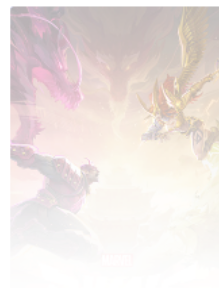
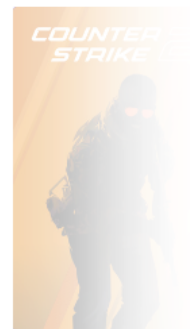
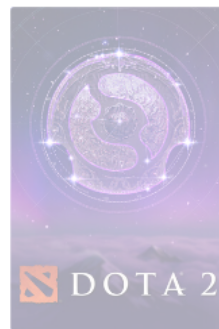
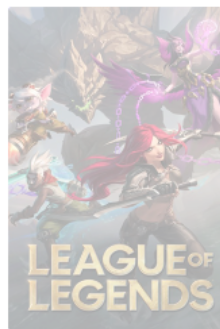
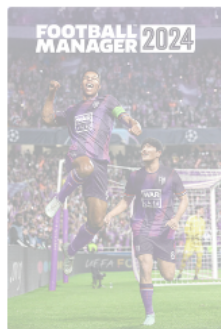
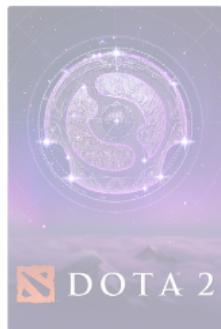


Bracket **hub** Produktrapport



Titelblad

Information om produktrapport

Deltager:	Julian Greger
Projektnavn:	Svendeprøve - BracketHub
Projektperiode:	15/09/2025 - 03/10/2025 (Aflevering)
Skole:	Aarhus Tech, Hasselager Allé 2.
Hold:	25hs37dat6pv1_1, - Data/It

Indholdsfortegnelse

Titelblad	1
Indholdsfortegnelse	2
1 Introduktion	3
2 Læsevejledning	3
3 Produktbeskrivelse	3
4 Kravspecifikation	4
4.1 Beskrivelse	4
4.2 Målgruppe	4
4.3 Funktionalitet	4
4.4 Frontend (Brugerflade - UI)	4
4.5 Backend (API & Database)	4
5 Arkitektur	5
5.1 Hjemmeside	5
5.2 API	5
5.3 Database	5
5.3.1 SQL	5
6 Design	6
6.1 Mock-ups	6
6.2 Designprincip - KISS	7
6.3 Farvepalette	7
6.4 Design elementer	8
6.4.1 UI	8
6.4.2 UI Font	8
6.4.3 Ikoner - FontAwesome	8
6.4.4 Produktets endelige design	9
7 Implementering af produkt	9
7.1 Krav for implementering	9
7.2 Implementering af hjemmeside	10
7.2.1 Hjemmeside indstilling	10
7.2.2 Hjemmeside API reference indstilling	10
7.3 Implementering af API	10
7.3.1 CORS indstilling	11
7.3.1 Database indstilling	11
7.4 Implementering af Database	11
9 Konklusion	11
10 Bilag	12
Mock-ups	13

1 Introduktion

Denne rapport vil forklare og fremvise det der er med til at udgøre produkets frontend såvel som backend. Rapporten vil gå igennem de teknologier og designvalg, der er taget i forbindelse med udseendet og funktionaliteten af produktet. Udover det vil den også forklare om de enkelte designelementer der er blevet brugt i produktet,

2 Læsevejledning

Projektet består af 2 dele, en processrapport og en produktrapport. Denne sektion vil give et overblik over hvordan man skal læse de 2 rapporter samt hvilken rækkefølge der vil passe bedst til læserens præferencer.

Rapporterne er skrevet med udgangspunkt i at læseren minimum har lidt viden og forståelse inden for programmering, webudvikling og webdesign samt relevante termer.

Det anbefales at man læser processrapporten først hvis man interesserer sig for det tekniske, såvel som processen, da processrapporten er med til at forklare mere i dybden vedrørende tanker, valg af problemer.

Er man mere interesseret i designet af produktet så burde man læse produktrapporten først da den er med til at forklare hvad produktet går ud på såvel som hvordan forskellige elementer hører sammen og ser ud.

Forklaring af jævnlige brugte termer:

Frontend og **UI**: Brugerflade (Det man ser når man bruger produktet)

Backend: Alt bag brugerfladen (Kode, services, API osv. som en bruger ikke vil se)

3 Produktbeskrivelse

Produktet er en hjemmeside der er designet til at hjælpe med at afholde turneringer nemmere. Dette kan være turneringer online eller i lokale sammenhæng. Hjemmesiden er designet til at man nemt kan oprette og administrere turneringer, såvel som at give brugere/deltagere et nemt overblik over diverse turneringer for mange forskellige spil.

Hjemmesiden snakker sammen med en simpel backend (API - Application Programming Interface), som interagerer med en SQL (Structured Query Language) database for at oprette, opdatere og læse data.

- Frontend'en (UI) er en hjemmeside lavet i Blazor WebAssembly (klient-side).
- Backend'en består af et OpenAPI/Swagger API, som snakker med en SQL Server database via frameworket; Entity Framework Core (EFCore).

4 Kravspecifikation

Denne kravspecifikation er baseret på det mindst levedygtige produkt som der skal laves.

4.1 Beskrivelse

Der skal laves et turneringssystem, som skal være tilgængelig via internettet (Web). Systemet skal være tilgængelig og brugbar for både arrangører af turneringer såvel som medlemmer og brugere.

4.2 Målgruppe

Produktet er som udgangspunkt designet til dem som spiller computerspil enten online eller i lokale sammenhæng. Dette kan være alle fra konkurrencedygtige spillere og organisationer såvel som dag til dag spillere.

4.3 Funktionalitet

Systemet skal være tilgængeligt på nettet uanset enhedstype; PC, Mobil, Tablet osv. I systemet skal det være muligt at oprette turneringer ud fra et givent spil. Det skal være muligt at give turneringer et navn, dato og beskrivelse, og det skal være nemt at tilrette hvis man senere vil ændre på informationen

Man skal kunne oprette en simpel konto som kan bruges til at tilmelde til turneringer. Det skal være muligt for eksisterende medlemmer og deltagere såvel som nye personer at tilgå systemet og nemt finde eksisterende og relevante turneringer ud fra spiltype.

4.4 Frontend (Brugerflade - UI)

Brugerfladen skal være simpel, ikke give/vise for meget information men stadig fyldig nok til at man har og får det man skal bruge. Med systemets brugerflade skal det være muligt at:

- Oprette en bruger.
- Vælge hvilket spil man er interesseret i.
- Se turneringer ud fra et givent spil.
- Tilgå en valgt turnering.
- Se information vedr. en given turnering.
- Tilmelde en bruger til turneringer.
- Oprette turneringer.
- Redigere turneringer.
- Opdatere kampe i turneringer.

4.5 Backend (API & Database)

Der skal laves en backend som består af en API og en database. API'et skal være lavet til at snakke med den tilhængende brugerflade (UI). API'et skal have endpoints som indeholder alt brugerflade har brug for, for at kunne udføre dens arbejde i forbindelse med brugerinput. Ud fra hvad API'et har brug for, så skal den kunne snakke med en database for at gemme, læse og opdatere data.

5 Arkitektur

Produktet består af 1 hjemmeside, 1 API og 1 database som er lavet/skrevet ved hjælp af Visual Studio (IDE - Integrated Development Environment), hvilket er et program lavet af Microsoft designet til at udvikle computerprogrammer, hjemmesider, webapplikationer, webservices og apps til telefoner og tablets.

5.1 Hjemmeside

Hjemmesiden er skrevet i Blazor WASM (Webassembly) som består af; HTML, CSS, JavaScript og C#. HTML og CSS er det der styrer det visuelle, JavaScript styrer opførslen af det visuelle og C# er det kodesprog som bliver kompileret om til JavaScript kode, som så er med til at styre hele fundamentet af hjemmesiden.

Hjemmesiden indeholder alt logikken og visuelle elementer som produktets brugerflade gør brug af.

5.2 API

API'et er skrevet i C# og består af endpoints som hjemmesiden kan spørge for at oprette, læse og opdatere data som ligger i databasen. API'et er lavet til kun at snakke med hjemmesiden og ikke andre systemer, enheder.

5.3 Database

Databasen indeholder data som API'et indsætter, læser og opdaterer. Databasens struktur er opbygget via EFCore (Entity Framework Core) som er et C# bibliotek som sørger for at daten der læses og skrives via API'et matcher databasens opbyggelse. I produktet er der blevet brugt Microsoft's SQL Server, som er en af de mere simple systemer at få sat op på en server der kører med et Microsoft operativsystem.

5.3.1 SQL

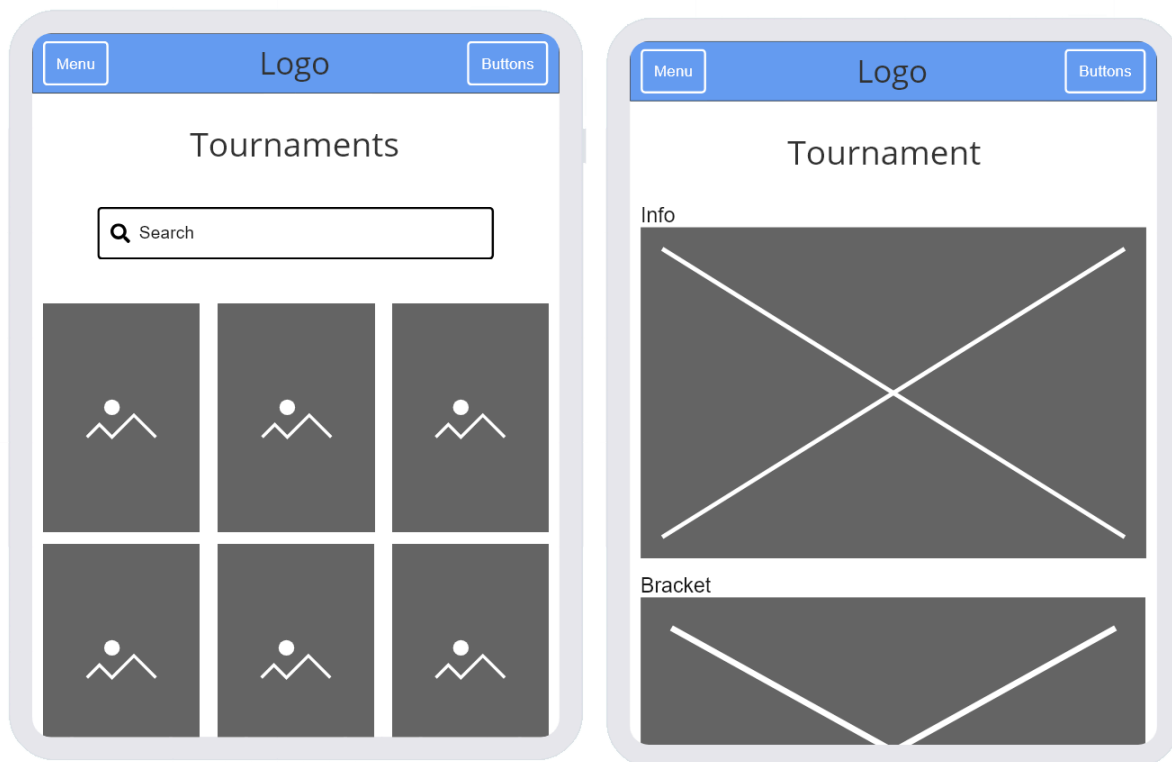
Databasens type er SQL, da dataen der bliver brugt i produktet er struktureret og indeholder begrænsninger for hvad der kan ligge af data. Strukturen er begrænset for at sørge for at der f.eks. ikke er dublinger af kampe og turneringer.

6 Design

6.1 Mock-ups

Mock-ups er med til at sørge for at der er en klar retning for udseendet af de sider der skal laves til hjemmesiden. Mock-ups er generelt simple i deres design for at hurtigt kunne skabe noget visuelt som man så kan bruge til at vurdere, om det er en retning man skal bevæge sig hen imod. Man kan nemt tilrette mock-ups indtil man får noget man vil og kan gå videre med.

Produktet består af flere sider, så der er blevet lavet et par mock-ups til de vigtigste sider.



Billede af forside og tournament side - Bilag 2.

De 2 mock-ups vist, viser 2 sider af hjemmesiden i et tabletformat. De indeholder begge 2 top-baren hvor det er muligt at navigere til f.eks. hjemmesidens menu oppe i venstre hjørne. Hjemmesidens logo er så det, der er vist i midten af top-baren og så er der ekstra funktioner oppe i højre hjørne. Top-baren er en af de elementer som går igen på alle sider og undersider på hjemmesiden.

Det generelle design af hjemmesiden sider vil være baseret på de 2 mock-ups vist herover (bilag 2). Flere mockups i bilag sektion: "Mock-ups"

Side 1 er forsiden hvor det skal være muligt at se populære spil samt turneringer. Udover det så er der et søgefelt hvor man kan få lov til at søge efter specifikke spil samt turneringer.

Side 2 er af en turneringsside hvor man kan se generel information om turneringen samt se hvordan spil brackets ser ud. Oprettelse og redigering af en turnering vil foregå via samme design og layout, bare med flere funktioner og knapper.

6.2 Designprincip - KISS

KISS står for “Keep It Simple, Stupid” og har været en af de principper, der er blevet gjort brug af i produktets Frontend (UI). Produktet indeholder billeder som er brugt som ikoner, bannere og baggrunde. Billederne er brugt i relation til specifikke spil, så de ikke forstyrrer det simple og brugbare underliggende design af hjemmesiden.

Designet er lavet, så det trækker interessen mod diverse spil og turneringer. Ideen med designet er at give brugere et nemt overblik og adgang til de områder de leder efter; det kunne f.eks. være et specifikt spil eller turnering som de selv er interesseret i.

6.3 Farvepalette

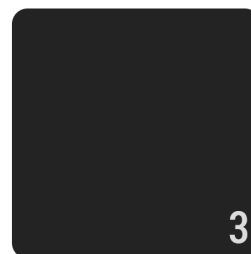
Produktet kører med en farvepalette af nedtonet og mørke farver. De mørke farver strækker sig fra en mørkeblå til 2 nuancer af mørkegrå. Udover de mørke farver er der også 1 lidt nedtonet hvid brugt til tekst på hjemmesiden.



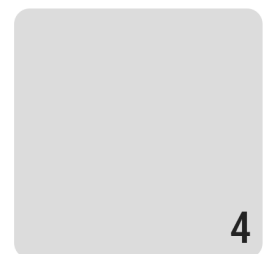
HEX #000d35
RGB 0,13,53



HEX #3c3c3c
RGB 60,60,60



HEX #252525
RGB 37,37,37



HEX #e0e0e0
RGB 224,224,224

1. Farve 1 er brugt til baggrundselementer såsom; sidebaggrunden og top-baren (headeren). Farven er med til at skabe et blå farvetema på hele hjemmesiden, da farven er synlig på en stor del af brugerfladen.
2. Farve 2 er til UI elementer som er placeret oven på baggrunden. Farven er en mørkere nuance af grå/sort for at være synlig oven på baggrund uden at være for lys i forhold til temaet.
3. Farve 3 er brugt ved knapper og inputfelter. Farven er lidt mørkere end farve 2 for at gøre det synligt at det er UI elementer man kan interagere med.
4. Den sidste farve er en nedtonet hvid som er brugt til tekst på hjemmesiden. Farven er nedtonet for ikke at være iøjneskærende i forhold til de andre mørkere UI elementer.

6.4 Design elementer

6.4.1 UI



Små UI elementer brugt på hjemmesiden

Hjemmesidens UI er bygget på et par enkelte visuelle elementer. Hovedsageligt en simpel afrundet kasse. Denne kasse er brugt til at holde de visuelle elementer på hjemmesiden ensartet. Kassen kan indeholde andre elementer, såsom ikoner, tekst og billeder. Knapper består f.eks. af en kasse med enten et ikon eller tekst i, imens der kan være andre statiske elementer som kun viser tekst.

Hjemmesiden har fokus på hvilket område af spil man er inde på. Så dvs. at der kan være et "tema" for specifikke spil som gør at man nemt og hurtigt kan se hvilket område man er inde under. Hvert spil har f.eks. et ikon som bliver vist i top-baren som kan bruges til at differentiere spil sider fra hinanden.

6.4.2 UI Font

Counter-Strike: Source 1337 CS2 Major - 2025

Link til fonten kan findes i Processrapportens kildeliste: "Roboto font (Google font)"

Fonten brugt i produktet hedder "Roboto". Fonten er gratis og kan hentes fra Google's liste over fonts. Fonten er meget tæt på 2 font standarder som forskellige browsere bruger; "Arial" og "Sans-serif", dog er "Roboto" mere moderne med mere lige linjer, hvor f.eks. "Arial" har nogle skrå afslutninger på nogle bogstaver. I de fleste tilfælde så er "Roboto" en mere neutral font i forhold til hvad man ser blive brugt i flere spil, både i deres UI men også i deres titelbilleder.

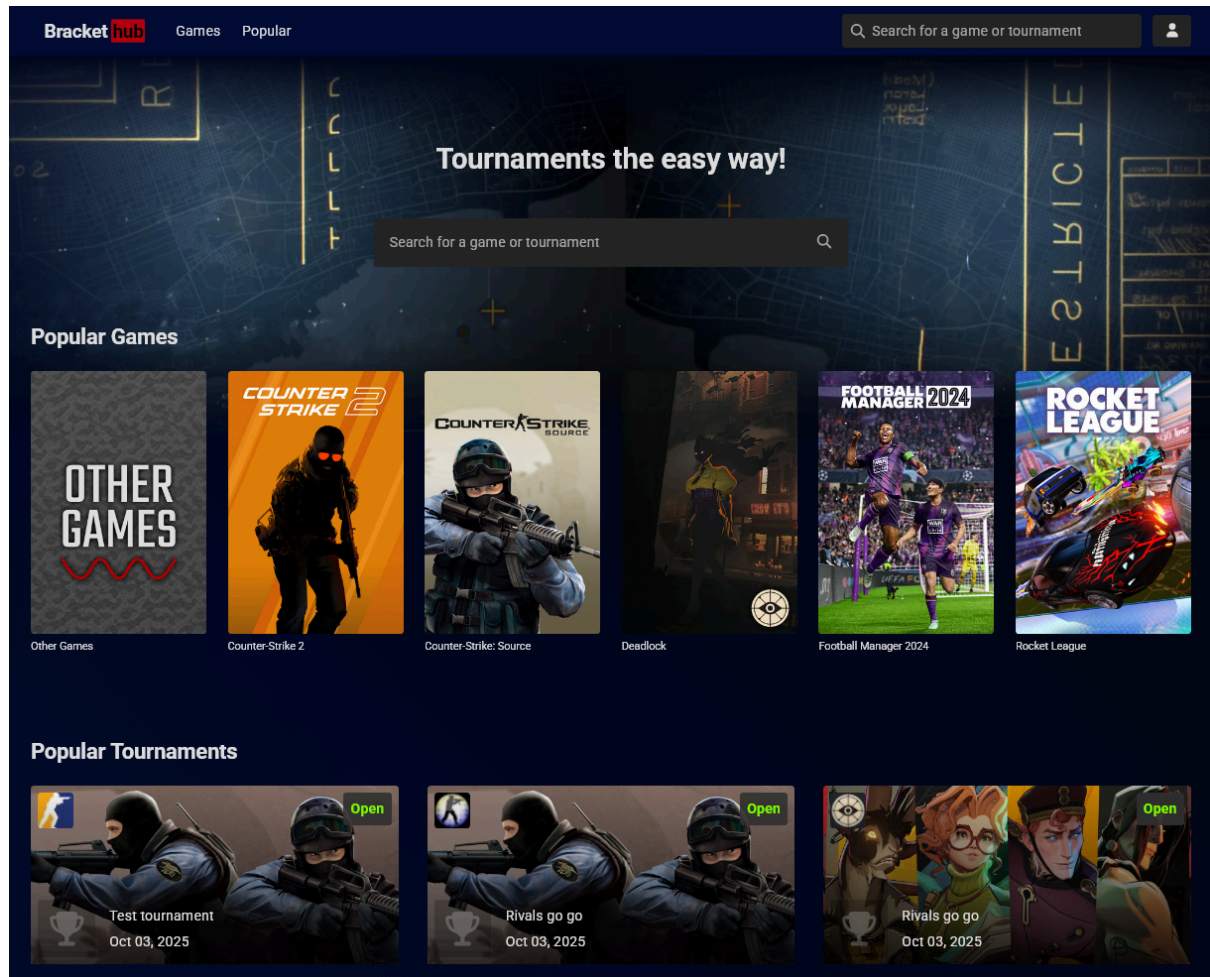
6.4.3 Ikoner - FontAwesome

Produktet gør brug af FontAwesome for nogle simple ikoner. FontAwesome er nem at integrere og bruge. De har mange gode ikoner som man kan bruge, både i form af gratis ikoner såvel som ikoner man kan betale for. I dette produkt er hele deres gratis katalog blevet integreret, men man kan sagtens nøjes med at vælge, hvilke specifikke ikoner man gerne vil bruge via deres hjemmeside. Så det vil sige at man kan få lov til at lave små og skræddersyet ikon-pakker som ikke fylder så meget og som er et perfekt egnet til ens hjemmeside.

6.4.4 Produktets endelige design

Med mock-ups, farvepalette og design elementer kunne hjemmesiden sættes sammen til et færdigt design til produktets hjemmeside.

Billedet herunder er fra hjemmesidens forside hvor man kan se et par spil såvel som nogle turneringer. Billedet er taget fra en PC størrelse i forhold til opløsning op skærmstørrelse.



Billede taget fra det færdige produkt (Bilag 1.)

7 Implementering af produkt

7.1 Krav for implementering

1. Windows 10 eller nyere.
2. Projektfiler
3. (Valgfri) SQL Server
4. (Valgfri) Visual Studio IDE

UI'en kan startes i "Debug" via Visual Studio uden at være afhængig af API'et, men dette gør at der vil blive vist statisk data som ligger i projektets kode, såvel som nogle funktioner som ville være ufunktionelle (funktioner som er designet til sende og modtage data fra en database).

For at implementere hjemmesidedelen af produktet anbefales det at man som minimum har en forståelse for hosting af hjemmesider og webapplikationer. For at hjemmesiden kan implementeres på en korrekt måde, så anbefales det at man bruger en Microsoft IIS server, da det er det produktet som udgangspunkt er bygget med.

7.2 Implementering af hjemmeside

Produktet er som udgangspunkt ikke lavet til at blive kørt lokalt, da det er sat op til at referere filer, filerstier og systemer som ikke vil være de samme steder lokalt som f.eks. i et IIS miljø.

Filerne fra mappen "BracketHub Release - Web" er dem som skal smides ind i IIS serverens mappe system, hvorefter man så kan tilføje den som hjemmeside via IIS Server systemet. Hjemmesiden er lavet som klient-side så alle filerne der skal bruges bliver hentet ned til klienters browsere, så selve hjemmesiden kræver ikke meget setup for at fungere.

7.2.1 Hjemmeside indstilling

Ud fra hvor man placerer hjemmesidens filer så skal man opdatere "index.html" -filens "<base>" værdi til at passe med hjemmesidens relative sti.

7.2.2 Hjemmeside API reference instilling

Når API'et er sat op og man kender til dens adresse såvel som dens sti offset, så kan hjemmesidens "appsettings.json" fil blive opdateret med korrekt data.

Så ligge API'et på adressen; "<https://thisisatest.com>", så er det det man skal bruge som værdien til "APIBaseAddress".

Ud fra hvor API ligger på IIS'en, så skal "APIOffset" værdier udfyldes med den korrekte relative sti til selve API endpointet. Så ligger API i en undermappe "/testapi" så skal man bruge dette som værdien i "APIOffset".

Med de 2 værdier udfyldt vil hjemmesiden f.eks. lave API kald mod "<https://thisisatest.com/testapi>".

7.3 Implementering af API

Produktets API er bygget og skal bruges som webapplikation. Filerne der ligger i mappen "BracketHub Release - API" kan ligesom hjemmesiden smides ind i IIS'en, i en undermappe hvis det er på samme IIS server som hjemmesiden. Når filerne så ligger inde i IIS'ens mappe-system, kan man tilføje API'et som en "AspNetCore" webapplikation i IIS'en.

7.3.1 CORS indstilling

API'et har som udgangspunkt CORS aktiveret så den kun kan snakke med hjemmesiden. Så når hjemmesiden er oppe at køre, kan man opdatere "appsettings.json" med adressen som hjemmesiden bruger under "CorsBaseAddress".

7.3.1 Database indstilling

Når databasen er oppe og køre og man har fået fat i databasens "connectionstring", kan man opdatere API'ets "appsettings.json" under "DefaultConnection" med værdien. Dette er med til at API'et kan snakke med databasen via EFCore.

7.4 Implementering af Database

Når der så er blevet sat en SQL Server op, kan man implementere arkitekturen/skemaet som skal bruges af API'et.

Via projektet er der blevet lavet et "script" som kan bruges til at køre en SQL migration på databasen for at få de korrekte tabeller og værdier. Scriptet ligger i projektfilerne og hedder "BracketHubDB-Script.sql". Dette script indeholder alt databasen skal bruge for at lave alle de nødvendige tabeller.

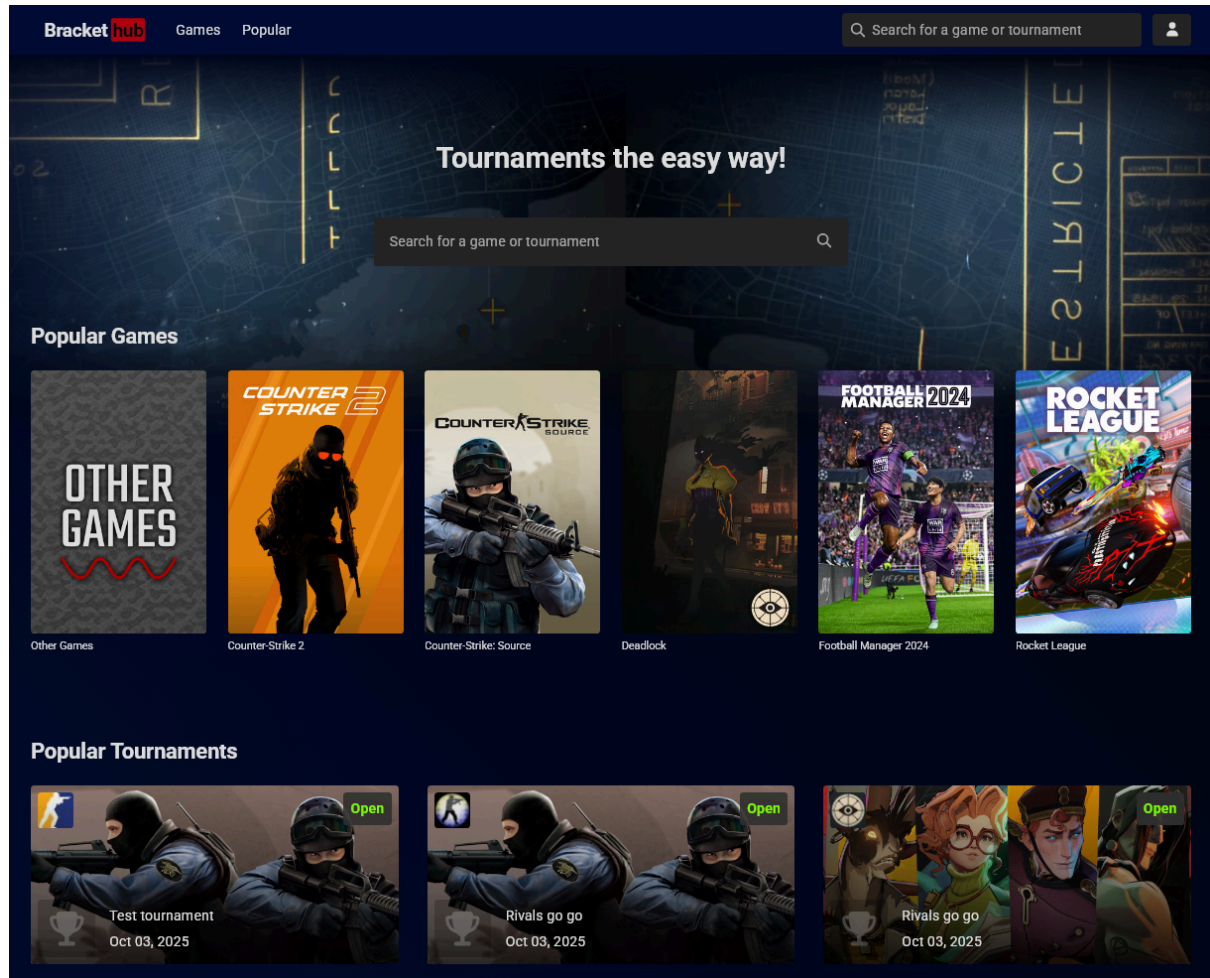
9 Konklusion

Produktet er som efterspurgt; et online turneringssystem som kan bruges af både arrangører og brugere. Produktet er fuldt funktionelt i forhold til kravene der er sat. Produktet kunne godt have haft mere funktionalitet for at forbedre brugeroplevelsen på hjemmesiden, men som prototype så er resultatet et godt udgangspunkt for et reelt produkt, som kan gøre turneringer nemmere at arrangere og bruge i et produktionsmiljø.

10 Bilag

Bilag 1.

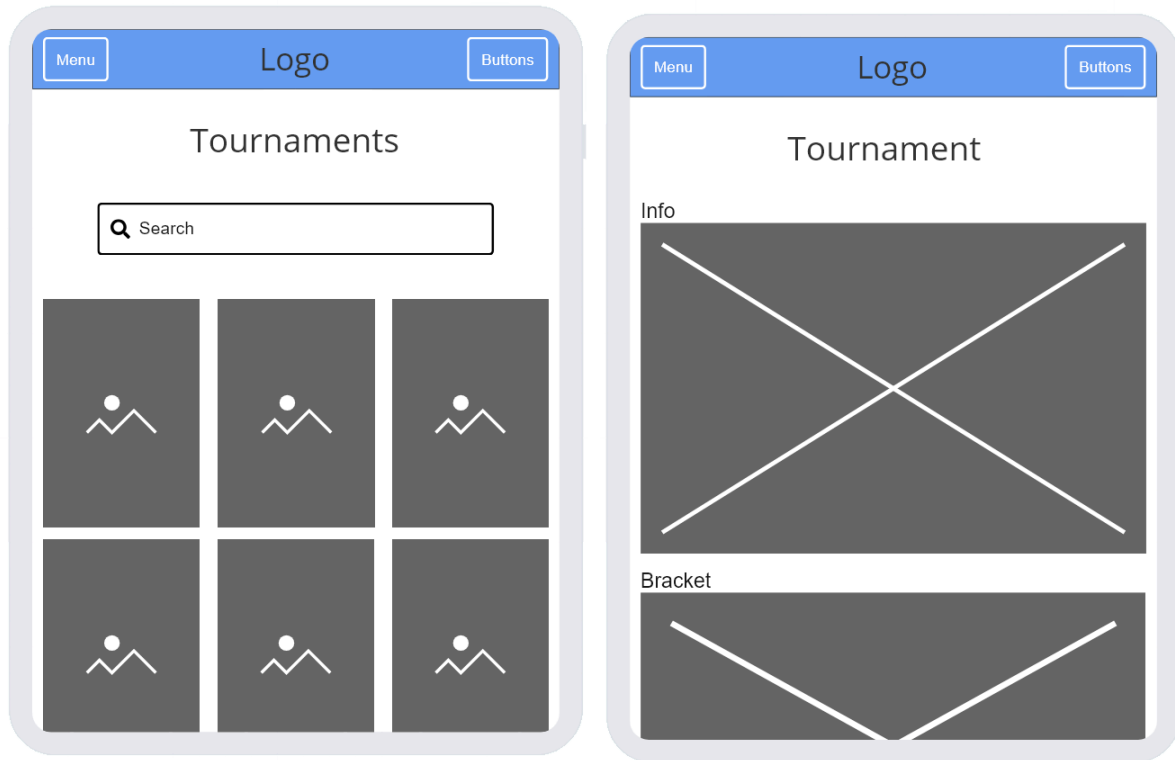
Screenshot af færdige produkt hjemmeside (Forsiden)



Mock-ups

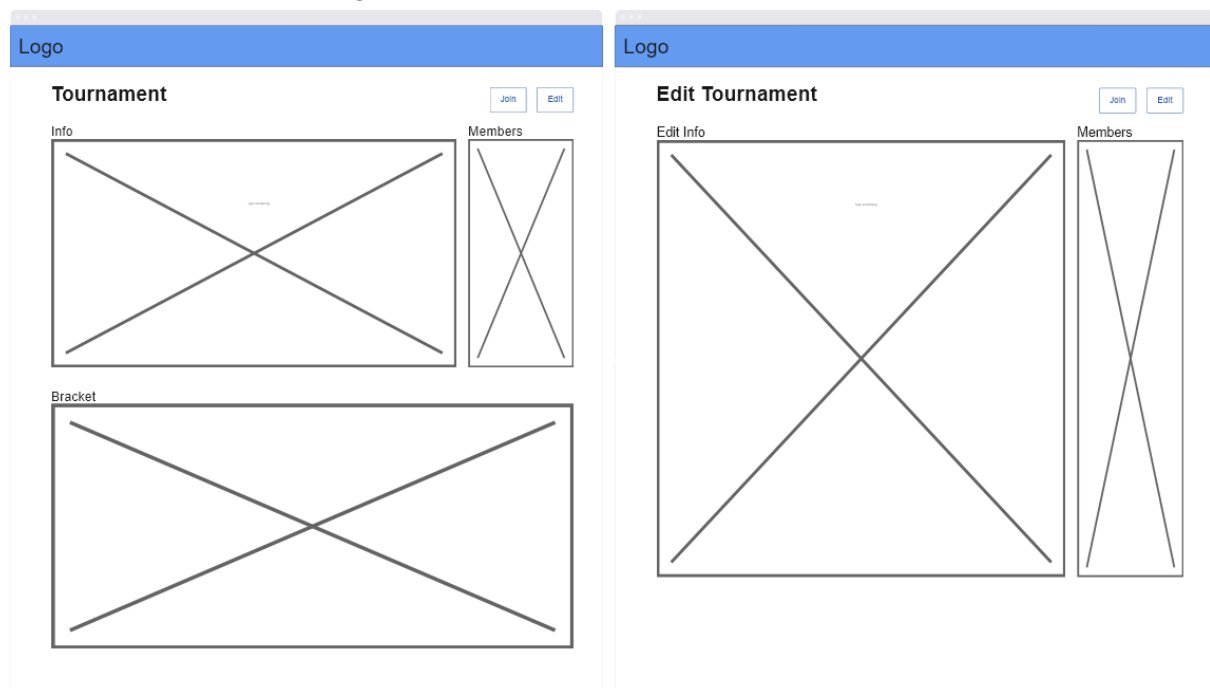
Bilag 2.

Mock-ups af 2 forsiden såvel som turneringssiden (mobil/tablet format).



Bilag 3.

Mock-ups af Tournament og Edit Tournament siderne (PC Format).



Bilag 4.

Mock-ups af Tournaments og Games siderne (PC Format)

