CAPSTONE PROJECT REPORT

NAME: D.sai prasanna kumar

REGISTER NO: 192210609

COURSE CODE/NAME: CSA0917- Programming with java with Strings

PROJECT TITLE : Develop a Java applet for an "On-Line Help Desk

OBJECTIVE:

With the goal of creating a Java applet for a "On-Line Help Desk" system, this project aims to improve customer satisfaction through the simplification and efficiency of the technical assistance process. The main goal is to give consumers an easy-to-use interface via which they can track their progress, submit help requests, and interact in real time with support personnel. Through features like ticket management, prioritizing, agent assignment, and escalation procedures, it also makes quick issue resolution possible. Moreover, the possibility of having a knowledge base or FAQ part gives users the ability to access self-help tools and maybe handle common problems on their own. The ultimate goal of this project is to close the communication gap that exists between users and support personnel in order to provide a more effective and timely environment for handling technical questions.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| TASK/DURATION | **21.02.2024** | **22.02.2024** | **23.02.2024** | **24.02.2024** | **26.02.2024** | **27.02.2024** |
| LITERATURE SURVEY |  |  |  |  |  |  |
| MODULE 1 DESIGN |  |  |  |  |  |  |
| MODULE 2 DESIGN |  |  |  |  |  |  |
| MODULE 1 IMPLEMENTATION |  |  |  |  |  |  |
| MODULE 2 IMPLEMENTATION |  |  |  |  |  |  |
| DEMO & PRESENTATION |  |  |  |  |  |  |

INTRODUCTION:

Offering quick and effective support services is essential for businesses to keep customers happy and loyal in today's fast-paced digital environment. The foundation for promptly and efficiently handling user-posted questions, concerns, and technical issues is a "On-Line Help Desk" system. As software systems and technology get more complicated, consumers frequently run into problems that call for professional advice and support. Thus, it is crucial to provide a user-friendly platform that makes asking for assistance easier and allows for smooth communication between users and support personnel.

The goal of this project is to create a Java-based "On-Line Help Desk" system that enables users to communicate with support agents in real-time, track the progress of their requests, and submit help requests. Our goal is to provide a scalable and resilient solution that can handle a wide range of user demands while maintaining security, effectiveness, and responsiveness by utilizing Java technology.

Comprehensive ticket management features, such as request prioritizing, support agent assignment, and escalation protocols for critical problems, will be included in this system. It will also include a knowledge base or FAQ part to give users access to self-help tools and solutions to frequently encountered issues, lessening the burden on support workers and boosting user autonomy.

We hope to enhance customer happiness, streamline the process of answering user concerns, and raise the standard of the organization's support services overall with the creation of this "On-Line Help Desk" system. We want to create by putting user experience, efficiency, and security first.

LITERATURE SURVEY:

1.ON USING SNA TECHNIQUES FOR ENHANCING PERFORMANCES OF ON-LINE HELP DESKS

This paper provides a state of art survey of works using social network analysis (SNA) techniques for improving performances of an on-line help desk by” Rushed Kanawati”

2. Design and Implementation of Web-based Multimedia Techniques for Construction Education

THE CONSTRUCTION INDUSTRY, as with other branches of manufacturing, is becoming increasingly complex and technologically sophisticated. The change has been dramatic in recent years and is likely to continue to increase due to rapid developments in, for example, the information technologies, materials science, and computerbased modeling tools. The problem has been made even more critical by increased performance expectations (such as lower maintenance costs and reduced environmental impact) from both the sponsors of construction and society. By” NABIL KARTAM and KHALED AL-RESHAID”

3.Internet‐based information and communication systems on remote construction projects based on help desk. In an attempt to bring the unique talents of various construction industry project participants together in a more productive and integrated manner, the Online Remote Construction Management (ORCM) project commenced in July by” A.wippert”.

4.Virtual Community Practice Toolkits Using 3D Imaging Technologies. Extending visual communications to the third dimension (3D) has been a dream over decades. The ultimate goal of the viewing experience is to create the illusion of a real environment in its absence by” George Triantafylliadis”

# 5. Opportunities in Web-Based Teaching: The Future of Education. Web-based teaching technology has become a popular tool for many institutions in this decade. It can be used for every educational level from K-12 to higher education and distance education in many different fields

# By” Orasa Tetiwat, Magid Igbaria”

6.Systems for Help Desks:Examine the body of research on help desk systems, taking note of its features, architecture, and user interfaces.

Examine several strategies for knowledge base integration, real-time communication, and ticket management in help desk systems.

7.Designing User Interfaces:To create interfaces that are easy to use and intuitive, research the principles of user interface design.

Examine methods for creating forms that can be used to track ticket status, submit support requests, and browse knowledge base articles.

8.InstantaneousCommunication:Examin real-time communication technologies like Comet or WebSockets that are appropriate for Java applets.

Review the research on the utilization of real-time chat features and user experience.

CODING:

import java.util.ArrayList;

import java.util.List;

class HelpRequest {

private int id;

private String description;

private boolean resolved;

private String assignedTo;

public HelpRequest(int id, String description) {

this.id = id;

this.description = description;

this.resolved = false;

this.assignedTo = "Unassigned";

}

public int getId() {

return id;

}

public String getDescription() {

return description;

}

public boolean isResolved() {

return resolved;

}

public void setResolved(boolean resolved) {

this.resolved = resolved;

}

public String getAssignedTo() {

return assignedTo;

}

public void setAssignedTo(String assignedTo) {

this.assignedTo = assignedTo;

}

}

private List<HelpRequest> helpRequests;

public HelpDesk() {

this.helpRequests = new ArrayList<>();

}

public void submitHelpRequest(int id, String description) {

HelpRequest request = new HelpRequest(id, description);

helpRequests.add(request);

System.out.println("Help request submitted successfully. Your request ID is: " + id);

}

public void trackHelpRequest(int id) {

for (HelpRequest request : helpRequests) {

if (request.getId() == id) {

System.out.println("Request ID: " + request.getId());

System.out.println("Description: " + request.getDescription());

System.out.println("Status: " + (request.isResolved() ? "Resolved" : "Pending"));

System.out.println("Assigned To: " + request.getAssignedTo());

return;

}

}

System.out.println("Help request with ID " + id + " not found.");

}

public void assignHelpRequest(int id, String assignedTo) {

for (HelpRequest request : helpRequests) {

if (request.getId() == id) {

request.setAssignedTo(assignedTo);

System.out.println("Help request with ID " + id + " assigned to " + assignedTo);

return;

}

}

System.out.println("Help request with ID " + id + " not found.");

}

public void resolveHelpRequest(int id) {

for (HelpRequest request : helpRequests) {

if (request.getId() == id) {

request.setResolved(true);

System.out.println("Help request with ID " + id + " resolved successfully.");

return;

}

}

System.out.println("Help request with ID " + id + " not found.");

}

}

public class Main {

public static void main(String[] args) {

HelpDesk helpDesk = new HelpDesk();

helpDesk.submitHelpRequest(1, "I am facing issues with my internet connection.");

helpDesk.trackHelpRequest(1);

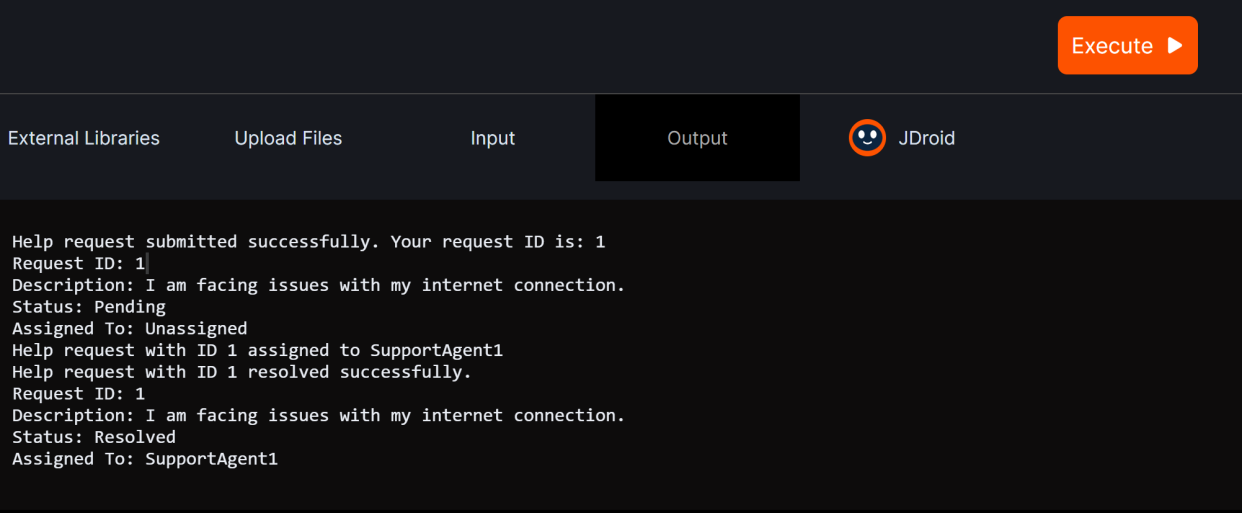
helpDesk.assignHelpRequest(1, "SupportAgent1");

helpDesk.resolveHelpRequest(1);

helpDesk.trackHelpRequest(1);

}

}



Conclusion:

To sum up, creating a Java applet for a "On-Line Help Desk" system offers a reliable way to make technical help and user support easier. Through the use of intuitive interfaces and instantaneous communication features, the applet expedites the process of submitting support requests, monitoring their progress, and corresponding with support personnel efficiently. Crucial functionalities including ticket management, prioritizing, support agent assignment, and escalation protocols guarantee that pressing problems are attended to and resolved promptly. Moreover, adding a knowledge base or frequently asked questions area enables users to solve typical issues on their own, decreasing reliance on support personnel and raising general customer satisfaction. Overall, using this applet increases user satisfaction by enhancing the effectiveness of answering questions from users and by contributing to a positive user experience.