

SEZIONE 17.

TECNOLOGIE E SISTEMI DELL'INFORMAZIONE

DOI 10.36074/logos-15.11.2024.040

FEATURES OF CREATING NEWS IN THE CAMPUS INFORMATION SYSTEM

Ivan Ternovyy¹

Research supervisor: Orest Khamula²

1. PhD student

Lviv Polytechnic National University, UKRAINE

2. PhD, professor, head of the department of multimedia technologies

Lviv Polytechnic National University, UKRAINE

ORCID ID: 0000-0003-0926-9156

In dormitories, there is often a problem of spreading important news. Wall paper posters are used for this. Their disadvantage is that they are easily missed by students in a hurry. Therefore, it is necessary to create a system of effective news production, that is, to develop a page on which all news will be displayed, as well as the ability to create important news for display on the main page of the site.

The publication of events can be greatly simplified and improved if the employees of the educational institution develop an online service that will help inform users about the latest news of the university. However, the main drawback of many notification systems is their inefficiency in providing information.

Therefore, developing a site that can quickly publish news and events while maintaining relevance, that is, separating them by criteria and placing key events in favorites, is a very important task.

Considering the fact that the system being developed must be integrated with the web product, it is necessary to develop a recommended structure of the database containing the data necessary for the functioning of the site, in particular, the storage of information about users, information about the type of users, articles and all information about them. Some of the most important aspects of a database are listed below. The integration of databases is necessary to optimize the behavior of the web service with important data stored in the process of functional use of the program. MySQL will be used as a database management system in this project.

Since the system is a web application, it is hosted on a remote server. All data is stored directly on the server and pulled locally for display to the user, requiring a constant connection between the server and the database, and between the client and the server. The solution was to host the database locally on the server along with the website.

To start developing a website, you need to decide which programming language to use. Given that the product is a web service, it is worth noting that the MVC architecture is currently actively used for development.

Among a number of popular and flexible programming languages for which frameworks that support MVC patterns have already been written, we chose Ruby [1] and the Ruby on Rails (RoR) [2] framework, which is convenient for developing web applications. Ruby is a high-level dynamic object-oriented programming language developed in Japan in the late 1980s. Known for its simplicity of syntax and accessibility for novice programmers, Ruby is based on the philosophy that programming should be exciting, which is reflected in its familiarity and expressiveness [3]. Dynamic and object-oriented: Ruby is a dynamic programming language, which means that data types are determined automatically at runtime. This makes it easy to write flexible and expressive code: most Ruby objects are instances of classes, making it completely object-oriented. Ruby on Rails (RoR) Library: A Ruby-based web development framework that provides a number of tools and libraries to speed up the process of building web applications. RoR is known for its friendly syntax and conventions and is preferred by many developers.

We made this choice for several reasons:

- a large and active Ruby community
- numerous auxiliary libraries and resources; and
- strict adherence to standards;
- Ruby - a dynamically typed language;
- the flexibility of the Ruby programming language;
- RoR is popular for writing startups.

Since the product requires the development of a large set of functions for organizing user work, we decided to use the well-known RoR-library devise [4], which provides high-quality and fast implementation of user authentication. The user subsystem must be divided into roles: student, manager and administrator. There are several ways to implement role separation. One of them is to create a user model and a role model. A relationship is then established between the "user" model and the "role" model, in which case the "user" model will have a number of properties that will be empty in the concrete object.

The news subsystem is of great importance and simplifies the transfer of information between students. Thanks to this subsystem, there is no need to

SEZIONE 17.

TECNOLOGIE E SISTEMI DELL'INFORMAZIONE

distribute paper ads that cannot be seen. To implement this subsystem, the Post entity was created and the necessary migrations were added to create tables in the database.

After the migration, the database schema was automatically changed to indicate the presence of the "posts" table. According to the manager case diagram, it is necessary to implement the functionality of creating news.

Added association between Post and Manager entity for convenience and tracking of people who created certain news. Associated model data in such a way that the manager can create many news. In turn, news can belong to only one manager. To achieve this, the syntactic construction "*has_man : posts*" was added to the Manager model and "*belongs_to :manager*" for the Post entity. The construct "*belongs_t :manager*" means that the Post model has a foreign key to the Manager entity, in this case it is "*manager_id*". Also added validations to the field "*title*" and "*body*" in the Post model. Display of news created by a specific manager can be seen on his page.

The Ruby on Rails framework uses the I18n library [5] to easily implement localization. The translation files can be found in the config/locales/ directory. Since there are two languages available, two additional directories, en/ and ua/, were created with translation files for each language. A translation.rb file has been created that is executed when the web service starts. This file contains the path to the translation files, the available locales, and the default locale.

The service is designed to facilitate interaction between the hostel administration and students. Using the web service, you can track students who have registered on the site, upload student data to the platform if they are not yet registered, post important news on the site page, notify students of urgent events by e-mail, and manage employee data with administrative tasks on the web resource.

REFERENCES:

- [1] Ruby programming language [Electronic resource]. - Available from: <https://github.com/ruby/ruby>.
- [2] Ruby on Rails framework [Electronic resource]. - Available from: <https://github.com/rails/rails>.
- [3] Navishcho potribna mova prohramuvannya Ruby? [Electronic resource]. - Available from: <https://lemon.school/blog/navishho-potribna-mova-programuvannya-ruby>.
- [4] Devise gem [Electronic resource]. - Available from: <https://github.com/plataformatec/devise>.
- [5] Rails I18n [Electronic resource]. - Available from: <https://guides.rubyonrails.org/i18n.html>.