# Technical Specification

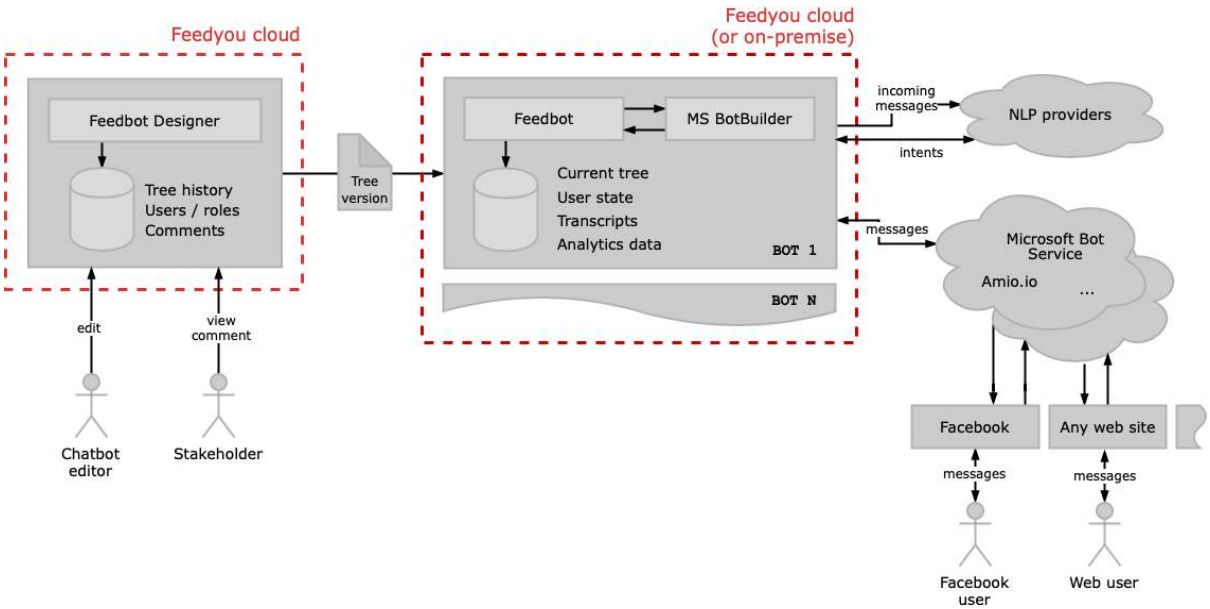
# Feedbot Designer Cloud

## General Information

Feedbot Designer Cloud (FDC) is a cloud-based solution for creating and managing communication robots (chatbots) based on a combination of communication trees and NLP models.

The solution consists of four main interconnected parts:

- **Feedbot Designer** cloud application for collaborative creation and management of bot communication tree and all its settings.
- **Instance** of chatbot itself (processes incoming and outgoing messages based on the current version of the communication tree, maintains the status of individual users including their position in the tree and conversation history).
- **Microsoft Bot Service** or **Amio.io** to connect specific communication channels to the chatbot instance.
- **NLP model providers** to which incoming messages are sent and the information of a detected intent is returned.



# Method of Use

FDC users have the option of building their own communication structure (communication tree) in Feedbot Designer, or they can select one of the pre-made communication templates for specific use (FAQ's, Lead Generation & Qualification, Sourcing, Onboarding, Pre-Sales) directly in Feedbot Designer and edit it. At the same time, NLP models for user message recognition (FDC primarily supports NLP models *Geneea, Alquist, Microsoft LUIS, IBM Watson etc.*) can be linked to the chatbot instance itself or these models can be created and connected on demand (however, NLP models are still run by the NLP providers, not directly in FDC).

The purpose of the FDC is to enable the user to easily build the communication structure in Feedbot Designer, which is then sent in the JSON format to the selected instance of the chatbot (a process called deploy). *JSON-schema* automatic validation ensures compatibility between versions. The new version is deployed without any downtime and immediately starts operating in the selected communication channel (see Supported Channels for more).

At the same time, FDC helps keep track of individual bot conversations or individual users and their current position in the tree. FDC can export all these data from each single bot into *Office365 Excel, Google Sheets, various IS such us Dynamics365, SAP, Salesforce, … and visualize them in Google Analytics, PowerBI etc*. This allows the bot to be continuously changed and improved based on the data collected.

# Used Technologies

Both the backend of the designer and the bot itself are created using the Node.js JavaScript environment and verified technologies such as *Microsoft Bot Framework* or *Express*. Frontend uses a verified combination of modern reactive libraries such as *React* or *Redux* which, connected with *WebSocket*, enable maximum interaction of multiple cooperating users.

## Infrastructure

The *Microsoft Azure* cloud is used to run the entire solution, where each chatbot and each designer runs in a separate resource group, reducing certain security risks and making administration easier. All these groups are listed under Feedyou's subscription which takes care of the cost management.

Instances of chatbots use *Azure Functions* serverless hosting, which reduces infrastructure management requirements and dramatically increases scalability. As a result, the cost of hosting is proportional to the actual use of the chatbot.

The database is used exclusively by *Azure Storage Tables*, a non-relational service whose simplicity ensures almost unlimited levels of scaling and high security.

## Login, authorization and user management

Every new user (company) gets its own clean FDC installation, where it logs in and authorizes through a dedicated *Auth0* platform using supported adapters such as *Facebook Login*, email + password, *GSuite, Microsoft AD*, etc.
Thanks to this possibility to connect to the existing authentication solution already used by a given customer, there is no further weakness in terms of security.

## WebChat

To connect the chat to any public or internal site, we provide an extended *Microsoft WebChat* library extension. This can be implemented in a number of ways, from simple code insertion to a typical pop-up chat window, to tight integration with the web, including the possibility of bidirectional background information transfer that can make chatbots more aware of user needs.

There is the possibility of any customization of the chat component. From colour adjustments to behavioural changes to deep involvement in the web structure itself.

GitHub Documentation:
https://github.com/wearefeedyou/feedbot-webchat
https://github.com/wearefeedyou/feedbot-webchat/blob/master/samples/feedyou/expandable.html

## Supported Channels

The FDC solution is unique in that each single created bot can be run simultaneously on multiple communication channels, and the results from all these channels meet in one place. At the same time, any changes made to the communication structure or the NLP model will automatically be reflected in all channels. Currently, you can run bots simultaneously on these channels:

- Feedyou WebChat
- Facebook Messenger
- Microsoft Teams
- Slack
- Viber
- Mluvii
- Skype
- WeChat
- Telegram
- Kik

More channels are being continuously connected by Feedyou.

# Own Infrastructure Operation

We are able to assist the customer with deploying the entire solution to his own infrastructure. Currently we only support *Azure Cloud* customers and with certain specifics.

Customers themselves will host within their *Microsoft Azure* account:

- a chatbot instance in the form of a *Node.js* application,
- database of all user data (message history, current tree position, etc.).

In our or our partners' Microsoft Azure Cloud will still need to run:

- *Feedbot Designer*, a tree and statistics management tool,
- *Bot Service*, processing all inbound and outbound messages
- services for recognizing user's message contents (NLP)

In the future, we also plan to run a chatbot instance on our own servers.