

Secure AI-Powered Customer Service Chatbot for E-commerce

Contributors

- Satpute Aniket Tukaram
- Harshit Pant
- Mahin Bansal

Overview

- Develop a secure, AI-powered customer service chatbot for an e-commerce platform, focusing on protecting customer data, preventing security breaches, and providing safe, automated customer support.s

Tech Stack

- Frontend: ReactJS
- Backend: FastAPI
- Database: SQLite

Features

- Secure login and registration system
- Secure chatbot for customer service
- Flagging of malicious logins and keystrokes
- Secured authentication and authorization

Installation

1. Clone the repository
 - `git clone https://github.com/anikettsatpute/cyber_security_project.git`
 - `cd cyber_security_project`
2. Install the required dependencies
 - `pip install -r requirements.txt`
 - `npm install`
3. Run the backend server
 - `cd backend`
 - `source venv/bin/activate`
 - `uvicorn main:app --reload`
4. Run the frontend server
 - `npm start`
5. Visit `localhost:3000` in your browser

Routes

- `/`: login page
- `/register`: registration page
- `/chatbot`: chatbot page
- `/admin`: admin page

File Structure

- Database: `backend/ecommerce.db`
- Controllers: `backend/controller.py`
- Models: `backend/models.py`
- Main: `backend/main.py`
- AI Model: `backend/AIModels/final_model.py`

API Endpoints

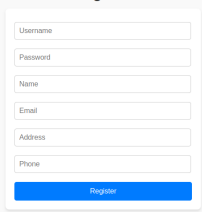
- `/login`: POST request to login
- `/register`: POST request to register
- `/chatbot`: POST request to chatbot
- `/admin`: GET request to admin
- `/loginAnomalies`: GET request to get login anomalies
- `/keystroke`: GET request to get keystroke anomalies
- `/logout`: GET request to logout

Cookies

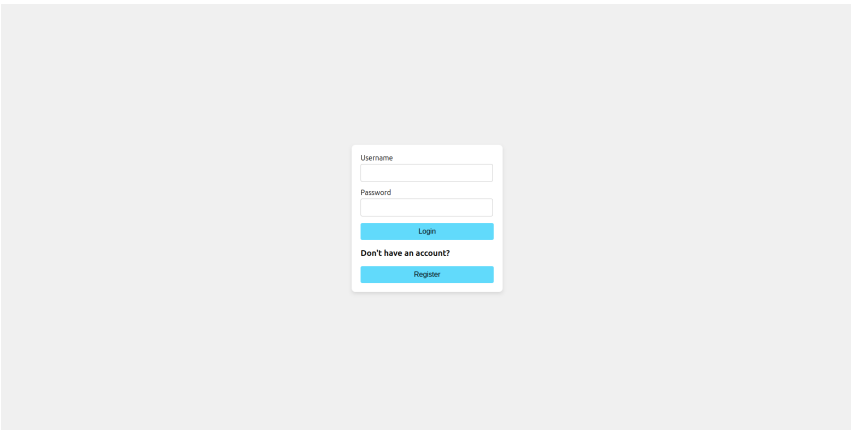
- HTTPOnly cookies are used for session management
- Secure and SameSite cookies are used for security
- The cookies contain access tokens created from `user_id` and timestamp

Screenshots

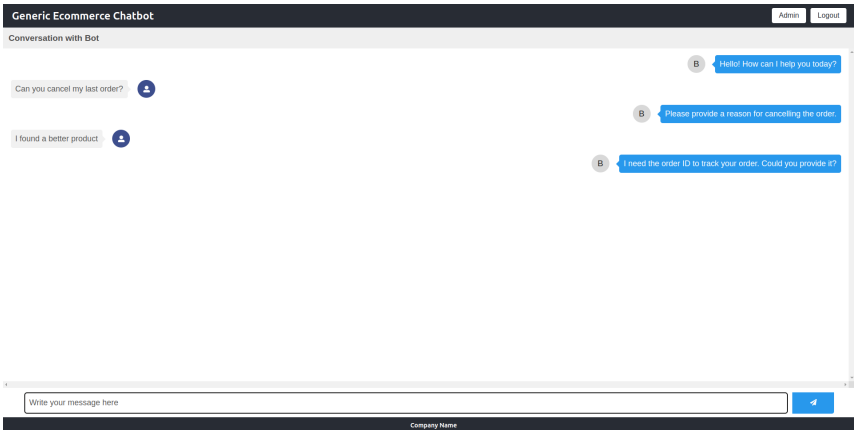
- Register Page



The screenshot displays a 'Register' form centered on a light gray background. The form is a white rectangle with a thin gray border. At the top, the word 'Register' is written in a small, bold, black font. Below this, there are six input fields, each with a placeholder label: 'Username', 'Password', 'Name', 'Email', 'Address', and 'Phone'. The fields are arranged vertically. At the bottom of the form is a blue button with the word 'Register' in white text.



- Login Page



- Chatbot Page

Generic Ecommerce Chatbot

Admin

Logout

UserID	Rating	IP Address	Rating
20101	10.65143389141191	127.0.0.1	1.812953915288873
1200	10.65143389141191	125.210.219.251	1.812953915288873
1201	10.204664992087709	12.95.78.49	1.812953915288873
1630	10.032995075013474	127.183.228.194	1.812953915288873
1987	10.032995075013474	141.88.41.250	1.812953915288873
142	9.738336733528175	150.46.26.134	1.812953915288873
358	9.738336733528175	152.114.1.120	1.812953915288873
544	9.738336733528175	154.90.205.202	1.812953915288873
1796	9.738336733528175	155.232.32.129	1.812953915288873
651	9.539192008409447	157.196.76.105	1.812953915288873
912	9.539192008409447	159.161.134.88	1.812953915288873
328	9.539192008409447	161.194.173.32	1.812953915288873
347	9.539192008409447	163.187.198.75	1.812953915288873
486	9.539192008409447	170.174.113.45	1.812953915288873
603	9.539192008409447	182.124.120.67	1.812953915288873
64	9.539192008409447	185.26.50.188	1.812953915288873
1265	9.539192008409447	197.143.217.46	1.812953915288873
1672	9.539192008409447		
1151	9.539192008409447		
1699	9.539192008409447		

Company Name

- Admin Page

References

- FastAPI Documentation

- [ReactJS Documentation](#)
- [SQLite Documentation](#)
- [Secure Cookies](#)