

Semestrálny projekt - mobilná aplikácia Tournify

Zdenko Kanoš & Tomáš Majerčík

MTAA

12. mája 2025

1 Účel aplikácie

1.1 Stručný popis aplikácie

Mobilná aplikácia je určená na správu športových turnajov, registráciu tímov, zobrazovanie lístkov (vrátane ich kontroly), či sledovanie výsledkov. Cieľom aplikácie je zjednotiť organizáciu športových podujatí a zefektívniť komunikáciu medzi organizátormi a účastníkmi.

Používatelia dostávajú odporúčania na turnaje na základe ich preferencií, môžu si prezerať zoznam dostupných turnajov, vytvárať nové tímy, alebo sa pridávať do existujúcich tímov. Po úspešnej registrácii získajú digitálny lístok s QR kódom, ktorý slúži ako vstupenka a identifikátor počas turnaja. Organizátori môžu skenovať QR kódy priamo v aplikácii a pridávať výsledky zápasov (upravovať leaderboard).

Aplikácia je prepojená s backendovým serverom prostredníctvom REST API a podporuje offline cache vybraných dát, ako sú lístky. Užívateľské rozhranie je navrhnuté s dôrazom na jednoduchosť, intuitívnosť a responzívnosť.

1.2 Cieľová skupina používateľov

Aplikácia je navrhnutá pre používateľov, ktorí počas používania aplikácie môžu zastávať tieto role:

- **Účastníci turnajov** – používatelia, ktorí sa chcú zapojiť do organizovaných turnajov, registrovať svoje tímy, získať lístok a sledovať výsledky. Každý, kto chce byť pripojený v turnaji, musí byť súčasťou tímu, hoc aj jednočlenného.
- **Organizátori turnajov** – používatelia, zodpovední za správu turnaja, kontrolu účastníkov (skenovanie QR kódov), evidenciu výsledkov zápasov (úpravu leaderboardu) a riadenie priebehu podujatia.

Aplikácia tak pomáha používateľom, ktorí sa chcú zapojiť do turnajov s **nájdением a prihlásením** sa do týchto turnajov a organizátorom s **propagáciou, organizáciou prihlásenia a záznamu výsledku**.

2 Funkcionalita aplikácie

Aplikácia poskytuje nasledovné funkcie:

- **Zobrazenie športových kategórií a turnajov**
- **Zobrazenie detailu turnaja so všetkými informáciami**
- **Registrácia do turnaja cez vytvorenie alebo výber tímu**
- **Generovanie a zobrazenie QR lístka po registrácii**
- **Offline zobrazenie uložených lístkov**
- **Skenovanie QR kódov (organizátor) na overenie účasti**

- Úprava leaderboardu
- Zobrazenie leaderboardu pre účastníkov
- Správa používateľského profilu a preferencií
- Odporúčanie turnajov na základe preferencií používateľa

2.1 Hlavné scenáre použitia

Aplikácia pokrýva viaceré bežné situácie, s ktorými sa môžu používatelia počas interakcie s turnajovým systémom stretnúť. Medzi hlavné scenáre použitia patria:

- **Prezeranie zoznamu turnajov** – používateľ si môže pozrieť zoznam všetkých verejne dostupných turnajov vrátane základných informácií (názov, dátum, lokalita, športová disciplína). Najprv vidí všetky dostupné kategórie, potom po kliknutí na konkrétnu kategóriu zobrazia všetky turnaje v tejto kategórii. Je sprístupnená aj možnosť vyhľadávania podľa názvu.
- **Zobrazenie detailu turnaja** – používateľ si otvorí detail turnaja, kde nájde podrobnosti, ako napr. popis turnaja, mapu miesta konania, zoznam prihlásených tímov či aktuálny stav leaderboardu.
- **Registrácia do turnaja** – používateľ si vyberie, či chce vytvoriť nový tím, alebo sa pripojiť k existujúcemu tímu. Po úspešnej registrácii získa digitálny lístok s QR kódom.
- **Zobrazenie a správa lístkov** – používateľ má prístup k svojim lístkom, ktoré zobrazujú QR kód a ďalšie základné informácie.
- **Skenovanie QR kódu (organizátor)** – organizátor môže na mieste turnaja skenovať QR kódy účastníkov, čím overí ich prihlásenie a platnosť lístka.
- **Zadávanie výsledkov (organizátor)** – organizátor môže počas turnaja a po jeho skončení pridávať alebo upravovať výsledky v leaderboarde daného turnaja.
- **Offline prístup k lístkom** – aplikácia uchováva lístky v cache, čím umožňuje ich zobrazenie aj bez internetového pripojenia.

3 Komunikácia s backendom

Aplikácia komunikuje s backendovým serverom pomocou REST API (Representational State Transfer). Backend je implementovaný pomocou **Node.js** frameworku **Express**. Každá funkcionálna aplikácie je naviazaná na konkrétny HTTP endpoint, ktorý slúži na čítanie, zápis alebo aktualizáciu údajov v databáze.

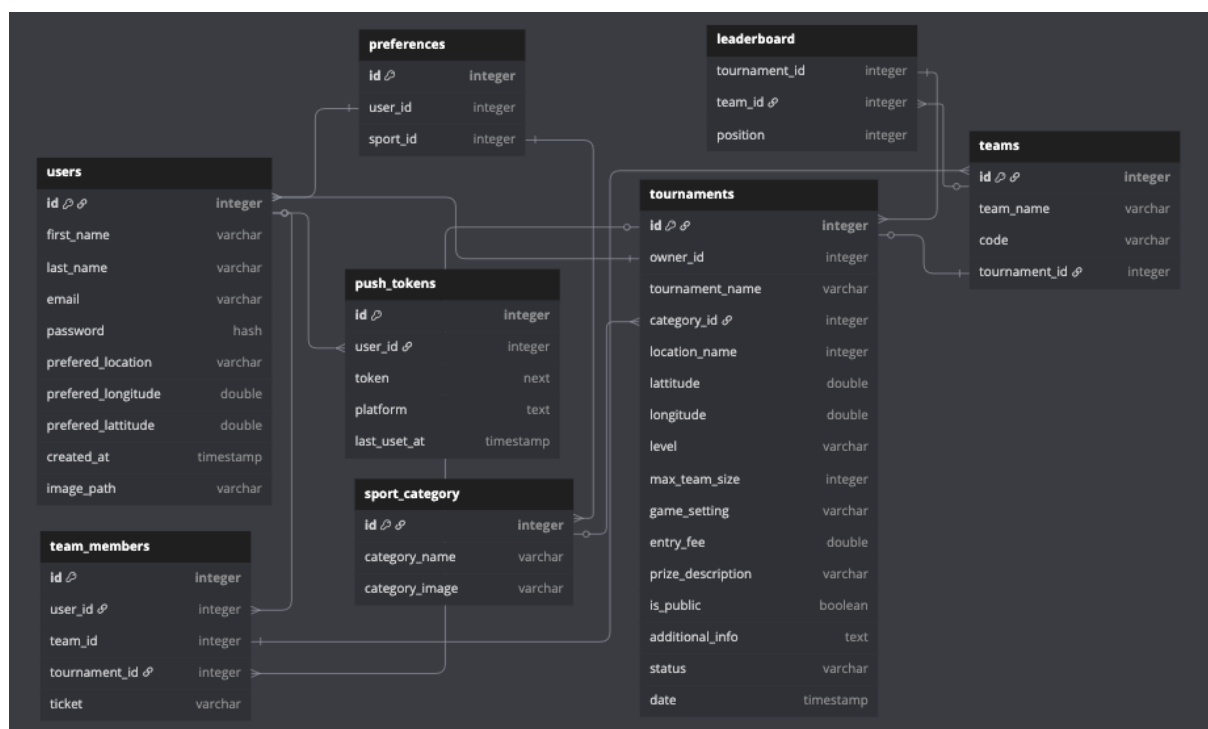
- **Formát správ:** Všetky požiadavky aj odpovede prebiehajú vo formáte JSON.
- **HTTP metódy:**
 - GET – načítanie údajov (napr. detail turnaja, zoznam turnajov, lístkov)

- POST – vytvorenie nových údajov (napr. registrácia do turnaja, prihlásenie tímu)
 - PUT – úprava existujúcich údajov (napr. aktualizácia turnaju)
 - DELETE – odstránenie údajov (napr. vymazanie záznamu z leaderboardu)
- **Autentifikácia:** Väčšina endpointov, ktoré vyžadovali autentifikáciu, je chránená pomocou Bearer tokenu, ktorý sa odosiela v hlavičke **Authorization**. Tokenizácia je riešená cez JWT (JSON Web Token).
 - **Cache:** Vybrané dáta (napr. lístky) sú po načítaní uložené lokálne kvôli offline prístupu.

3.1 API endpointy

Dokumentácia API endpointov je generovaná pomocou nástroja **Swagger**, ktorý poskytuje prehľadné rozhranie na testovanie a pochopenie fungovania jednotlivých volaní.

4 Model databázy



Obr. 1: Fyzický dátový model

5 API zariadenia a API tretích strán

Aplikácia využíva viacero natívnych rozhraní zariadenia a externých služieb tretích strán na rozšírenie funkcionality a zlepšenie používateľského zážitku.

API zariadenia

- **Galéria a fotoaparát** – pri vytváraní konta môže používateľ pridať obrázok z galérie zariadenia. Používame knižnicu `expo-image-picker`, ktorá umožňuje jednoduchý výber a spracovanie multimediálneho obsahu.
- **Geolokácia** – aplikácia dokáže načítať aktuálnu polohu používateľa (pre účely odporúčania turnajov v okolí). Na tento účel je použitá knižnica `expo-location`, ktorá poskytuje prístup k údajom o polohe po udelení príslušného oprávnenia.
- **Zobrazovanie máp** – pri zobrazení detailu turnaja je možné vizualizovať miesto konania priamo na mape pomocou komponentu `MapView`, ktorý je súčasťou knižnice `react-native-maps`. Mapa sa zobrazuje nad dynamicky načítanými súradnicami (latitude, longitude).
- **Skenovanie QR kódov** – organizátor turnaja môže použiť kameru zariadenia na načítanie QR kódov pomocou komponentu `CameraView` z knižnice `expo-camera`. Po naskenovaní sa vykoná POST požiadavka na backend pre overenie platnosti lístka.
- **Akcelerometer (shake-to-refresh)** – aplikácia dokáže detekovať pohyb zariadenia (otras) a na jeho základe spustiť obnovenie obsahu (refresh). Implementované cez knižnicu `expo-sensors`, konkrétne modul `Accelerometer`.

API tretích strán

- **Push notifikácie (Expo)** – aplikácia zasiela upozornenia na mobilné zariadenia používateľov pomocou služby Expo. Používame HTTP rozhranie `https://exp.host/--/api/v2/push/send`, cez ktoré sa odosielaajú notifikácie vrátane titulku, správy a dát. Odosielanie prebieha asynchrónne pomocou knižnice `axios` a vyžaduje platný Expo Push Token. Push notifikácie sa generujú na backende vždy hodinu pred začatím turnaja. Zároveň sa notifikácia odosiela pri manuálnom spustení turnaja.
- **Google Places Autocomplete** – pri zadávaní adresy turnaja a vyberaní preferovaného mesta aplikácia využíva komponent pre automatické dopĺňanie lokácie na základe reálne existujúcich miest. V pozadí komunikuje so službou Google Places API, čo umožňuje používateľovi pohodlnejšie a rýchlejšie vyplniť lokalitu. Táto API vracia Longitude & latitude na základe ktorých je vypočítaná aktuálne vzdialenosť používateľa od turnaja (vzdušnou čiarou).

6 Akceptačné testy

Test 1: Registrácia nového používateľa a nastavenie preferencií
Vstupné podmienky: Používateľ nie je prihlásený a nie je registrovaný email s jeho adresou
Výstupné podmienky: Nový účet je úspešne vytvorený a používateľ je automaticky prihlásený
Postup: <ol style="list-style-type: none">1. Otvorí sa aplikácia2. Používateľ klikne na tlačidlo “SIGN UP”3. Vyplní email, heslo a potvrdí heslo4. Klikne na tlačidlo “REGISTER”5. Vyplní preferované športy a mesto6. Zobrazí sa hlásenie o úspešnej registrácii7. Používateľ je automaticky prihlásený a presmerovaný na hlavnú stránku
Výsledok: PASS / FAIL

Test 2: Pripojenie sa k turnaju
Vstupné podmienky: Používateľ je prihlásený do aplikácie
Výstupné podmienky: Používateľ je prihlásený na turnaj
Postup: <ol style="list-style-type: none">1. Používateľ klikne na tlačidlo “explore“ v spodnej lište2. Zvolí kategóriu športu na ktorý sa chce prihlásiť3. Vyhľadá konkrétny turnaj pomocou vyhľadávania podľa názvu4. Stlačí tlačidlo na zobrazenie informácií o turnaji5. Pridá seba a ďalšieho člena tímu a potvrdí prihlásenie6. V karte “events“ vidí turnaj na ktorý sa práve registroval
Výsledok: PASS / FAIL

Test 3: Zobrazenie histórie turnaja

Vstupné podmienky:

Používateľ je prihlásený a zúčastnil sa turnaja, ktorý je už ukončený a má pridané umiestnenia

Výstupné podmienky:

Používateľ si vie pozrieť umiestnenie a informácie o turnaji, ktorého sa zúčastnil

Postup:

1. Používateľ na hlavnej obrazovke klikne v sekcii histórie turnajov na tlačidlo "VIEW ALL"
2. Zobrazia sa všetky turnaje, ktorých sa používateľ zúčastnil a boli ukončené
3. Používateľ vidí názov turnajov aj s pozíciou na ktorej sa umiestnil
4. Klikne na ikonu so symbolom "i"
5. Zobrazia sa podrobnosti o turnaji

Výsledok: PASS / FAIL

Test 4: Vytvorenie turnaja

Vstupné podmienky:

Používateľ je prihlásený

Výstupné podmienky:

Turnaj je úspešne vytvorený a zobrazí sa v sekcii events

Postup:

1. Používateľ klikne na tlačidlo "Create" na spodnej lište
2. Vyplní názov turnaja, dátum, typ hry, lokalitu a ďalšie potrebné informácie
3. Klikne na tlačidlo "SUBMIT"
4. Zobrazí sa potvrdenie o úspešnom vytvorení turnaja
5. Používateľ je presmerovaný na obrazovku turnaja
6. Vidí turnaj v zozname vytvorených turnajov

Výsledok: PASS / FAIL

Test 5: Offline režim
Vstupné podmienky: Používateľ je prihlásený a zaregistrovaný na turnaj, ale nemá prístup k internetu
Výstupné podmienky: Používateľ vidí svoju vstupenku aj bez internetu
Postup: <ol style="list-style-type: none"> 1. Používateľ otvorí aplikáciu bez internetu 2. Vidí v hlavnej obrazovke sekciu s lístkami 3. Kliknutím vyberie konkrétny lístok 4. Zobrazí sa jeho vstupenka so základnými informáciami a QR kódom 5. QR kód je možné zväčšiť kliknutím naň 6. Po opätovnom pripojení má prístup ku všetkým funkciám bez zatvorenia aplikácie
Výsledok: PASS / FAIL

Test 6: Zaznamenanie umiestnenia
Vstupné podmienky: Turnaj je vytvorený
Výstupné podmienky: V turnaji budú upravené umiestnenia
Postup: <ol style="list-style-type: none"> 1. Používateľ klikne tlačidlo "START" a spustí turnaj ak už nie je spustený 2. Používateľ klikne na "EDIT STATISTICS" a otvorí úpravu umiestnení 3. Klikne na tlačidlo pridať pri umiestnení 4. Vyhľadá a vyberie tím podľa názvu 5. Tím sa pridá do daného umiestnenia 6. Po stlačení tlačidla "SAVE" sa umiestnenia uložia a zobrazia aj ostatným používateľom v histórii po ukončení turnaja
Výsledok: PASS / FAIL

Test 7: Notifikácia o nastávajúcom turnaji
Vstupné podmienky: Používateľ je prihlásený do aplikácie
Výstupné podmienky: Používateľovi sa zobrazí upozornenie v podobe notifikácie
Postup: <ol style="list-style-type: none"> 1. Používateľ sa registruje na turnaj 2. Systém to zaznamená a pridá používateľa do zoznamu upozornení 3. 1 hodinu pred začiatkom turnaja, backend generuje notifikáciu na nasledujúcu udalosť 4. Používateľ dostane push notifikáciu na svojom zariadení 5. Po kliknutí na notifikáciu sa otvorí aplikácia
Výsledok: PASS / FAIL

Test 8: Pripojenie sa druhý krát na ten istý turnaj
Vstupné podmienky: Používateľ je prihlásený na daný turnaj
Výstupné podmienky: Používateľ dostane chybové hlásenie.
Postup: <ol style="list-style-type: none"> 1. Používateľ vyhladá turnaj 2. Iný používateľ sa ho snaží prihlásiť do svojho tímu 3. Klikne registrovať 4. Zobrazí sa chybové hlásenie, že daný používateľ sa už na tento turnaj registroval 5. Systém nedovolí aby bola osoba prihlásená 2 krát
Výsledok: PASS / FAIL

Test 9: Pokus o pripojenie sa do plného turnaja
Vstupné podmienky: Turnaj má nastavenú kapacitu, kapacita je plná
Výstupné podmienky: Systém nedovolí používateľovi sa prihlásiť
Postup: <ol style="list-style-type: none"> 1. Používateľ vyhladá turnaj 2. Stlačí tlačidlo na zobrazenie informácií o turnaji 3. Vyplní údaje a potvrdí prihlásenie 4. Systém zistí, že turnaj je plný a nedovolí používateľovi sa pridať 5. Vypíše chybovú hlášku, že turnaj je plný
Výsledok: PASS / FAIL

Test 10: Pridanie turnaja s názvom, ktorý už existuje
Vstupné podmienky: Používateľ má už aspoň jeden vytvorený turnaj a vytvára ďalší
Výstupné podmienky: Používateľ dostane chybové hlásenie
Postup: <ol style="list-style-type: none"> 1. Používateľ začne vytvárať nový turnaj 2. Zadá rovnaký názov turnaja, taký aký už má iný turnaj 3. Vyskočí chybová hláška 4. Používateľ má možnosť zmeniť meno 5. Po zmene názvu, môže byť turnaj pridaný.
Výsledok: PASS / FAIL

7 Zmeny oproti návrhu

Počas vývoja aplikácie došlo k niekoľkým úpravám v porovnaní s pôvodne navrhovanou funkcionalitou. Tieto zmeny boli realizované na základe praktických potrieb, používateľských preferencií a zjednodušenia používateľskej skúsenosti:

- **Zmena spôsobu registrácie na turnaj:** Pôvodný návrh predpokladal, že jeden používateľ môže pri registrácii prihlásiť celý tím vrátane ďalších hráčov. Toto bolo zmenené tak, že každý používateľ sa na turnaj prihlasuje samostatne. Po registrácii môže používateľ buď:
 - vytvoriť nový tím, pričom získa unikátny kód tímu, ktorý môže odoslať kamarátom
 - alebo sa pripojiť do existujúceho tímu zadáním tohto kódu v sekcii **Join Team**

Tento spôsob zjednodušuje správu tímov a eliminuje potrebu zadávať údaje za iných používateľov.

- **Zmena stavov turnaja:** Pôvodné stavy **Ongoing**, **Ended** a **Suspended** boli nahradené stavmi:
 - **upcomming** – turnaj sa ešte nezačal,
 - **ongoing** – turnaj aktuálne prebieha,
 - **closed** – turnaj bol ukončený.

Použitie týchto stavov je pevne definované a systém neakceptuje iné hodnoty.

- **Zmena frekvencie upozornení:** V pôvodnom návrhu boli notifikácie plánované na 24 hodín, 1 hodinu a 15 minút pred začiatkom turnaja. Toto sa zmenilo na:
 - Jedna notifikácia odoslaná 1 hodinu pred začiatkom turnaja,
 - Druhá notifikácia je odoslaná v momente, keď organizátor turnaj manuálne spustí.

Týmto sa eliminuje zbytočné zahlcovanie používateľa notifikáciami a zvyšuje sa relevantnosť informácií.

- **Pridanie tabuľky push_tokens:** Tabuľka `push_tokens` je potrebná pre správne fungovanie notifikácií.
- **Zmeny na backend**
 - Pribudla možnosť získavať obrázky kategórie turnajov.
 - Opravil sa problém s kategóriou pri úprave turnaja, aby sa správne načítavala a zobrazovala.
 - Zaviedla sa podpora pre black & white režim, vrátane šedotónových obrázkov kategórií.
 - Pri registrácii do existujúceho tímu cez kód sa teraz vracia správne ID lístka, čo umožňuje prechod na detail.
 - Pridané notifikácie a podpora websocketov, čím sa systém priblížil k real-time aktualizáciám.

8 Github repozitáre

- <https://github.com/zdenkokanos/MTAA-frontend>
- <https://github.com/zdenkokanos/MTAA-backend/>

Niektoré časti textu, boli preformulované alebo jazykovo upravené s využitím jazykového modelu spoločnosti OpenAI, ChatGPT.