# Automated Text Translation

Shafik Shaekhovich Khuzyatov* and Lenar Ajratovich Galiullin

*Kazan Federal University, Russia*

**Abstract:** The paper analyzed the problem of accessibility of content in other languages, it was found that many content may not be translated into the native language of users who want to access it, but at the same time there are many who want to help other users with this problem. The solution is a special information system that allows you to easily register and create your own translation, in which other users can participate, or join another already created one and help. As a result, the interested user can easily download the translation result and use it at his own discretion. The analysis of business processes for the creation and translation of the text was carried out. Based on this analysis, requirements for a future solution were developed. Business requirements were also identified. Among other things, a system use case model was developed and use case specifications were described. Lists with functional and non-functional requirements have also been developed. The functional model of the system was shown - algorithms: authorization, registration, password recovery, creating a new translation, generating a file with a new translation, generating a list of translations, managing users, viewing a translation, editing a translation text, checking the correctness of a translation, and moderating translations. A class diagram was developed, where you can see the main entities of the system and their relationships. A sequence diagram was also developed. The architecture of the information system was described. The system was implemented using the React.JS library and the Spring framework. The main processes of the system users were also described.

**Keywords:** Text, translation, system, language, automation.

## INTRODUCTION

In today's world, internationalization plays a very important role. Many software products that are developed in one language can get much wider distribution and access to other markets, provided that they are localized for other countries (Valiev, Valieva and Galiullin 2020). Among other things, many books, television series, films are not always localized into other languages due to lack of funds or time for this, and for this reason, users of other countries cannot watch them in their own language (Lenar, *et al*. 2019). If the content were available in other languages, it would increase its sales to users from other countries, and would allow its creators to make more profit, since many are interested in having the content available to them in their own language (Petrov, *et al*. 2019). For example, a game from an indie developer may come out initially only in his native language, because he has no money to translate into other languages, but if the game is interesting to the community, it can be translated quickly and for free. The same goes for any other content: programs, movies, books, and so on. In addition to the direct purpose - the work can be used for installation within the enterprise - for quick translation of any necessary text (for example, for localizing your software) (Umrzoqova 2020).

Thus, the relevance of the work is due to the general need for the ability to quickly localize various products that need it.

*Address correspondence to this author at the Kazan Federal University, Russia; Tel: 8-927-487-26-77; E-mail: k_search12@mail.ru

Currently, many software products face the problem of localizing them for distribution in other markets. In many startups, due to limited funding, it is not always possible to translate the application into other languages, this requires the services of a professional translator, on the other hand, if the problem that the startup solves is in high demand, Internet users who are native speakers of their native language can help for free (Katasonov, Valiev, Khafizov, Shakirov and Valiev 2018). Among other things - most of the content is currently being created for the English-speaking audience of the Internet and is not available to other users who do not speak English, this project also solves this problem. The essence of this system is that you create a translation, which is divided into paragraphs, and any user who is registered in the system can connect to the translation. The translation can be made either available for editing to all users of the information system, or private and visible to a certain narrow group of persons whom you choose yourself, everything is quite flexible. Thus, the main task of this project is to develop an information system for collective text translation, which will allow both localizing any software products into other languages, and providing access to content in another language for users who do not know it (Poncelas, Buts, JHadley and Way 2020).

## METHODS

Since the information system will be used by a large number of people, it is necessary to develop a logic that is understandable and familiar to most Internet

E-ISSN: 1929-4409/20 

users (Mustafina, *et al*. 2019). The collective translation of a text consists of the following main sub-processes:

- A1. Adding a translation;
- A2. Text translation;
- A3. Translation moderation.

Business process diagrams were implemented using BPMN notation according to (Zubkov, Galiullin, 2019).

Process A1 includes the following activities:

- choice of the type of transfer - public or private;
- choice of users who will be able to view and edit the translation (all or a specific narrow group);
- an indication of the name of the translation;
- entering the text to be translated;
- publication of the translation.

Responsible for this process is a person who is interested in translating any content, or his software product into a specific language (Kolesnikov, Mukhametzyanova and Zubkov 2016). This person is responsible for creating the translation and filling in the necessary parameters.

Process A2 includes the following activities:

- editing the paragraph that you want to change or translate;
- an indication of the reason for editing;
- saving the paragraph.

Responsible for this process is the translator who has access to editing this translation and is a registered user of the information system; he is responsible for correcting and editing the translation of paragraphs of the text.

The translator must select the desired paragraph and change it, he can also indicate any comment that characterizes the reason for editing the paragraph, if required (Zubkov, 2016). After these steps, the selected translation paragraph will be saved to the database.

Process A4 includes the following activities:

- approval / rejection of edits with an indication of the reason;
- saving the result.

The person responsible for this process is the person who moderates the translation.

Using the use case model, you can map the main user scenarios for working with the information system (Ilyukhin, Zubkov, 2015).

During the analysis of business requirements, business processes, four main roles were identified:

- “Translator” - a registered user of the site who has editing access to any publicly published translation;
- “Moderator” - a site user who checks translations for violations before publishing them, can block access to the system for various users, can approve or deny publication of a translation;
- “Administrator” - he manages the system, can block / unblock access to the system for any user, give any of the registered users the role of moderator;
- “Guest” - can view the list of translations, can register or log in to the system.

## RESULTS AND DISCUSSION

In the course of analyzing the requirements for the development of an information system for the collective translation of a text, the problem that this system solves was identified and analyzed - namely, the problem of the availability of various content in other languages (Khafizov, Shakirov, Valiev and Valiev 2019). It was found that many companies do not translate their software products into other languages due to lack of funding for this, since translation services cost a lot of money and require considerable time, which can be critical for new companies (startups), plus a lot of content initially is created in other languages, and foreign users do not have access to them because they do not know the required language (Katasonov, *et al*. 2018). The solution to these two problems is the information system being developed, where any user can enter the site and create a translation in which anyone registered on the site can participate.

In the course of analyzing the requirements for the development of an information system for the collective translation of a text, a model of the translation business process was developed; an analysis of third-party solutions in this area was carried out; stakeholders were identified and requirements developed, namely:

- business requirements for the system;
- user requirements in the form of a use-case model;
- functional requirements;
- non-functional requirements.

Based on the requirements for the system, it is possible to distinguish the main functions of the information system, as well as the initial data and the results obtained.

The application is designed as two independent components:

- frontend, which will be presented as a single page application and display the user interface;
- a backend that will provide a programming interface for interacting with the server database.

Since the rendering of the interface will take place in the user's browser, all the computational load will be assigned to his PC, and this will save server resources. Among other things, since our logic is separated from the view, this will allow us to reuse the API that provides the backend for a mobile or desktop application without changes, if required. For convenience, the functional diagram model will be developed with this concept in mind.

## SUMMARY

Taking into account the selected design solutions, the following system architecture was obtained (Figure **1**). The React.JS library was chosen to implement the browser client, and the Spring framework was chosen to implement the server-side.

The information system browser client was written using the React JS library, using the documentation and examples from (Katasonov, *et al*. 2018).

React is a tool for building user interfaces. Its main task is to provide the display of what can be seen on web pages. React makes creating interfaces a lot easier by breaking each page into small chunks. These fragments can be called components.

A React component is, in simple terms, a piece of code that represents a portion of a web page. Each component is a JavaScript function that returns a piece of code that represents a portion of the page. To form the page, you need to call these functions in a specific order, collecting the results of the calls together and showing them to the user.

React uses a programming language called JSX, which is similar to HTML, but runs inside JavaScript, which makes it different from HTML.

Building interactive user interfaces in React is fun and easy. To do this, it is enough to describe how parts of the application interface look in different states. React will update them in a timely manner when the data changes.

Declarative views make your code more predictable and easier to debug.

You can create encapsulated stateful components and then combine them into complex user interfaces.

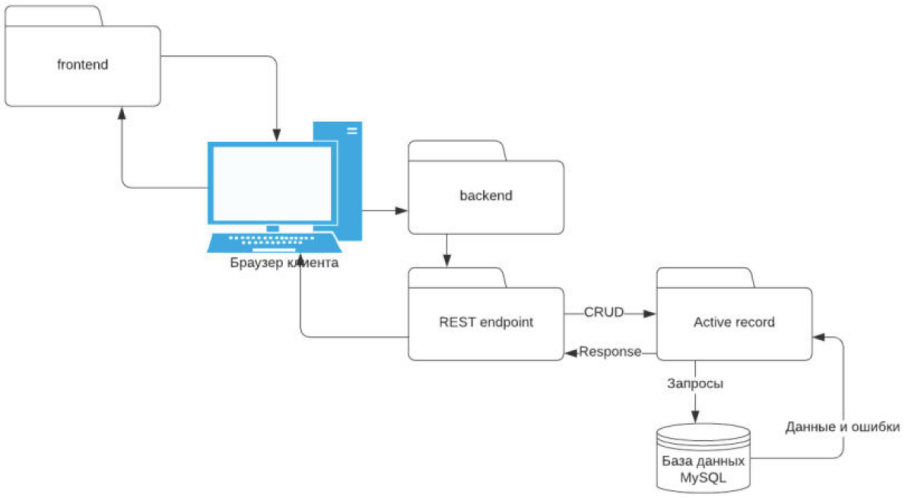


**Figure 1:** Application architecture.

Since the component's logic is written in JavaScript, and not contained in templates, it is easy to pass a wide variety of data throughout the application and keep state outside the DOM.

The information system server was written using the Spring framework, based on examples and documentation from (Katasonov, *et al*. 2018).

Spring is one of the most popular J2EE frameworks. Developers around the world use Spring to build reliable, quality applications. It was developed in June 2003 by Rod Johnson.

Any Java application can be developed with Spring.

It is impossible to understand what Spring is without understanding the term Dependency Injection (DI) - one of the types of Inversion of Control (IoC).

When writing really large and complex projects, developers are faced with the need to make application classes as independent as possible from each other for reusability and unit testing. It is DI that establishes the links between these classes, while maintaining their independence from each other.

Today Spring is divided into a number of separate modules.

The Core Container includes Beans, Core, Context and SpEL (expression language).

Beans are responsible for the Bean Factory, which is a complex implementation of the Factory (GoF) pattern.

The Core module provides key parts of the framework, including IoC and DI properties.

Context is built on the basis of Beans and Core and allows you to access any object that is defined in the settings. The key element of the Context module is the Application Context interface.

The SpEL module provides a powerful expression language for manipulating objects at runtime.

The Data Access / Integration container consists of JDBC, ORM, OXM, JMS, and the Transactions module.

JDBC provides an abstract JDBC layer and eliminates the need for the developer to manually write the monotonous code associated with connecting to the database.

ORM provides integration with popular ORMs like Hibernate, JDO, JPA and so on.

The JMS (Java Messaging Service) module is responsible for creating, transmitting and receiving messages.

Transactions support transaction management for classes that implement certain methods.

To develop the server, we need the modules Web, Security, JPA, Jackson, Kotlin, Mysql, JWT, Core. To simplify the initial configuration of the project, you need to use Spring Boot, in which you need to specify the required modules and then get a project in which you can start developing. The development will be carried out in the Kotlin programming language. MySQL is used as a database.

According to the above - after entering the site, an html page is downloaded to the client's computer, which contains scripts written in JavaScript, and further rendering takes place in the client's browser. During the user's interaction with the site, if it is necessary to receive or process any data, requests to the backend server are also made from the user's computer, the responses to which are processed by the browser client. The details of the interaction can be seen in the next few sections.

## CONCLUSIONS

In the course of the work, the architecture of the system was developed, which is a browser client built on React.JS that accesses the server based on the Spring framework. Screen forms were developed, with the help of which a description of the business process using the system was presented. A webserver was developed that is responsible for processing screen data.

## ACKNOWLEDGEMENTS

The work is performed according to the Russian Government Program of Competitive Growth of Kazan Federal University.

## REFERENCES

Ilyukhin, A.N., Zubkov, E.V. (2015). “Modernized algorithm of neural network initial weighting factors during the diagnosis of diesel engine faults,” International Journal of Applied Engineering Research, Volume 10, Issue 24, Pages 44848-44854.

Katasonov, P.A., Valiev, R.I., Khafizov, A.A., Shakirov, Yu.I., Valiev, R.A., (2018). “Plasma spray coating with iron oxides for manufacturing of the corrosion-resistant electrodes,” Journal

of Physics: Conference Series, Volume 1058, Issue 1, article №012004. https://doi.org/10.1088/1742-6596/1058/1/012004

Khafizov, A.A., Shakirov, Yu.I., Valiev, R.I., Valiev, R.A., (2019). "The influence of plasma power with a liquid electrode on the microhardness of gray cast iron," Journal of Physics: Conference Series, Volume 1328, Issue 1, article №012094. https://doi.org/10.1088/1742-6596/1328/1/012094

Kolesnikov, M.S., Mukhametzyanova, G.F., Zubkov, E.V. (2016). "Optimization of heat treatment modes of steel 4Kh5MFS for metal conduits of hot-chamber pressure casting machines according to results of endurance tests in molten TsAM-4-1," Metal Science and Heat Treatment, Volume 57, Issue 11-12, Pages 690-693. https://doi.org/10.1007/s11041-016-9944-4

Lenar, G., Jamila, M., Egor, P., Rustem, V. (2019). "Application of learning analytics tools in learning management systems," Proceedings - International Conference on Developments in eSystems Engineering, article №9072993, Pages 221-224. https://doi.org/10.1109/DeSE.2019.00049

Mustafina, J., Galiullin, L., Al-Jumeily, D., Petrov, E., Alloghani, M., Kaky, A. (2019). "Application of learning analytics in higher educational institutions," Proceedings - International Conference on Developments in eSystems Engineering, article №8648548, Pages 163-168. https://doi.org/10.1109/DeSE.2018.00039

Petrov, E.V., Mustafina, J., Alloghani, M.,Galiullin, L., Tan, S.Y.,(2019). "Learning analytics and serious games: Analysis of interrelation," Proceedings - International Conference on Developments in eSystems Engineering, article №8648548, Pages 153-156.

Poncelas, A., Buts, J., Hadley, J., & Way, A. (2020). Using Multiple Subwords to Improve English-Esperanto Automated Literary Translation Quality. arXiv preprint arXiv:2011.14190.

Umrzoqova, G. (2020). Simplification of translation competence by means of automated translation means and electronic applications. Архив Научных Публикаций JSPI.

Valiev, R.A., Valieva, D.I., Galiullin, L.A. (2020). "Automation of the ICE Testing Process," Lecture Notes in Electrical Engineering, Volume 641 LNEE, Pages 806-815. https://doi.org/10.1007/978-3-030-39225-3_88

Zubkov, E.V. (2016)."Computer modeling of the automated tests of diesel engines various conditions of their operation," 2nd International Conference on Industrial Engineering, Applications and Manufacturing, ICIEAM 2016 – Proceedings, 2016, article №7911586. https://doi.org/10.1109/ICIEAM.2016.7911586

Zubkov, E.V., Galiullin, L.A. (2019). "Modeling of the diesel engine under real conditions of driving," Journal of Advanced Research in Dynamical and Control Systems, Volume 11, Issue 12 Special Issue, Pages 1365-1370. https://doi.org/10.5373/JARDCS/V11SP12/20193355

Received on 20-10-2020 Accepted on 01-12-2020 Published on 29-12-2020

DOI: https://doi.org/10.6000/1929-4409.2020.09.278