

Spindelkonferensen 2018

Tävling – The Fappquiz

Beskrivning

Uppgiften till Torna Championship in Programming (TCP) kommer vara ett frågesportsspel. Ni ska i backend göra nödvändiga förberedelser för att sedan kunna hämta ner frågor till appen och visa dem. Tanken är sedan att när användaren väljer ett alternativ på frågan ska en request skickas från appen till servern som kommer svara om det valda alternativet är rätt eller fel. Om det är rätt ska servern svara med nästa fråga, som då ska visas. Är det fel alternativ ska svaret vara ett error och då ska användaren börja om från fråga ett. Mer detaljerad specifikation på uppgiften finner ni nedan till både rails och appen.

Rails

1. Skapa databastabellen `game_questions` för spelet, och tabellen `game_highscore` för highscoren. Varje fråga ska ha tre svarsalternativ.
2. Skapa motsvarande `models` för Rails. Tänk på valdieringar och associationer.
3. Migrera databasen och skapa några testfrågor genom konsollen.
4. Gör en `serializer` för frågorna till quizet. Tänk på att det rätta svaret inte får skickas till appen, för då kan folk fuska!
5. Skapa en `API controller` för spelet. Man ska kunna starta ett nytt spel, och servern ska då skicka den första frågan. Controllern ska också kunna ta emot ett svar på en fråga och kolla om det är rätt. I så fall ska den skicka nästa fråga. Annars skickas ett felmeddelande. Skapa även logik för spelets slut, dvs då den sista frågan besvaras korrekt.
6. Fixa `routes` och kolla att de fungerar. *Hint: Använd programmet Postman.*
7. Ordna så att kontrollern uppdaterar highscoren när en användare antingen har klarat spelet eller svarar fel på en fråga.
8. Fixa en controller och serializer för highscoren.

Appen

1. Skapa en ny HTML, CSS, JS fil för den nya sidan och länka dem korrekt i `index.html`.
2. Skapa navigation till den nya sidan i den femte tabben och fånga `init`-eventet för sidan.
3. Testa att hämta data från API-endpointen och skapa en template. Testa även så att templatens `compilas`, kan köras och läggas in i en container. Notera att templatens inte behöver innehålla den hämtade datan än, den ska endast kunna användas korrekt.
4. Konfiguera templatens så den genererar frågestrukturen till sidan (tre knappar och en rubrik med frågan) med den hämtade datan.
5. Skapa ett `onClick`-event för knapparna med alternativ och definiera en `post`-request för det valda alternativet.



6. Hantera svaret från servern:

Användaren svarade rätt: ladda nästa fråga, vilket ska ligga i responsen.

Användaren svarade fel: reseta sidan till första frågan. Responsen ska innehålla en statevariabel som säger att man har förlorat.

7. Definiera logik för då sista frågan blir besvarad korrekt. Responsen ska innehålla en statevariabel som säger att man har vunnit.

8. Styla sidan, så den ser fresh af ut.

Jakob Navrozidis
Konferensmästare 2018

Fredrik Lastow
Konferensmästare 2018

Olof Englund
Konferensmästare 2018

