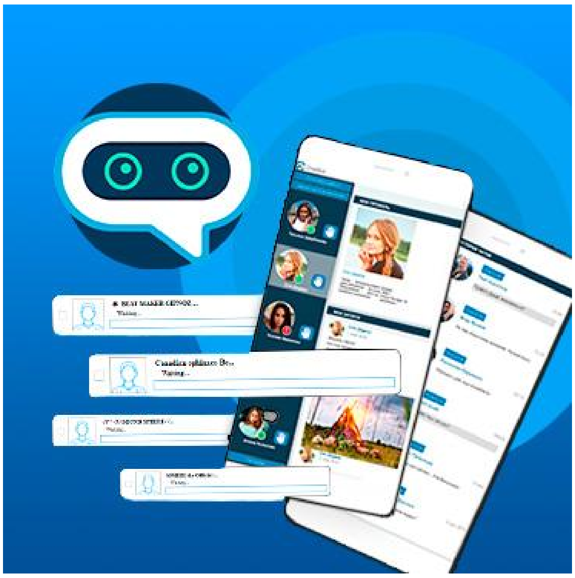
The QuBot Platform by QuData

Introduction

QuBot Platform was created to help people be more communicative at work, and give companies and organizations of all types and sizes extra benefits from using bots and automation.

A chatbot, or conversational bot, is an automated service that customers can communicate with using a button-based interface, free text input, or voice. Chatbots can be placed on many channels: websites, instant messengers (Facebook, Instagram, Telegram, WhatsApp), mobile applications, and phones.



There are two types of bots according to the **way of communication with a person:** those with push-button navigation and those that understand natural language.

In the first case, the user doesn't need to enter text; however, he is limited in his freedom of choice and the search for an answer to a question can be quite long. Bots that understand natural language can recognize the intent of a person from the entered text (both written and spoken) and immediately provide the information necessary for the user or perform the required action.

By their **algorithms of behavior**, bots can be of two types: those built on rules, following a rigid algorithm for responding to user actions; and self-learning bots using neural networks and other artificial intelligence technologies. If the user's request is not confidently recognized, the chatbot may ask the user to reformulate it or offer to connect the user with a live agent.

Typical tasks that chatbots solve: expansion of the client base; answers to frequently asked questions; notification of news and upcoming events; recruitment; sales and services; scheduling of appointments; product support; presentation of the company; collecting feedback and many more.

Common Architecture

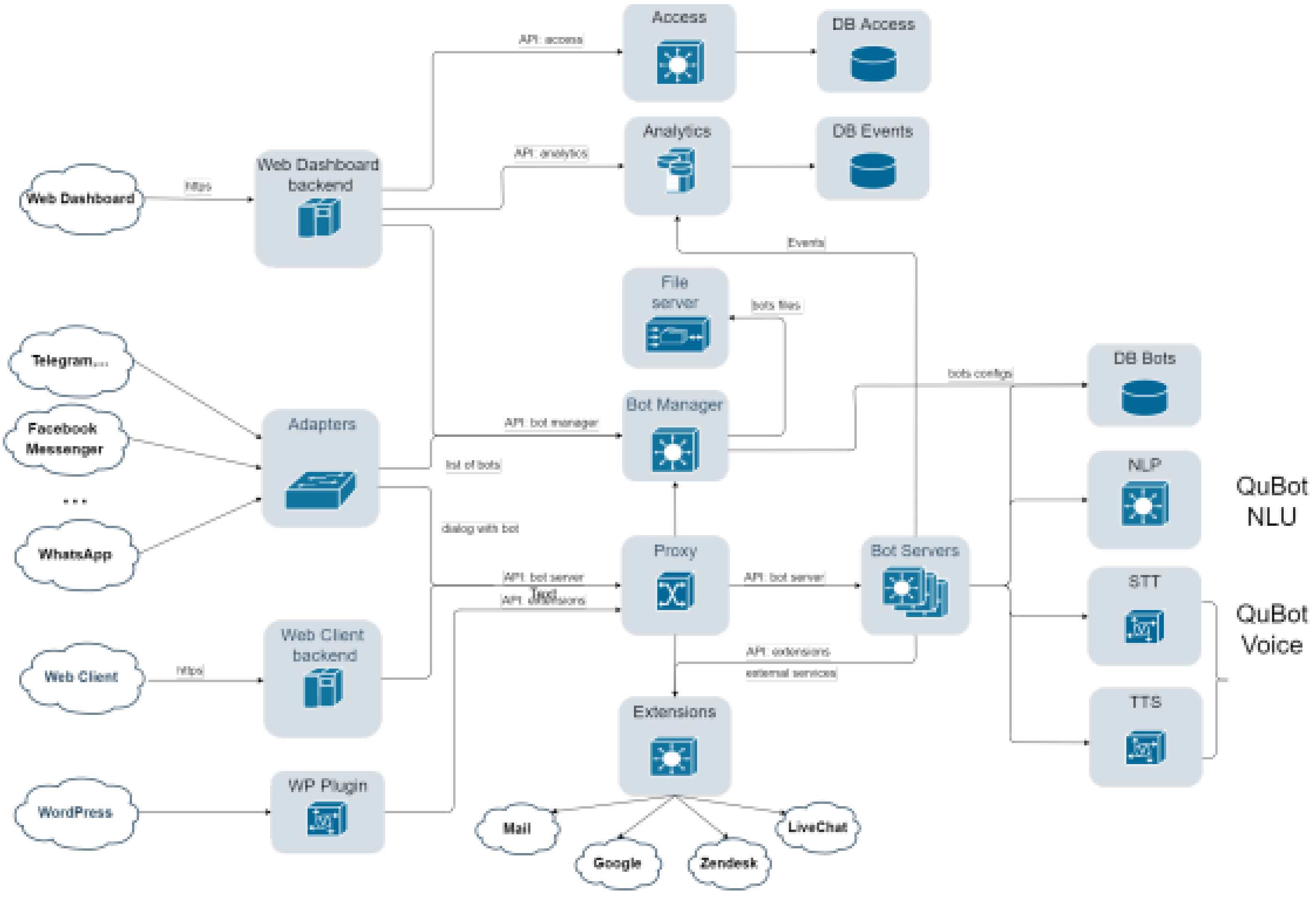
The QuData company has developed the QuBot platform for creating bots that have a customizable interaction interface, reaction algorithms, and work in different communication channels. The platform has a modular structure that can be modified to solve a specific problem. The platform consists of a logical engine and a visual frontend called the Web Console.

Logical Engine

The central module of the logical part is the Bot Server, which controls the logic of the various bots.

It can be scaled into separate blocks when the load on the system increases. Thus, one Bot Server can serve several less loaded bots, while the other will work with just one bot and a lot of traffic. The bots managed by Bot Server can be self-contained or rely on external modules to understand natural language and speech.

Bots interact with the visitors using a set of adapters, each one focused on a specific channel. As a result, the same bot can simultaneously work, for example, on a website, and on Facebook Messenger. Bot Server can interact with various extensions that provide integration with external data stores, mail clients, and operator terminals, if necessary.



Detailed Description of Individual Modules

Bot Servers

Servers for the interpretation of bot configurations provide an API for communication with messenger adapters**.**

There are two types of servers:

- **Buttons Bot Server:** for rule-based chatbots with GUI elements like buttons, images, and checkboxes. The server loads the bot configuration and the user-bot saved state and decides what the next actions and state must be. This server communicates with **Adapters** for transferring bot messages to users via messengers and vice versa;
- **Voice Bot Server**: for NLP-based chatbots executing voice-to-voice or text-to-text communication similar to a human-to-human dialogue. This server does the following tasks:
    - ○ VAD - voice activity detector to recognize the start of speech;
    - ○ STT - speech to text system (Kaldi, Vosk);
    - ○ NLP - natural language processing module for detecting appropriate text response (RASA)
    - ○ TTS - text to speech for synthesis of text into voice (Silero)

Technology stack: Python, aiohttp, WebSocket, RASA

Bot Manager

Allows to store bot configurations into the bots DB and save chatbot assets to the File Server. Provides the API to access the bot configuration from the **Web Console** and to make updates to bot configs.

Technology stack: Python, aiohttp, MongoDB, WebSocket/REST API

Access

Provides an API for **Web Console** registration, controls users and bot access rights. All access data is saved into MongoDB (**DB Access**). Provides JWT-tokens for secure communication between internal servers.

Technology stack: Python, aiohttp, MongoDB, WebSocket/REST API

Adapters

Modules for interacting with messenger APIs and bot servers APIs.

Presently implemented adapters (in production):

- Telegram
- Facebook Messenger
- WhatsApp
- SIP/Asterisk (voice access)

Technology stack: Python, aiohttp, WebSocket

Analytics

Collects chatbot user events into DB Analytics. Configuration consists of states, where each state is a window that is shown to users via messengers. All bot servers send events to the analytics server when the user switches states by pressing buttons or entering new text. After that, the bot creator can view and analyze analytics charts from the **Web Console**.

Technology stack: Python, aiohttp, MongoDB, WebSocket

Extensions

Modules for communication between **Bot servers** and external services:

- Google Sheets;
- Google Maps;
- Google Disk;
- Google Calendar;
- LiveChat;
- Zendesk;
- Mail

Technology stack: Python, aiohttp, WebSocket/REST API

File server

Hosting:

- **Bot assets**: https://services.qudata.com/bot/fs/
- **Web Console**: https://console.qudata.com
- **Web Client**: https://client.qudata.com

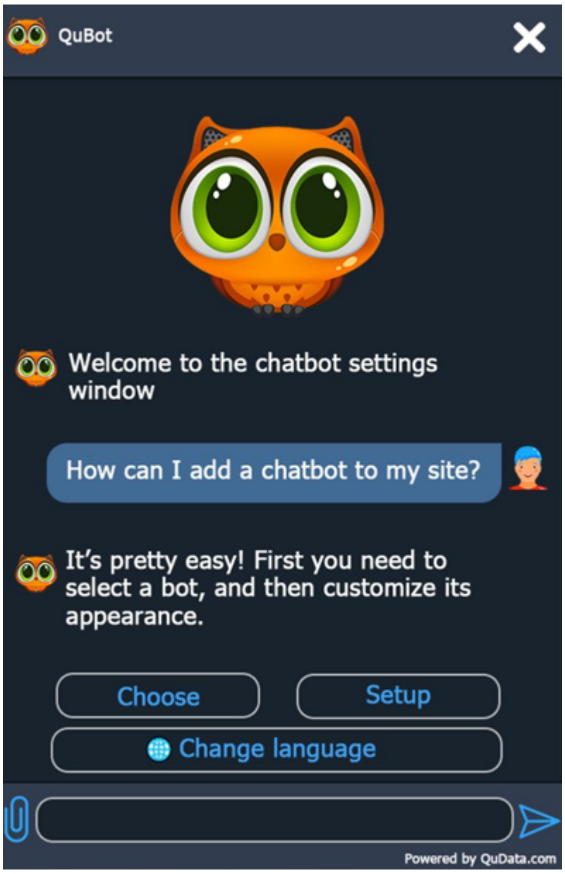
Service proxies:

- **Bot Server**: https://services.qudata.com/bot/server/
- **Access**: https://services.qudata.com/bot/access/
- **Bot manager**: https://services.qudata.com/bot/manager/
- **Extensions**: https://services.qudata.com/bot/extensions/
- **Analytics**: https://services.qudata.com/bot/analytics/

Technology stack: nginx

System requirements

CPU:	**Intel(R) Core(TM) i7-9700 CPU@3.00GHz**
RAM:	**16GB**
Storage:	**128GB**



Welcome to the chatbot settings window

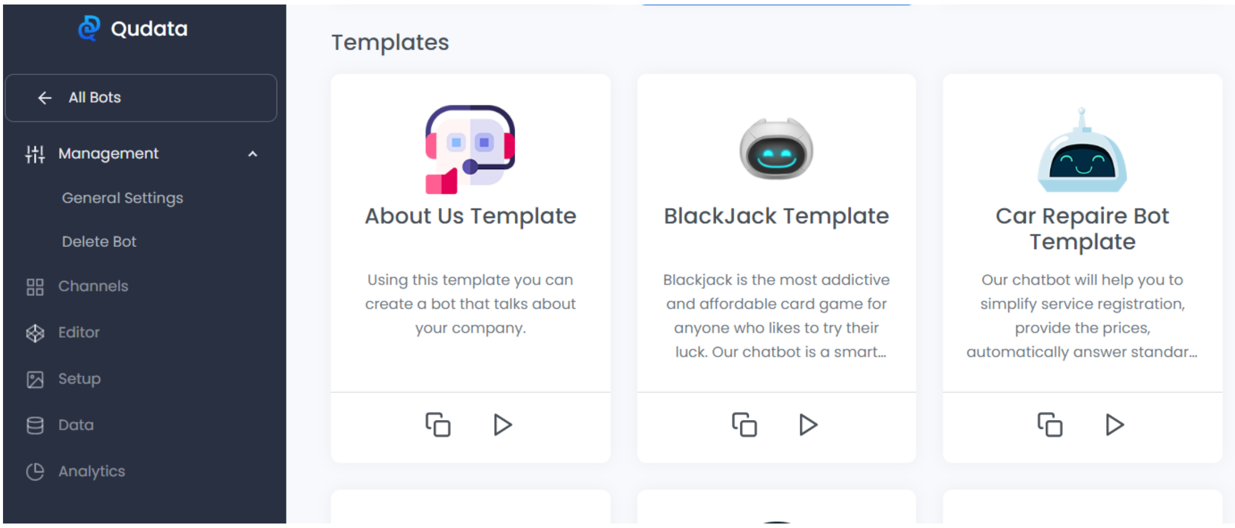
How can I add a chatbot to my site?

It's pretty easy! First you need to select a bot, and then customize its appearance.

Web Console

The Web Console (https://console.qudata.com/) is the visual component of the QuBot platform. It is a web dashboard for building and testing chatbots, viewing analytics, and integrating them into social messengers (Facebook (Meta), Telegram, WhatsApp, SIP-voice), and WordPress. Next comes the description of the main Web Console module.

1. Management: Contains the initial registration module for bot developers, installation of basic bot settings, implementation of bot's import/export, general account information, payment options, and other administrative features.



Templates

About Us Template

Using this template you can create a bot that talks about your company.

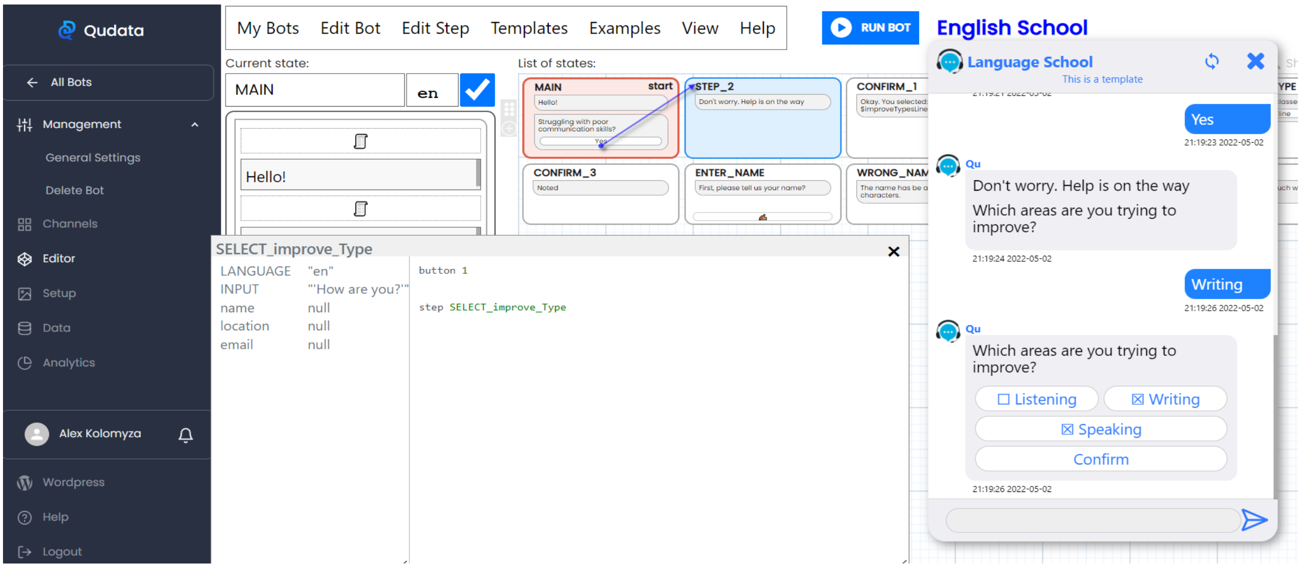
BlackJack Template

Blackjack is the most addictive and affordable card game for anyone who likes to try their luck. Our chatbot is a smart...

Car Repaire Bot Template

Our chatbot will help you to simplify service registration, provide the prices, automatically answer standar...

2. Testing: Allows for convenient testing of the bot through the web widget, as well as debugging logical steps and bot internal slots. All changes to the bot algorithm are reflected immediately in the test instance within the console.



Current state: MAIN en

Hello!

List of states:

MAIN start: Hello! / Struggling with poor communication skills? / Yes

STEP_2: Don't worry. Help is on the way

CONFIRM_1: Okay. You selected: $improveTypesLine

CONFIRM_3: Noted

ENTER_NAME: First, please tell us your name?

WRONG_NAM: The name has be a characters.

SELECT_improve_Type

```
LANGUAGE   "en"                button 1
INPUT      "'How are you?'"
name       null                step SELECT_improve_Type
location   null
email      null
```

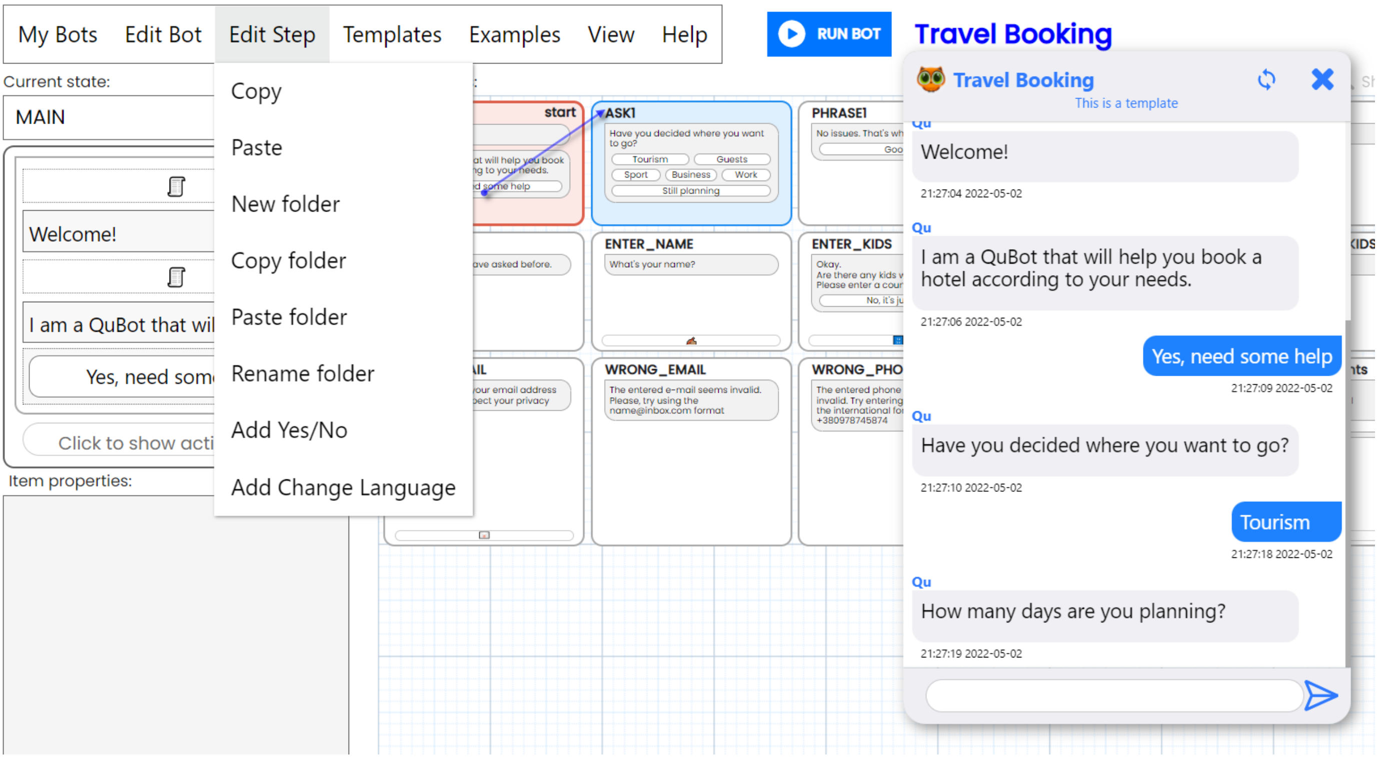
Yes
21:19:23 2022-05-02

Qu: Don't worry. Help is on the way
Which areas are you trying to improve?
21:19:24 2022-05-02

Writing
21:19:26 2022-05-02

Qu: Which areas are you trying to improve?
☐ Listening ☒ Writing
☒ Speaking
Confirm
21:19:26 2022-05-02

3. Editor: This is a visual bot building system. This builder allows you to quickly design behavior logic for rule-based bots. In addition to simple transitions between messages (steps), the builder uses a fairly powerful scripting language that allows you to process data, save it, etc. The editor has its own NLU module that allows, in simple cases, to not refer to a common NLU module using the RASA system.



Welcome!
21:27:04 2022-05-02

I am a QuBot that will help you book a hotel according to your needs.
21:27:06 2022-05-02

Yes, need some help
21:27:09 2022-05-02

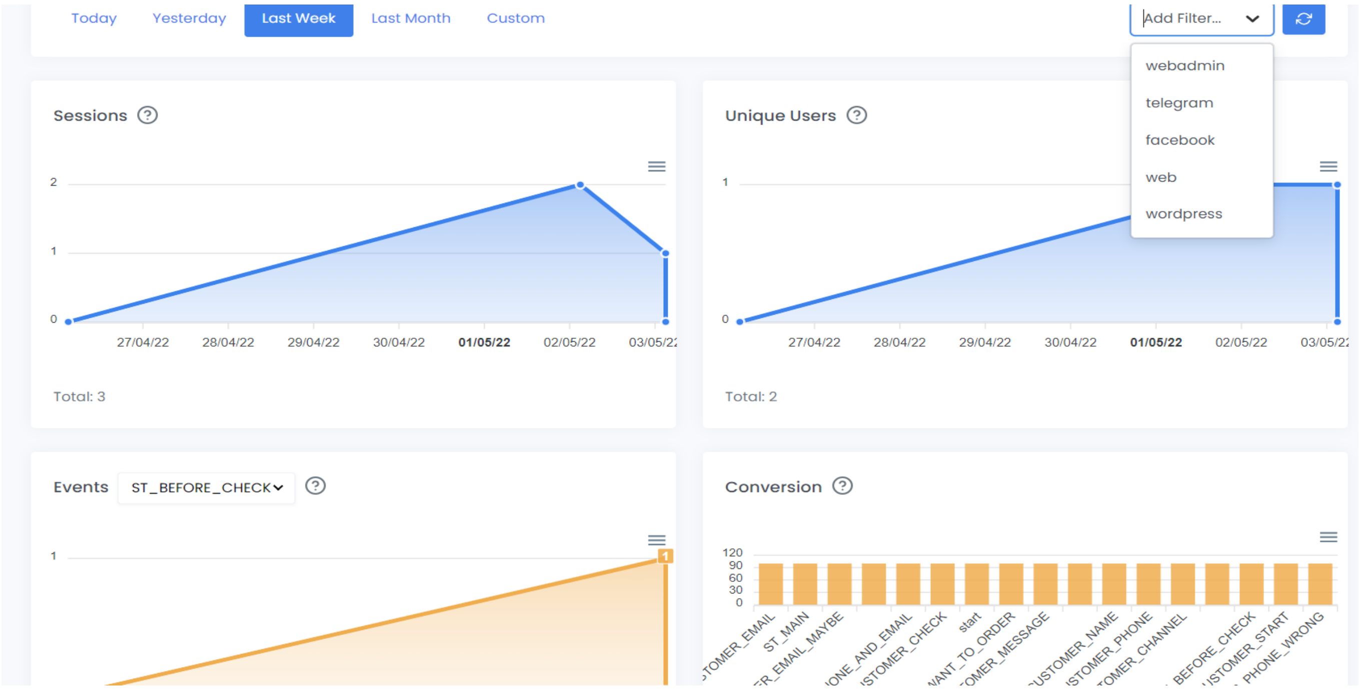
Have you decided where you want to go?
21:27:10 2022-05-02

Tourism
21:27:18 2022-05-02

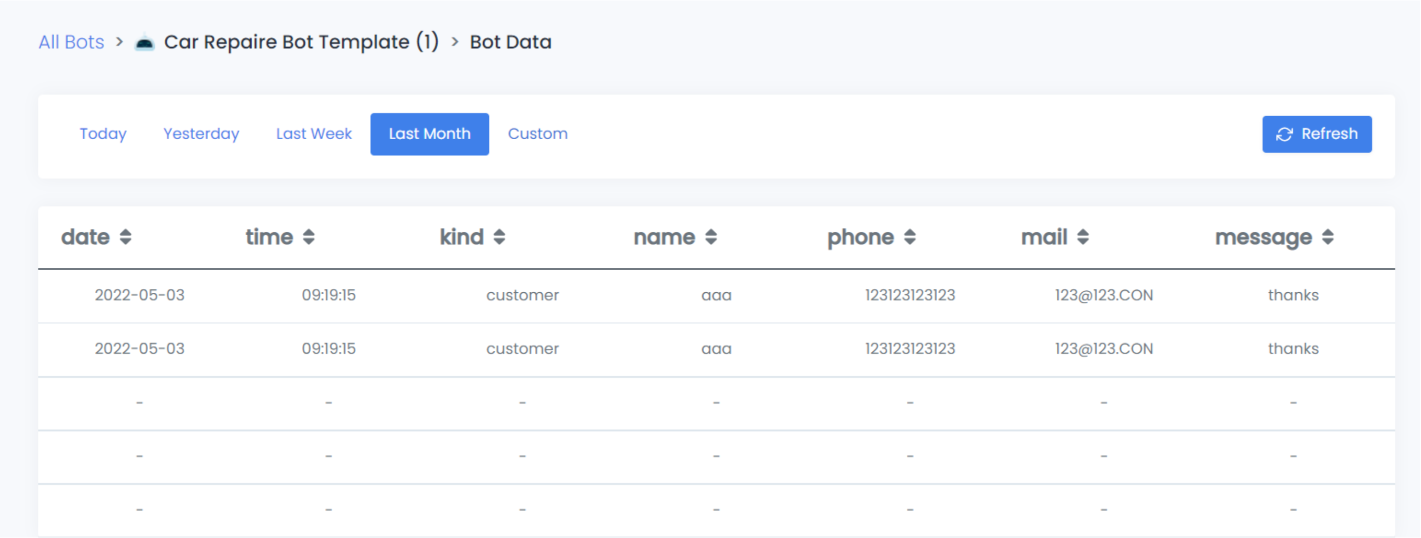
How many days are you planning?
21:27:19 2022-05-02

4. Channels: bot integration with the social messengers: Facebook, Telegram, WhatsApp, Viber, Web Widget, SIP-voice, and with the WordPress plugin.

5. Analytics: bot user events, sessions, and bot step conversion.



6. Bot Data: The store-and-view module for the data that is collected by the chatbots from end users during the communication sessions. The exact information saved here (for example, customer orders, names, addresses, and so on) is defined by the bot operator during the chatbot design stage.

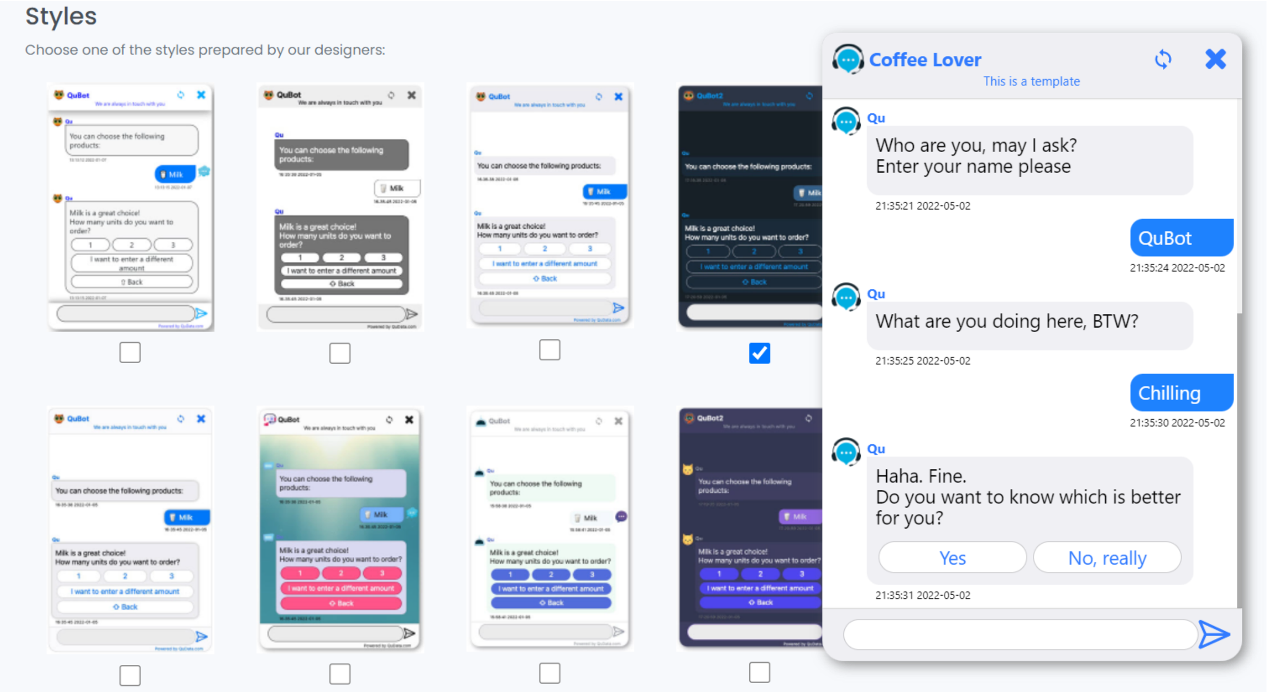


date	time	kind	name	phone	mail	message
2022-05-03	09:19:15	customer	aaa	123123123123	123@123.CON	thanks
2022-05-03	09:19:15	customer	aaa	123123123123	123@123.CON	thanks
-	-	-	-	-	-	-
-	-	-	-	-	-	-
-	-	-	-	-	-	-

Technology stack: React, JavaScript, WebSocket/REST API

Web Chatbot Widget

Web application for communication of users with bots or agents. The widget can be customized with CSS styles, using the visual editor, to match the target site design. Multilanguage support is available, and site visitors can change the communication language on the fly. A number of templates are provided, making the initial adoption easy for site operators.



Qu
Who are you, may I ask?
Enter your name please
21:35:21 2022-05-02

QuBot
21:35:24 2022-05-02

Qu
What are you doing here, BTW?
21:35:25 2022-05-02

Chilling
21:35:30 2022-05-02

Qu
Haha. Fine.
Do you want to know which is better for you?
21:35:31 2022-05-02

Technology stack: React, JavaScript, WebSocket/REST API

QuBot - Smart WordPress Plugin

Our plugin is available for websites developed on the popular WordPress platform that allows you to interact with the QuBot platform. The plugin client can independently develop a chatbot and place it on the website using the editor and ready-made templates. This bot can be local and does not require access to QuData servers in simple cases.

Easy website integration through a WordPress plugin provides a very accessible way of using the QuBot technology by a wide range of businesses and utilizes the popular distribution channel and extensive installation base.

About us

The QuData team develops solutions in analytics, artificial intelligence, data science, and big data, addressing innovations and challenges which shape future technologies by promoting and discussing ideas.

We help add value to your business through custom software development, product design, QA, and consultancy services.

Contact us

53 Voznesenska St, Dnipro, Ukraine, 49000

info@qudata.com

+1-332-3318652

+380674420588