

HelpDesk Chat Client

Summary:

We are developing a BOT for our company. The BOT will be able to transfer the chat to a human agent.

The purpose of this module of our project is to build the Front End web interface for Agents to interact with customers.

BOT logic and client to bot communication is NOT a part of this module/job.

All Chat messages will be saved to a database.

What we need is a Web Application that provides the screen for the agents to Log In, check messages that needs human replies, and chat with those customers.

Inbound messages will be read from database, and outbound messages will be saved to a database. All further communication from database to users will be developed by other team.

Framework:

We wish this application to be written with .NET Core, C#, Entity Framework, Local DB (SQL-Lite), MVC with Razor Pages. This is a must as its the environment our team can handle further support.

Model:

Our Model has 3 classes. Classes are already written and we need them to be used, as other parts of the project are already under development. We will provide you with our classes files:

[Chat Users.cs](#)
[Chat Messages.cs](#)
[Agents.cs](#)

Screens:

Login Screen:

Simple page asking email and password for Agents to access the application.

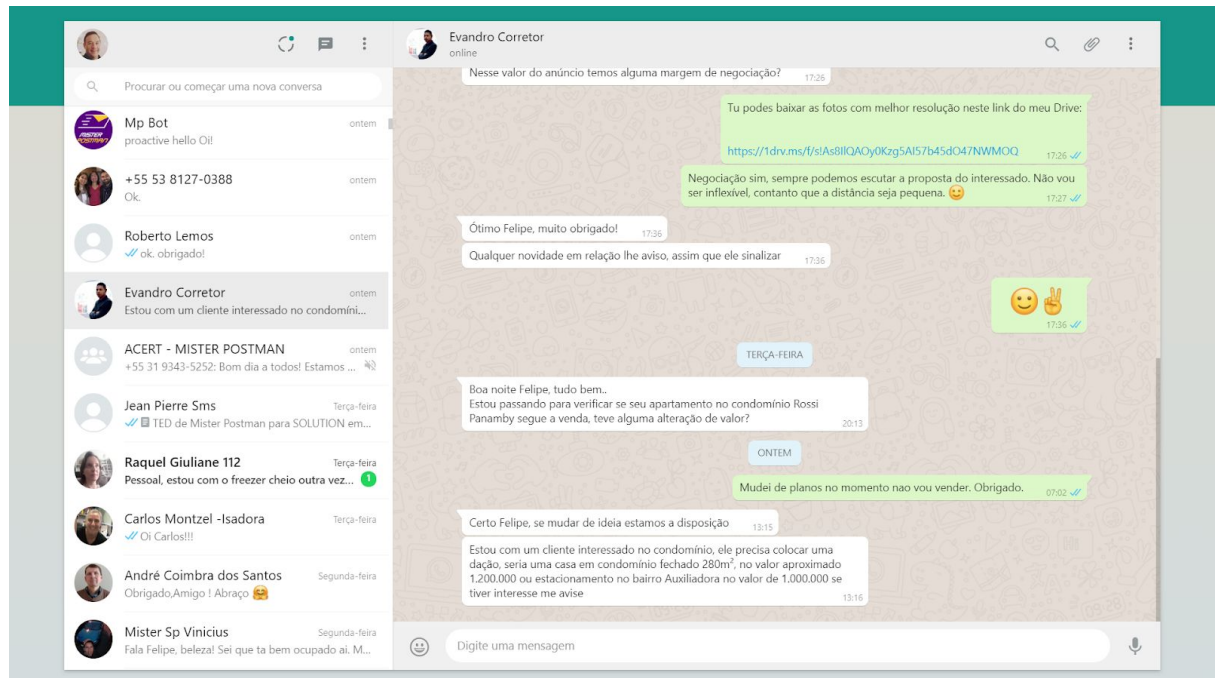
Access control will be made in a simple manner, with passwords saved at the database

Admin Screen:

Simple but elegant page where administrators can list, edit and delete Agent records.

Chat Screen:

This is the main screen of the application, where Agents will see all chats, and will be able to interact with customers. We wish a screen with the same look and feel of WhatsApp web interface. Bellow you may see an example as reference:



On the left panel we have a list of Users, sorted by date and time of last activity. Panel shows name and summary of last message. We WON'T have profile picture from users, so pictures circles may be changed to colored circles with users initials (first letter of name). Also it shows date/time of last message.

On the right panel we will see all inbound and outbound messages from each user. Messages are loaded in blocks so the interface responds quickly. When an agent scroll up message history, in case it reach the start of the block, script will load another block of older messages.

Screen activities should use JQuery/JavaScript functions to load messages and refresh data without page reloading.

On the bottom, there is a form where the agent can send text messages. Outbound messages should just be saved to database (further communication is NOT in the scope of this module - as it will be triggered later by other developers).

Attachment Files

Chat screen must also deal with attached files, that is, messages that carries image, voice, or other format files. In those case, file URL will be saved at the database. Image files will be rendered within the chat interface. Voice files should be played easily. Other format files should be downloads.

When sending image files, Chat Interface needs to upload file, save to a local path, and save record to database.

We will also need an applet to record voice from the agent, upload and save audio file to a server local path, with corresponding database record.

Menus

Chat screen is the main screen of the application.

After logging in, all Agents should be redirected to Chat Screen.

Normal Agents (there is an Agent Type field) will have a Menu where they can just change their name, password, and nickname.