

WEB APPLICATION FORUM

Honchar L.¹⁾, Matsuk A.²⁾, Smoliak O.³⁾, Serediak R.⁴⁾, Drichyk V.⁵⁾

West Ukrainian National University

¹⁾PhD; ²⁻⁴⁾master; ⁵⁾bachelor

I. Statement of the problem

The development of a Web Forum Application presents an opportunity to leverage modern technologies and approaches to enhance the user experience and functionality of online community platforms. The primary objective is to create a robust and scalable forum application that facilitates engaging discussions, knowledge sharing, and community building. Key challenges include designing an intuitive user interface [2], implementing efficient data management and storage solutions, and ensuring security and performance optimization. The successful development of the Web Forum Application requires a comprehensive understanding of client-server architecture [1], web technologies, database management, and user interface design principles. It also involves thorough testing, user feedback integration, and continuous improvement to deliver a high-quality, user-friendly forum experience.

II. The purpose of the work

The purpose of work is to create a modern and functional tool for communication and information exchange on the Internet based on such a platform as a forum. The work is aimed at researching and implementing the best practices of web development, using modern technologies and tools to build a convenient and effective user interface, ensuring the security and speed of the application, as well as creating a favorable environment for communication and information exchange in the online environment. The result of the work will be a functional Forum web application that will be able to meet the needs of users in communication and information exchange on the Internet.

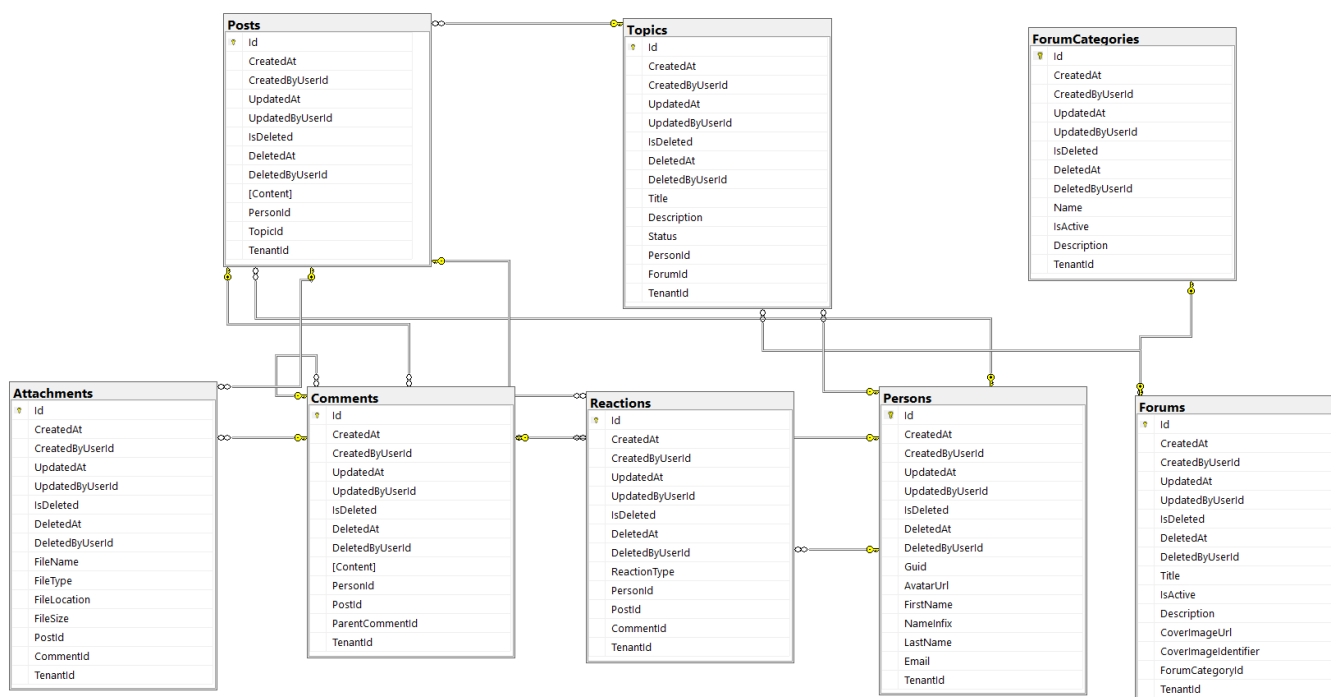
III. Main part

For the development of the Web Application Forum, a variety of technologies have been chosen to ensure its functionality, security, and user-friendliness. These technologies include:

- **Client-Server Architecture:** The application is built on a client-server architecture, which allows for efficient communication between the client-side and server-side components. This architecture is essential for handling user requests, managing data, and ensuring a seamless user experience.
- **.NET 8.0 and ASP.NET:** These frameworks were chosen to write the backend of application and create endpoints. With the help of ASP.NET, it is very convenient to build and support applications according to the MVC model. These frameworks provide a robust foundation for building web applications. They offer a wide range of features and tools that simplify the development process and improve application performance [3].
- **Entity Framework Core 8.0:** This ORM (Object-Relational Mapping) framework simplifies database interactions, making it easier to work with database entities in the application code. It ensures efficient data management and enhances the application's scalability[4]. EntityFramework, together with the Linq library, is a powerful tool in the hands of a developer when writing fast, easy, and understandable data base queries.
- **Refit:** Refit is used to simplify the consumption of RESTful APIs. It provides a clean and type-safe way to define API interfaces, making it easier to integrate external services into the application [5]. We use it to integrate our app with a CloudFlare API.
- **Swagger:** Swagger is used for API documentation, providing a way to document and test APIs. It helps understand the application's API structure and use it effectively[6].
- **Vue.js:** We use Vue.js to write the user interface (front-end), because it is easy and clear to use. It is also one of the top front-end frameworks at the moment. It provides a reactive and component-based framework that makes it easy to create dynamic and interactive UI components [7].
- **Vue-i18N:** Vue-i18N is used for internationalization in the application, allowing it to support multiple languages and regions. This ensures that the application is accessible to users from different parts of the world.

- **Bootstrap-Vue:** Bootstrap-Vue is used for building responsive and mobile-first UI components. It ensures that the application looks great and works well on a variety of devices and screen sizes.
- **Date-Fns:** Date-Fns is used for date manipulation and formatting in the application. It ensures that dates are displayed correctly and consistently across different parts of the application.
- **CloudFlare:** For image storage, our choice fell on CloudFlare, since storing a lot of photos in a database is not a good idea. We chose CloudFlare because it has good API documentation that makes it easy to integrate and use, as well as a great community and support. CloudFlare is used for CDN (Content Delivery Network) services, which improve the performance and reliability of the application. It caches content and optimizes delivery to users, ensuring a fast and seamless user experience [8].
- **MSSQLServer:** MSSQLServer is used as the database management system for the application. It provides a reliable and scalable database solution for storing and managing the application's data [9].

These technologies are essential for the development of a modern and functional Web Application Forum. They ensure that the application is secure, efficient, and user-friendly, providing a seamless experience for users.



Picture1 – Database diagram

Conclusion

The work aimed to research and implement best practices in web development, utilizing modern technologies and tools to build a user-friendly interface, ensuring application security and speed, and fostering a conducive environment for online communication and information exchange.

The result of this endeavor will be a functional Forum web application designed to meet users' needs for online communication and information exchange. Through the application of best practices and modern technologies, the developed forum platform is expected to provide an effective and convenient platform for users to engage in discussions, share knowledge, and build communities online.

List of used sources

1. Client-server architecture. <https://www.britannica.com/technology/client-server-architecture>
2. 3 Key Principles for Creating an Intuitive User Interface. <https://bootcamp.uxdesign.cc/3-key-principles-for-creating-an-intuitive-user-interface-6189a6165134>
3. Introduction to .NET <https://learn.microsoft.com/en-us/dotnet/core/introduction>
4. Entity Framework Core <https://learn.microsoft.com/en-us/ef/core/>
5. Using Refit in .NET. <https://medium.com/p/0843bb199987>
6. API Documentation <https://swagger.io/solutions/api-documentation/>
7. The Progressive JavaScript Framework <https://vuejs.org/>
8. Cloudflare API <https://developers.cloudflare.com/api/>
9. <https://www.microsoft.com/en-us/sql-server>