**USER MANAGEMENT DOCUMENT**

| MULTIPLE USER MANAGEMENT DOCUMENT | 24-November-2021 | |
| --- | --- | --- |
| **User Management** | |  |

| **REVISION HISTORY** | | | |
| --- | --- | --- | --- |
| **REV** | **Description of Change** | **Author** | **Effective Date** |
| 1.0 | Creation of the Document | Dipali Laxman Nagtilak | 24-November-2021 |
| 1.1 | Reviewed By |  |  |
| 1.1 | Approved By |  |  |

**INDEX**

[Introduction](#_30j0zll) **3**

[Purpose](#_1fob9te) 3

[About User Management](#_3znysh7) 3

[Prerequisite](#_2et92p0) **3**

[Basic Requirement](#_tyjcwt) 3

[Screenshots](#_3dy6vkm) 4

Drive 8

[MySQL workbench setting for bulk import](#_4d34og8) 6

[Application Code](#_1ksv4uv) 6

# **Introduction**

## **Purpose**

The purpose of the document is to describe the process of Importing Multiple Users to the database using a developed application.

## **About User Management**

Developed an application to be able to upload a spreadsheet of users and their information and to be able to import users to make changes easily.

# **Prerequisite**

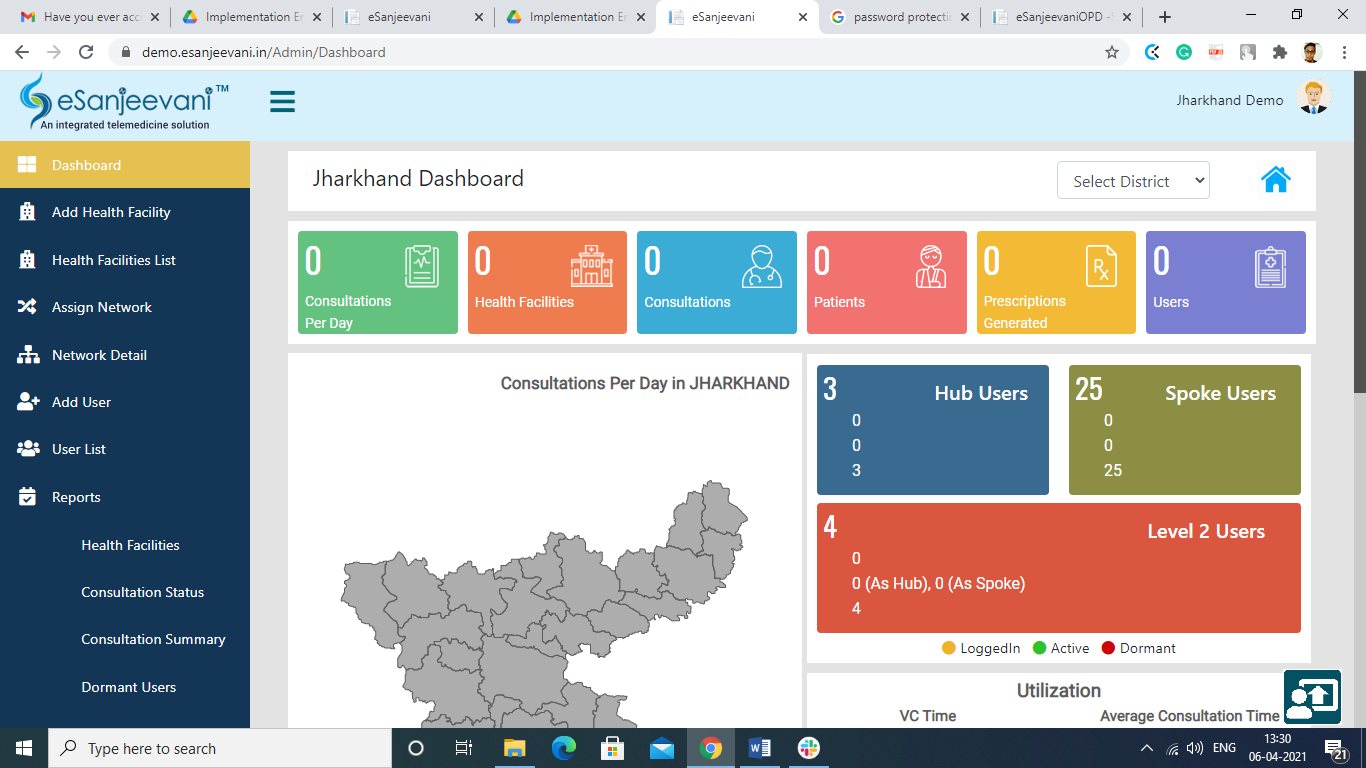
## **Required Changes**

| **Sl. No.** | **Category** | **Software Title** |
| --- | --- | --- |
| **1** | Packages | FluentValidation.AspNetCore(9.5.4),CSharpFunctionalExtensions(2.26.1) |
| **2** | Startup.cs file changes | 1. Add services.AddFormHelper(); under ConfigureServices() method. 2. Add app.UseFormHelper(); under Configure() method. |
| **3** | \_Layout.cshtml | //Add below things. <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/toastr.js/latest/css/toastr.min.css" /> |
| **4** | site.css | //Add below things  .input-validation-error {  border: 1px solid #ff0000 !important;  }  .bulkupload.table td, .table th {  min-width: 200px;  } |

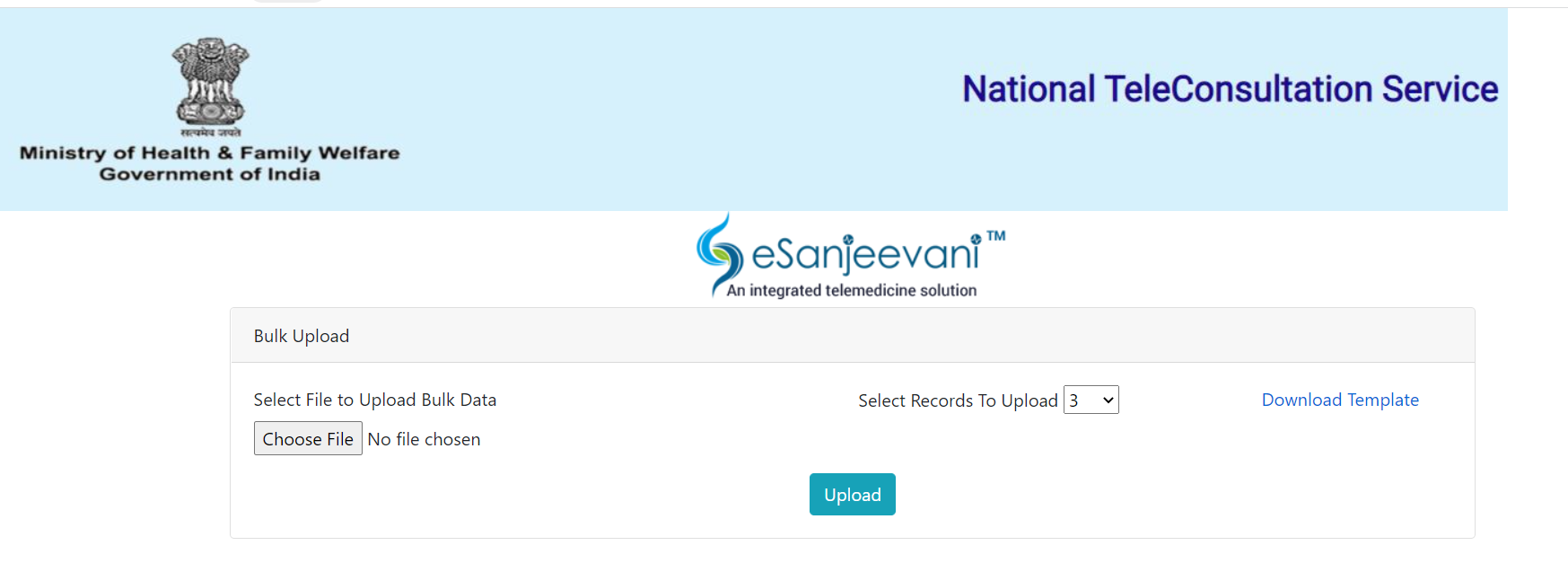
## **Screenshots**

Admin users should login to HWC portal using their credential and after successful login admin dashboard will land.

If admin want to import the users to HWC portal then user will click on bottom right corner . And it will redirect to another application for import users using a predefined excel sheet.



User will navigate to Bulk import on below screen

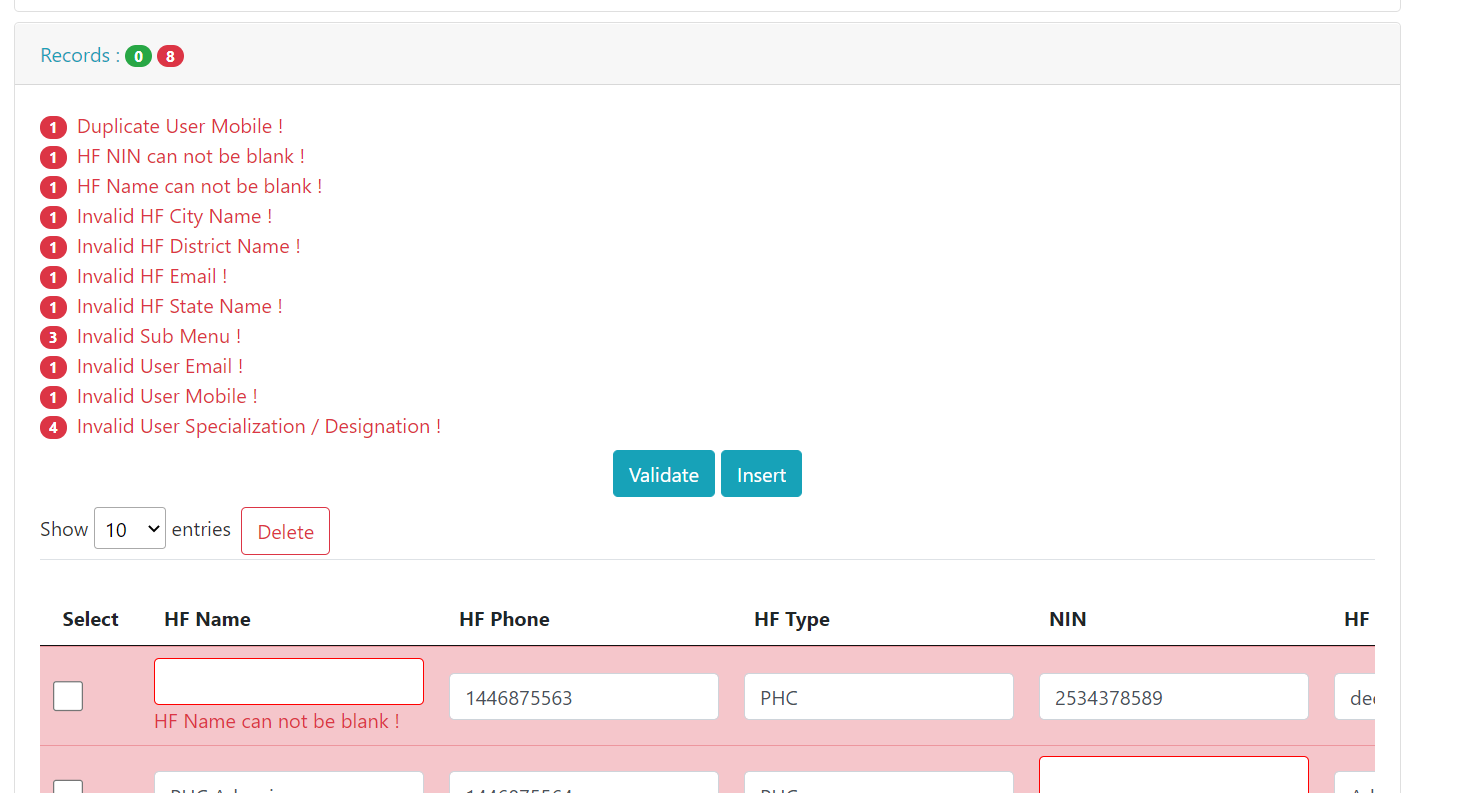


Click on “Download Template” and the user will get the template for the bulk import.Fill the data in the downloaded template.

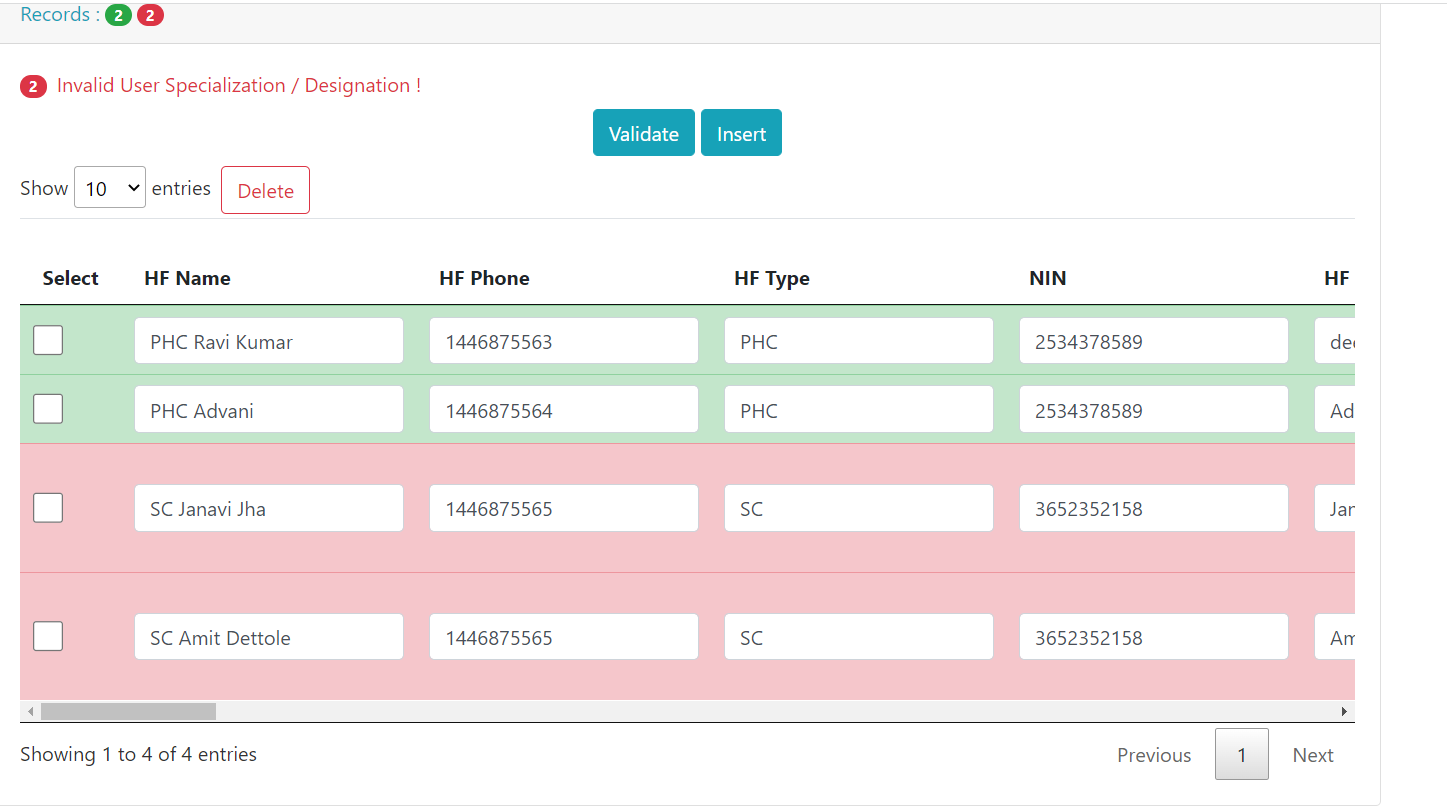
Click on “Choose File”. Select the filled XL sheet file.

Click on the “Select Records To Upload” it will select the number of records from the XL to upload.e.g if in your XL sheet there are 10 records if you select 3 in “Select Records To Upload” the 3 records will be selected to import in the database.

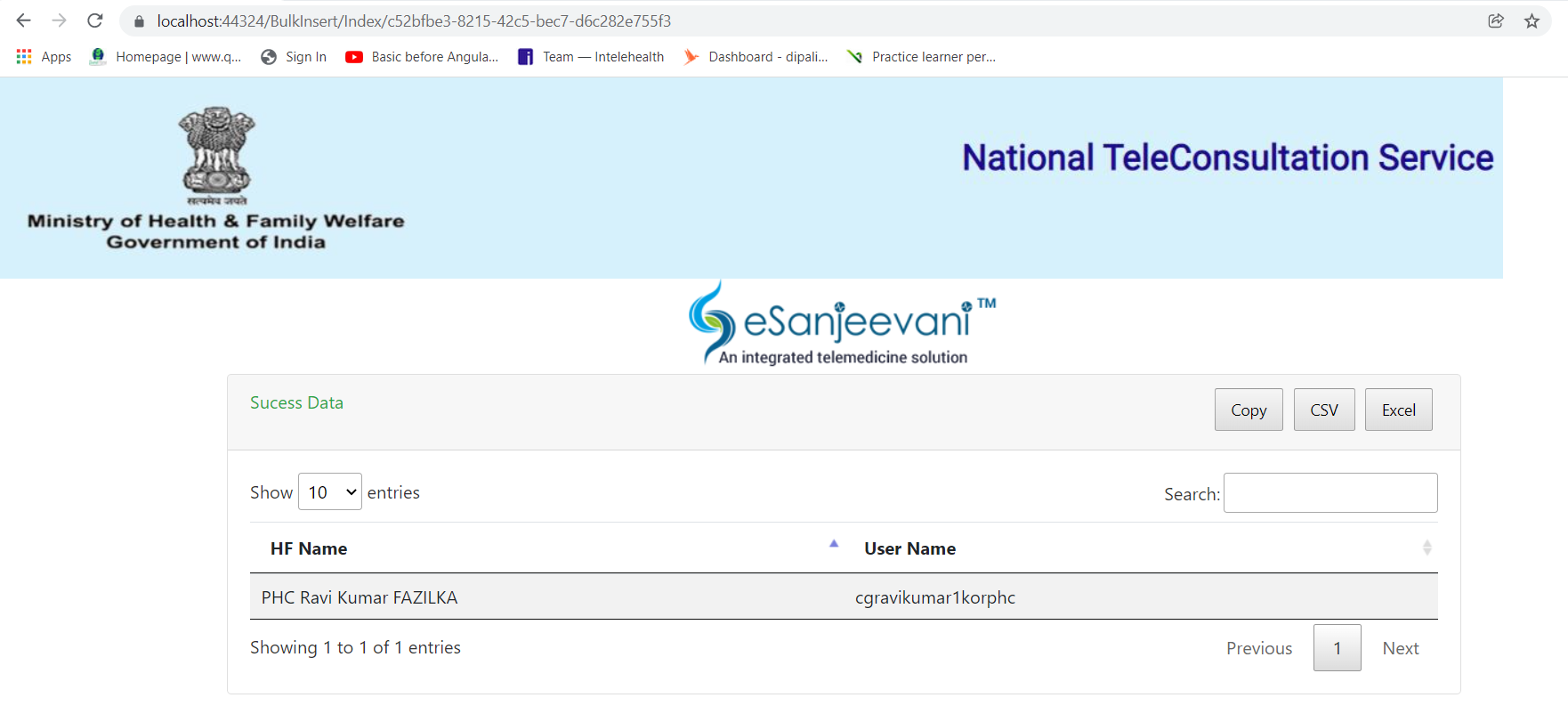
Below screen will appear where the user has facility to edit and delete records.

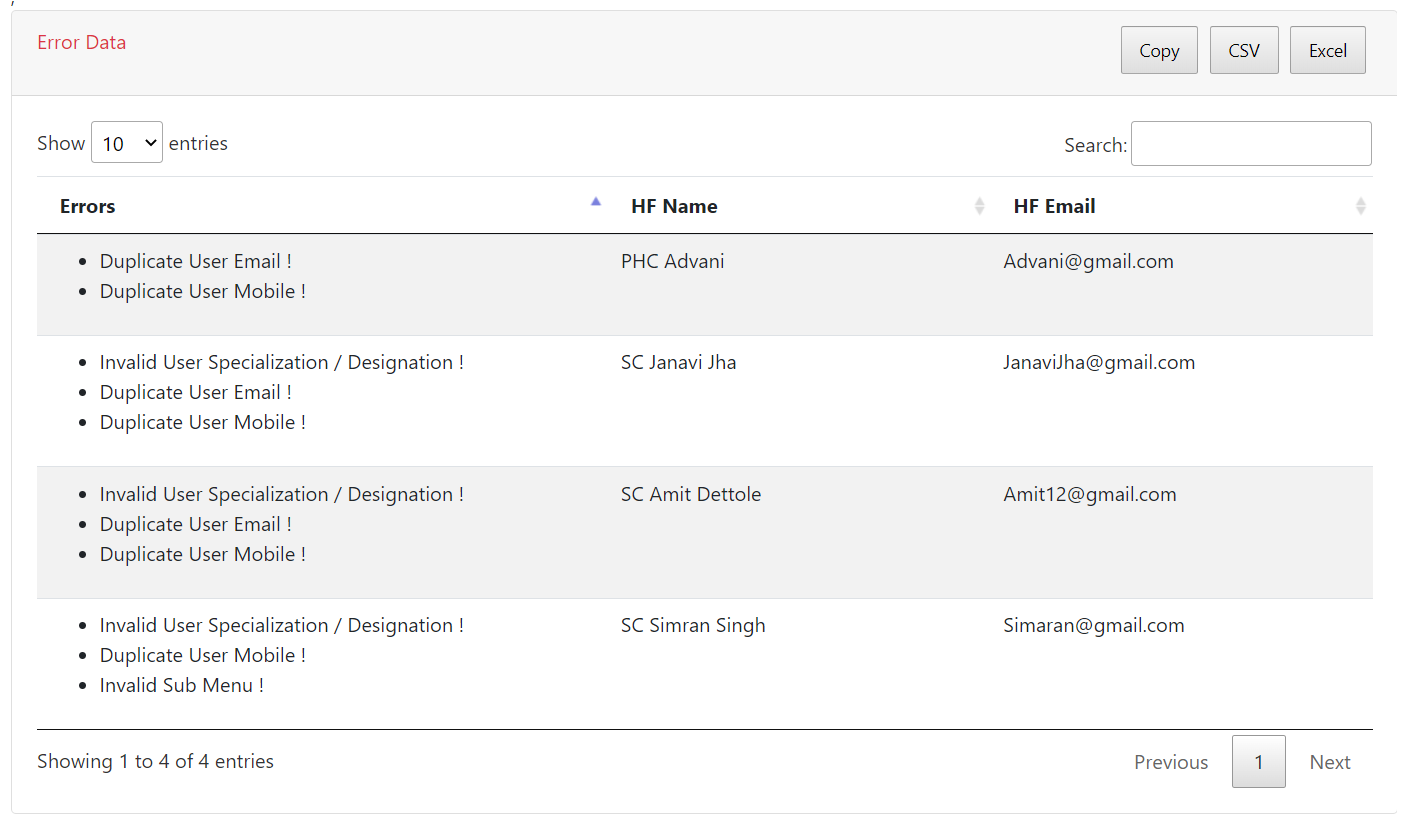


After validating the data, the above screen looks like the below screen. Records which are validated will appear in the green section and those which are not validated will appear in the read section.



After inserting the data successfully in the database below the screen will appear. Where the user has facility to export the data in CSV,Excel and Copy to clipboard format.





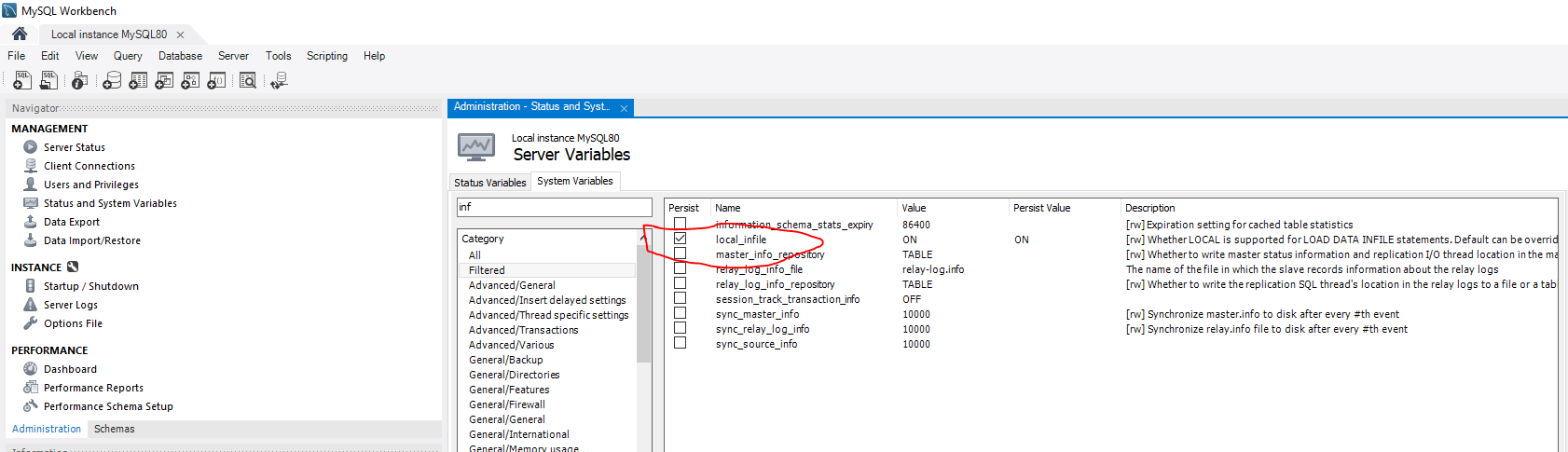
## **Drive**

URL :<https://drive.google.com/file/d/1_Ln6lDpJ0AIO-avotFw2CxWtZKYog83A/view?usp=sharing>

## **MySQL workbench setting for bulk import**

eSanjeevani-Registration-Module SET GLOBAL local\_infile = true;

By doing below steps on server side Administration-->Server--->Status and system variable--->system variable-->Local Infile-->true refer below image for above step



## **2.5 Application Code**

Application code is divided into business,contract,domain,infrastructure,UserManagement(Web).

**2.5.1 Business**

1. In the business module we have defined validators and services.

**2.5.2** **Contract**

1. In the contract we have created interfaces for factory,Repository,User and Utility.

**2.5.3** **Domain**

1. In the domain we have created ViewModel for bulk import.
2. Defined all models.

**2.5.4 Infrastructure**

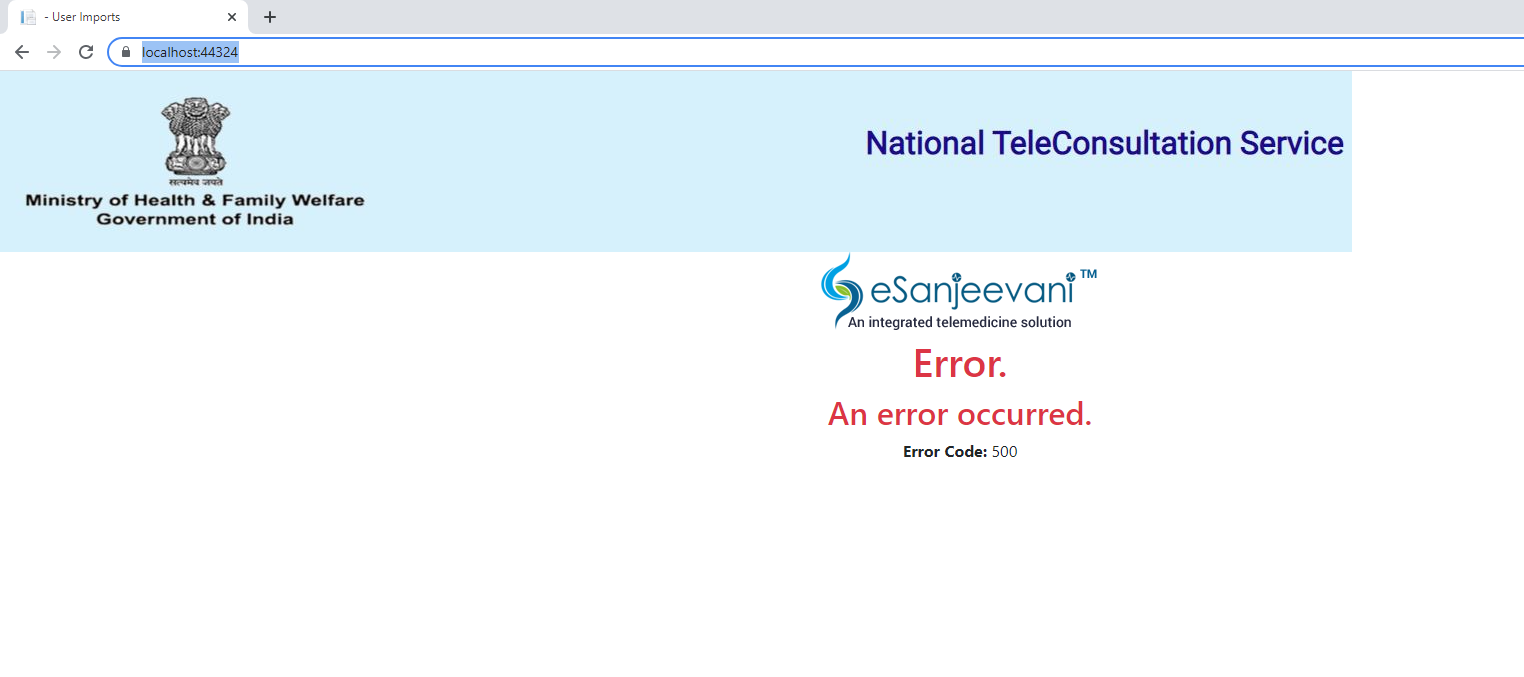
1. Defined Csv and Excel utility.
2. Created Mapping.
3. Created Repository.

**2.5.5 Web**

1. All controllers, models and views are present.

**2.6 Exception Handling and error logging**

1. We have created a user friendly and developer friendly screen to display the error.Below screen will appear in case of exception.



1. A Detailed log file has been maintained in UserManagement\UserManagement\Logs\Exception\log.txt. The log file will generate date wise. If you want to go in detail please refer below link
2. https://github.com/serilog/serilog-sinks-file
3. In case of data insertion we have maintained the csv log file in UserManagement\UserManagement\Logs\Csv folder. After 15 days the file will get deleted automatically.
4. Created CSV log file for the below model.
5. InstituteModelForCsv
6. LoginModelForCsv
7. MemberInstitutionModelForCsv
8. MembersSlotModelForCsv
9. MemberMenuModelForCsv
10. AuditTrailModelForCsv
11. MemberBulkValidCsv\_GUID
12. MemberBulkInvalidCsv\_GUID
13. MemberBulkValidCsv\_GUID\_processed
14. MemberBulkInvalidCsv\_GUID\_processed

**Please Note:**

1. **Don’t add the link in case of email id. It should be in plain text.**
2. **Don’t add any format for the excel column like date,time it should be plain text only.**
3. **Make sure you enter “Date of Birth DD-MM-YYYY” in the correct format e.g 05-02-2021.**
4. **If a user enters the wrong city then the application picks any right city from the respective District.**
5. **If the district short code is not present under lgd\_dist then it will give an error as “No District code found for {districtName}. Please Contact the Admin !"”**