

Titelblad

Novi Hogeschool

Eindopdracht Frontend

Functioneel ontwerp

Tahmina Hesari

Inleiding

Dit is mijn functioneel ontwerp voor de Frontend leerlijn, ik maak als eindopdracht een webapplicatie voor de fans van Zuid Koreaans K-Pop boyband "BTS". Als naam koos ik "Borahae" wat letterlijk betekent "I Purple You".

Door mijn interesse in K-Pop en K-drama ontstond het idee om een platform te maken, waarin BTS fans alle quotes van alle 7 members terugvinden op 1 plekje. Men kan ook quotes uploaden die door de BTS members geschreven zijn. Er is ook een spelletje waarin de gebruiker moet raden van wie een quote is.

De BTS fanclub heet A.R.M.Y. Deze afkorting staat voor Adorable Representative M.C. for Youth en heeft wereldwijd 40 miljoen leden.

BTS staat heel dichtbij hun fans en hun muziek is soms zelfs geïnspireerd door fans.

Inhoudsopgave

Titelblad	1
Inleiding.....	2
Inhoudsopgave.....	3
Probleembeschrijving.....	4
50 Functionele- en niet-functionele eisen.....	5
Inspiratie bronnen.....	9
4 Use case tabellen.....	10
Wireframes	15
Schermontwerpen.....	17
Verantwoordingsdocument	20

Probleembeschrijving

Dit is ideefase-probleembeschrijving voor mijn Frontend eindproject.

BTS maakt verschillende soorten content voor fans o.a. zoals bijvoorbeeld Game Shows en reality programma's.

Op dit moment bestaat nog geen platform waarin fans alle quotes van BTS kunnen terugvinden. Daarnaast wil ik ook een game maken met alle quotes van alle member. En ook mogelijkheid geven dat fans zelf geschreven quotes door de members kunnen uploaden op de platform.

De functionaliteit in het kort:

- Fan kan een account aanmaken
- Fan kan inloggen
- Een ingelogde fan kan een quote uploaden
- Alle quotes zien van alle members
- Gebruiker kan raden van wie de quote is (game)

Om data op te halen gebruik ik volgende API <https://github.com/oanhgle/bangtan-api>

Novi backend is voldoende om data van mijn gebruikers opslaan zoals gebruikersnaam, email en wachtwoord.

Novi Backend API:

<https://github.com/hogeschoolnovi/novi-educational-backend-documentation>

50 Functionele- en niet-functionele eisen

Home pagina niet functionele eisen

1. Als gebruiker homepagina refreshed krijg je een nieuwe quote te zien.
2. Bij de tabblad van een willekeurige internet browser komt naam van de webapplicatie te staan "Borahae" met een paars hart icoontje.

Sign Up pagina functionele eisen

3. Gebruiker kan zich met een Sign-Up formulier registreren.
4. Als de gebruiker niet alle invoervelden invult kan er niet geregistreerd worden.

Sign Up pagina niet functionele eisen

5. Als gebruiker geen gebruikersnaam invult tijdens registratie krijgt een foutmelding te zien.
6. Als de gebruikersnaam uit minder dan 2 tekens bestaat, wordt er een foutmelding weergegeven.
7. Als gebruikers e mailadres zonder "@" invult krijgt foutmelding.
8. Als het wachtwoord korter is dan 5 tekens, wordt er een foutmelding weergegeven.
9. Als het wachtwoord niet minimaal 1 cijfer bevat wordt er een foutmelding weergegeven.
10. Als de wachtwoorden niet overheen komen met elkaar, wordt een melding weergegeven.
11. Als de gebruiker een wachtwoord invoert wordt het wachtwoord afgeschermd.
12. Als registratie formulier niet volledig is ingevuld, kan formulier niet verzonden worden.
13. Als de gebruiker zich geregistreerd heeft wordt hij doorgestuurd naar de Sign-In pagina.

Sign In pagina functionele eisen

14. Als de gebruiker geregistreerd is kan de gebruiker via Sign-In pagina inloggen.
15. Als de gebruikersnaam en het wachtwoord niet overeen komen bij sign-in pagina kan er niet worden ingelogd.

Sign In pagina niet functionele eisen

16. Als de gebruiker geen gebruikersnaam heeft ingevuld bij het sign-in formulier, kan niet worden ingelogd.
17. Als de gebruiker bij sign-in pagina een verkeerde gebruikersnaam en/of wachtwoord invoert, wordt een foutmelding weergegeven.
18. Gebruiker blijft ingelogd als de pagina gerefreshed wordt.
19. Als de gebruiker ingelogd is worden er geen sign-in/sign-up links weergegeven in het navigatiemenu
20. Als gebruiker ingelogd is wordt de sign out optie weergegeven in het navigatiemenu.

Upload pagina functionele eisen

21. Als de gebruiker is ingelogd kan hij/zij een quote uploaden.
22. Upload pagina invoervelden dient volledig ingevuld te worden, anders kan het formulier niet verzonden worden.

Upload pagina niet functionele eisen

23. De upload pagina wordt weergegeven in het navigatiemenu als gebruiker is ingelogd anders niet.
24. Bij het uploaden mag het maximale aantal karakters in het tekstveld niet overschreden worden.
25. Als er meer karakters zijn ingevuld dan is toegestaan wordt er een foutmelding weergegeven.
26. Bij het dropdown menu van de upload pagina dient ook verplicht een member naam gekozen te worden, door wie de quote gezegd is.
27. In de upload pagina, als er geen naam gekozen is, wordt een foutmelding weergegeven bij het dropdown menu.
28. In de upload pagina, als "source" wordt leeg gelaten, wordt er een foutmelding weergegeven.

29. Het “Source” veld heeft een maximaal aantal karakters dat niet overschreden mag worden.

Quote pagina functionele eisen

30. Gebruiker krijgt een random quote te zien op de quote pagina.

Quote pagina niet functionele eisen

31. De zojuist geupload quote wordt bovenaan de quote pagina weergegeven.

Quote Game pagina functionele eisen

32. Gebruiker kan een spel spelen waarin geraden wordt van wie welke quote is.
33. Gebruiker krijgt een quote te zien, zonder de naam de persoon die het gezegd heeft.
34. Gebruiker ziet persoon namen wie het quote gezegd heeft als antwoord.
35. Als antwoord gekozen wordt, wordt een melding weergegeven of het juist of onjuist is.
36. Bij onjuiste antwoord wordt correcte antwoord weergegeven.
37. En bij correcte antwoord krijg je ook melding dat het antwoord goed is.
38. Als de gebruiker juiste antwoord kiest krijgt een hij/zij een punt.
39. Gebruiker krijgt volgende vraag nadat quiz beantwoord is.
40. Als gebruiker een antwoord heeft gekozen, kan het antwoord niet meer veranderd worden.

Quote Game pagina niet functionele eisen

41. Gebruiker krijgt bij het Quote spel automatisch na 5 sec volgende vraag te zien als de vraag is beantwoord.
42. Bij het Quote spel wordt het totale aantal antwoorden bijgehouden.
43. Bij correcte antwoord op een quizvraag, wordt melding in het groen weergegeven.

Navigatiemenu functionele eisen

- 44. Er is een navigatiemenu met links naar gewenste pagina's.

Navigatiemenu niet functionele eisen

- 45. Het navigatiemenu kan open en dicht, zodat het niet in de weg zit.
- 46. Als gebruiker hooft over navigatiemenu wordt de kleur veranderd naar roze.
- 47. Via navigatiemenu kunnen andere pagina's bezocht worden (m.u.v. de upload pagina, daarvoor dient ingelogd te zijn).

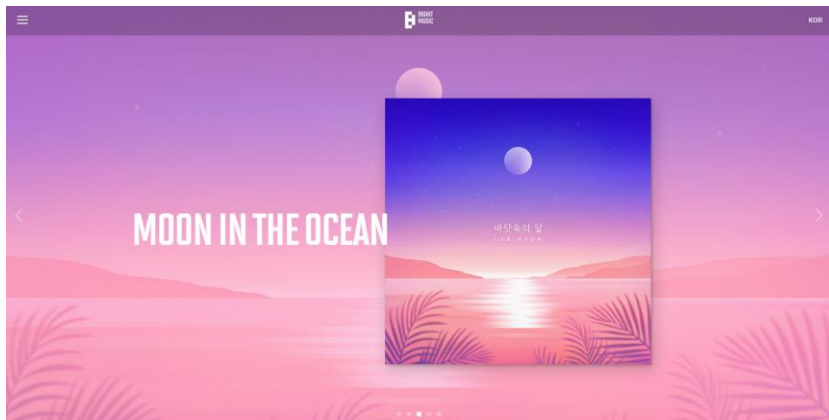
Overige niet functionele eisen

- 48. De webapplicatie is ook via smartphone, tablet en iPad etc. benaderbaar.
- 49. Alle foutmeldingen worden in rood weergegeven.
- 50. De foutmeldingen verdwijnen als alle invoervelden juist zijn ingevuld.

Inspiratie bronnen

Mijn verzameling van 3 verschillende inspiratiebronnen en screenshots.

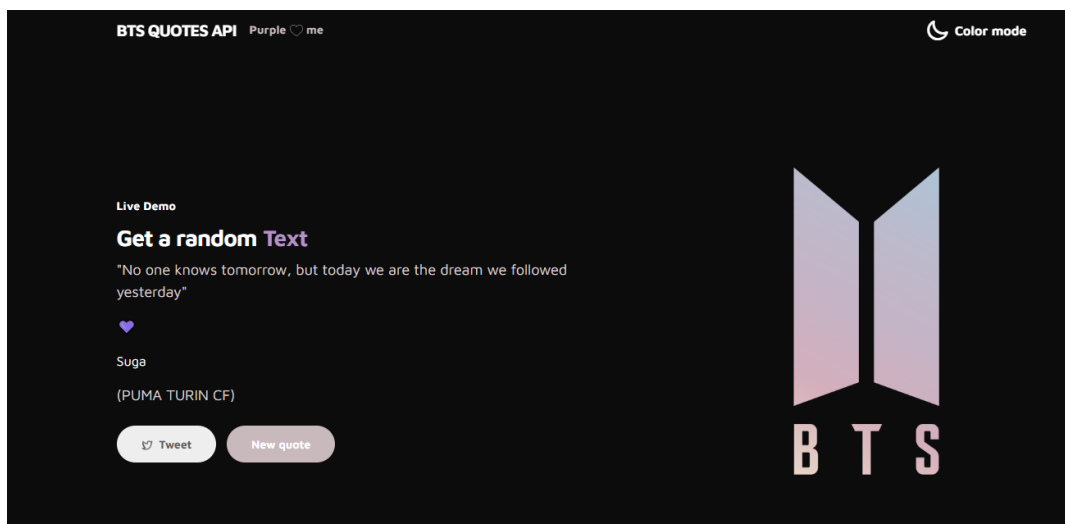
<https://ibighit.com/>



<https://github.com/>



<https://bts-quotes-api.herokuapp.com/>



4 Use case tabellen

4 use case tabellen: één voor authenticatie (registreren of inloggen) en drie voor zelfbedachte functionaliteit.

Use Case ID	1
Use Case Name	Registreren-Inloggen
Actor	Gebruiker
Trigger	Gebruiker komt op registratie pagina
Preconditie	Gebruiker wil registreren om te kunnen inloggen.
Main succes scenario	<ol style="list-style-type: none">1. Gebruiker klikt op registratie link2. Registratie formulier wordt getoond3. Gebruiker voert de gebruikersnaam, e-mailadres en wachtwoord in4. Klikt op registratie button om te bevestigen5. Gebruiker word doorgestuurd naar login pagina <p>Op login pagina:</p> <ol style="list-style-type: none">6. Voert gebruiksnaam en de wachtwoord in7. Klikt op inloggen8. Gebruiker is ingelogd9. Wordt doorgestuurd naar upload pagina
Postconditie	Gebruiker is geregistreerd en ingelogd

Alt. Scenario: Emailadres ontbreekt	<ol style="list-style-type: none"> 1. Gebruiker klikt op registratie link 2. Registratie formulier wordt getoond 3. Gebruiker voert de gebruikersnaam en wachtwoord in. 4. Klikt op registratie button om te bevestigen 5. Gebruiker krijgt een foutmelding te zien.
Postconditie	De gebruiker kan in de foutmelding lezen dat hij/zij ook een emailadres moet invullen. De gebruiker kan de missende informatie aanvullen en het formulier nogmaals opsturen.
Alt. Scenario: Emailadres bestaat al	<ol style="list-style-type: none"> 1. Gebruiker klikt op registratie scherm 2. Registratie formulier wordt getoond 3. Gebruiker voert de gebruikersnaam & e-mailadres en wachtwoord in 4. Klikt op registratie button om te bevestigen 5. Gebruiker krijgt een foutmelding te zien.
Postconditie	De gebruiker kan in de foutmelding lezen dat het ingevoerde emailadres al bestaat. De gebruiker kan emailveld opnieuw invullen met andere emailadres.
Alt. Scenario: Gebruikersnaam bestaat al	<ol style="list-style-type: none"> 1. Gebruiker klikt op registratie scherm 2. Registratie formulier word getoond 3. Gebruiker voert de gebruikersnaam & e-mailadres en wachtwoord in 4. Klikt op registratie button om te bevestigen 5. Gebruiker krijgt een foutmelding te zien.
Postconditie	De gebruiker kan in de foutmelding lezen dat de ingevoerde gebruikersnaam al bestaat. De gebruiker kan bij gebruikersnaam veld opnieuw andere gebruikersnaam invullen.
Alt. Scenario: Gebruikersnaam of wachtwoord verkeerd	<p>Op login pagina:</p> <ol style="list-style-type: none"> 6. Voert gebruiksnaam en de wachtwoord in 7. Klikt op inloggen 8. Gebruiker krijgt een foutmelding te zien
Postconditie	<p>De gebruiker kan in de foutmelding lezen dat de ingevoerde gebruikersnaam of wachtwoord onjuist is.</p> <p>De gebruiker kan gebruikersnaam en wachtwoord opnieuw invullen.</p>

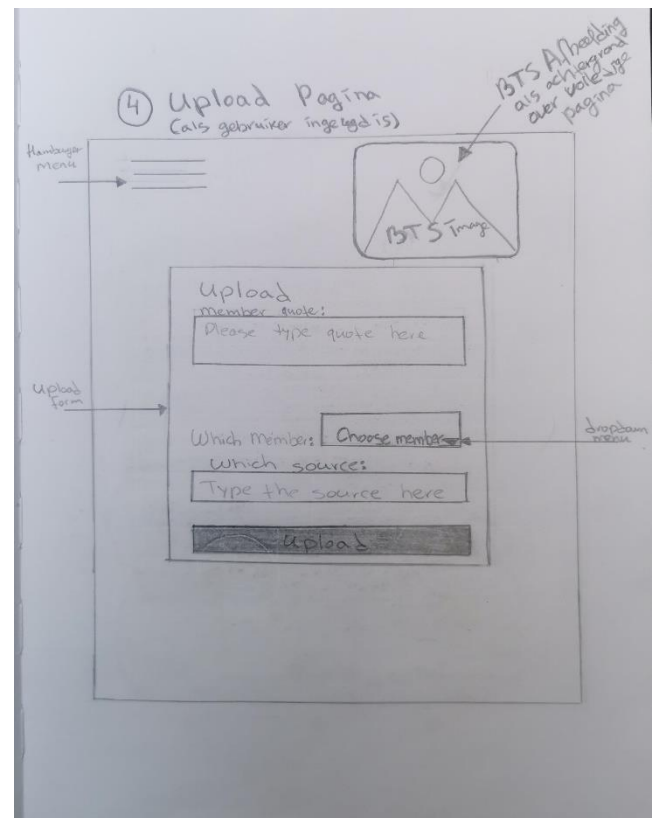
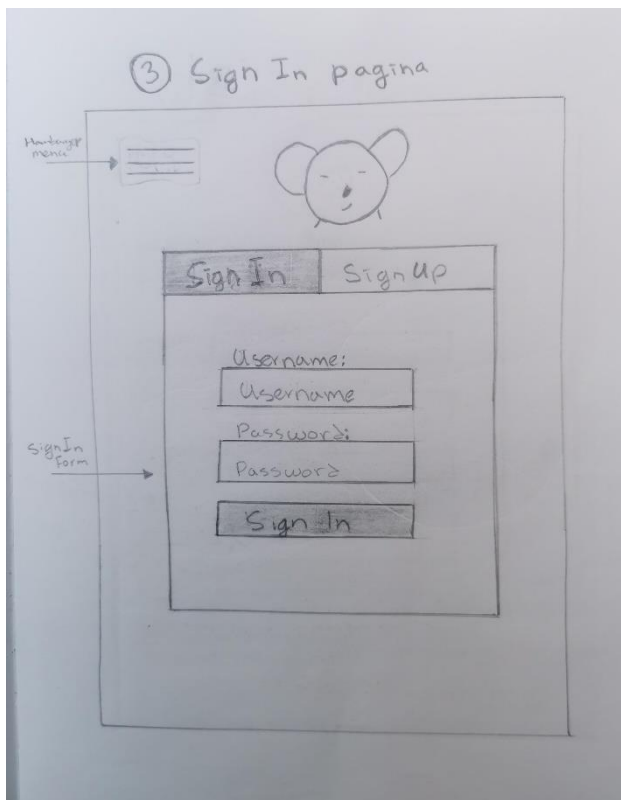
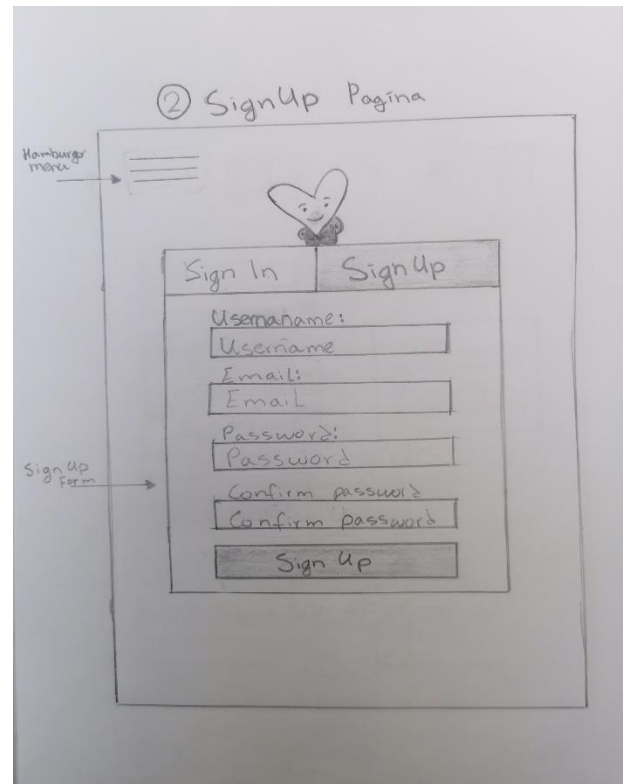
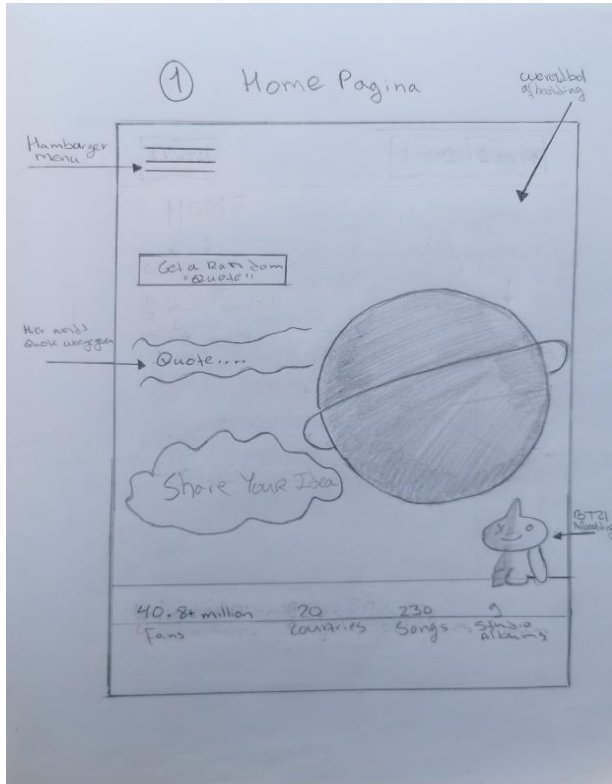
Use Case ID	2
Use Case Name	Quote Uploaden
Actor	Gebruiker
Trigger	Gebruiker klikt op upload pagina
Preconditie	Gebruiker moet ingelogd zijn
Main succes scenario	<ol style="list-style-type: none"> 1. De gebruiker voert de quote in het tekstveld 2. En kiest van wie de quote is uit de options 3. Voert source in 4. Klikkt op upload om te versturen
Postconditie	Word doorgestuurd naar quote pagina, toegevoegde quote staat bovenaan de lijst.
Alt. Scenario Source veld is leeg	<ol style="list-style-type: none"> 1. Voert quote in het tekstveld 2. Kiest optie door wie gezegd is 4. Klikkt op upload om te versturen 5. Gebruiker krijgt een foutmelding te zien
Postconditie	De gebruiker kan in de foutmelding lezen dat hij/zij ook de "source" in moet invullen. De gebruiker kan de missende informatie aanvullen en het formulier nogmaals opsturen.

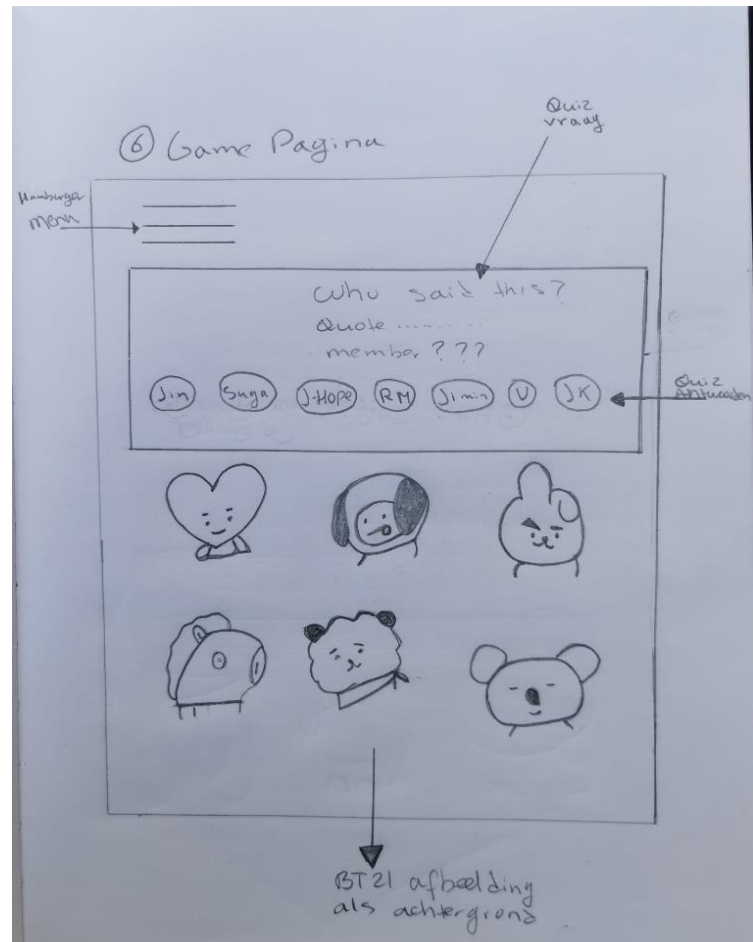
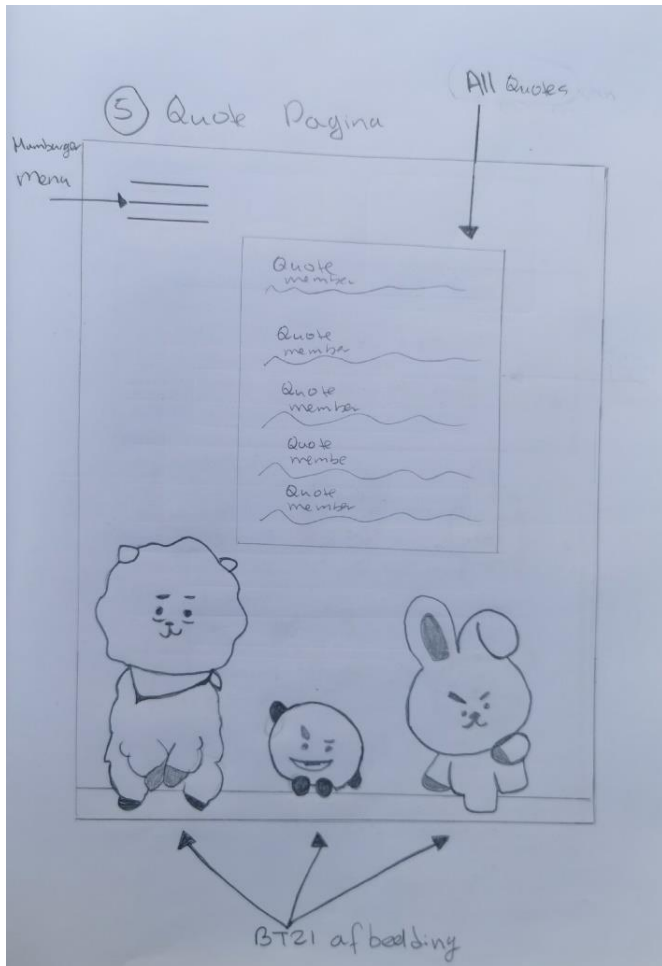
Use Case ID	3
Use Case Name	Quote pagina
Actor	Gebruiker
Trigger	Gebruiker klikt op de quote pagina
Preconditie	Gebruiker wil het overzicht van alle quotes zien
Main succes scenario	<ol style="list-style-type: none"> 1. Gebruiker klikt op quote scherm 2. Krijgt alle quotes te zien
Postconditie	Alle quotes wordt weergegeven
Alt. Scenario API stuurt geen data terug of is niet beschikbaar	<ol style="list-style-type: none"> 1. Gebruiker klikt op quote scherm 2. Krijgt een foutmelding te zien.
Postconditie	De gebruiker kan in de foutmelding lezen dat de API nu niet beschikbaar is, en het verzoek om het later nogmaals te proberen.

Use Case ID	4
Use Case Name	Quote Game
Actor	Gebruiker
Trigger	Gebruiker klikt op de quote game pagina
Preconditie	Gebruiker wil quote game spelen
Main succes scenario	1. Vraag wordt weergegeven 2. Gebruiker kiest een antwoord 1. Antwoord is juist 2. Wordt volgende vraag weergegeven
Postconditie	De gebruiker krijgt punten bij elk juist antwoord.
Alt. Scenario Quiz antwoord is onjuist	1. Vraag wordt weergegeven 2. Gebruiker kiest een antwoord 1. Antwoord is onjuist 2. Wordt volgende vraag weergegeven
Postconditie	Als antwoord fout is, worden er geen punten gegeven en het goede antwoord wordt gegeven.

Wireframes

Mijn wireframes van 5 pagina's met titels en beschrijvingen.





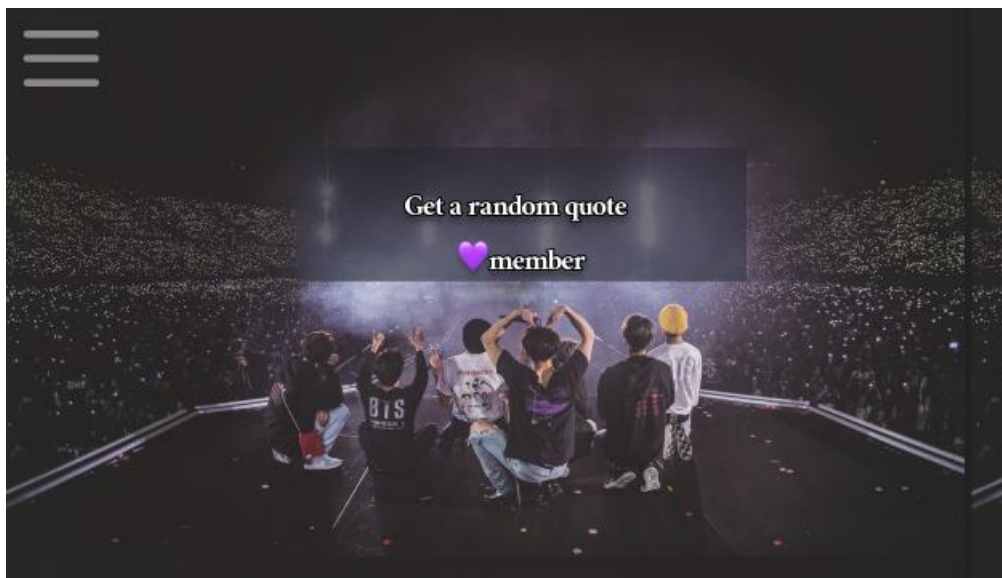
Schermontwerpen

Mijn schermontwerpen screenshot van 6 schermen met titels en beschrijvingen.

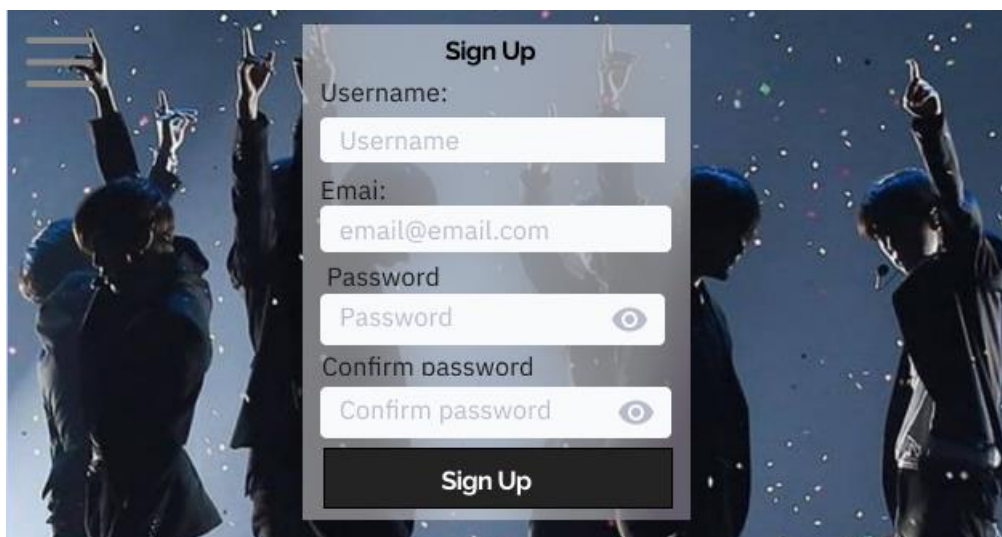
Link naar mijn prototype in Figma:

<https://www.figma.com/file/wFPdjaCkNdMrYdw2LPPrZN1/Prototype?node-id=510%3A365>

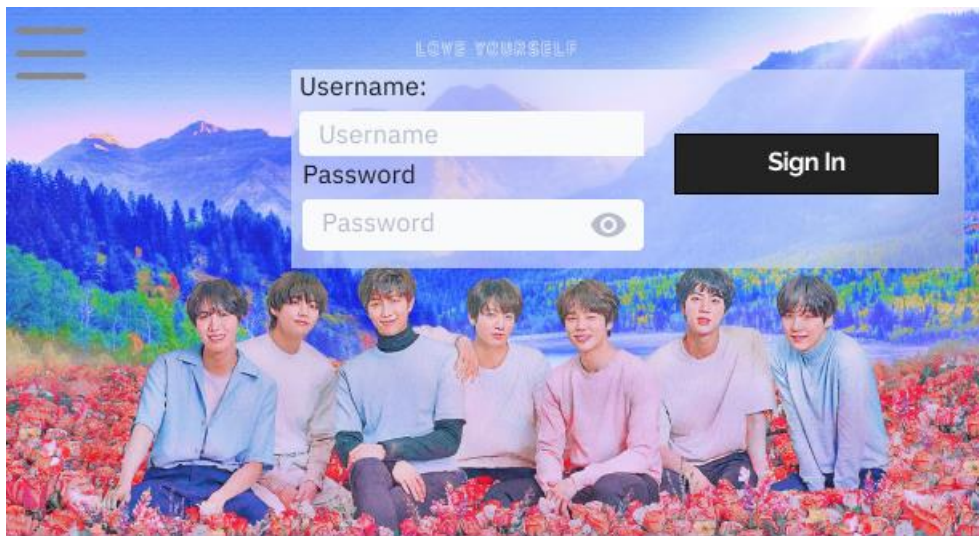
Dit is “Home” pagina hierin wordt willekeurige quotes van alle 7 members op willekeurige volgorde weergegeven met vermelding van de member naam. En bij het refreshen van de pagina worden nieuwe quotes opgehaald.



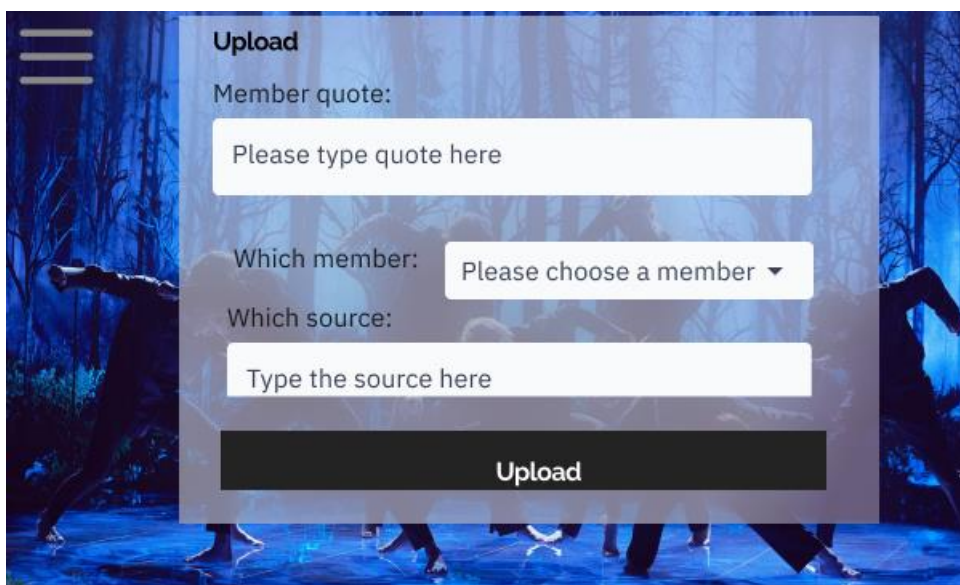
Dit is “Sign Up” pagina waar de gebruiker kan registreren om quote te kunnen plaatsen.



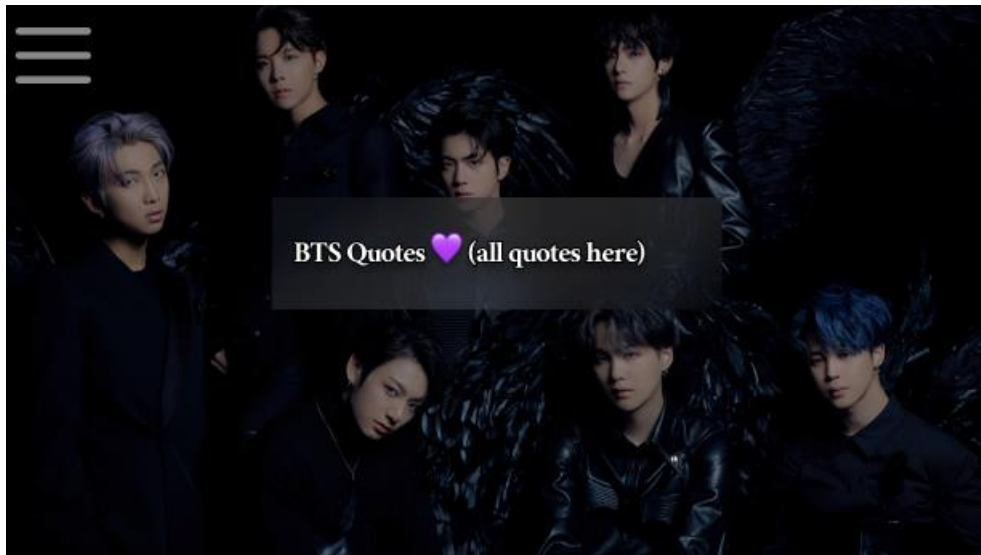
Dit is “Sign In” pagina, na registreren wordt de gebruiker doorgelinkt naar deze pagina.



Dit is de “Upload” pagina, als de gebruiker registreert kan hij/zij een quote die door een BTS member geschreven is plaatsen en de naam van de member in het dropdown menu kiezen, en vervolgens ook de bron intikken en dan op de “Upload” button klikken om de quote te uploaden.



Dit is de "Quote" pagina; hierin worden alle quotes van alle 7 members weergegeven.



Dit is de "Game" pagina, hierin kan de gebruiker raden door welke BTS member quotes gezegd werden door de naam van de member te kiezen. Als het goed wordt geraden dan wordt de score bijgehouden en vermeld dat het de juiste keuze was. En als er een verkeerd antwoord gegeven wordt dan verschijnt een melding en word de juiste member naam weergegeven. En bij het refreshen wordt scoren opnieuw op 0 gezet.



Verantwoordingsdocument

10 Beargumenteerde technische keuzes

- 1) Om de data op te halen heb ik voor de “Bangtang API” gekozen, omdat deze API ondersteund om willekeurige data op te halen, data te posten en alle data weergegeven. Dat sluit goed aan bij mijn project.
- 2) Voor de paginalayout heb ik uiteindelijk gekozen voor andere afbeeldingen en GIFs dan in mijn prototype in Figma omdat dat beter uitkwam qua styling in CSS en er mooier uit ziet.
- 3) Voor het ophalen van de data heb ik gebruik gemaakt van de Axios library. O.a. voor het maken van http requests in Node.js. Het is lichtgewicht, en goed geschikt voor het werken met JSON, en we hebben er al lessen over bijgewoond.
- 4) Voor de meeste formulieren heb ik gebruik gemaakt van react-hook-form library. Dit zorgt voor een betere gebruikerservaring omdat je makkelijk validatie regels kan opstellen en foutmeldingen kan weergegeven aan de gebruiker. Je hoeft er zelf weinig code voor te schrijven want er zit al veel functionaliteit in de library.
- 5) Ik heb gebruik gemaakt van de react-router-dom library omdat dat een makkelijke manier is om gebruikers te kunnen laten navigeren door de verschillende pagina's van de webapplicatie. Als de gebruiker naar een andere pagina gaat hoeft niet hele pagina te refreshen of opnieuw geladen te worden. Dit zorgt voor betere performance van je app.
- 6) Voor styling heb ik gebruik gemaakt van flexbox. Dankzij flexbox kan ik zonder veel moeite een responsive layout maken, waarbij ik de controle hou over de afmetingen van de elementen.
- 7) Om mijn webapplicatie online te zetten heb ik gebruik gemaakt van Netlify, omdat je als je één keer dit hebt geconfigureerd in Netlify je gemakkelijk kan pushen naar Github en automatisch deployen.
- 8) Voor de overzichtelijkheid heb ik gekozen voor een mappenstructuur indeling tussen components, context, img en pages. Elementen die vaker terugkomen zijn in components gezet.
- 9) Ik heb gebruik gemaakt van useNavigate hook van react-router om de gebruiker te laten doorsturen naar de volgende pagina, als bijv inloggen gelukt is. Dit zorgt voor een betere gebruikerservaring.
- 10) Bij het spel formulier heb ik geen gebruik gemaakt van react-hook-form, omdat bij het spel geen validatie regels nodig waren. En omdat ik de logica zelf moet schrijven, wanneer mensen een punt kregen.

Limitaties

- De quotes worden tijdelijk opgeslagen in de API van een derde.
- De gebruikers worden tijdelijk opgeslagen in de API van een derde.
- Er kunnen enkel quotes geüpload worden, in de toekomst zou ik graag willen dat gebruikers de mogelijkheid hebben om bijv. MP3 bestanden en plaatjes te uploaden.

Doorontwikkelingen

Als er meer tijd was dan had ik graag meer features aan mijn project willen toevoegen zoals:

- Liken van quotes door de gebruikers
- Home pagina uitbreiden
- Animaties bij de spelpagina
- Een eigen API maken

Reflectie

Door deze eindopdracht te maken heb ik geleerd om telkens kleine stapje te zetten. Elke dag iets doen is beter en zorgt voor minder stress. Ook heb ik geleerd wat vandaag af kan niet voor morgen laten (dus niet uitstellen).

Wat mij geholpen heeft was naast het volgen van de lessen om ook lessen nogmaals terug te kijken en Edhub terug te lezen. Ik heb gewerkt met de voorbeelden van de Edhub opdrachten en heb deze zelf aangepast. Ook heb ik video's op YouTube bekeken met uitleg.

Ik vond het prettig om me aan een volgorde te houden. Ik begon met HTML&CSS en heb dat later omgezet naar React. Toen heb ik elementen die meerdere keren gebruikt werken omgezet naar components. Pas aan het eind heb ik de interactieve delen en het ophalen van data uit de API toegevoegd.

Wat er beter kan:

In het vervolg wil ik naast mijn main project ook een oefenproject maken.

Ook wil ik foutmeldingen/errors beter leren lezen.

Ik wil mezelf meer tijd geven om stof beter te beheersen en ik wil meer oefenen met kleine stukjes code.