



DON'T GET SPICY

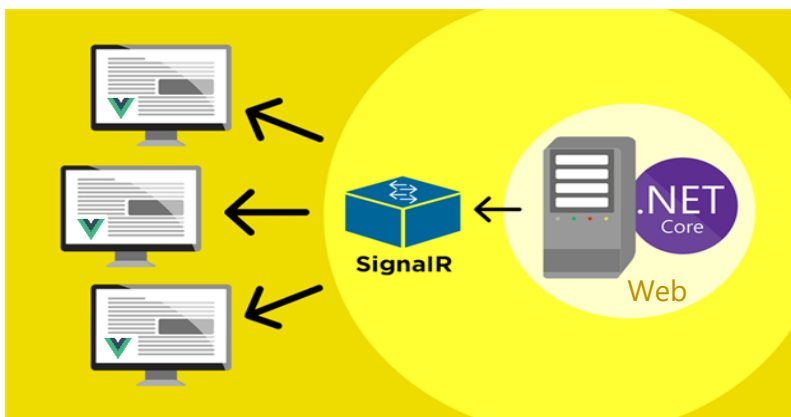
Web aplikacija za kolaborativno realtime igranje društvene igre Ludo (Ne ljuti se čoveče)

Faza 3: Komunikacioni model

Ognjen Damnjanović 16528

Mila Mirović 16742

U okviru projekta DontGetSpicy komunikacioni model sadži asinhronu komunikaciju servera i klijenta pomoću **SignalR** open-source biblioteke, koja omogućava serveru da šalje asinhrona obaveštenja web aplikacijama na strani klijenta, po principu Observer obrasca. Ova biblioteka služi za dodavanje funkcionalnosti ASP.NET aplikacijama u realnom vremenu. SignalR koristi *hub*-ove koji pružaju API visokog nivoa za klijenta i server za međusobno pozivanje metoda.



Za sve vrste komunikacije koristiće se SignalR biblioteka, a to podrazumeva konekciju korisnika u igru, odigravanje poteza i chat. Da bi se SignalR uspešno koristio u okviru Web API komponente, neophodno je dodati sledeće linije koda u Startup.cs fajl:

```
5 using System.Linq;
6 using System.Threading.Tasks;
7 using DontGetSpicy.JWT;
8 using DontGetSpicy.Models;
9 using DontGetSpicy.SignalR;
10 using Microsoft.AspNetCore.Authentication.JwtBearer;
11 using Microsoft.AspNetCore.Builder;
12 using Microsoft.AspNetCore.Hosting;
13 using Microsoft.AspNetCore.HttpsPolicy;

35 public void ConfigureServices(IServiceCollection services)
36 {
37     services.AddCors(opts=>...
41
42     services.AddControllers();
43     services.AddSwaggerGen(c =>...
47     services.AddAuthentication(JwtBearerDefaults.AuthenticationScheme).AddJwtBearer(opts=>...
64     services.AddSignalR();
65     //services.AddMvc();
66     services.AddDbContext<DontGetSpicyContext>(options =>
67     {

public void Configure(IApplicationBuilder app, IWebHostEnvironment env)
{
    if (env.IsDevelopment()) ...
    app.UseHttpsRedirection();
    app.UseRouting(); JwtSecurityTokenHandler.DefaultInboundClaimTypeMap.Clear();
    app.UseAuthentication();
    app.UseCors("Cors");
    app.UseAuthorization();

    app.UseEndpoints(endpoints =>
    {
        endpoints.MapControllers();
        endpoints.MapHub<ChatHub>("/ChatHub");
        endpoints.MapHub<GameHub>("/GameHub");
    });
}
```

Hub komponente u ovom projektu su:

```
namespace DontGetSpicy.SignalR
{
    3 references
    public class GameHub:Hub
    {
        0 references
        public async Task JoinGameGroup(string gameGUID,string username, string boja)
        {
            await Groups.AddToGroupAsync(Context.ConnectionId, gameGUID);
            await Clients.GroupExcept(gameGUID,Context.ConnectionId).SendAsync("userJoined",username,boja);
        }
    }
}
```

```
namespace DontGetSpicy.SignalR
{
    1 reference
    public class ChatHub:Hub
    {
        0 references
        public async Task StartChat(string gameGUID)
        {
            await Groups.AddToGroupAsync(Context.ConnectionId, gameGUID);
        }
        0 references
        public async Task SendMessage(string gameGUID, string message, string sender)
        {
            await Clients.GroupExcept(gameGUID,Context.ConnectionId).SendAsync("userSentMessage", message, sender);
        }
    }
}
```

Da bi se uspešno koristio SignalR i u okviru Vue.js fajlova potrebno izvršiti instalaciju:

```
npm i @microsoft/signalr
```

Zatim se u .vue fajlu importuje sama biblioteka:

```
<script>
import {HubConnectionBuilder} from '@microsoft/signalr'
import Igra from "../components/Igra"
```

Pa se može koristiti kao u sledećem primeru:

```

mounted()
{
  this.pristiglePoruke.push("Access code is: "+this.accessCode);
  connection2.start().then(()=>
  {
    connection2.invoke("StartChat",this.guid);
  })

  connection2.on("userSentMessage",(message, sender) =>{
    this.pristiglePoruke.push(sender+" : "+message)
  })

  this.igra=new Igra();
  this.$data[this.mojaBoja]=this.username;
  connection.start().then(()=>
  {
    connection.invoke("JoinGameGroup",this.guid, this.username,this.mojaBoja);
  })

  connection.on("userJoined",(username,b) =>{
    this.$data[b]=username;
    if(this.sviPrisutni()==true)
      this.started=true;
  })

  connection.on("krajIgre",(pobednik) =>{
    alert("pobednik je: "+pobednik);
    setTimeout(()=>
    {
      router.push({ name: 'Home' })
    },10000).catch(err =>console.log(err));
  });
}

```

```

connection.on("kockaBacena",(vrKocke, next) =>{
  this.kocka=vrKocke;
  if(next==true)
  {
    this.naPotezu=Next[this.naPotezu]; return;
  }
  if(this.naPotezu==this.mojaBoja)
  {
    alert("izaberite figuru");
  }
})

connection.on("figuraPomerena",(potezi, next) =>{
  if(next==true)
    this.naPotezu=Next[this.naPotezu];

  this.igra.odigranjPotez((staraPozicija:potezi[0].item1,novaPozicija:potezi[0].item2), (potezi[1]==undefined)?null:{staraPozicija:potezi[1].item1,novaPozicija:potezi[1].item2})
})

if(this.igrac1==null)
{
  this.igrac1.forEach((igrac,index) => {
    this.$data[Boja.naziv[index]]=igrac;
  })
}

```