Reparatieopdracht HTML/CSS/JS

# Opdracht

In deze opdracht ga je een aantal websites maken met behulp van HTML en CSS. Met behulp van Javascript maak je een speelbare quiz. Je werkt aan de hand van reeds gemaakte wireframes en user stories. Binnen het design van de wireframe ben je vrij om kleuren, lettertypen, etc. te kiezen.

Belangrijk: Dit is jouw kans om ons te overtuigen van het feit dat je HTML/CSS en Javascript *begrijpt* en *kunt*. Deze opdracht is (als het goed is) zo geschreven dat je hem niet helemaal af gaat krijgen in drie dagen; aan jouw de opdracht om zover mogelijk te komen door slim user stories te kiezen en gedurende de reparatieweek door te werken. Aangezien 100% niet mogelijk is, is ‘doorwerken’ belangrijker dan ‘afmaken’.

# Stappen

1. Neem de wireframes en user stories door; stel direct vragen als er dingen onduidelijk zijn;
2. Maak een repo aan op github en maak direct een scrumboard

## Herhalend

1. Kies user stories waar je aan gaat werken; plaats deze op je scrumboard. Houd hierbij rekening met:
   1. Hoeveel waarde levert deze user story op voor de klant? Je kunt bijvoorbeeld beginnen met het bedenken van 100 quizvragen, maar daar heeft je klant niets aan, zolang de quiz niet werkt. Hetzelfde geldt voor de achievements, power-ups, etc.
   2. Wat is een logische volgorde? Je kunt beginnen met het aanmaken van alle pagina’s (de quiz zelf plus de pagina’s voor regels, over en credits), maar aangezien ze alle vier hetzelfde design krijgen, is het waarschijnlijk efficiënter als je er eerst één helemaal afmaakt; deze kun je dan later gebruiker voor de andere drie pagina’s;
   3. Waarschijnlijk heb je te weinig tijd om alle user stories in te bouwen (daar is de opdracht op gemaakt); je MOET dus slim kiezen;
   4. Kies niet meer user stories dan je in één dagdeel kunt afronden. Als je eerder klaar bent, kun je altijd een extra user story oppakken, maar als je er teveel kiest, loop je het risico dat je niet weet waar je moet beginnen;
2. Werk deze user stories uit naar concrete taken op je scrumboard;
3. Pak één voor één de taken op;
4. Houd je scrum-board bij, zodat je altijd weet welke taken klaar zijn en welke nog niet;
5. Commit en push na ten minste iedere taak;
6. Het kan zijn dat je docent niet direct beschikbaar is als je vragen hebt of bent vastgelopen; probeer in dat geval een andere taak op te pakken, terwijl je wacht tot je docent beschikbaar is;
7. Lees ‘Beoordeling en feedback’ om er zeker van te zijn dat je alles correct hebt afgerond
8. Als je alle taken/US die je hebt gekozen hebt afgerond, ga je terug naar stap 3.

# Inleveren

* Push je werk naar github om het in te leveren.

# Beoordeling en feedback

Je wordt beoordeeld op de volgende punten

* Kennis en toepassing van HTML & CSS
* Kennis en toepassing van Javascript
* Werkhouding gedurende de reparatieweek
* Aantal afgeronde user stories
* Conventies & code kwaliteit (incl. comments)

# Wireframes

# 

# 

# User stories

|  |
| --- |
| Als docent wil ik een website waarop mijn studenten een quiz over computerwetenschappen kunnen spelen, zodat zij hun kennis over dit onderdeel kunnen testen; |
| Als docent wil ik links naar handige sites over computerwetenschappen en programmeren in de footer, zodat mijn studenten eenvoudig meer over deze onderwerpen kunnen leren |
| Als gebruiker wil ik een duidelijke ‘Game over’ of ‘Gewonnen’-boodschap zien als het spel voorbij is, zodat ik weet wanneer het spel voorbij is |
| Als gebruiker wil ik dat de vragen in willekeurige volgorde worden gesteld, zodat ik nooit weet wat de volgende vraag wordt |
| Als gebruiker wil ik dat de quiz uit tenminste 100 vragen bestaat, zodat ik niet in ieder spel dezelfde vragen krijg |
| Als gebruiker wil ik vanaf het ‘Game over’- en ‘Gewonnen’-scherm direct een nieuw spel kunnen starten, zodat ik direct opnieuw kan beginnen als ik niet tevreden ben over mijn score |
| Als gebruiker wil ik na de feedback op mijn antwoord na 5sec of als ik op de knop "Volgende vraag" druk door naar de volgende vraag, zodat het spel niet stilvalt |
| Als gebruiker wil ik een nieuw spel kunnen starten, zodat ik opnieuw kan beginnen als ik teveel fouten heb gemaakt |
| Als gebruiker wil ik mijn voortgang (reeds beantwoorde vragen + vragen goed) kunnen opslaan en laden, zodat ik op een later moment verder kan (local storage) |
| Als gebruiker wil ik de beschikking hebben over drie power-ups die ik gedurende het spel kan inzetten, zodat het spel dynamischer wordt. Ik kan een power-up inzetten nadat ik de vraag heb gelezen, maar voordat ik beantwoord: Eén power-up verdubbelt mijn score bij een goed antwoord, één power-up haalt alle foute antwoorden op één na weg (50/50), en één power-up hersteld het aantal levens als ik de vraag correct beantwoord |
| Als gebruiker wil ik zien hoeveel 'levens' ik nog heb, zodat ik weet hoeveel foute antwoorden ik kan geven voordat het spel is afgelopen |
| Als docent wil ik handige links met betrekking tot Tech nieuws in de footer, zodat mijn studenten zich eenvoudig kunnen informeren over belangrijk nieuws in onze sector |
| Als gebruiker wil ik zien hoeveel vragen ik heb beantwoord, hoeveel vragen ik goed heb beantwoord, en mijn plaats binnen de highscores in het ‘Game over’ en ‘Gewonnen’-scherm, zodat ik direct kan zien hoe veel ik weet in vergelijking met anderen |
| Als gebruiker wil ik dat er een maximum tijd is voor het beantwoorden van een vraag, zodat de vaart er lekker in blijft |
| Als gebruiker wil ik dat ik het spel gewonnen heb als ik alle vragen in het spel heb beantwoord, zodat het spel niet oneindig door gaat |
| Als gebruiker wil ik de spelregels kunnen lezen, zodat ik weet wat ik moet doen |
| Als gebruiker wil ik kunnen kiezen tussen een spel met timer en zonder, zodat ik zonder stress kan oefenen voordat ik de echte quiz ga spelen |
| Als gebruiker wil ik dat mijn score (het aantal vragen goed) wordt getoond, zodat ik kan laten zien hoeveel ik over het onderwerp weet |
| Als gebruiker wil ik dat een vraag per spel maar één keer gesteld wordt, zodat ik geen punten kan halen voor vragen die ik eerder gehad heb |
| Als gebruiker wil ik de credits kunnen zien, zodat ik weet wie het spel gemaakt heeft |
| Als gebruiker wil ik dat het spel voorbij is als ik drie vragen fout heb beantwoord, zodat ik niet kan blijven gokken tot ik alle vragen goed heb |
| Als gebruiker wil ik dat het juiste antwoord oplicht nadat ik gekozen heb, zodat ik weet of ik het goed had en wat het juiste antwoord was als ik het niet goed had |
| Als gebruiker wil ik op pc een mooie website met veel ruimte, op een tablet (breedte < 750px) dezelfde pagina maar met minder tussenruimte en op mobiel (breedte < 500px) een compacte website, zodat ik ten minste de vraag en de antwoorden altijd zichtbaar zijn |
| Als gebruiker wil ik dat één gebruikte power-up hersteld wordt als ik 10 vragen op een rij goed heb beantwoord, zodat hele goede spelers meer kans hebben op een hele goede score |
| Als gebruiker wil ik de highscores kunnen zien, zodat ik weet hoeveel vragen anderen goed hadden (local storage) |
| Als gebruiker wil ik wil ik de vraag zien, met daaronder de (klikbare) antwoorden, zodat ik het juiste antwoord aan kan klikken |
| Als gebruiker wil ik meer algemene informatie over het spel (“In opdracht van ROC Mondriaan”, etc.), zodat ik meer over de achtergrond van het spel weet |
| Als gebruiker wil ik kunnen kiezen tussen dark- en lightmode van de website, zodat ik de site kan aanpassen aan de hoeveelheid licht in mijn omgeving |
| Als gebruiker wil ik achievements kunnen verdienen (buiten de score om), zodat het spel aantrekkelijk blijft. Achievements: 3/5/10/15/25 vragen op rij goed, 25 vragen zonder power-ups te gebruiken, game over met nul vragen goed, alle vragen correct beantwoord |
| Als gebruiker wil ik zien hoeveel vragen de quiz in totaal heeft en hoeveel ik al beantwoord heb, zodat ik weet hoeveel punten ik nog kan halen |