STŘEDOŠKOLSKÁ ODBORNÁ ČINNOST

Obor č. 18: Informatika

Discit – webová stránka a webová aplikace pro sport discgolf

Tomáš Terč
Pardubický kraj

Pardubice, 2024

STŘEDOŠKOLSKÁ ODBORNÁ ČINNOST

Obor č. 18: Informatika

Discit – webová stránka a webová aplikace pro sport discgolf

Discit – web page and web application for sport discgolf

Autoři: Tomáš Terč

Prohlášení

Prohlašuji, že jsem svou práci SOČ vypracoval/a samostatně a použil/a jsem pouze prameny a literaturu uvedené v seznamu bibliografických záznamů.

Prohlašuji, že tištěná verze a elektronická verze soutěžní práce SOČ jsou shodné.

Nemám závažný důvod proti zpřístupňování této práce v souladu se zákonem č. 121/2000 Sb., o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon) ve znění pozdějších předpisů.

V Pardubice dne 7.2.2024 ……

Tomáš Terč

Poděkování

Upřímně děkuji Ing. Tomáš Pešek, za studijní materiál a odborné vedení při zpracovávání maturitního projektu.

Anotace

Cílem projektu bylo zvýšit povědomí o sportu discgolfu formou jednoduchého, uživatelský přijatelného a moderního online nástroje, který by byl přístupný a srozumitelný všem a sloužil by pro hráče, kteří o sportu dosud nic nevědí, ale i pro hráče, kteří už sport hrají. Discgolf je momentálně ne až tak rozšířený sport zde v Evropě, tudíž není moc nástrojů pro hru.
K vytvoření projektu jsem čerpal z těch pár nástrojů, které existují. Nejvíce inspirace jsem, ale čerpal z aplikace jménem UDisc, která slouží pro usnadnění průběhu hry. Ale aplikace mi připadala složitá na pochopení a také mi tam chyběl obsah pro nové hráče. Takže jsem se snažil udělat webovou stránku a aplikaci co nejvíce přátelskou pro uživatele a také obsahující obsah pro nové hráče, aby se mohlo množství hráčů discgolfu zvýšit. Projekt je momentálně dostupný na všech zařízeních a stránku může navštívit kdokoliv kdo má zájem si zahrát nebo jenom něco zjistit o tomto sportu.

Klíčová slova

Programování; React; Firebase; MUI; Webová stránka

Annotation

The aim of the project was to increase awareness of the sport of discgolf through an online tool accessible and understandable to everyone, catering to both players who know nothing about the sport and those who already play. Discgolf is currently not widely spread here in Europe, so there aren't many tools available for the game. I drew inspiration from a few existing tools, but primarily from an app called UDisc, which facilitates gameplay. However, I found the app somewhat complex and lacking content for new players. Therefore, I aimed to create a website and web app that are user-friendly and include content for newcomers, with the goal of increasing the number of discgolf players. The project is currently available on all devices, and anyone interested in playing or learning about this sport can visit the website.

Keywords

Programming; React; Firebase; MUI; Web

Obsah

1. Úvod 1
2. Teoretická část 2
2.1. Způsob zpracování 2
2.2. Pojmy 2
2.3. Zvolené technologie 3
2.4. Prozkoumání existujících nástrojů 5
3. Praktická část 6
3.1. Nápad 6
3.2. Začátek práce 6
3.3. Architektura 7
3.4. Frontend 7
3.5. Backend 9
3.6. API 9
3.7. Návrh databáze 11
3.8. Autentifikace 12
3.9. Systém přátel 13
3.10. Optimalizování využívání databáze 13
3.11. Problém s mapou hřišť 13
3.12. Rozložení webové stránky 14
3.13. Webová aplikace 15
4. Závěr 16

1. Úvod

Discgolf je relativně mladý a stále se rozvíjející sport, který vznikl v Americe během 60. let minulého století díky Edovi Headrickovi, který je považován za zakladatele discgolfu. Headrick si tento titul zasloužil díky svým dvěma patentům na frisbee disky a ocelové koše. Kombinace těchto prvků položila základy sportu discgolf, jak ho známe dnes.

I přesto, že discgolf získává na popularitě, stále není dostatek online nástrojů souvisejících s tímto sportem. Z tohoto důvodu jsem se rozhodl zvolit toto téma jako hlavní zaměření mého maturitního projektu s cílem šířit povědomí o tomto sportu mezi více lidí. Inspiraci jsem čerpal převážně z již existující mobilní aplikace UDisc, která usnadňuje průběh hry pro hráče, a webových stránek organizace PDGA, což je profesionální asociace discgolfistů.

Během zkoumání těchto online nástrojů, UDisc a PDGA, jsem zjistil, že mobilní aplikace je poměrně složitá na ovládání a může být matoucí pro nové nebo nezkušené hráče discgolfu. Na druhou stranu webové stránky PDGA poskytují užitečné informace o discgolfu, ale chybí jim interaktivní prvky. Mým záměrem je tyto nedostatky vyřešit vytvořením online nástroje, který kombinuje oba tyto nástroje. *[1][2]*

2. Teoretická část

2.1. Způsob zpracování

Tento projekt jsem se rozhodl pojmout jako kombinaci webové stránky a webové aplikace. Webová stránka bude sloužit jako vstupní brána pro uživatele, kde nejenom prezentuje samotný projekt, ale také jeho účel a obsah webové aplikace. Tato část bude zároveň sloužit jako cenný zdroj informací pro nové hráče, kteří ještě neznají discgolf, nebo nemají dostatek informací o tom, jak tento sport hrát.

Webová aplikace pak představuje interaktivní část projektu, kterou uživatelé mohou využít při své hře. Bude poskytovat různé nástroje a funkce, které pomohou hráčům vylepšit své dovednosti a zároveň zlepšit celkový zážitek z hry. Tímto spojením webové stránky a aplikace se tento projekt stává přístupným pro všechny hráče, ať už jsou zkušení discgolfisti nebo nováčci.

Díky jednoduchému ovládání bude projekt vhodný pro všechny hráče, bez ohledu na jejich technologické znalosti. Navíc, zpřístupnění informací o discgolfu pro nové hráče na webové stránce přispěje k rozšíření povědomí o tomto sportu a zvýší jeho dostupnost pro více lidí. Celkově tento projekt má přinést nový online nástroj pro discgolf a poskytnout zdroj informací pro milovníky tohoto sportu.

2.2. Pojmy

2.2.1. Frontend

Frontend je část webové stránky, kterou uživatel vidí a může s ní interagovat. Celý kód z frontendu se odehrává na straně uživatele. To zahrnuje veškerý vizuální obsah jako jsou obrázky, text, formuláře atd. Technologie používané pro frontend jsou HTML pro strukturu obsahu stránky, CSS pro stylování této struktury a jejích prvků a JavaScript pro dynamiku stránky. *[3]*

2.2.2. Backend

Backend je část webové stránky, kterou naopak od frontendu uživatel nemůže vidět. Tato část pomocí API umožňuje frontendu interagovat s daty a se serverem. Nejpopulárnější technologie používané pro backend jsou JavaScript, PHP, Python atd. *[4][30]*

2.2.3. API

API (Application Programming Interface) je rozhraní díky, kterému mohou frontend a backend mezi sebou komunikovat a zasílat data. *[5]*

2.2.4. UML Use Case Diagram

UML Use Case Diagram je grafická pomůcka pro znázornění softwarových systémů. Druhů UML Diagramů je mnoho, ale druh, který jsem zvolil má pomoct znázornit funkci jednotlivých částí systému a interakci, kterou mohou aktéři (uživatelé) provádět. *[6]*

2.2.5. Hashing

Hashing je proces, který transformuje citlivá data pomocí hashovací funkce. Tato funkce vezme vstupní hodnotu a vytvoří z ní unikátní řetězec pevné délky, který je reprezentací této hodnoty. Nejběžněji používané hashovací funkce zahrnují SHA-256, SHA-3, MD5 a bcrypt. [29*]*

Pro zvýšení bezpečnosti hashe se často používá metoda nazývaná „hashing with salt“. Tato metoda zahrnuje přidání náhodně generované soli ke vstupnímu heslu před samotným hashováním. Sůl je náhodný řetězec znaků, který se přidá k heslu před hashováním, což zaručuje, že i pro stejná vstupní hesla bude vygenerován odlišný hash. Tímto způsobem je znemožněno použití technik jako je útok hrubou silou nebo použití předpočítaných tabulek (rainbow tables) k odhalení hesel.

Výsledkem je dlouhý řetězec znaků, který reprezentuje původní heslo, ale je mnohem bezpečnější pro ukládání v databázi nebo přenos přes síť. Použití soli spolu s hashovací funkcí zvyšuje odolnost proti útokům na hesla a poskytuje vyšší úroveň bezpečnosti pro uživatele. *[7]*

2.2.6. Lobby

Toto označení se používá většinou ve hrách a označuje něco jako „čekárnu“, kde hráči čekají, než hra započne.

2.3. Zvolené technologie

2.3.1. React

React je knihovna, která slouží k vývoji webových aplikací a stránek. Dalo by se říct, že funguje na konceptu „rozděl a panuj“. Stránka je rozdělena na jednotlivé komponenty, které fungují každá sama o sobě. Když uživatel přejde na jinou stránku nebo stránku obnoví, tak komponenty, které jsou na obou stránkách V dnešní době je poměrně populární mezi vývojáři. Byl vyvinut společností Meta. *[8]*

2.3.2. Next.js

Next.js je open-source framework, který funguje ruku v ruce s knihovnou React. Přináší mnoho výhod jako předčasné vykreslování stránek na straně serveru (Server-side rendering), snadný vývoj a konfigurace a mnoho dalších. Byl vyvinut společností Vercel. *[9]*

2.3.3. MUI

MUI je open-source knihovna sloužící pro stylování stránky a opět jde ruku v ruce s knihovnou React. Jeho asi největší výhodou je sada už předdefinovaných komponent, které můžete jednoduše využít na svém webu. *[10]*

2.3.4. Firebase

Firebase je oblíbená platforma pro správu backendu. Poskytuje mnoho nástrojů, které vám usnadní práci v backendu. Já jsem používal převážně Cloud Firestore jako databázi pro můj projekt a vbudovanou autentifikaci ve Firebasu. Platforma byla vyvinuta společností Google. *[11]*

2.3.5. GraphQL

GraphQL je open-source dotazovací jazyk a prostředí pro programování sloužící k vývoji API. Umožňuje uživateli specifikovat požadavky na data, která chce získat, a server vrací pouze ta data, která uživatel vyžádal. Byl vyvinut společností Meta. *[12]*

2.3.6. TypeScript

Typescript je programovací jazyk postaven na bázi JavaScriptu. Jeho výhodou je to, že nutí uživatele, aby všechno v kódu bylo otypované. Díky tomu se potom uživatel vyvaruje chybám spojeným s typy. *[13]*

2.3.7. Vercel

Vercel je cloudová platforma, která poskytuje hosting a správu pro webové stránky a aplikace. Účelem této platformy je, aby vývojáři mohli rychle a jednoduše nasadit své webové stránky. Má jednu velkou výhodu a to, že podporuje hodně javascriptových frameworků a s nimi i jejich funkce jako je například Server Side Rendering, tudíž tato platforma je dobrým místem kam umístit tento projekt, který je postaven na frameworku Next.js. *[14]*

2.3.8. Apollo Codegen

Apollo Codegen je nástroj pro automatickou generaci TypeScript nebo JavaScript kódu podle uživatelova GraphQL schématu. Je to jeden z mnoha nástrojů z Apollo Client a byl vyvinut společností Apollo Graph. *[25][26]*

2.4. Prozkoumání existujících nástrojů

Před započetím práce na projektu je vhodný první krok, provést průzkum existujících nástrojů, které se již věnují podobnému tématu. Můj průzkum probíhal prostřednictvím internetu a obchodu Google Play. Během této analýzy jsem zjistil, že je k dispozici jen omezené množství webových stránek a aplikací zaměřených na sport discgolf. Například v obchodě Google Play je k dispozici pouze několik desítek aplikací souvisejících s discgolfem, a úspěšných webových stránek také není o moc více.

Vzhledem k malému počtu konkurenčních aplikací a webových stránek je prostor pro nové online nástroje související s tímto sportem stále veliký. To poskytuje širokou škálu možností, jakým způsobem projekt realizovat a jakými funkcionalitami jej vybavit. *[1][2]*

3. Praktická část

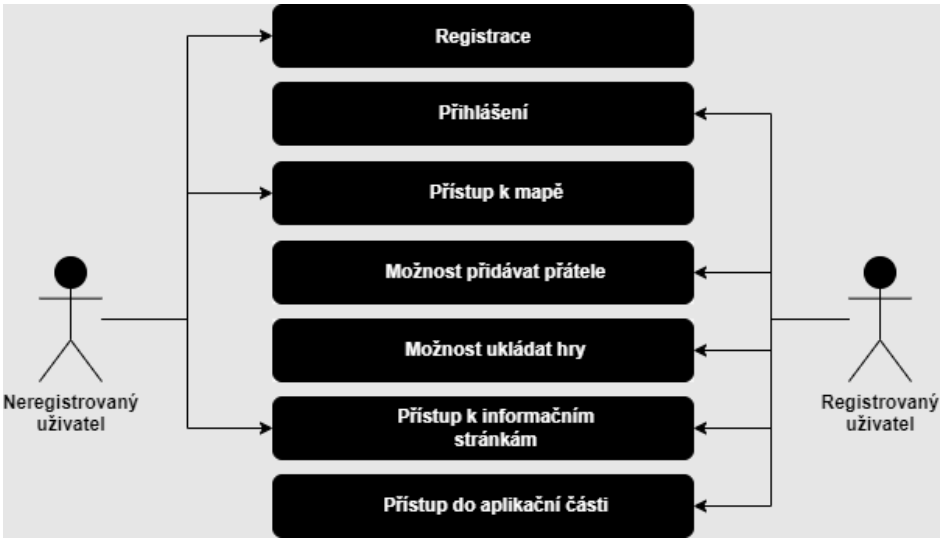
3.1. Nápad

Můj nápad na projekt spočívá v kombinaci webové stránky a webové aplikace. Webová stránka bude mít dvojí účel. Zaprvé poslouží jako informační zdroj pro nové hráče, kteří se chtějí dozvědět více o discgolfu, naučit se ho hrát a objevit zajímavosti spojené s tímto sportem. Zadruhé bude sloužit jako platforma, na které uživatelé mohou zobrazit výstupy ze webové aplikace.

Přidáním webové aplikace do stránky se uživatelům otevře možnost zábavné interakce s obsahem. Tato kombinace vytvoří prostředí, kde uživatelé najdou vše, co potřebují, na jednom místě, a kde se obě části stránky vzájemně doplňují a posilují.

3.2. Začátek práce

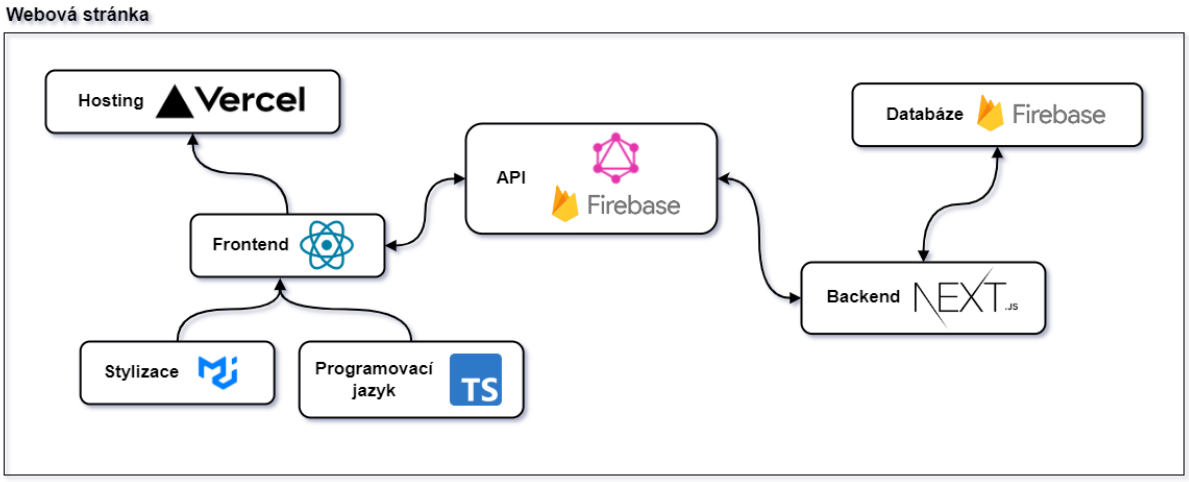
Práce na samotném projektu začala lehkým grafickým návrhem stránky a plánováním jaké rozložení stránky zvolit. Snaha byla o to, aby stránka byla co nejpřehlednější a jednoduchá na ovládání.



Obr. 1: UML Use Case Diagram

Poté zbývala už jen poslední věc před tím, než začalo kódování stránky, a to bylo vypracovat UML Use Case Diagram, který můžete vidět na Obr. 1. Během toho bylo úkolem ujasnit si, jak bude systém uživatelů fungovat, jaké druhy uživatelů budou existovat a jaké bude mít každý z nich pravomoce. *[15]*

3.3. Architektura



Obr. 2: Schéma architektury webu

Architekturu této webové aplikace, kterou můžete vidět na Obr. 2 je rozdělená na tři části: *[15]*

3.4. Frontend

Frontend mé webové stránky představuje klíčovou část uživatelského rozhraní, která interaguje s uživatelem přímo v prohlížeči. Využívám moderní technologie a nástroje, aby uživatelé měli co nejlepší uživatelskou zkušenost.

Jako hlavní framework pro vývoj frontendu jsem zvolil React. Ten umožňuje tvorbu komponentově orientovaných aplikací, což zjednodušuje správu stavu aplikace a opakované využití komponent.

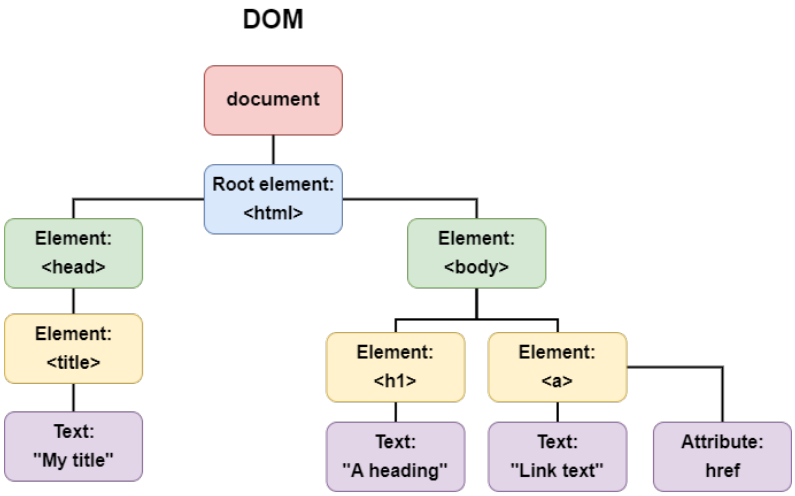
Pro psaní kódu využívám TypeScript, který přidává statické typování. Tím se zvyšuje bezpečnost, srozumitelnost a udržovatelnost kódu. TypeScript je ideální volbou pro vývoj větších a komplexnějších aplikací.

Pro stylizaci a design komponent jsem zvolil Material-UI (MUI), což je populární knihovna pro React, která poskytuje sadu již předdefinovaných komponent a stylů, které jsem v projektu využil.

Některé základní komponenty, jako jsou formuláře pro přihlášení a registraci, navigační panel a další, jsou již předdefinované v knihovně MUI a já je využívám pro zajištění konzistence a efektivity vývoje. Díky nim mohu rychle integrovat běžné prvky a soustředit se na implementaci vlastních funkcí.

Kód mé webové stránky je hostován na platformě Vercel, která poskytuje rychlé a spolehlivé nasazení aplikací. Díky integraci s verzovacím systémem GitHub mohu jednoduše aktualizovat a nasazovat nové verze mé aplikace při každé změně v kódu. Vercel také poskytuje škálovatelné a bezpečné prostředí pro provoz mé aplikace.

3.4.1. Document Object Model



Obr. 3: DOM (Document Object Model)

DOM (Document Object Model) je vyobrazení stránky pomocí strukturovaného stromu všech komponent na stránce, toto zobrazení můžete vidět na Obr. 3. Obsahuje „node“ (uzel) pro každý element, který se nachází na stránce. Jeho výhodou je přehlednost, která může vývoj stránky značně usnadnit, ale má také velkou nevýhodu a to tu, že při každém obnovením stránky tudíž obnovením celého DOMu, tak se každý element musí updatovat a tím pádem renderovat znova. To může brzdit načítání stránky a její plynulost.

A zde nastupuje React s věcí nazývanou Virtual DOM. Je to kopie normálního DOMu, takže každý element v DOMu, má svojí identickou kopii v React Virtual DOMu. Důležité je, že změny provedené ve Virtual DOMu nemají vliv na samotnou stránku, takže je manipulace s ním o dost rychlejší než s klasickým DOMem, kde se při každé změně updatuje každá komponenta, a to zahrnuje spousta složitých algoritmů a funkcí. Takže, když se provádějí změny na stránce, která využívá Virtual DOM, tak se nejdříve updatuje Virtual DOM a až poté ten klasický.

Díky tomu, když se updatuje některý element na stránce, tak vytvoří zcela nový strom ve Virtual DOMu. Tento strom je poté porovnán se stromem klasického DOMu a zaznamenají se rozdíly, které jsou nalezeny. Podle těchto změn se potom najde nejlepší způsob, jak updatovat celou stránku, takže se poté znova renderují už jen ty elementy, u kterých se našel nějaký rozdíl při porovnávaní stromů. *[15][16]*

3.5. Backend

Backend funguje skrz framework Next.js a zpracovává příkazy, které přijdou z API nebo například na žádost odesílá data z databáze přes API na frontend, kde je uživatel může vidět. Next.js také obsahuje dvě šikovné funkce, které jsou na tomto projektu využívány.

3.5.1. Server Side Rendering

Obsahuje funkci zvanou Server-Side Rendering (SSR), která zajišťuje, že se stránka a všechna data načtou na serveru, než jsou odeslána uživateli. Tato metoda výrazně zlepšuje a zpříjemňuje uživateli užívání stránky tím, že umožňuje rychlejší načítání a snižuje čas, který uživatel stráví čekáním na to, než se obsah načte. SSR také přispívá k menší zátěži při načítání stránky, což je výhodné zejména pro uživatele s méně výkonnými nebo zastaralými zařízeními. A také pro uživateli, kteří nemají přístup k rychlejšímu internetu nebo se zrovna nacházejí v oblasti s pomalejším připojením. *[17]*

3.5.2. File-routing system

File-routing system je druhou užitečnou funkcí poskytovanou Next.js, která nabízí snadný způsob, jak definovat nové cesty ke stránkám v projektu. Princip je jednoduchý: když vývojář přidá nový soubor do složky „pages“ nebo v novější verzi Next.js také do složky „app“, kterou Next.js automaticky vytvoří při vytváření nového projektu, tím vlastně definuje novou stránku. Název tohoto souboru přímo určuje URL cestu k dané stránce. Je to intuitivní a snadný způsob jak rychle nové stránky bez složitého konfigurování cest.

Dále file-routing system v Next.js může fungovat i dynamicky. To znamená, že část názvu souboru může obsahovat dynamické parametry, například identifikátor položky. Tato funkce je užitečná pro vytváření dynamických cest, které reagují na uživatelské vstupy nebo parametry URL. Díky tomu lze vytvářet stránky, které se automaticky přizpůsobují podle kontextu. Tato funkce při vývoji celkově zlepšila vytváření nových stránek a cest k nim. *[18]*

3.6. API

Pro komunikaci mezi frontendem a backendem používám GraphQL API a Firebase Firestore API. Zpracovává požadavky nebo pokyny, které přijdou z jednoho konce a odešle je do druhého.

3.6.1. GraphQL API

GraphQL API narozdíl od tradiční RESTful API, která umí jenom základní http požadavky (GET, POST, PUT, DELETE), umožňuje uživateli přesně specifikovat jaká data chce

v jediném dotazu. Toto umožňuje efektivnější komunikaci mezi oběma konci a zamezuje kolování nadbytečných dat. V projektu byl využit zejména u registrace. *[22]*

3.6.2. Apollo Codegen

Při vývoji jsem také využil Apollo generátor kódu, který mě vygeneroval z GraphQL schématu TypeScript kód. Takže by se to dalo považovat za něco jako konvertor kódu. V projektu jsou právě dva soubory „demo.graphql“ a „graphql.tsx“. První soubor je již zmíněné GraphQL schéma, které obsahuje typy a mutace využívané v projektu. Když je nová mutace nebo typ dopsána, tak stačí pomocí příkazu v terminálu spustit Codegen. Ten zkontroluje jestli je všechno správně v těchto dvou souborech. Pokud ano tak vygeneruje kód do TypeScriptového souboru. Ze kterého poté jsou typy a mutace použitelné v TypeScript kódu. *[26][27]*

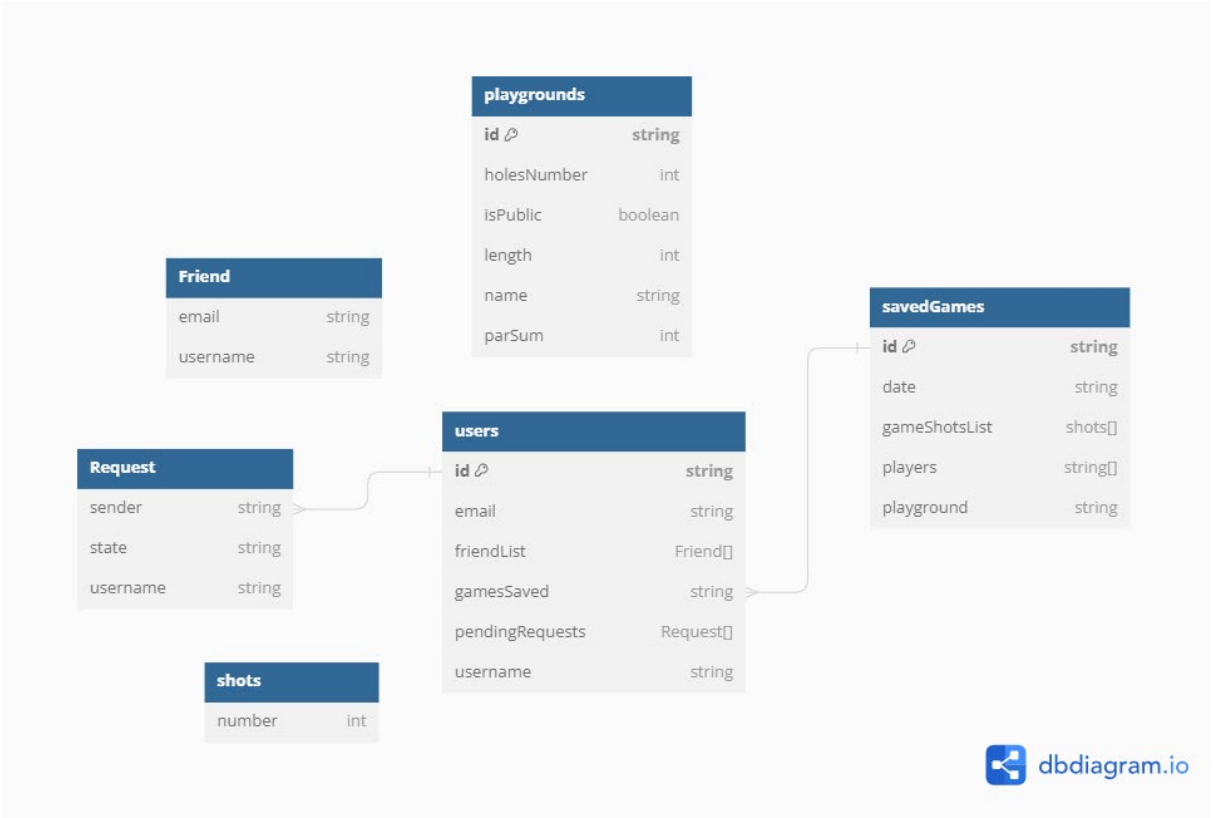
3.6.3. Firebase Firestore API

Firebase Firestore API je součástí Firebase a umožňuje interakci s Firebase databází pomocí základních CRUD operací. Navíc obsahuje i dotazovací mechanismus, který podobně jako u GraphQL API, umožňuje specifikovat požadovaná data z databáze.

Firebase dále poskytuje Firebase SDK (Software Development Kit), což je komplexní sada nástrojů a knihoven určených pro práci s různými funkcemi a službami Firebase.

Tyto dva klíčové nástroje od Firebasu byly široce využívány na většině stránek v tomto projektu. Například na stránkách s autentifikací, seznamem her, seznamem přátel a mnoha dalších částech projektu. Díky jejich funkčnosti a široké škále možností bylo možné efektivně implementovat požadovanou funkcionalitu a zajistit plynulý provoz aplikace.*[23]*

3.7. Návrh databáze



Obr. 4: Schéma databáze

Rozložení databáze, kterou můžete vidět na Obr. 4 je vytvořeno na základě tří hlavních tabulek: playgrounds, savedGames a users. *[19]*

3.7.1. playgrounds

Tabulka "playgrounds" uchovává seznam všech discgolfových hřišť v České republice a jejich parametry, které jsou využívány při zobrazení detailu nějaké z uložených her.

3.7.2. savedGames

Tabulka "savedGames" obsahuje záznamy o všech hrách, které si uživatelé uložili pro pozdější zobrazení.

3.7.3. users

Tabulka "users" obsahuje informace o všech uživatelích, kteří se zaregistrovali na stránce.

3.7.4. Speciální typy

Kromě těchto hlavních tabulek využívají databázové záznamy také speciální typy. Jsou to "Friend", "Request" a "shots". Tyto typy jsou implementovány jako tzv. "enums", což jsou speciální datové typy, u kterých lze definovat vlastní sady hodnot, které mohou nabývat.

3.8. Autentifikace

Při navštívení stránky uživatel má možnost si vytvořit účet pomocí registrace, což mu odemkne více možností, co na stránce může dělat. Všechny funkce používané při autentifikaci na stránce jsou poskytovány pomocí služby Firebase.

3.8.1. Registrace

Proces registrace začíná tím, že uživatel zadá všechny požadované informace, které jsou na stránce vyžadovány, jako je jméno, e-mailová adresa a heslo. Poté stránka provede kontrolu, zda zadané hodnoty splňují stanovené požadavky. Pokud ne, uživatel je informován prostřednictvím vyskakovacího okna. Pokud jsou všechny požadavky splněny, uživatel je zapsán do databázové tabulky "users", a zároveň je zaznamenán v seznamu uživatelů, který si udržuje Firebase. Každému uživateli je při registraci přidělen unikátní identifikační token (ID). Heslo uživatele je zabezpečeno a šifrováno pomocí metody "Hashing with salt" pro zvýšení bezpečnosti. *[20][24]*

3.8.2. Přihlášení

Při přihlašování je opět využívána funkce od Firebase. Tato funkce prochází seznam uživatelů podle zadaného e-mailu. Pokud najde shodu, porovná zadané heslo s uloženým zahashovaným heslem, které je uloženo na straně Firebase. Pro porovnání hesel se získá "salt" z uloženého hesla, který se poté spojí se zadaným heslem. Tato kombinace je opět zahashována stejnou funkcí jako při registraci. Nakonec jsou porovnány obě zahashovaná hesla a pokud se shodují, uživatel je úspěšně přihlášen do stránky. *[20][24]*

3.8.3. Obnova hesla

Obnova hesla je na stránce také možná. Pokud uživatel zapomene své heslo, může přejít na stránku určenou pro tento účel. Tam zadá e-mailovou adresu, kterou použil při registraci, a Firebase mu zašle odkaz, pomocí kterého si může uživatel heslo změnit.

3.9. Systém přátel

Stránka obsahuje také systém přátel, který umožňuje uživatelům navzájem se přidávat mezi své přátele. Tento systém je navržen tak, aby se uživatelé mohli snadno spojit a zahrát si spolu.

Pro přidání přítele uživatel použije příslušnou stránku, kde zadá jméno uživatele, kterého si přeje přidat. Po odeslání tohoto požadavku databáze zachytí tuto akci a přidá zadanému uživateli žádost o přátelství typu "Request". Tato žádost obsahuje informace o odesílateli (jméno, ID) a stav žádosti, který je v počáteční fázi označen jako "pending" (čekající).

Pokud je žádost o přátelství přijata, stav se změní na "accepted" (přijata) nebo "denied" (odmítnuta). Pokud je žádost přijata, oba uživatelé jsou automaticky přidáni do seznamu přátel druhého uživatele. Tímto způsobem se vytváří propojení mezi uživateli a umožňuje se jim spolu hrát hry na discgolf.

Tento systém přátel přispívá k budování komunity kolem discgolfu a poskytuje prostředí pro vzájemné spojení. Uživatelé mohou vytvářet nová přátelství a rozšiřovat svou síť spojení v rámci discgolfové komunity. Tím se podporuje jeho další rozvoj a popularita.

3.10. Optimalizování využívání databáze

Během vývoje projektu jsem narazil na problém související s nadměrným voláním databáze. Stránky využívaly databázi při běhu tak moc, že byl překročen limit využití a Firebase začal vracet chybu. Tato chyba zabraňovala stránce využívat databázi, dokud se chyba nespraví. Tím pádem bylo potřeba optimalizovat stránky volání databáze.

Toho bylo docíleno následovně. Před optimalizací stránka volala databázi průběžně každých několik sekund. Toto řešení však nebylo vhodné, jak už se ukázalo. Takže jsem upravil stránky, aby si z databáze načetli pokaždé, když se na stránku přijde a když se provede nějaká změna, aby se mohla zapsat.

3.11. Problém s mapou hřišť

Při průzkumu ostatních discgolfových online nástrojů, tak jsem si všiml, že většina z těchto stránek a aplikací obsahují mapu discgolfových hřišť. Takže aby se tento projekt mohl udržet s konkurencí tak jsem se rozhodl, že webová stránka také bude obsahovat interaktivní mapu s hřišti.

Byl tu jediný problém a to, že jsem doposud neměl s vytvářením něčeho takového žádné zkušenosti, a tak jsem začal průzkum na to, jak takovou interaktivní Google Mapu vytvořit. Při tomto průzkumu jsem zjistil, že Google Mapy mají svůj vlastní nástroj pro vytváření map zvaný Google My Maps a na tom je možné bezplatně si svoji mapu vytvořit. Jestliže, ale tuto mapu budu chtít použít na své webové stránce, tak do určitého počtu návštěvníků je použití sice bezplatné. Kdyby, ale na stránku chodilo, už větší množství návštěvníků, tak by použití této mapy bylo zpoplatněno. Takže jsem věděl, že toto není vhodné řešení, protože by to v budoucnu působilo komplikace.

Zbývala ještě další možnost, a to zkusit podobné open-source technologie pro tvorbu map, kde by použití mapy nebylo nějak zpoplatněno. Byla mi doporučena stránka zvaná OpenStreetMap, kde je možnost vytvořit si svoji mapu a zdarma ji využívat na stránce. Ale jelikož tato stránka nebyla až tak rozšířená a discgolf také není ještě moc známý, tak stránka neznala většinu discgolfových hřišť a bylo by třeba je označit a doplnit všechny po jednom. To, ale nebylo možné stihnout vzhledem k datumu odevzdání tohoto projektu.

Zvolil jsem tedy poslední možnost a to tu, že použiji už vytvořenou interaktivní mapu někoho jiného. Jelikož tato mapa byla už volně přístupná na jiné webové stránce, tak jsem ji mohl využít bezplatně i na té mé. Tímto se vyřešili všechny problémy, ale v budoucnu jistě zkusím vymyslet jiný způsob.

3.12. Rozložení webové stránky

Stránka je rozdělena tak, aby každá funkce měla svou vlastní stránku. Některé stránky jsou přístupné všem uživatelům, zatímco ostatní jsou dostupné pouze přihlášeným uživatelům.

3.12.1. Homepage

Homepage slouží jako uvítací stránka, kterou uživatel vidí jako první při návštěvě webu. Zde může uživatel najít přehled všech funkcí, které Discit nabízí, stejně jako jeho účel a stručný popis každé ze stránek.

3.12.2. O hře

Stránka „O hře“ slouží jako zdroj informací pro hráče, kteří ještě discgolf neznají a nejsou ještě seznámeni s pravidly a stylem hry.

3.12.3. Hřiště

Stránka „Hřiště“ je k tomu, aby si uživatel mohl podívat kolik hřišť je v České republice a popřípadě jestli je nějaké hřiště poblíž jeho bydliště, aby si mohl zahrát.

3.12.4. Záznamy her

Stránka „Záznamy“ slouží jako výstup pro webovou aplikaci. Zde si uživatel může zobrazit všechny uložené hry, které si může také navíc rozkliknout a podívat se na detail dané hry.

3.12.5. Seznam přátel

Na stránce „Přátelé“ si může uživatel zobrazit seznam všech svých přátel, poslat žádost o přátelství někomu jinému a také řídit žádosti, které přišli jemu.

3.13. Webová aplikace

Součástí stránky je také zmíněná část s webovou aplikací, která slouží k větší interakci stránky s uživatelem. Je rozdělena na následující dvě části.

3.14. Nastavení parametrů hry

Tato stránka slouží jako "lobby", kde si uživatel může vybrat, na kterém hřišti chce hrát. Na základě této volby se určuje počet jamek, PAR a další informace, které může uživatel později vidět v detailu hry, pokud si ji uloží. Zároveň si zde může uživatel vybrat, s kým ze svých přátel chce hrát.

3.15. Probíhající hra

Tato stránka se zobrazuje uživateli poté, co začne hru, a má za cíl usnadnit průběh hry. Poskytuje uživateli veškeré nástroje a informace, které může při hře potřebovat.

Hlavní funkcí této stránky je umožnit uživateli sledovat bodování a počet hodů, které provede na každé jamce zvoleného hřiště. Dále umožňuje uživateli nastavit pořadí, v jakém hráči provádějí své hody na jednotlivých jamkách. Kromě toho stránka obsahuje textové pole, do kterého uživatel může zadat, jaký PAR má daná jamka, což je později důležitá informace pro správné hodnocení výkonu hráče.

Tyto funkce poskytují vše, co uživatel potřebuje k pohodlnému sledování hry a díky jednoduchosti těchto funkcí je ovládání stránky velmi uživatelsky přívětivé.

4. Závěr

Ke konečnému shrnutí tak všechny stanovené cíle projektu byly splněny. Provedl jsem výzkum existujících řešení, navrhl novou aplikaci pro hráče, vymyslel architekturu, implementoval všechny funkce, které jsem si stanovil, aplikaci otestoval a spustil na adrese https://disc-it.vercel.app, kde momentálně čeká, než bude otestován hráči.

Při práci na tomto projektu jsem se naučil hodně nových informací o používaných technologiích v projektu. Už před začátkem práce na projektu jsem byl s nimi lehce seznámen, ale věděl jsem o nich spíše povrchové informace. V průběhu práce na projektu jsem zjistil o technologiích více detailních informacích a také jak vlastně jednotlivé technologie fungujou na detailnější úrovni. Samozřejmě jsem se v nich i zlepšil a některé z nich jsem si i oblíbil, to zejména knihovny React a MUI.

Nicméně můj zájem o tento projekt nekončí jeho dokončením. Plánuji na něm nadále pracovat a rozvíjet ho ve všech směrech, které jsou pro jeho další zdokonalení vhodné. Prvním krokem bude vyřešení problému s mapou, aby stránka měla mapu, která se stránkou ladí, a ještě měla pár funkcí navíc.

Zdroje

[1] PROFESSIONAL DISC GOLF ASSOCIATION [PDGA]. *PDGA Professional Disc Golf Association* [online]. 1998 [cit. 2024-02-19]. Dostupné z: https://www.pdga.com

[2] UDisc. *The App for Disc Golfers* [online]. 2024 [cit. 2024-02-19]. Dostupné z: https://udisc.com

[3] Co je to frontend? Open AI. *ChatGPT* [online]. 2022 [cit. 2024-02-19]. Dostupné z: https://chat.openai.com

[4] Co je to backend? Open AI. *ChatGPT* [online]. 2022 [cit. 2024-02-19]. Dostupné z: https://chat.openai.com

[5] Co je to API? Open AI. *ChatGPT* [online]. 2022 [cit. 2024-02-19]. Dostupné z: https://chat.openai.com

[6] UML – Use Case Diagram. ITnetwork. *ITnetwork* [online]. 2024 [cit. 2024-02-19]. Dostupné z: https://www.itnetwork.cz/navrh/uml/uml-use-case-diagram

[7] Co je to Hashing a jak funguje? Open AI. *ChatGPT* [online]. 2022 [cit. 2024-02-19]. Dostupné z: https://chat.openai.com

[8] React (webový framework) Wikipedie. *Wikipedie Otevřená encyklopedie* [online]. 2001 [cit. 2024-02-19]. Dostupné z: https://cs.wikipedia.org/wiki/React_(webový_framework)

[9] Co je to framework Next.js? Open AI. *ChatGPT* [online]. 2022 [cit. 2024-02-19]. Dostupné z: https://chat.openai.com

[10] Co je to MUI? Open AI. *ChatGPT* [online]. 2022 [cit. 2024-02-19]. Dostupné z: https://chat.openai.com

[11] Co je to Firebase? Open AI. *ChatGPT* [online]. 2022 [cit. 2024-02-19]. Dostupné z: https://chat.openai.com

[12] Co je to GraphQL? Open AI. *ChatGPT* [online]. 2022 [cit. 2024-02-19]. Dostupné z: https://chat.openai.com

[13] Co je to TypeScript? Open AI. *ChatGPT* [online]. 2022 [cit. 2024-02-19]. Dostupné z: https://chat.openai.com

[14] Co je to Vercel? Open AI. *ChatGPT* [online]. 2022 [cit. 2024-02-19]. Dostupné z: https://chat.openai.com

[15] draw.io [online]. [cit. 2024-02-19]. Dostupné z: https://app.diagrams.net

[16] ReactJS Virtual DOM. Sanchhaya Education Private Limited. *GeeksforGeeks* [online]. 2008 [cit. 2024-02-19]. Dostupné z: https://www.geeksforgeeks.org/reactjs-virtual-dom/

[17] What is server-side rendering: definition, benefits and risks. EPAM Systems, Inc. *SolutionsHub* [online]. 2019 [cit. 2024-02-19]. Dostupné z: https://solutionshub.epam.com/blog/post/what-is-server-side-rendering

[18] Routing. Vercel, Inc. *Next.js* [online]. 2016 [cit. 2024-02-19]. Dostupné z: https://nextjs.org/docs/pages/building-your-application/routing

[19] Software pro tvorbu databázových diagramů [online]. [cit. 2024-02-19]. Dostupné z: https://dbdiagram.io/home

[20] Authentication. Google. *Firebase* [online]. 1998 [cit. 2024-02-19]. Dostupné z: https://firebase.google.com/docs/auth

[21] Oprava textů a asistence s gramatikou. Open AI. *ChatGPT* [online]. 2022 [cit. 2024-02-19]. Dostupné z: https://chat.openai.com

[22] Jak funguje GraphQL API? Open AI. *ChatGPT* [online]. 2022 [cit. 2024-02-20]. Dostupné z: https://chat.openai.com

[23] Jak funguje Firebase Firestore API? Open AI. *ChatGPT* [online]. 2022 [cit. 2024-02-20]. Dostupné z: https://chat.openai.com

[24] Jak funguje Autentifikace nabízená Firebase SDK? Open AI. *ChatGPT* [online]. 2022 [cit. 2024-02-20]. Dostupné z: https://chat.openai.com

[25] Apollo Client docs. Apollo Graph, Inc. *Apollo Client* [online]. 2017 [cit. 2024-02-20]. Dostupné z: https://www.apollographql.com/docs/react/

[26] Apollo Codegen Basics. Apollo Graph, Inc. *Apollo Codegen* [online]. 2017 [cit. 2024-02-20]. Dostupné z: https://www.apollographql.com/tutorials/lift-off-part1/09-codegen

[27] Jak funguje Apollo Codegen? Open AI. *ChatGPT* [online]. 2022 [cit. 2024-02-20]. Dostupné z: https://chat.openai.com

[28] Mapa discgolfových hřišť v ČR. Google. *Google Maps* [online]. 1998 [cit. 2024-02-20]. Dostupné z: https://www.google.com/maps/d/viewer?hl=cs&hl=cs&mid=1l1e3vtbuWur1NVGMC9x0dx_tQTk&ll=49.665209155635154%2C15.088174549211889&z=7

[29] Co je hashování a jeho typy? Jan Štrafelda. *Strafelda* [online]. 2005 [cit. 2024-03-01]. https://www.strafelda.cz/hash.

[30] Průzkum popularity jazyků a technologií. Stack Exchange Inc. *Stackoverflow* [online]. 2008 [cit. 2024-03-01]. https://survey.stackoverflow.co/2023/#section-admired-and-desired-programming-scripting-and-markup-languages

Seznam obrázků

Obr. 1: UML Use Case Diagram 6
Obr. 2: Schéma architektury webu 7
Obr. 3: DOM (Document Object Model) 8
Obr. 4: Schéma databáze 11

Zadání maturitního projektu z informatických předmětů

Jméno a příjmení: *Tomáš Terč*

Pro školní rok: *2022/2023*

Třída: *4.*

Obor: *Informační technologie 18-20-M/01*

Téma práce: *Webová aplikace pro hráče discgolfu*

Vedoucí práce: *Tomáš Pešek*

Způsob zpracování, cíle práce, pokyny k obsahu a rozsahu práce:

Cílem práce je navrhnout a realizovat webovou aplikaci pro hru discgolf. Řešení bude obsahovat jak anonymní část s obecnými informacemi o hře, tak funkcionalitu podobnou malé sociální síti, konkrétně:

1. **Obecné informace:** Informační stránky pro potenciální zájemce, pravidla triky apod.
2. **Mapa:** Interaktivní mapa s lokacemi hřišť po České republice.
3. **Systém přátel:** Možnost přidávat ostatní uživatele do přátel a spravovat přátelství.
4. **Aplikace pro zaznamenání hry:** Uživatelé si v aplikaci mohou počítat body, hody a jamky v aktuální hře. Ke konci si lze odehranou hru uložit.
5. **Uložené hry:** Pro uložené hry lze zobrazit průběh hry a další statistiky (porovnání s minulými hrami, žebříčky v porovnání s přáteli apod.)

Doporučenými technologiemi jsou React, TypeScript, Firebase a GraphQL.

Stručný časový harmonogram (s daty a konkretizovanými úkoly):

Září: Rozvrhnutí stránky + vytvoření UML Use Case diagramu

Říjen: Začátek práce na informačních stránkách

Listopad: Dokončení informačních stránek + začátek práce na částí s aplikací

Prosinec: Dokončení části s aplikací

Leden: Dokončení posledních detailů a začátek práce na dokumentaci

Únor: Práce na dokumentaci

Březen: Dokončení celého projektu s dokumentací + odevzdání

Přihlašovací údaje

Uživatel 1 (vedoucí)

Uživatel 2 (oponent)