NAAN MUDHALVAN –IBM

PROFESSIONAL READLINESS FOR INNOVATION, EMPLOYABILITY AND ENTERPRENEURSHIP

3105-DHANALAKSHMI SRINIVASAN COLLEGE OF ENGINEERING AND TECHNOLOGY -ECR,MAMALLAPURAM-603104



DOMAIN- “APPLIED DATA SCIENCE”

GROUP-01

# TEAM ID: NM2023TMID07999

TEAM SIZE:05

Team Leader : SREEKANTH K(310520104125)

Team member : BAGAVATHI S(310520104020)

Team member : MOHANRAJ K(310420104073)

Team member : MOHAMED SHEIK A(310420104071)

Team member : SITRARASU R(310520104118)

PROJECT TITTLE

**THE ISSUE TRACKER : A Reliable Energy Consumption Analysis system for Energy-Efficient appliances**

TABLE OF CONTENTS:

1. **INTRODUCTION**
   1. Project Overview
   2. Purpose
2. **IDEATION & PROPOSED SOLUTION**
   1. Problem Statement Definition
   2. Empathy Map Canvas
   3. Ideation & Brainstorming
   4. Proposed Solution
3. **REQUIREMENT ANALYSIS**
   1. Functional requirement
   2. Non-Functional requirements
4. **PROJECT DESIGN**
   1. Data Flow Diagrams
   2. Solution & Technical Architecture
   3. User Stories
5. **CODING & SOLUTIONING (Explain the features added in the project along with code)**
   1. Feature 1
   2. Feature 2
   3. Database Schema (if Applicable)
6. **RESULTS**
   1. Performance Metrics
7. **ADVANTAGES & DISADVANTAGES**
8. **CONCLUSION**
9. **FUTURE SCOPE**

INTRODUCTION

# A complaint management system is a software application that helps organizations to track and manage complaints. It can be used to collect, store, and analyze complaints, as well as to track the progress of complaints and ensure that they are resolved in a timely manner.

# A complaint management system can be used by any organization that receives complaints from its customers, employees, or other stakeholders. It can be a valuable tool for improving customer satisfaction, reducing the risk of legal action, and improving the overall efficiency of an organization.

# The following are some of the benefits of using a complaint management system:

# Improved customer satisfaction: A complaint management system can help organizations to improve customer satisfaction by providing a more efficient and effective way to handle complaints. This can lead to increased customer loyalty and repeat business.

# Reduced risk of legal action: A complaint management system can help organizations to reduce the risk of legal action by providing a more efficient and effective way to handle complaints. This can help to protect organizations from costly lawsuits.

# Improved efficiency: A complaint management system can help organizations to improve

# their efficiency by providing a more centralized and organized way to handle complaints. This can free up employees to focus on other tasks and can help to reduce the time it takes to resolve complaints.

# There are many different complaint management systems available on the market. When choosing

# a system, it is important to consider the specific needs of your organization. Some factors to

# consider include the size of your organization, the number of complaints you receive, and the features you need.

# Once you have chosen a complaint management system, it is important to train your employees on how to use it. This will help to ensure that complaints are handled in a consistent and efficient manner.

* 1. **PROJECT OVERVIEW**

Users can submit complaints through a variety of channels, such as a web form, email, or phone call.

Complaint tracking: The system tracks the status of each complaint, including the date it was submitted, the assigned department, and the current status.

Complaint resolution: The system allows users to resolve complaints by providing updates, assigning tasks, and closing the complaint.

Reporting: The system provides reports on the number and type of complaints, as well as the status of complaints.

If you are looking for a way to improve the way your organization handles complaints, a CMS is a great option.

Here is an overview of the steps involved in implementing a CMS:

* Identify your needs: The first step is to identify your organization's needs. What are your goals for implementing a CMS? What features are important to you?
* Research different systems: Once you know your needs, you can start researching different CMSs. There are many different systems available, so it's important to compare features and prices.
* Choose a system: Once you've chosen a system, it's time to implement it. This process will vary depending on the system you choose, but it will typically involve training your employees on how to use the system and configuring the system to meet your needs.
* Monitor and improve: Once your system is in place, it's important to monitor its performance and make improvements as needed. This will help you ensure that the system is meeting your needs and that complaints are being handled effectively

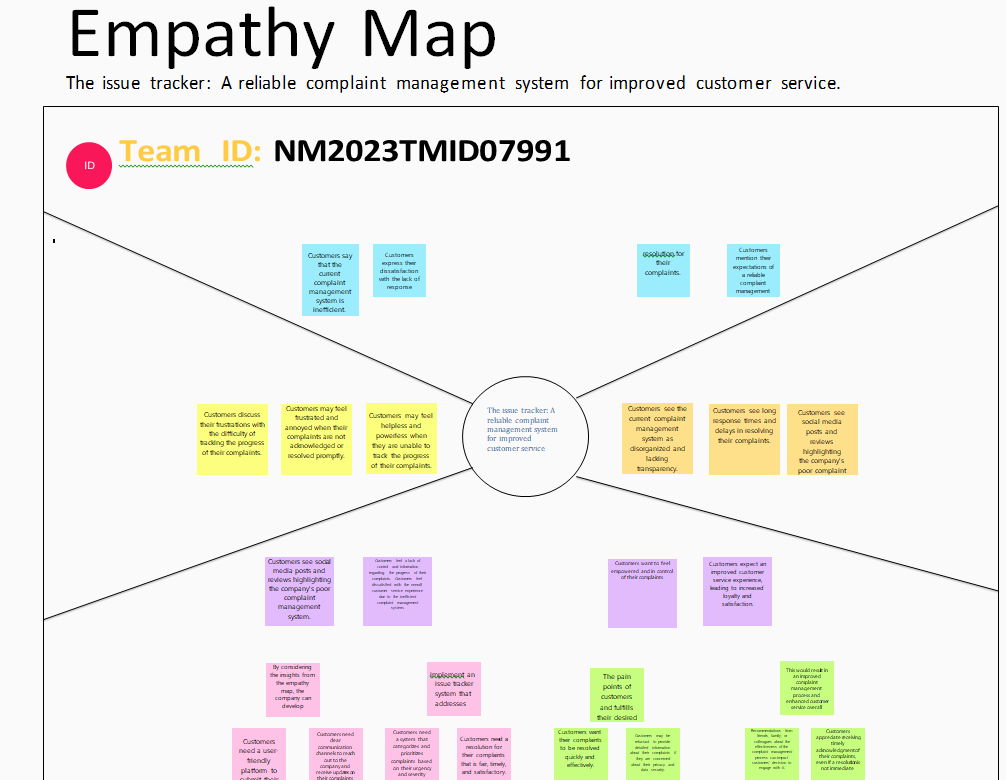
**PURPOSES**

The purpose of a complaint management system (CMS) project is to improve the way an organization handles complaints. A CMS can help organizations to improve customer satisfaction, reduce the risk of legal action, and improve the overall efficiency of an organization.

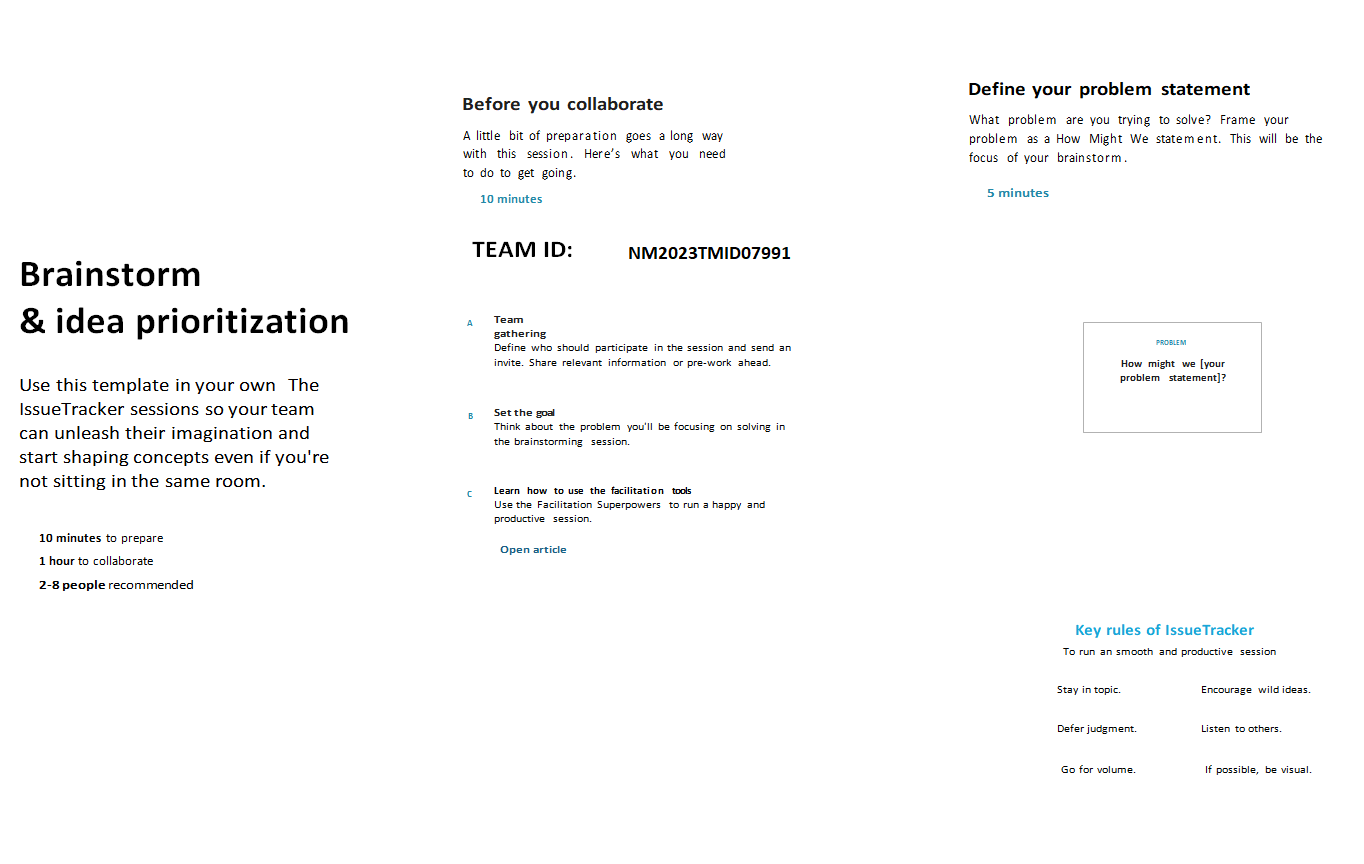
Here are some of the specific benefits of implementing a CMS:

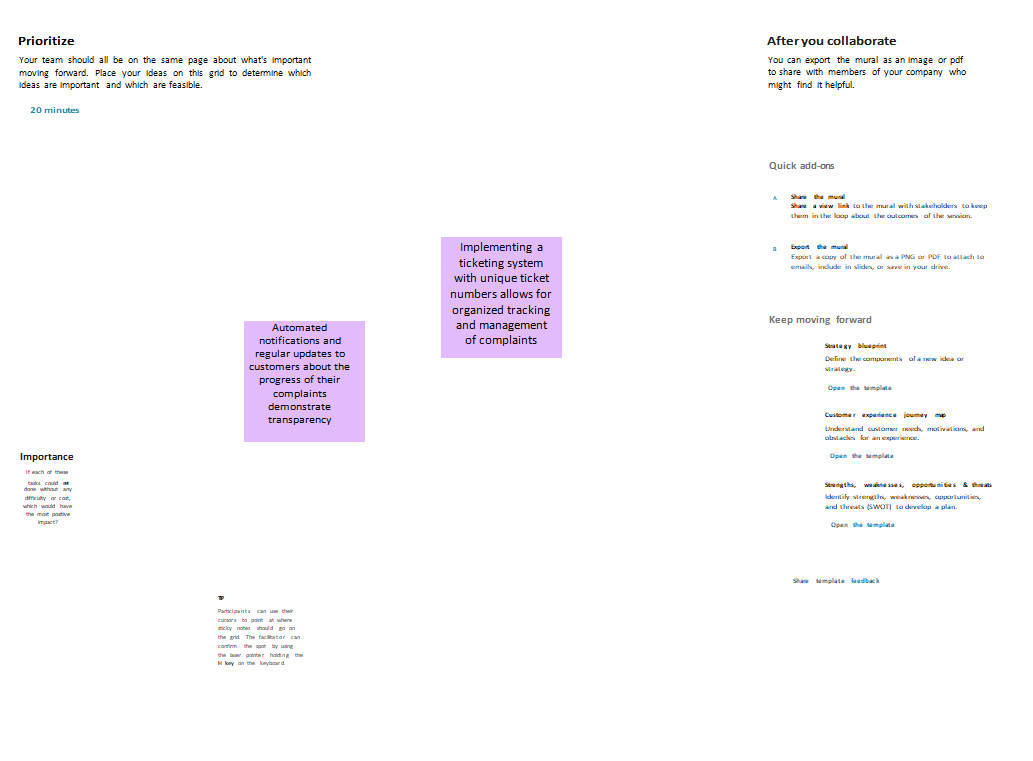
* Improved customer satisfaction: A CMS can help organizations to improve customer satisfaction by providing a more efficient and effective way to handle complaints. This can lead to increased customer loyalty and repeat business.
* Reduced risk of legal action: A CMS can help organizations to reduce the risk of legal action by providing a more efficient and effective way to handle complaints. This can help to protect organizations from costly lawsuits.
* Improved efficiency: A CMS can help organizations to improve their efficiency by providing a more centralized and organized way to handle complaints. This can free up employees to focus on other tasks and can help to reduce the time it takes to resolve complaints.

1. **IDEATION & PROPOSED SOLUTION**
   1. **Empathy Map Canvas**



* 1. **Ideation & Brainstorming**





* 1. **Proposed Solution**

The system will be a web-based application that will allow users to submit complaints, track the status of their complaints, and resolve complaints. The system will also provide reports on the number and type of complaints, as well as the status of complaints.

The system will be designed to be secure from unauthorized access. It will also be designed to be able to handle a large volume of complaints.

The system will be implemented in three phases:

Phase 1: The system will be developed and tested.

Phase 2: The system will be deployed to users.

Phase 3: The system will be monitored and maintained.

**3.REQUIREMENT ANALYSIS**

The following are some of the specific requirements that may be considered for a complaint management system project:

* Complaint submission: Users should be able to submit complaints through a variety of channels, such as a web form, email, or phone call.
* Complaint tracking: The system should track the status of each complaint, including the date it was submitted, the assigned department, and the current status.
* Complaint resolution: The system should allow users to resolve complaints by providing updates, assigning tasks, and closing the complaint.
* Reporting: The system should provide reports on the number and type of complaints, as well as the status of complaints.
* Security: The system should be secure from unauthorized access.
* Performance: The system should be able to handle a large volume of complaints.
* Scalability: The system should be able to be scaled to handle a growing number of complaints.
  1. Functional requirement

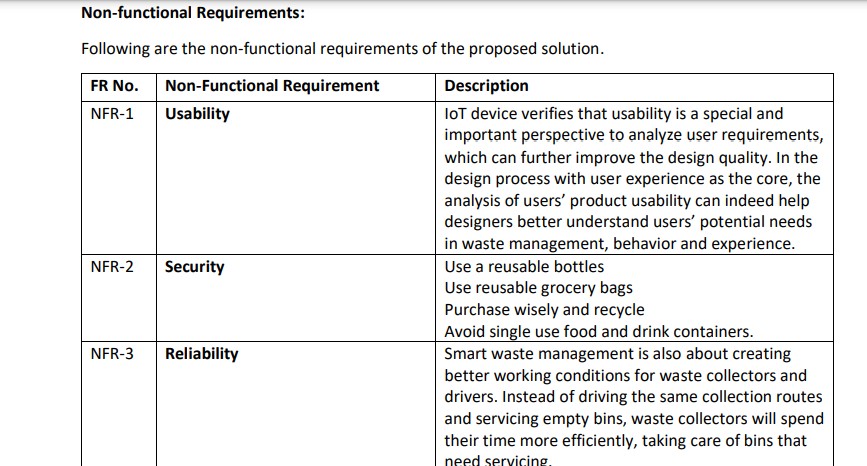
Following are the functional requirements of the proposed solution.

|  |  |  |
| --- | --- | --- |
| **FR No.** | **Functional Requirement (Epic)** | **Sub Requirement (Story/ Sub-Task)** |
| FR-1 | Detailed Tracking inventory. | Users should be able to submit complaints through a variety of channels, such as a web form, email, or phone call.  The system should provide a clear and concise way for users to describe their complaint.  The system should collect the following information from users:  Name  Contact information  Complaint details  Department or location of the complaint |
| FR-2 | Real time monitoring. | The system should provide reports on the number and type of complaints, as well as the status of complaints.  The system should provide reports that can be used to identify trends and patterns in complaints.  The system should provide reports that can be used to improve the complaint management process. |

...,, .........

|  |  |  |
| --- | --- | --- |
| FR-3 | Complain resolution | The system should allow users to resolve complaints by providing updates, assigning tasks, and closing the complaint.  The system should provide users with the ability to provide feedback on the resolution of their complaint. |
| FR-4 | Reporting: | The system should provide reports on the number and type of complaints, as well as the status of complaints.  The system should provide reports that can be used to identify trends and patterns in complaints.  The system should provide reports that can be used to improve the complaint management process. |
| FR-5 | Complaint resolution time report: | This report will show the average  time it takes to resolve complaints,  as well as the longest and shortest times it has taken to resolve complaints. |

* 1. Non-Functional requirements



-

|  |  |  |
| --- | --- | --- |
| NFR-4 | **Performance** | The system should be regularly monitored and evaluated to ensure that it is meeting the needs of the organization.  The system should be customized to meet the specific needs of the organization.  The system should be integrated with other systems, such as customer relationship management (CRM) systems and human resources (HR) systems. |
| NFR-5 | **Availability** | The system should be available 24/7. |
| NFR-6 | **Scalability** | he system should be able to be scaled to handle a growing number of complaints. |

1. **PROJECT DESIGN**

**Employee Collaboration**:

Internal messaging or chat system to facilitate communication among employees.

Assign tasks, set deadlines, and monitor progress.

Collaborative document sharing for gathering evidence or compiling reports.

Supervisory roles with access to view and oversee complaint handling activities.

**Reporting and Analytics:**

Generate real-time reports on complaint trends, resolution time, and employee performance.

Identify recurring issues or bottlenecks in the complaint resolution process.

Analyze customer satisfaction ratings and feedback.

Provide graphical representations and data visualization for easy interpretation.

**Technology Stack:**

Front-end: HTML, CSS, JavaScript.

Back-end: Python.

Database: Ibm Cloud.

Additional libraries or frameworks as per development requirements.

**Data Security and Privacy:**

Implement robust security measures to protect customer data.

Apply encryption techniques for sensitive information.

Define access controls and user roles to ensure data integrity and confidentiality**.**

* 1. **Data Flow Diagrams**

**Data Flow Diagrams:**

A Data Flow Diagram (DFD) is a traditional visual representation of the information flows within a system. A neat and clear DFD can depict the right amount of the system requirement graphically. It shows how data enters and leaves the system, what changes the information, and where data is stored.

DATA FLOW DIAGRAMS

The customer interacts with the Customer Interface to submit a complaint.

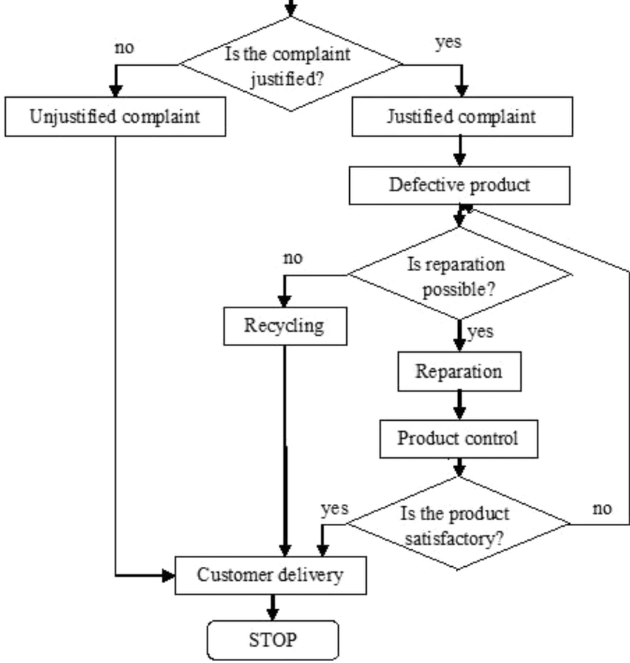
The Complaint Submission Process creates a complaint record and stores it in the Complaint Database.

The Complaint Assignment Process assigns the complaint to an appropriate employee based on category or severity.

The assigned employee accesses the Employee Dashboard to handle the complaint and updates the complaint status in the Complaint Database.

The customer can check the complaint status through the Customer Interface, which retrieves the information from the Complaint Database.

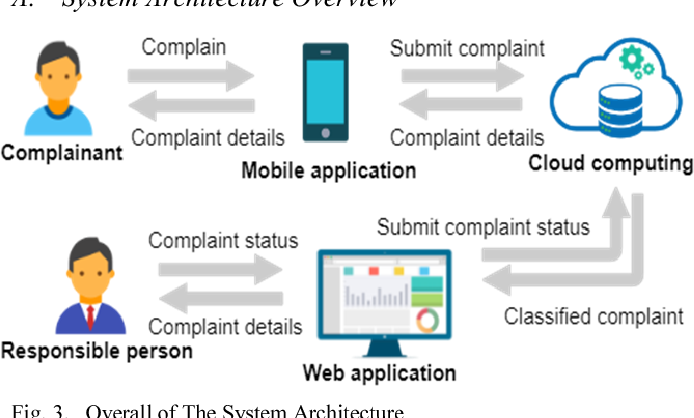
The Reporting and Analytics component generates reports and provides insights based on the data stored in the Complaint Database.



* 1. Solution & Technical Architecture

**Technical Architecture:**

The Deliverable shall include the architectural diagram as below and the information as per the table1 & table 2

****

**Table-1 : Components & Technologies:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **S.No** | **Component** | **Description** | **Technology** | |
| 1. | User Interface | How user interacts with application e.g. Web UI, Mobile App, Chatbot etc. | HTML, CSS, JavaScript /php etc. | |
| 2. | Application Logic-1 | Logic for a process in the application | Python | |
| 3. | Application Logic-2 | Logic for a process in the application | IBM database | |
| 4. | Application Logic-3 | Logic for a process in the application | Html server and php | |
| 5. | Database | Data Type, Configurations etc. | Ibm db2 | |
| 6. | Cloud Database | Database Service on Cloud | IBM DB2, IBM Cloudant et | |
| 7. | File Storage | File storage requirements | IBM Block Storage or Other Storage Service or Local Filesystem |  |
| 8. | External API-1 | Purpose of External API used in the application | IBM ,flask |
| 9. | External API-2 | Purpose of External API used in the application |  | |

**Table-2: Application Characteristics:**

|  |  |  |  |
| --- | --- | --- | --- |
| **S.No** | **Characteristics** | **Description** | **Technology** |
| 1. | Open-Source Frameworks | List the open-source frameworks used | Technology of Opensource framework |
| 2. | Security Implementations | List all the security / access controls implemented, use of firewalls etc. | e.g. SHA-256, Encryptions, IAM Contr |
| 3. | Scalable Architecture | Justify the scalability of architecture (3 – tier, Micro-services) | Technology used |
| 4. | Availability | Justify the availability of application (e.g. use of load balancers, distributed servers etc.) | Technology used |
| 5. | Performance | Design consideration for the performance of the application (number of requests per sec, use of Cache, use of CDN’s) etc. | Technology used |

* 1. User Stories

**User Stories**

Use the below template to list all the user stories for the product.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **User Type** | **Functional Requirement (Epic)** | **User Story Number** | **User Story / Task** | **Acceptance criter** |
| ADMIN (WHO MANAGE WEB SERVER) | REGISTRATION | USN-1 | As a user, I can register for the application by entering my email, password, and confirming my password. | I can access my ac dashboard |
| CO ADMIN | LOGIN | USN-2 | As a user, I will receive confirmation email once I have registered for the application | I can receive confirm email &  click confirm |
| MANAGER | LOGIN | USN-3 | As a user, I can register for the application through Facebook | I can register & acc  dashboard with right credencials  Login |
| USER | LOGIN | USN-4 | As a user, I can register for the application through Gmail |  |

1. **CODING & SOLUTIONING (Explain the features added in the project along with code)**
   1. **FEATURE 1**

A Customer Interface:

User registration and login functionality.

User-friendly complaint submission form.

Ability to attach supporting documents or evidence.

Automatic acknowledgment and complaint reference number.

Complaint Tracking:

Real-time status updates for customers.

Internal tracking of complaints throughout the resolution process.

Escalation mechanism for unresolved complaints.

Assigning and Routing:

Automatic assignment of complaints based on category or severity.

Routing complaints to the appropriate department or employee.

Email notifications to assigned employees.

Collaboration and Communication:

Internal messaging or chat system for employees involved in complaint resolution.

Task assignment and deadline setting for employees.

Collaborative document sharing for gathering evidence or compiling reports.

Supervisory roles with access to oversee complaint handling activities.

Workflow Management:

Workflow automation to streamline complaint resolution processes.

Define and enforce standard operating procedures.

Track and manage deadlines and response times.

Analytics and Reporting:

Generate reports on complaint trends, resolution time, and employee performance.

Identify recurring issues or bottlenecks in the complaint resolution process.

Analyze customer satisfaction ratings and feedback.

Data visualization for easy interpretation.

Integration and Data Management:

Integration with other systems or databases for data exchange.

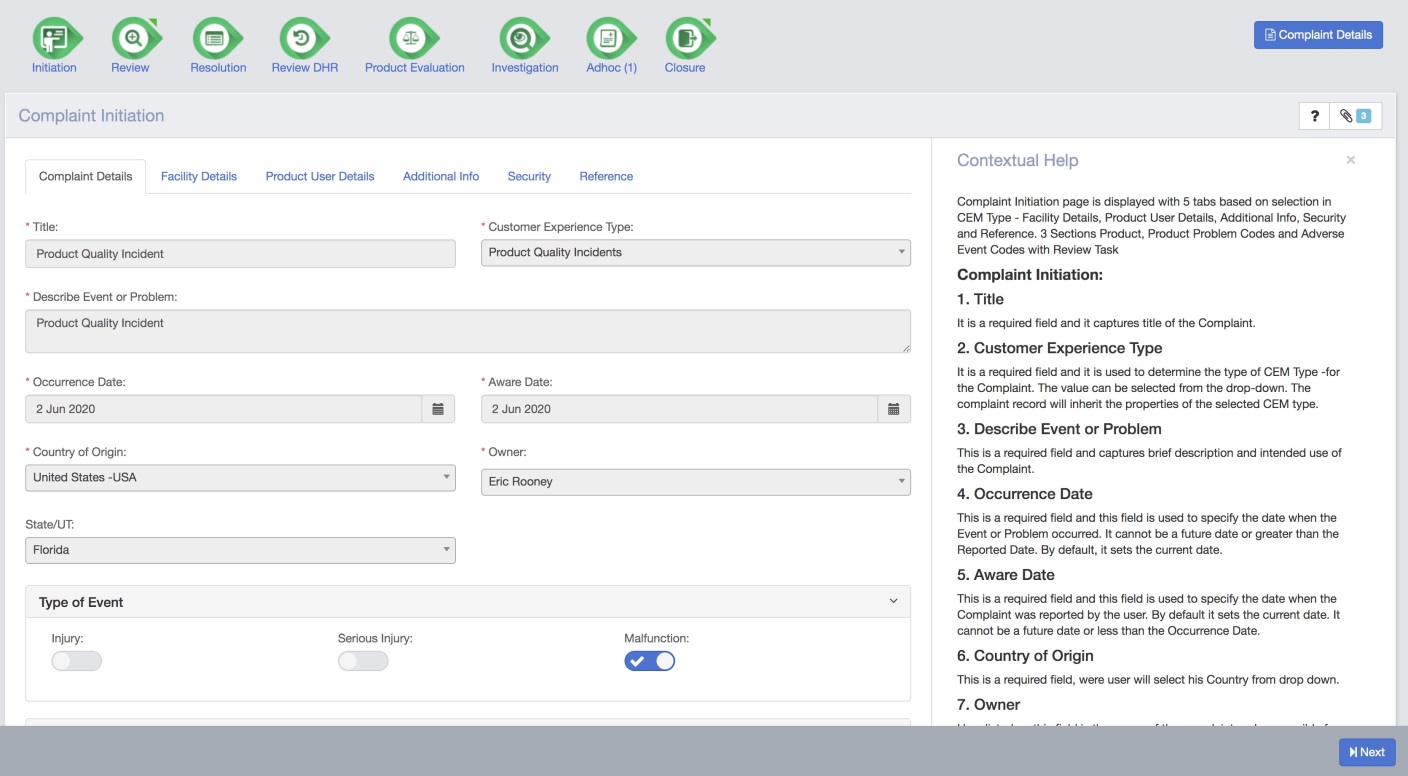
Secure storage and management of customer data and complaint records.

Compliance with data protection regulations.

Notification and Alerting:

Automated notifications to customers and employees at key stages of the complaint resolution process.

Alerts for urgent or high-priority complaints.



* 1. **FEATURE 2**

**Complaint submission**: Users will be able to submit complaints through a variety of channels, such as a web form, email, or phone call.

**Complaint tracking**: The system will track the status of each complaint, including the date it was submitted, the assigned department, and the current status.

**Complaint resolution**: The system will allow users to resolve complaints by providing updates, assigning tasks, and closing the complaint.

**Reporting**: The system will provide reports on the number and type of complaints, as well as the status of complaints.

Additional features:

**Security**: The system will be secure from unauthorized access.

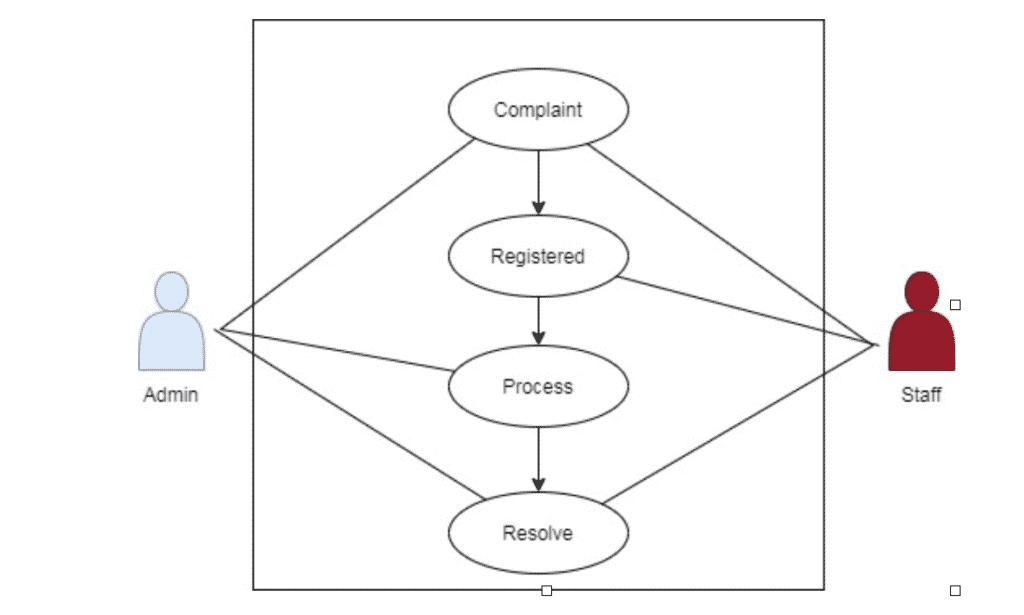
**Performance**: The system will be able to handle a large volume of complaints.

**Scalability:** The system will be able to be scaled to handle a growing number of complaints.

**Customization**: The system will be customizable to meet the specific needs of the organization.

**4.2 BLOCK DIAGRAM**

)I



1. **RESULTS**

### Improved customer satisfaction: The issue tracker can help to improve customer satisfaction by providing a way for customers to submit complaints and track the status of their complaints. This can help to make customers feel like their complaints are being heard and that they are being taken seriously.

* 1. PERFORMANCE METRICES

## By tracking these performance metrics, organizations can identify areas where they can improve their complaint management process. For example, if the time to resolution is too long, organizations can take steps to streamline the process. If customer satisfaction is low, organizations can take steps to improve the way they communicate with customers and resolve their complaints. By tracking performance metrics, organizations can make sure that they are using the issue tracker effectively to manage complaints and improve customer satisfaction.

## **Make it easy to use**: The issue tracker should be easy to use for both customers and employees. The interface should be user-friendly and the system should be intuitive.

## **Provide clear instructions**: The issue tracker should provide clear instructions for customers and employees on how to submit complaints and track their status.

## Offer multiple ways to submit complaints: Customers should be able to submit complaints through a variety of channels, such as a web form, email, or phone call.

## **Track the status of complaints**: The issue tracker should track the status of each complaint, including the date it was submitted, the assigned department, and the current status.

## Provide updates on the status of complaints: Customers should be able to receive updates on the status of their complaints.

## Resolve complaints in a timely manner: Complaints should be resolved in a timely manner.

## **Provide feedback to customers**: Customers should be provided with feedback on the resolution of their complaints.

## Measure the effectiveness of the issue tracker: Organizations should measure the effectiveness of the issue tracker by tracking performance metrics, such as the number of complaints submitted, the time to resolution, and customer satisfaction.

1. **ADVANTAGES :**

## **Improved customer satisfaction**: Issue trackers can help to improve customer satisfaction by providing a way for customers to submit complaints and track the status of their complaints. This can help to make customers feel like their complaints are being heard and that they are being taken seriously.

## **Reduced time to resolution**: Issue trackers can help to reduce the time it takes to resolve complaints by providing a way for users to track the status of complaints and assign tasks to the appropriate people. This can help to ensure that complaints are resolved in a timely manner.

## **Improved data collection**: Issue trackers can help to improve data collection by providing a way to track the number and type of complaints. This data can be used to identify trends and patterns in complaints, which can help organizations to improve their products and services.

1. **DISADVANTAGES :**

## **They can be difficult to use**: Issue trackers can be complex and difficult to use, especially for those who are not familiar with them. This can lead to frustration and errors.

## **They can be expensive**: Issue trackers can be expensive to purchase and maintain. This can be a barrier for small businesses and organizations with limited budgets.

## **They can be time-consuming**: Issue trackers can take a lot of time to set up and use. This can be a challenge for organizations that are short on time or resources.

1. **CONCLUSION**

Centralized Complaint Management: An issue tracker consolidates all customer complaints in one place, allowing businesses to have a centralized view of the issues and track their resolution progress. This ensures that no complaint goes unnoticed or unresolved.

Efficient Issue Tracking: With an issue tracker, businesses can assign and prioritize issues, set deadlines, and track the status of each complaint. This streamlines the complaint resolution process and ensures that issues are addressed promptly and effectively.

Improved Communication: An issue tracker enables effective communication between customers, support teams, and other relevant stakeholders. It provides a platform for customers to report their complaints, and for support teams to respond, seek clarification, or provide updates on the resolution progress. This enhances transparency and keeps all parties informed.

Customer Satisfaction and Retention: By efficiently managing and resolving complaints, businesses can enhance customer satisfaction. A reliable complaint management system demonstrates that the business values customer feedback and is committed to addressing their concerns. This, in turn, improves customer loyalty, reduces churn, and fosters long-term relationships.

1. **FUTURE SCOPE:**

**Artificial Intelligence (AI) Integration**: Issue trackers may utilize AI algorithms to automate certain tasks, such as categorizing and prioritizing complaints, suggesting potential solutions, or even resolving common issues automatically.

**Natural Language Processing (NLP):** Advanced NLP capabilities can enhance the system's ability to understand and interpret user complaints, enabling more accurate and efficient issue tracking and resolution.

**Real-time Tracking and Analytics**: Issue trackers can provide real-time monitoring and reporting of complaints, allowing organizations to track the status of each issue, measure response times, identify recurring problems, and analyze data to gain insights for process improvements.

1. **APPENDIX**

## SOURCE CODE

from flask import Flask, render\_template

app = Flask(\_\_name\_\_)

@app.route('/')

def index():

return render\_template('index.html')

@app.route('/ragistration')

def registration():

return render\_template('registration.html')

@app.route('/profile')

def profile():

return render\_template('profile.html')

@app.route('/registration\_complaint')

def profile():

return render\_template('complaint.html')  
@app.route('/logout')

def logout():

return render\_template('logout.html')

@app.route('/ complaint-detail ')

def profile():

return render\_template(' complaint-detail.html')

@app.route('/ complaint-history')

def complaint-history():

return render\_template(' complaint-history ')

@app.route('/change -password')

def change-password():

return render\_template('change-password.html')

@app.route('/check-ability')

def check-ability():

return render\_template('check-ability.html')

@app.route('/dashbord')

def dashbord():

return render\_template('dashbord.html')

@app.route('/profile')

def change-passeord():

return render\_template(‘change-passeord.html')

@app.route('/getsubcat')

def getsubcat():

return render\_template('getsubcat.html')

if \_\_name\_\_ == '\_\_main\_\_':

app.run()

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="utf-8">

<meta http-equiv="X-UA-Compatible" content="IE=edge">

<meta name="viewport" content="width=device-width, initial-scale=1">

<meta name="description" content="">

<meta name="author" content="">

<title>Isue Complaint Management System</title>

<link href="css/bootstrap.min.css" rel="stylesheet">

<link href="css/half-slider.css" rel="stylesheet">

</head>

<body>

<!-- Navigation -->

<nav class="navbar navbar-inverse navbar-fixed-top" role="navigation">

<div class="container">

<!-- Brand and toggle get grouped for better mobile display -->

<div class="navbar-header">

<button type="button" class="navbar-toggle" data-toggle="collapse" data-target="#bs-example-navbar-collapse-1">

<span class="sr-only">Toggle navigation</span>

<span class="icon-bar"></span>

<span class="icon-bar"></span>

<span class="icon-bar"></span>

</button>

<a class="navbar-brand" href="#">COMPLAINT MANAGEMENT </a>

</div>

<!-- Collect the nav links, forms, and other content for toggling -->

<div class="collapse navbar-collapse" id="bs-example-navbar-collapse-1">

<ul class="nav navbar-nav">

<li>

<a href="http://localhost/Complaint Management System/users/">User Login</a>

</li>

<li>

<a href="http://localhost/Complaint Management System/users/registration.php">User Registration</a>

</li>

<li>

<a href="http://localhost/Complaint Management System/admin/">admin</a>

</li>

</ul>

</div>

<!-- /.navbar-collapse -->

</div>

<!-- /.container -->

</nav>

<!-- Half Page Image Background Carousel Header -->

<header id="myCarousel" class="carousel slide">

<!-- Indicators -->

<ol class="carousel-indicators">

<li data-target="#myCarousel" data-slide-to="0" class="active"></li>

<li data-target="#myCarousel" data-slide-to="1"></li>

<li data-target="#myCarousel" data-slide-to="2"></li>

</ol>

<!-- Wrapper for Slides -->

<div class="carousel-inner">

<div class="item active">

<!-- Set the second background image using inline CSS below. -->

<div class="fill" style="background-image:url('file:///C:/xampp/htdocs/Complaint%20Management%20System/img/c10.jpg');"></div>

<div class="carousel-caption">

</div>

</div>

<div class="item">

<!-- Set the third background image using inline CSS below. -->

<div class="fill" style="background-image:url('file:///C:/xampp/htdocs/Complaint%20Management%20System/img/c2.jpg');"></div>

<div class="carousel-caption">

</div>

</div>

</div>

<!-- Controls -->

<a class="left carousel-control" href="#myCarousel" data-slide="prev">

<span class="icon-prev"></span>

</a>

<a class="right carousel-control" href="#myCarousel" data-slide="next">

<span class="icon-next"></span>

</a>

</header>

<!-- Page Content -->

<div class="container">

<div class="row">

<div class="col-lg-12">

<div class="com">

<h1><b>WELCOME TO ISSUE CONTROL SYSTEM</b></h1>

<p>

<h2>The Easiest Way To Complain For Your Issue Online </h2><br> <b><i>Tired off getting ripped off? Fight back and file a customer complaint now.Sign Up Today And You Can Leave Complaints on various topics/company.</b> </i>

</p>

</div>

</div>

</div>

<style>

.com {

text-align: center;

}

</style>

<hr>

<!-- Footer -->

<footer>

<div class="row">

<div class="col-lg-12">

<p>Copyright &copy; 2023 ICS</p>

</div>

</div>

<!-- /.row -->

</footer>

</div>

<style>

.com {

text-align: center;

}

</style>

<!-- /.container -->

<!-- jQuery -->

<script src="js/jquery.js"></script>

<!-- Bootstrap Core JavaScript -->

<script src="js/bootstrap.min.js"></script>

<!-- Script to Activate the Carousel -->

<script>

$('.carousel').carousel({

interval: 5000 //changes the speed

})

</script>

</body>

</html>

**REGISTRAION PAGE**

<!DOCTYPE html>

<html lang="en">

<head>

include('includes/config.php');

if(strlen($\_SESSION['login'])==0)

{

header('location:index.php');

}

else{

if(isset($\_POST['submit']))

{

$uid=$\_SESSION['id'];

$category=$\_POST['category'];

$subcat=$\_POST['subcategory'];

$complaintype=$\_POST['complaintype'];

$state=$\_POST['state'];

$noc=$\_POST['noc'];

$complaintdetials=$\_POST['complaindetails'];

$compfile=$\_FILES["compfile"]["name"];

move\_uploaded\_file($\_FILES["compfile"]["tmp\_name"],"complaintdocs/".$\_FILES["compfile"]["name"]);

$query=mysqli\_query($bd, "insert into tblcomplaints(userId,category,subcategory,complaintType,state,noc,complaintDetails,complaintFile) values('$uid','$category','$subcat','$complaintype','$state','$noc','$complaintdetials','$compfile')");

$sql=mysqli\_query($bd, "select complaintNumber from tblcomplaints order by complaintNumber desc limit 1");

while($row=mysqli\_fetch\_array($sql))

{

$cmpn=$row['complaintNumber'];

}

$complainno=$cmpn;

echo '<script> alert("Your complain has been successfully filled and your complaintno is "+"'.$complainno.'")</script>';

}

?>

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="utf-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<meta name="description" content="">

<meta name="author" content="Dashboard">

<meta name="keyword" content="Dashboard, Bootstrap, Admin, Template, Theme, Responsive, Fluid, Retina">

<title>CMS | User Register Complaint</title>

<!-- Bootstrap core CSS -->

<link href="assets/css/bootstrap.css" rel="stylesheet">

<!--external css-->

<link href="assets/font-awesome/css/font-awesome.css" rel="stylesheet" />

<link rel="stylesheet" type="text/css" href="assets/js/bootstrap-datepicker/css/datepicker.css" />

<link rel="stylesheet" type="text/css" href="assets/js/bootstrap-daterangepicker/daterangepicker.css" />

<link href="assets/css/style.css" rel="stylesheet">

<link href="assets/css/style-responsive.css" rel="stylesheet">

<script>

function getCat(val) {

//alert('val');

$.ajax({

type: "POST",

url: "getsubcat.php",

data:'catid='+val,

success: function(data){

$("#subcategory").html(data);

}

});

}

</script>

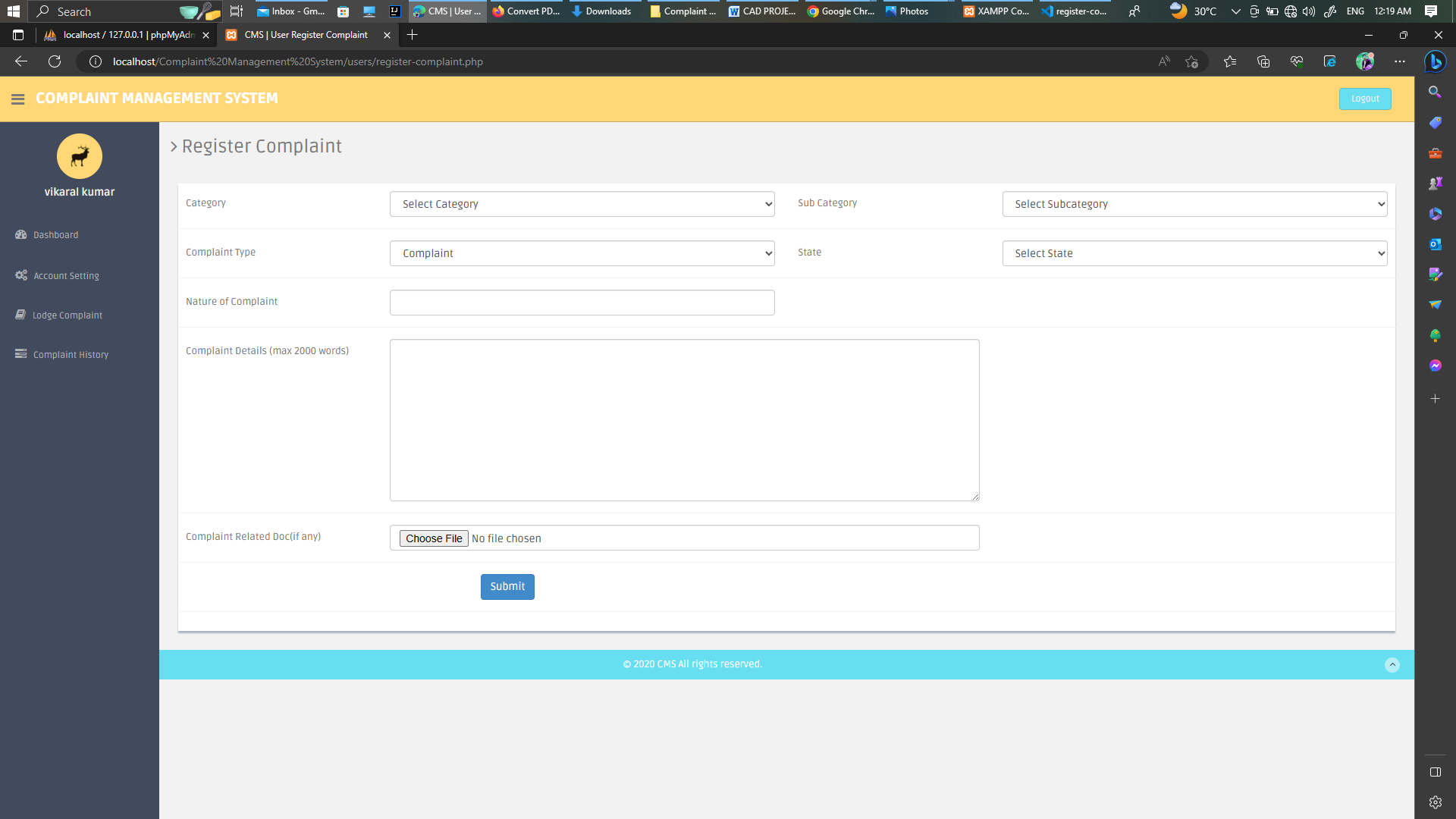
</head>

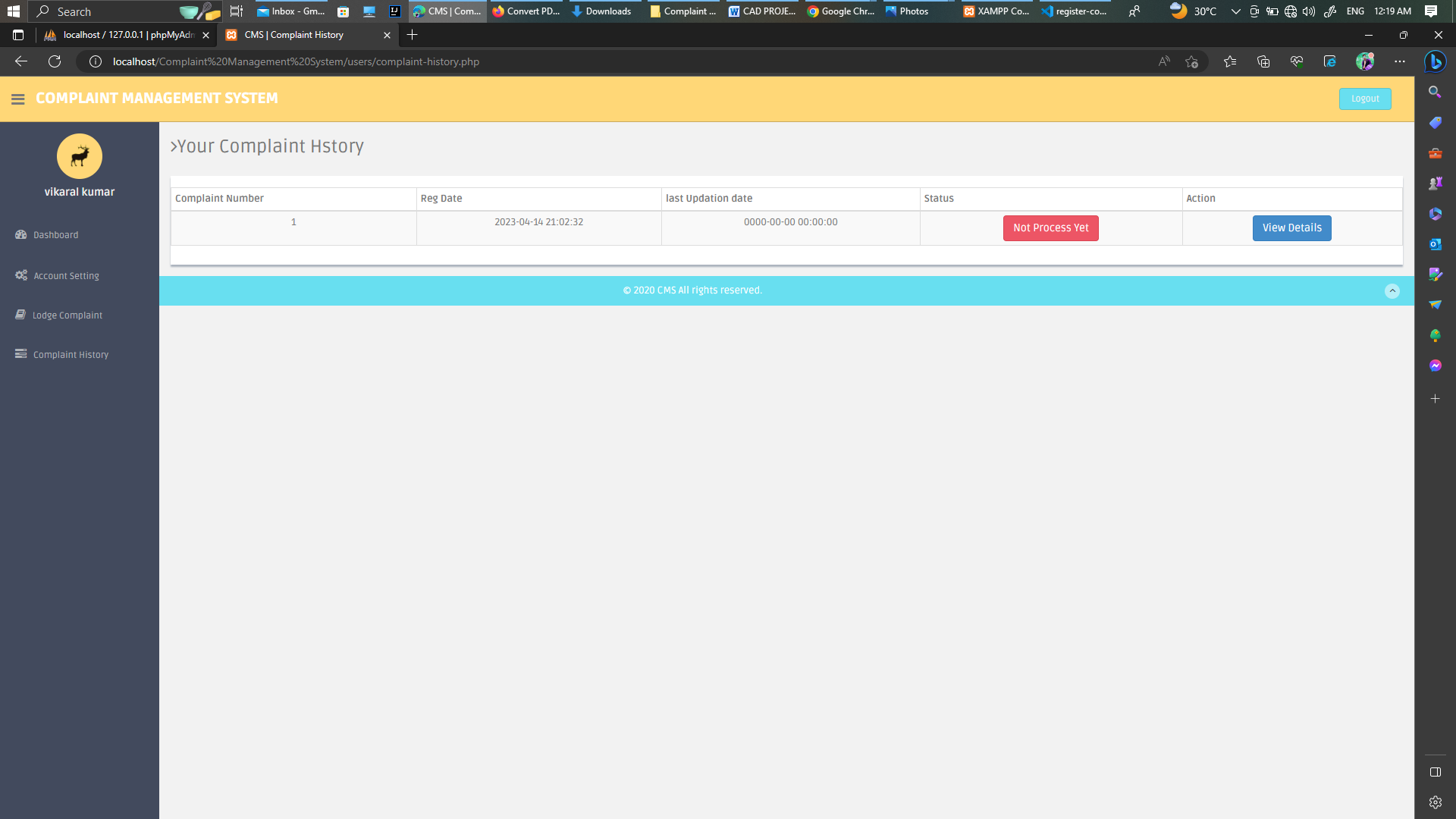
<body>

**SCREENSHOTS**









GitHub & Project Video Demo Link

<https://github.com/naanmudhalvan-SI/PBL-NT-GP-14371-1682764589>