

Functioneel ontwerp

Project : Portfolio
Opdrachtgever : Christian Schroth
Auteur : Sandrine Prumpeler
Datum : 12/04/'25
Versie : 2.0

Inhoud

1. Opdracht.....	1
2. Doelgroep.....	1
3. Belangrijke functies.....	2
3.1 Bezoekersrollen.....	2
3.2 Functionaliteiten	2
4. Hoe gebruikers de site gebruiken.....	2
4.1 Navigatie	2
4.2 Gebruikersstromen	2
5. Technische basis	3
6. Beperkingen en Mogelijke uitbreidingen	3
7. Beargumentering & Relatie met Ethiek, Privacy & Security	4
8. Haalbaarheid	4

1. Opdracht

De opdracht is om een portfolio website (opnieuw) te maken voor Christian. Hiervoor wilde hij graag als thema league of legends hebben, en verder graag een vrij simpele portfolio website. Ook wilde hij liefst geen afbeeldingen van zichzelf in het ontwerp hebben, dit hadden we de vorige keer wel. Ook waren er een aantal dingen die ik nu anders ga aanpakken.

Hiervoor heb ik ook bestand gemaakt [Plan van aanpak portfolio](#).

Hierin staan alle dingen die ik nu anders wil aanpakken en die bij de vorige opdracht niet goed gingen.

2. Doelgroep

De website is bedoeld voor:

- Studenten die andere portfolio's willen bekijken
- Studenten die een goed groepje zoeken en willen bekijken of hij een eventuele goede match is.

- Scholen voor eventuele vervolgoopleidingen
- Werkgevers die Christian willen aannemen.
- Klanten die zijn werk willen bekijken

3. Belangrijke functies

3.1 Bezoekersrollen

Er is maar één type gebruiker, dat is de bezoeker. Iedereen kan de site bekijken, er is namelijk geen inlog nodig en ook geen beheerder.

3.2 Functionaliteiten

De website moet het volgende hebben:

- Intro met een afbeelding in zijn gekozen game-stijl
- Pagina "Over mij" met informatie over Christian
- Projectenpagina met voorbeelden van zijn werk
- Skills pagina met een aantal skills waar hij goed in is
- Werkervaring-pagina
- Contactpagina met contact formulier dat werkt
- Scroll-Buttons, zodat je maar op een knop hoeft te klikken om naar de volgende sectie te gaan.

4. Hoe gebruikers de site gebruiken

4.1 Navigatie

Op de site komen secties met knoppen, als je op de knop drukt word je automatisch gebracht naar de volgende sectie. Je hebt maar een aantal secties: de intro, over mij, mijn werkervaring, mijn skills, mijn projecten en het contact formulier.

4.2 Gebruikersstromen

Een bezoeker zal ongeveer meestal dit doen/zien:

- Komt op de intro pagina aan en bekijkt de afbeelding + tekst.
- Vervolgens scrollt de gebruiker naar onder en komt uit op de sectie "over mij", hier leest de gebruiker wie Christian is.
- Hierna kan de gebruiker op een knop drukken die hem/haar brengt naar de volgende sectie "mijn werkervaring", hier krijgen ze al zijn stages/bijbaantjes te zien.
- Als ze nu op de knop drukken komen ze uit bij "mijn skills", hier kunnen ze bekijken waar Christian goed in is en wat zijn kwaliteiten zijn.
- Vervolgens kunnen ze op de knop drukken om uit te komen bij "mijn projecten", hier kunnen ze de projecten van Christian bekijken.
- En ten slotte hebben ze nog een button waarmee ze naar de laatste sectie gaan, "contact formulier". Hier kunnen ze ervoor kiezen om een bericht naar Christian te sturen of eventueel terug te gaan naar het begin van de portfolio.

5. Technische basis

Create React App met TypeScript

- De website wordt gemaakt met in een create-react-app (react), ik werk hierin met TypeScript (voor duidelijkere code en minder fouten).

CSS Modules of Styled Components

- Ik maak met CSS de opmaak van de site netjes, gescheiden en goed verdeeld per onderdeel.
- Met CSS maak ik de gekozen thema voor Christian.

C# (ASP.NET)

- Het contactformulier voor de website werkt via een back-end die gemaakt wordt in C# met ASP.NET. Deze zorgt ervoor dat berichten van bezoekers worden doorgestuurd.

HTTPS

- De website is beveiligd met HTTPS, zodat alle gegevens (zoals van het contactformulier) veilig verzonden worden en betrouwbaar is. Zonder HTTPS kunnen gegevens namelijk onversleuteld over het internet gaan.

Componenten (React)

- De website wordt opgebouwd in kleine losse stukjes (componenten), zoals de secties, projectkaart of formulier. Dit maakt de site overzichtelijk en makkelijk aan te passen.

Versiebeheer met Gitkraken

- Ik gebruik Gitkraken om alle codeversies op te slaan en bij te houden. Dit houdt in dat als er iets mis gaat ik de laatst opgeslagen (werkende) versie kan terug halen, ook kan ik bekijken wat ik allenmaal heb aangepast (per opgeslagen sectie, om het verschil te kunnen bekijken).

Azure DevOps

- ik gebruik Azure DevOps voor het plannen van taken en eventueel het automatische testen en publiceren van de website.

6. Beperkingen en Mogelijke uitbreidingen

Beperkingen:

- De klant wilt liever dat we geen foto's van hem gebruiken op de site.
- De klant geeft ook liever niet zijn projecten vrij, vandaar worden het lege linkjes.

Mogelijke uitbreidingen:

- Website eventueel beschikbaar maken in meerdere talen.
- Mogelijkheid om projecten toe te voegen.

7. Beargumentering & Relatie met Ethiek, Privacy & Security

Ethiek

- De site bevat geen nepgegevens of misleidende content.
- Er wordt respectvol omgegaan met de wensen van Christian (geen foto's, geen projecten online).

Privacy

- Geen opslag van gevoelige data.
- Contactformulier verstuurt enkel e-mails en slaat niks lokaal of in de database op.
- Er worden geen trackers, cookies of analytics gebruikt.

Security

- HTTPS voorkomt dat gegevens onversleuteld verzonden worden.
- Front-end en back-end input-validatie voorkomt spam of kwaadaardige invoer.
- Er wordt gebruikgemaakt van veilige verbindingen tussen componenten.
- TypeScript voorkomt runtime fouten en verhoogt de codebetrouwbaarheid.

8. Haalbaarheid

- De technologieën (React, TypeScript, C#, Azure DevOps) zijn al bekend en eerder toegepast.
- Er is geen complex beheerportaal nodig, waardoor het project eenvoudiger blijft.
- De wensen van Christian zijn duidelijk en beperkt (eenvoudige site, geen gevoelige info), wat de haalbaarheid vergroot.
- De planning en voortgang worden beheerd via Azure DevOps en Git, wat zorgt voor overzicht en grip.