Developing an Intelligent Online Learning System for Foreign Language Vocabulary Training Based on Gamification

Mariya Hrendus[0000-0003-3832-4716]1, Vasyl Andrunyk[0000-0003-0697-7384]2, Maksym Yavir[3], Yuriy Ryshkovets[0000-0001-5831-4000]4, Andrii Khudyi[5], Viktor Hryhorovych[0000-0002-5828-067X]6, Maksym Korobchynskyi[0000-0001-8049-4730]7

[1-3]Lviv Polytechnic National University, Lviv, Ukraine
[5]SoftServe, Lviv, Ukraine
[6]Drohobych Ivan Franko State Pedagogical University, Drohobych, Ukraine
[7]Military-Diplomatic Academy named after Eugene Bereznyak, Kyiv, Ukraine

mariya.h.hirnyak@lpnu.ua[1], vasyl.a.andrunyk@lpnu.ua[2], jestermaxrko@gmail.com[3], Yuriy.V.Ryshkovets@lpnu.ua[4], Khudyy@ukr.net[5], viktor.grigorovich@gmail.com[6], maks_kor@ukr.net[7]

Abstract. The purpose of the work is to develop an intelligent online system for learning foreign language vocabulary based on gamification. In order to achieve this goal, it is necessary to analyze similar systems and take into account their advantages and disadvantages. This will help take the best features for the system. Next we need to design the system, determine the basic functionality that will be developed. Then you need to choose the means of implementation of system. When the tools are selected, it is necessary to divide all the functionality into smaller components and start developing the software. When the development is completed, you must test the performance and performance of developed system. Object of study is process of learning vocabulary of foreign languages and languages in general. The subject of the study is teaching of foreign languages vocabulary on the basis of gamification. Since learning any language begins with learning words, this system aims to accelerate this process.

Keywords. Language, Vocabulary, Data Aggregation, Content, Gamification, Intelligent System, Content Analysis, Data Mining, Context Filtering, Online Learning System

1 Introduction

In today's world, more and more intellectual systems play an important role in helping people accelerate their work or study in a particular field. Intellectual learning systems are widely used in schools and colleges. There are also many intelligent systems for self-education [1-6]. The system under development will be aimed at facilitating the study of foreign languages. In our time, knowledge of foreign languages is a pass in the world of success. By learning a foreign language, one gets the opportunity to use foreign sites and literature. Specialists who speak a foreign language become more popular every year. Therefore, in order to be competitive we need constantly develop and learn something new. This process can be greatly simplified and accelerated with information technology.

As the opportunities open to foreign-speaking people are increasing every day, the system is relevant, especially for Ukraine. Most of Ukrainian population do not speak any foreign languages, so this system may increase this figure. Increasing a number of people who speak foreign language has a positive impact on Ukraine's development. Specialists will be able to attend various scientific conferences and adopt the latest solutions of modern countries for the development of infrastructure of our country. Also, such an intellectual system will be useful for students who want to get an education in the most prestigious institutions of higher education in the world [7-9]. Knowledge of foreign language opens up the possibility of using different foreign-language literary sources, which contain more up-to-date information on scientific and technical fields. Since a variety of games for any system are extremely popular these days, gaming based systems are a good solution [10-16]. With this solution, learning process will be more productive, because it will combine the pleasant with the useful. So, to summarize, it can be noted that this topic is relevant [17-21].

This system contains several new approaches to solving this problem. To begin with, let's consider the flexibility of the system, where user can create a dictionary for any language. The user can add words that he wants to learn, but not predefined in the system. For users who do not want to add words by themselves, it is possible to add a dictionary already created in the system. Another unique solution is to generate vocabularies independently from the user-added words. This means that the system is completely self-contained. An important function of the system is ability to import words from a file. The system also offers the user words from the categories which he most actively learns. Another innovation is gamification; it is an interactive learning of words and the ability to generate games from the words available in the dictionary. Each game has a scoreboard in which all the participating users are sorted in descending order of points scored. This approach fosters a spirit of rivalry and positively influences the interest in using this system. This system can be used both for self-education and for school or educational introduction. In the case of implementation for school teaching, the teacher can add vocabularies to the system with the words provided in the curriculum. You can also monitor student’s knowledge of the embedded rating system. For personal use, the user can use existing dictionaries on the system or create new ones. To improve the learning process, it is recommended that you set an interval for the reminders that the system will send to invite you to play a short version of the game. Because this system is adaptive, it allows you to use it in public transport when commuting or studying.

2 Analysis of Literary and Other Sources

Electronic dictionaries allow for extensive and advanced search of the text of the dictionary, allowing the user to obtain much needed information; but there is no certainty that the information obtained is comprehensive and completely accurate. The problem is that in most cases the data is simply copies from physical sources transmitted without any changes to the electronic format [22-34]. As a result, all labeling and definition discrepancies are repeated electronically with the original source [35-37].

Another advantage of the electronic dictionary is the storage capacity. The hard copy space limit is not copied to the CD-ROM media. The amount of available space in the electronic dictionary is more or less unlimited. For example, the size of a CD is approximately 650 MB (= 650 million characters). An ordinary English dictionary will only require about two hundredths of this space.

Another characteristic difference between electronic dictionaries and printed dictionaries is the structure of the dictionary. Some traditional structures have become less important or even out of date, and the structure that prevails in the electronic dictionary is an access structure, which, however, functions quite differently than it does in the printed dictionary. An access structure is a dictionary structure that directs users to the information they are looking for in the dictionary. There are two types: external access and internal access structure. An external structure is a structure of metrics that, at the macrostructural level, directs users to the lemma they seek.

An internal access structure is a structure of indicators that, at the microstructural level, directs users to the information they are looking for. Communication is the most important means of access. The link can be either an internal or external dictionary [38-45]. The type of internal dictionary that most closely resembles printed dictionaries is a cross-reference from which a mouse click will take the user to the place where the nomination is submitted. In order to store such a large amount of information, it is necessary to use the latest technologies. During the development of the literature, a set of optimal means for solving the set tasks was determined. After that, a study of literary sources was revealed, which revealed the nuances of realization of the task. The study of this material provided the basis for systematic analysis.

3 System Analysis and Justification of the Problem

3.1 System Analysis

Goal tree is a graphical representation of the relationship and subordination of goals that reflects the division of mission and goals into goals, goals, tasks and individual actions. At the highest tree level, the main goal (end result) is represented - an intelligent online system for learning vocabulary of foreign languages based on gamification (Fig. 1). In order to fully reveal the structure, it is necessary to decompose this level. Decomposition is a method of discovering the structure of a system in which it is divided into separate components on one basis. It can be used to associate a primary goal with a view to achieving it. The peculiarity of each level is that it displays goals that are the same in scale and value to create the whole system.

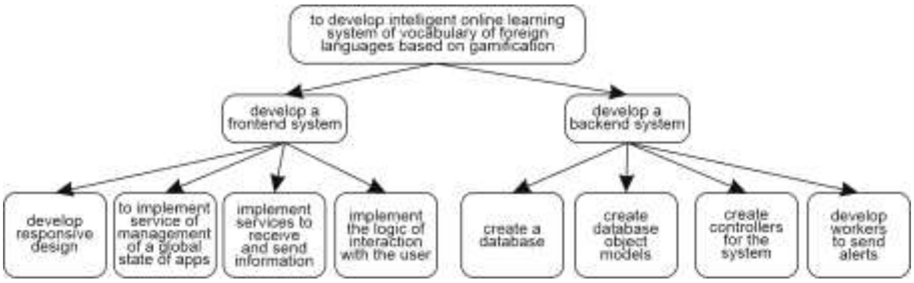

to develop intelligent online learning system of vocabulary of foreign languages based on gamification
develop a frontend system
develop a backend system
develop responsive design
to implement service of management of a global state of apps
implement services to receive and send information
implement the logic of interaction with the user
create a database
create database object models
create controllers for the system
develop workers to send alerts

Fig. 1. Goal tree

The first level shows two goals that are roughly the same in scope. The first of these is to design the frontend. This part is extremely important as this component forms the first impression of the user. Frontend is the calling card of any system, this is what will make the system recognizable. The second goal of this level is to design a backend. The backend will serve as a server on which the main cost calculations will be performed. The two components of this level must be holistic and resistant to any influence. Next, consider the second level, which is a decomposition of goals from the first level. Let's start with the goals related to the front.

The first is to develop an adaptive design. In order to achieve this goal, basic technologies were analyzed and the best of them were assembled. Ensuring system adaptability is a good solution to help you use the system on any device. In terms of system development, this useful feature requires a little more time, but this time is justified. For example, if you compare the time it takes to develop an adaptive design and create a separate application for each device, the time required for an adaptive design is much less. Therefore, the chosen solution can be considered optimal. Another goal from this level is the implementation of a data exchange service. Achieving this goal is crucial because it ensures the integrity of the information on the system. This service will serve to connect the frontend with the backend. The development of this service will facilitate further development, as this service will contain a set of methods that can be used from any part of the system to obtain backend data and send.

Also at this level is a goal that is responsible for implementing the logic of user interaction. Achieving this goal will ensure the correct reading and conversion of the information entered by the user, and further transfer it to the service described above for sending to the backend. It is important that the interface is intuitive. Achieving this goal also involves displaying certain validation errors in the input fields. Dealing with bugs takes extra time, but it's an integral part of a good system.

The next goal that relates to the frontend part is to implement a global application state management service. Achieving this goal allows you to access information that is often used by any component of the program. This service greatly simplifies the control of the data flow, since it avoids the transmission of information across multiple components of the system to the destination. Implementation of this service does not require much time and saves time for further development.

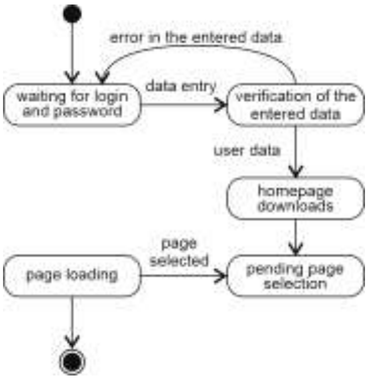
Also at this level are placed goals that are related to the backend part.

The first goal of this layer is to create a database. A database is a body of information that is organized so that it can be easily accessed, managed, and updated. A database is a must-have part of any system. Since the designed system will handle a lot of data, it is necessary to choose the right type of database. For this purpose, an object-oriented database is ideally suited to avoid complex queries to obtain related information. The database will store information about registered users, their dictionaries and many information needed to operate the system. The following goal is also related to the database. This is the creation of database object models. To achieve this, you need to create the essential entities that will be used in the system. The model should be composed of atomic units that fully describe the essence. Each atomic unit must have its own type, typing database objects will preserve its integrity, because all data will have the same structure. Typing will also make it easier to write methods for working with objects further, as methods will know exactly what data structure to work with.

Moving further down the tree, the next goal is to create controllers for the system. A controller, which provides the basic logic of the program, the interconnection of all elements of the system, and routing. The controller must have a set of methods necessary for the system to function. The controller is an important component, because it will perform all logical operations. It is important that the controller is responsible for certain types of operations. For example, methods that perform functions over similar objects are described in a single controller. Proper description of the controller will provide better speed of software development, because all methods that are similar in implementation will be collected in one specific place.

This system has to send notifications to the user, it is necessary to develop workers. Worker is a mechanism that allows you to run creaky operations in the background stream, separately from the main current application stream. The advantage of this method is that time-consuming processes can be performed in a separate thread, allowing the main (usually UI) flow to operate without blocking / slowing down.

State diagram. The main purpose of these diagrams is to describe the possible sequences of states and transitions that collectively characterize the behavior of a model element over its life cycle. The state diagram presents the dynamic behavior of an entity based on the specification of its response to the perception of certain specific events (Fig. 2). After clicking the link to the website, the authorization page is loaded. To get started with the system, the user must be logged in to their account. To do this, the user must enter his login and password. After entering these data, they are checked for correctness (Figure 3). If the user enters false data, an error message will be issued and a reauthorization will be requested. In the successful case, the user is redirected to the main page of the site. After loading the home page, the system enters standby mode for user selection. When a user makes a choice, they are redirected to the appropriate page and the page is loaded. Therefore, it can be noted that the system has 5 basic states, which consist of certain nested states. Consider the nested processes for verifying the correctness of the data entered. First, it is checked that all the required form fields are filled in, if not the error is sent as a result of the check. If all fields are filled in, this data is sent to the backend, where the user is checked for such data, if the data match sends a successful verification result.



error in the entered data
waiting for login and password
data entry
verification of the entered data
user data
homepage downloads
page loading
page selected
pending page selection

Fig. 2. UML state diagram

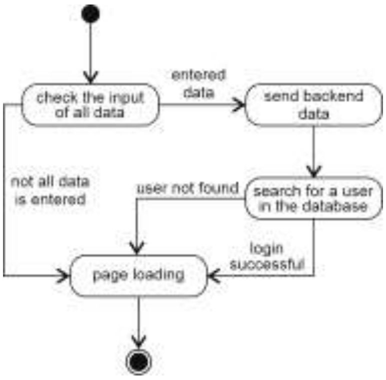


Fig. 3. UML state diagram for checking the input data

Sequence diagram. When constructing data activity for an application being developed, you need to describe all the essential entities and the relationships between them (Fig. 4). The developed system has the following entities: user, frontend, backend. Initially, the user interacts with the frond that displays the authorization page. Then the user enters the data into the form, then the data is validated on the frontend, and then transmitted to the backend to search for the user in the database. After finding the user, a success message is sent to the frontend and the main page of the application is displayed.

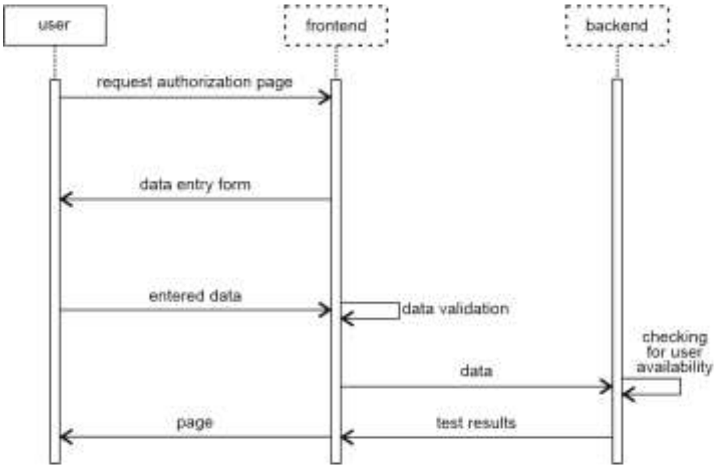


Fig. 4. Sequence diagram

3.2 Formulation and Justification of the Problem

The purpose of development is to create an intelligent online learning system for vocabulary of foreign languages based on gamification. Looking closer, you need to create a system that provides tools to help you learn new words in a foreign language. The development must be a fully operational system that will be filled with data coming from users. Because the system is standalone, it does not need support after implementation by the developer.

Description of the purpose of the system. The system is designed to simplify and accelerate the process of replenishing vocabulary and learning foreign languages.

The developed system has several variants of use as it includes a rather large number of functions. Each user can use the system according to their needs and choose their own way of learning. For example, a user can create their own dictionaries or use existing dictionaries created by other users on the system. The user can also generate interactive games from their vocabularies to facilitate word learning. The system can also be used as a regular electronic dictionary, with the ability to search for words in certain categories (Fig. 5)

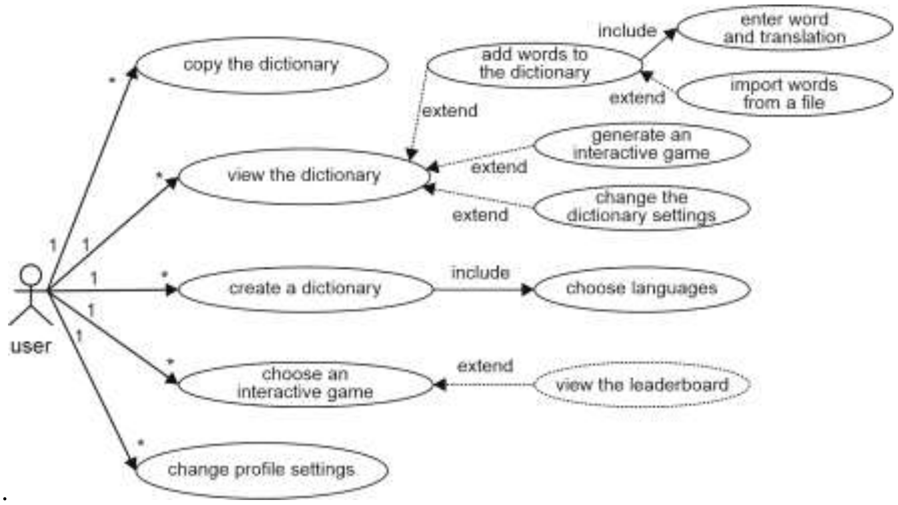
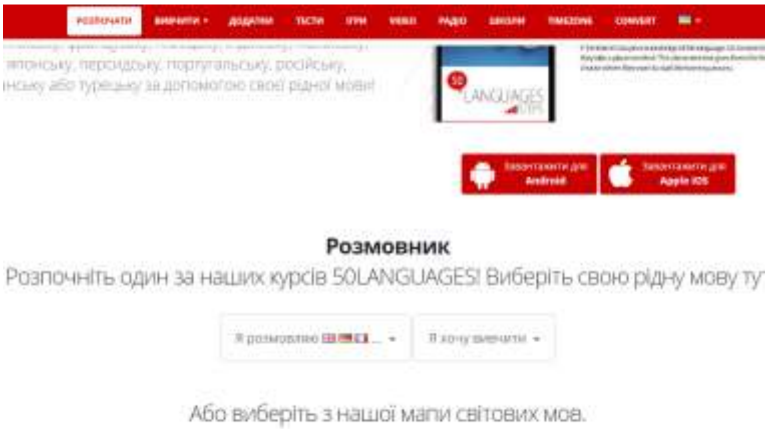


Fig. 5. Use case diagram

Place of application of the system. This system can be used for self-education and for school or educational introduction. In case of implementation for school teaching, the teacher can add vocabularies to the system with the words provided in the curriculum. You can also monitor students' knowledge of the embedded rating system. For personal use, the user can use existing dictionaries on the system or create new ones. To improve the learning process, it is recommended that you set an interval for the reminders that the system will send to invite you to play a short version of the game. Because this system is adaptive, it allows you to use it in public transport when commuting or studying.

Rationale for the development and implementation of the system. As the opportunities open to foreign-speaking people are increasing every day, the system is relevant, especially for Ukraine. Most of the Ukrainian population does not speak any foreign languages, so this system may increase this figure. Increasing the number of people who speak a foreign language has a positive impact on Ukraine's development. Specialists will be able to attend various scientific conferences and adopt the latest solutions of modern countries for the development of infrastructure of our country. Also, such an intellectual system will be useful for students who want to get an education in the most prestigious institutions of higher education in the world. Knowledge of a foreign language opens up the possibility of using different foreign-language literary sources, which contain more up-to-date information on scientific and technical fields. Since a variety of games for any system are extremely popular these days, gaming based systems are a good solution. There are already several analogues on the market, which have their advantages and disadvantages, let's closer look at some of them.

50Languages system. This system allows you to select languages to learn from the list of existing ones. It has a beautiful design. The system also has a built-in game, but unfortunately its results are not saved and the user is not able to track their progress. Also, this system is not flexible as it offers ready dictionaries without being able to change them (Fig. 6)



японську, перську, португальську, російську,
інську або турецьку за допомогою своєї рідної мови!

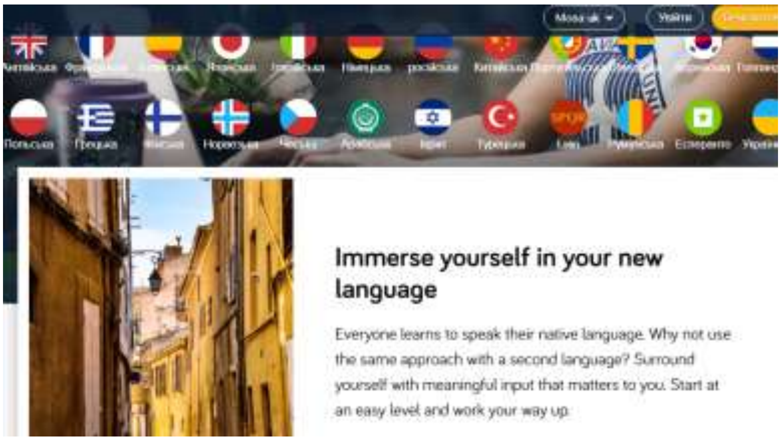
Розмовник

Розпочніть один за наших курсів 50LANGUAGES! Виберіть свою рідну мову ту

Або виберіть з нашої мапи світових мов.

Fig. 6. 50Languages system

LingQ system. This system also allows you to select languages to learn from your existing list. It has a somewhat intuitive interface. This system does not integrate with the game mode. The downside is that this system allows you to use the incomplete version. You need to purchase the full version to get full access to the system.



Immerse yourself in your new language

Everyone learns to speak their native language. Why not use the same approach with a second language? Surround yourself with meaningful input that matters to you. Start at an easy level and work your way up.

Fig. 7. LingQ system

Duolingo system. Among such systems, this system is the closest to the developed one. It has all the necessary ingredients for a good system. The only downside is that the user has to follow a certain program of choice. Also, the built-in dictionary does not allow you to see all the words you need, you need to enter each word one by one in the search.

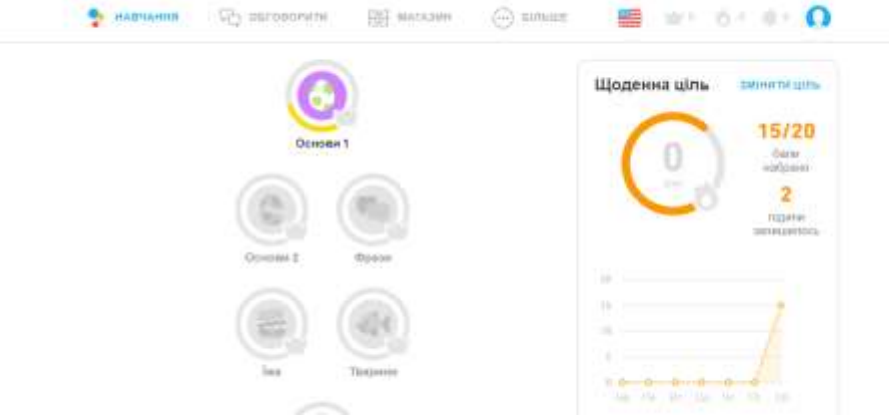


Fig. 8. Duolingo system

Development of conceptual model. The input to get started is the text data that is required to register a user on the system. Without this information, you can not start the program. In the application phase, the input data may vary depending on the function selected. For example, there are two different ways to enter input to add words to the dictionary. The first one is keyboard management, the second is file import. The output is a structured dictionary that can be viewed online on any device or exported to a file.

Description of project functions and structure. To describe the features and structure of a project, designing a class diagram is an integral step. This diagram shows the logical structure of the developed system at the object level, shows the interaction between them and displays the functionality of each.

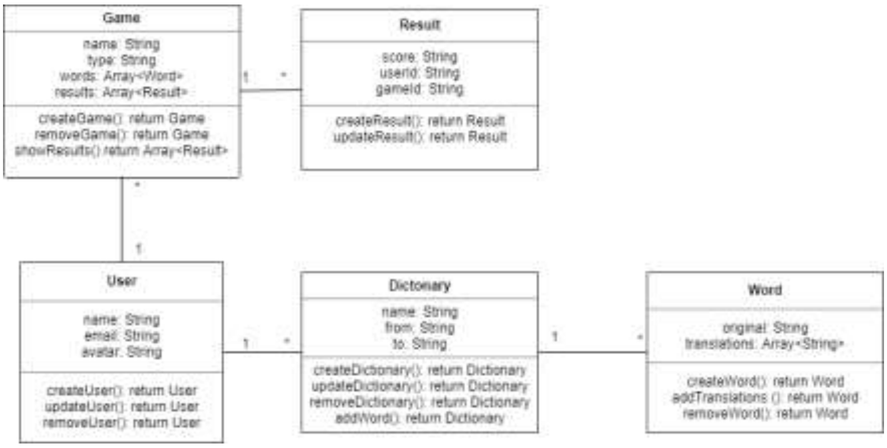

Game
name: String
type: String
words: Array<Word>
results: Array<Result>
createGame(): return Game
removeGame(): return Game
showResults() return Array<Result>
1
*
Result
score: String
userId: String
gameId: String
createResult(): return Result
updateResult(): return Result
*
1
User
name: String
email: String
avatar: String
createUser(): return User
updateUser(): return User
removeUser(): return User
1
*
Dictionary
name: String
from: String
to: String
createDictionary(): return Dictionary
updateDictionary(): return Dictionary
removeDictionary(): return Dictionary
addWord(): return Dictionary
1
*
Word
original: String
translations: Array<String>
createWord(): return Word
addTranslations (): return Word
removeWord(): return Word

Fig. 9. Class diagram

As can be seen in the diagram (Fig. 9), the main essence of the system without which it cannot function is the user. The user has a set of fields and methods. It is combined with other classes of the program, namely, the game and the dictionary. One user can have many dictionaries and games created. The dictionary, in turn, consists of words that have a set of methods for working with them. The game consists of a set of words that will be used in it and the results. The result is a class that contains the game and the user who owns the points earned.

Description of system requirements. The requirements for the software are as follows: for the full operation of the system with a large number of users, the server side must quickly and error-free process all requests and return the result of execution to the users.

Performance requirements. A graphical user interface must be created that is simple, functional and user friendly. Considerations of users will be taken into account, and thus the interface will be developed.

Security requirements. All personal data of the users must be stored in encrypted form. At the time of writing this section, a systematic analysis of the task was carried out. In order to understand what stages of work should be completed, a goal tree was built. Requirements for this system were also created using the goal tree. The main requirement is the reliability and security of the system. The flowchart described the basic input and output data that will be required for the system to function. Similar systems were also analyzed in the market, resulting in the best features of each system. In addition, a class diagram was designed to help you understand what kind of logical structure a system should have.

4 Software for Solving Problems

4.1 Choosing and Justifying the Means of Solving the Problem

In order to achieve this goal, you must first choose a programming language and a software development environment that is able to solve the problems. The most popular languages for creating web applications are considered; C#, JavaScript, Java, PHP, Python, Ruby. All of these languages have advantages and disadvantages, and are suitable for specific tasks and are not completely universal. C# is the best architecture for Windows programming and automation. The Java language is more appropriate than C#, but the impact of the garbage collection on performance has been chosen not to choose this language.

Next, consider the pros and cons of the PHP language.

Pros:

- This is a fast and easy server-side scripting tool for web development and sharing.
- Large community, widely used. Most of the problems that a web developer faces are pre-existing solutions.
- It works well with databases, file systems, images and more.
- It has garbage collection and dynamic stretch to fit the type of container that shies away from accounting required by static languages.

Cons:

- Global configuration settings can change the semantics of the language, complicating deployment.
- Many special purpose forms, but not general purpose forms.
- It is constantly changing. There was no PHP release that had a number of problems. The only way to get rid of them is to upgrade, but then you get new problems because they do not have separate sustainability and development industries.
- An ugly and unpredictable type system.

After analyzing all the languages, the chosen language is JavaScript, because it is the most complete task. Next, select the database. The most popular solutions in this field are MySQL, PostgreSQL, Firebird, SQLite, ComosDB, MongoDB, Casandra and TSQL. MySQL is the most popular open source database. Because the struggle between open source software and adaptability. Here are some pros and cons of MySQL to consider.
Advantages of MySQL

- Data security
- Zoom on demand
- Hours of operation time
- Comprehensive transaction support
- Full control of the workflow
- Open source flexibility
- Widely accepted and easy to use
- Fast, portable and secure

The disadvantages of MySQL

- Limited to complex business logic
- Stability
- Operations are not handled very efficiently
- Functionality is usually highly addictive
- Prior knowledge is required

MongoDB is a standard, document-based, distributed database created for advanced application developers and the cloud era. No database is more productive to use.
Pros

- It's fast and easy to use.
- The engine supports JSON and other NoSQL documents.
- Data of any structure can be stored and accessed quickly and easily.
- The scheme can be written without downtime.

Cons

- Setting up can be a long process.
- The default settings are not secure.

Therefore, JavaScript and the MongoDB database management system will be used to implement this project. The selected software products are optimal for solving the tasks. In addition, you do not need to buy a license or incur any additional costs for the non-commercial use of these applications. All these arguments make the chosen tools the best solution to the problem.

4.2 Specifications of the Selected Software

Since the main development tools are selected, you can consider them and additional tools in more detail. JavaScript is a scripting language used to create and control the dynamic content of a website, that is, anything that moves, updates, or otherwise. In case of developed system, this language will be used for server and client side. Node.js platform will be used to use the language on the server side. It is a runtime environment that includes everything you need to run a JavaScript program.

The React framework was selected for the client part. React is a JavaScript library for creating user interfaces. This is the browsing layer for web applications.

In general, React components are written to write React components that correspond to different elements of the interface. These components are then organized inside the top-level components that determine the structure of the program.

It is important that every component of React adheres to strict data management principles. Complex, interactive user interfaces often include complex data and program status.

Components are written in HTML style and CSS is used to style them.

HTML is a computer language designed to create websites. Anyone else connected to the Internet can then browse these websites. Learning is easy enough, since the basics are accessible to most people in one go. It is constantly being revised and evolved to meet the needs and demands of a growing Internet audience led by W3C, an organization responsible for language development and support.

CSS stands for Cascading Style Sheets with a Focus on Style. Although HTML is used to structure your web document (defining things like headers and paragraphs, and allowing you to insert images, videos, and other media), CSS determines the style of your document - page layouts, colors, and fonts are determined by CSS. The developed system will use the SCSS preprocessor.

SCSS is a preprocessor that uses features that are not yet part of the broader CSS standard and provides better workflows to support style sheets.

With the SCSS preprocessor, you can reduce the number of repetitions and ensure that clean, accessible code is written.

SCSS is fully compatible with CSS syntax while supporting all Sass power.

GraphQL will be used to link the backend and the frontend.

GraphQL is a syntax that describes how to query data, and is typically used to download data from a server to a client. GraphQL has three main features:

- Allows the client to specify exactly what data they need.
- Facilitates data aggregation from multiple sources.
- A type system is used to describe the data.

With GraphQL you can make one call to get the information you need, not to build multiple REST requests to get the same. Since fixed assets are selected, the principles of their interaction can be considered. Therefore, the main entities are the server and the client. The client can be any device on which the browser is installed. When a user opens a browser and navigates to a site, a site page created using the React framework, which in turn is based on Javascript, opens. The frontend then sends a request to a server that builds on Node.js and the server processes the request, accesses the database, and sends the client the necessary data.

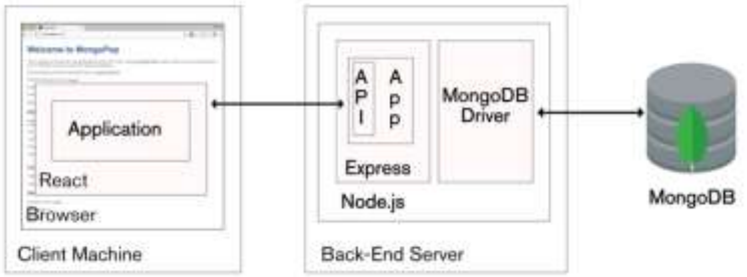


Fig. 10. System architecture

JavaScript is the main language of development, React is selected for use with the web browser, which is composed using CSS and HTML. MongoDB is selected to handle the data. The Visual Studio Studio Code is selected as the programming environment.

5 Practical Implementation

5.1 Description of Implementation

General Information. The software created is called Language Helper. The software needed to deploy the program:

- Operating system (Linux, Windows, Mac OS)
- Node.js
- Yarn package manager
- Apollo server

Software required to use:

- Web browser (Chrome, Opera, Firefox, IE version 6.0)

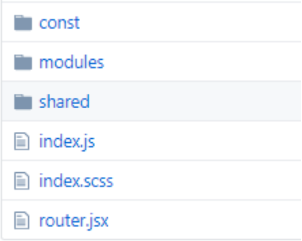
The product is designed using the following programming languages:

- JavaScript - logical modules, backend and system frontend
- HTML and CSS - visualization of the system interface

Functional purpose. The system is designed to increase the vocabulary of a foreign language and to study the language as a whole, it solves the following classes of problems:

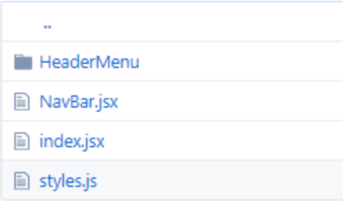
- Create dictionaries.
- Import and export words
- Creating interactive games
- Suggest new words to the user
- Use shared dictionaries

Description of the logical structure. The project under development consists of two repositories. The first repository is the frontend part, the second one is the backend. Let's take a closer look at each of them. The frontend repository contains two files, the main files, index.jsx and index.scss (Fig. 11). The first file is a component that connects all the files that contain the program code. The second file is responsible for connecting all styles that are indispensable for visual display of the system. The next important file is router.jsx, which is responsible for navigating the pages of the system. Next, consider the folders. The first is the const folder, which contains certain constants of variable values, such as page names, configurations, and a list of languages. The next folder is modules. This folder is the largest by volume because it contains all the pages of this system. Each page has a separate folder that contains the component source code file and its stylesheets. The query component folders also contain the files needed for this (Fig. 12).



const
modules
shared
index.js
index.scss
router.jsx

Fig. 11. Logical structure of the frontend repository



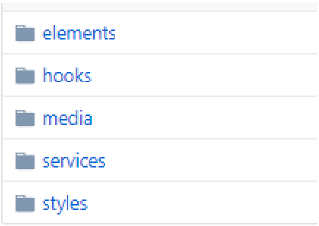
..
HeaderMenu
NavBar.jsx
index.jsx
styles.js

Fig. 12. Logical structure components

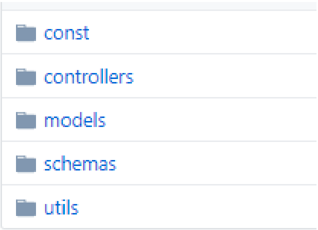
Next let's look at the shared folder. This folder stores clearly structured code that is used in several modules of the program (Fig. 13). Consider its structure in more detail. Elements is a folder that contains elements that are used in modules, such as input components, modal windows, and more.

- Hooks - contains code that is responsible for logical operations that are used in several components.
- Media - Contains images used on the system.
- Services - contains services that provide work between system components.

- Styles - Provides a description of system colors and styles that are common to several components.



elements
hooks
media
services
styles



const
controllers
models
schemas
utils

Fig. 13. Folder logical structure shared.

Fig. 14. The logical structure of the backend repository

Let's go to the backend of the repository. It has a somewhat similar structure to the frontend of the repository (Fig. 14).

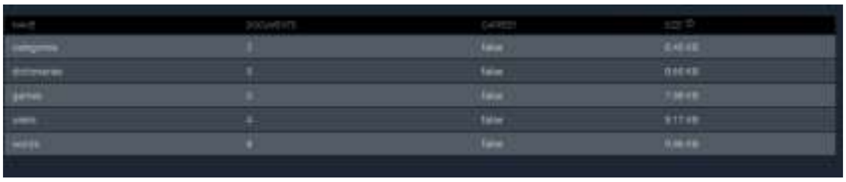
- Const - Contains database configurations, image repositories and parameters for and encrypted parameters for authorization.
- Controllers - contains all controllers used in the application.
- Models - contains a description of the structure of the database tables.

Schemas are query schemes that are used in the GraphQL language to send and receive data. Utils is a set of functions that perform computational operations such as encrypting user data, uploading and processing images, and more.

Description of the database structure. The database consists of 5 main tables (Fig. 15).

- Users - Contains user information. Each user has his / her login and password and additional information.
- Dictionaries - Includes an array of words, the name of the dictionary if it is user-defined and the languages of translation.
- Words - consists of the original word and an array of foreign language analogues
- Games - contains a list of words for the game and the results of the game passing by users.

Categories is a list of categories that are available on the system for selection when adding words. Users can create their own categories.



NAME	DOCUMENTS	CAPPED	SIZE
categories	2	false	0.40 KB
dictionaries	5	false	0.60 KB
games	0	false	7.38 KB
users	4	false	9.17 KB
words	8	false	0.98 KB

Fig. 15. Database structure

Technical means used. Because developed software can run on any platform, any hardware device on which the browser is installed can be used to operate it.

Call and download. To boot the system, you must use the Yarn Batch Manager to install all the required modules that system uses. Then, in the host.js file, specify the port on which the server part will run. To configure the database, you must modify the database link in the database-config.js file. If a remote server is used as the database server, the system is ready to run. If you use a local server, you should run it using the MongoDB command in the installed server folder. Then use the yarn start command to start the server. To run the frontend on this system, you must specify in the base-url.js file the address and port on which the backend server is running. Then run the yarn start command. To deploy the system to a remote server, you must run the yarn deploy command instead of the yarn start command.

Incoming data. In the program, the input is done using the keyboard and the mouse. The initial input to the system is user data. The following information is required to register in the system: user name; email; password. Once registered, this information will be used to log in. As you continue to work with the system, the input data may vary. To create a dictionary, the input is: the language of the original; language of translation; dictionary name (optional field). To add a word to the dictionary, you must enter the following information: the word in the original; word category (optional field); a list of translations. The input may also be an xls document that contains a list of words. The editing page of the input may have the same data as the login, as well as additional fields that are responsible for setting up the profile.

Output data. As a result of the system, the output can be represented as:

- a list of dictionaries;
- a list of words;
- a list of interactive games;
- rating of users;
- activity schedules;
- PDF file with word list.

Classes of solved problems

- creating dictionaries;
- creating interactive dictionary-based games;
- storage of game results;
- display of user activity;
- export dictionaries to a file.

Functional restrictions on use. Minimum hardware requirements for the system to be deployed:

- RAM - 4GB
- Processor frequency - 3.6 GHz
- Memory capacity is 4GB
- Internet access speed - 20 MB/s

5.2 Analysis of the results

In order to get better acquainted with all the functionality of the system, let's consider a test example. Getting started with the system starts from the authorization page where you need to fill in all the fields (Fig. 16). If not all fields are filled, the button is inactive. If the field contains an error, the corresponding text with an error will be displayed (Fig. 17).

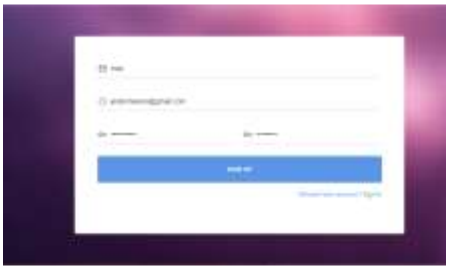
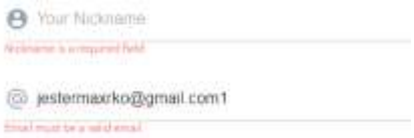


Fig. 16. The registration page



Your Nickname

Nickname is a required field

jestermaxrko@gmail.com1

Email must be a valid email

Fig. 17. Display input errors

After successful registration, the user can log in with their personal data on the corresponding page (Fig. 18). The login page is similar to the registration page, the only difference is the number of fields for entering information.

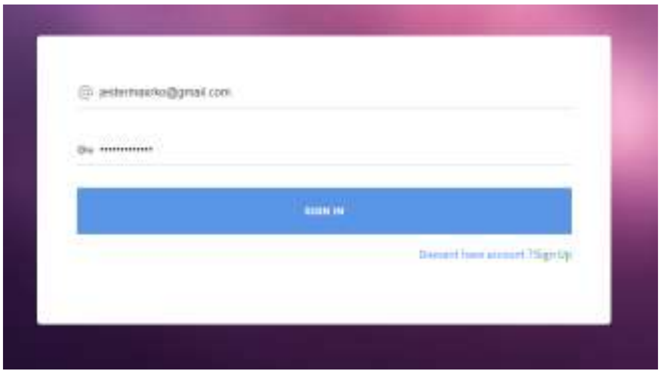


Fig. 18. Login Page

After logging in, the main page is displayed (Fig. 19). On the main page we see a list of added dictionaries. One of them has the name entered by the user, while the others have the languages that are present in the name. The page header contains links to system pages. In the upper right corner is a button with a username and uploaded photo. When this button is clicked, the user will be taken to the edit page of personal information. Above the list of dictionaries is a dictionary search bar, which makes it easier to navigate between dictionaries if there are a large number of dictionaries. If the search did not match, a corresponding message will be displayed (Fig. 20). At the bottom of the dictionary list is a button that allows you to create a new dictionary. Since this system has a responsive design, each page is displayed on any device (Fig. 21).

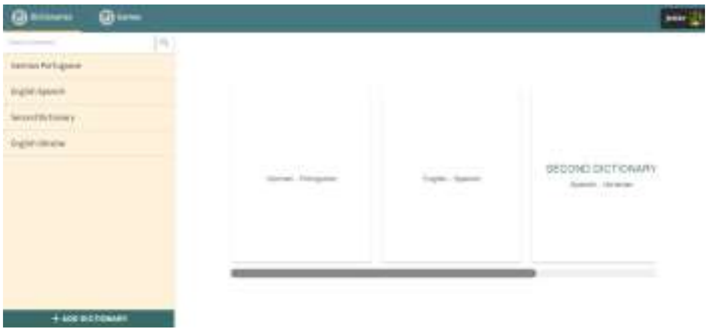
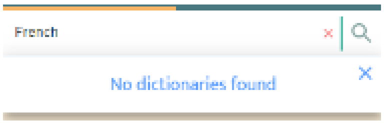


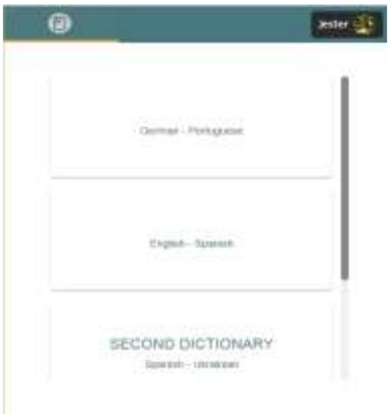
Fig. 19. Home page



French

No dictionaries found

Fig. 20. Search result



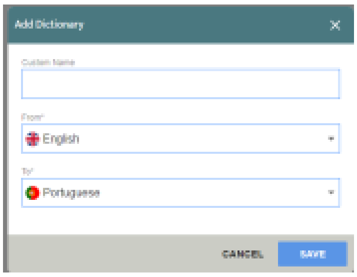
German - Portuguese

English - Spanish

SECOND DICTIONARY
Spanish - Ukrainian

Fig. 21. Home is open on your phone

Next, let's look at the process of creating a dictionary. To do this, press the button described above. After that, a modal window opens with information input fields (Fig. 22). The next page is the dictionary display page (Fig. 23).



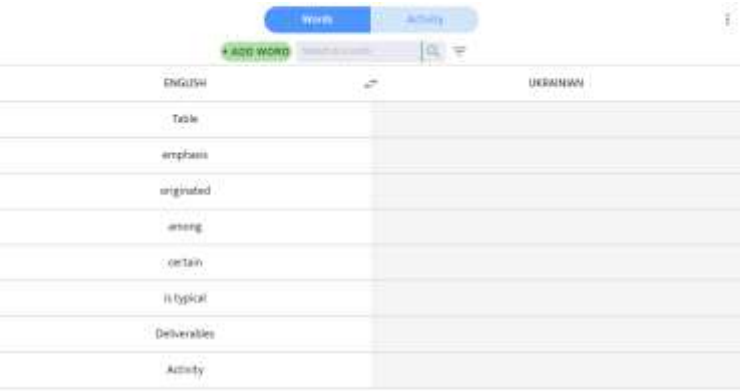
Add Dictionary

Custom Name

From*
English

To*
Portuguese

Fig. 22. Creating a dictionary



ENGLISH	UKRAINIAN
Table	
emphasis	
originated	
among	
certain	
is typical	
Deliverables	
Activity	

Fig. 23. Dictionary

Let's look at components of this page. At the top level there are two main tabs, the first one is the tab that is responsible for displaying the word list, the second one is the user activity. At the same level, in the right corner is a button that is responsible for customizing the dictionary. The next line is word search and filters. It also has a button that adds words to the dictionary. Most of the page is occupied by the word list itself. Let's look at the dictionary settings in more detail (Fig. 24).

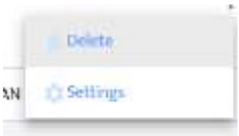
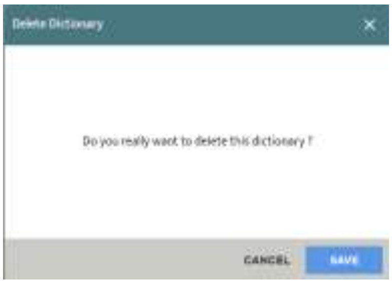


Fig. 24. Customizing the dictionary



Do you really want to delete this dictionary ?

Fig. 25. Deleting a dictionary

The first item is to delete the dictionary. When you call this action, the system asks the user whether he really wants to perform this action (Fig. 25). Clicking on another button from this list will open a dictionary settings page where you can configure the dictionary access rights. Also, at this level there is a button that allows you to open a graph showing user activity (Fig. 26). The Y axis is responsible for the number of words added, the X axis is the day these words were added. Hovering your cursor over a point on the chart displays detailed information about the day.

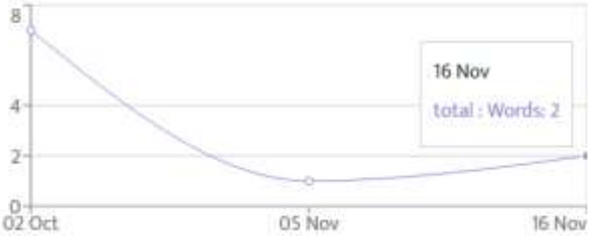


Fig. 26. Activity schedule

Next, let's look at the next level and start with the add word button. By clicking on this button, the user will see a modal window in which all the necessary data for adding a word should be entered (Fig. 27). The main fields are the original word and its translation. It's worth noting that the system allows you to enter multiple translations into one word. To add a translation, you must click on the plus sign. Also, if a user adds a word that is already stored in his vocabulary, then only a new translation will be added, to that word, if no such translation has been made, otherwise a corresponding informational message will be displayed.

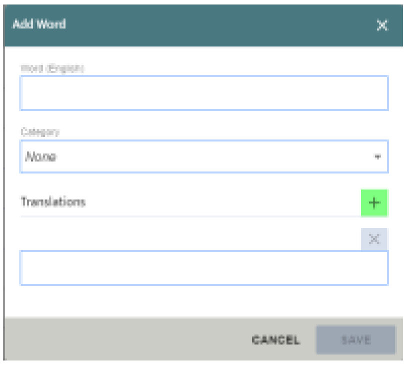


Fig. 27. Adding a word

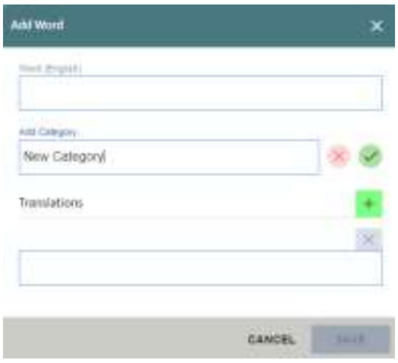


Fig. 28. Creating a category

When adding a word, the user can select a category or create a new one directly in the modal window (Fig. 28). To do this, select the appropriate option from the list. When a category is created, it is automatically inserted into the form.

The next element of the page is the word search box in the dictionary (Fig. 29). The search is performed on any match of the entered character sequence with the word character sequence in the database. The search result can be one or more results.

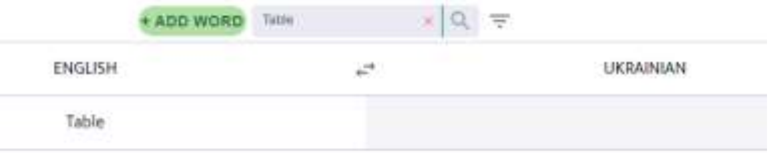


Fig. 29. Word search

A filter that sorts words by a certain criterion performs a similar function. This criterion can be an add date, meaning you can sort the words so that the oldest or newest ones are displayed first. You can also filter words by category (Fig. 30). The filter also has a button that allows you to reset it to its original values.

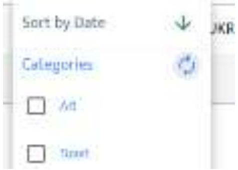
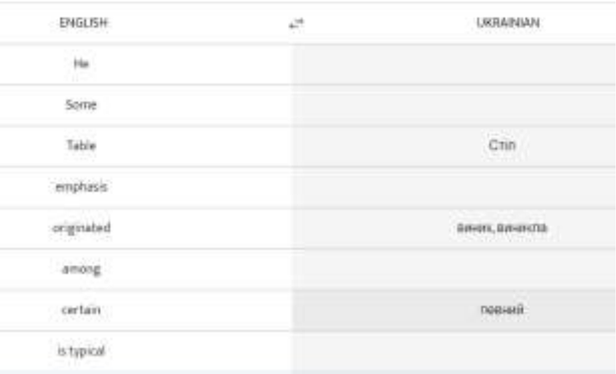


Fig. 30. Filter for words

The most important element of the page is word table (Fig. 31). It consists of two columns. Between the column names is a button that can be used to reposition them.

The right default column does not show words, it allows the user to test themselves for their knowledge of words. Clicking on the second column will display the word.



ENGLISH	⇄	UKRAINIAN
He		
Some		
Table		Стіл
emphasis		
originated		виник, виникла
among		
certain		певний
is typical		

Fig. 31. List of words

This section describes the developed system. A step-by-step guide to deploying the system has been created and basic hardware requirements are described. The inputs and outputs of this system were also described. The main data tables and the logical structure of the system as a whole are considered. The main classes of problem solving are the creation and use of dictionaries. To provide the user with an understanding of how to use the system, a detailed instruction has been created that describes each element of the interface.

6 Conclusions

The purpose of the work is to develop an intelligent online learning system of vocabulary of foreign languages based on gamification. Literature sources were analyzed to achieve this goal, and the most recent approaches and methods were chosen.

After reviewing the literature, a goal tree was built, consisting of several levels. Thereafter, diagrams are created that describe the business logic of the system and the states of the system that cover all levels of the goal tree. The status diagram describes the basic states of the system when interacting with the user. The usage diagram describes the basic functionality of this system that can be used by an authorized user. To show the logical structure of the system, a class diagram was drawn, showing connections across the system objects. The following were considered software tools to achieve the goal. JavaScript is chosen as the language of development of this system, this language is perfectly suited for the task, because it can be used for writing both server and client. After a detailed analysis of the database management solutions, MongoDB was selected. It fits perfectly with the programming language you choose, because the data is stored as objects, meaning that you don't need to use additional adapters to format the data. After selecting the tool, work began to write the system code. The algorithms used in the system, the logical structure of the server and the client were described in detail. The structure of the database was illustrated and the internal implementation of each essence of the database was considered in detail. Basic technical requirements for the system and classes of tasks were also described. The input and output data used in the process of user operation with the system were described. There are also two ways to deploy a system, one on a local server and another on a remote server. Detailed deployment instructions have been created for both of the above options. The developed system has certain security requirements, the main of which is the storage of user data. To demonstrate the performance of the system, there is a reference example, which can also be used as an illustrated instruction manual. The control example looks at all the basic elements of the GUI system.

An economic analysis of the feasibility of the work was also carried out and the main options and methods of implementation of this system were given. The results of the economic analysis show that the implementation of the system has a positive economic potential, this is evidenced by a coefficient of competitiveness greater than 2. Therefore, we can conclude that the goal has been achieved, because the result is a fully operational and competitive system, which has a wide functional spectrum and can be implemented by different methods and cover different classes of problems.

References

1. Osipov, I. V., Nikulchev, E., Volinsky, A. A., Prasikova, A. Y.: Study of gamification effectiveness in online e-learning systems. In: International Journal of advanced computer science and applications, 6(2), 71-77. (2015)
2. Caponetto, I., Earp, J., Ott, M.: Gamification and education: A literature review. In: European Conference on Games Based Learning, Academic Conferences International Limited, 1, 50. (2014)
3. Yeh, M. K. C., Toshtzar, A., Guertin, L., Yan, Y.: Using spaced repetition and gamification to enhance K-12 student science literacy with on-demand mobile short reads. In: IEEE Frontiers in Education Conference (FIE), 1-4. (2016)
4. Tejedor García, C., Escudero Mancebo, D., González Ferreras, C., Cámara Arenas, E., Cardeñoso Payo, V.: Improving L2 production with a gamified computer-assisted pronunciation training tool, TipTopTalk!. (2016)
5. Reitz, L., Sohny, A., Lochmann, G.: VR-based gamification of communication training and oral examination in a second language. In: Computer-Assisted Language Learning: Concepts, Methodologies, Tools, and Applications, IGI Global, 811-828. (2019)
6. Flores, J. F. F.: Using gamification to enhance second language learning. In: Digital Education Review, (27), 32-54. (2015)
7. Rachels, J. R., Rockinson-Szapkiw, A. J.: The effects of a mobile gamification app on elementary students' Spanish achievement and self-efficacy. In: Computer Assisted Language Learning, 31(1-2), 72-89. (2018)
8. Leba, M. O. N. I. C. A., Ionica, A., Apostu, D. R. A. G. O. Ș.: Educational software based on gamification techniques for medical students. In: Recent Advances in Computer Engineering, Communications and Information Technology. Proceedings of the 5th International Conference on Applied Informatics an dComputer Theory. Tenerife: AICT, 225-230. (2014)
9. Rego, I. D. M. S.: Mobile language learning: How gamification improves the experience. In: Handbook of mobile teaching and learning, 705. (2015)

10. Villagrá-Arnedo, C., Gallego-Durán, F. J., Molina-Carmona, R., Llorens-Largo, F.: PLMan: towards a gamified learning system. In: International Conference on Learning and Collaboration Technologies, Springer, Cham, 82-93. (2016)
11. Shakhovska, N., Vysotska, V., Chyrun, L.: Features of E-Learning Realization Using Virtual Research Laboratory. In: Computer Science and Information Technologies, Proc. of the XI–th Int. Conf. CSIT'2016, 143–148 (2016)
12. Shakhovska, N., Vysotska V., Chyrun, L. Intelligent Systems Design of Distance Learning Realization for Modern Youth Promotion and Involvement in Independent Scientific Researches. In: Advances in Intelligent Systems and Computing 512. Springer International Publishing AG, 175–198 (2017)
13. Rzheuskyi, A., Gozhyj, A., Stefanchuk, A., Oborska, O., Chyrun, L., Lozynska, O., Mykich, K., Basyuk, T.: Development of Mobile Application for Choreographic Productions Creation and Visualization. In: CEUR Workshop Proceedings, 2386, 340-358. (2019)
14. Leoshchenko, S., Oliinyk, A., Skrupsky, S., Subbotin, S., Zaiko, T.: Parallel Method of Neural Network Synthesis Based on a Modified Genetic Algorithm Application. In: CEUR Workshop Proceedings, Vol-2386, 11-23. (2019)
15. Bisikalo, O., Ivanov, Y., Sholota, V.: Modeling the Phenomenological Concepts for Figurative Processing of Natural-Language Constructions. In: CEUR Workshop Proceedings, Vol-2362, 1-11. (2019)
16. Levchenko, O., Romanyshyn, N., Dosyn, D.: Method of Automated Identification of Metaphoric Meaning in Adjective + Noun Word Combinations (Based on the Ukrainian Language). In: CEUR Workshop Proceedings, Vol-2386, 370-380. (2019)
17. Shakhovska, N., Vysotska V., Chyrun, L.: Intelligent Systems Design of Distance Learning Realization for Modern Youth Promotion and Involvement in Independent Scientific Researches. In: Advances in Intelligent Systems and Computing 512. Springer International Publishing AG, 175–198 (2017)
18. Lytvyn, V., Vysotska, V., Veres, O., Rishnyak, I., Rishnyak, H.: The Risk Management Modelling in Multi Project Environment.. In: Computer Science and Information Technologies, Proc. of the Int. Conf. CSIT, 32-35 (2017)
19. Korobchinsky, M., Vysotska, V., Chyrun, L., Chyrun, L.: Peculiarities of Content Forming and Analysis in Internet Newspaper Covering Music News, In: Computer Science and Information Technologies, Proc. of the Int. Conf. CSIT, 52-57 (2017)
20. Lytvyn, V., Vysotska, V., Burov, Y., Veres, O., Rishnyak, I.: The Contextual Search Method Based on Domain Thesaurus. In: Advances in Intelligent Systems and Computing, 689, 310-319 (2018)
21. Rusyn, B., Lytvyn, V., Vysotska, V., Emmerich, M., Pohreliuk, L.: The Virtual Library System Design and Development, Advances in Intelligent Systems and Computing, 871, 328-349 (2019)
22. Kanishcheva, O., Vysotska, V., Chyrun, L., Gozhyj, A.: Method of Integration and Content Management of the Information Resources Network. In: Advances in Intelligent Systems and Computing, 689, Springer, 204-216 (2018)
23. Naum, O., Chyrun, L., Kanishcheva, O., Vysotska, V.: Intellectual System Design for Content Formation. In: Computer Science and Information Technologies, Proc. of the Int. Conf. CSIT, 131-138 (2017)
24. Vysotska, V., Lytvyn, V., Burov, Y., Gozhyj, A., Makara, S.: The consolidated information web-resource about pharmacy networks in city, CEUR Workshop Proceedings, 239-255 (2018)
25. Gozhyj, A., Kalinina, I., Vysotska, V., Gozhyj, V.: The method of web-resources management under conditions of uncertainty based on fuzzy logic, 2018 IEEE 13th Interna-

tional Scientific and Technical Conference on Computer Sciences and Information Technologies, CSIT 2018 – Proceedings 1, 343-346 (2018)

26. Lytvyn, V., Vysotska, V., Dosyn, D., Burov, Y.: Method for ontology content and structure optimization, provided by a weighted conceptual graph, Webology, 15(2), pp. 66-85 (2018)
27. Lytvyn, V., Vysotska, V., Dosyn, D., Lozynska, O., Oborska, O.: Methods of Building Intelligent Decision Support Systems Based on Adaptive Ontology, Proceedings of the 2018 IEEE 2nd International Conference on Data Stream Mining and Processing, DSMP 2018, 145-150 (2018)
28. Gozhyj, A., Vysotska, V., Yevseyeva, I., Kalinina, I., Gozhyj, V.: Web Resources Management Method Based on Intelligent Technologies, Advances in Intelligent Systems and Computing, 871, 206-221 (2019)
29. Su, J., Vysotska, V., Sachenko, A., Lytvyn, V., Burov, Y.: Information resources processing using linguistic analysis of textual content. In: Intelligent Data Acquisition and Advanced Computing Systems Technology and Applications, Romania, 573-578, (2017)
30. Rusyn, B., Vysotska, V., Pohreliuk, L.: Model and architecture for virtual library information system, 2018 IEEE 13th International Scientific and Technical Conference on Computer Sciences and Information Technologies, CSIT 2018 – Proceedings 1, 37-41 (2018)
31. Vysotska, V.: Linguistic Analysis of Textual Commercial Content for Information Resources Processing. In: Modern Problems of Radio Engineering, Telecommunications and Computer Science, TCSET'2016, 709–713 (2016)
32. Lytvyn, V., Sharonova, N., Hamon, T., Vysotska, V., Grabar, N., Kowalska-Styczen, A.: Computational linguistics and intelligent systems. In: CEUR Workshop Proceedings, Vol-2136 (2018)
33. Vasyl, Lytvyn, Victoria, Vysotska, Dmytro, Dosyn, Roman, Holoschuk, Zoriana, Rybchak: Application of Sentence Parsing for Determining Keywords in Ukrainian Texts. In: Computer Science and Information Technologies, Proc. of the Int. Conf. CSIT, 326-331 (2017)
34. Chyrun, L., Vysotska, V., Kis, I., Chyrun, L.: Content Analysis Method for Cut Formation of Human Psychological State, Proceedings of the 2018 IEEE 2nd International Conference on Data Stream Mining and Processing, DSMP 2018, 139-144 (2018)
35. Lytvyn, V., Vysotska, V., Chyrun, L., Chyrun, L.: Distance Learning Method for Modern Youth Promotion and Involvement in Independent Scientific Researches. In: Proc. of the IEEE First Int. Conf. on Data Stream Mining & Processing (DSMP), 269-274 (2016)
36. Vysotska, V., Rishnyak, I., Chyrun L.: Analysis and evaluation of risks in electronic commerce, CAD Systems in Microelectronics, 9th International Conference, 332-333 (2007).
37. Burov, Y., Vysotska, V., Kravets, P. Ontological approach to plot analysis and modeling. CEUR Workshop Proceedings, Vol-2362, 22-31 (2019)
38. Chyrun, L., Kis, I., Vysotska, V., Chyrun, L.: Content monitoring method for cut formation of person psychological state in social scoring, 2018 IEEE 13th International Scientific and Technical Conference on Computer Sciences and Information Technologies, CSIT 2018, 106-112 (2018)
39. Vysotska, V., Lytvyn, V., Burov, Y., Berezin, P., Emmerich, M., Fernandes, V. B.: Development of Information System for Textual Content Categorizing Based on Ontology. In: CEUR Workshop Proceedings, Vol-2362, 53-70 (2019)
40. Lytvyn, V., Pukach, P., Bobyk, I., Vysotska, V.: The method of formation of the status of personality understanding based on the content analysis. In: Eastern-European Journal of Enterprise Technologies, 5/2(83), 4-12 (2016)

41. Lytvyn V., Vysotska V., Pukach P., Nytrebych Z., Demkiv I., Kovalchuk R., Huzyk N.: Development of the linguometric method for automatic identification of the author of text content based on statistical analysis of language diversity coefficients, Eastern-European Journal of Enterprise Technologies, 5(2), 16-28 (2018)
42. Lytvyn, V., Vysotska, V., Rzheuskyi, A.: Technology for the Psychological Portraits Formation of Social Networks Users for the IT Specialists Recruitment Based on Big Five, NLP and Big Data Analysis. In: CEUR Workshop Proceedings, Vol-2392, 147-171. (2019)
43. Lytvyn, V., Vysotska, V., Rusyn, B., Pohreliuk, L., Berezin, P., Naum O.: Textual Content Categorizing Technology Development Based on Ontology. In: CEUR Workshop Proceedings, Vol-2386, 234-254. (2019)
44. Vysotska V., Lytvyn V., Kovalchuk V., Kubinska S., Dilai M., Rusyn B., Pohreliuk L., Chyrun L., Chyrun S., Brodyak O.: Method of Similar Textual Content Selection Based on Thematic Information Retrieval. In: 2019 IEEE 14th International Scientific and Technical Conference on Computer Science and Information Nechnologies (CSIT), 1-6. (2019)
45. Lytvyn V., Vysotska V., Peleshchak I., Basyuk T., Kovalchuk V., Kubinska S., Chyrun L., Rusyn B., Pohreliuk L., Salo T.: Identifying Textual Content Based on Thematic Analysis of Similar Texts in Big Data. In: 2019 IEEE 14th International Scientific and Technical Conference on Computer Science and Information Nechnologies (CSIT), 84-91. (2019)