

## Lab 5 - Socket.IO quiz

Uppgiften är att bygga ett frågespel i form av en webserver med express som levererar frågor via socket.io och kontrollerar de inkomna svaren för att ge spelarna poäng.

Själva frågorna (och svaren) ska servern själv hämta från [Open Trivia DB](#), via deras API. Detta löser ni enklast genom att använda er av [axios](#) som är en http-klient för node. Ex.

```
const response = await axios.get('https://opentdb.com/api.php?amount=5&type=multiple');
```

Använd er endast av 'multiple choice'-frågorna. (Fyra alternativ; ett korrekt svar). Frågan presenteras för spelarna tillsammans med 4 knappar för de olika alternativen. Se till att "blanda" alternativen *innan* de skickas till klienten så man inte kan fuska. Kolla gärna om ni kan hitta något lämpligt paket på [npmjs](#) som sorterar en array i slumpmässig ordning.

**Ni kan välja mellan två olika utförande av uppgiften:** En för godkänt-nivå, där den första klienten som ansluter sig mot servern blir spelare och får svara på frågorna, medan övriga (som ansluter efterhand) blir åskådare som endast kan se frågorna och spelarens svar. Den andra versionen är för VG-nivå, och där kan valfritt antal personer ansluta till servern och spela mot varandra.

Båda versionerna bör ha en klientdel bestående av en html- och en js-fil som presenterar frågorna och spelets gränssnitt i webbläsaren. Båda bör också ha en serverdel som tillhandahåller filerna till klienten, samt skickar frågorna och sköter logiken.

## **G-uppgiften (en spelar, resten tittar på)**

Den första klienten som ansluter till servern utses till spelare och spelet börjar direkt med att servern skickar "fråga 1" med fyra svarsalternativ (vilket alternativ som är korrekt ska inte skickas till klienten). När spelaren klickat på ett alternativ skickas svaret tillbaka till servern som jämför med det korrekta svaret. Därefter skickas direkt "fråga 2" till klienten och så upprepas det hela tills spelaren stänger sin webbläsare.

Övriga klienter som ansluter ska se en "logg" där de kan följa frågorna och spelarens svar. Loggen behöver inte sparas i helhet på servern (så att de som ansluter efterhand kan se tidigare frågor), utan det räcker att ni skickar aktuell fråga till åskådarna samtidigt som den skickas till spelaren, samt att ni skickar spelarens svar (när spelaren svarat). Ni ska även visa om spelaren är 'connected' eller 'disconnected'. Om spelaren stänger sin webbläsare blir han alltså 'disconnected'. I så fall ska nästa person som ansluter till servern bli ny spelare, och spelet startar om med en ny fråga 1. (Då ska även åskådarnas loggar tömmas).

Exempel på hur webbsidan kan se ut för åskådare:

### **Player connected!**

**Q1: According to the International System of Units, how many bytes are 1 kilobyte of RAM?**

Player answer: 1000 (Correct)

**Q2: Who created the 2011 Video Game "Minecraft"?**

Player answer: Jens Bergensten (Wrong)

**Q3: In the TV show "Cheers", Sam Malone was a former relief pitcher for which baseball team?**

*I exemplet har spelaren ännu inte svarat på fråga 3.*

För att skicka olika data till spelare och åskådare föreslår jag att ni skapar olika rum så att spelaren t.ex "joinar" rummet 'player', medan övriga klienter "joinar" rummet 'spectators'.

## **VG-uppgiften (flera spelar mot varandra)**

När servern startar påbörjar den en nedräkning 3, 2, 1 ... och skickar då ett meddelande i sekunden till alla klienter om hur långt tid det är kvar innan spelet startar. När de 3 sekunderna gått kollar servern hur många klienter som är anslutna och låter alla dessa "joina" ett rum vi kallar 'playRoom'. Klienter som ansluter *efter* detta "joinar" istället 'lounge' och får vänta där tills nästa omgång startar. Varje klient som ansluter ska få ett unikt namn t.ex player A, player B etc.

Varje spelomgång består av 3 frågor, och man har 5 sekunder på sig att svara på varje fråga. Därefter följer en paus på 5 sekunder där resultat visas, innan nästa omgång startar. Servern går alltså i en loop om totalt 20 sekunder, och alla som är i 'lounge' får ett meddelande varje sekund som säger att nästa spel startar om X antal sekunder. (Och när X sekunder gått flyttas alla klienter övre till 'playRoom' och spelet startar om).

När spelet startar så hämtas tre frågor från Open Trivia DB. Den första frågan skickas till alla klienter i rum 'playRoom'. Spelaren ska då se att detta är fråga 1 av 3, samt se de fyra alternativen. Klickar man på ett alternativ så skickas detta till servern som jämför med det korrekta svaret och internt håller reda på alla spelares poäng. Så fort man svarat ska servern skicka ut det korrekta svaret till klienten och detta ska markeras för spelaren (t.ex genom att rätt "knapp" blir grön och de andra röda). Servern måste också ha koll på om en spelare redan svarat på en fråga så man inte kan peppra på alla svarsalternativ och få poäng.

När 5 sekunder gått så presenteras nästa fråga för spelarna. (Har man inte hunnit svara får man inget poäng). Efter 3 frågor \* 5 sekunder så visas en lista över alla spelares poäng samtidigt som spelarna får veta att nästa omgång börjar om 5, 4, 3, 2, 1 ...

Det ska framgå i resultatet som presenteras för spelaren vem "man själv" är. Övriga presenteras med namnet som de fick när de först anslöt till servern. Det skulle alltså kunna se ut så här när man är klar:

## **Next game starts in 5 seconds**

### **Result from last game:**

Player B: 3 points

You: 1 point

Player C: 0 points

## Redovisning

Uppgiften utförs enskilt eller i grupper om 2-4 personer. För redovisning lägger ni upp ert projekt på github och länkar dit. Det ska finnas en beskrivning (readme.md) som beskriver vad jag behöver göra för att kunna testa/köra projektet. Alla i gruppen lämnar in på ithsdistans genom kommentar vilka ni jobbat med och länk till github.

## Betygskriterier

För godkänt krävs:

- Spelaren får en fråga i taget med fyra svarsalternativ.
- Övriga klienter kan följa spelet genom en logg där de ser frågor och svar.
- Åskådarna kan se om det finns en spelare ansluten eller ej.
- När spelaren stänger sin anslutning, blir nästa person som ansluter ny spelare.
- När en ny spelare kommer in töms åskådarnas loggar och börjar på nytt.

För väl godkänt krävs istället:

- Var 20:e sekund startar en ny spelomgång.
- Alla klienter som är anslutna då en ny omgång påbörjas är med i spelet.
- Alla klienter som ansluter medan en omgång pågår ombeds vänta till nästa omgång.
- Klienter som väntar ska se hur lång tid det är kvar tills nästa omgång börjar.
- Varje omgång består av 3 frågor. 5 Sekunder på varje fråga.
- Det korrekta svaret skickas till (och visas för) spelaren först efter att spelaren svarat.
- När omgången är slutet visas resultat för spelarna under 5 sekunder.