

Webbutveckling semifinaluppgift

Filerna som behövs för att utföra uppgiften är tillgängliga på GitHub:

<https://github.com/Taitaja2025/semifinal>

Teknisk information gällande servern, databaserna och produktionsmiljöns URL-adresser finns i ett särskilt dokument. Varje tävlande har en egen URL och användar-ID och lösenord till servern. Bedömningen genomförs på den givna URL-adressen, och kodens kvalitet granskas från den tävlandes Git-repository.

Allmän beskrivning av uppgiften

Målet i semifinalen är att förverkliga ett fungerande och användarvänligt frågespel, där lärarna kan hantera frågor och de studerande kan testa sitt kunskapsområde på olika ämnesområden. Appen ska innehålla två huvudfunktionaliteter: en spelvy för de studerande och en hanteringspanel för lärarna. Spelet riktas till alla åldersgrupper och utbildningsnivåer, och den ska därför vara mångsidig och betjäna olika slags användargrupper.

Detta frågespel är den första versionen av produktionen, varför lösningen ska vara tydlig, välorganiserad och öppen för utveckling i framtiden. Användarupplevelsen ska vara flexibel och visuellt tilltalande samt fungera både för lärarnas och de studerandes behov. Allt arbete bedöms utifrån en fungerande webbplats, som publiceras på den individuella URL-adress som getts åt dig. Noggrannare tekniska instruktioner finns i tävlingens tekniska dokument.

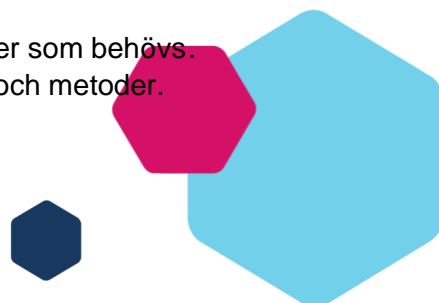
Projektets tekniska krav

I förverkligandet av appen kan den tävlande använda endera nativlösningar (vanilla HTML, CSS, JavaScript och PHP) eller moderna JavaScript-frameworks, såsom React, Vue.js eller Angular. Backend-förverkligandet ska göras med PHP, och i det kan man använda nativlösningar eller Laravel-framework. Det viktigaste målet med alla lösningar är att appen fungerar felfritt på den givna URL-adressen. Förverkligandet ska också följa god programmeringspraxis och uppfylla uppdragets krav. Grafiska instruktioner och en designskiss finns i **assets**-mappen.

Användargränssnittets stilar och komponenter ska förverkligas utan färdiga CSS-ramar eller UI-bibliotek (t.ex. Bootstrap, Tailwind, Material UI). Alla stilar och gränssnittskomponenter ska man själv planera och förverkliga.

Det är tillåtet att använda AI, men om man använder AI ska man meddela ifrågavarande punkter i koden genom att kommentera. Den slutliga lösningen ska ändå vara producerad av den tävlande själv och den ska vara originell.

Koden ska vara tydlig, välorganiserad och innehålla sådan kommentarer som behövs. För granskning av kodens kvalitet används en kombination av verktyg och metoder.



Förstasida

URL: /

Syftet med förstasidan är att vara startpunkt för användaren, och där ska de viktigaste funktionerna lätt kunna hittas. På framsidan finns en "Pelaa nyt" -knapp, som styr användaren direkt till spelets början. I början väljer den studerande den lärare, vars frågor hen vill besvara. Dessutom innehåller förstasidan en kort beskrivning av spelets funktioner samt lärarnas inloggningslänk. I sidfoten, i footer, visas uppgifter om upphovsman och läroanstaltens namn.

Förstasidan ska ha en modern, tydlig layout som ska vara tillgänglig på alla apparater. Grafiska instruktioner och en designskiss finns i **assets**-mappen.

Obligatoriska element på förstasidan:

- **Header:** Spelets logo och länk till "Kirjaudu sisään".
- **Pelaa nyt -delen:** En kort beskrivning ("Valitse opettaja ja aihealue, ja aloita tietosi testaaminen heti!") och en knapp, som styr till spelets startvy.
- **Miten pelataan -delen:** Tydliga anvisningar för spelets funktioner (kategorier, lärare, antal frågor, resultat).
- **Footer:** Uppgifter om upphovsman (För- och efternamn | Läroanstaltens namn) och Mästare 2025 -semifinaluppgift ska nämnas.

Spelvyn

URL: /game

Spelet startar när användaren förflyttar sig från förstasidan till spelvyn genom att trycka på "Pelaa nyt" -knappen. Användaren ska göra två val: först lärare och sedan ämnesområde (kategori).

Antalet frågor kan vara kort (5 frågor), medellångt (10 frågor) eller långt (15 frågor). Det förvalda antalet är 10 frågor. Varje fråga har fyra svarsalternativ, av vilka ett är rätt. Rätt svar ger en poäng.

Under spelets gång får användaren i realtid respons på varje svar. För ett rätt svar visas en uppmuntrande animering, och vid fel svar visas det rätta svaret visuellt betonat. Din uppgift är att planera animeringarna som stöder spelaren och passar målgruppen. Spelet ska visa spelarens poängtal och uppgifter om hur långt man kommit (t.ex. 2/10 frågor besvarade).

I slutet av spelet frågas det efter användarens namn, och resultaten laddas ner i databasen. Resultaten visas på en high score -lista, som organiseras enligt poäng och svårighetsgrad. Till sist visas knappen "Pelaa uudestaan" för användaren.

Förstasidan och spelvyn ska vara mobilresponsiv, för spelet riktas till en bred krets av användare, som kan använda olika slags huvudapparater.

Lärarens hanteringspanel

URL: /admin

I hanteringspanelen används en färdigt given databas, med två förvalda lärare och färdigt skapade kategorier samt frågor. Den tävlande ska använda denna databas och vid behov lägga till blad som behövs för att förverkliga uppgiften. Alla data som används i appen hämtas från databasen och laddas ner i den. Hanteringspanelen är optimerad för skrivbordsvyn och den behöver inte fungera i mobilen.

Hanteringspanelen har skyddats med lösenord, och användaren ska logga in före hen kommer åt att hantera frågor och kategorier. Vid förverkligandet av inloggningslogiken ska det säkerställas att tillträde till hanteringspanelen utan inloggning är spärrat.

Inloggningsvy

URL: /login

Inloggningsvyn är enkel och avsedd för användning av lärarens hanteringspanel. Vyn ska innehålla fält för användar-ID och lösenord samt inloggningsknapp. Användar-ID och lösenord har förvalts färdigt för läraren:

användar-ID: Olevi Opettaja

lösenord: Taitaja2025!

Efter inloggning flyttar användaren över till hanteringspanelen. Om inloggningen misslyckas, ska ett tydligt felmeddelande visas för användaren.

I hanteringspanelen kan läraren:

- Skapa nya kategorier eller välja de som finns.
- Granska frågorna i den valda kategorin och lägga till nya frågor.
- Bearbeta frågor som finns och ta bort dem.
- Skriva nya frågor och svarsalternativ samt ange det rätta svaret.

Om läraren väljer en färdig kategori, visar programmet de frågor och svar som redan finns i den och erbjuder möjlighet till att lägga till nya frågor. Om läraren skapar en ny kategori, kan man genast lägga till frågor i den. När läraren lägger till en ny fråga, ska hen skriva frågan och ange vilket av svarsalternativen som är rätt.

Resultat från alla kategorier ska synas i hanteringspanelen. Resultaten kan organiseras endera enligt poäng eller datum. Om det finns mer än 10 resultat, ska vyn erbjuda sidnumrering (pagination).

Alla tävlande ska använda den givna databasen och spara de färdiga uppgifterna i den. De färdiga kategorierna och lärarnas uppgifter stöder spelets och hanteringspanelens funktioner samt uppgiftens bedömning.

Projekthantering och dokumentation

Tävlingsarbetets viktigaste mål är att få alla huvudfunktionaliteter att fungera inom den givna tidtabellen och publicerade som en webbplats på den individuella URL-adress som getts åt dig. Lösningen ska innehålla en heltäckande dokumentation, som stöder uppgiftens bedömning och visar förverkligandets tydlighet. Arbetet ska vara snyggt, välkommenterat, och alla funktionaliteter ska fungera felfritt. Alla granskningar av koden görs via den tävlandes Git-repository. Noggrannare tekniska detaljer finns i tävlingens tekniska dokument.

I GitHub-repository ska man lägga till en heltäckande README.md-fil, där du beskriver projektets syfte, struktur och användningsinstruktioner. Förklara hur spelet spelas och hur hanteringspanelen används, och erbjud anvisningar till domarna för testning och granskning, medräknat inloggningsuppgifter och databasförbindelser.

Koden ska vara tydligt strukturerad och välkommenterad. Kommentarer ska läggas till särskilt för invecklade punkter, så att arbetets logik lätt kan förstås. PHP-koden ska följa god programmeringssed, såsom modularitet, återanvändbarhet och dataskydd. HTML- och CSS-koden ska vara semantiskt korrekt och valideringsfel ska vara korrigerade. Dessutom ska koden vara optimerad för tillgänglighet och prestandaförmåga.

Bedömning av uppgiften

I bedömningen av uppgiften betonas särskilt appens funktionalitet, kodens kvalitet, användargränssnittets användarvänlighet samt dokumentationens omfattning. Lösningen ska vara flexibel och svara mot både lärarnas och spelarnas behov, och samtidigt vara lätt att förstå och passa för vidareutveckling. Appens alla huvudfunktioner ska fungera felfritt och fylla uppgiftens krav. PHP-, HTML-, CSS- och JavaScript-koden ska vara tydlig, strukturerad och kunna valideras utan fel. Användargränssnittet ska vara tillgängligt, visuellt tilltalande och lätt att använda för alla användargrupper. För dokumentationens del ska README.md-filen erbjuda en heltäckande beskrivning av projektet, dess struktur och funktionaliteter, och den ska innehålla tydliga anvisningar till domarna till stöd för testning och bedömning. Det är tillåtet att använda AI, men de delar som AI producerat ska tydligt markeras med kodkommentarer, och den slutliga lösningen ska vara självständigt skapad av den tävlande och originell.