

Project Title: Help Desk Management System

1. Introduction

A Help Desk Management System is a tool that allows users to raise support tickets and helps administrators manage and respond to those tickets effectively. In this project, we have developed a simple web-based help desk system using Python (Flask framework) and SQLite for backend data storage. The system is useful for managing technical queries or issues raised by users in a structured way.

2. Objective:

- To develop a simple and intuitive support ticketing system.
 - To allow users to submit technical issues via a web form.
 - To enable the admin to view, manage, and close support tickets.
 - To implement a ticket status tracking feature (Open/Closed).
-

3. Tools and Technologies Used:

- **Programming Language:** Python 3
 - **Framework:** Flask
 - **Database:** SQLite
 - **Frontend:** HTML, CSS, Bootstrap
 - **IDE Used:** VS Code / Jupyter Lab / PyCharm
-

4. System Description

This Help Desk Management System includes the following features:

- A ticket submission form for users.
- A responsive user interface using Bootstrap.
- Admin interface to view and close tickets.

- Tickets stored in a SQLite database with a status field (Open/Closed).
 - Basic analytics like ticket count and filtering.
-

5. Steps Followed in the Project:

Step 1: Environment Setup

Installed Flask and created the basic project structure with templates and static folders.

Step 2: Designed User Interface

Developed two main pages:

- User Page: Allows users to enter their name, email, and issue description.
- Admin Page: Displays submitted tickets in tabular format with the option to close them.

Step 3: Created Backend Logic

- Used Flask routes for different pages (/ , /submit, /admin, /close_ticket/<id>).
- Connected to SQLite and created a table tickets with fields like id, name, email, issue, and status.

Step 4: Database Integration

- Used SQLite commands to insert, fetch, and update ticket data.
- Default status for a ticket is "Open". Admin can change it to "Closed".

Step 5: Applied Styling

- Used Bootstrap and custom CSS to enhance the appearance.
 - Used badges and buttons for ticket status and actions.
-

6. Screenshots:

1. User Ticket Submission Form:

The screenshot shows a web browser window with two tabs: 'Help_Desk_Management_System' and 'Submit Ticket'. The address bar shows '127.0.0.1:5000'. The page has a blue header with 'Help Desk'. The main content area is titled 'Submit a Support Ticket' and contains a form with three input fields: 'Your Name', 'Your Email', and 'Describe your issue...'. Below the form is a blue 'Submit Ticket' button. At the bottom of the form is a grey bar labeled 'Admin Panel'. The Windows taskbar is visible at the bottom, showing the time as 11:53 PM on 4/30/2025.

Help Desk

Submit a Support Ticket

Your Name

Your Email

Describe your issue...

Submit Ticket

Admin Panel

2. Admin panel:

The screenshot shows a web browser window with two tabs: 'Help_Desk_Management_System' and 'Admin Panel'. The address bar shows '127.0.0.1:5000/admin'. The page has a blue header with 'Help Desk'. The main content area is titled 'Admin Panel - View Tickets' and contains a table with five rows of ticket data. Each row has columns for Name, Email, Issue, Status, and Action. The Status column shows 'Open' in green buttons, and the Action column shows 'Close' in red buttons. At the bottom of the table is a blue button labeled 'Back to Home'.

Help Desk

Admin Panel - View Tickets

Name	Email	Issue	Status	Action
kashish kazi	kashishkazi002@gmail.com	i want help for repairing my pc	Open	Close
kjhsck	kk@gmail.com	widhiw	Open	Close
Kashish	kashishkazi002@gmail.com	bfbv	Open	Close
kash	kashishkazi002@gmail.com	Open	Close
ABC	ABC@gmail.com	ticketing issue	Open	Close

Back to Home

7. Conclusion :

The Help Desk Management System simplifies the support process. - Flask and SQLite combination made the system light and fast. - Admin can manage tickets efficiently through the web interface. - Bootstrap ensured responsiveness across devices.

8. Future Scope :

Add authentication for admin login. - Send email notifications for ticket submission and resolution. - Integrate with a chatbot or AI system for auto-responses. - Export tickets as CSV or Excel reports

9. References :

-Flask Documentation: <https://flask.palletsprojects.com>

- Bootstrap Framework: <https://getbootstrap.com>

- SQLite Docs: <https://sqlite.org>