|  |  |
| --- | --- |
| **Datum** | 23-06-2025 |
| **Klas** | SOD 2F |
| **Groepsamenstelling** | Husseen |

## Proef examen opdracht

## Algemeen

De opdracht duurt in totaal 3 weken, **de deadline** om het project in te leveren is 28 juni

Het project moet ingeleverd worden via GitHub en de laatste commit moet **28 juni zijn.**

In dit document ga je samen met je groepsgenoten vaststellen wat jullie opdracht gaat zijn om alle competenties aan te tonen die bij deze opleiding horen.

Je moet laten zien dat je kan:

* Documenteren
* Ontwerpen
* Programmeren
* Testen
* Reflecteren
* Samenwerken

Iedereen moet bewijsstukken laten zien van bovenstaande onderwerpen, het is niet toegestaan dat een persoon alleen het ontwerpen doet en de ander alleen het programmeren.   
  
Bij de presentatie zal naar ieders aandeel worden gevraagd van alle onderwerpen. Hoewel het een gezamenlijke presentatie is zal de beoordeling op individueel niveau zijn.

## Opdracht

Dit is een vrije opdracht, je bent zelf vrij om de taal, framework en onderwerp te kiezen. Om ervoor te zorgen dat iedereen wel qua moeilijkheidsgraad een gelijkwaardig project heeft vul je dit document zorgvuldig in en wordt deze vervolgens goed/afgekeurd door je docent.

## Beschrijving project

|  |  |
| --- | --- |
| **Beschrijving van het project (Wat ga je maken)** | **Akkoord** |
| * Algemene beschrijving wat je gaat maken * Hier noteer je nog niet de gekozen technieken   Een gewichts tracker waarin je een grafiek hebt waarin je je progressie kan zien |  |
| **Beschrijving van de technieken (Hoe ga je het maken)** | **Akkoord** |
| * Benoem hier specifiek welke technieken je gaat gebruiken * Back-end framework bijv. Welke front-end technieken   VueJS en Tailwind CSS |  |
| **Beschrijving van de software die je gaat gebruiken** | **Akkoord** |
| * Denk aan een IDE, Dev omgeving, versie van PHP, maar ook andere relevante software om je ontwerpen te maken   Visual Studio Code |  |
| **Waar sla je de documentatie op?** | **Akkoord** |
| * Verantwoord hier hoe jullie effectief samen gaan werken en veilig je documenten bewaren   Ik werk alleen dus ik houd het in de github map |  |
| **Beschrijf waar je de planning op gaat slaan** | **Akkoord** |
| * Hoe geef je prioriteiten aan * Hoe werk je samen * Hoe is de structuur * Gebruiker kan gewicht invoeren via invoerveld * Gegevens worden opgeslagen in localStorage (dus zonder back-end) * Gewicht wordt visueel weergegeven in een grafiek * Mogelijkheid om laatste 5 of 10 gewichten te tonen * Reset-knop om data te wissen   Ik werk alleen maar ik ga alles bij houden in trello  Stuctuur is Backlog – To Do – Doing - Review - Done |  |
| **Hoe bewijs je de voortgang tijdens het project** | **Akkoord** |
| * Laat hier zien hoe jullie aantonen dat er elke week netjes volgens de planning wordt gewerkt. |  |
| **Beschrijf welke schematechnieken en ontwerptechnieken je gaat gebruiken (use-case diagram, klasse diagram, activiteiten diagram, flowchart, erd, wireframe, sitemap) en benoem wie verantwoordelijk is voor wat.** | **Akkoord** |
| * Welke schematechnieken * Welke ontwerptechnieken * Wie maakt wat   Wireframe  Usecase Diagram  Klassen Diagram   |  |  |  | | --- | --- | --- | | **Wireframe** | **UI-ontwerp (User Interface Design)** | Laat zien hoe de interface eruitziet – knoppen, invoervelden, layout. Helpt bij het visueel ontwerpen vóór het coderen. |  |  |  |  | | --- | --- | --- | | **Use-case diagram** | **Functioneel ontwerp** | Laat zien wat de gebruiker kan doen in het systeem. Focus ligt op functies en interacties. |  |  |  |  | | --- | --- | --- | | **Klassen diagram** | **Technisch ontwerp / OOP ontwerp** | Laat zien hoe je code is opgebouwd met klassen, attributen en methodes. Wordt gebruikt om de structuur van de applicatie te ontwerpen. | |  |
| **Welke code conventions ga je hanteren** | **Akkoord** |
| * Welke afspraken maak je met je team om een uniforme code base te garanderen * camelCase voor variabelen en functies * PascalCase voor klassen * 2 spaties inspringing * Altijd semicolons ; * Dubbele aanhalingstekens (") * Netjes commentaar bij belangrijke logica * ref, computed, methods, watch in logische volgorde   **Vue-conventies:**   * <script setup> gebruiken * Data boven, functies onder * Grafiek met Chart.js, styling met TailwindCSS   **Solo-project, maar teamregels toegepast:**   * Consistente code * Overzichtelijke structuur * Klaar voor uitbreiding met backend |  |
| **Beschrijf hier hoe je gaat bewijzen dat je versiebeheer correct hebt toegepast** | **Akkoord** |
| * Wat voor versiebeheer gebruik je * Hoe ga je het gebruiken   Github Desktop  Na het afronden van het taak ga ik het committen en dan pushen. |  |
| **Welke technieken ga je toepassen om te bewijzen dat je op niveau zit? (OOP, Functioneel programmeren)** | **Akkoord** |
| * Beschrijf hier welke techniek je gaat toepassen   OOP |  |
| **Hoe ga je de userstories testen?** | **Akkoord** |
| * Welke technieken ga je gebruiken? * Hoe is de aanpak?   Ik ga alles handmatig testen |  |
| **Hoe maken jullie het testrapport** | **Akkoord** |
| * Hoe rapporteren jullie de bevindingen uit te tests * **Structuur testrapport:** titel, inleiding, testomgeving, testmethoden, testresultaten, conclusie, aanbevelingen * **Bevindingen rapporteren:** per test stap beschrijven wat is gedaan, resultaat, status (geslaagd/mislukt) en eventuele opmerkingen * **Gebruik:** duidelijke taal, overzichtelijke tabellen, eventueel screenshots * **Doel:** inzicht geven in welke onderdelen goed werken en waar verbeteringen nodig zijn |  |
| **Hoe gaan jullie om met het overleg?** | **Akkoord** |
| * Hoe vaak? * Welke afspraken maken jullie? * Waar bewaren jullie de notulen? |  |
| **Hoe bewijzen jullie dat je hebt gereflecteerd op je werk?** | **Akkoord** |
| * Waar noteer je het? * Hoe bewijs je het?   Klein verslagje maken |  |

|  |  |
| --- | --- |
| **Naam** | **Naam** |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |