UDC 004.9 **doi: 10.26906/SUNZ.2022.4.088**

A. Kapiton[1], O. Skakalina[1], R. Baranenko[2]

[1] National University «Yuri Kondratyuk Poltava Polytechnic», Poltava, Ukraine
[2] Uman National University of Horticulture, Uman, Ukraine

DESIGN AND SOFTWARE IMPLEMENTATION OF A KNOWLEDGE EXCHANGE WEB SERVICE

Abstract. The article analyzes the problem of system development and implementation of a knowledge sharing web service. The purpose of the article is to research the possibilities of current technologies used in the process of designing and software implementation of a knowledge exchange web service. The methods of developing the designed software product were analyzed. The paper considers the purpose of a web service for knowledge exchange, taking into account the definition of the target audience of the developed product. Web technologies for software development are selected and substantiated by the authors. The proposed structure of the knowledge sharing web service.

Keywords: web service, information exchange, programming language, design, software product.

Introduction

The centralized nature of data management in the database requires the existence of some person (group of persons) who is entrusted with the functions of administering the data stored in the database. The main task of the database is the guaranteed preservation of significant amounts of information and providing access to it to the user or the application program. Thus, the database consists of two parts: stored information and its management system. In order to ensure the efficiency of access, data records are organized as a set of facts (data element).

Designing a relational database consists of designing the tables that represent the subject area and defining the relationships between them. The "entity-relationship" model (ER-model) is a data model that allows you to describe conceptual schemes using generalized block structures.

An ER model is a data meta-model, that is, a means of describing data models.

Analysis of recent research and publications. The problem of developing special knowledge exchange software is of interest to many scientists. To create the key characteristics of the model of the knowledge exchange system being developed, an analysis of a number of publications dedicated to the analysis of existing services was carried out and their features and shortcomings were determined. M. Beshley, M. Klymash, H. Beshley, O. Urikova, Yu. Bobalo, Yu. investigate the problems of using network equipment for the implementation of a knowledge exchange web service [1]. J. Bloomberg, N. Shpak, O. Kuzmin, Z. Dvulit, T. Onysenko, W. Sroka studies the main issues of digitization and digital transformation [2, 9]. The problem of structuring the fuzzy knowledge base of impact factors of the IT industry is considered by M. Bublyk, A. Karpyak, O. Rybytska, Y. Matseliukh, N. Chukhrai, T. Shcherbata [3, 4, 5, 6]. The modern development of IT clusters and ways of its intensification are studied by M. Melnyk, E. Korcelli-Olejniczak, N. Chorna, N. Popadynets, O. Pyrog, A. Poritska, K. Tan, S. Chong and others [7, 8, 10].

Main part

In the aspect of the investigated problem, the task was set to design and programmatically implement a web service for knowledge exchange, which should ensure the following functions: separation of the user and administrative parts of the software complex; the registered user must be able to log into the system without hindrance; ensuring that users create questions and answer them; ensuring the creation of a user profile; ensuring the search for questions; ensuring the creation of tags and filtering of questions by tags; draft for questions and answers; the possibility of evaluating answers and questions; commenting on answers; admin panel with full control.

A database management system (DBMS) is a set of software and language tools necessary for creating databases, maintaining them in an up-to-date state, and organizing the search for the necessary information in them.

The service uses the PostgreSQL DBMS. PostgreSQL is an object-relational database management system. It is an alternative to both commercial databases (Oracle Database, Microsoft SQL Server, IBM DB2 and others) and open source databases (MySQL, Firebird, SQLite). Supports simultaneous database modification by several users using the MultiVersion Concurrency Control (MVCC) mechanism. This fulfills ACID requirements and virtually eliminates the need for read blocking.

It is based on important semantic information and is intended for logical presentation of data. Defines the value of data in the context of their relationship with other data.

The fact that all existing data models (hierarchical, network, relational, object) can be generated from the entity-relationship model is important, so it is the most general. Any fragment of the visual area can be represented as a set of entities, between which there are a certain set of connections. Django uses multiple related tables. One-to-many connections are established between them. For example, one user may have many questions.

We will describe the database of the web service, which was developed in the process of performing the assigned task. It consists of the following main tables:

1. Question_questions (table of questions), it stores all data about questions:

• title (question title field);
• image (picture field, if available);
• slug (field for searching by url);
• content (question description field);
• status (issue status: printed\draft);
• create_date (date of creation);
• update_date (update date);
• uuid (unique id);
• create_user (user who created).

2. Question_answers (table of answers):

• answer (answer field);
• date (date of creation);
• uuid (unique id);
• question_id (id of the question to which the answer is given);
• user_id (id of the user who answered).

3. Question_tags (table of tags):

• tag (tag field);
• uuid (unique id);
• question_id (id of the question to which the tag belongs).

4. Useraccout_useraccout (user profile table):

• avatar (user avatar field);
• description (information about the user);
• uuid (unique id);
• user_id (id of the user to whom the profile belongs).

5. Auth_user (user authorization table):

• password (password);
• last_login (last login);
• is_superuser (administrator rights);
• username (user name);
• firt_name (name);
• last_name (surname);
• date_joined (date of registration).

6. Question_upvotes (table of the number of likes):

• answer_id (answer id);
• question_id (question id);
• uuid (unique id).

Most tables are linked to one or more tables by a single field. This field is a unique identifier for each record. This is reflected in more detail in Table 1.

This model of database organization is most suitable for this system. This organization allows you to make some changes to the system. The user can change his data, edit questions and answers. Fields are not duplicated, the database is normalized.

Table 1 – **Table of common connections**

Table	Data	Connections with other tables
Django_session	Django session information	Absent
Django_migrations	Information about django migrations	Absent
Django_site	Information about the project	Absent
Question_upvotes	Data about preferences	Question_answers through answer_id
Question_answers	Response data	Auth_user through auth_user_id, question_questions through question_id
Question_questions	Question data	Auth_user through auth_user_id
Authtoken_token	Access key for authorized user	Auth_user through auth_user_id
Useraccount_useraccount	User profile	Auth_user through auth_user_id
Account_emailconfirmation	Data on sending confirmation mail	Account_emailadress through Account_emailadress_id
Account_emailadress	Mail confirmation data	Auth_user through auth_user_id
Socialaccount_socialaccount	Data about social accounts	Auth_user through auth_user_id

This model of database organization is most suitable for this system. This organization allows you to make some changes to the system. The user can change his data, edit questions and answers. Fields are not duplicated; the database is normalized.

The choice to use Python for backend development is justified by the fact that it is an interpreted language, which allows you to save a significant amount of time that is usually spent on compilation.

Python is an extensible language: knowing C allows you to add new built-in functions or modules to perform critical operations at maximum speed or to write an interface to commercial libraries available only in binary form. A Python language interpreter can be built into a program written in C and used as an extension or command language for that program. Python is currently used by tens of thousands of programmers worldwide, and the number of people using it is growing rapidly, doubling and tripling every year. This language works equally well on Windows, UNIX, Macintosh, and OS/2, and can be used to easily develop both small applications or scripts and to deploy large programs. The performance of the created application depends on its features. Of course, for a numerical algorithm that performs ordinary integer arithmetic in a 'for' loop, it doesn't matter what language it's written in. However, in the future, modern computers have so much unused computing potential

that for 90% of applications, performance is related to the choice of language. Java also compiles to bytecode, but is currently slower than Python in most cases. In addition, it is very easy to combine Python with modules written in C or C++, which can be used to increase the speed of programs in critical areas.

The choice to use the Django framework is based on the fact that it is a high-level open Python framework (software framework) for developing web systems. A Django site is built from one or more parts, which are recommended to be modular. Django's initial development as a tool for working with news resources has had a strong influence on its architecture: it provides a number of tools that help in the rapid development of informational websites. Django was created to manage news sites, but since the beginning of its distribution in the status of open software, it has gained enormous popularity around the world as a platform for numerous systems.

The use of Nginx as a web server (a free web server and a proxy server) is due to the fact that its working processes serve many connections at the same time, multiplexing them with operating system calls select, epoll (Linux) and kqueue (FreeBSD). Work processes perform a cycle of processing events from descriptors (see event-oriented programming). The data received from the client is analyzed using a finite state machine.

Conclusions

In the process of work, a web service was developed for the exchange of knowledge, which, according to the authors, is completely ready for use. The software product is a separate web application that will significantly affect the search for answers to questions that arise during training.

This application is primarily aimed at teachers of higher education institutions, students and ordinary users. The purpose of creating the application is to provide answers to questions that may arise in the learning process, which will affect the process of knowledge exchange.

This software product has the following development prospects: it can be used in various fields; in the future, it is possible to implement a system of recommendations based on the previous questions; implementation of influence on emerging content.

During the development of the software product, modern web technologies were selected and substantiated. The developed web service for knowledge exchange fully meets all the requirements set at the task setting stage and is ready for use.

REFERENCES

1. Beshley, M., Klymash, M., Beshley, H., Urikova, O. and Bobalo, Yu. Future intent-based networking for QoE-driven business models. Lecture Notes in Electrical Engineering, 831. 2022. 1-18.
2. Bloomberg J. Digitization, digitalization, and digital transformation: confuse them at your peril. Forbes. 2018.
3. Bublyk M., Karpyak A., Rybytska O. and Matseliukh Y. Structuring the Fuzzy Knowledge Base of the IT Industry Impact Factors. IEEE 13th International Scientific and Technical Conference on Computer Sciences and Information Technologies (CSIT). 2018. 21–24. [Electronic resource] - Access mode: https://doi.org/10.1109/STC-CSIT.2018.8526760.
4. Bublyk M.I., Karpiak A.O., Rybytska O.M. The perspectives of ITindustry development in Ukraine on the basis of data analysis of the world economic forum. Innovative management: theoretical, methodical, anapplied grounds/ Pražský Institut zvyšování kvalifikace Prague institute for Qualification Enhancement/ 2018. 115 -128.
5. Chukhrai, N. I., & Shcherbata, T. S. Cooperation between IT development enterprises and universities. Marketing And Management Of Innovations. (3). 2016. 161-169.
6. Karpyak A., Rybytska O. Cluster Analysis Of Motivational Management Of Personnel Support of IT Companies COLINS-2022: 6th International Conference on Computational Linguistics and Intelligent Systems, May 12–13, 2022 Gliwice, Poland. [Electronic resource] - Access mode: http://ceurws.org/Vol-3171/paper119.pdf
7. Melnyk, M., Korcelli-Olejniczak, E., Chorna, N., & Popadynets, N. Development of Regional IT clusters in Ukraine: institutional and investment dimensions. Economic Annals-XXI, (173). 2018. 19-25.
8. Pyrog, O., & Poritska, A. The formation of digital business factors. Economics, Entrepreneurship, Management. 1 (7). 2020. 10-17. [Electronic resource] - Access mode: https://doi.org/10.23939/eem 2020.01.010
9. Shpak, N., Kuzmin, O., Dvulit, Z., Onysenko, T. and Sroka, W. Digitalization of the marketing activities of enterprises: case study. Information, [online] 11(2). 2022.
10. Tan, K. S., Chong, S. C. Internet-based ICT adoption: Evidence from Malaysian SMEs. Industrial Management and Data Systems 109(2). 2009. 224-244.

Received (Надійшла) 30.08.2022
Accepted for publication (Прийнята до друку) 26.10.2022

Проектування та програмна реалізація веб сервісу з обміну знаннями

А. Капітон, О. Скакаліна, Р. Бараненко

Анотація. У статті проаналізовано проблему системної розробки та впровадження веб сервісу з обміну знаннями. Метою статті є дослідження можливостей актуальних технологій, що застосовують у процесі проектування та програмної реалізації веб сервісу з обміну знаннями. Проаналізовано методи розробки спроектованого програмного продукту. У роботі розглянуто призначення веб-сервісу для обміну знаннями, з урахуванням визначення цільової аудиторії розробленого продукту. Авторами обрано та обґрунтовано веб-технології для розробки програмного забезпечення. Запропонована структура веб сервісу з обміну знаннями.

Ключові слова: веб сервіс, обмін інформацією, мова програмування, проектування, програмний продукт.