International Journal of Latest Engineering and Management Research (IJLEMR)
ISSN: 2455-4847
www.ijlemr.com || Volume 08– Issue 10 || October 2023|| PP. 08-09

Model of building low-code business processes using GPT

Maxim Gavrilov
Senior Developer

Today, large and medium-sized companies increasingly rely on cloud solutions from software vendors and move part of their business processes to cloud services. Their high popularity is largely due to the fact that they allow designing information algorithms using no/low-code solutions - constructors consisting of hundreds of universal information processing blocks. Assigning specific properties to such blocks makes it possible to apply them in the context of a particular task. In terms of functionality, this set of blocks represents a finite automaton. Thus, designers of no/low-code solutions actually serve as tools for creating finite automata, which provides flexibility and efficiency in business process management.

Some of the key advantages of such solutions include the following:

They require significantly less specific knowledge, which reduces the skill requirements of candidates for IT positions.

Business processes take less time to create because no generic code needs to be written. Instead, pre-designed blocks are used to construct business processes.

At the same time, such designers have some disadvantages:

Limitations in the use of multiple IT solutions and technologies. Some blocks are difficult to make universal, and such constructors do not support functionality inheritance mechanisms.

Difficulties in applying classic design and programming patterns for these solutions.

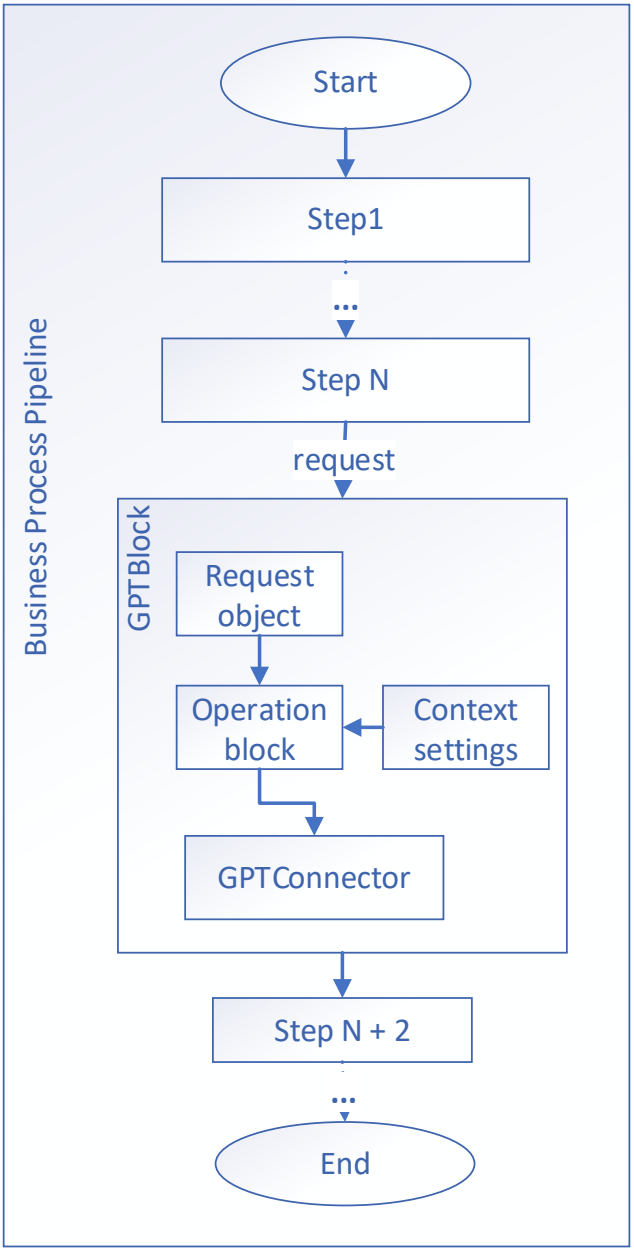
Weak flexibility of implementing projects created using no/low-code platforms.

Lower performance compared to classical patterns based on interpreted or compiled modern programming languages.

Thus, the limited capabilities of no/low-code constructors do not make them a panacea. Nevertheless, they represent a promising technology that will not replace classical developments based on high-level languages, but may supplant them (as high-level languages supplanted hardware and low-level languages in their time).

One of the directions of development of low/no-code solutions is the use of neural networks as workflow building blocks. Already now large software vendors are trying to create interaction blocks with neural networks in general and GPT-systems in particular. An example is Microsoft with its Power Automate platform and a connector to the ChatGPT chatbot. However, at the moment such connectors are only an interface to interact with GPT, while the way to apply GPT as a business process unit is the task of the end organization. The ways and models of embedding GPT into such platforms are still a matter of debate. While some experts believe that the use of GPT models in such systems does not bode well, others assure that there are many ways to use GPT systems. One such model is discussed in this article.

The model of embedding GPT into low/code platforms is shown in the diagram below.



Here we see an algorithm for embedding GPT into the Pipeline of an abstract business process. The main idea is that at the next step a certain query to GPT is formulated, it is a predefined set of data. However, in addition to the query, this model also includes context setting. When a query arrives in the back-end operational unit, the query is transformed into a query with a context that is predefined. The context in this case is nothing but a description of the required actions, information processing methods of dispensing methods and other relevant procedures. Thus, a request to GPT sent together with a context can return a result depending on the data processed in a predefined way. Importantly, unlike a conventional block, in which the form of the data must be defined and explicitly specified, in this case the form of the data need not be specifically typed. It is enough just to describe the task and the required result, and all the necessary data conversion GPT is able to carry out independently.

Such a model can be used as a recommendation model - integrating GPT in this way the user can easily build a system of recommendations. As a context to the input of the model can be fed information about the allocation of product properties, and as a query can be the product itself. Thus, defining properties and specifying the category of properties in the context makes available the selection of properties in the query object and the subsequent selection of objects with identical characteristics.