



MYSORE UNIVERSITY SCHOOL OF ENGINEERING

Manasagangotri campus, Mysuru-570006
(Approved by AICTE, New Delhi)



UNIVERSITY OF MYSORE

Full Stack Development(21CD71) Assessment Report On:
“COMPLAINT MANAGEMENT SYSTEM ”

Under the guidance :
Mr. Karthik M N
Assistant Professor,
Department of Computer
Science & Design,
MUSE.

Submitted by:
V LAVANYA
Reg No : 21SECD53

COMPLAINT MANAGEMENT SYSTEM - DJANGO PROJECT

REPORT

INTRODUCTION

The **Complaint Management System** is a web-based application built using Django that allows users to submit complaints through a form.

The submitted complaints are stored in a database and managed via the Django Admin panel.

After submission, users are redirected to a **success page**.

KEY FEATURES:

- ✓ User-friendly complaint submission form
- ✓ Admin panel for complaint management
- ✓ Success page after submission
- ✓ Light-themed UI for better user experience

TECHNOLOGIES USED

The project is developed using the following technologies:

- **Backend:** Django (Python Web Framework)
- **Frontend:** HTML, CSS
- **Database:** SQLite3
- **Development Environment:** VS Code
- **Version Control:** GitHub

PROJECT SETUP & FOLDER STRUCTURE

Installation Steps:

Install Django:

```
pip install django
```

Create Django Project & App:

```
django-admin startproject complaint_system
```

```
cd complaint_system
```

```
python manage.py startapp complaints
```

Apply Migrations & Run Server:

```
python manage.py migrate
```

```
python manage.py runserver
```

FOLDER STRUCTURE:

```
complaint_system/
```

```
| — complaints/
```

```
|   | — migrations/
```

```
|   | — static/
```

```
|   |   | — complaints/style.css
```

```
|   | — templates/
```

```
|   |   | — complaints/home.html
```

```
|   |   | — complaints/complaint_file.html
```

```
|   |   | — complaints/success.html
```

```
|   | — models.py
```

```
|   | — views.py
```

```
|   | — urls.py
```

```
| — complaint_system/
```

```
| — db.sqlite3
```

```
| — manage.py
```

MODELS (DATABASE STRUCTURE)

The **Complaint model** is used to store user complaints in the database.

models.py file:

```
from django.db import models

class Complaint(models.Model):

    PRIORITY_CHOICES = [

        ('High', 'High'),

        ('Medium', 'Medium'),

        ('Low', 'Low'),

    ]

    STATUS_CHOICES = [

        ('Pending', 'Pending'),

        ('Resolved', 'Resolved'),

    ]

    name = models.CharField(max_length=100)

    email = models.EmailField()

    product = models.CharField(max_length=100)

    description = models.TextField()

    priority = models.CharField(max_length=10, choices=PRIORITY_CHOICES, default='Medium')

    status = models.CharField(max_length=10, choices=STATUS_CHOICES, default='Pending')

    def __str__(self):

        return self.product
```

FORMS & VIEWS

The **complaint form** collects user input and saves it to the database.

forms.py file:

```
from django import forms

from .models import Complaint

class ComplaintForm(forms.ModelForm):

    class Meta:

        model = Complaint

        fields = ['name', 'email', 'product', 'issue_description', 'priority']
```

views.py file:

```
from django.shortcuts import render, redirect

from django.urls import reverse_lazy

from .forms import ComplaintForm

def submit_complaint(request):

    if request.method == 'POST':

        form = ComplaintForm(request.POST)

        if form.is_valid():

            form.save()

            return redirect(reverse_lazy('complaint_success'))

        else:

            form = ComplaintForm()

            return render(request, 'complaints/complaint_file.html', {'form': form})
```

Users are redirected to a success page after submitting the form.

URL CONFIGURATION

Project-level (complaint_system/urls.py) and app-level (complaints/urls.py) URL patterns are defined as follows:

Project URLs (complaint_system/urls.py)

```
from django.contrib import admin

from django.urls import path, include

urlpatterns = [

    path('admin/', admin.site.urls),

    path("", include('complaints.urls')),

]
```

App URLs (complaints/urls.py)

```
from django.urls import path

from .views import home, submit_complaint, complaint_success

urlpatterns = [

    path("", home, name='home'),

    path('submit/', submit_complaint, name='submit_complaint'),

    path('success/', complaint_success, name='complaint_success'),

]
```

TEMPLATES

Templates (HTML Files):

- **home.html** → Landing page

```
home.html X
complaints > templates > complaints > home.html > ...
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4      <meta charset="UTF-8">
5      <meta name="viewport" content="width=device-width, initial-scale=1.0">
6      <title>Home - Complaint System</title>
7      {% load static %}
8      <link rel="stylesheet" type="text/css" href="{% static 'complaints/style.css' %}">
9  </head>
10 <body>
11     <div class="container">
12         <h1>Welcome to the Customer Complaint System</h1>
13         <p>If you have any issues, feel free to submit a complaint.</p>
14         <a href="{% url 'submit_complaint' %}">Submit a Complaint</a>
15     </div>
16 </body>
17 </html>
18
```

- **complaint_file.html** → Complaint submission form

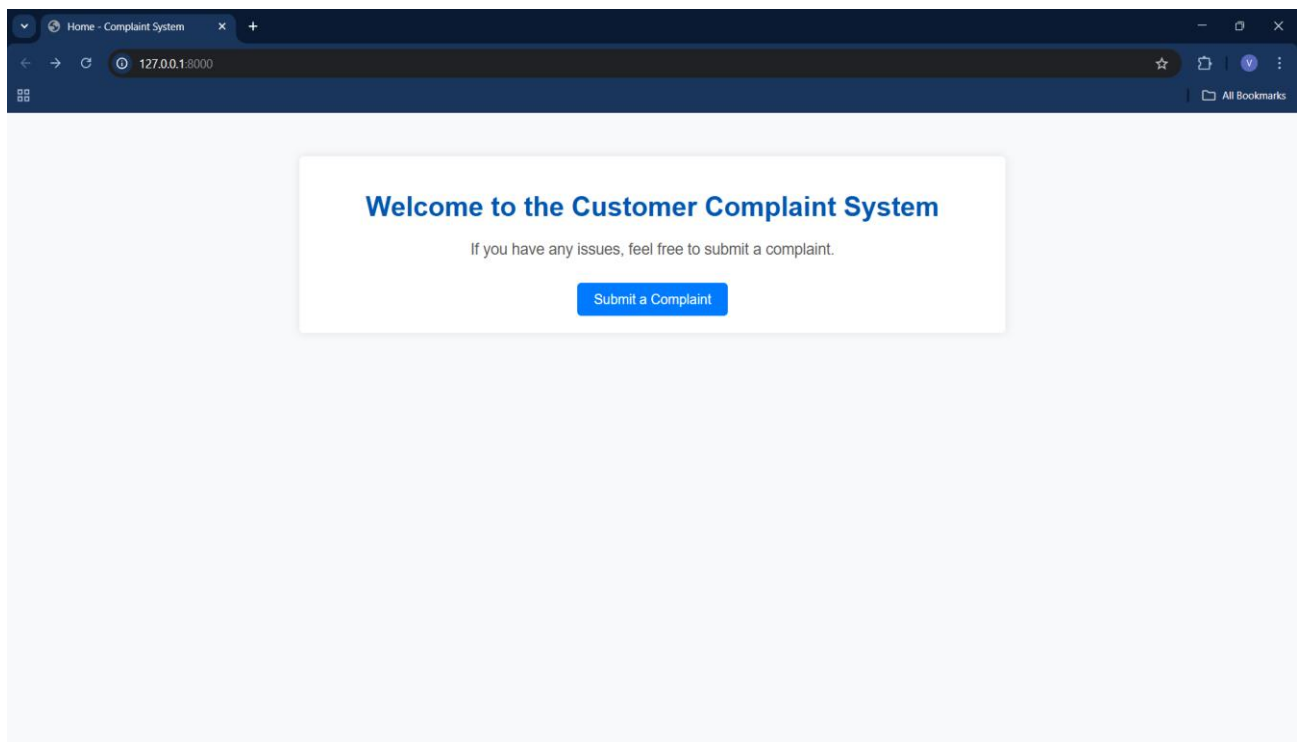
```
complaint_file.html X
complaints > templates > complaints > complaint_file.html > ...
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4      <meta charset="UTF-8">
5      <meta name="viewport" content="width=device-width, initial-scale=1.0">
6      <title>Submit Complaint</title>
7      {% load static %}
8      <link rel="stylesheet" type="text/css" href="{% static 'complaints/style.css' %}">
9  </head>
10 <body>
11     <h2>Submit Your Complaint</h2>
12     <form method="POST">
13         {% csrf_token %}
14         {{ form.as_p }}
15         <button type="submit">Submit</button>
16     </form>
17 </body>
18 </html>
19
20
```

- **success.html** → Complaint successfully submitted message

```
success.html X
complaints > templates > complaints > success.html > ...
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4      <meta charset="UTF-8">
5      <meta name="viewport" content="width=device-width, initial-scale=1.0">
6      <title>Success</title>
7      {% load static %}
8      <link rel="stylesheet" type="text/css" href="{% static 'complaints/style.css' %}">
9  </head>
10 <body>
11     <h2>Thank you! Your complaint has been submitted.</h2>
12     <a href="{% url 'submit_complaint' %}">Submit Another Complaint</a>
13
14 </body>
15 </html>
16
```

SCREENSHOTS

Home Page Screenshot



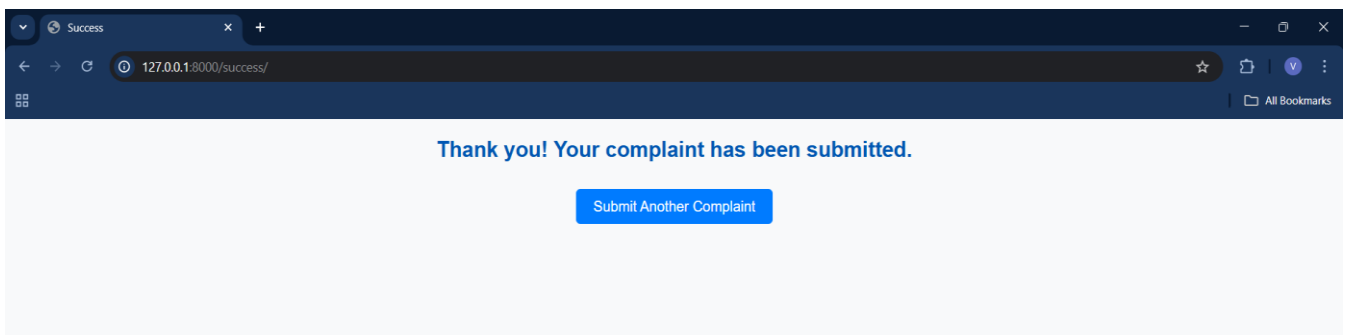
Complaint Form Screenshot

The screenshot shows a web browser window with the address bar displaying `127.0.0.1:8000/submit/`. The page title is "Submit Complaint". The main heading is "Submit Your Complaint". The form contains the following fields:

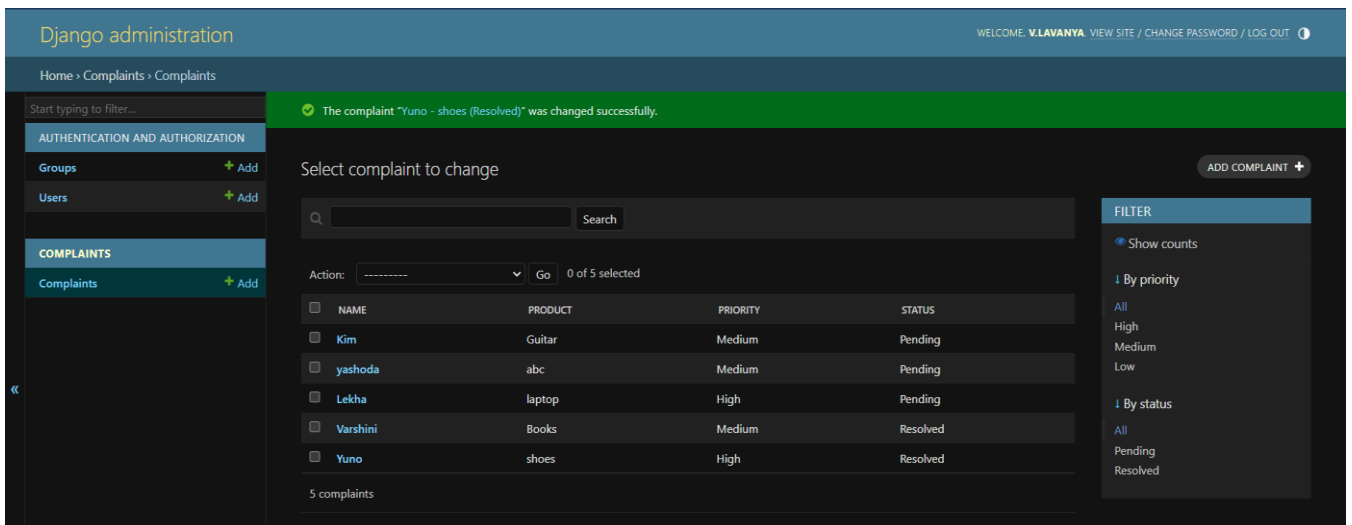
- Name: Kim
- Email: kim@gmail.com
- Product: Guitar
- Issue description: Broken Strings
- Priority: Medium (selected from a dropdown menu)

A blue "Submit" button is located at the bottom of the form.

Success Page Screenshot



Admin Panel Screenshot



Implement `reverse_lazy()` to redirect users to a success page after submission.

```
views.py 2 X
complaints > views.py > submit_complaint
1 from django.shortcuts import render, redirect
2 from django.urls import reverse_lazy
3 from .forms import ComplaintForm
4
5 def home(request):
6     return render(request, 'complaints/home.html')
7
8 def submit_complaint(request):
9     if request.method == "POST":
10         form = ComplaintForm(request.POST)
11         if form.is_valid():
12             form.save()
13             return redirect(reverse_lazy('complaint_success'))
14     else:
15         form = ComplaintForm()
16         return render(request, 'complaints/complaint_file.html', {'form': form})
17
18 def complaint_success(request):
19     return render(request, 'complaints/success.html')
20
```