Kerntaken

- Plannen c24
- Ontwerpen c24
- Testen C24
- Ontwerpen C22
- Testen C22

Plannen c24

Waarom?

Inleiding

ICT, is veel meer dan coderen. Als ICT'er maak je documentatie, ontwerpen, planningen en ga zo maar door. Voor het examen moeten jullie ook een aantal van deze zaken leren. Eén van deze zaken is het kunnen maken van een planning.

Wat, hoe en waarom plannen?

Stel jij wilt een nieuwe keuken laten bouwen en de aannemer zegt;

"*we maken een nieuwe keuken, de prijs weet ik pas als ik klaar ben en ik denk dat ik over vier weken klaar ben, maar dat weet ik nog niet helemaal zeker want ik weet niet wat en wie ik precies allemaal nodig zal hebben.*".

Ga jij hiermee akkoord? Waarschijnlijk niet, je wilt eerst weten wat het kost en wanneer het klaar is en dat staat allemaal in een planning.

Er zijn heel veel redenen waarom je een planning nodig hebt.

1. Je wilt weten wanneer je klaar bent, zodat de klant of opdrachtgever weet wanneer hij wat kan verwachten.
2. Als je een project uitvoert met meerdere personen dan wil je weten wanneer je deze personen nodig hebt.
3. Je wilt weten hoeveel tijd (en daarmee geld) het kost om iets te maken. De klant, of jou baas wil dat *van tevoren* weten.
4. Bij projecten kunnen zich problemen voordoen. Soms kun je daar alvast van tevoren over nadenken, dan ben je voorbereid mochten deze problemen zich voortdoen.
5. Via een planning kun je iedereen op de hoogte brengen van het project en kun je tijdens het project bepalen of je nog op schema bent. Je kunt op deze manier snel escaleren (met je baas gaan praten dat er mogelijk tegenslagen zijn).
6. Omdat je jezelf dwingt via een planning van tevoren na te denken over allerlei zaken zal dit de kwaliteit ten goede komen.

7. Door precies te omschrijven wat je gaat doen geef je eigenlijk ook aan wat je niet gaat doen. Dit voorkomt dat er dingen van jou worden verwacht waar je niet op heb gerekend.

In deze module gaan we stap-voor-stap leren hoe je een planning moet maken.

Opdracht

Bedenk een situatie waarin jij geen planning had, maar waarbij het achteraf gezien wel handig was geweest om die wel te hebben. Omschrijf dit kort in je eigen woorden.

Stap 1, SMART

De eerste stap is het bepalen van je doel. Dat doe je zo nauwkeurig mogelijk en daarbij gebruik je zo min mogelijk algemene termen.

Als je een doel beschrijft dan houd je rekening met SMART:

Specifiek, Meetbaar, hAalbaar, Relevant, Tijdsgebonden.

SMART komt uit het Engels en staat eigenlijk voor Specific, Measurable, Achievable, Relevant en Time Restricted

Als je niet precies weet wat SMART is, zoek het dan nog eens op.

Oefening

Bij elke doel dat hieronder beschreven is, geef jij aan aan welke letters het voldoet en waarom wel of juist niet., bijvoorbeeld.

Ik wil een webpagina-aanpassingen in systeem XYZ waardoor de sortering van alle studenten in het overzicht worden gesorteerd op achternaam in plaats van op voornaam. Het moet vrijdag af zijn.

Specifiek	Zeker er wordt heel duidelijk precies verteld wat er gedaan moet worden.
Meetbaar	Ja hoor, het is makkelijk te controleren of het vrijdag af is.
Haalbaar	Ik denk het wel, het lijkt niet zo moeilijk
Relevant	Er staat niet waarom dit moet, dit is dus niet aangetoond.
Tijdsgebonden	Ja/nee er staat wel vrijdag maar niet welke vrijdag, kan dus beter.

Nu jij.....(en er staan nog een uitgebreider voorbeeld onderaan deze opdracht).

Opdracht

Student: <vul hier e naam in>

case 1

Kan je iets maken in de Canvas Monitor zodat studenten niet meer kunnen kopiëren van elkaar? Kan het over twee maanden af zijn. Het is belangrijk omdat studenten steeds meer werk kopiëren van elkaar en daar niets van leren. De studenten lopen dan later vast als ze op stage moten of als ze examen moeten doen.

Specifiek	
Meetbaar	
Haalbaar	
Relevant	
Tijdsgebonden	

case 2

Ik wil een uitbreiding op de Canvas Monitor waardoor studenten in hun overzicht kunnen zien of ze op schema lopen. Dit moet automatisch worden berekend. Het zou handig zijn als we dit met ingang van volgend schooljaar kunnen gaan gebruiken, zodat we onze nieuwe 1ste jaars daar vanaf het begin mee kunnen leren omgaan. Iedereen die dan meer dan twee weken achter loopt moet dan dinsdag verplicht naar school komen (ipv thuiswerken).

Specifiek	
Meetbaar	
Haalbaar	
Relevant	
Tijdsgebonden	

case 3

De website moet een stuk strakker, het ziet er nu niet uit. Ik vind het belangrijk dat dit over twee dagen is geregeld omdat de klant dan komt kijken hoe het met het project staat. Als hij dit ziet dan schrikt hij zich helemaal rot!

Specifiek	

Meetbaar	
Haalbaar	
Relevant	
Tijdsgebonden	

case 4

Ontwikkel een webapplicatie om de administratie van klantinteracties te verbeteren voor een klein verkoopbedrijf. Het project moet binnen zes maanden voltooid zijn. Het project moet een vermindering van 20% in de tijd die nodig is voor het invoeren en verwerken van klantgegevens opleveren. De applicatie moet gebruiksvriendelijk zijn en integreren met de bestaande CRM-software.

Specifiek	
Meetbaar	
Haalbaar	
Relevant	
Tijdsgebonden	

Inleveren

Maak een PDF-document, kopieer de vier opdrachten erin, bepaal van alle vier de cases of deze SMART zijn.

Zet je naam in het PDF document.

Voorbeeld

Ontwikkel een geautomatiseerd facturatiesysteem voor een middelgroot productiebedrijf om de verwerking van inkomende en uitgaande facturen te versnellen. Het systeem moet binnen een maand geïmplementeerd zijn en moet leiden tot een 50% snellere facturatieverwerking dan het huidige systeem. Het projectteam zal bestaan uit vier softwareontwikkelaars. De applicatie moet ook gebruiksvriendelijk zijn voor niet-technische medewerkers van de financiële afdeling. Zonder dit systeem zal er een extra medewerker moeten worden aangenomen om al de facturaties op tijd te verwerken.

Specifiek	Ja, want er wordt genoemd dat er een geautomatiseerd facturatiesysteem moet komen en nee omdat er wordt gezegd gebruikersvriendelijk" en dat wordt niet gespecificeerd. Verder wordt het systeem niet echt gespecificeerd; hoeveel facturen moeten worden verwerkt en wat moet er precies met de facturen gebeuren?
Meetbaar	Ja, want er wordt gezegd dat de applicatie moet zorgen voor van de vermindering van de facturatieverwerkingstijd met 50%.
Haalbaar	Onduidelijk want de eisen zijn onvoldoende gespecificeerd; waar moet het systeem precies aan voldoen?
Relevant	Ja, want zonder deze software moet er een extra medewerker worden aangenomen en dat kost per jaar 2 minimaal 34 000 euro.
Tijdsgebonden	Ja, er wordt gevraagd om het project in een maand af te ronden, het is echter onduidelijk waarom precies. Wat gaat er mis als het project 2 of 3 maanden kost?

Stap 2a, eisen

In examens worden veel fouten gemaakt omdat zaken te algemeen worden opgeschreven.

Voorbeeld:

De website moet er strak uitzien , of

de website moet veilig zijn , of

de website moet een geode performance laten zien

Voor al deze uitspraken geldt dat er discussie kan ontstaan, over bijvoorbeeld wat er strak uit ziet (dat kan immers heel persoonlijk zijn) of wat is veilig, of wat is goede performance? Hoe dan?

"*De website moet er strak uitzien*"

Dit moet je specifiek maken bijvoorbeeld.

- De website mag maximaal drie kleuren hebben, één hoofdkleur, een steunkleur en één accent kleur.
- Het menu moet op ale pagina's hetzelfde zijn en er hetzlfde uitzien.
- Er mag maar één font-family worden gebruikt, "Roboto".
- De achtergrondkleur van de website is $f8f8f8 (bijna wit).
  Enzovoorts.....

Zie je dat je op deze manier minder ruimte voor discussie over laat? Nu hoeven deze zaken niet allemaal in de planning te worden opgenomen, maar je moet wel voorkomen dat je algemene termen als 'strak', 'veilig', 'mooi' en 'snel' gebruikt. In de planning houd je het op hoofdlijnen en werk je deze later uit in het ontwerp.

Dus je kan ook zeggen: de website ziet er uit volgens de styleguide die onderdeel van het ontwerp is. Dit is dan gelijk een activiteit die je moet plannen, het maken van een styleguide kost immers tijd.

Opdracht

Maak een de uitspraak " *De website moet passen bij de frisse uitstraling van verse groenten* " specifiek.

Bedenk zelf iets, zie het als een voorstel aan de klant. De klant heeft jou gevraagd om een website met een *frisse uitstraling van verse groenten,* jij maakt dit nu concreet en doet een voorstel aan de klant. Beschrijf dit en maak ook een heel eenvoudig voorbeeld (plaatje).

Inleveren

PDF met specifieke omschrijving met voorbeeld plaatje.

Zet ook je naam in de PDF.

Stap 2b eisen

Opdracht

Beschrijf de opdracht die je vorig jaar hebt gemaakt voor de CRUD Challenge.
Beschrijf wat je doel is, wat ga je maken en wat zijn alle eisen?

Noem ten minste 10 concrete eisen.

image-1687948362350.png

Ken je de opdracht niet (meer)? Ga dan naar de volgende link en bekijk wat er moet worden gebouwd.

https://stampwerk.nl

Inleveren

TXT-bestand met ten minste 10 ***specifieke*** eisen in eigen woorden.

Zet je naam in de PDF.

Kennis-check 1

Voer de kennis-check uit tijdens de les. Vraag de docent om de juisten kennis-check open te zetten.

Inleveren

Een screendump van je resultaat en het resultaat moet 60% of hoger zijn.

Kennis-check "Specifiek"

Als je goed begrijpt wat specifiek is voer dan de kennis-check 'Specifiek' uit tijdens de les. Vraag de docent om de juisten kennis-check open te zetten.

Inleveren

Een screendump van je resultaat en het resultaat moet 80% of hoger zijn .

Stap 3, opdelen in stappen

Welke stappen moet je allemaal nemen om het project af te ronden?

Om te kunnen bepalen hoeveel tijd iets kost en wanneer het klaar is, helpt het om je project in stappen op te delen.

Laten we een voorbeeld nemen.

Opdracht

Stel je moet een CRUD maken voor, -laten we zeggen een te-laat-kom-system. Dus je wilt bijhouden van alle werknemers hoe laat ze binnenkomen 's morgens. Het systeem moet responsief zijn en op een mobiel device draaien. Je gaat dit in Yii maken. Houd er rekening mee dat in Yii een CRUD systeem vrij snel kan genereren (dus zonder code te tikken).

Je moet het invoeren van de te laat meldingen zo eenvoudig mogelijk maken en gebruik maken van drop downs om in zo min mogelijk aantal handelingen een te laat melding te maken.

Wat schat je hoeveel tijd dat kost. Denk hier even over na en probeer in te schatten hoeveel tijd dit kost. Heb je een aantal uren in je hoofd, schrijf dit op en ga dan verder.

Laten we nu het volgende eens bekijken.

1. Overleg met opdracht gever wat er precies moet worden vastgelegd in de database. Als uitgangspunten nemen we de velden genoemd bij 7 en 8.
2. Eventueel aanpassen planning naar aanleiding van gesprek.
3. database opzetten met phpMyAdmin voor het te laat kom systeem met de 2 tabellen te laat voor de tijden en een tabel waarin alle werknemers staan. Velden staan bij 7 en 8.
4. Installeren nieuw Yii project.
5. Maken model (2x) met de Yii model-generator.
6. Maken CRUD (2x) met de Yii model-generator.
7. Aanpassen 4 CRUD schermen voor te laat meldingen met werknemer_id (FK), datum, aantal minuten te laat en reden.
8. Aanpassen 4 CRUD schermen voor het registreren van medewerkers met medewerker_id (PK), naam, telefoon, datum in dienst.
9. Te laat meldingenscherm aanpassen voor eenvoudige invoer (dropdowns en dergelijke) om vastleggen te laat melding zo snel mogelijk te kunnen doen.
10. Testen op laptop.
11. Testen op mobiel.
12. Laten zien aan opdracht gever.
13. Laatste wijzigingen maken en/of fouten oplossen.

Zet nu bij elk van deze taken een aantal uren, tel deze op en kijk hoeveel je afwijkt van de eerste schatting?

Vraag : wat is je conclusie?

Inleveren

Beschrijf het aantal uren dat je eerst inschatte (dus toen je nog niet alle stapjes had gelezen) en het aantal uren dat je inschatte toen je alle stapjes had gelezen.

En beschrijf wat jouw conclusie is.

Stap 4, specifieke stappen

Stappen in de planning zijn specifiek of niet.

Niet specifieke stappen gelden kunnen gelden voor veel (ICT) projecten. Je zou bij wijze van spreke een standaar planning kunnen maken met de volgende stappen:

- overleg opdracht gever over opdracht
- opzetten development omgeving
- maken planning
- overleg opdracht gever over planning
- maken ontwerp
- overleg opdracht gever over ontwerp
- opzetten database

Enzovoorts.

Zie je dat al deze stappen voor elk project zouden kunnen gelden? Of je nu een video-registratie systeem of een te-laat melding systeem maakt, deze stappen zouden voor beiden van toepassing kunnen zijn.

Wat zijn dan specifiek stappen?

Voorbeelden

Opzetten database	Niet specifiek
Opzetten database met de tabellen inkoop_order, factuur, klant en order_boek.	Specifiek, geldt niet voor een ander project.
Maken loginscherm	Niet specifiek
Maak login dat is gekoppeld aan Canvas waarbij gebruikers dan kunnen aanloggen met hun schoolaccount.	Specifiek

Opdracht

Bepaal van elk van deze stappen of ze specifiek of niet specifiek zijn.

Stap	Omschrijving	Specifiek?	Toelichting

1	Overleg met opdracht gever wat er precies moet worden vastgelegd in de database. Als uitgangspunten nemen we de velden genoemd bij 7 en 8.	ja/nee	
2	Eventueel aanpassen planning naar aanleiding van gesprek.	ja/nee	
3	database opzetten met phpMyAdmin voor het te laat kom systeem met de 2 tabellen te laat voor de tijden en een tabel waarin alle werknemers staan. Velden staan bij 7 en 8.	ja/nee	
4	Installeren nieuw Yii project.	ja/nee	
5	Maken model (2x) met de Yii model-generator.	ja/nee	
6	Maken CRUD (2x) met de Yii model-generator.	ja/nee	
7	Aanpassen 4 CRUD schermen voor te laat meldingen met werknemer_id (FK), datum, aantal minuten te laat en reden.	ja/nee	
8	Aanpassen 4 CRUD schermen voor het registreren van medewerkers met medewerker_id (PK), naam, telefoon, datum in dienst.	ja/nee	
9	Te laat meldingenscherm aanpassen voor eenvoudige invoer (dropdowns en dergelijke) om vastleggen te laat melding zo snel mogelijk te kunnen doen.	ja/nee	
10	Testen op laptop.	ja/nee	
11	Testen op mobiel.	ja/nee	
12	Laten zien aan opdrachtgever.	ja/nee	
13	Laatste wijzigingen maken en/of fouten oplossen.	ja/nee	

Neem de tabel over en vul de laatste kolommen in. Gebruik eigen woorden en gebruik geen ChatGPT of (andere) AI!

Inleveren

PDF met de complete tabel en plaats jouw naam boven aan in de PDF.

Stap 5, uren en prioriteit

We gaan onze planning uitbreiden. De uren hadden we ingeschat bij de vorige stap.

We gaan deze aanvullen met de *datum af* en *prioriteit* .

Datum af

Dit is de deadline voor de stap. Op deze datum is de stap dus af.

Als je per stap aangeeft wanneer het klaar moet zijn, dan kun je tijdens het project controleren of je nog op schema ligt of dat je voor- of achter loopt.

Prioriteit

De prioriteiten spreken we van tevoren af zodat we weten wat we per se moeten afmaken of wat we mogelijk later zouden kunnen opleveren als we in tijdnood komen.

Prioriteiten

We kiezen een eenvoudig systeem om prioriteiten te stellen:

1. Absoluut noodzakelijk
2. Erg fijn om te hebben
3. Niet strikt noodzakelijk

Opdracht

Neem de tabel over en vul de laatste vier kolommen in.

Uren

Bij uren, neem je de uren over die je in de vorige opdracht ook al had bedacht.

Datum af

Je begin te plannen vanaf volgende week en je hebt 4 uur per dag beschikbaar om aan dit project te werken. Zet dan ij de datum wanneer de stap klaar moet zijn.

Prioriteit

Vul de prioriteit in 1, 2 of 3. Hierboven staat wat 1,2 en 3 betekenen.

Toelichting

Leg uit waarom je deze stap prioriteit 1,2 of 4 hebt gegeven.

Stap	Omschrijving	Uren	Datum af	Prioriteit	Toelichting
1	Overleg met opdracht gever wat er precies moet worden vastgelegd in de database. Als uitgangspunten nemen we de velden genoemd bij 7 en 8.				
2	Eventueel aanpassen planning naar aanleiding van gesprek.				
3	database opzetten met phpMyAdmin voor het te laat kom systeem met de 2 tabellen te laat voor de tijden en een tabel waarin alle werknemers staan. Velden staan bij 7 en 8.				
4	Installeren nieuw Yii project.				
5	Maken model (2x) met de Yii model-generator.				

6	Maken CRUD (2x) met de Yii model-generator.				
7	Aanpassen 4 CRUD schermen voor te laat meldingen met werknemer_id (FK), datum, aantal minuten te laat en reden.				
8	Aanpassen 4 CRUD schermen voor het registreren van medewerkers met medewerker_id (PK), naam, telefoon, datum in dienst.				
9	Te laat meldingenscherm aanpassen voor eenvoudige invoer (dropdowns en dergelijke) om vastleggen te laat melding zo snel mogelijk te kunnen doen.				
10	Testen op laptop.				
11	Testen op mobiel.				
12	Laten zien aan opdrachtgever.				
13	Laatste wijzigingen maken en/of fouten oplossen.				
	TOTAAL				

Inleveren

Plaats je naam in de PDF.

PDF met aangevulde tabel. Vergeet niet je toelichting in eigen woorden toe te voegen.

En vergeet niet het totaal aantal uren (onderaan) te vermelden.

Kennis-check 2

Voer de kennis-check uit tijdens de les. Vraag de docent om de juisten kennis-check open te zetten.

Inleveren

Een screendump van je resultaat en het resultaat moet 60% of hoger zijn.

Event Tracker

Opdracht

Maak een planning voor onderstaand project

Project: Event Tracker

Ontwikkel een eenvoudige webapplicatie genaamd Event Tracker, die speciaal is ontworpen voor studenten om belangrijke school- en vrijetijdsevenementen bij te houden. Deze applicatie stelt gebruikers in staat evenementen zoals schoolproject, deadlines, sportwedstrijden, en sociale bijeenkomsten te creëren en te beheren.

Doel

Het hoofddoel is om een eenvoudige tool te bieden waar studenten hun school- en persoonlijke evenementen efficiënt kunnen beheren, wat hen helpt bij het organiseren van hun tijd en verantwoordelijkheden.

Functionaliteiten van de Webapplicatie

- - **Evenementenbeheer:** Gebruikers kunnen evenementen aanmaken, bekijken, bewerken en verwijderen.
  - **Evenementenoverzicht:** Een eenvoudige kalenderweergave waarop alle aankomende evenementen worden getoond.
  - **Gebruikersinteractie:** Gebruikers kunnen details van evenementen invoeren zoals datum, tijd, en een korte beschrijving.

Technische Specificaties

- - **Talen:** PHP voor de backend logica, JavaScript voor de client-side dynamica, en HTML/CSS voor de frontend presentatie.
  - **Database:** MySQL met één enkele tabel:
    - `Events` met kolommen voor evenement-ID, gebruikers-ID (voor identificatie van wie het evenement heeft aangemaakt), titel, beschrijving, datum, tijd en eventuele extra opmerkingen.
  - **Veiligheid:** Basis sessiemanagement om de interacties van gebruikers te beheren.

Design

Het design zal helder en gebruiksvriendelijk zijn, met een moderne look die aansluit bij de visuele voorkeuren van tieners. De interface zal ook responsief zijn om goed te werken op zowel smartphones als computers.

Maak een planning, maak minimaal 12 stappen en maak ten minste 4 specifieke stappen .

Stap	Omschrijving	Specifiek	Uren	Prioriteit
1				
2				
3				

4				
5				
6				
7				
8				
9				
10				
11				
12				
13				
14				
15				
16				
17				
18				
	TOTAAL			

Inleveren

PDF met planning. Zet je naam in het pdf-document.

Gamebuddy

Opdracht

Maak een planning voor onderstaand project

Project: GameBuddy

Doel

Ontwikkel een eenvoudige webapplicatie, GameBuddy, die fungeert als een platform voor jonge gamers, specifiek gericht op jongeren. Het platform dient als een basisnetwerk voor gamers om verbinding te maken, spelervaringen te delen, en speelschema's te organiseren.
Het hoofddoel is het creëren van een interactief en gebruiksvriendelijk platform waar jonge gamers gemakkelijk vrienden kunnen maken, hun speeltijden kunnen delen, en hun gaming ervaringen kunnen coördineren. Dit project streeft ernaar om een toegankelijke en functionele online gemeenschap te bouwen voor jonge gamers.

Functionaliteiten van de Webapplicatie

- **Gebruikersprofielen:** Gebruikers kunnen profielen aanmaken waarin ze hun favoriete spellen en gaming tijden vermelden.
- **Vriendschapsnetwerk:** Functionaliteit om vrienden toe te voegen en een lijst van vrienden te onderhouden.
- **Speelschema's:** Een eenvoudige kalender waar gebruikers hun geplande speeltijden kunnen posten en bekijken.

Technische Specificaties

- **Talen:** PHP voor de backend logica, JavaScript voor dynamische gebruikersinteracties, en HTML/CSS voor de frontend.
- **Database:** MySQL met drie tabellen:
- `Users` voor het opslaan van gebruikersgegevens zoals gebruikersnaam, e-mail, en wachtwoord.
- `Friends` voor het beheren van de vriendenlijst, met verwijzingen naar gebruikers-ID's.
- `Schedules` voor het opslaan van gebruikers geplande speeltijden.
- **Veiligheid:** Basis gebruikersauthenticatie voor het inloggen en registreren van gebruikers.

Design

Het design zal strak en modern zijn, aantrekkelijk voor jongeren, met een eenvoudige en heldere gebruikersinterface. Het zal ook responsief zijn om toegankelijk te zijn via zowel desktops als mobiele apparaten. Indien er tijd over is, kan een light- en dark theme worden gemaakt.

Maak een planning, maak minimaal 12 stappen en maak ten minste 6 specifieke stappen .

Stap	Omschrijving	Specifiek	Uren	Prioriteit
1				

2				
3				
4				
5				
6				
7				
8				
9				
10				
11				
12				
13				
14				
15				
16				
17				
18				
	TOTAAL			

Inleveren

PDF met planning en vergeet niet je naam in het document te zetten.

Voortgang

Tijdens de uitvoering van het project moeten we aan de hand van de planning de voortgang bewaken.

Je kunt daarvoor deze template gebruiken.

Projectlog-ls.docx

De bedoeling is dat je elke dag of om de paar dagen controleert of je voortgang in overeenstemming met de planning is.

Opdracht

Doe alsof je met het project bent begonnen.

1. Gebruik de template, maak deze leeg en zet je naam er in.
2. Vul de eerste 5 dagen van het project de status in.
3. Op dag 4 heb je onverwacht bijna niet kunnen werken.
4. Op dag 5 heb je de opgelopen achterstand weer ingehaald door extra hard te werken.

Inleveren

PDF met ingevulde projectlog.

Plaats je naam bovenaan in de projectlog.

Optioneel - QuizMaster

Deze opdracht is niet verplicht, maar als je te weinig punten hebt, dan kun je hier extra punten mee halen!

Opdracht

Maak een planning voor onderstaand project

Opdracht Titel : Ontwikkeling van een Online Quiz Applicatie - "QuizMaster"

Projectbeschrijving : Het doel van dit project is het ontwikkelen van een webgebaseerde applicatie, genaamd "QuizMaster", die docenten in staat stelt om aangepaste quizzes te maken en studenten deze te laten beantwoorden. De applicatie moet functioneel vergelijkbaar zijn met de Socrative-app, maar met enkele toegevoegde kenmerken.

Belangrijkste functies en eisen :

1. **Gebruikersaccounts** : Er moeten twee typen gebruikersaccounts zijn: docenten en studenten. Docenten kunnen quizzes maken en resultaten bekijken. Studenten kunnen quizzes beantwoorden.
2. **Quiz Creatie** : Docenten moeten de mogelijkheid hebben om quizzes te maken en te beheren. Ze moeten in staat zijn om vragen toe te voegen, te bewerken en te verwijderen, evenals om quizzes op te slaan voor later gebruik. Elke vraag moet een multiple choice vraag zijn met één correct antwoord en tot vier verkeerde antwoorden.
3. **Categorieën** : Docenten moeten in staat zijn om vragen in verschillende categorieën in te delen. Elke quiz kan een of meer categorieën bevatten.
4. **Quiz Beantwoorden** : Studenten moeten in staat zijn om in te loggen, een quiz te selecteren en de quiz te beantwoorden. Ze moeten hun naam invoeren voor het begin van de quiz.
5. **Resultaten** : Na het voltooien van de quiz moeten studenten hun scores kunnen zien. Docenten moeten ook in staat zijn om de scores van elke student te zien en te downloaden.
6. **UI/UX** : De applicatie moet een strak en modern ontwerp hebben. Het moet responsive zijn om goed te werken op zowel desktops als mobiele apparaten.
7. **API:** via een API-interface kunnen gegevens vanuit de applicaties worden gelezen. Do kunnen de quiz vragen via de API worden geëxporteerd. Eventueel (indien niet te veel werk) zou het ook handig zijn als quiz vragen via de API kunnen worden geüpdate.

Technische specificaties : De applicatie moet worden ontwikkeld met behulp van het Laravel PHP framework. MariaDB moet worden gebruikt als de database.

Deliverables :

1. Volledige broncode van de applicatie.
2. Gedocumenteerde API en databasestructuur.
3. Gebruikershandleiding voor zowel studenten als docenten.
4. Technische documentatie, inclusief installatie-instructies en systeemvereisten.
5. Testplan en testrapport.

Tijdlijn : Het project moet binnen 12 weken voltooid zijn.

Budget : Het budget voor het project wordt nog bepaald, maar het aantal te besteden uren moet tussen de 200 en 450 uur uitkomen.
Let op: deze beschrijving is een startpunt. Het zal waarschijnlijk nodig zijn om verder te verfijnen en meer details toe te voegen op basis van gesprekken met de ontwikkelaar(s) en andere stakeholders.

Bepaal alle stappen/taken, en zet deze onder elkaar in Excel. Bepaal de uren per taak. Tel de uren op en bepaal per taak de prioriteit.

Maak een planning, maak minimaal 18 stappen en maak ten minste 8 specifieke stappen .

Stap	Omschrijving	Specifiek	Uren	Prioriteit
1				
2				
3				
4				
5				
6				
7				
8				
9				
10				
11				
12				
13				
14				
15				
16				
17				
18				
	TOTAAL			

Inleveren

PDF met planning en vergeet niet je naam in het document te zetten.

Ontwerpen c24

01 De Webwinkel algemeen

Inleiding, we gaan een ontwerpt maken voor een webwinkel. We doen dit om stapje-voor-stapje te oefenen met het maken van een ontwerp.

Bedenk voor jezelf een webshop die je zou kunnen maken.

Laten we beginnen met wat algemene zaken.

Algemene vragen:

1. Wat ga je in de webwinkel verkopen?
2. Hoe moet de lay-out van de productpagina eruitzien?
3. Zijn er specifieke ontwerpvereisten of stijlen die moeten worden gevolgd?
4. Wat zijn de beveiligings- en privacyvereisten?
5. Hoe wordt klantinformatie beschermd, vooral bij het schrijven van beoordelingen?
6. Hoe moet de navigatie op de productpagina werken?
7. Zijn er links naar gerelateerde producten, categorieën, of recent bekeken items?

Opdracht

In je hoofd heb je een webwinkel die jij wilt maken.

Kopieer de vragen naar MS Word, en geef zo goed mogelijk antwoord op alle vragen.

Inleveren

PDF met antwoord op alle vragen.

02 Productpagina

De beschrijving die we voor de productpagina hebben is:

- **Productpagina**: Elke productpagina moet details en klantbeoordelingen tonen.

Dit is niet specifiek. Om een ontwerp te maken moeten we dit specifiek maken.

Vragen die je zou kunnen stellen zijn:

Voor productdetails:

1. Hoe moet de lay-out van de productpagina eruitzien?
2. Welke productinformatie moet worden weergegeven?
3. Hoeveel productfoto's moeten er getoond worden en hoe groot zijn de foto's?
4. Hoe worden klantbeoordelingen weergegeven?
5. How worden de beoordelingen gesorteerd. Wat komt bovenaan te staan?
6. Wat moet er in de tekst van een beoordeling staan?
7. Moeten beoordelingen gesorteerd kunnen worden, zo ja, hoe?
8. Op welke manieren kunnen beoordelingen gesorteerd worden?
9. Kan iedereen een beoordeling schrijven of alleen geregistreerde klanten?
10. Zijn er vereisten voor het controleren of iemand het product daadwerkelijk heeft gekocht??
11. Moeten er ook foto’s of video's in beoordelingen kunnen staan?
12. Wat zijn de beveiligings- en privacyvereisten?
13. Hoe wordt klantinformatie beschermd, vooral bij het schrijven van beoordelingen?
14. Hoe moet de navigatie op de productpagina werken?

Opdracht

Ontwerp (in je hoofd) een ontwerp en geef antwoord op alle gestelde vragen.

Wees zo concreet mogelijk, dus let op dat je dit soort woorden ***niet*** gebruikt.

> **groot, groter, snel, sneller, slim, slimmer, mooi, mooier, efficiënt, efficiënter, geavanceerd, geavanceerder, aantrekkelijk, aantrekkelijker, gebruikersvriendelijk, gebruikersvriendelijker.**
>
> meer info over concreet

Inleveren

PDF met antwoord op alle vragen, 1 tot en met 14.

03 Wire frame

Een wire-frame is een voorbeeldtekening van hoe iets eruit gaat zien, zoals een website of een app. Het helpt om te zien en te bespreken hoe het eindresultaat eruit zal zien voordat je het echt gaat maken.

Voorbeelden van tools, die je gratis kan gebruiken (per 2024):

- Figma
- Sketch
- Adobe XD
- InVision
- Mockplus
- Balsamiq
- Canva
- Gravit Designer

Kies zelf een tool.

Powerpoint

Deze wireframe is gemaakt met powerpoint.

In dit voorbeeld (hieronder) zie je een website waar je CV's kan plaatsen en waar werkgevers stage plaatsen kunnen aanbeiden.

image-1717163619893.png

Figma

Onderstaand voorbeeld van een product pagina is gemaakt met Figma. De wire-frame is nog geen detail ontwerp waarin alles precies is uitgewerkt, maar het geeft op hoofdlijnen een beeld van wat je gaat maken.

Voorbeeld in Figma

image-1717162834195.png

Omschrijving

In dit ontwerp wordt de naam van de webshop bovenaan de pagina getoond. Rechtsboven is het navigatiemenu. Je kunt daarmee naar andere onderdelen van de website navigeren.

Op de productpage wordt het product genoemd met daaronder een beschrijving. Naast de beschrijving worden twee plaatjes van het product getoomd. Onder de productomschrijving staat een knop om het product naar mijn Winkelmandje te verplaatsen.

Opdracht

Teken een wire-frame van de productpagina in een tool naar keuze. Je kunt de wireframe die hierboven staat als uitgangspunt nemem, maar denk wel aan punt 4 (zie hieronder).

Zorg dat de volgende punten minimaal in het ontwerp staan getekend.

1. Het menu.
2. De banner.
3. Productomschrijving, afbeelding en bestelknop.
4. Verder laat je op de pagina een paar product reviews zien. Je kunt vanuit hier ook via een knop navigeren naar alle product reviews.

Inleveren

PDF met jouw persoonlijke mock-up van een product pagina, met daaronder een korte beschrijving van de pagina.

04 Onderbouwen ontwerp

Als je een webpagina ontwerpt dan moet je rekening houden met een aantal 'good practices' dat zijn zaken die over het algemeen als goed worden gezien.

- bezoekers bekijken de pagina van boven naar beneden en dan van rechts naar links. Dus belangrijke zaken zet je links van minder belangrijke zaken.
- navigatie (hoe ga je naar andere pagina's) is belangrijk en staat meestal bovenaan.
- op een pagina moet niet te veel informatie staan, omdat gebruikers dat te verwarrend vinden.

- informatie moet niet te dicht op elkaar staan, omdat anders de leesbaarheid niet goed is.
- letter moeten goed leesbaar zijn, dus gebruik een goed contrast tussen voor- en achtergrond. Dus geen blauwe letters op een zwarte achtergrond.
- te veel kleuren maken een site al snel te druk en daardoor slechter leesbaar.

Je hebt bij de vorige opdracht een wire-frame gemaakt.

Misschien heb je er niet zo over nagedacht, maar waarom heb je het gemaakt zoals je het gemaakt hebt?

1. Waarom staan de plaatjes links en de tekst rechts?
2. Waarom staat de bestelknop aan de linkerkant?
3. Waarom staat er een hamburger menu en geen 'gewoon' menu?
4. Waarom staan er twee plaatjes en geen 3 of 4?

Jij hebt misschien een ander ontwerp gemaakt. Of misschien als je over deze vragen nadenkt wil je je ontwerp nog aanpassen?

Opdracht

Neem je Wire Frame van de vorige opdracht. Als je wilt mag je hem nog aanpasen.

Onderbouw waarom je het ontwerp zo maakt.

Voorbeelden:

- Er worden twee plaatje getoond omdat .....
- Het menu staat bovenaan omdat .....
- Aan de rechterkant van het scherm zie je de ..... omdat .....
- Als kleuren worden ..... en ..... gebruikt omdat .....

Inleveren

1. Wire Frame
2. PDF met daarin onderbouwing van het ontwerp (minimaal 4 punten).

05 Responsive

Responsive design betekent dat een website of applicatie zich automatisch aanpast aan verschillende schermgroottes en apparaten, zoals mobiele telefoons, tablets en computers. Het doel is om de inhoud altijd goed leesbaar en bruikbaar te maken, ongeacht op welk apparaat je de site bekijkt.

Hier zijn de belangrijkste punten:

1. **Aanpassing aan schermgrootte**: De website past zich aan aan de breedte en hoogte van het scherm. Dit gebeurt automatisch, zodat je niet hoeft te scrollen of in te zoomen om alles te kunnen zien.
2. **Flexibele lay-out**: De elementen op de website, zoals tekst, afbeeldingen en knoppen, verschuiven en schalen mee met de schermgrootte. Dit zorgt ervoor dat de inhoud er altijd goed uitziet en goed werkt.

Als je een CSS-framework als Bootstrap of Tailwind gebruikt, is dit meestal allemaal al geregeld voor je.

Voorbeeld van responisve design, links zie je een formulier zoals het op een PC/Laptop te zien is en rechts zoals je het op een mobiele telefoon ziet.

https://www.youtube.com/embed/oNSX21dh6kw?si=CiHCq6PuDlGpjj0E

Opdracht

Neem je Wireframe als voorbeeld en omschrijf op welke punten jouw website voldoet aan 'responsive design'. Noem minimaal drie punten.

Inleveren

PDF met een beschrijving van minimaal 3 punten waar je rekening mee gaat houden ten aanzien van een repsonsive design.

06 Site map

Een site map van een website is als een plattegrond voor een gebouw. Het laat zien hoe alle pagina's van de website met elkaar verbonden zijn en waar je alles kunt vinden.

Wat is een site map?

- **Visuele gids**: Het is een diagram of lijst die de structuur van de website weergeeft. Het toont alle hoofdpagina's en subpagina's.
- **Navigatiehulp**: Het laat zien hoe je van de ene pagina naar de andere kunt komen.

Waarom is een site map belangrijk?

- Een site map kan helpen in de **communicatie** met je opdrachtgever/klant.
- Een site map kan helpen bij de **planning**; je vergeet minder makkelijk een pagina.

Opdracht

Bedenk een site map voor een eenvoudige web winkel. Je kunt producten opzoeken, details bekijken, producten in een winkelmandje plaatsen, winkelmandje bekijken en afrekenen/bestellen. Daarnaast staat er ergens een contactpagina.

Maak een site map voor deze website.

Inleveren

PDF, met site map, je kunt diverse tools gebruiken, bijvoorbeeld Powerpoint, Word of Lucichart.

07 RBA tabel

Een RBA-tabel (Role-Based Access tabel) is een lijst die bepaalt welke personen binnen een organisatie toegang hebben tot welke functies of gegevens, gebaseerd op hun rol.

Hoe het werkt:

1. **Gebruikers:** De mensen die het systeem gebruiken.
2. **Rollen:** Groepen met bepaalde rechten, zoals 'beheerder', 'manager', of 'medewerker'.
3. **Toegangsrechten:** Wat iemand met een bepaalde rol mag doen, zoals gegevens bekijken, bewerken, of verwijderen.

In een RBA-tabel staat welke gebruikers welke rol hebben en wat ze met die rol mogen doen. Dit maakt het makkelijk om rechten te beheren en de veiligheid te waarborgen.

Voorbeeld: stel je hebt een webshop en je wilt aangeven wie welke rechten krijgt op een *bestelling*.

Je ziet bijvoorbeeld dat iemand van de administratie een bestelling niet mag aanpassen (update) maar wel kan lezen.

Bestelling RBA tabel

Rol	Create	Read	Update	Delete
klant ingelogd	ja	ja	ja	ja
klant niet ingelogd	?	?	?	?
administrator	nee	ja	nee	nee
hoofd administratie	?	?	?	?

Opdracht

Ga uit van de bovenstaande RBA-tabel.

De opdrachtgever heeft aangegeven dat in enkele gevallen een klant met de administratie belt als er wat is misgegaan. De medewerker van de administratie moet dan de mogelijkheid hebben om een bestelling te verwijderen.

In enkele uitzonderlijke gevallen moet soms een *Bestelling* worden aangepast. Om fraude te voorkomen mag niet iedereen dat van de administratie zomaar een bestelling aanpassen. Dat mag allleen het hoofd van de administratie. Er moet dus een extra rol komen en deze rol mag een bestelling lezen of aanpassen.

Pas de bestaande RBO tabel aan:

1. lees de bovenstaande text en vul de 'hoofd administratie' regel in (vervang de ?).
2. bedenk zelf wat er zou staan op de regel 'klant niet ingelogd', dit is dus een bezoeker zonder account (vervang de ?).

Inleveren

Aangepaste RBA tabel in PDF.

08 Security / veiligheid

Op het nternet zijn veel hackers actief. Om je web-site veilig te houden moeten we met veel zaken rekning houden. Een paar belangrijke zijn:

Controleer invoer

Door 'vreemde' invoer te geven zou je dode iets onvoorspelbaars kunnen doen. Stel de prijs wordt als volgt berekend:

```
$afrekenprijs = $aantal * prijs_per_stuk;
```

Stel je koopt een iPhone van 895,= euro en bij aantal vul je -1 in. Wat zou er dan kunnen gebeuren?

Juist je moet -895,= euro betalen, je krijgt dus geld gestort op je rekening!

Hoe voorkom je dit?

Dit kan op front-end en op de back-end. De front-end is gebruikersvriendelijker (sneller), maar is ook makkelijker te hacker (weet jij hoe?). Dus voor de veiligheid checken we de invoer op de back-end én voor de gebruikersvriendelijkheid op de front-end!

> Controleer zoveel mogelijk alle invoer op de front-end én op de back-end!

Geef *geen* duidelijk foutmeldingen

Als je ontwikkeld wil je zo goed mogelijk de foutmelding krijgen. Dat helpt bij het zoeken naar fouten.

Maar als hacker heb je ook veel aan foutmeldingen. Je ziet al snel welke framework is gebruikt of welke database is gebruikt. Een Yii foutmelding ziet er namelijk anders uit dan een Laravel foutmelding. En elke informatie is voor een hacker waardevol. Jij wilt de hacker zo min mogelijk informatie verstrekken. Dus foutmeldingen gaan naar een log file waar jij alleen bij kan, en de eindgebruiker (of hacker) ziet niets, of hooghuit een vage melding dat er een fout is opgetreden.

Dit zag je ook terug in de Plesk module. Super onhandig want er ging wat fout en je zag niets (behalve als je in de log file keek).

> Op een productie systeem laat je geen specifieke foutmeldingen zien.

Encryptie

Stel je database wordt gehacked en in de database staan logins en wachtwoorden en die wachtwoorden zijn gewoon leesbaar. Dan heb je een probleem, zeker als gebruikers een wachtwoord vaker gebruiken (op andere systemen).

Als je wachtwoorden encrypted in de database opslaat dan is het lastiger om wachtwoorden te 'stelen'. Hackers maken gebruik van zogenaamde rainbow tabellen waardoor encrypted wachtwoorden soms toch te lezen zijn. Om dan weer te voorkomen zou je een encrypty moeten gebruiken met 'salting'. Je gooit dat als het ware zout door een password. Hierdoor kunnen rainbow tabellen niet wordne gebruikt. het gaat nu te ver om uit te leggen hoe dat precies werkt, maar wat je moet onthoduen is:

> Wachtwoorden in de database (en evetnueel andere kwetsbare data) moet je beschermen met encrypty het liefst met *salting*.

Voorkom SQL injection

Met SQL-injection kan je ongewenste code meesturen in bijvoorbeeld een wachtwoord. De code wordt dan uitgevoerd als het wachtwoord wordt gecontroleerd. Op deze manier kan je teogang krijgen tot een systeem zonder het wachtwoord te weten.

Hier kan je je op verschillend emanieren tegen beschermen.

> Afhankelijk van wat voor technieken je gebruikt (framework, PDO, ...) kun je verschillende dingen doen om SQL-injection te voorkomen.

Voorkom cross site scripting

Cross site scripting is van af één verkeerde server een (aangepast) formulier gebruiken en daarmee ongecontroleerde gegevens naar een andere server sturen.

Om dit te voorkomen kan je CSRF-tokens gebruiken. In de meeste frameworks zoals Laravel en Yii is dit al voor je ingebouwd.

> Voorkom cross site scripting door gebruik te maken van CSRF-tokens.

Wil je meer weten van een bepaalde techniek, vraag dan je docent.

Voor jou is het belangrijk om te bepalen wat jij moet doen met jow project om een secure systeem te maken.

Opdracht

...

09 Ethiek

10 Privacy

Testen C24

Ontwerpen C22

Ontwerp - Web site

De site als geheel heeft vaak een ontwerp (template). Je hebt een soort pagina structuur die vaak terugkomt op alle pagina's.

Dit kun je weergeven in een wireframe, bijvoorbeeld:

image-1688328258540.png

Wireframes kun je bijvoorbeeld maken met Lucidchart of Powerpoint.

Screenshot 2023-07-14 143000.png

Wireframe in Powerpoint

Een wireframe kun je in principe gebruiken om een schets te maken voor alle pagina's niet alleen voor de template-pagina.

Site map

In een site map laat je de structuur van je gehele website zien. Dus hoe je navigeert van de ene naar de andere pagina. Bijvoorbeeld:

Screenshot 2023-07-14 142428.png

Een website sitemap geeft de structuur van je website visueel weer. Daarom is het bepalen en vastleggen van een sitemap een belangrijke eerste stap tijdens de ontwikkeling van je website. Alle informatie die je op je website wil aanbieden zal ergens op een web pagina in de sitemap een plek krijgen.

(bron: https://onlineexpert.be/online-tips/website-ontwikkeling/website-sitemap-maken)

ERD

ERD's hebben we het over gehad bij DB ontwerp. Een duidelijk ERD kan de database structuur goed weergeven en is erg handig voor developers die mogelijk na jou met de site aan de slag gaan.

Screenshot 2023-07-14 143533.png

Opgave

Opdracht: Maak een Sitemap voor een Online Boekwinkel

Context: Je bent de hoofdontwerper van een nieuwe online boekwinkel genaamd 'BoekParadijs'. De boekwinkel wil een brede selectie van boeken aanbieden in verschillende genres, evenals audio- en e-boeken. Daarnaast zal de winkel een blogsectie hebben waarin boekrecensies, auteurinterviews en literair nieuws worden gedeeld. De website zal ook pagina's hebben voor klantenservice, over ons, FAQ, en contactinformatie.

Taak: Jouw taak is om een sitemap te cre ren voor de BoekParadijs website. De sitemap moet alle belangrijke pagina's en hun hi rarchische relaties weergeven. Je moet ook rekening houden met de gebruikerservaring en hoe de klant door de website zou navigeren.

Stappen:

1. Identificeer de belangrijkste secties en pagina's van de website. Dit omvat de homepage, productpagina's (verschillende genres, audio boeken, e-boeken), de blog, klantenservice, FAQ, over ons en contactpagina's.
2. Bepaal de hi rarchie van de website. Wat zijn de belangrijkste categorie n en subcategorie n? Hoe verhouden deze pagina's zich tot elkaar?
3. Teken je sitemap uit op papier of gebruik een online tool zoals Lucidchart, Balsamiq, Microsoft Visio of POwerpoint (of een andere goede tool). Zorg ervoor dat de structuur van de website gemakkelijk te begrijpen is vanuit de sitemap. Zet er een korte omschrijving/uitleg bij.
4. Geef je sitemap aan een collega of klasgenoot en vraag hen om feedback . Is de structuur van de website duidelijk? Zijn er pagina's die ze niet kunnen vinden? Geef aan wie de feedback heeft gegeven!

Inleveren

Lever je definitieve sitemap in als een PDF-bestand.

Het document bevat drie onderdelen:

1. De sitemap.
2. Korte uitleg.
3. Feedback van klasgenoot (met naam).

Ontwerpen

Inleiding

Een ontwerp moet de klant een idee geven van wat hij krijgt. Het moet wat vertellen over de functionaliteiten en de gebruikers interface. De user story is een manier om uit tel leggen wat er gebouwd gaat worden. De user story's kunnen ook zeker deel uit maken van de planning, maar komen ook vaak terug in het ontwerp.

User Story

Een user story is een eenvoudige manier om te beschrijven wat een gebruiker van een product of software wil kunnen doen. Het helpt de mensen die de software maken om zich te focussen op de behoeften van de gebruiker. Een user story ziet er meestal zo uit: "Als [een type gebruiker], wil ik [een bepaalde actie kunnen doen], zodat ik [een bepaald resultaat kan bereiken]".

Hier is een voorbeeld: "Als een bezoeker van de website, wil ik een zoekfunctie kunnen gebruiken, zodat ik snel de informatie kan vinden die ik nodig heb."

Dus, het idee is om te beschrijven wie iets wil doen (de bezoeker van de website), wat ze willen doen (een zoekfunctie gebruiken), en waarom ze het willen doen (om snel informatie te kunnen vinden). User story's helpen teams om te begrijpen wie hun gebruikers zijn, wat ze nodig hebben en waarom. Ze helpen ook om het werk op te delen in behapbare stukken die makkelijker te plannen en te beheren zijn.

Voorbeeld

Stel je voor dat je werkt voor een softwarebedrijf dat een e-commerce website ontwikkelt voor een lokaal modebedrijf. Het bedrijf wil dat klanten hun producten online kunnen bekijken, in hun winkelwagentje kunnen plaatsen, kunnen afrekenen en hun bestelgeschiedenis kunnen bekijken.

Je hebt een ontmoeting gehad met de bedrijfseigenaar en ze heeft je de volgende informatie gegeven:

- "We willen dat onze klanten onze collectie kleding en accessoires online kunnen bekijken. Ze moeten ook in staat zijn om de items te filteren op categorie, prijs en maat."
- "Klanten moeten in staat zijn om items aan hun winkelwagentje toe te voegen en te verwijderen."
- "Eenmaal klaar met winkelen, moeten klanten in staat zijn om veilig af te rekenen met hun creditcard of PayPal."

- "We willen ook dat klanten hun bestelgeschiedenis kunnen bekijken en de status van hun huidige bestellingen kunnen volgen."

Jouw taak: Identificeer en schrijf minstens vier user stories op basis van de bovenstaande informatie.

De user story's voor deze case zouden er zo uit kunnen zien:

1. **Producten bekijken en filteren** : "Als klant wil ik de collectie kleding en accessoires online kunnen bekijken en deze kunnen filteren op categorie, prijs en maat, zodat ik gemakkelijk de items kan vinden die ik wil kopen."
2. **Items aan winkelwagentje toevoegen/verwijderen** : "Als klant wil ik in staat zijn om items aan mijn winkelwagentje toe te voegen en te verwijderen, zodat ik kan beslissen wat ik wil kopen voordat ik afreken."
3. **Veilig afrekenen** : "Als klant wil ik veilig kunnen afrekenen met mijn creditcard of PayPal, zodat ik mijn aankopen op een veilige en betrouwbare manier kan voltooien."
4. **Bestelgeschiedenis en status bekijken** : "Als klant wil ik mijn bestelgeschiedenis kunnen bekijken en de status van mijn huidige bestellingen kunnen volgen, zodat ik ....."

Opdracht

Maak deze laatste user story af en lever het in als TXT bestand.

User Stories - bibliotheek

Stel je voor dat je werkt voor een softwareontwikkelingsbedrijf dat is ingehuurd om een nieuwe mobiele app te bouwen voor een lokale bibliotheek. De bibliotheek wil haar leden in staat stellen om boeken en andere bronnen te zoeken, reserveren, en te verlengen via de app. Ook willen ze dat de leden meldingen kunnen ontvangen als hun boeken bijna te laat zijn.

Je hebt een ontmoeting gehad met de bibliothecaris en ze heeft je de volgende informatie gegeven:

- "We willen dat onze leden in staat zijn om te zoeken naar boeken in onze catalogus. Ze moeten kunnen zoeken op titel, auteur, of genre."
- "Eenmaal gevonden, moeten leden in staat zijn om een boek te reserveren als het momenteel uitgeleend is."

- "Leden moeten ook in staat zijn om de uitleentermijn van hun boeken te verlengen via de app."
- "We willen ook dat leden meldingen kunnen ontvangen wanneer hun boeken bijna te laat zijn, zodat ze kunnen worden verlengd of geretourneerd."

Jouw taak: Identificeer en schrijf minstens vier user stories op basis van de bovenstaande informatie. Vergeet niet de structuur te volgen: "Als [een type gebruiker], wil ik [een bepaalde actie kunnen doen], zodat ik [een bepaald resultaat kan bereiken]".

Tips:

- Elke functie of doel dat de bibliothecaris noemt, kan worden omgezet in een user story.
- Bedenk wie de "gebruiker" is in elk geval - het kan de bibliotheeklid, de bibliothecaris, of iemand anders zijn.
- Probeer je te concentreren op wat de gebruiker wil bereiken en waarom, niet op hoe ze het gaan doen.

Als je klaar bent met de oefening, vergelijk dan je user stories met die van je medestudenten. Zijn er functies die je hebt gemist, of heb je iets opgenomen dat zij niet hebben? Bespreek de verschillen en zie of je een consensus kunt bereiken.

Inleveren

Uitgewerkte User story's in PDF.

User Stories - projectmanagement-systeem

Stel je voor dat je werkt voor een softwareontwikkelingsbedrijf dat is ingehuurd om een geavanceerd projectmanagement-systeem te bouwen. Het systeem moet projectmanagers in staat stellen om projecten te maken en te beheren, taken toe te wijzen aan teamleden, de voortgang van het project te volgen en rapporten te genereren.

Je hebt een ontmoeting gehad met de opdrachtgever en ze hebben je de volgende informatie gegeven:

- "We willen dat projectmanagers in staat zijn om projecten te cre ren, te bewerken en te sluiten. Ze moeten ook details van het project zoals deadline, budget, en projectdoelstellingen kunnen specificeren."

- "Projectmanagers moeten taken kunnen toewijzen aan teamleden en deadlines voor deze taken kunnen instellen."
- "We willen dat projectmanagers en teamleden de status en voortgang van taken en het algemene project kunnen bijhouden."
- "We willen dat projectmanagers rapporten kunnen genereren die de huidige status, voortgang, en prestaties van het project laten zien."

> *Als projectmanager wil ik rapporten kunnen maken om de huidige status, voortgang en prestaties van een project te kunnen laten zien. - is dit goed ?*

Dit is gedeeltelijk goed, maar denk in het doel van de actor. De actor is in dit geval de projectmanager. Waar wordt hij voor betaald, wat is zijn doel? Het kan zijn dat hij moet rapporteren over de status van zijn project aan zijn baas of dat hij de voortgang wil kunnen zien om vroegtijdig in te grijpen als het misgaat, of om medewerkers te complimenteren als het goed gaat. Het is immers de verantwoordelijkheid van de PM om ervoor te zorgen dat de medewerkers weten wat ze moeten doen en dat ook daadwerkelijk goed doen.

Bij een user story denk je dus in **doelen** en vanuit de v **erantwoordelijkheid van de actor** .

Jouw taak: Identificeer en schrijf minstens vier user stories op basis van de bovenstaande informatie.

Inleveren

Uitgewerkte User story's in PDF.

User Stories - sportschoolketen

Stel je voor dat je deel uitmaakt van een softwareontwikkelingsteam dat werkt aan een nieuwe app voor een sportschoolketen. Deze app moet leden van de sportschool de mogelijkheid geven om hun fitnessdoelen te volgen, lessen te boeken, persoonlijke trainingssessies te organiseren en hun voortgang bij te houden.

Tijdens een meeting met de manager van de sportschoolketen, ontvang je de volgende informatie:

- "We willen dat onze leden hun fitnessdoelen kunnen instellen en bijhouden op basis van hun persoonlijke behoeften en wensen."
- "Leden moeten in staat zijn om lessen te boeken, zoals yoga of spinning, en hun boekingen te wijzigen of te annuleren."
- "Het zou geweldig zijn als leden ook persoonlijke trainingssessies kunnen boeken via de app, waarbij ze hun voorkeurstijden en -dagen kunnen opgeven."

- "We willen dat leden hun voortgang kunnen bijhouden, inclusief het aantal bijgewoonde lessen, calorie n verbrand en verbeteringen in hun fitnessniveau."

Jouw taak: Identificeer en schrijf minstens vier user stories op basis van de bovenstaande informatie.

Inleveren

Uitgewerkte User story's in PDF.

Van User Storie naar Ontwerp

Het omzetten van de user story naar een ontwerp vraagt om creativiteit en technisch inzicht. Je moet namelijk weten wat er kan en wat er niet (makkelijk) kan. Laten we het voorbeeld uit de vorige opgave nemen.
User story 1: "Als lid van de sportschool wil ik mijn fitnessdoelen kunnen instellen en bijhouden op basis van mijn persoonlijke behoeften en wensen, zodat ik gemotiveerd blijf en mijn voortgang kan zien."

Ontwerpvoorstel:

Voor deze user story zou je een 'Mijn Fitnessdoelen' sectie in de app kunnen ontwerpen. Deze sectie zou kunnen bestaan uit:

1. **Doelen instellen** : Maak een interface waar gebruikers hun fitnessdoelen kunnen instellen. Dit kan een formulier zijn waar ze hun huidige gewicht, streefgewicht, gewenste spiermassa, het aantal dagen dat ze willen sporten, en andere relevante details kunnen invoeren. Je zou ook een kalender of een slider kunnen toevoegen waarmee gebruikers een streefdatum kunnen instellen.
2. **Voortgang bijhouden** : Laat gebruikers hun dagelijkse of wekelijkse voortgang bijhouden door hun gewicht, genomen lessen, verbrande calorie n, enz. in te voeren. Dit zou kunnen worden weergegeven in de vorm van een grafiek of tabel, die hun voortgang over tijd laat zien in vergelijking met hun doelen.
3. **Meldingen** : Om gebruikers gemotiveerd te houden, kun je pushmeldingen of reminders toevoegen. Deze kunnen bijvoorbeeld een herinnering sturen om hun voortgang in te voeren of ze feliciteren als ze een mijlpaal hebben bereikt.

Nu zou je de ontwerpen moeten illustreren aan de hand van GUI ontwerpen. Neem bijvoorbeeld punt 2. De GUI zou er zo uit kunnen zien:

image-1688325663652.png image-1688325695287.png

Toelichting

Als omschrijving zou je nog kunnen toevoegen wat je bij de verschillende velden kan en mag invullen, bijvoorbeeld:

Als GUI kiezen we voor Bootstrap. Dit is een vele gebruikte interface die snel ontwikkeld en die er clean/eenvoudig uitziet. Later kunnen we de GUI eventueel aanpassen maar een meer "op-maat" ontwerp.

- Bij huidige gewicht moet je je gewicht in kilo's invullen. Dat kan op n decimaal nauwkeurig. Bijvoorbeeld 61.4. Het gewicht moet tussen de 0 en 300 liggen. De gebruiker kan met behulp van een drop down menu het juiste gewicht kiezen.
- Het aantal genomen lessen voor deze week is n getal, de opties zijn 0,1,2,3,4,5,6,6+
  Hier wordt ook een drop down gebruikt (zie voorbeeld rechter plaatje).
- Bij het aantal verbranden calorie n kan de waarde w=volgens de fitness watch worden ingevoerd. Waardes die kunnen worden gekozen zijn 40, 50, 60, 70, 80, 90, 100, 120, 140, 160, 180, 200, 240, 280, 320, 360, 400, 450, 500, 550, 600, 650, 700, 750, 800, 900, 1000, 1200, 1200+
  De gebruiker moet ene waarde kiezen die zo dicht mogelijk tegen de gemeten waarde licht.

Hoe maak je een GUI ontwerp?

Deze GUI is gemaakt met behulp van ChatGPT. Er zijn talloze handige online tools om GUI ontwerpen te maken.

Tools voor ontwerpen GUI:

- https://www.lucidchart.com (g ebruik 'Website blockframe')
- http://www.phpform.org/
- https://app.lucidchart.com

Opdracht

Maak een mooi ontwerp van user story 1 uit de vorige opgave:

Fitnessdoelen instellen en volgen : "Als lid van de sportschool wil ik mijn fitnessdoelen kunnen instellen en bijhouden op basis van mijn persoonlijke behoeften en wensen, zodat ik gemotiveerd blijf en mijn voortgang kan zien."

1. Maak een ontwerp met een sportieve uitstraling.

2. leg uit waarom je dit gekozen hebt en waarom het een sportieve uitstraling heeft.

3. Leg uit hoe het formulier werkt en wat de gebruiker moet/mag invullen.

Inleveren

PDF met ontwerp (plaatje) n de uitleg; waarom heeft dit een sportieve uitstraling en hoe werkt het formulier?

Ontwerp - pushbericht

We hebben nog een derde user story;

Meldingen : Om gebruikers gemotiveerd te houden, kun je pushmeldingen of reminders toevoegen. Deze kunnen bijvoorbeeld een herinnering sturen om hun voortgang in te voeren of ze feliciteren als ze een mijlpaal hebben bereikt.

Opdracht

Maak een ontwerp van minimaal drie pushberichten. Probeer verschillende soorten pushberichten te bedenken.

Laat zien hoe de berichten er uitzien en licht toe wat de functie is en waarom je voor het ontwerp hebt gekozen.

Ethiek, privacy en security

De onderwerpen ethiek, privacy en security worden steeds belangrijker. Wat betekent deze begrippen eigenlijk?

Ethiek

Ethiek is een verzameling regels en principes waarmee wordt bepaald wat goed en wat slecht is.

Hoe zorg jij er dus voor dat er op jouw website:

- Geen racistische uitingen kunnen worden gedaan?
- Hoe zorg je ervoor dat er respectvol met elkaar wordt omgegaan via de website?
- Is de website inclusief en hoe heb je daar rekening meer gehouden?
- Staan er geen teksten en/of symbolen op jouw website die een groep mensen kan kwetsen?
- Heb je er voor gezorgd dat oplichters niet (zo makkelijk) via jouw site mensen kunnen oplichten?

Deze dingen kunnen niet van toepassingen zijn, maar het is goed om je deze dingen af te vragen en te kijken of er zaken zijn waarmee je rekening meer moet houden.

Privacy

Privacy betekent dat iemand dingen kan doen zonder dat de buitenwereld daar weet van heeft, inbreuk op maakt, of invloed op heeft. Rekening houden met privacy is belangrijk omdat niet alle mensen willen dat al hun gegevens openbaar op het Internet te vinden zijn.

Hoe zorg jij er dus voor dat er op jouw website:

- Geen onnodige gegevens worden opgeslagen?
- Gegevens die niet meer nodig zijn worden verwijderd (bijv. als iemand een half jaar niet heeft aangelogd; leg ook uit hoe)?
- Alleen die mensen persoonlijke gegevens kunnen zien van die mensen die echt nodig zijn. De bank die de betalingen regelt hoeft bijvoorbeeld niet te weten welke producten er zijn besteld maar natuurlijk wel wat de naam van de gebruiker is en het te betalen bedrag.
- Gebruikers niet onnodig worden verplicht om zich te registreren.
- Er bij de registratie van gebruikers geen onnodige gegevens worden gevraagd en op opgeslagen. Je zou er bijvoorbeeld voor kunnen kiezen om een creditcard nummer *niet* op te slaan.

Security

Security zorgt ervoor dat de website minder makkelijk kan worden gehackt.

Hoe zorg jij ervoor dat:

- Gegevens encrypted worden opgeslagen. Let dan uit welke gegevens op welke manier encypted worden.

- Wordt een 'Salt' gebruikt bij de encryptie van gegevens?
- Dwing je een wachtwoord policy af, zo ja welke?
- Hoe voorkom je SQL injection?
- Hoe voorkom je XSS (serve cross site scripting)?
- Gebruik je gegevens validatie aan de client en de server kant?
- Gebruik je (in productie) https in plaats van http?
- Hebben gebruikers niet onnodig veel rechten?
- Kun je IP nummers blokkeren of maak je gebruik van een IP white list?

Al deze vragen zijn voorbeelden van vragen die je jezelf kan stellen bij het ontwerpen van een website.

Kijk of er vragen zijn die voor jouw website relevant zijn en beschrijf ze dan.

Voorbeeld

In het voorbeeld van de sportschool was er gekozen voor een drietal dropdowns waarmee waarden konden worden ingevoerd. Het nadeel si misschien dat dit iets onhandig invoert, maar het voordeel is dat de gebruiker wordt gelimiteerd in het posten van 'ongeldige' invoer. Er kunnen geen negatieve waarden of strings worden ingevoerd. Dit is goed voor de security want daarmee hebben hackers minder mogelijkheden om de site te hacken (bijv. SQL injection).

Opdracht

Je hebt een eerder een ontwerp voor pushberichten gemaakt. Dat gingen over user story:

Meldingen : Om gebruikers gemotiveerd te houden, kun je pushmeldingen of reminders toevoegen. Deze kunnen bijvoorbeeld een herinnering sturen om hun voortgang in te voeren of ze feliciteren als ze een mijlpaal hebben bereikt.

Ga na of en hoe de onderwerpen ethiek, privacy en security te maken kunnen hebben met het ontwerpen van deze functionaliteit?

Dus kun je rekening houden met ethiek, privacy of security bij het ontwerp van deze functionaliteit? Leg uit waar dit het geval kan zijn.

Inleveren

PDF-documentje waarin jij uitlegt hoe je met het ontwerp van de user story pushberichten rekening kan houden met n of meer van de onderwerpen ethiek, privacy en/of security.

Eindopdracht

We gaan nog eens terug naar de user story over de bibliotheek:

Boek reserveren : "Als lid van de bibliotheek wil ik een boek kunnen reserveren dat momenteel is uitgeleend, zodat ik het kan lenen zodra het beschikbaar is."

Opdracht

Werk de user story uit.

- Maak een GUI, leg uit waarom je voor de GUI hebt gekozen en hoe deze werkt. Leg ook uit dat een lid van bibliotheek natuurlijk eerst moet zijn aangelogd. Zet het ontwerp (n of meer) plaatjes in een PDF en leg uit waarom je voor rit ontwerp hebt gekozen en hoe het werkt.
- Ga de onderwerpen ethiek, privacy en security na en kijk in hoeverre jij rekening kan houden met deze onderwerpen en wat dat dan betekent voor jou ontwerp.

Inleveren

Zet dit allemaal in een PDF en lever dat in.

Testen C22

Testrapport

Een testrapport bestaat uit

- **Inleiding** : Geef een korte beschrijving van de software die wordt getest, de versie van de software, het doel van het testen en andere relevante informatie, zoals de datum van de test.
- **Lijst van use cases** : Maak een lijst van de use cases die je hebt getest. De lijst van use cases heb je al vanuit je planning en ontwerp.
- **Testplan** : Geef een overzicht van je testplan. Per use case stel je scenario's op. En per scenario stel je alle stappen op.

De volgorde (of eigenlijk hi rarchie) is dus: **use case - scenario - stappen**

- **Testresultaten** : Je voert elk scenario n of meer keer uit en beschrijft je bevindingen.
    1. **Fouten en problemen** : Documenteer alle fouten of problemen die je tegenkwam tijdens het testen. Dit kan van alles zijn, van bugs in de software tot problemen met de gebruikservaring.
- Als alles is getest beschrijf je een conclusie, daarin staat:
    1. **Bevindingen en conclusies** : Op basis van de resultaten van de tests, wat zijn je algemene bevindingen? Werkt de software zoals verwacht in alle geteste use cases? Zo niet, welke problemen heb je ge dentificeerd?
    2. **Aanbevelingen** : Geef aanbevelingen op basis van je testresultaten. Dit kunnen suggesties zijn voor het verbeteren van de software, voor aanvullende tests die moeten worden uitgevoerd, of voor het oplossen van ge dentificeerde problemen.
- **Bijlagen** : Voeg eventuele relevante bijlagen toe, zoals screenshots, logbestanden of gedetailleerde resultaten van individuele tests.

In de volgende lessen gaan we deze stappen oefenen.

Opgave

Neem als voorbeeld de CRUD Challenge die je vorig jaar hebt gemaakt.

Benoem n use case en bedenk n scenario die bij deze use case hoort.

Inleveren

Use case (in de vorm als....wil ik....zodat.....) met een bijbehorend test scenario.

Stappen

Voorbeeld stappen

Je bent een software tester voor de website van de Openbare Bibliotheek Amsterdam (https://www.oba.nl/). Je taak is om een testrapport te maken over de volgende use case:

Use Case: Als lid van de bibliotheek wil ik een Boek Zoeken en reserveren zodat ik het boek kan ophalen en kan lenen.

Scenario 1: Een gebruiker wil een specifiek boek vinden en hij weet alleen de titel van het boek.

Scenario 2: Een gebruiker wil een specifiek boek vinden en hij weet alleen het ISBN-nummer van het boek.

Scenario 3: Een gebruiker wil een specifiek boek vinden en hij weet alleen de naam van de schrijver van het boek.

Stappen voor scenario 1

1. De gebruiker gaat naar de homepage van de website.
2. De gebruiker klikt op het zoekveld in het midden van de pagina.
3. De gebruiker voert de titel van het boek in.
4. De website geeft een lijst met resultaten weer die overeenkomen met de zoekterm van de gebruiker.
5. De gebruiker klikt op de titel van het boek dat hij/zij wil reserveren.
6. De gebruiker wordt naar een pagina geleid met details over het geselecteerde boek.

Opdracht

Beschrijf het stappenplan voor scenario 1,2 en 3. Let op het stappenplan zoals het hier staat *mist een paar details* .

Inleveren

3 keer een volledig stappenplan voor scenario 1 2 en 3.

Tip

Let op je mag een stappenplan maken omdat de stappen bijna allemaal hetzelfde zijn. Op het punt waar het verschillend is, maak je dan een onderverdeling:

1. ....
2. ....
3. Hier zijn drie mogelijkheden:
    1. sub scenario 1
    2. sub scenario 2
    3. sub scenario 3
4. ....
5. ....

Hang Man - inleiding

Inleiding

Je gaat een Python spelletje installeren. Dit spel gaan we testen.

Spel

Je hebt een *hang man* spel, waarvan hier de code:

```
# versie 0.9 hangman

import random

def hangman():
    word_list = ["python", "hangman", "programming", "game", "computer"]
    word = random.choice(word_list).lower()
    guessed_letters = []
    attempts = 8
```

```
    print("Welkom bij Hangman!")
    print("Probeer het woord te raden door letters te kiezen.")
    print(f"Je hebt {attempts} pogingen voordat het spel voorbij is.")

    while True:
        hidden_word = ""
        for letter in word:
            if letter in guessed_letters:
                hidden_word += letter + " "
            else:
                hidden_word += "_ "

        print("\n" + hidden_word)

        if "_" not in hidden_word:
            print("Gefeliciteerd! Je hebt het word geraden.")
            break

        if attempts == 1:
            print(f"Helaas, je hebt het woord niet geraden. Het woord was {word}.")
            break

        guess = input("Kies een letter: ").lower()

        if len(guess) != 1 or not guess.isalpha():
            print("Invalid input. Voer   n enkele letter in.")
            continue

        if guess in guessed_letters:
            print("Je hebt deze letter al geraden. Probeer het opnieuw.")
            continue

        guessed_letters.append(guess)

        if guess in word:
            print("Goed geraden!")
        else:
            attempts -= 1
            print(f"Fout! Je hebt nog {attempts} pogingen over.")

if __name__ == "__main__":
    hangman()
```

Kopieer de code en speel het spel een keer.

Opdracht 1

Maak een inleiding.

Geef een korte beschrijving van de software die wordt getest, de versie van de software, het doel van het testen en andere relevante informatie, zoals de datum van de test.

Inleveren

De inleiding van het testrapport

Hang man - use cases en scenario's

Use Cases

Je hebt eigenlijk maar een paar use cases:

1. Als speler wil het spel starten omdat ik het spel wil gaan spelen
2. Als speler wil een beurt spelen en een letter raden om het woord te raden
3. Als speler wil ik het spel stoppen omdat ik het niet wil afmaken
4. Als beheerder wil ik de woorden die een speler kan raden veranderen

Bedenk bij alle use cases de stappen die je wilt uitvoeren om het te testen.

Bij use case 2 moet je meerdere zaken testen, denk aan ongeldige invoer, of een letter invoeren die je al hebt geraden, of een hoofdletter invullen van een kleine letter die al is geraden, bedenk verschillende scenario's en beschrijf de stap of stappen.

Bij use case 2 eindigt het spel ook. Het eindigt of omdat je te veel beurten hebt gehad, of omdat je hebt gewonnen en het woord hebt geraden.

Opdracht

Maak een hoofdstukje van elke use case en beschrijf bij elke use case een scenario. Beschrijf de stap of stappen die je moet doorlopen voor het scenario.

Bij use case 2 beschrijf je minimaal 5 scenario's met de bijbehorende stappen.

Als alle stappen zijn beschreven dan voer je alle suit en beschrijf je per proces de bevindingen.

Inleveren

Werk alle scenario's uit en beschrijf deze. Verdeel de scenario's per use case.

--

Hang Man - bevindingen

In de vorige opgave heb je een aantal stappenplannen gemaakt.

Voer alle stappenplannen uit en beschrijf per stappen plan je bevindingen.

Je voert minimaal alle stappenplannen uit, maar mocht je tegen zaken aanlopen die misschien niet in de stappenplannen staan, dan is dat prima. Noteer ook deze bevindingen.

Hoe beschrijf je een bevinding?

Stel je hebt een scenario waarin je de letter A hebt geraden en dat de letter A is afgekeurd. Als je de A opnieuw raadt wil je niet dat dit als een beurt wordt gerekend.

Je bevinding zou er dan als volgt kunnen uitzien:

Nadat de letter A was ingevoerd en was afgekeurd werd de beurt geteld. Ik had n raad-poging minder over. Toen ik opnieuw de A invoerde kreeg ik een boodschap dat ik A al had geprobeerd en het aantal raad-pogingen bleef gelijk. Bij het invoeren van een a (kleine letter) nadat ik een A had geraden was het resultaat hetzelfde. De raadpogingen zijn dus case insensitive.

Opdracht

Voer alle scenario's uit, en beschrijf je bevindingen.

Je kan de scenario's die je had opgesteld nummeren zodat je de bevindingen ook kan nummeren zodat je weet welke bevinding bij welke scenario hoort.

Nog overzichtelijker is om alles in een tabel te zetten!

Inleveren

Scenario's (die je al had) met daarbij de bevindingen.

Hang Man - conclussies

Als laatste maken we paragraaf conclusies.

In de conclusies benoem je het algemene beeld, zoals:

- Wat vind je van de kwaliteit van de software?

- Zijn er zaken die los van de use cases en scenario's aandacht nodig hebben?
- Wat zijn de belangrijkste bevindingen?
- Wat zijn je belangrijkste aanbevelingen of zijn die er niet?

Opdracht

Beschrijf de conclusies van je test.

Tafels leren

Je gaat opnieuw een Python programma testen.

Tafels leren

Met het Python-programma tafels leren .

Use cases

- Als een student wil ik mijn tafels oefenen binnen een bepaalde tijdslimiet, zodat ik mijn snelheid en nauwkeurigheid bij het vermenigvuldigen kan verbeteren."

  Het programma heeft een tijdslimiet. Bij elke nieuwe som wordt de tijdslimiet getoond.

- "Als een student wil ik in staat zijn om de moeilijkheidsgraad van mijn oefeningen te verhogen, zodat ik kan blijven uitdagen en verbeteren van mijn rekenvaardigheden."

  Na elke drie goede antwoorden wordt het niveau verhoogd en worden de vragen moeilijker.

- "Als een student wil ik via een game-achtig programma worden uitgedaagd, zodat ik ga proberen een betere score dan te halen en zodoende motivatie krijg om meer te oefenen. "

  Het 'spel' telt punten en geeft aan op welk niveau je bent uitgekomen. De punten kunnen worden vergeleken met anderen.

- "Als een student wil ik dat dezelfde vraag niet twee keer achter elkaar wordt gesteld, zodat ik een breder scala aan vragen kan ervaren en mijn vermogen om verschillende vermenigvuldigingsproblemen op te lossen kan verbeteren."

  Het programma zou niet twee maal dezelfde vraag achter elkaar moeten stellen.

Opstellen testplan

Maka een testplan zoals je dat hebt geoefend.

Maak de volgende hoofdstukken:

1. **Inleiding** ,
2. **Use cases.**
3. **Scenario's** per use case.
4. S **tappenplan** per scenario.
5. Voer je stappenplan uit.
6. Beschrijf je **bevindingen** per scenario.
7. Beschrijf de algehele **conclussie** .

Inleveren

Volledig testplan in pdf.

Code

```
# version 3.1 (July 2023)
import time
import random

# De tijd wanneer het spel start
start_time = time.time()

# Tijdslimiet voor het spel in seconden
time_limit = 20

# Teller voor het aantal goede antwoorden
correct_count = 0

# Bepaal het initi le bereik voor de getallen
number_range = 5

# Bewaar de laatste vraag
last_question = None

while True:
    # Bereken hoeveel tijd er al verstreken is
    elapsed_time = time.time() - start_time
```

```
    # Als de tijdslimiet is bereikt, stop dan met het spel
    if elapsed_time > time_limit:
        print("Tijd is op!")
        print(f"Je hebt {correct_count} vragen goed beantwoord.")
        print(f"Je eindigde op moeilijkheidsgraad {number_range - 4}.")
        break

    # Bereken de resterende tijd en print deze
    remaining_time = time_limit - elapsed_time
    print(f"Resterende tijd: {remaining_time:.2f} seconden")

    # Verhoog het bereik elke keer dat de gebruiker vijf vragen correct beantwoordt
    if correct_count % 3 == 0 and correct_count > 0:
        number_range += 1
        print(f"Moeilijkheidsgraad verhoogd! Je zit nu op moeilijkheidsgraad {number_range - 4}.

    # Genereer een nieuwe vraag
    num1, num2 = random.randint(1, number_range), random.randint(1, number_range)

    # Controleer of de nieuwe vraag hetzelfde is als de vorige vraag
    # Als dat het geval is, genereer dan een nieuwe vraag
    while (num1, num2) == last_question:
        num1, num2 = random.randint(1, number_range), random.randint(1, number_range)

    # Bewaar de huidige vraag als de laatste vraag
    last_question = (num1, num2)

    correct_answer = num1 * num2

    # Stel de vraag aan de gebruiker
    user_answer = int(input(f"Wat is {num1} * {num2}? "))

    # Controleer het antwoord
    if user_answer == correct_answer:
        print("Correct!")
        correct_count += 1
    else:
        print("Fout! Het juiste antwoord was: ", correct_answer)
```