

Verantwoordingsdocument – SportData App

1. Technische keuzes en onderbouwing

1.1 Gebruik van de React-componentstructuur

Keuze: De applicatie is opgebouwd met herbruikbare componenten, waaronder NavBar, Home, TeamsList, TeamsDetails en FavoriteTeams.

Onderbouwing: Deze aanpak bevordert de herbruikbaarheid, leesbaarheid en onderhoudbaarheid van de code.

Reflectie: Het gebruik van componenten maakte het mogelijk om logica en styling per onderdeel gescheiden te houden. In toekomstige projecten zal ik van te voren een beter plan maken over welke componenten ik nodig heb. Een extra component toevoegen aan al bestaande code heeft mij veel tijd gekost omdat dit moeilijker in te bouwen was dan vooraf gedacht.

1.2 Routing met React Router

Keuze: Voor de navigatie tussen pagina's is gebruikgemaakt van de react-router-dom-bibliotheek.

Onderbouwing: Hiermee kan de routing worden toegepast, wat resulteert in een vloeiende gebruikerservaring zonder herladen van de pagina.

Reflectie: De implementatie verliep probleemloos. Ter bescherming van afgeschermdede onderdelen, zoals de favorietenpagina, is een ProtectedRoute-component toegevoegd.

1.3 Gebruik van React Hooks (useEffect en useState)

Keuze: Voor het ophalen en beheren van API-data zijn de React-hooks useState en useEffect toegepast.

Onderbouwing: Deze hooks maken het mogelijk om component state en side effects op een overzichtelijke manier te beheren.

Reflectie: Dit heeft geleid tot dynamische en responsieve componenten. Er is veel inzicht opgedaan in het correct omgaan met afhankelijkheden in useEffect.

1.4 Opslag van favorieten via localStorage

Keuze: De lijst met favoriete teams wordt opgeslagen in de browser via localStorage.

Onderbouwing: Hierdoor blijven gegevens behouden na het herladen van de pagina of het afsluiten van de browser, zonder dat een backend nodig is.

Reflectie: Deze oplossing werkte stabiel tijdens het testen. Voor een productieomgeving zou gekozen worden voor persistente opslag gekoppeld aan gebruikersprofielen via een backend.

1.5 Conditionele rendering en foutafhandeling

Keuze: De applicatie controleert of gegevens beschikbaar zijn voordat deze worden weergegeven, en toont waar nodig een alternatieve weergave of melding.

Onderbouwing: Dit verhoogt de gebruikersvriendelijkheid en voorkomt fouten in de interface bij ontbrekende of onjuiste data.

Reflectie: In eerste instantie waren deze controles niet overal aanwezig, maar deze zijn tijdens het ontwikkelproces stapsgewijs toegevoegd.

2. Beperkingen en mogelijkheden voor doorontwikkeling

2.1 Beperkte gegevens via de gratis API

Beperking: De gratis versie van de TheSportsDB API levert niet altijd actuele of volledige data voor alle teams hiervoor heb je de betaalde versie nodig. Doordat ik de gratis versie gebruik zie je bij teamdetails altijd Arsenal als voorbeeld terwijl het een ander team is met een ander teamID, helaas laat de gratis versie dit niet toe. Dit is op meer pagina's het geval.

Doorontwikkeling: De betaalde versie van de API kan rijkere en betrouwbaardere informatie leveren, wat ten goede komt aan de datakwaliteit en gebruikerservaring.

2.2 Geen mogelijkheid om favorieten te sorteren of te beheren

Beperking: Gebruikers kunnen wel teams aanmerken als favoriet, maar hebben geen mogelijkheid om deze te sorteren op bijvoorbeeld laatst gespeelde wedstrijd, alfabetische volgorde of competitie. Daarnaast ontbreekt de optie om meerdere teams tegelijk te verwijderen of om notities toe te voegen.

Doorontwikkeling: Door het toevoegen van sorteer- en beheermogelijkheden in de favorietenlijst wordt de applicatie gebruiksvriendelijker en krijgt de gebruiker meer controle over zijn of haar voorkeuren. Dit draagt bij aan een meer gepersonaliseerde ervaring.

2.3 Geen zoekfunctionaliteit in teamoverzicht

Beperking: Teams worden per competitie getoond, maar er is geen mogelijkheid tot zoeken of filteren.

Doorontwikkeling: Een zoekveld of filteropties (zoals per land of teamnaam) kunnen de gebruiksvriendelijkheid verhogen.

2.4 Beperkte formulier-validatie

Beperking: De invoervelden voor registratie en inloggen maken enkel gebruik van standaard HTML-validatie.

Doorontwikkeling: Het toevoegen van veldspecifieke validatie (bijvoorbeeld

wachtwoordsterkte of controle op unieke gebruikersnaam) draagt bij aan betere gebruikersfeedback en veiligheid.

2.5 Favorieten zijn niet accountgebonden

Beperking: Favoriete teams worden opgeslagen in localStorage en zijn dus alleen beschikbaar op het gebruikte apparaat.

Doorontwikkeling: Door favorieten te koppelen aan een gebruikersaccount via een backend, kunnen deze op meerdere apparaten worden gesynchroniseerd.

3. Persoonlijke reflectie

Tijdens de ontwikkeling van deze applicatie heb ik veel kennis opgedaan over het bouwen van een schaalbare frontend met React, het integreren van externe API's, en het toepassen van routing en component-based development. Naarmate het project

vorderde, kreeg ik meer grip op het structureren van componenten en het adequaat afhandelen van state en side effects.

Het belangrijkste leerpunt binnen dit project is het belang van een duidelijke componentstructuur en tijdige foutafhandeling binnen dynamische webapplicaties.