

УДК 004

**Ю. Береза, Д. Мацик, В. Никитюк, канд. техн. наук; доц.**

(Тернопільський національний технічний університет імені Івана Пулюя, Україна)

## **РЕАЛІЗАЦІЯ КОМУНІКАЦІЇ У КЛІЄНТ-СЕРВЕРНОМУ ЗАСТОСУНКУ ЗА ДОПОМОГОЮ SIGNALR**

UDC 004

**Y. Bereza, st., D. Matsyk, st., V. Nykytyuk, Ph.D.; Assoc. Prof.**

## **IMPLEMENTATION OF COMMUNICATION IN THE CLIENT-SERVER APPLICATION USING SIGNALR**

Інформаційні технології на сьогоднішній день охоплюють багато напрямків, в тому числі і медицину. Поєднавши комунікацію між лікарем та пацієнтом, розробимо клієнт-серверний застосунок. Для реалізації SignalR у проєкті спершу потрібно налаштувати в стартовому класі сервера. Налаштування в стартовому класі наведено у лістингу 1.

Лістинг 1 – Конфігурація SignalR в стартовому класі на стороні сервера

```
public void ConfigureServices(IServiceCollection services) =>
services.AddSignalR();
public void Configure(IApplicationBuilder app, IWebHostEnvironment env,
ServerDbContext dataContext)
{ app.UseEndpoints(endpoints => {
endpoints.MapDefaultControllerRoute();
endpoints.MapHub<UpdateHub>("/update-hub"); });}
```

Далі необхідно створити клас UpdateHub, який буде наслідувати клас Hub. Клас UpdateHub буде реалізовувати у собі обробку даних, та надсилатиме їх у певні групи клієнтам. У лістингу 2 наведено реалізацію UpdateHub.

Лістинг 2 – Реалізація UpdateHub

```
[Authorize]
public class UpdatesHub : Hub {
private readonly IRepository _repository;
public UpdatesHub(IRepository repository){ _repository = repository;}
public async override Task OnConnectedAsync(){ var user = await
_repository.GetUserById(Guid.Parse(Context.User.Claims.First(p => p.Type ==
"Id").Value));
await Groups.AddToGroupAsync(Context.ConnectionId,
user.CustomerCompanyId.ToString()); }}
```

SignalR чудовий інструмент для оновлення баз даних в режимі реального часу. Використання SignalR у проєкті значно заощаджує час реалізації комунікації клієнтів зі сервером.