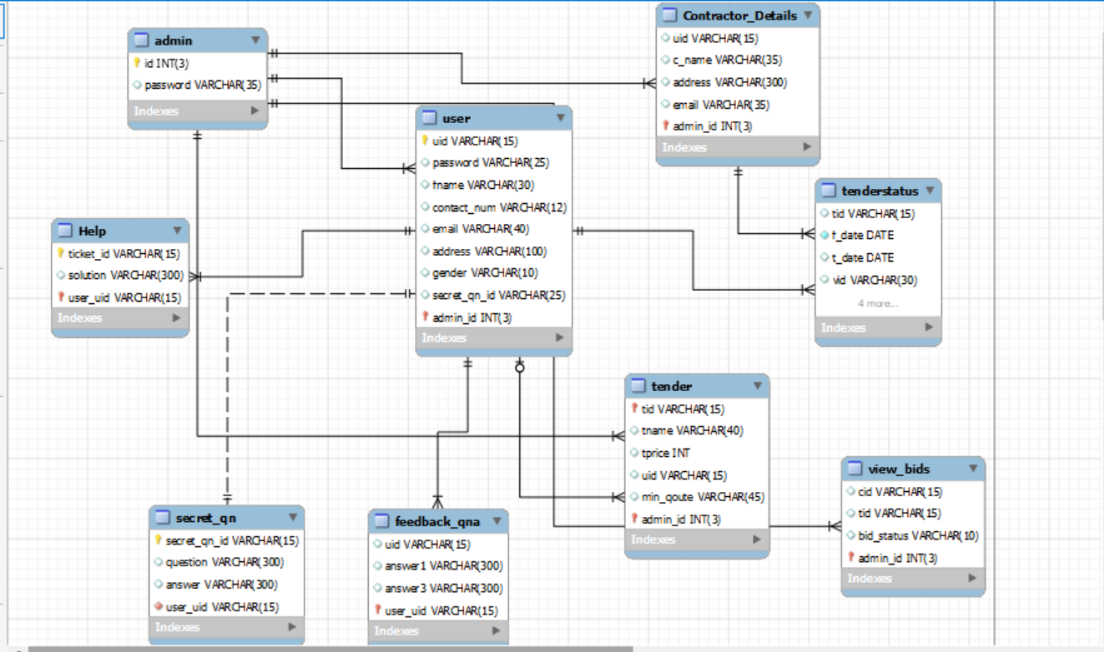
* Bid Tender Srv
* Create Tender Srv
* Accept Bid Srv
* Reject Bid Srv

1. **Utility Classes:**

* DBUtil
* IDUtil

5.0 Database Design

5.1 Data Base Model Design



5.2 Tables Structure

* Admin Table

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Field** | **Type** | **Null** | **Key** | **Default** |
| Login ID | Varchar (15) | NO | Primary Key | Admin |
| Password | Varchar (15) | NO | Unique Key | Admin |

* User Table

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Field** | **Type** | **Null** | **Key** | **Default** |
| First\_Name | Varchar(15) | NO | NA | Null |
| Last\_Name | Varchar(15) | YES | NA | Null |
| DOB | Date | NO | NA | Null |
| Email\_ID | Varchar(15) | YES | NA | Null |
| Gender | date | NO | NA | Null |
| Contact\_ Number | Varchar(15) | NO | NA | Null |
| User\_ID | Varchar(15) | NO | Primary Key | Null |
| Password | Varchar(15) | NO | NA | Null |
| Secret\_Qn\_ID | Varchar(15) | NO | NA | Null |

* Tender Table

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Field** | **Type** | **Null** | **Key** | **Default** |
| Tender\_ID | Varchar(15) | NO | Primary Key | Null |
| Tender\_Name | varchar(40) | YES | NA | Null |
| Tender\_Type | varchar(20) | YES | NA | Null |
| Tender\_price | int | YES | NA | Null |
| Tender\_Desc | varchar(300) | YES | NA | Null |
| Tender\_Deadline | date | YES | NA | Null |
| T\_Location | Varchar(70) | YES | NA | Null |

* Tender Status Table

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Default** | **Key** |
| Tender\_ID | Varchar(15) | Null | Foreign Key |
| Bid | varchar(15) | Null | NA |
| Tender\_Status | varchar(15) | Null | NA |
| Tstatus\_Id | Varchar(15) | Null | Primary Key |

* Help Table

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Field** | **Type** | **Null** | **Key** | **Default** |
| Ticket\_ID | Varchar(15) | NO | Primary Key | Null |
| Issue | Varchar(50) | NO | NA | Technical Issue |
| Description | Varchar(200) | NO | NA | NA |
| Response | Varchar(200) | NO | NA | NA |

* FeedBack Table

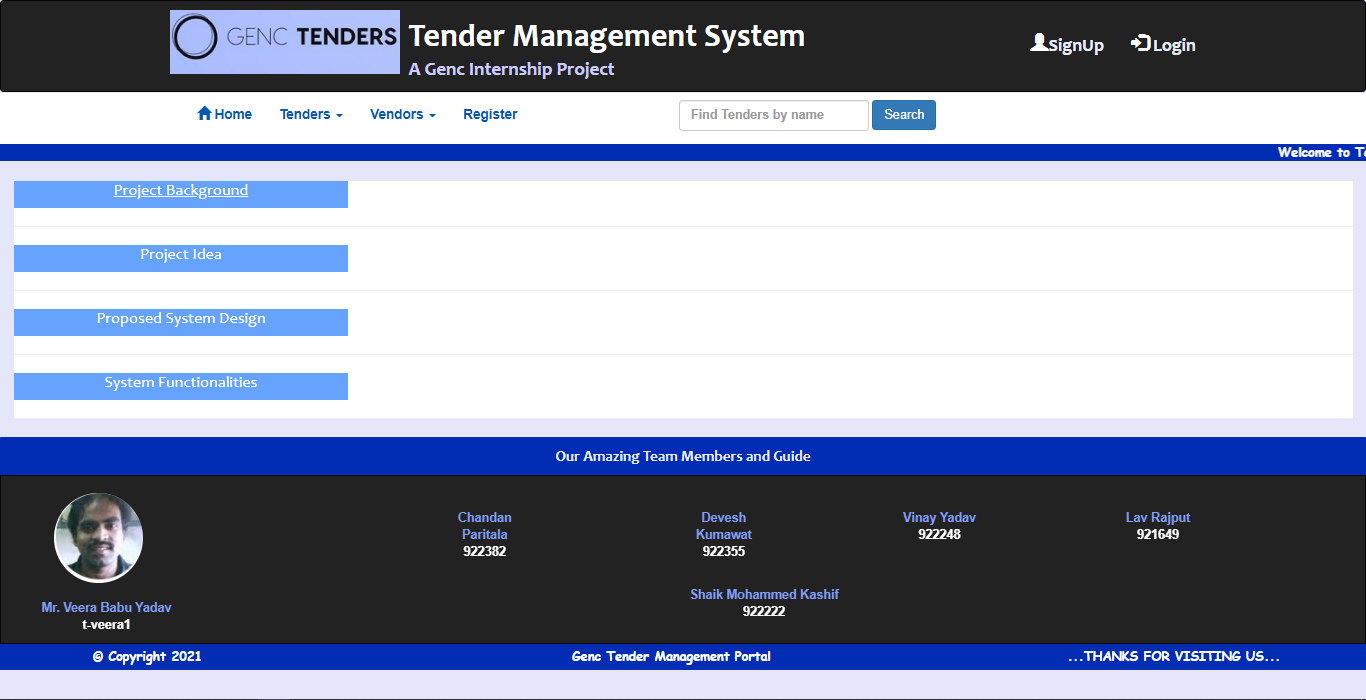
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Field** | **Type** | **Null** | **Key** | **Default** |
| User\_ID | Varchar(15) | NO | Foreign Key | Null |
| F\_ID | Varchar(50) | NO | Primary Key | Null |
| Description | Varchar(200) | NO | NA | NA |
| Review | Varchar(200) | NO | NA | NA |

* Secret\_Qn Table

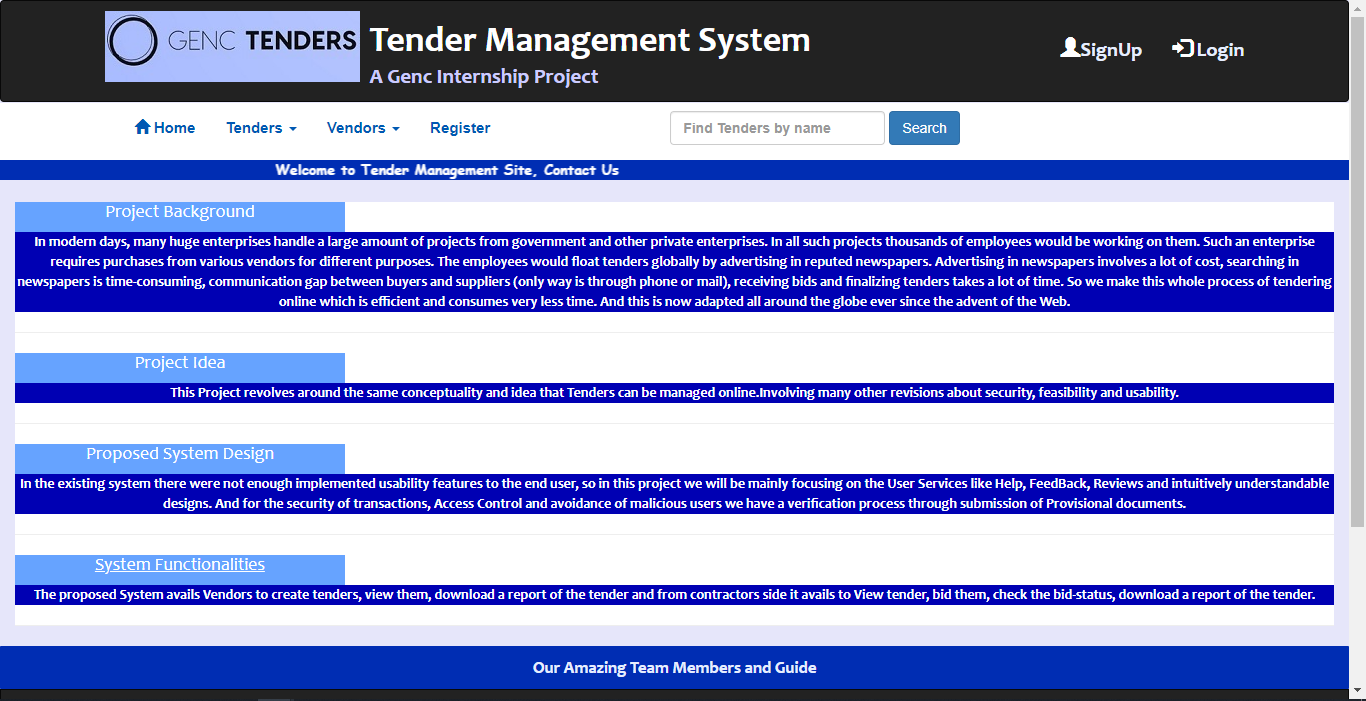
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Field** | **Type** | **Null** | **Key** | **Default** |
| Secret\_Qn\_ID | Varchar(15) | NO | Primary Key | Null |
| User\_ID | Varchar(50) | NO | Foreign Key | Null |
| Question | Varchar(200) | NO | NA | NA |
| Response | Varchar(200) | NO | NA | NA |

6.0 UI Design

**6.1. Home Page**

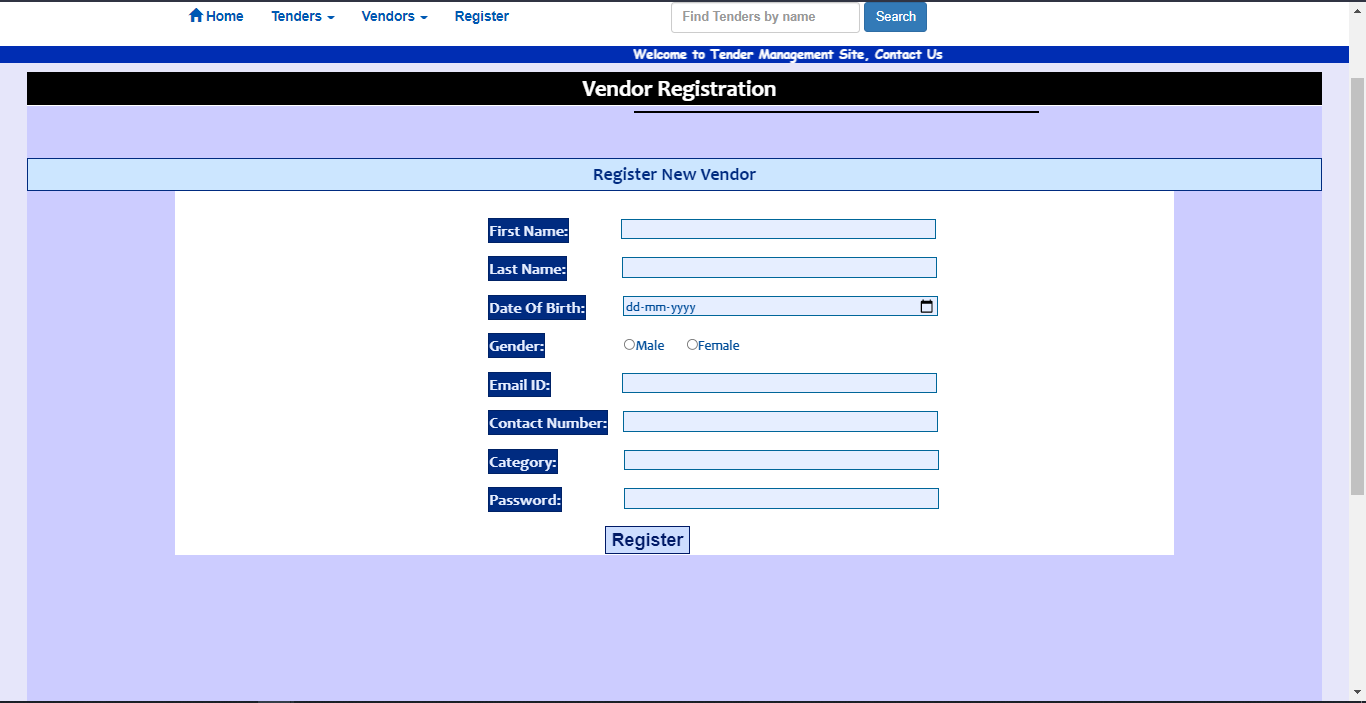


Upon Selecting each section, the Hidden text can be seen.



**6.2. Register Page**

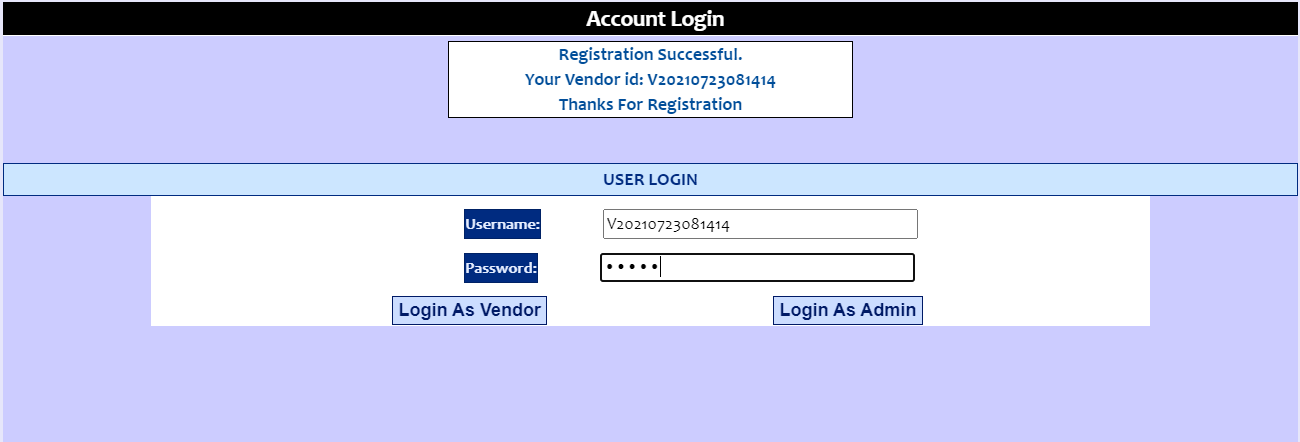
Upon selecting to Register, User can Register as a new Vendor.



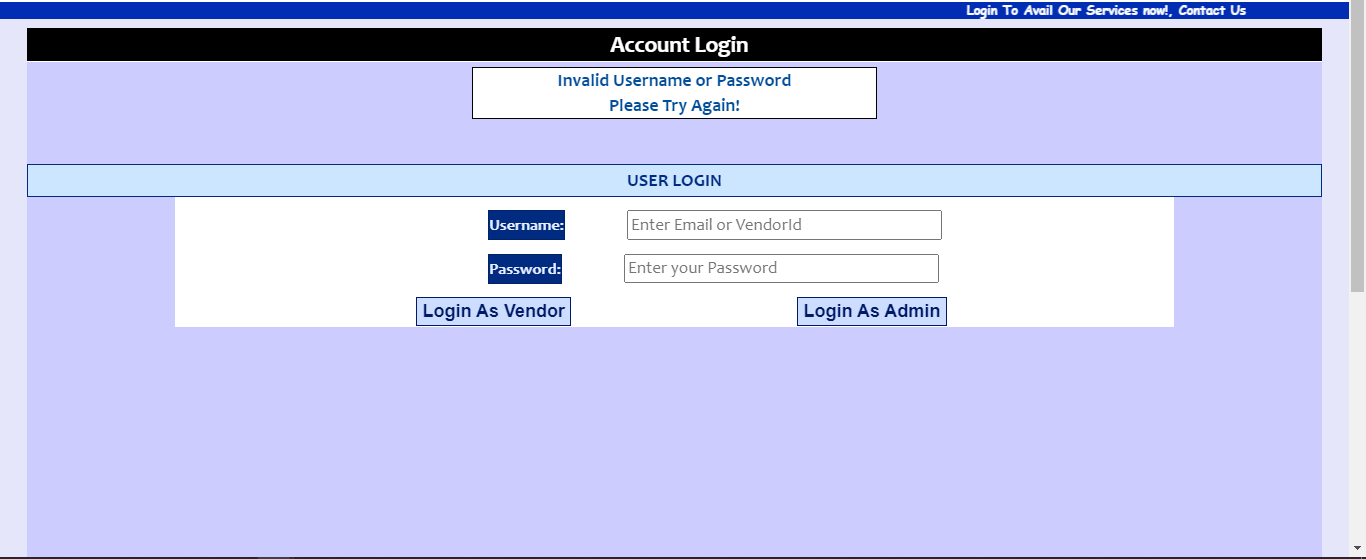
After, Registration User will be redirected to LoginPage Automatically.



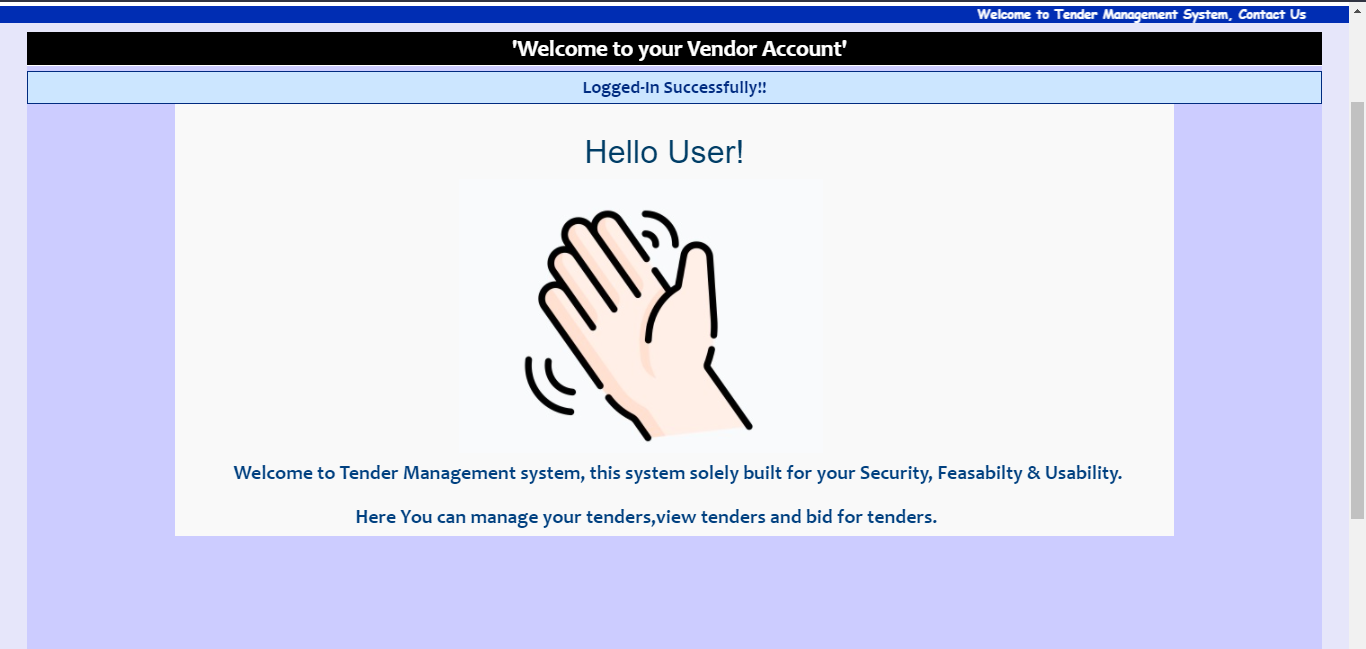
**6.3. Login Page**



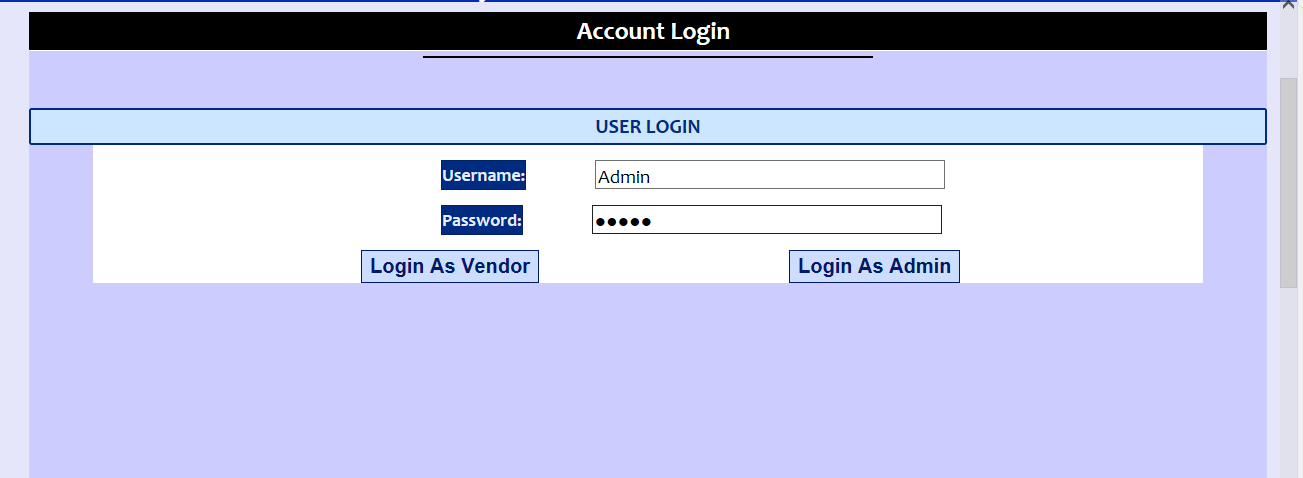
Invalid Login Details,

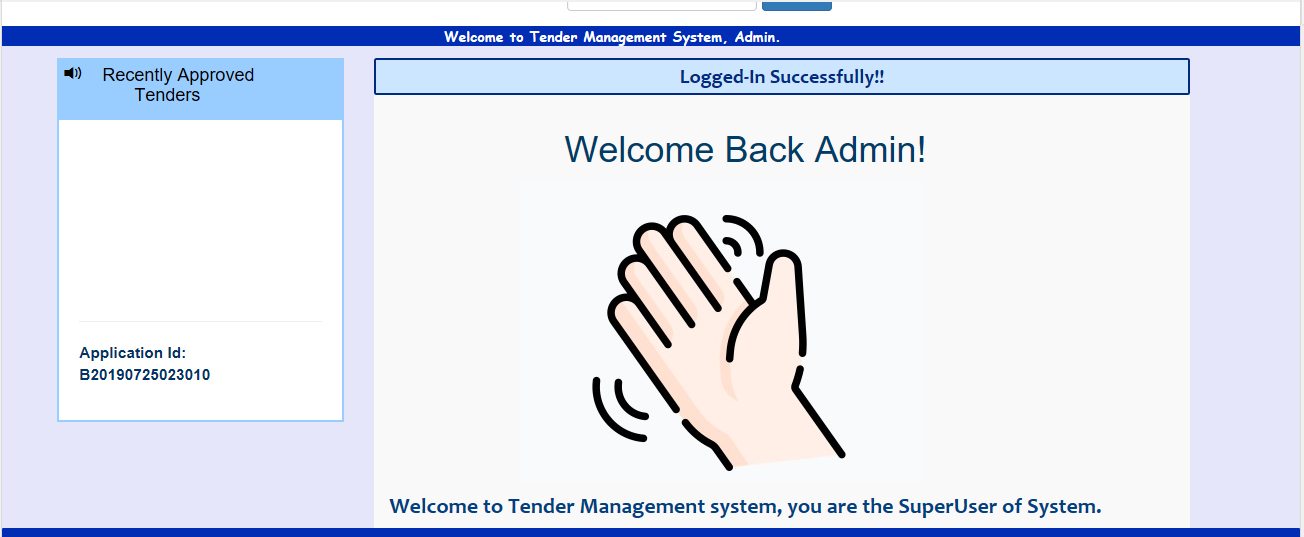


After Entering Valid Credentials,



Admin Login,



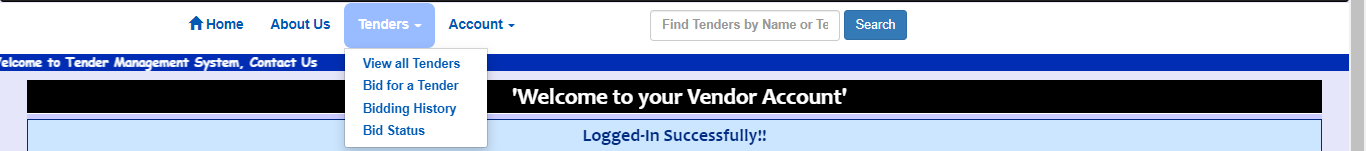


**6.4. Contractor View**

1. Contractor Menu:



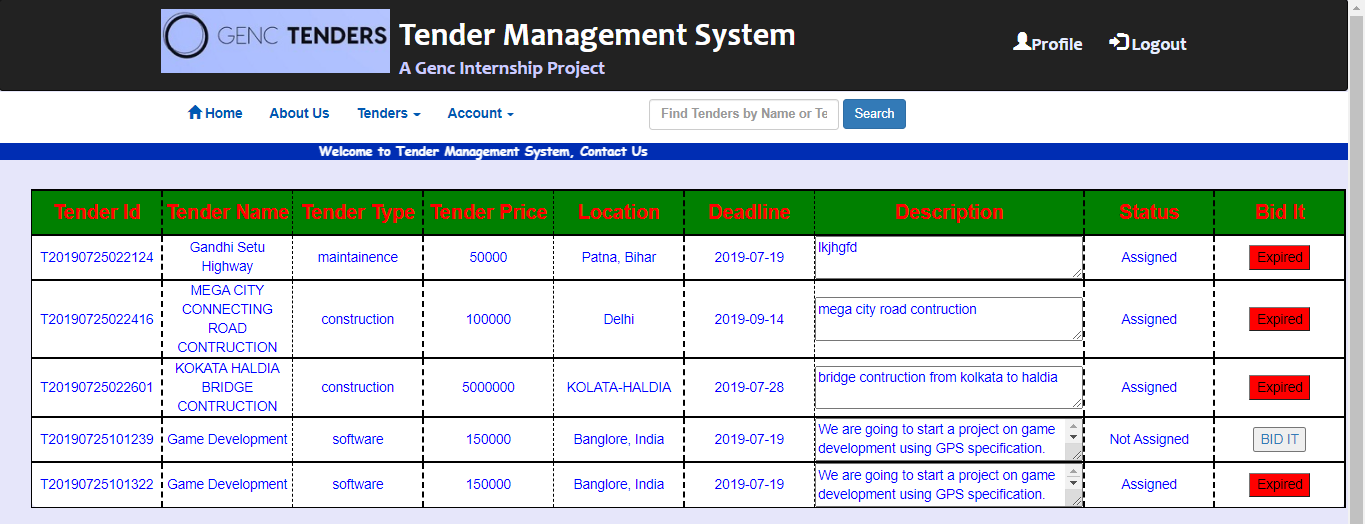
1. Tenders Section:



**View all Tender,**



**Bid for a Tender,**



User can only bid Unassigned Tenders; Assigned Tenders are shown as Expired.

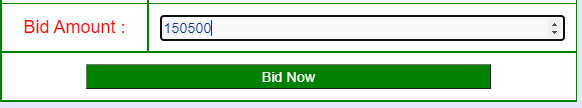
Click ON Bid It;



User needs to bid an Amount Greater than, the BasePrice given by Vendor.



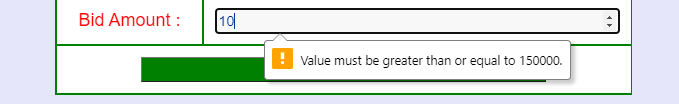
Enter the Bid Amount and Cick Bid Now.



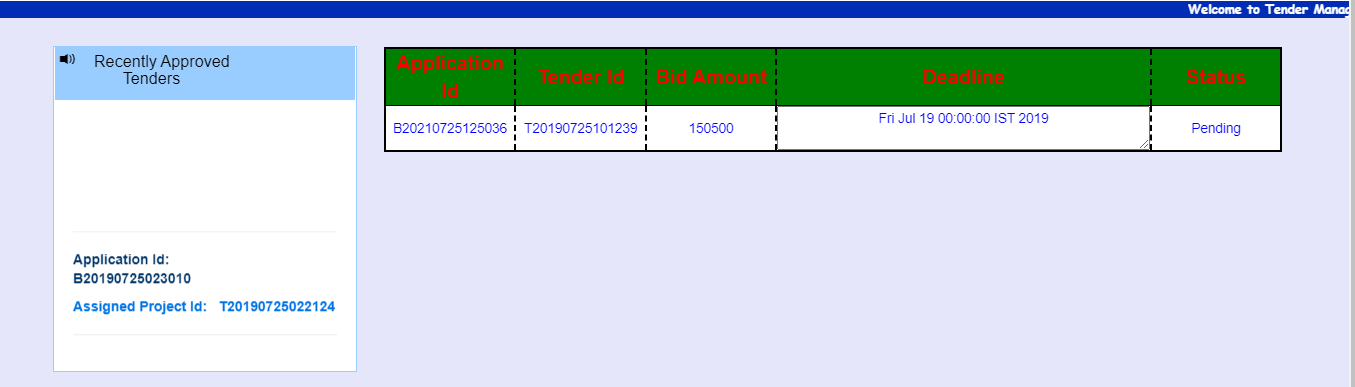
User now Successfully Placed a bid.



If user Bids a Low Amount than BasePrice,



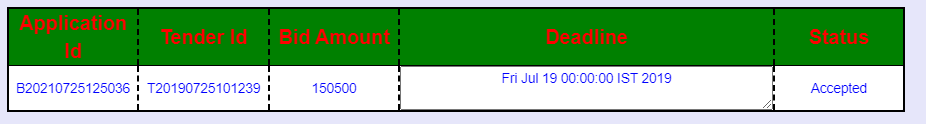
**Bidding History of a User,**

****

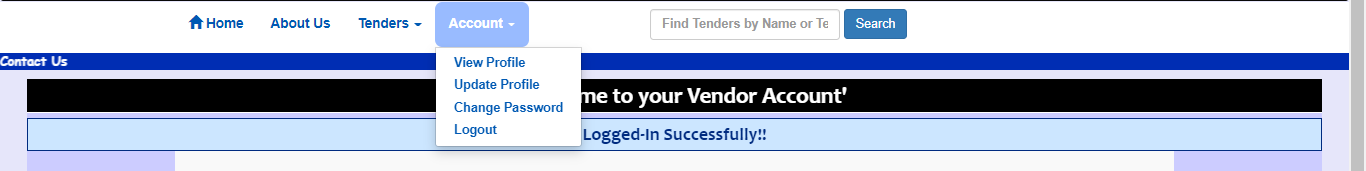
Status is By Default Pending Untill the Admin Accepts/Rejects the bid.

**Bidding Status of a User,**

Status is Updated through DB, when Admin Accepts/Rejects the bid.



1. Account Section:



**View Profile,**

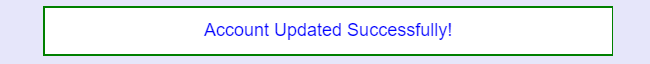
****

Click to Update User Profile,

**Update Profile,**

Update details and Verify through Password,

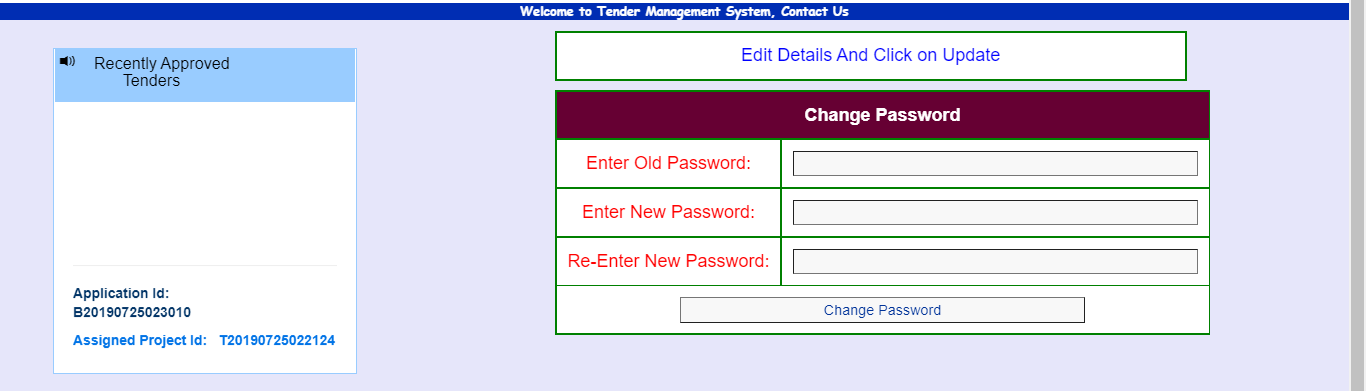
****

****

Check Updated Profile.

****

**Change Password,**

****

The Re-Entered Password & new Password should match.



Verify the Old Password, and enter a new Password.

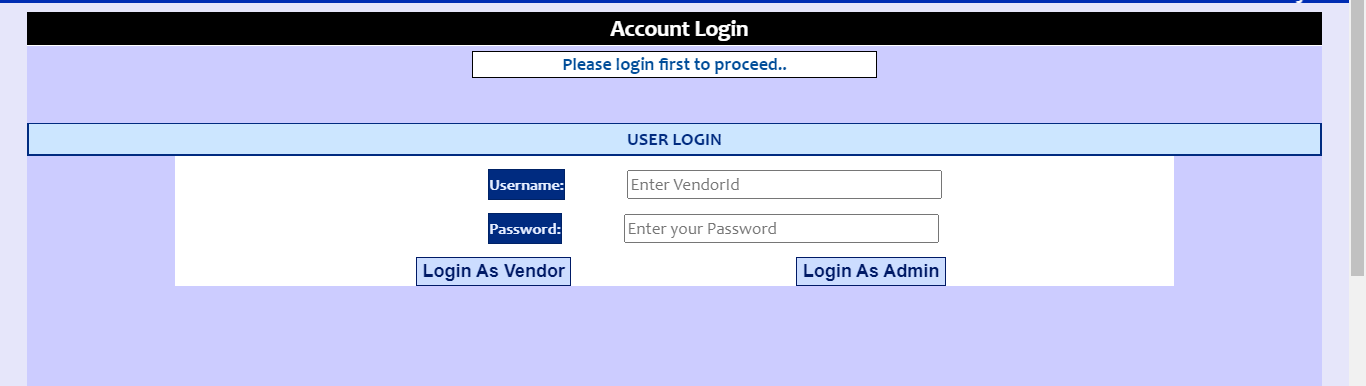
****

**Logout,**

****

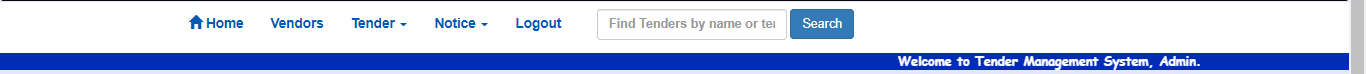
1. Login Failed:

User needs to Login Successfully, to enter through any of the Services provided by the system, else,

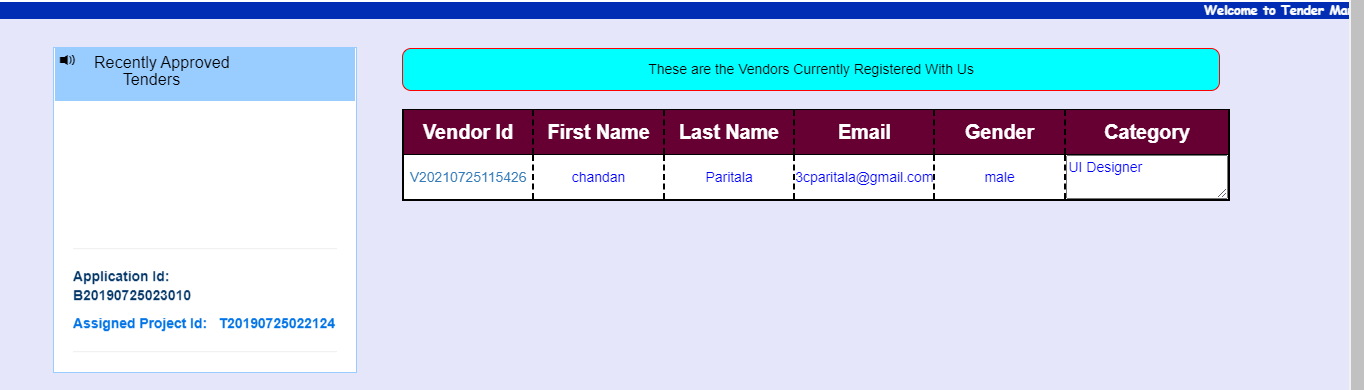


**6.4. Admin View**

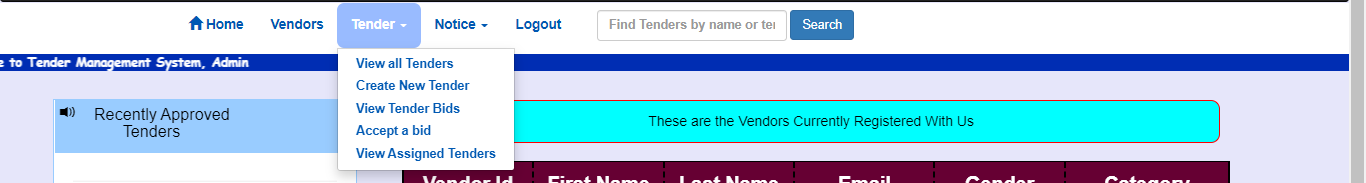
1. Admin Menu:



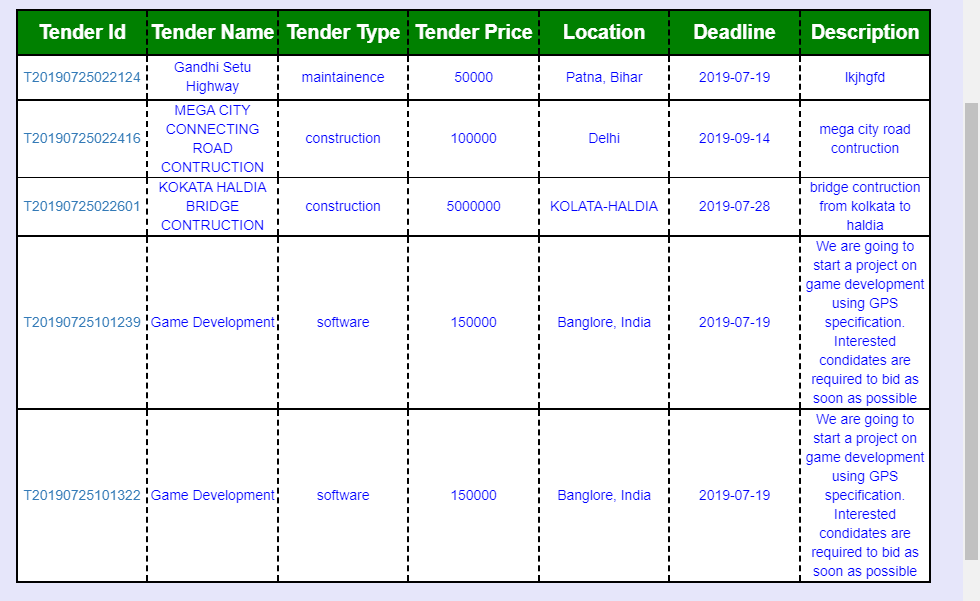
1. View All Vendors:



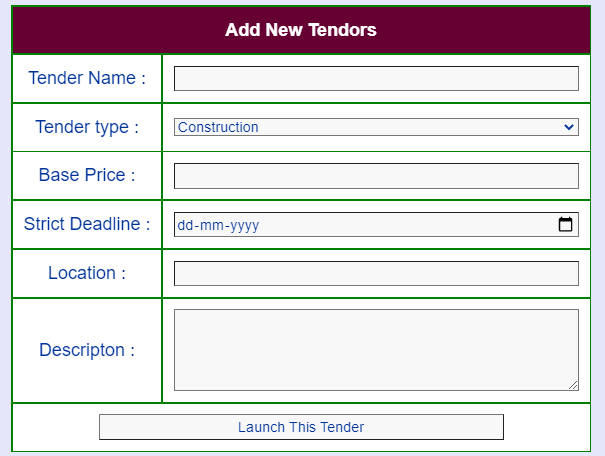
1. Tenders Section:



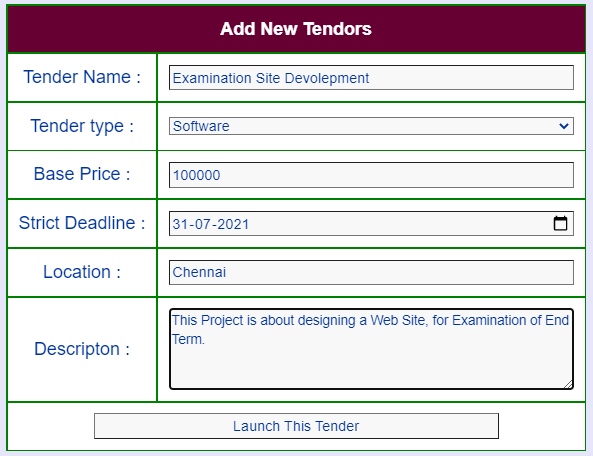
**View All Tenders,**

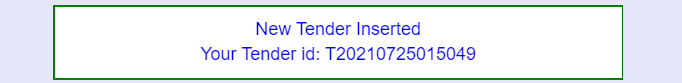
****

**Create new Tenders,**

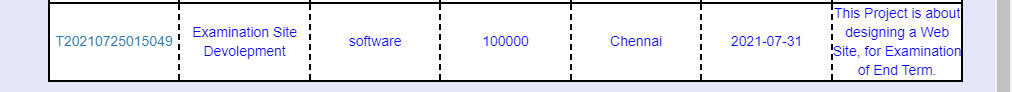
****

Fill the Details of the new Tender, Launch the Tender.

****

****

Tender List Updated,

****

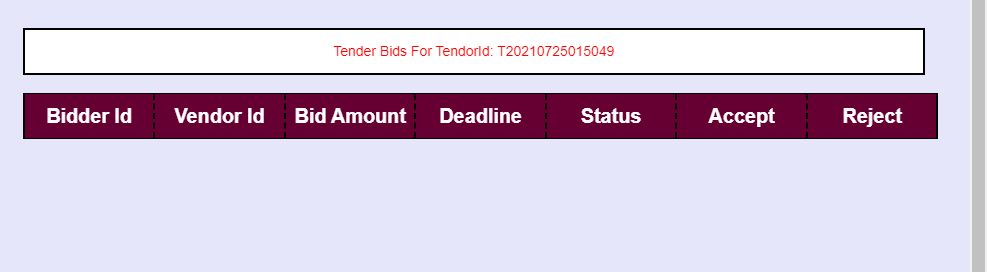
**View the Bids of Tenders,**

****

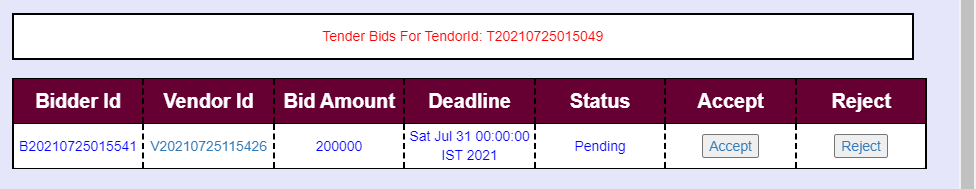
Select View Bids to show the Bids,

****

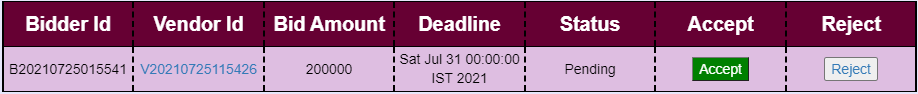
Empty List if There were no Bids, from Bidders.



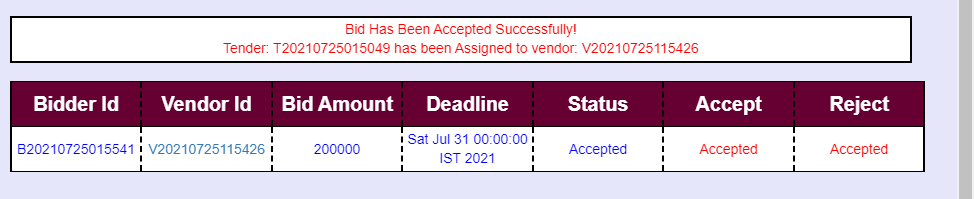
**Accept/Reject Bids of Tenders,**



Click on either Accept or Reject,



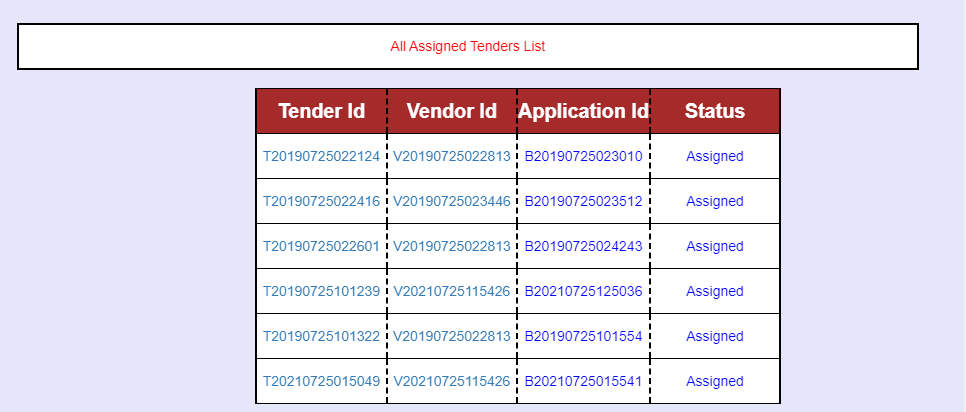
After Accepting a Bid for a tender, the Tender is now Assigned to The Bidder Mentioned with the ID, and now NO OTHER Bidder can bid this Tender Furtherly.



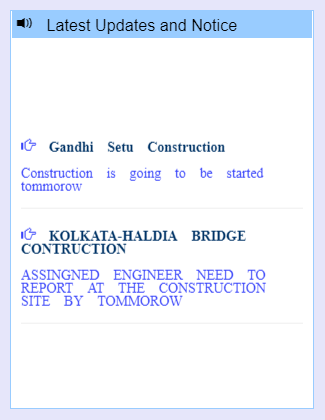
**View Assigned Tenders,**

After Accepting a Bid for a tender, the Tender is now Assigned to The Bidder Mentioned with the ID, and now NO OTHER Bidder can bid this Tender Furtherly.

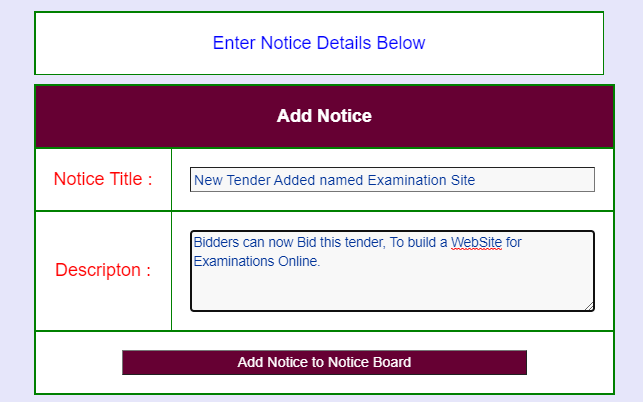
All the Assigned Tenders are now Expired and Admin can View them.

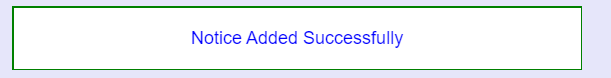


1. Notice Section:



**Add a new Notice,**

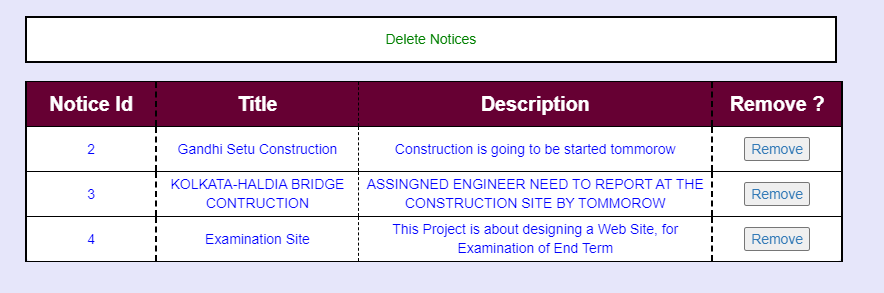
****

****

Notice Added to Notice List,

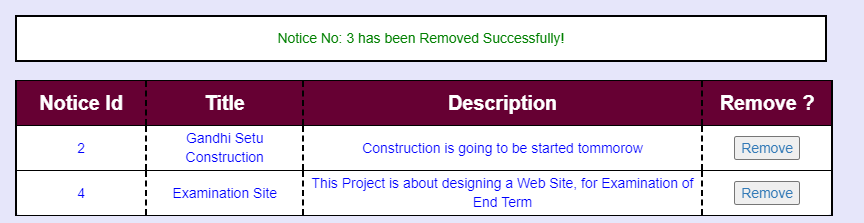
****

**Remove a Notice,**

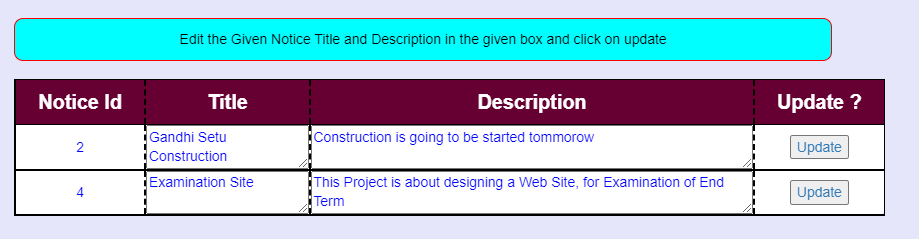
****

Click on Remove to delete the notice,

****

****

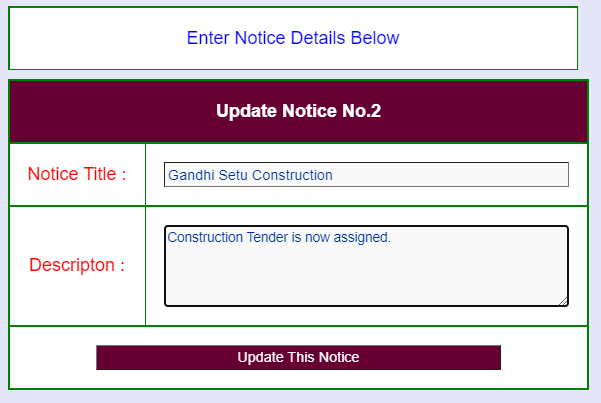
**Update a Notice,**

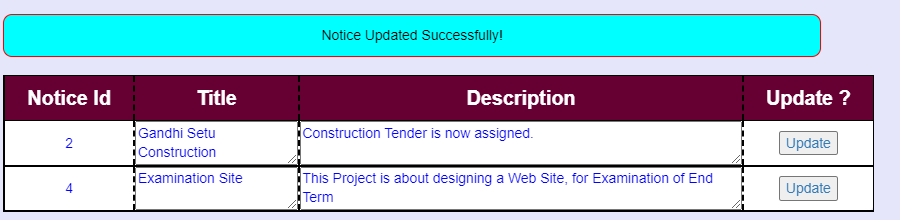


Click on Update to update the notice,

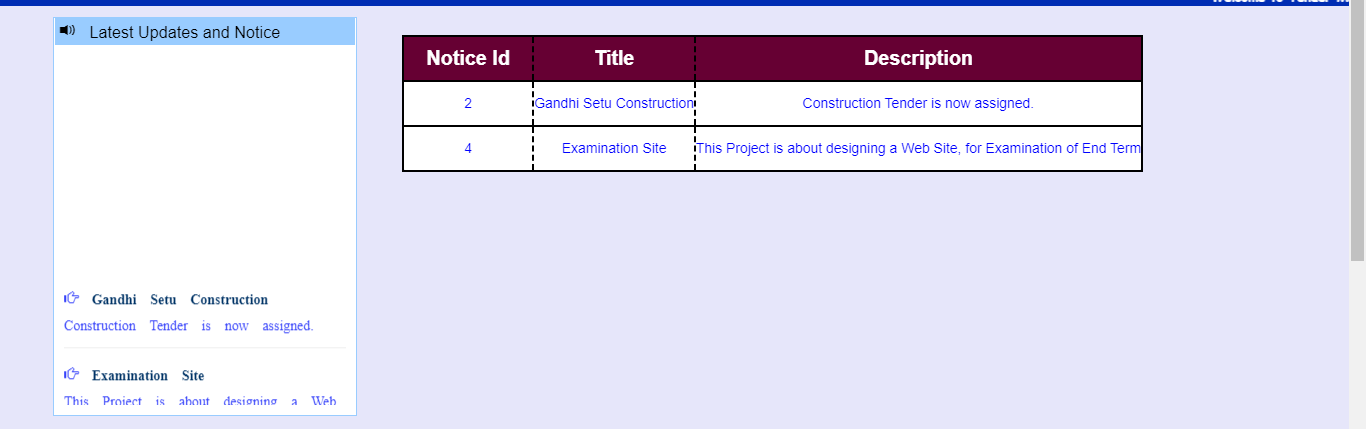


update the notice,



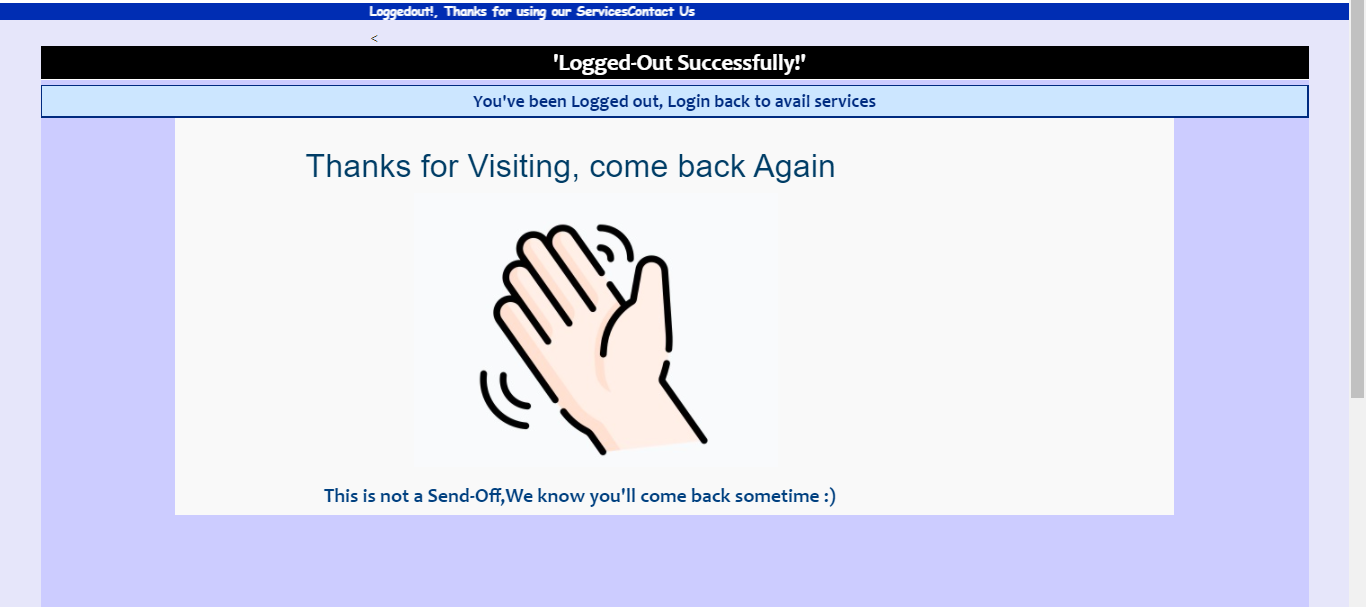


**View all Notices,**

****

**6.5. Log Out Page**

Both Admin & User have a same send-off page.

****

7.0 Terms & Conditions

Interns shall be solely responsible for all its acts and omissions under this program. Interns will comply at all times with all applicable laws. Interns shall not use Cognizant’s name, logo and trademark in any promotional materials or other communications with third parties without the prior written consent of Cognizant. Any materials used by interns in relation to program will not infringe the copyrights, trademarks, patents, trade secrets or other intellectual property rights, privacy or similar rights of any person or entity. Interns agrees not to post, draw, make, display any content that is threatening, libelous, obscene, defamatory, abusive, pornographic, or advocates/encourages any conduct that could constitute a criminal offence or give rise to any civil liability. Cognizant its associates’ personal details including but not limited to name, address, contact number shall not be shared or forwarded to any third party, without prior written consent of Cognizant, its associates. All intellectual property provided by Cognizant, as part of program shall be owned exclusively by Cognizant. Intern shall indemnify, defend and indemnify Cognizant its associates, officers, directors from and against any claims, demands, loss, damage, liability, causes of action, judgments, or costs and expenses of every nature (including attorney’s fees and expenses) incurred by Cognizant based on any claim that any breach of terms and conditions of this program

8.0 Change Log

*Please note that this table needs to be maintained even if a Configuration Management tool is used.*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Version Number** | **Changes made** | | | |
| V<n.n> | *<If the change details are not explicitly documented in the table below, reference should be provided here>* | | | |
| Page no | Changed by | Effective date | Changes effected |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |