**SANOFI VENDOR MANAGEMENT TOOL DOCUMENTATION**

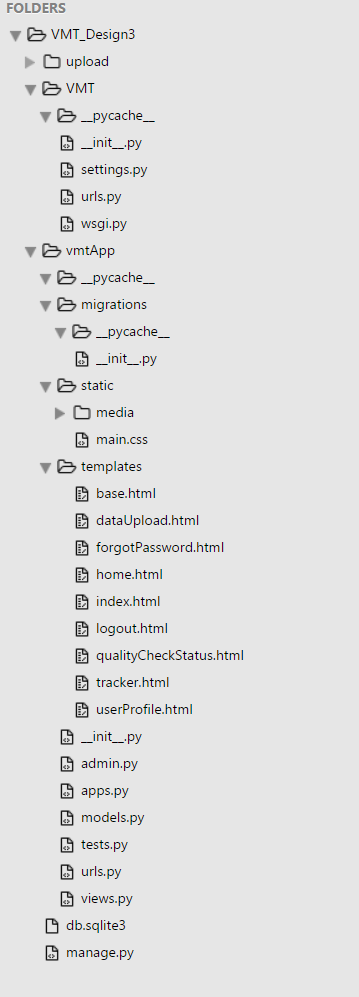
**Project Name:** Vendor Management Tool

**Introduction:** This project automates the entire manual process of data validation through a single robust platform.

**Requirements:**

1. IDE: Sublime Text
2. Python version 3.7.2
3. Django version 2.1.4
4. Bootstrap Frameworks version 4.2

**Folder Structure:**



**Instructions:**

1)Creating a new project:

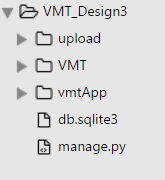
**django-admin.py startproject VMT**

2)Creating a new app in the project:

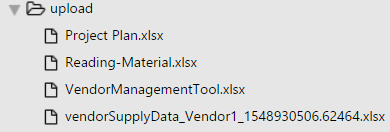
**python manage.py startapp vmtApp**

3)Rename the Project folder as VMT\_Design#, according to the version control and create a new folder “upload”.

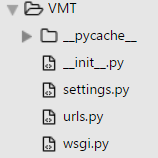
* Now the folder structure looks like this.



* Upload folder contains the files that are uploaded from the users end.



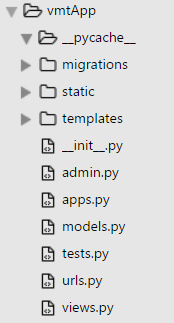
4)VMT folder contains all the files that are required for the configuration of the application.

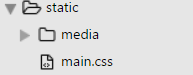


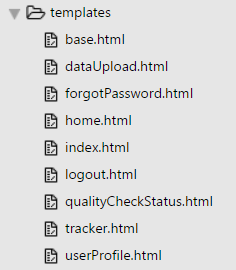
* “Settings.py” is the central configuration for all Django projects. It consists of a series of variables to configure things like Django applications, databases, templates and middleware, among other things.
* “Urls.py” consists of all the URL’s to be defined in the Django application.

5)The application folder structure looks like this.

* The typical way for Django applications to interact with data is through Django models.
* A Django model is an object orientated Python class that represents the characteristics of an entity.
* Since Django models revolve around data, they are prone to change.
* Maintaining these Django models changes throughout time is also an important aspect of Django models and is managed using migration files.
* All the migrations made are stored in the migration folder.
* Static files consist of the media files and the CSS files.
* Templates consist of the html files that are being used in the application.

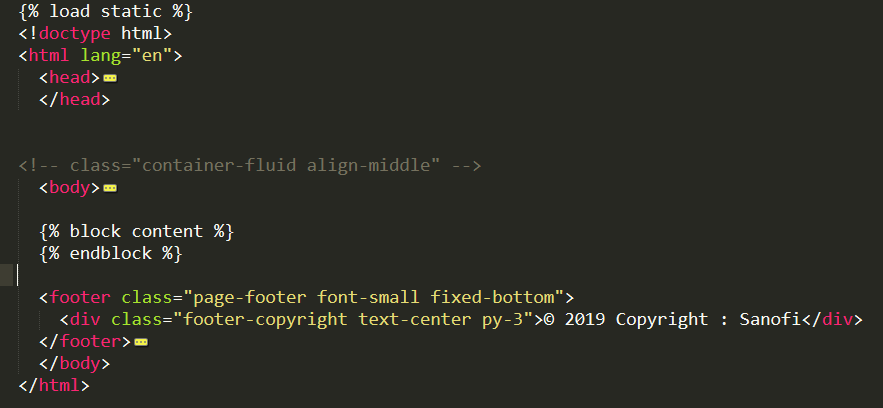






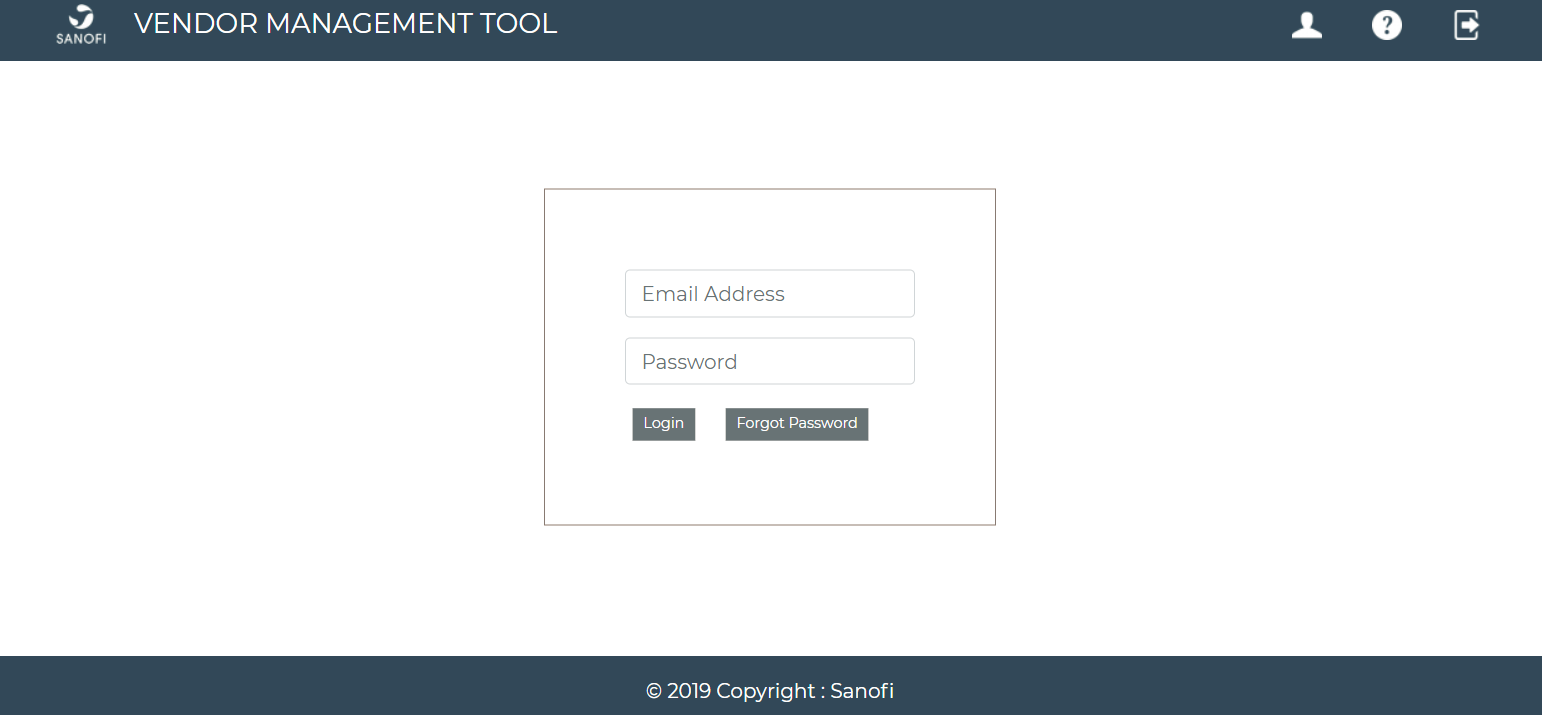
* Base.html => It is the generic format for all the pages.
* dataUpload.html => HTML file for data upload page.
* forgotPassword.html => HTML file for forgot password page.
* home.html => HTML file for home page.
* Index.html => HTML file for main website page
* logout.html => HTML file for logout page
* qualityCheckStatus.html => HTML file for checking quality status of the uploaded file
* tracker.html => HTML file for tracker page
* userProfile.html => HTML file for user profiles

Below is shown the code for the “base.html”.

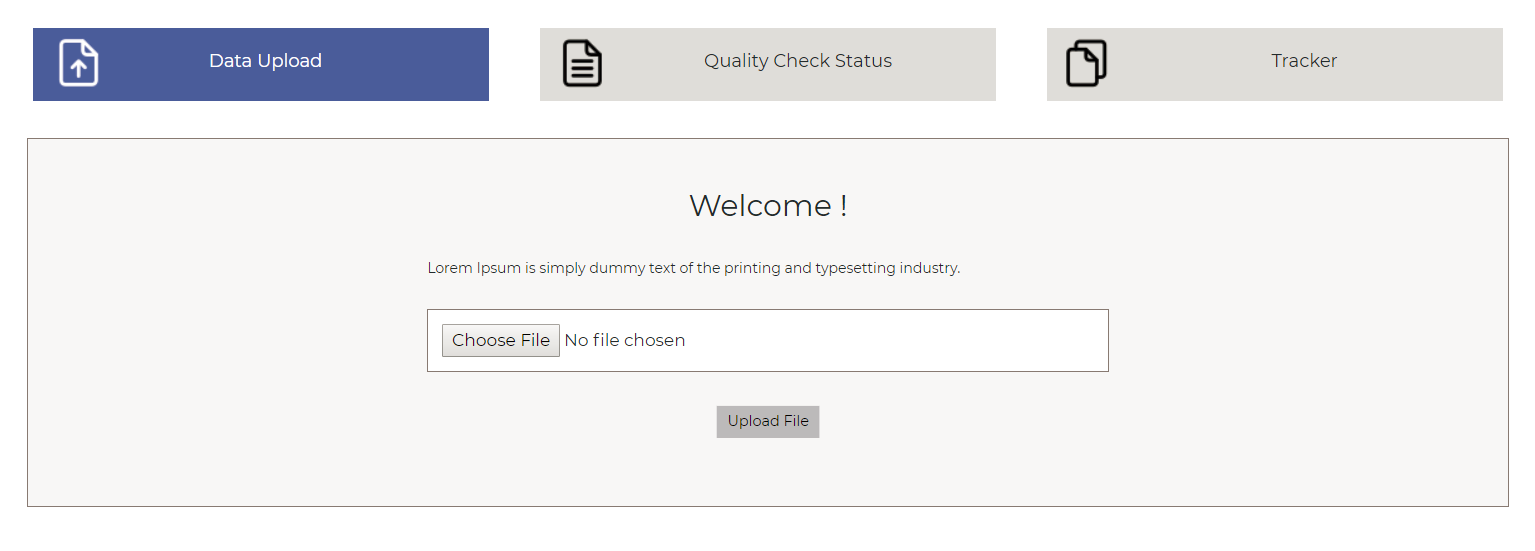
{% load static %}, uses Django's special template tag syntax to tell the template engine to use the files in the static folder in this template.

It loads the “base.html” page first and the remaining pages are loaded on top of the page so that it serves as the basic template for the remaining pages.

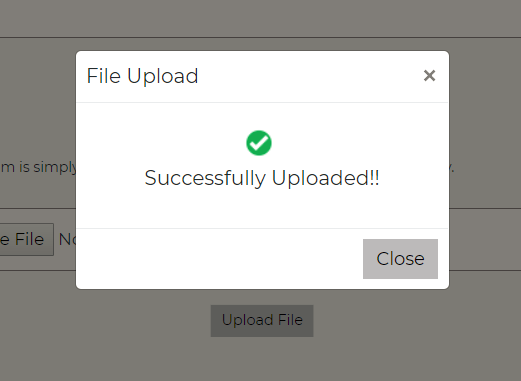
This is how the login page looks like.



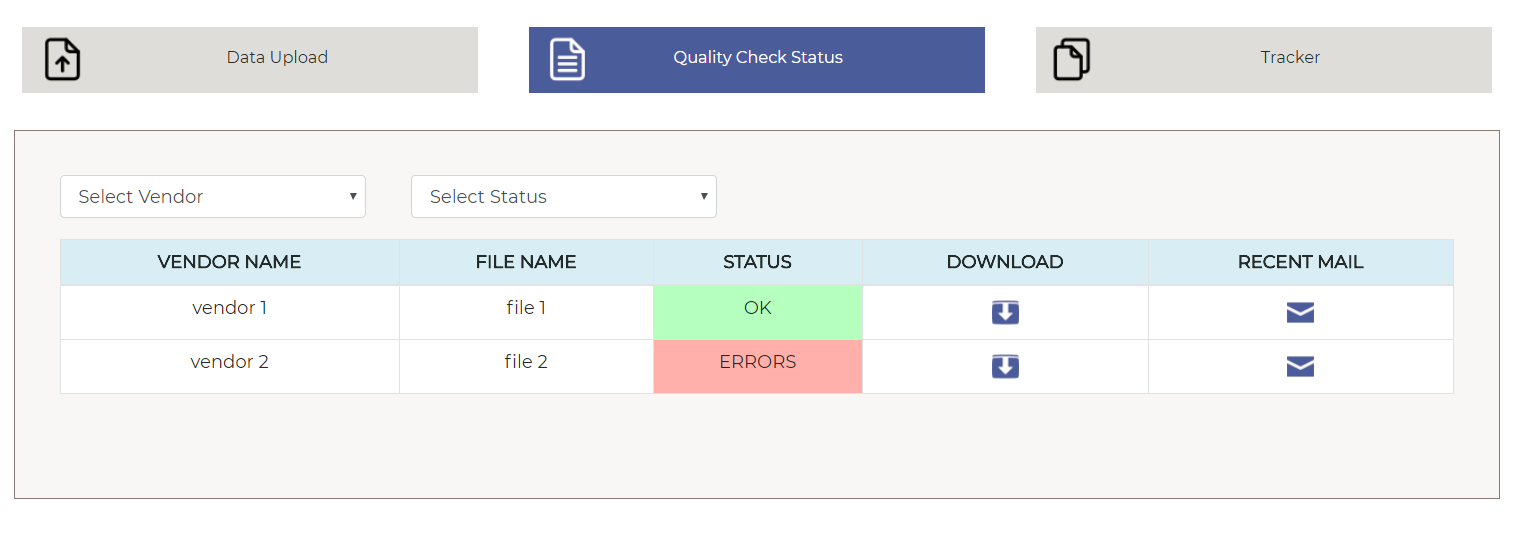
This is the Data Upload section where user can upload excel files only with size specified.

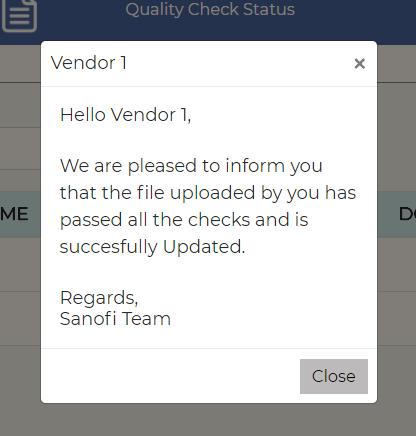


Following is the response of the Application on the successful upload of an excel file.



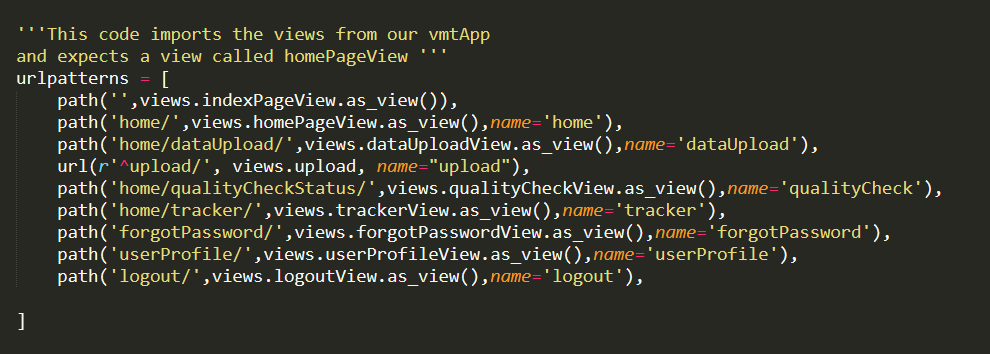
The image below shows the quality checks applied on the uploaded data along with an option to download the uploaded file and check the recent mails associated with it.





Checking the recent mails that shows the communication with the vendor.

All the URL’s present in the application are mentioned in the file at “urls.py” file of the vmtApp



To show a page you must create views. Following are the views that were created for the different page.

