Helpdesk Ticketing System

UIT2211 – SOFTWARE DEVELOPMENT PROJECT –I

A PROJECT REPORT

Submitted by

Devanithimaran E	**3122 23 5002 029**
Dharanikaran S	**3122 23 5002 030**
Dhaya Ananth M M	**3122 23 5002 031**
Dhivakaran S	**3122 23 5002 032**

SSN COLLEGE OF ENGINEERING,

KALAVAKKAM

JUNE 2024

Sri Sivasubramaniya Nadar College of Engineering
(An Autonomous Institution, Affiliated to Anna University)

BONAFIDE CERTIFICATE

Certified that this project titled "Helpdesk Ticketing System" is the bonafide work of "Devanithimaran E-3122235002029, Dharanikaran S - 3122235002030, Dhaya Ananth M M - 3122235002031, Dhivakaran S – 3122235002032 and is submitted for project viva-voce examination held on 18.06.2024.

Signature of examiner(s)

INDEX

PROBLEM STATEMENT

An organization like SSN College of Engineering wants to develop a Helpdesk ticketing system. Any member of the organization should be able to log a request/complaint regarding certain services. The system should assign a unique ticket number for each request. A ticket should be assigned to a service engineer based on the service type. A service engineer should be able to view all the tickets assigned to them. Once the service is completed, the service engineer will close the ticket. The management should be able to generate various reports for analyzing and improving the services.

ABSTRACT

The Helpdesk Ticketing System is a comprehensive software solution developed to address the specific needs of SSN College of Engineering. This system is designed to streamline the process of managing service requests and complaints, ensuring efficient resolution and improving overall service quality within the institution. The existing manual system for handling service requests results in inefficiencies, delays, and lack of accountability. To overcome these challenges, the college requires an automated system that ensures seamless request management and enhances operational efficiency.

The motivation behind developing the Helpdesk Ticketing System is to optimize workflow, improve service delivery, and enhance the experience for all members of SSN College of Engineering. By implementing a comprehensive software solution, the college aims to minimize administrative burdens, reduce response times, and ensure accountability. This will result in increased user satisfaction and improved service management. The objective is to deliver a user-friendly and efficient platform that enables members to log service requests or complaints, track their progress, and ensure timely resolution. The system will streamline the entire process from request logging to resolution, facilitate effective communication, and provide valuable insights through report generation.

The Helpdesk Ticketing System includes features such as automated ticket generation, assignment to service engineers based on service type, and progress tracking. Members of the organization can easily log requests or complaints, which are then assigned unique ticket numbers for tracking. Service engineers benefit from a clear view of all assigned tickets, enabling efficient management and resolution. Once a service is completed, the engineer closes the ticket, ensuring transparency and accountability. The management can generate various reports for analyzing service performance and identifying areas for improvement. This centralized platform fosters seamless communication among members, service engineers, and management, enhancing overall service quality and operational efficiency. The system is securely hosted and ensures data protection, offering reliable access through a web-based interface.

.

CLIENT DETAILS

Dr.Jaish P,

Department of Mathematics,

SSNCE,

Chennai – 603110.

Email ID: jaishp@ssn.edu.in

Phone Number: 9585366456

INTRODUCTION

In the dynamic environment of educational institutions, efficient management of service requests and prompt resolution of issues are crucial for maintaining smooth operations and ensuring satisfaction among students, faculty, and staff. The Helpdesk Ticketing System project, developed by a dedicated team of engineers, addresses these needs by offering a comprehensive software solution for SSN College of Engineering. The system aims to revolutionize the way service requests and complaints are managed, promoting seamless communication between service engineers, administrators, and users.

At the core of the Helpdesk Ticketing System is an intuitive admin page, where administrators can easily view and manage service requests and assignments. Each service engineer is granted personalized login credentials, enabling them to access the application and perform a variety of tasks. Service engineers can view all assigned tickets, update their status, and ensure timely resolution of issues. The system's ticket assignment functionality ensures that each service request is directed to the appropriate engineer based on the service type, facilitating efficient issue resolution.

Users, upon logging in, gain access to a straightforward interface where they can log service requests or complaints. Each request is assigned a unique ticket number, enabling users to track the progress of their requests. Users can also view the history of their previous requests and any communications related to their tickets. Email notifications are sent to confirm the creation of new tickets and to update users on the status of their requests, ensuring a seamless and transparent process.

Administrators play a pivotal role in the Helpdesk Ticketing System. They can generate various reports for analyzing service performance, identify areas for improvement, and ensure accountability. Additionally, administrators have the capability to manage user accounts, view detailed logs of all activities within the system, and ensure the overall security and efficiency of the ticketing process.

To enhance security and user convenience, the system incorporates a robust authentication mechanism. Moreover, the project has been successfully hosted on PythonAnywhere, offering secure and reliable access to the system.

The Helpdesk Ticketing System project aims to optimize the management of service requests, improve response times, and enhance the overall experience for all members of SSN College of Engineering. By leveraging technology and implementing a user-friendly software solution, the institution can streamline its processes, maintain accurate records, and deliver exceptional service to its community

REQUIREMENT ENGINEERING

SPRINT	EPIC	USER STORY #	REQUIREMENT/ USERSTORY	ESSENTIAL /DESIRABLE	DESCRIPTION OF REQUIREMENTS	REMARKS
1	User Authentication	1	Create login	Essential	By entering the correct credentials, the user must be able to access their data.	The login will be successful after entering the correct password.
1	User Authentication	2	New Registration	Essential	To create new user (client) login credentials.	New credentials will be generated after registration.
1	User Home Page	4	View Homepage	Essential	Every user will have their own customized home pages.	Each user cannot access any other users' homepage.
1	Complaint Management	5	Log Complaint	Essential	Users can log complaints regarding various issues.	A ticket number is generated and sent via email.
1	Complaint Management	6	Check Complaint Status	Essential	Users can check the status of their complaints using a ticket number.	Status is fetched from the database and displayed.
1	Complaint Management	7	Update Complaint Status	Essential	Technicians can update the status of complaints.	Status is updated in the database.
1	Agent Dashboard	8	View Assigned Tickets	Essential	Technicians can view tickets assigned to them based on their help topic.	Tickets are displayed sorted by preferred time.
1	Admin Dashboard	9	Admin Login	Essential	Admin can log in to access administrative functionalities.	Admin has access to view all feedback and resolved tickets.
1	Feedback Management	10	Submit Feedback	Essential	Users can submit feedback for resolved tickets.	Feedback is saved in the database.
1	Feedback Management	11	View Feedback	Desirable	Admin can view feedback submitted by users.	Feedback is displayed in the admin dashboard.

SPRINT	EPIC	USER STORY #	REQUIREMENT/ USERSTORY	ESSENTIAL /DESIRABLE	DESCRIPTION OF REQUIREMENTS	REMARKS
2	Agent Dashboard	12	Agent Login	Essential	Technicians can log in to view and manage their assigned tickets.	Technicians can update ticket statuses and view details.
2	Ticket Management	13	Load Data to Hash Table	Essential	Load existing complaint data into hash tables on startup.	Ensures efficient retrieval and management of tickets.
2	Ticket Management	14	Save Data from Hash Table	Essential	Save the current state of hash tables to JSON files before shutdown.	Prevents data loss and ensures persistence.
2	Ticket Management	15	Append to Resolved JSON	Essential	Append resolved tickets to their respective JSON files.	Helps in maintaining a record of resolved tickets.
2	Notification System	16	Send Email Notification	Essential	Send email notifications to users upon ticket creation and updates.	Users receive confirmation and status update emails.
2	Data Validation	17	Validate Ticket Information	Essential	Validate ticket information before saving to the database.	Ensures data integrity and prevents errors.
3	Admin Dashboard	18	View All Feedback	Desirable	Admin can view all feedback submitted by users.	Feedback is displayed in a summarized view on the admin dashboard.
3	Admin Management	20	Manage Users and Agents	Essential	Admin can view, add, and manage users and agents.	Admin has the ability to update roles and credentials.
3	Data Security	21	Session Management	Essential	Ensure secure session management for users, agents, and admin.	Prevent unauthorized access and maintain session integrity.

SPRINT	EPIC	USER STORY #	REQUIREMENT/ USERSTORY	ESSENTIAL /DESIRABLE	DESCRIPTION OF REQUIREMENTS	REMARKS
3	Ticket Assignment	22	Assign Tickets to Agents	Essential	Automatically assign tickets to agents based on help topics.	Ensures that the correct agent handles the appropriate tickets.
4	Data Analysis	23	Generate Reports	Desirable	Generate reports based on ticket data, including resolved and pending tickets.	Reports can be used for analysis and improving service efficiency.
4	User Interface	24	Improve UI for Ticket System	Desirable	Enhance the user interface for better usability and user experience.	UI improvements to make navigation and interaction more intuitive.
4	Feedback Management	25	Analyze Feedback	Desirable	Analyze feedback to identify common issues and areas for improvement.	Feedback analysis helps in improving the overall service quality.
4	Real-Time Updates	26	Real-Time Ticket Updates	Desirable	Implement real-time updates for ticket status changes.	Users and agents receive immediate updates on the status of tickets.

IMPLEMENTATION AND RISK MANAGEMENT

Name and Register Number: DEVANITHIMARAN E – 3122 23 5002 029
DHIVAKARAN S – 3122 23 5002 032

Role in the Project: Developer

a) Implementation

Sprint	Epic	Requirement / User Story	Remarks on implementation
1	Frontend Development & Routing	User friendly pages to perform various tasks	Created various pages and configured their respective routes
3	Form handling	Submit complaint and feedback	Created and managed form handling

b) Risk Management

Risk Management	Risk Description	Probability	Impact	Mitigation Plan
Integration Issues	Challenges in integrating frontend with backend for complaint logging and ticket system.	Medium	High	Conduct regular integration testing ,referring many websites,github and youtube
Coordination issues	Coordination of my pair coder	Medium	High	Discussing through Whatsapp and having clear conversation through phone calls
Client demand management	Meeting client demands	High	Medium	Completing essential requirements first and working on desirable ones

c) Test Log Report

TC id	RS #	Test case description/ condition	Test case input	Expected Output	Result (PASS/ FAIL)
1.	1	Verify complaint log form submission with all required fields filled.	Enter valid details for help topic, description, location, room no, mob no, preferred time, click submit.	Complaint is successfully logged, and confirmation email is sent.	PASS
2	3	Verify complaint log visibility after logging in.	Log in with a valid user account and navigate to complaint log.	User's complaint log is displayed with all relevant complaints listed.	PASS
3	2	Verify all essential details in all pages and proper functionality	Regular review of html,css and js code	All essential details are displayed with proper functionality	PASS

Name and Register Number: DHARANIKARAN S – 3122 23 5002 030

Role in the Project: Developer

a) Implementation

Sprint	Epic	Requirement / User Story	Remarks on implementation
1	Complaint Logging system	log a complaint with details such as help topic, description, location, and preferred time.	Implemented form handling, data validation, ticket ID generation, and integration with HashTable for efficient ticket management.
3	Email Notification System	receive an email notification when submit a complaint.	Implemented email sending functionality using SMTP and smtplib, ensuring users receive ticket confirmation emails upon complaint submission.
4	Feedback System	provide feedback for resolved tickets..	Designed and implemented the feedback system using a stack-based ADT to store and manage feedback entries efficiently.

b) Risk Management

Risk Management	Risk Description	Probability	Impact	Mitigation Plan
Monitoring & Testing	Difficulty in integrating HashTable with complaint logging system	Medium	High	Ensure thorough testing and debugging of the HashTable implementation
Monitoring & Testing	Email notification failures due to SMTP server issues	Medium	High	Monitor SMTP server status and maintain logs for email transactions.
Data Integrity	Feedback system inefficiencies leading to data loss or corruption	Low	Medium	data validation for feedback entries. Regularly back up feedback data.

c) Test Log Report

TC id	RS #	Test case description/ condition	Test case input	Expected Output	Result (PASS/ FAIL)
1.	5	Validate email notification upon complaint submission	Enter a proper email address while signing in	Confirmation email received by the user with the correct ticket details	PASS
2	9	Verify feedback submission for resolved tickets	Ticket ID: 181083, Rating: 5, Feedback: “Your service is excellent"	Feedback is recorded and stored in the system	PASS

Name and Register Number: DHAYA ANANTH M M – 3122 23 5002 031

Role in the Project: Developer

a) Implementation

Sprint	Epic	Requirement / User Story	Remarks on implementation
1.	JSON File Management	To manage JSON files for storing and retrieving data so that the application maintains data consistency.	Implemented functions to read and write JSON files, ensuring all necessary files are initialized and managed.
4	Resolved Tickets Handling	To manage resolved tickets separately so that they are stored and retrieved efficiently	Implemented logic to handle resolved tickets, ensuring they are stored in separate JSON files and can be retrieved as needed.

b) Risk Management

Risk Management	Risk Description	Probability	Impact	Mitigation Plan
Data Integrity Management	Risk of data corruption or loss during file read/write operations	Medium	High	Implement robust error handling, validation checks
KeyError	Encountered when trying to access a key that does not exist in a dictionary.	High	Medium	proper name convention with respect to the keys in hashtable

c) Test Log Report

TC id	RS #	Test case description/ condition	Test case input	Expected Output	Result (PASS/ FAIL)
1.	11	Ensure data is correctly written to and read from a JSON file	JSON data structure to write	Data is written to file and correctly read back	PASS

MEETING REPORTS

- Had in-person group meetings every Tuesdays and Thursdays.
- Had one team member meet the client once every 15 to 20 days, updating requirements.
- Had an online meet on 8th April 2024, regarding the design of front end.
- Had an online meet on 2nd May 2024, regarding data flow diagram.
- Had an online meet on 27th May 2024, regarding project report.
- All team members were present in all the meets.

RESULTS

```python
def submit_clog():
    data = request.get_json()
    name=None
    help_topic = data.get('help_topic')
    description = data.get('description')
    location = data.get('location')
    room_no = data.get('room_no')
    mob_no = data.get('mob_no')
    preferred_time = data.get('preferred_time')
    user_email = session.get('email')
    ticket=randint(100000,1000000)
    sign_up_data=read_json_file(signup_json_file_path)
    for data in sign_up_data:
        if data['email']==user_email:
            name=data['fullname']
            break
    if name:
        send_signin_email(user_email, ticket, name)
    else:
        return jsonify(success=False, message="User not found in signup data"), 404


    # Define the path for the help topic-specific JSON file
    complaint_json_file_path = os.path.join(current_directory, f'{help_topic}.json')

    # Ensure the JSON file exists
    ensure_file_exists(complaint_json_file_path)

    # Load existing complaints
    complaints = read_json_file(complaint_json_file_path)

    new_complaint={
        'help_topic': help_topic,
        'description': description,
        'location': location,
        'room_no': room_no,
        'mob_no': mob_no,
        'preferred_time': preferred_time,
        'ticket id':ticket,
        'user-email':user_email,
        'status': 'open'
    }
    # Append the new complaint
    complaints.append(new_complaint)

    # Save the updated complaints
    write_json_file(complaint_json_file_path, complaints)

     # Insert into hash table
    hash_tables[help_topic].insert(ticket, new_complaint)

    return jsonify(success=True), 200

```

The ‘submit_clog’ method handles the submission of a new complaint log (clog).It serves as the main entry point for users to report complaints, ensuring the system captures all necessary information in a consistent manner

```python
class FeedbackStack:
    def __init__(self):
        self.stack = []
    def push(self, feedback):
        self.stack.append(feedback)
    def pop(self):
        if self.is_empty():
            return None
        return self.stack.pop()
    def peek(self):
        if self.is_empty():
            return None
        return self.stack[-1]
    def is_empty(self):
        return len(self.stack) == 0

class FeedbackList:
    def __init__(self, file_path):
        self.file_path = file_path
        self.feedback_stack = FeedbackStack()
        self.load_feedback_data()

    def ensure_file_exists(self):
        if not os.path.exists(self.file_path):
            with open(self.file_path, 'w') as file:
                json.dump([], file)

    def load_feedback_data(self):
        self.ensure_file_exists()
        with open(self.file_path, 'r') as file:
            try:
                data = json.load(file)
                for feedback in reversed(data):
                    self.feedback_stack.push(feedback)
            except json.JSONDecodeError:
                pass  # Empty or invalid JSON file

    def save_feedback_data(self):
        feedback_data = self.get_all_feedback()
        with open(self.file_path, 'w') as file:
            json.dump(feedback_data, file, indent=4)

    def add_feedback(self, feedback):
        # Check if feedback already exists
        if feedback not in self.feedback_stack.stack:
            self.feedback_stack.push(feedback)
            self.save_feedback_data()

    def get_all_feedback(self):
        return list(reversed(self.feedback_stack.stack))

```

This snippet shows the creation and management of List ADT and stack for efficiently handling feedbacks.

```python
class HashTable:
    def __init__(self, size):
        self.size = size
        self.table = [[] for _ in range(size)]

    def _hash(self, key):
        return int(key) % self.size

    def insert(self, key, value):
        hash_key = self._hash(key)
        key_exists = False
        for i, kv in enumerate(self.table[hash_key]):
            k, v = kv
            if key == k:
                key_exists = True
                break
        if key_exists:
            self.table[hash_key][i] = (key, value)
        else:
            self.table[hash_key].append((key, value))

    def delete(self, key):
        hash_key = self._hash(key)
        for i, kv in enumerate(self.table[hash_key]):
            k, v = kv
            if key == k:
                del self.table[hash_key][i]
                return True
        return False

    def retrieve(self, key):
        hash_key = self._hash(key)
        for k, v in self.table[hash_key]:
            if k == key:
                return v
        return None

    def get_all(self):
        all_items = []
        for bucket in self.table:
            all_items.extend(bucket)
        return [item[1] for item in all_items]

```

This snippet shows the creation and management of hash tables for tickets.

SCREENSHOTS OF USER-INTERFACE

HOMEPAGE:

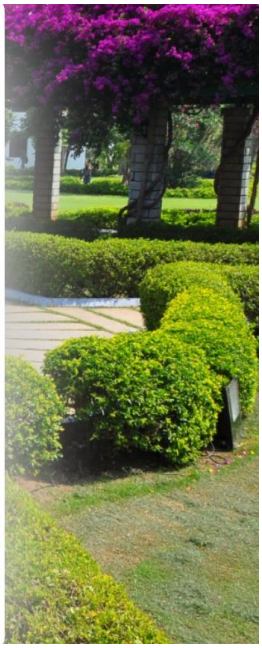
WELCOME TO SSN HELPDESK

Welcome to the Campus Helpdesk, your one-stop destination for resolving all campus-related issues promptly and efficiently. From WiFi connectivity to electrical glitches, plumbing emergencies, and more, we're here to ensure a seamless experience for all members of our campus community.

STEPS FOR LOGGING A PROBLEM:

1. Click open new ticket
2. Sign in using your credentials
3. Select appropriate help topic
4. Fill in the mandatory fields
5. Click create ticket

Admin-login | Client-login | Technician-login



ADMIN,CLIENT,TECHNICIAN LOGIN:

Admin login

Enter your login details

Username or Email Address

Enter your Password

Remember me

Log in

Client login

Enter your login details

Username or Email Address

Enter your Password

Remember me

Log in

Not a member? Sign up

Technician login

Enter your login details

Username or Email Address

Enter your Password

Remember me

Log in

TECHNICIAN'S DASHBOARD:

Civil Tickets

Help Topic	Description	Location	Room No	Mobile No	Preferred Time	Ticket ID	User Email	Status
civil_mason	Wood broken	gents_hostel	e-34	9344246599	01:06	844648	dharanikaran2310582@ssn.edu.in	In I
civil_mason	Floor tile damage	gents_hostel	F-232	9095804444	01:32	246731	dharanikaran2310582@ssn.edu.in	In I

Carpentery Tickets

Help Topic	Description	Location	Room No	Mobile No	Preferred Time	Ticket ID	User Email	Status
carpentery	Cupboard broken	gents_hostel	F-678	9095804444	01:32	713344	dharanikaran2310582@ssn.edu.in	Oper
carpentery	door fault	gents_hostel	E - 51	7890653531	06:40	251893	deva2310887@gmail.com	Oper

Plumbing Tickets

Help Topic	Description	Location	Room No	Mobile No	Preferred Time	Ticket ID	User Email	Status
plumbing	wash basin is broken	gents_hostel	p-90	6268969512	05:56	226439	bm110018@thebvb.com	Op
plumbing	Toilet flush issue	gents_hostel	F-990	9095804444	09:32	669784	dharanikaran2310582@ssn.edu.in	Op

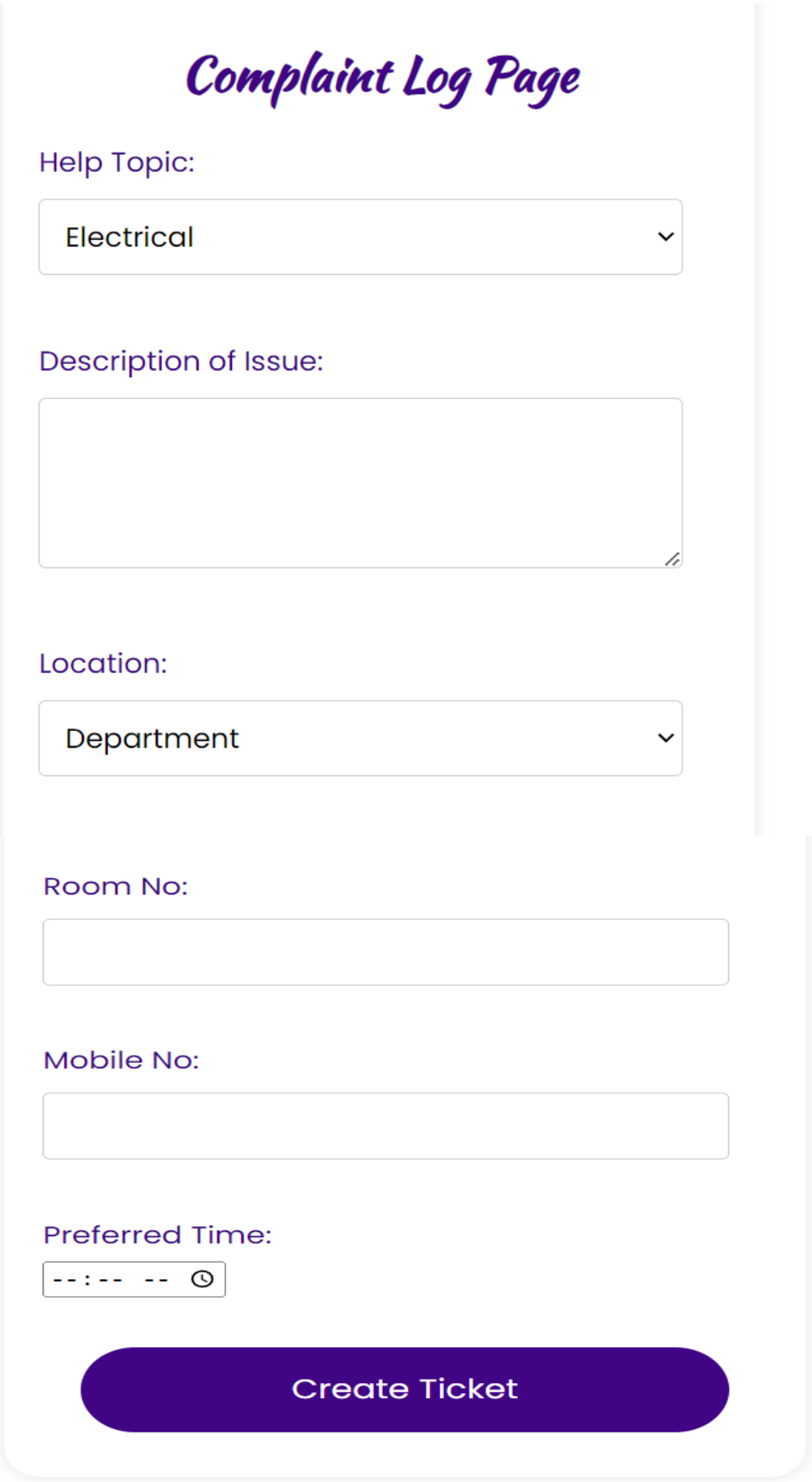
wifi-problem Tickets

Help Topic	Description	Location	Room No	Mobile No	Preferred Time	Ticket ID	User Email	Statu
wifi_problem	Firewall issue	gents_hostel	E-47	9812763450	00:23	443994	chennakesavansuresh1007@gmail.com	Op
wifi_problem	Router is broken fix me a new one	gents_hostel	S78	9865989638	00:28	284470	kungfugouthaman@gmail.com	Op

Electrical Tickets

Help Topic	Description	Location	Room No	Mobile No	Preferred Time	Ticket ID	User Email	Status
electrical	fan not working	ladies_hostel	e44	9095804444	19:21	601360	anandhalaxmi2310810@ssn.edu.in	Oper
electrical	working speed of fan is slow	ladies_hostel	f-90	7676766620	19:34	394390	prarthna2310607@ssn.edu.in	Oper

COMPLAINT LOG PAGE:



ADMIN PAGE:

Client Feedback Data

Ticket ID	Rating	Feedback	Details
181083	5	Your service is excellent	Ticket Details
920527	5	good	Ticket Details
465895	5	Good	Ticket Details
621749	4	good..	Ticket Details
550377	5	excellent service	Ticket Details
504051	3	Average	Ticket Details
147332	5	Good	Ticket Details

TICKET STATUS(RESOLVED/OPEN/INPROGRESS):

Resolved Ticket

Help Topic	electrical
Description	Current is fluctuating
Location	gents_hostel
Room No	S-66
Mobile No	7904548485
Preferred Time	18:47
Ticket ID	777486
User Email	dharanikaran44@gmail.com
Status	resolved

Ticket Status

Help Topic	carpentery
Description	Cupboard broken
Location	gents_hostel
Room No	F-678
Mobile No	9095804444
Preferred Time	01:32
Ticket ID	713344
User Email	dharanikaran2310582@ssn.edu.in
Status	in_progress

Ticket Status

Help Topic	carpentery
Description	Cupboard broken
Location	gents_hostel
Room No	F-678
Mobile No	9095804444
Preferred Time	01:32
Ticket ID	713344
User Email	dharanikaran2310582@ssn.edu.in
Status	open

FEEDBACK SUBMISSION FORM:

Submit Feedback

Rating:

Good

Feedback:

excellent work

Submit Feedback

CLIENT EVALUATION REPORT

Project's Name: Helpdesk Ticketing System
Name of the client: P.Jaish

Rating System - 1: Strongly disagree 2: Disagree 3: Neutral 4: Agree 5: Strongly Agree

Questions	1	2	3	4	5
The problem was well discussed and the requirements and goals were clear.					✓
The project plan was well-defined and communicated from the start.					✓
The resources were adequate for achieving the goals.					✓
The original timeline was realistic and was followed.				✓	
The teamwork was well demonstrated.				✓	
The client was communicated on regular intervals and given updates on the progress of the project.				✓	
The expected project requirements have been satisfied.					✓

CONCLUSION

In conclusion, the Helpdesk Ticketing System project has effectively tackled the challenges associated with managing service requests within our organization. By developing and implementing this robust software solution, we have streamlined workflow processes, enhanced communication, and significantly improved the efficiency of issue resolution.

The project has delivered an intuitive user interface, allowing users to easily submit and track their service requests. The categorization of requests by type (e.g., carpentry, masonry, plumbing, internet, and electrical services) ensures that tasks are directed to the appropriate technicians, facilitating swift and accurate responses. Technicians, in turn, benefit from organized queues and detailed task descriptions, enabling them to prioritize and address issues effectively.

By integrating features such as real-time notifications and status updates, the system ensures that users are kept informed throughout the resolution process. The inclusion of reporting and analytics tools provides valuable insights into service performance, helping management to identify trends, allocate resources efficiently, and make data-driven decisions.

The use of secure, scalable Azure hosting guarantees reliable access to the system, promoting continuous availability and flexibility to accommodate future growth. Enhanced security measures, including authentication protocols and data encryption, ensure the protection of sensitive information.

Looking ahead, the Helpdesk Ticketing System can be further refined by incorporating additional features and functionalities based on user feedback and evolving organizational needs. Continuous monitoring, maintenance, and updates will ensure the system remains responsive and effective in addressing future requirements.

Overall, the Helpdesk Ticketing System project has significantly transformed the way our organization manages service requests. By leveraging advanced technology, we have successfully addressed the specific challenges of our helpdesk operations, resulting in improved efficiency, better communication, and a seamless experience for both users and technicians.

REFERENCES

- https://www.stackoverflow.com/
- https://youtube.com/playlist?list=PL-osiE80TeTs4UjLw5MM6OjgkjFeUxCYH&si=MzepalBTSZBv5K0D
- https://www.geeksforgeeks.org/learn-data-structures-and-algorithms-dsa-tutorial/)