



**MATEMATICKO-FYZIKÁLNÍ  
FAKULTA**  
Univerzita Karlova

# **BAKALÁŘSKÁ PRÁCE**

Ricardo Bolemant

## **Správce sportovních soutěží**

Katedra distribuovaných a spolehlivých systémů

Vedoucí bakalářské práce: doc. RNDr. Jan Kofroň, Ph.D.

Studijní program: Informatika

Studijní obor: Informatika

Praha rok 2022

Týmto by som chcel poďakovať vedúcemu mojej práce, pánovi doc. RNDr. Janovi Kofroňovi, Ph.D. za užitočné rady pri vývoji bakalárskej práce. Ďalej by som chcel poďakovať mojim rodičom za ich podporu pri štúdiu.

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně a výhradně s použitím citovaných pramenů, literatury a dalších odborných zdrojů.

Beru na vědomí, že se na moji práci vztahují práva a povinnosti vyplývající ze zákona č. 121/2000 Sb., autorského zákona v platném znění, zejména skutečnost, že Univerzita Karlova má právo na uzavření licenční smlouvy o užití této práce jako školního díla podle § 60 odst. 1 autorského zákona.

V ..... dne.....

podpis

Název práce: Správce sportovních soutěží

Autor: Ricardo Bolemant

Katedra / Ústav: Katedra distribuovaných a spolehlivých systémů

Vedoucí bakalářské práce: doc. RNDr. Jan Kofroň, Ph.D., Katedra distribuovaných a spolehlivých systémů

Abstrakt: Aplikace je určena pro organizátory florbalových a hokejových soutěží, ať už ligových nebo turnajových. Pomocí má hlavně amatérským soutěžím při jejich správě a počítání různých statistik. Organizátorům usnadní práci a ušetří čas díky jednoduchému rozhraní. Díky této aplikaci budou moci i menší sportovní soutěže zaznamenávat detailní statistiky jako celostátní či mezinárodní profesionální soutěže. Hlavním cílem je zjednodušení vyplňování údajů o zápasech a jejich zpracování.

Klíčová slova: sportovní soutěže, sportovní statistiky

Title: Sports Manager

Author: Ricardo Bolemant

Department: Department of Distributed and Dependable Systems

Supervisor: doc. RNDr. Jan Kofroň, Ph.D., Department of Distributed and Dependable Systems

Abstract: The application is intended for organizers of floorball and hockey competitions, whether league or tournament. It is mainly to help amateur competitions in their management and calculation of various statistics. It simplifies the work for the organizers and saves time thanks to a simple interface. Thanks to this application, even smaller sports competitions will be able to record detailed statistics such as national or international professional competitions. The main goal is to simplify the filling in of match data and their processing.

Keywords: sports competitions, sports statistics

Názov práce: Správca športových súťaží

Autor: Ricardo Bolemant

Katedra / Ústav: Katedra distribuovaných a spoľahlivých systémů

Vedúci bakalárskej práce: doc. RNDr. Jan Kofroň, Ph.D., Katedra distribuovaných a spoľahlivých systémů

Abstrakt: Aplikácia je určená pre organizátorov florbalových a hokejových súťaží, či už ligových alebo turnajových. Pomôcť má hlavne amatérskym súťažiam pri ich spravovaní a počítaní rôznych štatistík. Organizátorom zjednoduší prácu a ušetrí čas vďaka jednoduchému rozhraniu. Vďaka tejto aplikácii budú môcť aj menšie športové súťaže zaznamenávať detailné štatistiky ako celoštátne či medzinárodné profesionálne súťaže. Hlavným cieľom je zjednodušenie vyplňania údajov o zápasoch a ich spracovania.

Kľúčové slová: športové súťaže, športové štatistiky

# Obsah

<b>1. Úvod.....</b>	<b>1</b>
1.1 Popis a zameranie softwarového diela .....	1
1.2 Motivácia k vzniku softwarového diela .....	1
<b>2. Analýza požiadaviek .....</b>	<b>2</b>
2.1 Definície pojmov .....	2
2.2 Požadované vlastnosti aplikácie.....	3
2.3 Prostredie aplikácie .....	5
2.4 Analýza .....	5
2.4.1 Návrh aplikácie .....	5
2.4.2 Programovací jazyk – C# .....	8
2.4.3 Grafické užívateľské rozhranie – WPF .....	9
2.4.4 Databáza – MySQL.....	10
2.4.5 Optické rozpoznávanie znakov – Python.....	10
2.5 Použité knižnice .....	11
2.5.1 Softwarové knižnice pre Python .....	11
2.5.2 Iné softwarové knižnice .....	11
<b>3. Implementácia .....</b>	<b>13</b>
3.1 Architektúra aplikácie .....	13
3.2 Databázová schéma.....	14
3.3 Hlavné triedy aplikácie .....	16
3.3.1 Models.....	16
3.3.2 Views .....	17
3.3.3 View models .....	17
3.3.4 Commands .....	19
3.3.5 Ostatné pomocné triedy.....	19
3.4 Ukladanie obrázkov .....	20
3.5 Prehľad štatistík.....	20
3.6 Vytvorenie zápasu.....	22
3.6.1 Nutné údaje .....	23
3.6.2 Stav hry .....	24
3.6.3 Pravidlá hry .....	25
3.6.4 Načítanie z naskenovaného záznamu.....	27
3.7 Navigácia medzi pohľadmi .....	29
3.8 Pavúk.....	30
3.9 Rozpis zápasov .....	31
3.10 Poradie tímov v základnej časti.....	32
3.11 Aktuálny stav aplikácie .....	32
3.12 Pri spustení aplikácie.....	33
3.13 Konzistencia údajov .....	33
3.13.1 Tvorba sezóny .....	33
3.13.2 Uzamknutie editácie zápasov .....	34
3.13.3 Pridávanie a odstraňovanie hráčov.....	34
3.14 Práca s databázou .....	35

3.15	Grafy .....	35
3.16	Export.....	35
<b>4.</b>	<b>Funkcionalita.....</b>	<b>37</b>
4.1	Navigácia.....	37
4.2	Súťaže .....	37
4.3	Sezóny .....	37
4.3.1	Formát sezóny .....	37
4.4	Zápasy .....	38
4.4.1	Periódy .....	39
4.5	Tímy .....	40
4.6	Hráči.....	40
4.7	Prehľady štatistík.....	40
4.8	Rozpis zápasov .....	41
4.9	Detail zápasu .....	41
4.10	Tabuľky poradia tímov.....	42
4.11	Štatistiky.....	42
4.11.1	Štatistiky v aplikácii.....	42
4.11.2	Štatistiky podporované modelom aplikácie .....	44
4.11.3	Nepodporované štatistiky .....	45
<b>5.</b>	<b>Záver.....</b>	<b>47</b>
5.1	Možnosti budúceho rozšírenia .....	47
	<b>Zoznam použitej literatúry.....</b>	<b>49</b>
	<b>Zoznam obrázkov.....</b>	<b>51</b>
<b>A.</b>	<b>Prílohy .....</b>	<b>52</b>
A.1	Elektronická príloha .....	52
A.2	Zjednodušená databázová schéma .....	53
A.3	Užívateľská dokumentácia .....	54

# 1. Úvod

Organizovanie športových súťaží je časovo náročné a častokrát komplikované. Organizátor okrem zabezpečenia priestorov, vybavenia a veľa ďalších vecí, musí prehľadne spravovať priebeh súťaže. Tým sa myslí zoznam prihlásených tímov a ich hráčov, podrobné záznamy o zápasoch, ale aj samotný formát súťaže, rozdelenie tímov do skupín a následné nasadenie pavúka vyradovacej časti.

Pre atraktivitu súťaže je žiadané viesť si rôzne štatistiky pre porovnávanie tímov a samotných hráčov, sledovanie rekordov a pre odmenenie najlepších hráčov na konci sezóny. Bez využitia softwaru je to pri obrovskom množstve dát prakticky nemožné.

Cieľom tejto práce je navrhnúť a implementovať spravovací systém pre športové súťaže, konkrétne florbalové [1] a hokejové.[2]

## 1.1 Popis a zameranie softwarového diela

Sports Manager je aplikácia cielená pre amatérske športové ligy a turnaje. Zaznamenáva údaje o súťaži, jej účastníkov, zápasy a samotné prehľadné štatistiky. Organizátorom zjednoduší prácu a ušetrí čas vďaka jednoduchému rozhraniu. Vďaka tejto aplikácii budú môcť aj menšie športové súťaže zaznamenávať detailné štatistiky ako celoštátne či medzinárodné profesionálne súťaže. Jedným z cieľov je minimálna veľkosť vstupu od používateľa pre kompletne a pokročilé štatistiky. Celá aplikácia aj s dokumentáciou a testovacími dátami sú dostupné v prílohe A.1 Elektronická príloh. Uživatelská dokumentácia sa nachádza v prílohe A.3 Uživatelská dokumentácia.

## 1.2 Motivácia k vzniku softwarového diela

Množstvo organizátorov amatérskych súťaží si potrebuje uľahčiť a urýchliť spravovanie údajov a štatistík zo zápasov. Organizátori si na začiatkoch vedú štatistiky na papieroch, neskôr pomocou programu Microsoft Office Excel. Stále to je ale časovo náročné a preto som sa rozhodol vytvoriť aplikáciu presne pre túto problematiku.

## 2. Analýza požiadaviek

V tejto kapitole najskôr definujeme potrebné pojmy a potom zanalyzujeme požadované vlastnosti aplikácie. Podľa toho prevedieme analýzu návrhu aplikácie a výber technológií na jej vývoj.

### 2.1 Definície pojmov

*Súťaž* je hlavným objektom v danom športe. Skladá sa z jednotlivých sezón. Patrí konkrétnemu športu.

*Sezóna* obsahuje jednotlivé tímy. Má určitý formát. Väčšinou sa hrá každý rok. Dĺžka sezón sa môže líšiť.

*Liga* je typ súťaže, ktorý sa väčšinou delí do sezón a pozostáva zo skupinovej časti a niekedy aj play-off. Aplikácia Sports Manager ligu ako formát súťaže nijak nerozlišuje.

*Turnaj* je typ súťaže zväčša hraný formátom play-off, ktorému môže predchádzať skupinová časť. Hrá sa v krátkom časovom období a môže pozostávať z viacerých sezón. Aplikácia Sports Manager turnaj ako formát súťaže nijak nerozlišuje.

*Formát* sezóny určuje, akým spôsobom sa odohrávajú zápasy medzi tímami a akým spôsobom sa určuje poradie či postup tímov. Možné formáty sú skupinová časť, kvalifikácia, play-off a ich kombinácia.

*Skupinová* alebo *základná časť* sezóny je formát, v ktorom sú tímy rozdelené do skupín. Poradie tímov sa určuje v rámci ich skupiny.

*Play-off* je formát sezóny, kde tímy hrajú proti sebe v štruktúre pavúka. Môže mu predchádzať základná časť alebo kvalifikácia. Jedna séria play-off môže obsahovať viacero zápasov.

*Kvalifikácia* je formát sezóny, pozostávajúci z viacerých pavúkov. Jedna séria obsahuje len jeden zápas (takzvaná rýchla smrť).

*Pavúk* je štruktúra zápasov známa hlavne z tenisu, kde tímy proti sebe hrajú v stromovitom formáte rozdelenom do kôl. Každý stret sa nazýva séria. Porazený tím je automaticky vyradený a víťazný tím postupuje do ďalšieho kola, kde sa stretne s víťazom inej série v danom kole určenej štruktúrou pavúka.

*Séria* je stret dvoch tímov, ktorý môže obsahovať viacero zápasov. Je určené, na aký počet víťazných zápasov sa hrá. Kto daný počet dosiahne ako prvý, vyhráva sériu.

*Kolo* (hracie kolo) môže byť skupina sérií v pavúku na rovnakej úrovni alebo skupina zápasov v základnej časti zväčša hraných v spoločnom časovom úseku.

*Skupina* pozostáva z tímov, ktorých poradie sa určuje v rámci danej skupiny. Tímy nie sú obmedzené na hranie zápasov iba v rámci skupiny.

*Záznam o zápase* (zápis o zápase) je dokument alebo súbor údajov dôkladne a presne opisujúci celý priebeh zápasu a základné informácie o ňom, ako napríklad čas konania zápasu, tímy, záznam účasti hráčov, strelcov a časy gólov a podobne.

*Periód*a je časový úsek zápasu. V hokeji a florbale sa väčšinou hrajú tri periód

## 2.2 Požadované vlastnosti aplikácie

Aplikácia bude môcť zaznamenávať údaje z viacerých súťaží súčasne. Tie sa delia podľa daného športu. Každá súťaž sa skladá z viacerých sezón. Tým pádom bude potrebné súťaže a ich sezóny usporiadať hierarchicky.

Súťaže môžu byť rôzneho typu, preto by mala aplikácia umožniť tvorbu ligových súťaží pozostávajúcich z rôzneho počtu skupín súťažiacich, ale aj turnajových súťaží vo formáte pavúka. Ďalej bude obsahovať možnosť spájať formáty pre súťaže typu skupinová časť sezóny a následné play-off a taktiež kvalifikácia vo forme pavúkov pred samotnou play-off časťou sezóny.

Aplikácia bude automaticky počítat štatistiky tímov a hráčov a zobrazovať ich v prehľadných tabuľkách.

Najdôležitejšou funkciou je samozrejme zaznamenávanie samotných zápasov, tým používateľ strávi najviac času. Malo by to byť preto čo najjednoduchšie a najefektívnejšie. Aplikácia kladie dôraz na minimalizáciu vstupu od používateľa a prehľadnosť vyplňovacích formulárov.

Súčasťou by mal byť chronologický rozpis odohratých a nadchádzajúcich zápasov. Pre skupinovú časť sezóny by zápasy mali byť rozdelené do hracích kôl a pre kvalifikáciu a play-off by mali byť zobrazené vo forme pavúka.

Aplikácia musí vedieť vypočítať aktuálne poradie tímov v skupinách v základnej časti sezóny na základe ich štatistík. Používateľ by mal mať možnosť vyhlásiť víťaza sezóny a tím ju ukončiť.

Pri vytvorení pavúka by mala existovať možnosť náhodného rozmiestnenia súťažiacich, rovnako aj pre rozmiestnenie do skupín (automatický náhodný los). Pri štarte play-off po ukončení základnej časti musí byť k dispozícii možnosť nasadenia súťažiacich do pavúka play-off podľa poradia v základnej časti.

Aplikácia bude môcť exportovať zoznamy a štatistiky sezón, zápasov, tímov a hráčov, ako aj záznam o zápase. Export bude možný vo formáte PDF<sup>1</sup> alebo XLSX<sup>2</sup>

Pre väčšiu prehľadnosť by malo byť možné k súťažiam, sezónam a tímom priradiť logo a k hráčom fotografiu vo formáte PNG<sup>3</sup> alebo JPEG<sup>4</sup>. Aplikácia bude fotografie uchovávať v lokálnom priečinku.

Pomocou grafov by aplikácia mala vizualizovať rôzne štatistiky tímov a hráčov ako napríklad podiel strelených gólov v sezóne alebo výkon hráčov po zápasoch. Grafy bude taktiež možné exportovať vo formáte PNG alebo JPEG.

Aplikácia by mala mať možnosť načítať dáta o zápase z naskenovaného zápisu pre rýchlejšie vyplňanie. Používateľ si bude môcť vytlačiť predpripravený formát záznamu o zápase.

Dostupná by mala byť aj jednoduchá editácia dát a ich odstránenie. Používateľ by ale nemal byť schopný meniť dáta, na ktorých závisia iné dáta, napríklad odstrániť zápas v prvom kole play-off po tom ako sa už odohralo nadväzujúce druhé kolo play-off.

Sports Manager by mal byť dostupný pre operačné systémy Windows a to ako tie najnovšie, tak aj staršie systémy (Windows 7, Windows Vista).

---

<sup>1</sup> PDF - Portable Document Format je súborový formát pre ukladanie dokumentov nezávisle na software a hardware [3]

<sup>2</sup> XLSX - Microsoft Excel Open XML Spreadsheet je súborový formát pre Microsoft Excel podporovaný viacerými tabuľkovými procesormi

<sup>3</sup> PNG - Portable Network Graphics je grafický formát určený pre bezstratovú kompresiu rastrovej grafiky. [4]

<sup>4</sup> JPEG – (JPG) je digitálny grafický formát, ktorý obsahuje komprimované obrazové údaje

## 2.3 Prostredie aplikácie

Aplikácia je určená výhradne pre operačný systém Windows. Jedná sa tak o desktopovú aplikáciu. Sports Manager má slúžiť na privátnu správu vlastných súťaží, preto sa nepočíta s webovou verziou. Popri behu aplikácie je potrebné mať spustený MySQL server, s ktorým aplikácia pracuje.

## 2.4 Analýza

V tejto podkapitole prediskutujeme a analyzujeme návrh aplikácie a výber technológií použitých pre jej vývoj.

### 2.4.1 Návrh aplikácie

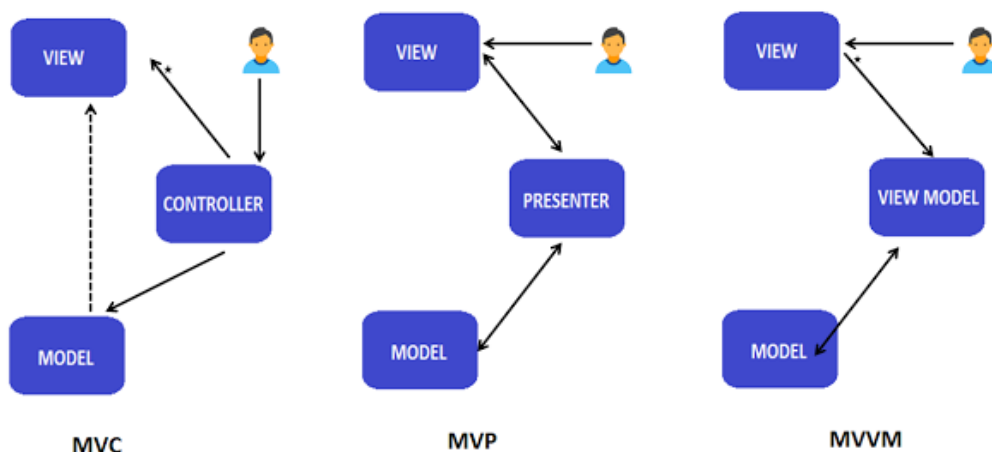
Väčšina bežných užívateľov používa operačný systém Windows. Sports Manager bude preto primárne určená pre tieto operačné systémy. Aplikácia je cieleňá pre všetkých užívateľov, bez ohľadu na ich počítačové znalosti. Výstup bude preto prezentovaný v grafickom rozhraní, pozostávajúcom z jedného hlavného okna. Cez toto hlavné okno bude užívateľ zadávať aj všetok vstup. Aplikácia by teda taktiež mala byť intuitívne ovládateľná. Preto by malo byť grafické rozhranie čo najjednoduchšie. Hlavné okno bude zobrazovať náhľad pre konkrétny šport, ktorý sa po spustení aplikácie vyberie z ponuky. V okne bude vždy dostupná navigácia medzi súťažiami, sezónami, hráčmi a tímami, a taktiež navigácia konkrétnou sezónou. Pred zmenou alebo odstránením dát sa zobrazí vyskakovacie okno na potvrdenie akcie, aby sa nestalo, že používateľ náhodou príde o väčšie množstvo už inde nezaznamenaných dát.

Dáta budú uložené v lokálnej relačnej databáze kvôli ich povahe. Ráta sa aj s pomerne veľkým množstvom dát. Dáta je možné zobecniť do jednotlivých objektov a vzťahov medzi nimi. Pre objekty, s ktorými bude aplikácia pracovať, budú vytvorené samostatné tabuľky. Tieto objekty sú súťaže, sezóny, zápasy, tímy, hráči a iné. Vzťahy medzi objektami budú reprezentované v osobitných tabuľkách. Napríklad vzťah sezóna patrí súťaži alebo hráč sa zúčastnil zápasu. Pre počítanie najrôznejších štatistík by dáta mali byť uložené v čo najobecnejšej forme. To znamená potrebu uložiť dáta tak, aby boli všetky vzťahy medzi nimi jednoznačné, najlepšie aj inak neodvoditeľné. Tým bude štruktúra dát jednoduchšia. Samostatné objekty z dát by mali byť aj čo najviac navzájom nezávislé, čo umožní

komplexnejšie pohľady na dáta a dotazy na nich. Preto bude každá udalosť v zápase ovplyvňujúca nejakú štatistiku zaznamenaná ako jeden objekt. Z toho vyplýva potreba vytvoriť tabuľky pre každú kategóriu udalostí zápasu, ktoré ich budú detailne popisovať. Udalosť zápasu je napríklad gól, vylúčenie alebo striedanie brankára. Každá udalosť by mala mať odkaz na hráčov, ktorých sa týka, a na zápas, v ktorom nastala. Hierarchia podujatí bude nasledovná. Zápas bude priradený sezóne, sezóna súťaži a súťaž športu. Keďže objekty ani ich štatistiky z rôznych športov navzájom nemajú súvis, bude každý šport reprezentovaný vlastnou databázou. Tým sa docieli aj možnosť lepšie rozširovať aplikáciu o nové športy s rôznymi pravidlami. Tímy a hráči sa zúčastňujú zápasov, preto budú existovať dve tabuľky znázorňujúce vzťahy hráč hral zápas a tím hral zápas. Každý hráč ale môže hrať za viacero tímov v rôznych súťažiach alebo tím v rôznych sezónach meniť. Na tento vzťah priradenia hráča na súpiske tímu bude tiež potrebná osobitná tabuľka. Taktiež jednotlivé tímy sa môžu účastniť viacerých súťaží, konkrétne ich sezón. To bude zaznamenávať tabuľka pre vzťah tím sa zúčastnil sezóny. Nakoniec v prípade hokeja a florbalu sa rozlišuje, či sa hráč zúčastnil zápasu ako hráč v poli alebo ako brankár. Táto skutočnosť bude zaznamenaná rozdelením tabuľky pre vzťah hráč sa zúčastnil zápasu na dve tabuľky pre hráča v poli a pre brankára. Databázy tak budú pomerne komplikované a tabuľky budú prepojené mnohými vzťahmi. Niektoré dotazy budú musieť spojiť aj viacero tabuliek do jednej. Preto by mala byť pre vývoj vybraná databáza s lepším výkonom no zároveň s jednoduchou správou. Podrobná schéma finálnej databázy je pre prehľad vyobrazená v prílohe A.2 Zjednodušená databázová schéma.

Pre vývoj aplikácie s užívateľským grafickým rozhraním, ktorá pracuje s dátami uloženými v databáze je dobré použiť niektorý návrhový vzor na to určený. Jednou z možností je vzor Model-view-viewmodel (MVVM). Aplikácie s architektúrou riadiacou sa MVVM oddelujú svoje dáta a logiku od grafického rozhrania a prezentačnej logiky. Vývoj je tak jednoduchší a rýchlejší. Prípadne zmeny v jednej časti kódu nepotrebujú vykonať zmeny v druhej časti. Iné alternatívy sú návrhové vzory Model-view-controller (MVC) a Model-view-presenter (MVP). Všetky tri návrhové vzory pozostávajú z častí view a model, ktoré sú u všetkých prakticky rovnaké. Rozdiel je v tom, ako view a model spolu komunikujú a ako sú prepojené. Pri MVC view môže byť prepojený s viacerými controllermi a naopak. Navyše, view nemá žiadne informácie o controlleri, ale za to o modele áno. Pri MVP

a MVVM je každý view prepojený s jedným presenterom alebo view modelom. Pri MVP, view nevie o modele a presenter aktualizuje model. Rovnako to je aj pri MVVM, kde view o modele nevie, pričom model aktualizuje view model. Navyiac, presenter z MVP vie o svojom view, a view má odkaz na presentera. View model z MVVM nemá žiadnu informáciu o svojom view. Čiže návrh MVVM oddeluje dáta, logiku a grafické rozhranie aplikácie čo najviac, pričom u MVC alebo MVP by sa zmena kódu v jednej časti mohla prejavíť zmenou kódu v druhej časti. Preto bol pre vývoj zvolený práve návrhový vzor MVVM. Pre lepšiu predstavu porovnania vzorov posluži nasledujúci obrázok.



Obrázok 1: Porovnanie návrhových vzorov MVC, MVP a MVVM (zdroj: medium.com)[5]

Väčšina zo všetkých údajov o súťažích a sezóne sú údaje z odohraných zápasov. Tieto údaje sa bežne zaznamenávajú pomocou dokumentu zvanom záznam alebo zápis o zápase. Pri obvyklom priebehu amatérskeho zápasu zapisovateľ pred zápasom skontroluje súpisu tímov vyplnenú kapitánmi. Potom počas zápasu zapisuje všetky jeho udalosti. Jedná sa napríklad o góly alebo vylúčenia. Záznam o zápase má väčšinou podobu viacerých tabuliek, každú určenú pre odlišnú kategóriu udalostí, do ktorých sa údaje zapisujú. Aplikácia bude ponúkať možnosť tieto údaje automaticky spracovať z vyplneného naskenovaného zápisu. Na načítanie dát o zápase z naskenovaného dokumentu bude použitá technológia optického rozpoznávania znakov (OCR). Dokument zápisu o zápase musí byť jednotný. Používateľ bude mať preto možnosť stiahnuť si daný nevyplnený dokument z aplikácie. Predpripravený dokument bude predom vytvorený v programe Microsoft Office Excel a vložený do aplikácie pre stiahnutie. Po vyplnení dokumentu údajmi o

zápase ho používateľ naskenuje a nahrá do aplikácie. Naskenovaný dokument bude potrebné najskôr upraviť, aby bolo rozpoznávanie znakov čo najúspešnejšie. To znamená správne ho otočiť, zväčšiť a odprahovať šum. Túto úpravu aplikácia vykoná automaticky. Používateľ počká na načítanie dát z dokumentu do formulára aplikácie. V prípade chybného rozpoznania znaku alebo chybné vyplneného údaju, bude používateľ upozornený, a údaje bude môcť dolpniť, či opraviť. Na samotné OCR bude použitá niektorá z dostupných knižníc.

Dôležitou súčasťou vedenia športovej súťaže je aj jej propagácia. Používateľ bude preto môcť štatistiky exportovať vo formáte PDF alebo XLSX a následne zverejňovať, alebo v prípade formátu XLSX, s nimi inak manipulovať. Pre každý možný export bude Excel súbor buď kódom generovaný, alebo v zložitejších prípadoch, už predpripravený. Následne sa podľa požiadavky uloží alebo exportuje vo formáte PDF. Rôznymi knižnicami sa súbory PDF dajú generovať aj priamo. Je to ale zbytočné, keďže používateľ má možnosť exportovať aj vo formáte XLSX a ten je ako Excelový súbor možné konvertovať do PDF priamo v aplikácii Excel.

Technológie použité pre vývoj aplikácie neboli zvolené osobitne. Pri výbere bolo brané do úvahy ich vzájomné použitie. Napríklad výber programovacieho jazyka značne obmedzí výber frameworku pre vývoj užívateľského grafického rozhrania. V nasledujúcich podkapitolách sú jednotlivé voľby technológií viac priblížené.

#### **2.4.2 Programovací jazyk – C#**

Aplikácia bude pracovať s rôznymi objektami ako súťaž, sezóna alebo hráč. Pre vývoj bude preto vhodné objektovo orientované programovanie. Na výber sú tri hlavné alternatívy a to C#, Java a C++.

Bol vybraný programovací jazyk C#, keďže aplikácia je primárne určená pre Windows a so C# mám najväčšie skúsenosti. Navyše pre C# existuje mnoho užitočných knižníc a frameworkov. C# tiež obsahuje Language Integrated Query (LINQ), čo je dotazovací jazyk, ktorý bude pri práci s dátami ako sú v aplikácii Sports Manager veľmi užitočný. Pomocou LINQu za využitia lambda funkcií bude kód jednoduchší, prehľadnejší a programátorovi to ušetrí mnoho času. Keďže je C# vyvíjaný firmou Microsoft, dajú sa v ňom aj priamočiaro generovať a upravovať Excelové súbory použité na export štatistík a dát aplikácie. So C# bude možné

použiť framework WPF pre tvorbu grafických rozhraní, ktorý priamo podporuje architektúru MVVM.

Java je veľmi podobná jazyku C#, avšak sú medzi nimi nejaké rozdiely. Rozdiel je napríklad pri generickom programovaní, ktoré je v oboch jazykoch riešené odlišne. Už spomínané lambda funkcie a systém delegátov sú taktiež implementované rôzne. Java je omnoho lepšie prenositeľná medzi rôznymi systémami, kdežto C# je primárne určený pre Windows. S Javou ale nemám skoro vôbec žiadne praktické skúsenosti, a hlavne preto sa nejavila ako hlavná alternatíva.

C++ je rovnako ako Java oveľa lepšie prenositeľný medzi rôznymi systémami narozdiel od C#. Programy napísané v C++ bežia spravidla rýchlejšie ako tie napísané v C#, ale programátor pri ich písaní zväčša strávi viac času. C++ oproti C# potrebuje menej operačnej pamäte pri behu programu. V C++ sa taktiež dá využiť generické programovanie pomocou templateov, a to za kompilačného času, ktorý je ale tiež dlhší ako pri C#. Pri C++ je ale jeho najväčšou nevýhodou, že nemá automatickú správu pamäte. To sa odrazí pri čase programátora strávenom pri vývoji aplikácie, keďže chyby môžu nastávať častejšie ak si pamäť musí spravovať sám.

C# poskytuje najlepšiu alternatívu pre písanie aplikácií s užívateľským grafickým rozhraním pre systémy Windows. Umožňuje využiť framework WPF podporujúci návrhový vzor MVVM, ktorý je detailnejšie popísaný v nasledujúcej podkapitole. Zároveň šetrí čas strávený pri samotnom programovaní a debugovaní programu.

### 2.4.3 Grafické užívateľské rozhranie – WPF

Pre C# existujú tri hlavné možnosti pre výber frameworku pre vytváranie grafického užívateľského rozhrania a to Windows Forms, WPF a UWP (Universal Windows Platform).

WPF podporuje MVVM<sup>5</sup> architektúru, ktorá oddeluje grafické rozhranie od logiky aplikácie. Na vytváranie grafického rozhrania používa značkový jazyk XAML vďaka, ktorému je možné prvky grafického rozhrania jednoducho ľubovoľne modifikovať. WPF využíva metódu data binding, pomocou ktorej sa jednotlivé grafické prvky môžu prepojiť s dátami aplikácie pre intuitívnejšiu synchronizáciu.

---

<sup>5</sup> MVVM – Model-view-viewmodel je softvérový architektonický vzor, ktorý uľahčuje oddelenie vývoja grafického používateľského rozhrania od vývoja back-end logiky. [6]

Windows Forms je už pomerne starý framework, preto sa čoraz menej odporúča pre vývoj modernejších aplikácií. Implementuje udalosťami riadené programovanie. Modifikovanie už existujúcich grafických prvkov alebo tvorba nových je zložitejšie ako pri WPF.

UWP je najmodernejší framework spomedzi všetkých troch. Aj práve kvôli tomu je cieleň iba na tie najnovšie systémy. UWP nie je podporovaný pre Windows 7 a staršie operačné systémy.

Pre vývoj aplikácie bol zvolený WPF framework kvôli podpore MVVM architektúre, relatívne jednoduchému používaniu a podpore ako najnovších operačných systémov, tak aj tých starších.

#### **2.4.4 Databáza – MySQL**

Pre uchovávanie dát aplikácie je potrebná databáza. Rozhodoval som sa medzi relačnými databázami SQLite a MySQL, keďže aplikácia pracuje s objektami, ktoré navzájom súvisia definovanými vzťahmi.

SQLite je takzvaná embedded alebo vstavaná databáza, čo znamená, že nepotrebuje server a beží ako súčasť aplikácie. Databázy sú uložené v samostatných súboroch. Pri väčšom množstve dát je pomalšia ako MySQL.

MySQL je databáza typu klient-server. Je rýchlejšia ako SQLite, lepšia pre väčšie databázy a jednoducho sa používa. Sports Manager môže potenciálne spravovať súťaže s veľkým množstvom dát. Kvôli tomu bola uprednostená MySQL databáza pred SQLite databázou. Pre vývoj aplikácie bola použitá MySQL databáza v rámci balíčka XAMPP<sup>6</sup>. XAMPP navyše obsahuje aj server Apache HTTP Server. Alternatív ku XAMPP je viacero (WampServer<sup>7</sup>, MAMP<sup>8</sup>...). Nezáleži na tom, ktorá je zvolená. Podstatné je mať lokálne nainštalované MySQL.

#### **2.4.5 Optické rozpoznávanie znakov – Python**

Aplikácia umožňuje načítať dáta o zápase z naskenovaného dokumentu. O to sa stará skript napísaný v jazyku Python kvôli využitiu jeho knižníc tretích strán

---

<sup>6</sup> XAMPP – multiplatformný softwarový balíček vyvinutý firmou Apache Friends [7]

<sup>7</sup> WampServer – balík pre Windows, obsahuje Apache server, databázu MySQL a jazyk PHP

<sup>8</sup> MAMP – balík riešení zahŕňajúci aj webový server a MySQL databázu

určených presne na takéto úkony. V jazyku Python je totiž omnoho jednoduchšie písať skripty na optické rozpoznávanie znakov ako v C#. Pre jazyk C# neexistuje toľko užitočných knižníc pre spracovanie obrazu. Skript využíva knižnice Opencv, SciPy, NumPy a Python-tesseract. Pre C# síce existujú knižnice NumSharp a Numpy.NET, ktoré implementujú knižnicu NumPy v jazyku C#, ale jedná sa o totožnú vec ako v Pythone. Python je dobrým jazykom na prácu s viacerými knižnicami a píše sa v ňom oveľa rýchlejšie a prehľadnejšie ako v C#. Skript na OCR by teda bolo možné implementovať aj priamo v C#, no v jazyku Python to je jednoduchšie a rýchlejšie.

Taktiež je potrebné mať nainštalovaný Tesseract OCR[8] pre rozpoznávanie jednotlivých znakov v dokumente.

## **2.5 Použité knižnice**

### **2.5.1 Softwarové knižnice pre Python**

Opencv-python – Knižnica použitá pre spracovanie obrázkov pre optické rozpoznávanie znakov.

SciPy – Obsahuje mnoho algoritmov pre optimalizáciu, interpoláciu algebraické rovnice a ďalšie. V projekte sa využíva pre otáčanie obrázkov.

Python-tesseract – Je to knižnica pre optické rozpoznávanie znakov. Jedná sa o obalenie Tesseract-OCR Engine od firmy Google.[8]

NumPy – Rozsiahla matematická knižnica. V projekte bola použitá najmä pre prácu s poľami.

### **2.5.2 Iné softwarové knižnice**

MySQL.Data – Pre spojenie s databázou bola použitá knižnica MySQL.Data od Oracle.

LiveCharts.WPF – Aplikácia by mala vedieť generovať rôzne grafy zo štatistik. Na to bola použitá knižnica LiveCharts.WPF [9] pre jej

jednoduchosť. Knižnica umožňuje zobrazovať viacero typov grafov a vďaka zakomponovaniu objektu grafu ako prvku WPF je možné grafy jednoducho exportovať do formátu PNG alebo JPEG.

## 3. Implementácia

### 3.1 Architektúra aplikácie

Aplikácia sa riadi návrhovým vzorom Model-view-viewmodel (MVVM) pre WPF aplikácie. Oddeluje logiku aplikácie od užívateľského rozhrania. Používa data binding a command<sup>9</sup> namiesto udalostí.[10]

Model obsahuje dáta aplikácie. V Sports Manager to sú napríklad triedy Player, Team, Competition, Season a podobne. Model pomocou týchto tried popisuje štruktúru a význam dát. Dáta sú štruktúrované v inštanciách tried, ktoré okrem dát obsahujú metódy na zmenu dát alebo špecifický prístup k nim.

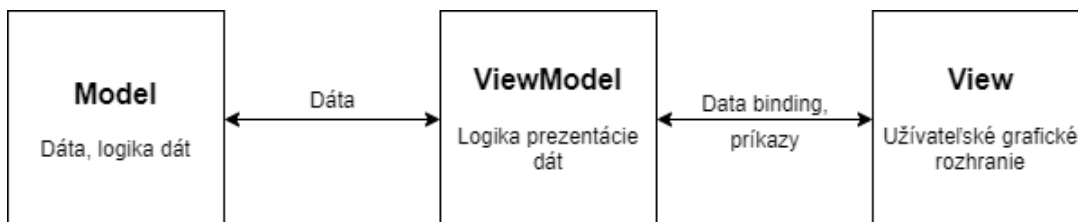
View popisuje grafické užívateľské rozhranie jazykom XAML, ako napríklad hlavné okno aplikácie a všetky ostatné ovládacie prvky. Zobrazuje reprezentáciu modelu na obrazovke a prijíma interakcie od užívateľa. Pomocou data bindingu prepája a synchronizuje dáta skrz view model.[6] Neobsahuje samotnú logiku aplikácie a vďaka tomu môže byť upravovaný bez nutnosti zmeny kódu logiky aplikácie, čo platí aj naopak. Jazyk XAML je značkový jazyk a je ním možné vytvárať špecializované grafické prvky a jednoducho prepájať dáta a logiku aplikácie s jej grafickým rozhraním. Už spomenutý data binding automaticky aktualizuje dáta alebo vzhľad aplikácie. Môže fungovať buď ako jednosmerné alebo ako dvojsmerné prepojenie. Pri jednosmernom prepojení prvky grafického rozhrania získajú potrebné dáta z view modelu a zobrazia ich. Pri dvojsmernom prepojení sa prepísaním dát zobrazených v grafickom rozhraní prepíšu aj dáta aplikácie uložené vo view modeli. Dáta môžu byť taktiež prepojené jednosmerne opačným smerom, kedy zmena v grafickom rozhraní zmení dáta ale osobitná zmena dát sa už v grafickom rozhraní neprejaví. Prvky grafického rozhrania sú navzájom usporiadané v stromovitej štruktúre. Každý XAML prvok má vlastnosť **DataContext**, na ktorú sa prípadne jeho potomkovia môžu odkazovať a mať prepojenie s dátami o úrovne vyššie. To sa môže využiť napríklad pri zobrazovaní objektu sezóny obsahujúcom zoznam hráčov, kde sa pre každého hráča zobrazia informácie o ňom a navyše aj meno sezóny, na ktoré

---

<sup>9</sup> Command – je návrhový vzor, ktorý zapuzdruje informácie o akcii do objektu pre jej neskoršie zavolanie

je odkazované z úrovne vyššie. Pri prepojení s príkazmi (commands, triedy implementujúce rozhranie *ICommand*) je možné do view modelu odovzdať parametre, s ktorými sa má príkaz vykonať. Pri prvku **Button** na to slúži vlastnosť **CommandParameter**. Samotný binding sa skladá z cieľa, cielenej vlastnosti, zdrojového objektu a cesty k hodnote zdrojového objektu.[11] Cieľ je konkrétny grafický prvok, napríklad **TextBox**. Cielená vlastnosť je niektorá z vlastností prvku, napríklad *Text*. Zdrojový objekt je vlastnosť z view modelu a cesta k jeho hodnote je názov danej vlastnosti objektu, napríklad objekt **Player** a jeho vlastnosť **FisrtName**.

View model reprezentuje stav aplikácie a prepája view s modelom. Nemá referenciu k view a dáta sú prepojené bindingom. Všetky dáta, ktoré majú byť prepojené s grafickým rozhraním musia byť implementované ako vlastnosť (property) view modelu. Objekty dát musia byť inštancie triedy implementujúcej rozhranie *INotifyPropertyChanged*. Kolekcie dát musia implementovať rozhranie *INotifyCollectionChanged*. WPF ponúka predimplementovanú triedu **ObservableCollection<T>**. Prepojené s pohľadom môžu byť aj inštancie príkazov, ktoré sa po akcií v grafickom rozhraní vykonávajú. Aby sa zmena vlastnosti vo view modeli prejavila aj v pohľade, musí táto vlastnosť po jej zmene zavolať metódu **OnPropertyChanged()** danej inštancie, ktorá pohľad upozorní aby sa aktualizoval.



Obrázok 2: Diagram architektúry Model-view-viewmodel

### 3.2 Databázová schéma

Vždy pri spustení aplikácia skontroluje, či všetky potrebné databázy a tabuľky existujú, a ak nie, vytvorí ich. To je výhodné hlavne pri prvom spustení, keďže používateľ nijak zvlášť nemusí konfigurovať databázu.

Databáza pre jeden šport pozostáva z 26 tabuliek. Z toho tabuľka country je spoločná a nachádza sa v databáze sports\_manager. Pre každý šport je samostatná databáza s menom daného športu (ice\_hockey, floorball). Databázy pre hokej

a florbal sú totožné. Zjednodušená databázová schéma je k dispozícii v prílohe A.2

Zjednodušená databázová schéma.

Hlavné entity sú súťaže (competitions), sezóny (seasons), zápasy (matches), tímy (team) a hráči (players). Každá sezóna patrí niektorej zo súťaží a každý zápas niektorej zo sezón. Zápasy môžu byť ďalej priradené skupine (groups), kvalifikácii a kolu (rounds).

V zápase stĺpec `qualification_id` odkazuje na danú kvalifikáciu inak je rovný -1 ak patrí buď do play-off alebo skupiny. Stĺpec `round` značí identifikačné číslo (id) hracieho kola alebo kolo v pavúku (brackets). Stĺpec `serie_match_number` značí poradie v sérii. Stĺpec `bracket_index` poradie v kole pavúka od vrchu. Stĺpec `bracket_first_team` zas vrchný tím pri zobrazení série. Zápas patrí do kvalifikácie ak sa `qualification_id` nerovná -1, do skupiny ak sa `bracket_index` rovná -1 a do play-off ak sa `qualification_id` rovná -1 a zároveň `bracket_index` nerovná -1.

Tímy sú priradené k sezónam pomocou tabuľky `team_enlistment` a taktiež hráči sú priradení k sezónam a jednotlivým tímom pomocou tabuľky `player_enlistment`.

Dáta o zápase pozostávajú z viacerých tabuliek. Základne dáta sú v tabuľke `matches`. V tabuľkách `player_matches` a `goalie_matches` sú k jednotlivým zápasom priradení hráči a brankári, ktorí v nich hrali. Ďalej sú tabuľky s udalosťami v zápase slúžiace pre štatistiku (v prílohe A.2 Zjednodušená databázová schéma vyznačené v červenom rámečku) ako góly, vylúčenia, trestné strielania, shutouty (čisté kontá) a tak ďalej. V každom zázname sa nachádzajú dáta, o ktorý zápas sa jedná (`match_id`), o perióde, čase, zúčastnených sa hráčov a tímov, aktuálnom skóre, poradí v zápase a špecifické dáta podľa typu udalosti. Tabuľka `game_state` reprezentuje stavy hry, čiže úseky hry s udaným začiatkom a koncom, v ktorých bol po celý čas rovnaký stav skóre a počet hráčov v hre.

Tabuľky `strength`, `penalty_type`, `penalty_reason` ale aj `country` sú stále a obsahujú dáta o každom z typov počtu hráčov v hre, vylúčení, a nakoniec v prípade `country`, o štátoch sveta.

V prípade, že databáza nie je z akéhokoľvek dôvodu dostupná, sa používateľovi zobrazí vyskakovacie okno, ktoré ho na to upozorní. Žiadne dáta sa v okne aplikácie nezobrazia.

## 3.3 Hlavné triedy aplikácie

### 3.3.1 Models

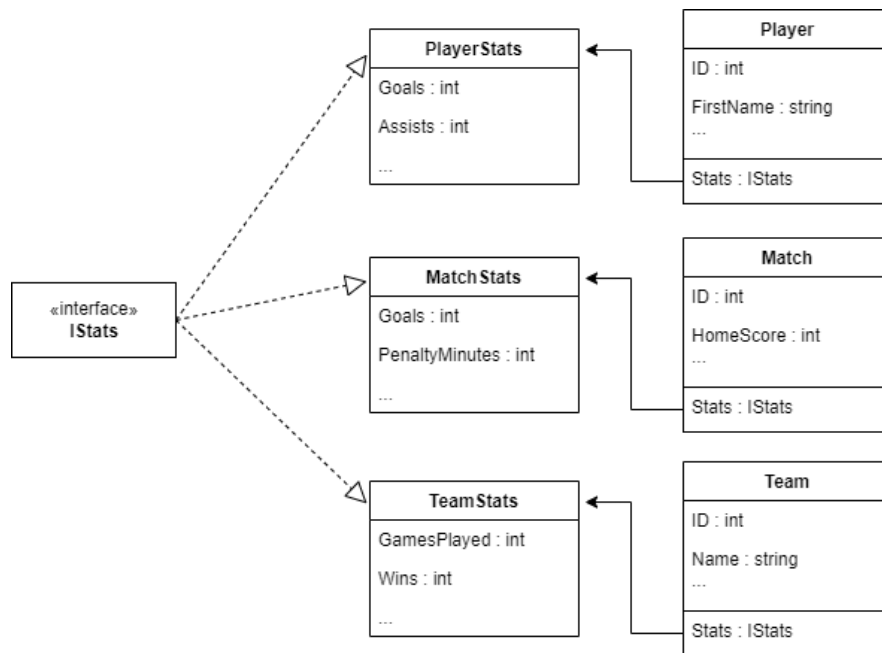
Modely predstavujú hlavné objekty aplikácie uložené v databáze reprezentované inštanciami tried ako sú napríklad **Player** (hráč), **Team** (tím), **Competition** (súťaž), **Season** (sezóna), **Match** (zápas), **Bracket** (pavúk) a ďalšie. Slúžia na reprezentáciu dát z databázy a na prácu s nimi. Každá z tried reprezentujúca nejaký model dedí od triedy **NotifyPropertyChanged**, ktorá implementuje rozhranie **INotifyPropertyChanged**. Vďaka tomu sa dáta inštancií tried automaticky aktualizujú pri prepojení data bindingom medzi view modelom a view.

Trieda **Competition** obsahuje vlastnosti **ID**, **Name**, **Info** a **ImagePath**. Od nej potom dedia triedy **Season** a **Team**. Trieda **Season** ďalej obsahuje odkaz na súťaž, ktorej je súčasťou, a vlastnosti popisujúce jej formát. **Team** navyše obsahuje vlastnosť o jeho vzniku **DateOfCreation** typu `DateTime`, odkaz na štát pôsobenia **Country** a vlastnosť **Status** typu `boolean` reprezentujúcu to, či je tím ešte súťažne aktívny. Obe triedy **Season** a **Team** obsahujú aj vlastnosť **Stats** implementujúcu rozhranie **IStats**, ktoré bude podrobnejšie vysvetlené nižšie.

Triedy **Group**, **Country**, **PenaltyReason**, **PenaltyType**, **Position** a **Strength** reprezentujú adekvátne objekty z databázy a obsahujú vlastnosti totožné so stĺpcami jednotlivých tabuliek v databázy. Trieda **Group** navyše obsahuje zoznam obsiahnutých tímov v danej skupine.

Triedy **Player** a **Match** reprezentujú hráča a zápas a okrem vlastností, ktoré sú totožné so stĺpcami tabuliek `player` a `matches` v databáze, obsahujú aj vlastnosť **Stats** implementujúcu rozhranie **IStats**, rovnako ako triedy **Team** a **Season**.

Rozhranie **IStats** nedefinuje žiadne pravidlá a slúži iba na oddelenie štatistík entít od ich základných údajov. Tým pádom, môže ten istý objekt v dvoch rôznych pohľadoch obsahovať rôzne štatistiky podľa toho, čo sa má práve zobrazovať. V súbore `Stats.cs` sa okrem rozhrania **IStats** nachádzajú aj triedy reprezentujúce a počítajúce štatistiky jednotlivých objektov. Sú to napríklad triedy **PlayerStats**, **TeamStats** alebo **SeasonStats**. Inštancie týchto tried budú potom odkazované v konkrétnych objektoch, ktorým prislúchajú.



Obrázok 3: Znáozornenie použitia tried implementujúcich rozhranie IStats

### 3.3.2 Views

Pohľady predstavujú grafické užívateľské rozhranie, s ktorým užívateľ interaguje. Každému pohľadu (view) je udelený špecifický view model, s ktorým je prepojený pomocou techniky zvanej binding. Pohľad **SportView** definuje väčšinu týchto párov prepojení.

Hlavné okno je reprezentované pohľadom **MainWindow**. To v sebe môže zobrazovať buď pohľad **SportsSelection** pre výber športu alebo **SportView** pre konkrétny šport.

Pohľad **SportView** sa delí na dve časti, pričom v ľavej časti vždy zobrazuje navigáciu implementovanú pohľadom **NavigationBarView** a vo zvyšku sa zobrazujú už konkrétne pohľady ako napríklad **PlayerView** alebo **ScheduleView**.

### 3.3.3 View models

View model budem nazývať model pohľadu. Modely pohľadu slúžia na prepojenie grafického rozhrania s logikou a dátami aplikácie.

V aplikácii Sports Manager sa modely pohľadu môžu rozdeliť na viaceré typy. Všetky implementujú rozhranie **INotifyPropertyChanged**, aby mohli so sebou prepojený pohľad upozorniť na zmenu, čím sa pohľad aktualizuje.

Hlavné modely pohľadu sú **MainViewModel** pre samotné hlavné okno aplikácie, **SportsSelectionViewModel** pre výber športu a **SportViewModel** pre pohľad športu, ktorý obsahuje ďalší, práve prehliadaný model pohľadu a model pre navigáciu.

Model pohľadu **NavigationBarViewModel** sa stará o prepínanie medzi pohľadmi a obsahuje príkazy na to určené. Je vždy súčasťou inštancie **SportViewModel**.

Modely pohľadov predstavujúce samotné objekty ako hráč či súťaž dedia od spoločnej triedy **TemplateEntityViewModel**.

Od triedy **TemplateEditViewModel** dedia všetky modely pohľadov, ktoré slúžia na editáciu objektov, napríklad zmenu mena hráča alebo pridanie loga tímu.

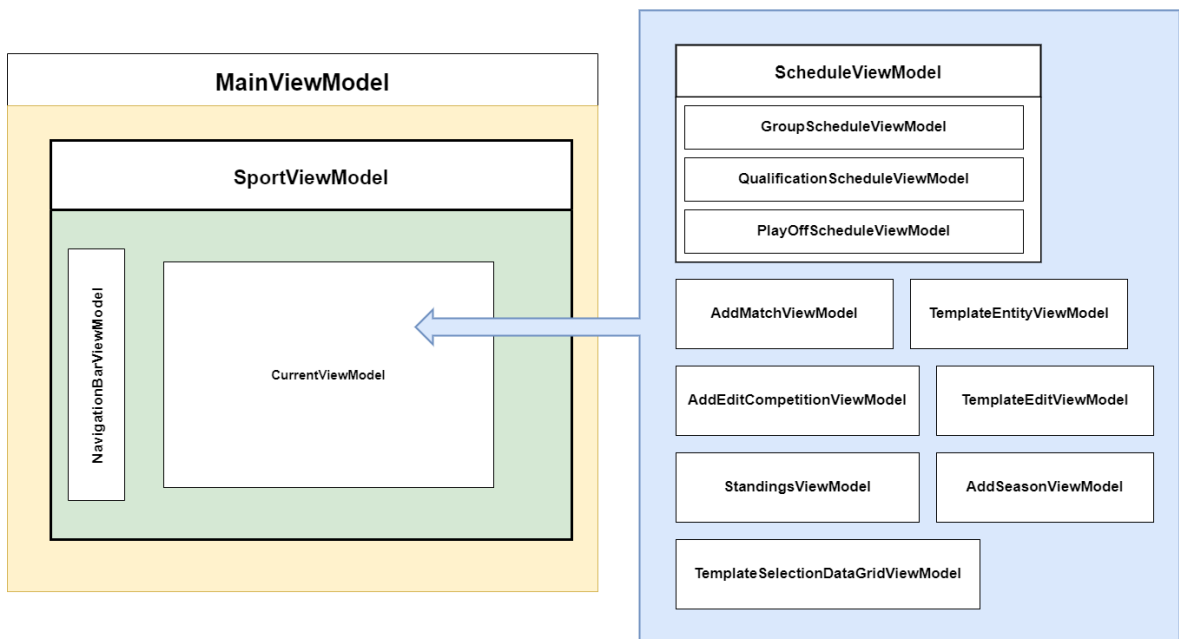
Špecifickou kategóriou sú modely pohľadov predstavujúce zoznamy objektov zobrazujúce aj ich štatistiky. Tie dedia od triedy **TemplateSelectionDataGridViewModel**.

Prehľad zápasmi sezónou umožňuje model pohľadu **ScheduleViewModel**, v ktorom sa prepína medzi modelmi pohľadu pre jednotlivé dostupné časti sezóny, a to **GroupScheduleViewModel**, **QualificationScheduleViewModel** a **PlayOffScheduleViewModel**. Posledné dve navyše dedia od spoločnej triedy **TemplateBracketScheduleViewModel**.

Dve najrozsiahlšie modely pohľadu sú **AddSeasonViewModel** a **AddMatchViewModel**, ktoré obsahujú formuláre na pridanie novej sezóny a nového zápasu.

Model pohľadu **AddEditCompetitionViewModel** implementuje naraz pridanie novej súťaže aj jej editáciu, pretože sú si veľmi podobné.

Posledným modelom pohľadu je **StandingsViewModel**, ktorý načíta prípadné dáta o skupinách základnej časti a spočíta údaje pre grafy ilustrujúce štatistiky tímov v sezóne.



Obrázok 4: Hierarchia pohľadov modelov

### 3.3.4 Commands

Príkazy (commands) slúžia v architektúre MVVM hlavne na riadenie a spúšťanie príkazov a funkcií používateľom, kedy sa prepojením cez pohľad zavolá príkaz vytvorený modelom pohľadu.

Príkaz je inštancia triedy implementujúcej rozhranie  *ICommand*  zahrnutá v modeli pohľadu ako jeho vlastnosť. Obsahuje delegáta metódy, ktorú má neskôr zavolať. Pri inšancovaní view modelu sa inšancujú aj jeho príkazy s daným delegátom ako jeho akciou.

Sports Manager implementuje dva typy príkazov. **NavigateCommand** prepína pohľady a nastavuje aktuálne vybraný šport, súťaž a sezónu. Pomocou príkazu **RelayCommand** je možné zavolať ľubovoľnú metódu, pričom sa používa napríklad na volanie metód implementujúcich export dát do súborov alebo kontrolu vstupu pri pridávaní nového zápasu. Oba príkazy dedia od triedy **CommandBase**.

### 3.3.5 Ostatné pomocné triedy

Ďalšie pomocné triedy zaisťujú rôzne funkcie aplikácie. Trieda **DatabaseHandler** obsahuje pomocné metódy pre prácu s databázou. Trieda **ImageHandler** zase pomocné metódy pre načítanie a ukladanie obrázkov. **NotifyPropertyChanged** implementuje rozhranie  *INotifyPropertyChanged*  a od

tejto triedy potom dedia všetky modely pohľadu a mnohé modely objektov pre predídanie opakovania sa kódu. Skoro všetky metódy zaisťujúce export dát sa nachádzajú v statickej triede **Export**.

### 3.4 Ukladanie obrázkov

Používateľ má možnosť ku každej súťaži, sezóne, tímu a hráčovi priradiť obrázok, či už sa jedná o logo alebo fotografiu.

Obrázky sú ukladané lokálne do jedného z priečinkov:

- Competition\_Logos pre súťaže
- Player\_Photos pre hráčov
- Season\_Logos pre sezóny
- Team\_Logos pre tímy

Tieto priečinky sa nachádzajú na ceste "C:\Users\{POUŽÍVATEL}\AppData\Roaming\SportsManager\Images" (prípadne iný názov disku).

Všetky tieto priečinky sa v prípade, že neexistujú, vytvoria pri spustení aplikácie. Jednotlivé obrázky sú uložené pod názvom daného športu a identifikačného čísla objektu. Napríklad logo florbalového tímu s identifikačným číslom 25 v databáze bude uložené ako "C:\Users\{POUŽÍVATEL}\AppData\Roaming\SportsManager\Images\Team\_Logo\floorball25.png" (prípadne s príponou .jpg).

### 3.5 Prehľad štatistík

Všetky údaje a záznamy sú rozdelené hierarchicky. Hlavné rozdelenie je podľa športu, kde dva rôzne športy spolu nijakým spôsobom nesúvisia a všetky ich údaje ako aj štatistiky sa navzájom neovplyvňujú. Na najvyššej úrovni sa šport skladá zo súťaží a tie z jednotlivých sezón. Zápasy sú priradené sezónam. Všetky tímy a hráči sú viditeľní v rámci jedného športu pre všetky súťaže v prípade zápisu tímu alebo hráča do novej sezóny.

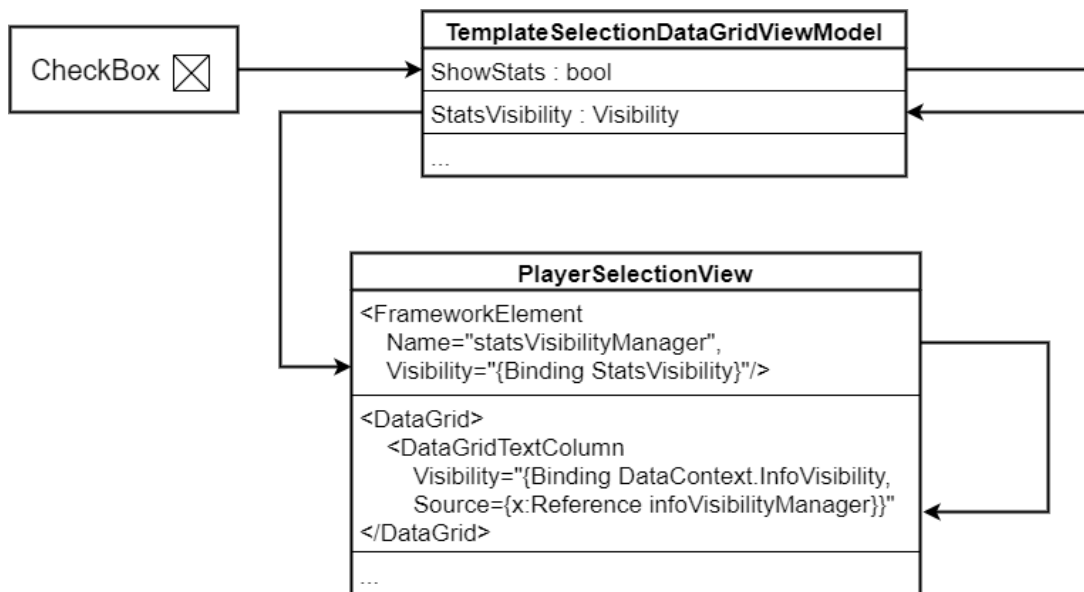
Navigačnou lištou na ľavej strane okna, ktorá je po výbere športu vždy dostupná, je možné navigovať sa medzi spoločnými pohľadmi športu a súťaží alebo

samostatnými pohľadmi individuálnych súťaží či líg. To znamená, že používateľ si môže prezerať štatistiky a dáta v rámci celého športu, kde sa štatistiky spočítajú skrz všetky súťaže. Rovnako si používateľ môže pozrieť akumulované štatistiky pre konkrétnu súťaž cez všetky jej sezóny. Nakoniec pri výbere jedinej sezóny sa štatistiky počítajú a zobrazujú len pre ňu.

K dispozícii sú prehľady sezón, zápasov, tímov, hráčov a brankárov v tabuľkovej forme implementované triedami dediacimi od triedy **TemplateSelectionDataGridModel<T>**. Na to sa používa WPF prvok **DataGrid**, ktorý sa naplní údajmi z prepojeného view modelu načítaných z databázy pri vytvorení pohľadu. Tabuľky v riadkoch obsahujú zoznam daných objektov, pričom zároveň slúžia aj ako odkaz na samostatný pohľad objektu (tímu, hráča a podobne) po dvojito kliknutí myšou.

Tabuľky je možné exportovať vo formáte XLSX alebo PDF. Dáta sa môžu vopred vytriediť alebo obmedziť na konkrétne položky (to znamená, či majú obsahovať aj všeobecné informácie o objekte alebo štatistiky).

Na filtrovanie, ktoré dáta sa majú zobrazovať sa používajú grafické prvky **CheckBox**, ktoré majú hodnotu "true" alebo "false". Prepínajú tým hodnotu vo vlastnostiach view modelu **ShowPhoto**, **ShowInfo**, **ShowStats** a **ShowExtendedStats**, ktoré sú typu bool. Tie zas pri nastavení hodnoty prepínajú viditeľnosti **PhotoVisibility**, **InfoVisibility**, **StatsVisibility** a **ExtendedStatsVisibility** typu **Visibility** medzi hodnotami **Visibility.Visible** a **Visibility.Collapsed**. Tieto viditeľnosti sú pomocou data bindingu prepojené s viditeľnosťami prvkov **FrameworkElement** v danom pohľade, ktoré slúžia ako správca viditeľností pre všetky stĺpce **DataGridu**, ktoré zobrazujú dáta danej kategórie. Tieto stĺpce sa bindingom cez **DataContext** odkazujú na viditeľnosť svojej kategórie vo view modeli, pričom ako zdroj je udaný konkrétny **FrameworkElement**. Je to tak riešené preto, lebo stĺpce **DataGridu** od neho nededia **DataContext** a obyčajné prepojenie bez udania zdroja by nefungovalo.



Obrázok 5: Prepínanie viditeľnosti stĺpca v Datagrid

Pri exporte tabuľky sa vždy vytvorí nový Excel dokument, v ktorom sa vytvorí a naplní tabuľka s údajmi. Tento dokument sa v prípade potreby exportuje do formátu PDF a následne uloží na užívateľom zvolené miesto. O export sa stará metóda **Exports.ExportTable(DataGrid dataGrid, string format, int? exportTop)**, pričom parametrami sú daný DataGrid so všetkými údajmi, reťazec konkretizujúci formát exportovaného súboru a počet prvých riadkov DataGridu, ktoré sa majú exportovať, ktorý používateľ zadá v pohľade do prvku **TextBox** (v prípade, že je prázdny sa exportujú všetky riadky). Predať metóde aj inštanciu DataGridu je potrebné kvôli tomu, aby sa stĺpce a riadky exportovali v správnom poradí. Používateľ totiž môže stĺpce predom prehádzať a dáta, riadky, zotriediť ľubovoľným spôsobom, čo je vlastnosť prvku DataGrid. Zistiť aktuálne poradie stĺpcov a riadkov je možné iba cez danú inštanciu DataGridu.

### 3.6 Vytvorenie zápasu

Aplikácia kladie dôraz na čo najväčšiu jednoduchosť a minimalizáciu vstupu, hlavne v prípade zaznamenávania nového zápasu. To znamená, že užívateľ by nemal udávať žiadne nadbytočné dáta, ktoré by bolo možné odvodiť z ostatných, už uvedených údajov. Jeden z príkladov je údaj o tom, či daný gól padol za početnej prevahy (v presilovej hre) alebo v početnej nevýhode, či za vyrovnaného stavu hráčov v hre. Keďže užívateľ musí zadať aj samotné vylúčenia hráčov a čas kedy

nastali, aplikácia si sama vypočíta koľko hráčov bolo v hre počas toho ako padol gól. Navyše sa vypočíta, že daná presilová hra skončila strelením gólu (ak sa jedná o trest, ktorý končí strelením gólu zvýhodneného tímu).

### 3.6.1 Nutné údaje

Najprv je potrebné definovať, ktoré údaje sú nutné (nedajú sa žiadnym iným spôsobom odvodiť) a ktoré je možné odvodiť.

Hlavné údaje zápasu, informácie o ňom, ako sú dátum a čas stretnutia, či počet periód a ich dĺžka, je samozrejme nutné explicitne zadať. Ďalej rovnako súperiace tímy (v prípade zápasu v pavúku, kde sú tímy dané výsledkom z predošlého kola to nutné nie je) a zúčastnení hráči. Zoznam hráčov, čiže súpisky tímov, sa načítajú podľa vybraných tímov automaticky.

Nutné štatistiky sú:

*Góly* (goals) s údajom o strane, ktorá skórovala (domáci, hostia), číslom strelca, číslom asistenta, časom hry a typom gólu (vlastný gól, trestné strielanie).

*Vylúčenia* (penalties) s údajom o strane, číslom vylúčeného hráča, časom hry, typom a dôvodom trestu.

*Trestné strielania* (penalty shots) s údajom o strane, ktorá skórovala, číslom strelca, časom hry, znaku či padol gól.

*Zmeny brankárov* (goaltender shifts) s údajom o strane, číslom brankára, časom začiatku a časom konca striedania.

*Time-outy* (time-outs) s údajom o strane a čase hry.

*Samostatné nájazdy* (shootout) s číslom strelca, číslom brankára a znakom či padol gól.

Pri výbere hráča, sú navyše pre jednoduchosť k dispozícii vždy len zúčastnení hráči daného tímu. Rovnako nie je potrebné udávať periódu, pretože dáta sú už vopred rozdelené do periód. Taktiež nie je potrebné udávať číslo série nájazdu a stranu tímu, keďže sa striedajú.

Všetky ostatné údaje sa dajú odvodiť. Sú to napríklad záznamy ako konce presilových hier, získané body hráčov, čisté kontá brankárov, góly v presilových hrách a oslabeniach, góly do prázdnej brány, víťazný gól, dĺžka presilovej hry,

využitie presilovej hry a iné. Zároveň je pre každú udalosť automaticky zaznamenané, za akého počtu hráčov a za akého stavu skóre nastala.

### 3.6.2 Stav hry

Na automatické počítanie a odvodenie niektorých štatistík je potrebné vopred vypočítať priebeh hry a uložiť si chronologicky zmenu stavu hry. Stav hry zachytáva počet hráčov v hre (vrátane brankárov), aktuálnu periódu a aktuálne skóre (počet gólov). V aplikácii stav hry obsahuje aj čas, v ktorý sa stav hry zmenil na ten aktuálny a taktiež čas kedy sa aktuálny stav hry zmenil na iný. Takto je možné reprezentovať súvislé úseky hry s rovnakým stavom.

Pri rovnakom čase dvoch rôznych udalostí môže nastať problém. Ako príklad si vezmime, že bol odvolaný brankár z brány v čase 19:00. V čase 19:30 súper strelil gól do prázdnej brány, časomiera sa zastavila a brankár sa vrátil späť do brány v rovnaký čas 19:30. Obe udalosti, gól aj návrat brankára, majú rovnaký čas. Podľa času samotného nie je možné rozhodnúť, ktorá udalosť nastala skôr. To je ale veľmi dôležité vedieť, keďže situácia mohla byť aj naopak, čo sa v zápasoch stáva, a gól mohol padnúť tesne po návrate brankára do hry (mohla byť ešte čiastočne odokrytá brána). To ale vedie k inej štatistike. Jednak v rozdielnom počte inkasovaných gólov brankára a aj v type gólu. Kvôli takýmto prípadom je nutné udalosti zápasu zoradiť, označiť ich poradovým číslom podľa chronologického poradia v akom reálne v zápase nastali. Pre minimalizáciu vstupu to nemusí používateľ robiť pre každú udalosť. Po vyplnení nutných údajov používateľ tlačidlom "Process" zavolá rovnomennú metódu, ktorá skontroluje ak nejaké udalosti nastali v rovnaký čas a používateľ a vyzve, aby ich poradie skontroloval a prípadne prehodil.

Stále však môže nastať kolízia v poradí udalostí, a to pri koncoch presilových hier. Môže sa napríklad stať, že v čase keď jedna presilová hra skončí, či už gólom alebo vypršaným časom, nejaká ďalšia presilová hra začne. Podľa času opäť nie je možné určiť, či vylúčenie nastalo v poslednej sekunde predchádzajúcej presilovej hry alebo tesne po jej konci. Pre kontrolu takýchto kolízií používateľ stlačí tlačidlo "Check", ktoré zavolá rovnomennú metódu, ktorá spočíta stavy hry a skontroluje, či nejaká kolízia nastala. V prípade, že nastala kolízia, bude používateľ opäť upozornený a nejasné udalosti môže priradiť daným presilovým hrám alebo ich zaradiť až za ne. Nakoniec je možné zápas uložiť.

Stav hry reprezentuje trieda **State** obsahujúca informácie o aktuálnej perióde, počte hráčov v hre v oboch tímoch, skóre a čas začiatku a konca stavu hry.

Metóda **Process()** v triede **AddMatchViewModel** všetky udalosti zápasu načíta do spoločného zoznamu, ktorý vytriedi podľa času, v ktorom nastali a priradí im poradové čísla. Nasladne skontroluje časové kolízie.

Metóda **Check()** do zoznamu udalostí pridá aj kocne periód a postupne prechádza všetky udalosti chronologicky, pričom si uchováva momentálny stav hry. Simuluje tak priebeh hry. Taktiež si uchováva zoznam momentálne prebiehajúcich presilových hier a zoznam už ukončených presilových hier. Pri každej udalosti najskôr skontroluje, či už náhodou práve prebiehajúce presilové hry neskončili. O to sa stará metóda **CheckEndOfPenalties()**, ktorá v prípade ukončenia presilovej hry uloží momentálny stav hry a posunie sa na nový, práve prebiehajúci. Vďaka tomu sa pre udalosť uložia dáta z aktuálneho stavu hry ako skóre a počet hráčov v hre. Ak je udalosť presilovou hrou, pridá sa do zoznamu práve prebiehajúcich presilových hier. Ak je udalosťou gól, upraví sa skóre momentálneho stavu a ukončia sa prípadné presilové hry podľa pravidiel hry. Nakoniec po prejdení všetkých udalostí a prepočítaní všetkých stavov hry sa skontrolujú časové kolízie.

### 3.6.3 Pravidlá hry

Pre počítanie štatistík a stavov hry je potrebné poznať presné pravidlá hry, teda športu. Momentálne zabudované dva športy v aplikácii sú florbal a ľadový hokej, ktoré majú z hľadiska štatistík a priebehu hry zaznamenanom aplikáciou Sports Manager identické pravidlá. Skutočné podrobné pravidlá týchto športov sa samozrejme líšia vo viacerých detailoch akými sú napríklad existencia voľných úderov vo florbale zatiaľ čo v hokeji daný pojem neexistuje alebo postavenia mimo hry, ktoré sú naopak prítomné v hokeji ale nie vo florbale. Tieto a iné rozdiely ale prácu s údajmi a štatistikami v aplikácii nijakým spôsobom neovplyvňujú.

Dôležité pravidlá sa týkajú presilových hier, konkrétne ich typu a spôsobu ukončenia. Podľa týchto pravidiel sa počíta stav hry ako aj samotná presilová hra. V oboch športoch sa jedná o rovnaké pravidlá.

Existuje šesť typov vylúčení. Prakticky, po každom z nich nasleduje presilová hra ale napríklad v prípade vylúčenia do konca zápasu sa udeľuje aj ďalšie menšie

vylúčenie (napríklad vylúčenie na dve alebo päť minút), ktoré priamo spôsobuje presilovú hru, pretože vylúčenie do konca zápasu samo osebe túto vlastnosť nemá.

Typy vylúčení:

- Vylúčenie na 2 minúty (menší trest)
  - Štatisticky sa zaznamenávajú dve trestné minúty.
  - Nasleduje dvojminútová presilová hra o jedného hráča.
  - Presilová hra končí po strelení gólu z hry zvýhodneným tímom.
  - Hráč po odpykaní trestu môže nastúpiť do zápasu.
  
- Vylúčenie na 5 minút (väčší trest)
  - Štatisticky sa zaznamenáva päť trestných minút.
  - Nasleduje päťminútová presilová hra o jedného hráča.
  - Presilová hra nie je ukončená gólom.
  - Hráč už nemôže nastúpiť do zápasu.
  
- Vylúčenie na 10 minút (osobný trest)
  - Štatisticky sa zaznamenáva desať trestných minút.
  - Nenasleduje presilová hra.
  - Hráč nemôže nastúpiť do hry desať minút.
  
- Vylúčenie do konca zápasu 1
  - Štatisticky sa zaznamenáva dvadsať trestných minút.
  - Nenasleduje presilová hra.
  - Jedná sa o osobný trest, hráč už nemôže nastúpiť do zápasu.
  
- Vylúčenie do konca zápasu 2
  - Štatisticky sa zaznamenáva dvadsať trestných minút.
  - Nenasleduje presilová hra.
  - Jedná sa o osobný trest, hráč už nemôže nastúpiť do zápasu.
  - Hráč nemôže nastúpiť ani na najbližší zápas.

- Vylúčenie do konca zápasu 3
  - Štatisticky sa zaznamenáva dvadsať trestných minút.
  - Nenasleduje presilová hra.
  - Jedná sa o osobný trest, hráč už nemôže nastúpiť do zápasu.
  - Hráč nemôže nastúpiť na najbližší, komisiou určený, počet zápasov.

Ďalej platí pravidlo, že v jednom okamihu môžu prebiehať najviac dve presilové hry. Takže najviac môže nastať početná výhoda o dvoch hráčov (okrem odvolania brankára). Preto ak už prebiehajú dve presilové hry, všetky nasledujúce presilové hry sa zaradia do poradia a najbližšia z nich začne v momente, ak jedna zo súčasných skončí.

Podľa vyšších pravidiel sa teda odvodí, že presilové hry môžu trvať dve alebo päť minút s tým, že gólom končí iba presilová hra na dve minúty. Aby gól ukončil presilovú hru, musí padnúť z hry, čo znamená, že góly z trestných strielaní ani odložených vylúčení presilovú hru neukončia. Vlastný gól ale naopak presilovú hru ukončuje.

Tieto pravidlá riadia rozhodovanie v algoritme počítania stavu hry pri zaznamenaní gólu.

#### **3.6.4 Načítanie z naskenovaného záznamu**

Vypĺňanie záznamu o zápase môže byť pri veľkom počte udalostí v zápase namáhavou a zdĺhavou prácou. Sports Manager preto ponúka možnosť načítať dáta automaticky z naskenovaného zápisu o zápase.

Pre používateľa je pripravený prázdny zápis o zápase v súbore Excel. Na začiatku formulára pre pridanie zápasu si ho môže klikom na tlačidlo “Export gamesheet” stiahnuť vo formáte XLSX alebo PDF. Pre načítanie dát z naskenovaného dokumentu funguje iba tento formát zápisu o zápase. Údaje ako čas a dátum stretnutia, meno súťaže a sezóny, mená tímov a hráčov sa vyplnia automaticky vopred podľa daného zápasu.

Pre načítanie dát sa používa technika OCR<sup>10</sup>. Po stlačení tlačidla “Load gamesheet” sa zobrazí súborový prehliadač pre výber naskenovaného zápisu. Po

---

<sup>10</sup> OCR (Optical Character Recognition) – Optické rozpoznávanie znakov je technika, ktorou sa digitalizuje tlačенý alebo ručne písaný text.

výbere správneho skenu sa spustí skript GamesheetOCR.exe napísaný v jazyku Python. Skript bol začlenený do aplikácie ako aplikačný súbor namiesto obyčajného skriptu. Výhodou je, že používateľ nemusí mať nainštalovaný Python ani žiadne potrebné knižnice. Nevýhoda je veľkosť súboru, ktorá je omnoho väčšia. Používateľ ale musí mať nainštalovaný Tesseract OCR[8], ktorý jednotlivé znaky v dokumente rozpoznáva. Pre využívanie tejto techniky načítania dát, používateľ tiež musí zadať cestu k súboru tesseract.exe do textového súboru "C:\Users\{POUŽÍVATEĽ}\AppData\Roaming\SportsManager\tesseract\_path.txt", vytvorenom po prvom spustení aplikácie Sports Manager.

Časť skriptu GamesheetOCR.py na detekovanie kontúr je prevzatá z internetového článku autora Hucker Marius.[12]

Skript najskôr načíta a upraví naskenovaný dokument. Obrázok prekonvertuje na čierno-biely, zbaví sa šumu a zmení jeho veľkosť. Potom nájde hrany tabuľky (zápis o zápase sa skladá z tabuliek) a vyrovná ju. Hrany pre lepšie spracovanie textu a čo najväčšie eliminovanie šumu odstráni.

Podobným spôsobom boli vopred nájdené hrany tabuliek prázdneho zápisu o zápase. Pozície o hranách sa uložili a pevne zakódovali do skriptu GamesheetOCR.py. Podľa týchto údajov si skript zrekonštruje tabuľky na presnej pozícií. Nakoniec sa pomocou knižnice pytesseract prečítajú dáta zo zápisu. Pozícia odlišných dát v tabuľkách je vopred pevne určená. Skript vráti reťazec dát, v ktorom sú údaje zoskupené do skupín podľa typu, každý na samostatnom riadku, oddelené riadkom obsahujúcim reťazec "END". Rozpoznávanie znakov je obmedzené len na očakávané znaky pre väčšiu presnosť. V prípade zlyhania rozpoznania znaku sa daný údaj odošle ako reťazec "empty".

Aplikácia Sports Manager dáta prijme a spracuje. Spracovanými dátami potom vyplní formulár a používateľa upozorní na chybné alebo nejasné údaje, ktoré môže skontrolovať a upraviť alebo doplniť. Aplikácia zväčša načíta jeden dokument za minútu až minútu a pol. Rozpoznávanie znakov najlepšie funguje na strojom napísanom a vytlačennom texte. V takomto prípade, podľa otestovania, je rozpoznávanie úspešné približne na 80%. Pri ručne písanom texte závisí úspešnosť rozpoznávania od kvality, typu a jednoznačnosti daného písma. Pre načítanie dát z ručného písma by preto mal byť kladený dôraz na čo najpresnejšie zapisovanie. Pri desiatkach vyplnených údajoch, aplikácia väčšinou nerozpozna pár z nich, na čo následne upozorní, a používateľ tieto údaje ľahko upresní, či doplní. Automatické

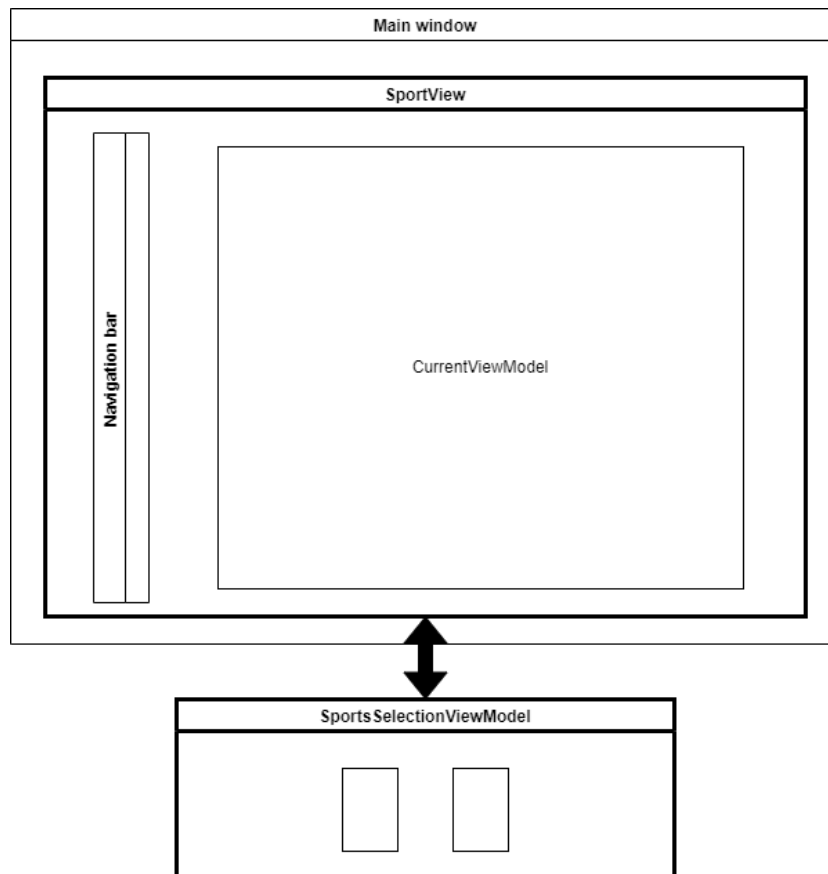
načítanie dát z naskenovaného dokumentu preto môže ušetriť veľa času, najmä pri zápasoch s množstvom údajov.

### 3.7 Navigácia medzi pohľadmi

Hlavné okno aplikácie po výbere športu obsahuje na ľavej strane zvislú navigačnú lištu, vo vrchnej časti sa nachádza názov práve vybraného športu, súťaže a sezóny. Zvyšok okna zaberá konkrétny pohľad, **SportView**.

Trieda **SportViewModel** má položku **CurrentViewModel**, ktorá predstavuje aktuálny pohľad. Každý pohľad implementuje rozhranie **INotifyPropertyChanged**. Pre uloženie aktuálneho pohľadu slúži trieda **NavigationStore**. Pohľad sa mení pomocou triedy **NavigateCommand<TViewModel>**, ktorá vytvorí nový pohľad a uloží ho do jedinej inštančii triedy **NavigationStore**. Počas toho sa prípadne nastaví aktuálna súťaž a sezóna. **NavigationStore** upozorní triedu hlavného okna **MainViewModel** na zmenu pohľadu a tá skrz metódu **OnCurrentViewModelChanged()** upozorní hlavné okno, že sa má prekresliť s novými údajmi.

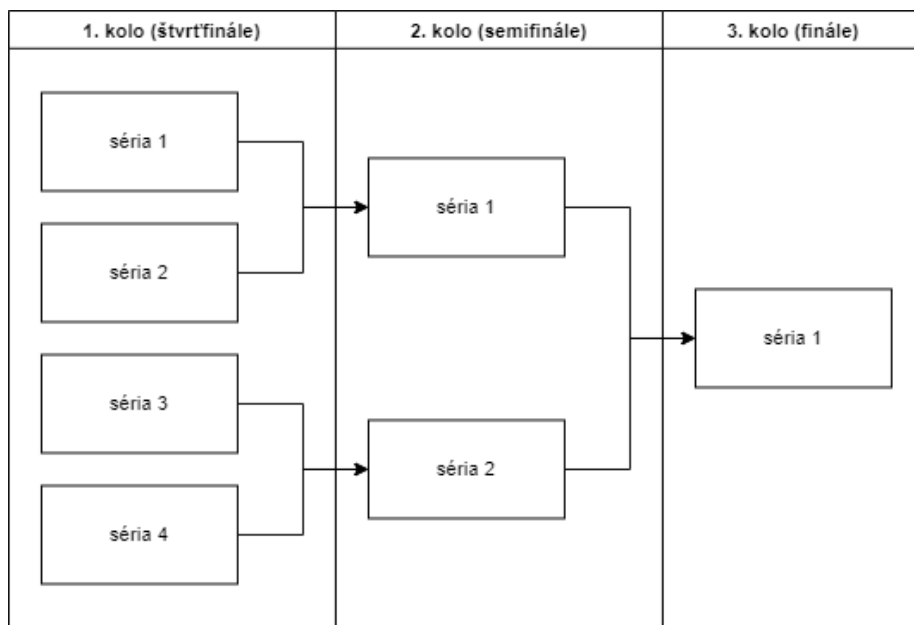
Pri spustení aplikácie sa v hlavnom okne zobrazuje pohľad výberu športu, ktorý zabezpečuje trieda **SportsSelectionViewModel**. Nasledujúci Obrázok 6: **Hierarchia hlavného okna** znázorňuje hierarchiu hlavného okna.



Obrázok 6: Hierarchia hlavného okna

### 3.8 Pavúk

Trieda **Bracket** implementuje štruktúru pavúka pre play-off a kvalifikáciu. Obsahuje premennú **ObservableCollection<List<Serie>> series**, čo je zoznam kôl so svojimi sériami. Každá položka typu **List<Serie>** reprezentuje jedno kolo, v poradí od prvého kola až po finále. Počet sérií v jednom kole musí byť mocninou dvojky. Série v kole sú zoradené od vrchu nadol.



Obrázok 7: Rozdelenie pavúka

Séria je reprezentovaná triedou **Serie**. Tá obsahuje zoznam svojich zápasov v chronologickom poradí. Udržiava si údaj o víťaznom tíme série, ktorý po vložení tímu prepočíta, podľa počtu víťazných zápasov.

Pri načítaní pavúka z databázy sa najskôr zostavia všetky kolá, série a ich zápasy. Potom sa rekurzívne celý pavúk prechádza. Pritom sa počítajú víťazi sérií, ktorí sa dosadzujú do ďalších kôl. Série, ktoré sú už nemenné, čiže tie, na ktorých sú závislé ďalšie série v nasledujúcich kolách, sa uzamknú.

Ak používateľ pridá, upraví alebo odstráni zápas či celú sériu, daná vetva pavúka sa oboma smermi prepočíta a aktualizuje.

V okne sa pavúk zobrazuje vnorený do prvku **ZoomBorder**, ktorý slúži na približovanie, oddialovanie a posun pavúka pre väčšiu prehľadnosť. Tento prvok WPF bol prevzatý z webovej stránky od používateľa Wiesław Šoltés.[13]

### 3.9 Rozpis zápasov

V sekcii rozpisu zápasov, do ktorej sa dá navigovať skrz tlačidlo “Schedule” v navigačnej lište sa zobrazujú všetky odohrané aj ešte neodohrané zápasy.

Sekcia obsahuje tri rôzne pohľady, pričom naraz sú vždy prístupné najviac dve z nich podľa formátu súťaže. Pohľad Groups obsahuje rozpis zápasov v základnej časti rozdelený do jednotlivých kôl. Pohľad Qualification obsahuje pohľad každého pavúka kvalifikácie. Nakoniec pohľad Play-off obsahuje pavúka play-off. Medzi

pohľadmi sa dá prepínať pomocou hornej lišty, v ktorej sú prvky WPF RadioButton, každý prepínajúci pohľad na formát s ním spojený.

V pohľade kvalifikácie sa pôvodne mali zobrazovať všetky jej pavúky súčasne pod sebou. To bolo ale náročné na vykresľovanie a pri veľkom počte pavúkov obsahujúcich veľa kôl aplikácia na pár sekúnd zamrzla. Taktiež to bolo neprehľadné, hľadanie konkrétneho pavúka pri väčšom množstve nebolo ideálne. Preto bola uprednostnená možnosť, kde sa zobrazuje iba jeden pavúk a medzi pavúkmi sa dá v poradí prepínať. Takto sa dosiahla aj väčšia prehľadnosť konkrétneho pavúka.

### 3.10 Poradie tímov v základnej časti

Sekcia “Standings” je dostupná z navigačnej lišty v prípade, že je vybraná konkrétna sezóna. Nachádzajú sa v nej tabuľky poradia tímov podľa skupín (ak sezóna obsahuje základnú časť obsahujúcu aspoň jedno kolo) a tlačidlo pre vyhlásenie víťaza.

Poradie sa určuje na základe získaných bodov zo zápasov a v prípade rovnakého počtu bodov podľa štatistík v tomto poradí: vyššie skóre (rozdiel strelených a inkasovaných gólov), vyšší počet strelených gólov, nižší počet trestných minút. Ak majú viaceré tímy identické aj tieto štatistiky, poradie sa určí náhodne.

Tabuľky poradia je možné zobrazit' po konkrétnom kole. To znamená, že sa zobrazia štatistiky a poradie, aké bolo po odohratí daného kola. Ak sa kolo zmení, poradie sa prepočíta znova. Tímy sa v skupinách triedia algoritmom triedenia zlievaním.

### 3.11 Aktuálny stav aplikácie

Stav aplikácie je určený vybraným športom, súťažou a sezónou. Pre uchovávanie aktuálneho stavu aplikácie slúži statická trieda **SportsData**, ktorá obsahuje premenné pre práve vybraný šport, súťaž a sezónu. Zmena stavu nastáva pri navigácii aplikáciou, kedy sa vykonáva akcia na zmenu pohľadu triedou **NavigateCommand**. Tá dostáva ako parameter novo vybraný šport, súťaž alebo sezónu a zavolá metódu **SportsData.Set(object parameter)**, v ktorej sa následne podľa typu parametra zavolá konkrétna funkcia **Set()**.

## 3.12 Pri spustení aplikácie

Pri spustení aplikácie sa zaistí, že všetky potrebné priečinky lokálne existujú. Jedná sa o priečinky na ukladanie obrázkov pre fotografie hráčov, logá tímov a podobne. Všetky pridané obrázky sú uložené v priečinkoch na ceste “C:\Users\{POUŽÍVATEL}\AppData\Roaming\SportsManager\Images“.

Ďalej sa v prípade potreby do priečinka \SportsManager na rovnakej ceste skopíruje spustiteľný súbor GamesheetOCR.exe slúžiaci na načítanie záznamu o zápase z naskenovaného dokumentu.

Potom sa skontroluje existencia všetkých databáz a ich tabuliek a ak niektoré chýbajú tak sa vytvoria.

Následne sa z databázy v samostatnom vlákne načíta zoznam všetkých krajín sveta pre zobrazovanie medzi informáciami o hráčoch a tímoch akými sú napríklad miesto narodenia a štátne občianstvo.

Nakoniec sa vytvorí nová inštancia triedy **NavigationStore** a nový začiatkový pohľad **SportsSelectionViewModel**, ktorý jej bude priradený. Vytvorí sa nové okno a ako jeho **DataContext** použitý na prepojenie sa s pohľadom sa nastaví novovytvorený pohľad **MainViewModel** s parametrom predtým vytvorenej inštancie **NavigationStore**.

## 3.13 Konzistencia údajov

### 3.13.1 Tvorba sezóny

Pri pridávaní novej sezóny je nutné presne určiť jej formát a následne priradiť všetky zúčastnené tímy buďto niektorej zo skupín alebo pavúku. Akonáhle je sezóna vytvorená, už nie je možné jej formát meniť alebo pridať či odstrániť nejaký tím. Tiež nie je možné prihlásenému tímu zmeniť priradenie do skupiny alebo pavúka. Ak by daná zmena nastala, mohlo by to úplne zmeniť a narušiť celú sezónu.

Toto obmedzenie vychádza z toho, že pri skutočnom usporiadaní súťaže, sú všetci jej účastníci známi ešte pred jej štartom a počas sezóny sa ich priradenia skupinám nemenia, rovnako ako sa nemení formát sezóny. Taktiež sa počas sezóny nepridáva žiaden nový tím.

V skutočnom prípade ale môže nastať situácia, že sa niektorý z tímov zo súťaže v jej priebehu odhlási alebo z nej bude vylúčený. Na takýto prípad slúži možnosť kontumácie zápasu, ktorá sa využíva pri všetkých typoch súťaží.

### **3.13.2 Uzamknutie editácie zápasov**

V niektorých prípadoch nie je editácia dát možná. Je to kvôli vzájomnej závislosti dát. Konkrétne, nie je možné upraviť ani odstrániť daný zápas základnej časti ak sa už začalo play-off alebo sezóna má vyhláseného víťaza. To isté platí pre zápas v pavúku a navyše musí byť ešte splnená podmienka, že nebol odohraný žiadny zápas v nasledujúcej sérii odvodenej z daného zápasu. Ak by to bolo možné, dáta by si navzájom odporovali. Poradie tímov v tabuľke by mohlo byť odlišné alebo by sa mohol zmeniť víťaz danej série.

Taktiež nie je možné po ukončení základnej časti odstraňovať a pridávať jednotlivé kolá.

V prípade nutnosti takúto zmenu uskutočniť, je potrebné najkôr buď zmazať celé play-off, nasledujúce zápasy v pavúku alebo zrušiť vyhlásenie víťaza sezóny. To ale nie je obmedzenie aplikácie, pretože takéto zmeny by pri skutočnom prípade nemohli nastať a práve naopak je tým podporená logika postupného priebehu sezóny.

### **3.13.3 Pridávanie a odstraňovanie hráčov**

Konkrétny hráč nemôže existovať sám o sebe a zakaždým musí byť priradený aspoň jednému tímu v aspoň jednej sezóne. Ak neexistuje zápis hráča v žiadnom z tímov, tak bude z databázy odstránený.

Preto sa noví hráči vytvárajú v pohľade konkrétneho tímu pri zápise do niektorej zo sezón.

Hráča možno zo zápisu (súpisky) odstrániť len vtedy, ak ešte neodohral žiaden zápas. V opačnom prípade je najskôr nutné odstrániť ho zo súpisky všetkých jeho odohraných zápasov. Je to tak kvôli tomu, že pri odstránení hráča sa môže stať, že v niektorom zo zápasov mal nejaký zápis, napríklad strelil gól, a to by mohlo zmeniť výsledok a aj víťaza zápasu a následne postupujúci tím do ďalšieho kola play-off a tak ďalej, čo je samozrejme nepriaznivé.

### 3.14 Práca s databázou

Načítanie, ukladanie, upravovanie a odstraňovanie dát z databázy je vykonávané knižnicou MySQL.Data od Oracle. Zakaždým sa nastaví a otvorí nové pripojenie pomocou triedy **MySqlConnection** a v triede **MySqlCommand** sa nastaví daný príkaz.

Uprednostnil som takýto základný spôsob práce s databázou a písaní príkazov MySQL syntaxou pred použitím rôznych frameworkov ako napríklad Entity Framework Core.[14] Pre mňa osobne je táto syntax prehľadnejšia a zreteľnejšia ako LINQ<sup>11</sup> príkazy. Častokrát bolo potrebné daný príkaz otestovať v rozhraní databázy a to v MySQL syntaxi, čiže preklad do LINQ príkazu by bola práca navyše.

### 3.15 Grafy

Grafy štatistík sú generované pomocou knižnice LiveCharts.WPF. V aplikácií sa objavujú koláčové grafy, čiarové grafy a bodové grafy.

V pohľade hráča sa zobrazujú tri čiarové grafy znázorňujúce jeho výkonnosť po jednotlivých zápasoch. Dáta pre grafy sa načítajú z databázy metódou **PlayerInMatchStatsLoader.LoadPlayerInMatchStats(...)** a sú uložené v zozname obsahujúcom prvky triedy **PlayerInMatchStats**, ktorá obsahuje dátum konkrétneho zápasu a štatistiku hráča v ňom. Taktiež sa uložia dátumy zápasov hráča vo forme reťazca a zotriedia. Tie sú následne použité ako popisy jednej z osí grafu. Druhá os znázorňuje číslom veľkosť konkrétnej štatistiky.

V pohľade poradia tímov v sezóne sa zobrazujú koláčové grafy znázorňujúce podiel rôznych štatistík tímov a ďalšie bodové grafy, v ktorých sa body zobrazujú ako logá daných tímov s ich názvom. Grafy sa aktualizujú podľa vybraného kola, pre ktoré sa majú štatistiky akumulovať.

### 3.16 Export

Všetky tabuľky štatistík sa dajú exportovať vo formáte XLSX alebo PDF. Zakaždým sa vygeneruje nový súbor excelu a ten sa prípadne exportuje do formátu PDF.

---

<sup>11</sup> LINQ – Language Integrated Query je dotazovací jazyk v prostredí .NET [15]

Grafy a vizualizácie pavúkov je možné exportovať do formátu PNG. O to sa stará statická metóda **Export.ExportControlToImage(FrameworkElement control, string name)**, v ktorej sa grafický prvok vykreslí do bitmapy a následne uloží do súboru typu PNG.

Pre export tabuliek poradia tímov je predpripravený Excel súbor so šablónou, podľa ktorej sa tabuľky vygenerujú. Vytvorí sa nový dočasný súbor skopírovaním šablóny, všetky dáta sa do neho zapíšu a nakoniec sa súbor uloží na používateľom dané miesto. Dočasný súbor sa potom zmaže. Export je opäť možný vo formáte XLSX alebo PDF.

Rovnako je predpripravená šablóna v Excel súbore aj pre export súhrnu zápasu. Totožným postupom ako pre tabuľky poradia tímov sa vygeneruje a uloží aj súhrn zápasu.

## 4. Funkcionalita

### 4.1 Navigácia

Pri štarte aplikácie sa zobrazí hlavné okno s možnosťou výberu športu. Po výbere športu je v ľavej časti okna dostupná navigácia. Je možné navigovať sa medzi zoznamami športov, súťaží, sezón, zápasov, tímov, hráčov a brankárov. V prípade, že je práve vybraná nejaká sezóna, sa v navigácii zobrazia aj položky “Schedule“ (rozpis) a “Standings“ (poradie tímov).

V zozname súťaží alebo sezón je možné vybrať si konkrétnu z nich, čo následne obmedzí zoznamy hráčov, tímov a zápasov s ich štatistikami len na danú súťaž alebo sezónu.

### 4.2 Súťaže

Aplikácia podporuje sledovanie štatistík a spravovanie sezón viacerých súťaží súčasne. Každá súťaž okrem jej názvu môže obsahovať aj logo súťaže a informácie o nej vo forme bloku textu.

### 4.3 Sezóny

Každá súťaž sa môže skladať z ľubovoľného počtu sezón, pričom každá z nich môže byť iného formátu a obsahovať rôzne tímy a hráčov. Rovnako ako súťaž, aj sezóna môže byť priradené logo a informácie o nej.

#### 4.3.1 Formát sezóny

K dispozícii sú tri základné formáty, ktoré je prípadne možné spojiť a prispôbiť k potrebám sezóny a súťaže. Tieto formáty sú skupinová časť, kvalifikácia a play-off. Sezóna sa môže skladať z jednej z týchto častí alebo z kombinácie skupinovej časti a play-off, či kvalifikácie a play-off.

Skupinová časť sa skladá z jednej alebo viacerých skupín. Maximálny počet skupín je 16. Počet tímov v skupine nie je obmedzený. Skupiny určujú vlastné

poradie tímov. Zápasy ale nie sú nijak obmedzené na tímy v rámci jednej skupiny. Tímy z rôznych skupín môžu hrať proti sebe. Sezóna sa skladá z ľubovoľného počtu kôl, ktoré obsahujú zápasy. Poradie v skupine sa určuje na základe získaných bodov tímu. Používateľ si pri vytvorení novej sezóny sám určí, koľko bodov tím získa za výhru v riadnom hracom čase, výhru po predĺžení, remízu, prehru po predĺžení a prehru v riadnom hracom čase.

Kvalifikácia pozostáva z jedného alebo viacerých pavúkov. Jeden tím môže hrať len v jednom z pavúkov. Maximálny počet pavúkov je 16, maximálny počet súťažiacich v kvalifikácii je 256 a maximálny počet kôl jedného pavúka je 6. Každá séria sa hrá iba na jeden zápas. Viacero pavúkov v kvalifikácii umožňuje vytvorenie kvalifikácie pre väčší turnaj, na ktorý sa postupne kvalifikujú viaceré tímy. Každý pavúk má rovnakú veľkosť a v prípade potreby rôznych veľkostí pavúkov sa nastaví veľkosť najväčšieho z nich, a v menších pavúkoch sa súťažiaci nasadia do vyššieho kola. V priebehu sezóny tímy automaticky postupujú do ďalšieho kola po vyhratom zápase.

Play-off tvorí jeden pavúk s maximálne 8 kolami. Série sa môžu hrať na jeden až päť víťazných zápasov. Po výhre v sérii tím automaticky postúpi do ďalšieho kola. Play-off môže predchádzať skupinovú časť alebo kvalifikácia. Po štarte play-off sa predchádzajúca časť zablokuje pre ďalšiu úpravu. Používateľ má možnosť automaticky nasadiť tímy do play-off podľa výsledkov z predchádzajúcej časti sezóny.

Sezónu je možné kedykoľvek ukončiť vyhlásením víťaza. Ten ale nie je obmedzený poradím v skupinách alebo víťazstvom v pavúku pre väčšiu flexibilitu vlastných pravidiel sezóny.

Sports Manager teda umožňuje tvorbu rozličných formátov sezón. Aplikáciu je možné použiť pre turnaje, ligy, samostatné kvalifikácie alebo aj samostatné zápasy, napríklad exhibície. Pre väčší prehľad štatistik a lepšiu správu sa kvalifikácia a hlavná časť súťaže môže aj rozdeliť do dvoch sezón, kde jedna bude slúžiť na kvalifikáciu a druhá na nasledujúcu časť súťaže.

## 4.4 Zápasy

Zápasy sa môžu ukladať buď už odohraté alebo ako naplánované. Pre naplánovanie zápasu sa vyplnia údaje o dátume a čase plánovaného stretnutia a dva

súperiace tímy. Naplánovaný zápas sa neskôr môže upraviť doplnením zápisu o zápase a uložiť ako odohratý. Pre prepínanie medzi tým, či sa zápas už odohral alebo nie, slúži vo formulári pridávania zápasu tlačidlo “Played”.

Zápas je možné aj kontumovať v prospech jedného z tímov. Deje sa tak v prípade, že jedno z mužstiev sa na zápas nedostavilo, alebo bolo vylúčené zo súťaže, alebo ako trest pri podvádzaní a podobne. Ak sa zápas kontumoval až po jeho odohraní alebo počas neho, všetky dovtedy zaznamenané štatistiky oboch tímov sa zarátajú. Skóre zápasu bude vždy nastavené ako nula pre tím, kvôli ktorému sa kontumuje, a ako jedna alebo vyššie dosiahnuté skóre pre súpera.

Zo zápasu je možné zaznamenať údaje o účasti hráčov, góly s asistenciami, trestné strielania, vylúčenia, time-outy, samostatné nájazdy, striedania brankárov a samotný výsledok stretnutia.

Pre všetky zápasy je systém hry nastavený ako piati hráči proti piatim hráčom okrem brankárov a toto nastavenie je nemenné.

#### **4.4.1 Periódy**

Každý zápas v sezóne sa môže hrať na odlišný počet períód, maximálne však na dvanásť plus predĺženie. Každý zápas môže mať rovnako odlišnú dĺžku trvania períódy. Táto dĺžka sa udáva v minútach a môže byť maximálne rovná 600 minútam.

Predĺženie je poslednou períódou, ktorá sa väčšinou pridáva pri nerozhodnom stave, no nemusí tomu byť vždy tak a preto pridanie predĺženia nie je obmedzené stavom stretnutia. Niektoré súťaže totiž môžu rozhodovať vyrovnanú sériu prdlžením v jej poslednom zápase, v prípade, že oba tímy v celej sérii strelili rovnaký počet gólov. Predĺženie obvykle končí gólom. Maximálne však môže trvať 600 minút.

Ďalej sa na koniec zápasu môže pridať aj rozstrel pozostávajúci zo sérii samostatných nájazdov. Samostatný nájazd je pokus hráča streliť gól súperovému brankárovi v situácii jeden na jedného. Jedna séria v rozstrele je pojem pre dva po sebe nasledujúce samostatné nájazdy oboch tímov. Tímy sa v nájazdoch striedajú. Obvykle sa hrá napríklad na tri alebo päť sérii, kde vyhráva tím s vyšším počtom gólov a v prípade remízy sa pokračuje vždy po jednej ďalšej sérii až kým jeden z tímov gól strelí a ten druhý nie (to sa nazýva rýchla smrť). Rozstrel ale taktiež môže skončiť už skôr ak jeden z tímov stratí teoretickú šancu na víťazstvo. To môže nastať napríklad v prípade kedy sa hrá na tri série a jeden z tímov úspešne premení

prvé dva samostatné nájazdy pričom druhý tím nepremení ani jeden z prvých dvoch. Potom sa tretia séria už neodohrá, keďže by sa rozdiel gólov tímov znížil najviac o jeden, čo konečný stav už nezmení. Pri rozstrele preto používateľ udá ľubovoľný počet sérii podľa pravidiel súťaže, najviac však 99.

Každý zápas musí obsahovať aspoň jednu periódu a prípadne môže pozostávať aj z predĺženia či rozstrelu, alebo oboch zároveň.

## 4.5 Tímy

Tímu sa dá okrem názvu priradiť aj jeho logo a informácie o ňom. Tím pozostáva zo zápisov do sezón, ktorých sa účastní, a zo súpisiek hráčov pre jednotlivé sezóny, pričom každá súpisica môže byť odlišná, nazávisle od iných. Vytvorenie nového tímu sa uskutočňuje pri vytváraní novej sezóny.

V pohľade tímu sa nachádzajú základné informácie o ňom a všetky jeho zápisy v jednotlivých súťažiach a sezónach s jeho súpiskami pre ne.

## 4.6 Hráči

Ako už bolo spomenuté, hráč nemôže existovať bez tímu a sezóny. Vždy musí byť priradený aspoň jednému tímu a zapísaný aspoň v jednej sezóne. Nový hráč sa vytvára v pohľade tímu pri zápise do súpisky. Do súpisky sa môže pripísať buď novovytvorený hráč alebo ľubovoľný, v danom športe už existujúci hráč.

U hráča môžu byť zaznamenané základné informácie o ňom ako sú napríklad výška, váha, miesto a dátum narodenia, občianstvo, informácia o tom, či hrá ako pravák alebo ľavák, pohlavie, a údaj o tom, či je ešte aktívny hráč. Taktiež je možné uložiť jeho fotku. Všetky tieto informácie sa zobrazujú v pohľade hráča, kde sa ďalej nachádzajú aj grafy o jeho výkonnosti v zápasoch a štatistiky podľa súťaží a sezón, ako aj zoznam všetkých jeho odohraných zápasov. Tieto štatistiky sa delia na dve časti, pre zápasy v pozícii hráča v poli a pre zápasy, v ktorých chytal ako brankár.

## 4.7 Prehľady štatistík

V zoznamoch sezón, zápasov, tímov, hráčov a brankárov je možné si prezerat' aj všetky ich štatistiky. Tieto zoznamy sa dá exportovať do formátu PDF alebo

XLSX. Zoznamy pozostávajú z fotky, názvu alebo mena, základných informácií a štatistík. Tieto skupiny je možné po jednej skryť alebo zobraziť pre lepšiu prehľadnosť. Zoznam sa dá triediť podľa každého alebo viacerých stĺpcov.

Zoznam zobrazuje údaje vždy v rámci vybranej súťaže a sezóny, alebo pre celú súťaž či sezónu ak nie je vybraná žiadna.

## 4.8 Rozpis zápasov

Nachádzajú sa tu odohrané aj naplánované zápasy. Nový zápas sa pridáva práve cez túto sekciu, dosiahnuteľnú z navigačnej lišty tlačidlom “Schedule”. Rozpis je rozdelený podľa časti sezóny.

V rozpise zápasov pre základnú časť sa zápasy zobrazujú zoskupené podľa kôl. Vidieť tu skrátené informácie o zápase a tlačidlom “Detail” pri každom zápase sa dá navigovať na podrobný detail zápasu. Každý zápas musí byť v nejakom kole. Dajú sa tu pridávať nové kola a taktiež odstraňovať spolu so všetkými ich zápasmi.

V časti pre play-off je rozpis zobrazený v štruktúre pavúka, ktorým sa dá pohybovať a je približovateľný. Skrz kliknutie na zápas v sérii pavúka sa dá navigovať na buď pridanie nového zápasu série alebo na detail zápasu. Tímy v pavúku automaticky postupujú do ďalšieho kola v prípade, že dosiahli potrebný počet víťazstiev v sérii. Ak play-off časť sezóny ešte nezačala, je ju tu možné spustiť aj s automatickým nasadením hráčov podľa poradia v predchádzajúcej časti sezóny alebo náhodne. Používateľ si súťažiacich môže nasadiť aj sám a to hocikedy počas sezóny, v prípade, že je to možné (nie sú blokované inými zápasmi). Celá časť play-off sa dá hocikedy odstrániť spolu so všetkými jej zápasmi ak ešte nebol vyhlásený víťaz sezóny.

Sekcia kvalifikácie je podobná tej pre play-off s tým, že sa tu dá prepínať medzi viacerými pavúkmi.

## 4.9 Detail zápasu

V pohľade zápasu je vypísaný detailný priebeh stretnutia po periódach. Vypisujú sa tu hlavné udalosti zápasu ako góly a vylúčenia. Taktiež tu vidieť súpiscky zúčastnených hráčov oboch tímov. Cez tento pohľad je možné prejsť na úpravu zápasu alebo zápas odstrániť. Detail zápasu možno uložiť vo formáte PDF.

## 4.10 Tabuľky poradia tímov

V sekcii “Standings” dostupnej z navigačnej lišty sa zobrazujú tabuľky poradia tímov podľa skupín v základnej časti sezóny. Tabuľky možno zobrazovať pre uvedené kolo, pričom sa vždy zobrazia štatistiky a body dosiahnuté tímami od začiatku sezóny až po dané kolo. Aktuálne poradie tímov je možné exportovať vo formáte PDF. Správca sezóny (používateľ) má tak prehľad nad doterajšou úspešnosťou tímov a môže ju vo forme tabuliek prezentovať napríklad na internetovej stránke súťaže.

V tejto sekcii sa taktiež vyhlasuje víťaz sezóny. Vybrať sa dá hociktorý zúčastnený tím. Vyhlásením víťaza sa všetka editácia zápasov sezóny zablokuje. Víťaza je možné kedykoľvek odstrániť.

## 4.11 Štatistiky

Sports Manager zaznamenáva čo najviac údajov pre možné počítanie čo najväčšieho množstva rôznych štatistík. Aplikácia zobrazuje všetky základné štatistiky, ktoré sú zaujímavé pre amatérske súťaže, ale aj niekoľko nie až tak častých štatistík ako napríklad strelené góly bez predošlej asistencie. V tejto podkapitole sa najskôr prejdú a vysvetlia všetky štatistiky zobrazované v aplikácii, potom sa rozoberie potenciál počítania ďalších štatistík a nakoniec budú spomenuté štatistiky, ktoré aplikácia nezaznamenáva.

### 4.11.1 Štatistiky v aplikácii

Rôzne druhy štatistík sa v aplikácii zobrazujú na viacerých miestach ako napríklad v pohľade hráča či tímu, alebo v jednotlivých zoznamoch sezón, zápasov, hráčov, brankárov a tímov. Jedná sa o nasledujúce štatistiky:

- Góly – počet strelených gólov
- Asistencie – počet nahrávok na gól
- Trestné minúty – počet obdržaných trestných minút
- Odohrané zápasy – počet odohraných stretnutí

- Čisté kontá – počet zápasov bez inkasovaného gólu, pre brankára sa započítavajú iba zápasy, ktoré celé odohral
- Inkasované góly – počet inkasovaných gólov
- Inkasované góly za zápas
- Čas v hre – celkový čas, ktorý hráč strávil v hre v minútach
- Čas v hre za zápas
- Čas v hre za inkasovaný gól – priemerný čas v minútach medzi dvoma inkasovanými gólmi brankára (alebo, brankár v priemere inkasuje gól každú udanú minútu)
- Trestné strielania proti – počet trestných strielaní proti brankárovi
- Inkasované góly v trestných strielaniach
- Úspešnosť v trestných strielaniach – vyjadrená v percentách
- Samostatné nájazdy proti
- Inkasované góly v samostatných nájazdoch
- Úspešnosť v samostatných nájazdoch
- Výhry – počet výhier v riadnom hracom čase
- Výhry po predĺžení
- Remízy
- Prehry po predĺžení
- Prehry
- Gólový rozdiel – rozdiel strelených a inkasovaných gólov
- Góly za zápas
- Asistencie za zápas
- Trestné minúty za zápas
- Body – súčet gólov a asistencií
- Body zo zápasov – body získané podľa výsledku zápasu
- Góly v presilovej hre
- Góly v oslabení
- Góly pri vyrovnanom počte hráčov
- Góly do prázdnej brány – počas odvolania súperovho brankára
- Vlastné góly
- Góly v ponechanej výhode – počas avizovaného vylúčenia

- Rozhodujúce (víťazné) góly – rozhodujúci gól je posledný gól víťazného tímu v zápase, ktorý by stačil na víťazstvo aj bez skórovania akéhokoľvek ďalšieho gólu
- Góly bez asistencie – počet gólov bez predošlej asistencie, iba góly z hry
- Trestné strielania – počet pokusov trestných strielaní
- Góly v trestnom strielaní
- Úspešnosť trestného strielania – podiel počtu gólov a pokusov

#### 4.11.2 Štatistiky podporované modelom aplikácie

Z údajov z databázy sa dajú odvodiť mnohé štatistiky. Nie všetky sa však v aplikácii zobrazujú. Niektoré totiž nemusia byť dostatočne zaujímavé alebo prínosné pre bežné amatérske súťaže. Nasledujúce štatistiky, ale aj mnohé iné, je možné z uložených údajov v databáze odvodiť:

- Štatistiky hráča:
  - Počet výhier, remíz, prehier, predĺžení, úspešnosť výhier
  - Počet prvých gólov, vyrovnávajúcich gólov, vedúcich gólov
  - Počet gólov podľa skóre (tesné góly, góly za nepriaznivého a priaznivého stavu)
  - Počet gólov podľa periódy, góly v predĺžení
  - Počet asistencií podľa skóre, periódy, počtu hráčov, asistencie na víťazný gól, na gól do prázdnej brány
  - Počet bodov podľa skóre, periódy, počtu hráčov, typu
  - Počet vylúčení, vylúčenia podľa typu, podľa dôvodu vylúčenia
  - Počet potrestaných vylúčení
  - Čas na trestnej lavici
  - Samostatné nájazdy, počet gólov v nich, úspešnosť, víťazné nájazdy
- Štatistiky brankára:
  - Totožné ako u hráča, v prípade, že má štatistika zmysel
  - Začaté zápasy, celé zápasy, počet stiahnutí zo zápasu
  - Inkasované vlastné góly
  - Počet gólov tímu počas hry v bráne

- Inkasované góly podľa počtu hráčov, periódy, skóre
- Rozhodujúce zákroky v samostatných nájazdoch
- Štatistiky tímu:
  - Totožné ako u hráča a brankára, v prípade, že má štatistika zmysel
  - Výsledky jednotlivých periód, úspešnosť podľa periód
  - Počet sezón, počet účastí v play-off
  - Úspešnosť v presilových hrách a oslabeniach
  - Góly, asistencie, vylúčenia na hráča, inkasované góly na brankára
  - Počet time-outov
  - Čas v početnej prevahe, oslabení
- Ostatné štatistiky:
  - Priemerný hráč tímu, sezóny, súťaže, priemerný hráč za zápas
  - Rozdiel hráča od priemerného hráča
  - Počet gólov hráča podľa brankára, tímu
  - Asistencie od hráča a hráčovi
  - Pri štatistikách kde to dáva zmysel sa môžu deliť podľa počtu hráčov, periódy, skóre, časti sezóny, výsledku zápasu, typu (napríklad počet inkasovaných gólov vo vyhraných zápasoch)
  - Štatistika za zápas, minútu, periódu, sezónu (napríklad počet gólov za periódu)
  - Vyššie spomenuté štatistiky pre zápas, sezónu, súťaž

#### 4.11.3 Nepodporované štatistiky

Aplikácia Sports Manager nazaznamenáva údaje o jednotlivých strelách a ich výsledku, blokovaných strelách, plusových a mínusových bodoch, striedaniach hráčov, a vhadzovaniach. Je to z dôvodu niekoľkonásobne väčšieho vstupu, v prípade ich zaznamenávania. Tieto štatistiky zaznamenávajú len tie najväčšie národné a medzinárodné ligy a turnaje sveta, ktoré na to zamestnávajú desiatky ľudí. Preto sa zaznamenávanie týchto štatistík v amatérskych ligách nepredpokladá. V prípade, že by tieto štatistiky bolo predsa len potrebné zaznamenávať, nie je potrebný veľký zásah do kódu. Stačí vytvoriť nové tabuľky v databázy pre konkrétnu

štatistiku a do formulára zápisu zápasu pridať podformulár pre danú štatistiku. Následné zobrazovanie medzi ostatnými štatistikami je tiež priamočiaré naprogramovať (v zásade ide len o vloženie malej časti kódu reprezentujúcu novú štatistiku medzi ostatné, oddelené).

## 5. Záver

Všetky hlavné požiadavky boli v aplikácii implementované. Bohužiaľ boli v čase písania práce všetky amatérske športové súťaže buď zrušené alebo značne obmedzené. Osobne som ale v minulosti spoluorganizoval menšiu florbalovú ligu a na základe jej dát som aplikáciu nechal otestovať ostatnými spoluorganizátormi. Podľa ich odozvy bola aplikácia dostatočne jednoduchá a intuitívna na ovládanie. V porovnaní s ručným zapisovaním a spravovaním štatistik aplikácia Sports Manager ušetrí značné množstvo času, eliminuje výpočtové chyby a sprehládňuje celkové riadenie súťaže. Oproti iným komerčným aplikáciám sa Sports Manager zameriava výlučne na správu súťaže narozdiel od správy konkrétneho športového klubu.

### 5.1 Možnosti budúceho rozšírenia

Aplikácia ponúka mnohé možnosti na správu rôznych typov súťaží. Stále je tu ale priestor pre vylepšenie a pre pridanie novej funkcionality. Ponúka sa napríklad pridanie podpory pre ďalšie športy. Taktiež by bolo užitočné pridať zápas o tretie miesto v play-off. Prípadne aj ďalšie zápasy (o piate, siedme miesto). Tieto zápasy sa hrajú na majstrovstvách sveta vo florbale.

Pridanie možnosti nastavenia triedenia poradia tímov v skupinách podľa vlastných pravidiel by zase rozšírilo podporované formáty súťaží. V prípade rovnakého počtu bodov môže rozhodovať zisk bodov iba vo vzájomných zápasoch tímov alebo počet gólov v nich.

Zmena systému hry by taktiež rozšírila podporované formáty súťaží. Aktuálne je podporovaný iba systém piati proti piatim. Menšie súťaže sa ale môžu hrať aj systémom traja na troch, bez brankárov a podobne.

Sports Manager sa tiež môže rozšíriť o rôzne ďalšie pohľady štatistik. Porovnanie hráča proti hráčovi. Tabuľky gólov hráča podľa brankára a asistenta. Modifikovateľné grafy. Existuje mnoho možností, ako štatistiky zobrazovať. Pridanie nepodporovaných štatistik, ktoré boli spomenuté v predošlej kapitole ale aj ďalších je pomerne priamočiare a aplikáciu to môže iba zlepšiť.

Zaujímavým rozšírením by bola podpora živého zapisovania zápasu. Tu by bolo za potreby implementovať uloženie medzivýsledku rozohraného zápasu, kvôli prevencii straty dát.

## Zoznam použitej literatúry

- [1] Pravidlá florbalu:  
<https://floorball.sport/rules-and-regulations/rules-of-the-game/>
- [2] Pravidlá ľadového hokeja:  
<https://www.iihf.com/en/statichub/4719/rules-and-regulations>
- [3] PDF Wikipédia:  
[https://sk.wikipedia.org/wiki/Portable\\_Document\\_Format](https://sk.wikipedia.org/wiki/Portable_Document_Format)
- [4] PNG Wikipédia:  
[https://sk.wikipedia.org/wiki/Portable\\_Network\\_Graphics](https://sk.wikipedia.org/wiki/Portable_Network_Graphics)
- [5] MVC, MVP and MVVM Design Pattern od autora Ankit Sinhal:  
<https://medium.com/@ankit.sinhal/mvc-mvp-and-mvvm-design-pattern-6e169567bbad>  
[4.5.2022]
- [6] Model-view-viewmodel Wikipédia:  
<https://en.wikipedia.org/wiki/Model-view-viewmodel>
- [7] XAMPP:  
<https://www.apachefriends.org/index.html>
- [8] Tesseract-OCR Engine:  
<https://github.com/tesseract-ocr/tesseract>
- [9] LiveCharts.WPF:  
<https://lvcharts.net/>
- [10] Článok o MVVM zo stránky dotnetportal.cz:  
<https://www.dotnetportal.cz/clanek/4994/MVVM-Model-View-ViewModel>  
[28.4.2022]
- [11] Data binding vo WPF aplikáciach:  
<https://docs.microsoft.com/en-us/dotnet/desktop/wpf/data/?view=netdesktop-6.0>  
[27.4.2022]
- [12] Detekcia tabuliek od autora Hucker Marius:  
<https://towardsdatascience.com/a-table-detection-cell-recognition-and-text-extraction-algorithm-to-convert-tables-to-excel-files-902edcf289ec>  
[15.2.2022]

[13] WPF prvok ZoomBorder.cs od autora Wiesław Šoltés:

<https://stackoverflow.com/a/6782715/16441573>

[5.8.2021]

[14] Entity Framework Core:

<https://docs.microsoft.com/en-us/ef/core/>

[15] LINQ:

<https://docs.microsoft.com/en-us/dotnet/csharp/programming-guide/concepts/linq/>

## Zoznam obrázkov

Obrázok 1: Porovnanie návrhových vzorov MVC, MVP a MVVM .....	7
Obrázok 2: Diagram architektúry Model-view-viewmodel .....	14
Obrázok 3: Znáznornenie použitia tried implementujúcich rozhranie IStats.....	17
Obrázok 4: Hierarchia pohľadov modelov.....	19
Obrázok 5: Prepínanie viditeľnosti stĺpca v Datagrid.....	22
Obrázok 6: Hierarchia hlavného okna.....	30
Obrázok 7: Rozdelenie pavúka .....	31

# A. Prílohy

## A.1 Elektronická príloha

Elektronická príloha obsahuje:

- Zdrojové kódy aplikácie Sports Manager v priečinku `source_code`.
- Inštalačný súbor `SportsManagerInstaller.msi` aplikácie Sports Manager.
- Vygenerovanú programátorskú dokumentáciu v HTML a LaTeX forme.
- Testovacie dáta na ukážkovú funkcionálnosť aplikácie v priečinku `testing_data`.
- Tento text bakalárskej práce.

Aplikácia bola zostavená v prostredí Visual Studio 2019 a je cielená pre framework .NET 5.0. Aplikácia má závislosť na knižnici Microsoft Excel 16.0 Object Library. Taktiež používa NuGet balíčky LiveCharts.WPF a MySql.Data. Pri behu aplikácie je potrebné mať spustenú MySQL databázu. V prípade, že MySQL databáza nie je na počítači nainštalovaná, je potrebné tak urobiť. Jedna z možností je nainštalovať si balíček XAMPP zo stránky <https://www.apachefriends.org/index.html>, ktorý umožňuje MySQL databázu jednoducho spúšťať cez grafické rozhranie. Taktiež je potrebné mať nainštalovaný Tesseract OCR[8], ktorý slúži na jednotlivé rozpoznávanie znakov v dokumente zápisu o zápase. Pre využívanie tejto techniky načítania dát, používateľ tiež musí zadať cestu k súboru `tesseract.exe` do textového súboru `“C:\Users\{POUŽÍVATEĽ}\AppData\Roaming\SportsManager\tesseract_path.txt”`, vytvorenom po prvom spustení aplikácie Sports Manager.

Ukážkové testovacie dáta obsahujú sezónu so šiestimi tímami rozdelenými do štyroch skupín. V priečinku `testing_data` sa nachádza súbor `floorball.txt` obsahujúci tabuľky databázy a priečinok `Images` obsahujúci logá daných tímov a súťaží. Taktiež sa v ňom nachádza súbor `README.txt`, v ktorom je popísaný návod, ako testovacie dáta vložiť do aplikácie.



### **A.3 Užívateľská dokumentácia**

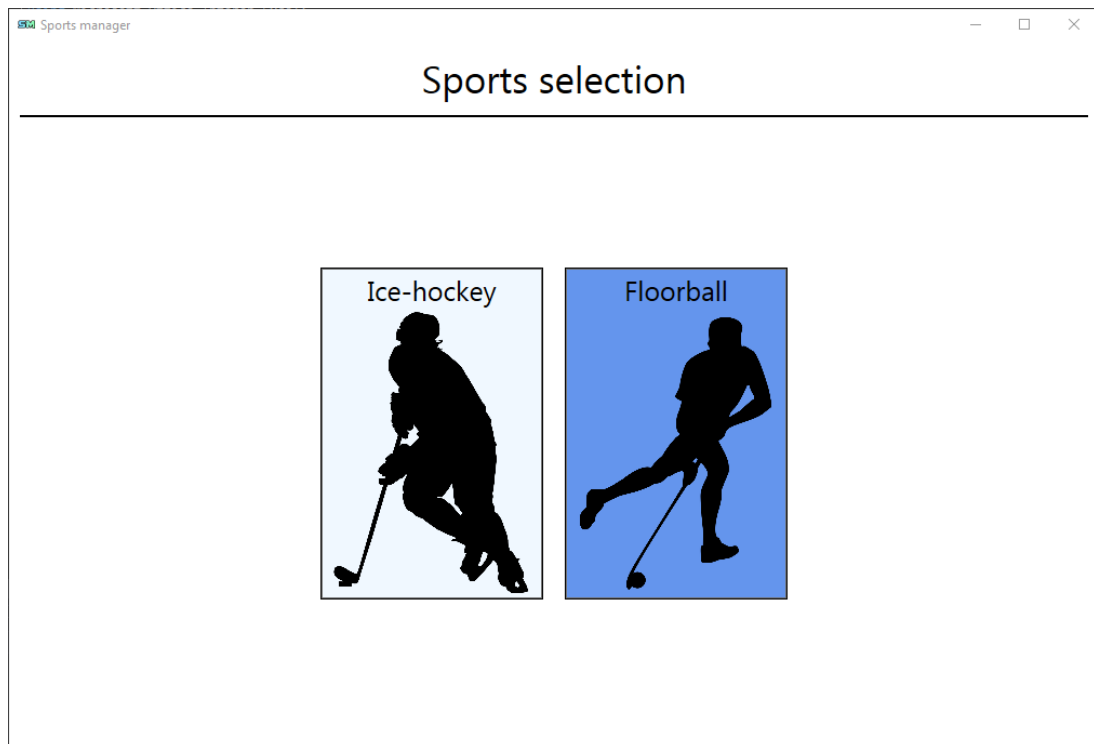
Aplikácia Sports Manager slúži na správu športových súťaží a na zaznamenávanie ich štatistík. Sports Manager umožňuje spravovať viacero súťaží rôzneho typu, či už ligy alebo turnaje. V aplikácii je možné zaznamenávať informácie o tímoch a ich jednotlivých hráčoch. Pomocou Sports Manager je taktiež možné spravovať rozpis zápasov a zaznamenávať priebeh jednotlivých zápasov. Z týchto dát potom aplikácia automaticky počíta rôzne štatistiky, ktoré prehľadne zobrazuje. Ďalšími možnosťami sú exportácia údajov do formátov PDF a XLSX, alebo automatické načítanie zápisu o zápase z naskenovaného dokumentu.

#### **Inštalácia aplikácie**

Aplikáciu je potrebné nainštalovať pomocou inštalačného súboru SportsManagerInstaller.msi, ktorý vytvorí skratku na pracovnej ploche pre spúšťanie aplikácie. Pri behu aplikácie je potrebné mať spustenú MySQL databázu. V prípade, že MySQL databáza nie je na počítači nainštalovaná, je potrebné tak urobiť. Jedna z možností je nainštalovať si balíček XAMPP zo stránky <https://www.apachefriends.org/index.html>, ktorý umožňuje MySQL databázu jednoducho spúšťať cez grafické rozhranie. Taktiež je potrebné mať nainštalovaný Tesseract OCR[8], ktorý rozpoznáva jednotlivé znaky v dokumente zápisu o zápase. Pre využívanie tejto techniky načítania dát, používateľ tiež musí zadať cestu k súboru tesseract.exe do textového súboru "C:\Users\{POUŽÍVATEL}\AppData\Roaming\SportsManager\tesseract\_path.txt", vytvorenom po prvom spustení aplikácie Sports Manager.

#### **Hlavné okno aplikácie**

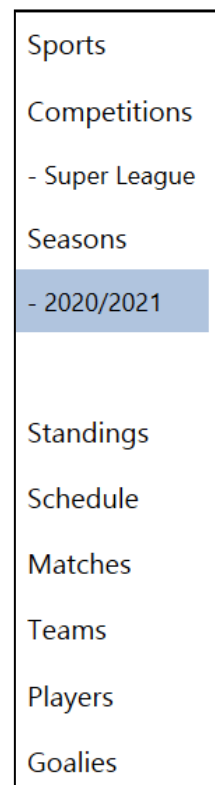
Po spustení aplikácie sa zobrazí okno s ponukou výberu športu. Na výber je florbal a ľadový hokej.



Obrázok 1: Výber športu

Po výbere športu sa zobrazí náhľad pre konkrétny šport. V ľavej časti okna sa nachádza navigačná lišta. Pomocou nej sa dá navigovať medzi pohľadmi:

- Sports – späť na pohľad výberu športu
- Competitions – pohľad výberu súťaže
- (názov vybranej súťaže) – pohľad detailu súťaže
- Seasons – pohľad výberu sezóny
- (názov vybranej sezóny) – pohľad vybranej sezóny
- Standings – pohľad aktuálneho poradia tímov
- Schedule – pohľad rozpisu zápasov
- Matches – pohľad zoznamu zápasov
- Teams – pohľad zoznamu tímov
- Players – pohľad zoznamu hráčov
- Goalies – pohľad zoznamu brankárov



Obrázok 2:  
Navigačná lišta

Vo vrchnej časti okna sú zobrazené názvy aktuálne vybraného športu, súťaže a sezóny.



Obrázok 3: Hlavné okno / pohľad sezóny

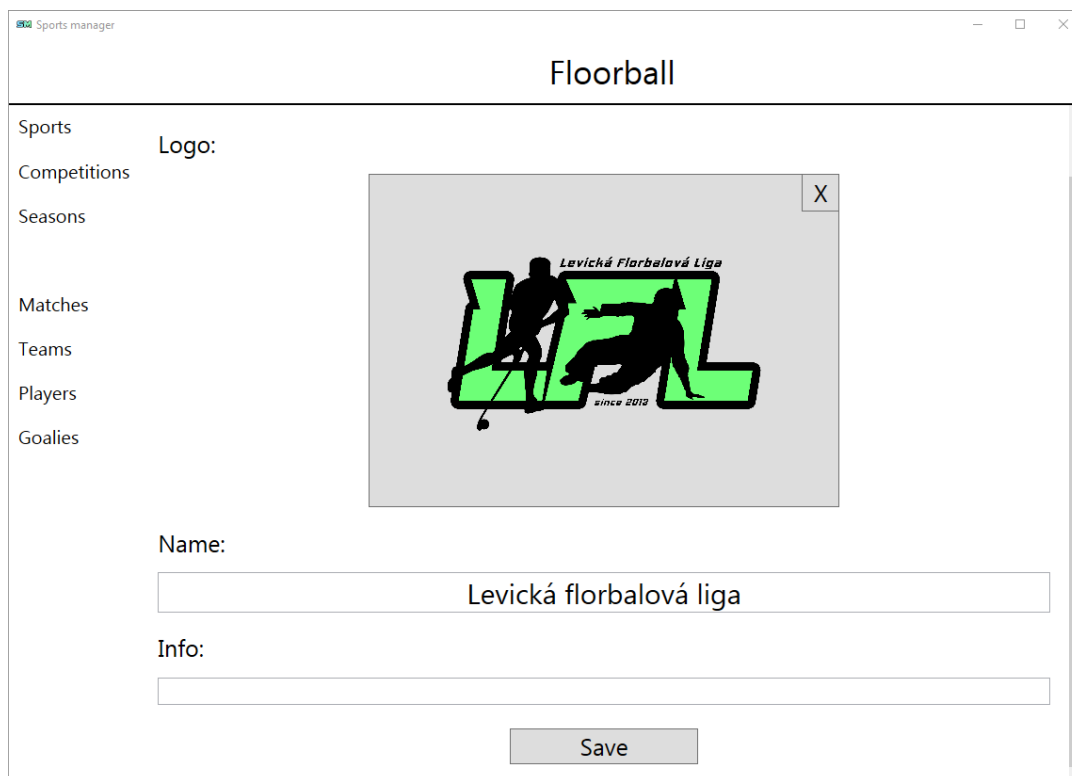
Vo zvyšku okna sa zobrazuje aktuálny náhľad.

## Vytvorenie novej súťaže

Pre vytvorenie novej súťaže kliknite na tlačidlo “ADD NEW” v pohľade výberu súťaže. Zobrazí sa formulár, v ktorom je potrebné vyplniť meno súťaže. Možnosťou je aj pridanie loga a ďalších informácií. Súťaž sa uloží po kliknutí na tlačidlo “Save”. Po úspešnom vytvorení sa súťaž pridá na zoznam medzi ostatné súťaže, odkiaľ bude dostupná.



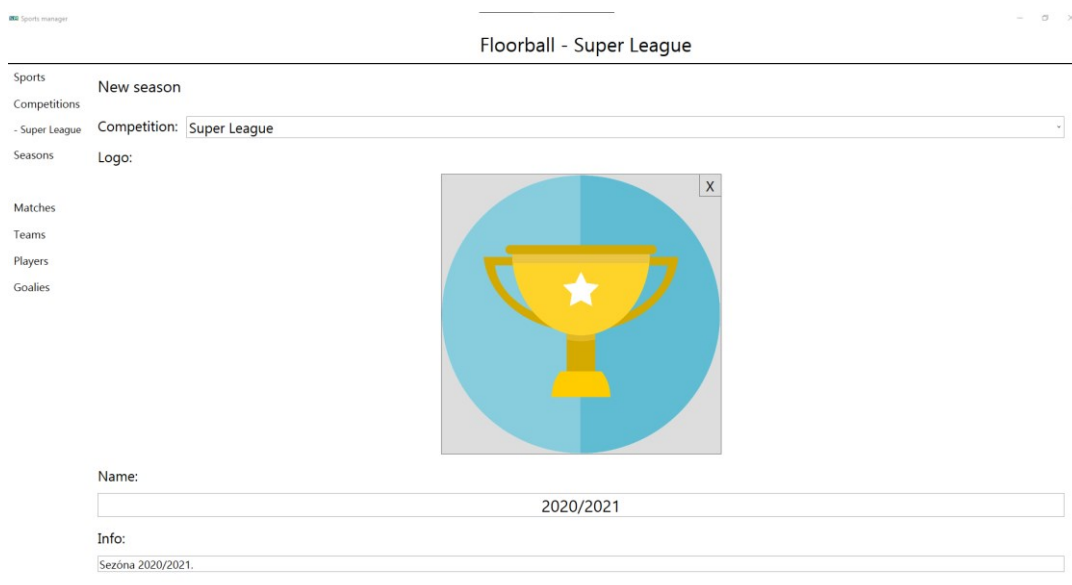
Obrázok 4: Výber súťaže



Obrázok 5: Tvorba súťaže

## Vytvorenie novej sezóny

Novú sezónu možno vytvoriť po kliknutí na tlačidlo “Add” v pravej dolnej časti okna v pohľade zoznamu sezón. Vo formulári je potrebné vyplniť meno novej sezóny a zvoliť súťaž. Je tu možnosť pridania loga a ďalších informácií.



Obrázok 6: Tvorba sezóny

Nižšie sa nachádza formulár pre vytvorenie nového tímu, ktorý sa zapojí do sezóny. Taktiež je tu možné pridať logo tímu, názov, informácie, dátum vzniku, štát pôsobenia a momentálny status tímu, čo znamená, či ešte aktívne pôsobí. Tlačidlom “Add team” sa tím pridá do sezóny. V prípade, že chcete pridať už existujúci tím z inej súťaže alebo sezóny, pod formulárom tvorby tímu sa nachádza výberový zoznam všetkých tímov v rámci športu. Opäť svoj výber potvrdíte tlačidlom “Add team”.

Sports manager

Floorball - Super League

Sports Add new team

Competitions Logo:

- Super League

Seasons

Matches

Teams

Players

Goalies

Active:

Name: Sharks

Info: Florbalový tím z Bratislavy založený v roku 2016.

Country: Slovakia Date of creation: 26. 4. 2016

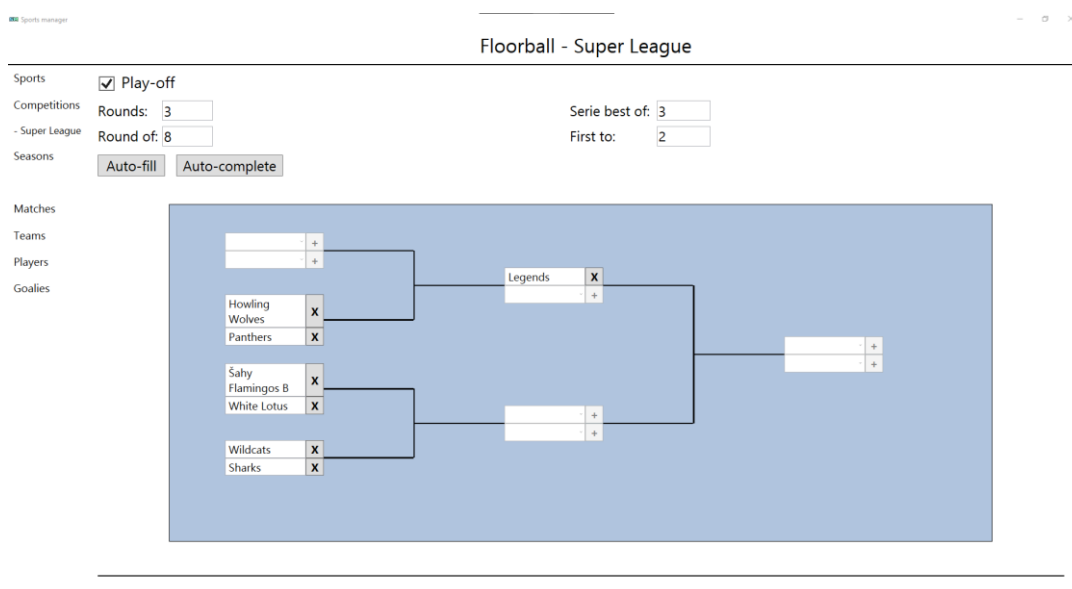
Obrázok 7: Tvorba tímu

Nižšie sa nachádza zoznam prihlásených tímov s ich menami, logami a tlačidlom “Remove”, ktorým tím zo sezóny odstránite.



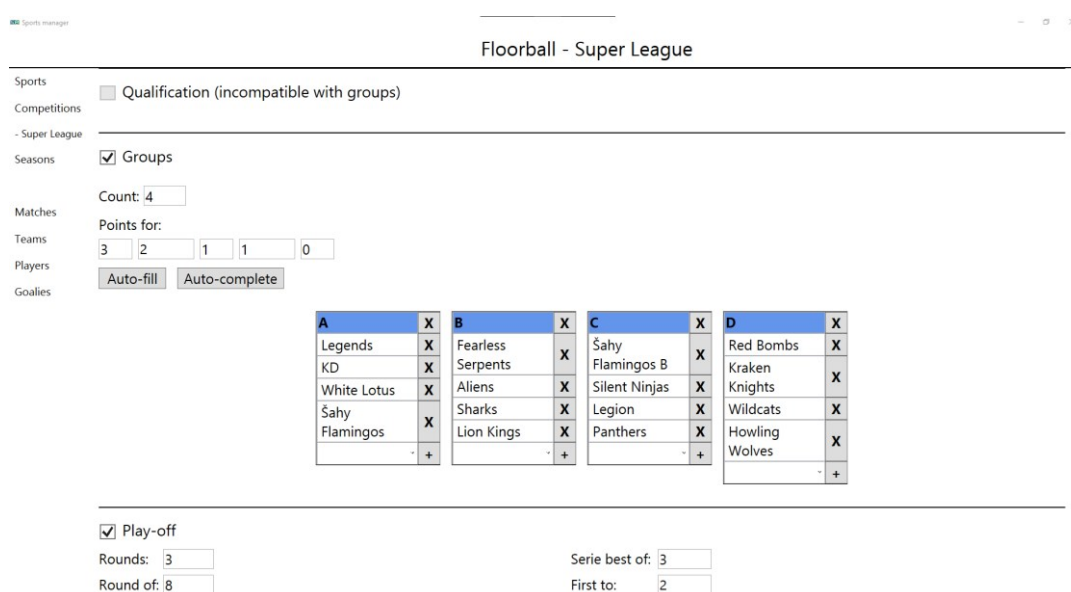
Obrázok 8: Zoznam tímov pri tvorbe sezóny

Nakoniec je potrebné zvoliť formát súťaže. Na výber sú tri hlavné možnosti. Kvalifikácia (“Qualification”), pozostáva z viacerých pavúkov, pričom ich veľkosť si zvolíte vyplnením príslušného formulára. “Rounds” znamená počet kôl pavúka a “Round of” počet zápasov v najnižšom kole, pričom vyplnením jedného z nich sa druhá položka automaticky prepočíta. Formát play-off (“Play-off”) má nastavenia rovnaké kvalifikácii, pričom tu je možné hrať jednu sériu na viac víťazných zápasov. Počet víťazstiev potrebných na výhru série sa nastavuje vyplnením políček “Serie best of”, čo znamená maximálny počet zápasov v sérii, a “First to”, čo znamená potrebný počet víťazstiev.



Obrázok 9: Nastavenie play-off

Tímy sa do pavúka nasadzujú výberom zo zoznamu tímov v sérii a stlačením tlačidla “+”. Následne je možné tím zo série odstrániť tlačidlom “X”. Posledný formát na výber je skupinová časť (“Groups”), v ktorej sa poradie tímov počíta v rámci ich skupín. Tento formát sa používa najmä v ligových súťažiach. Políčkou “Count” nastavíte počet skupín. V políčkach “Points for” nastavíte postupne body získané tímom za výhru v riadnom hracom čase, výhru v predĺžení, remízu, prehru v predĺžení a prehru v riadnom hracom čase. Tímy sa do skupín pridávajú a odoberajú rovnako ako v pavúku.



Obrázok 10: Nastavenie skupinovej časti

Formát sezóny môže pozostávať z jednej časti alebo dvoch častí, pričom skupinová časť a kvalifikácia nemôžu byť vybrané naraz. Tímy je do každej z častí možné automaticky náhodne nasadiť. Tlačidlom “Auto-fill” sa nasadia všetky tímy a aktuálne nasadenie sa odstráni. Tlačidlom “Auto-complete” sa nasadia všetky ešte nenasadené tímy.

Sezóna sa vytvorí a uloží po kliknutí na tlačidlo “Save”.

## Výber súťaže / sezóny

Súťaž alebo sezónu je možné vybrať skrz zoznam súťaží alebo zoznam sezón, a to kliknutím na ňu. Výberom konkrétnej súťaže alebo sezóny sa všetky ostatné zoznamy budú zobrazovať vyfiltrované tak, aby obsahovali iba údaje z aktuálneho

výberu. Pre zrušenie výberu súťaže alebo sezóny je potrebné navigovať sa späť na zoznam súťaží.

## Pohľady zoznamov

V pohľade zoznamu sú zobrazené všetky objekty v tabuľkovej forme. Tabuľku je možné ľubovoľne triediť podľa stĺpca, a to kliknutím na jeho hlavičku. Pre triedenie podľa viacerých stĺpcov je potrebné nechať stlačený kláves Shift a kliknúť na dané hlavičky stĺpcov v danom poradí. Zoznamy zobrazujú základné informácie o objektoch a ich štatistiky. Tieto údaje je tiež možné odfiltrovať odkliknutím zaškrtnutých políčok nad tabuľkou. “Logo” alebo “Photo” prepína viditeľnosť stĺpca s logom alebo fotografiou. “Info” prepína viditeľnosť stĺpcov základných údajov. “Stats” a “Extended stats” prepína viditeľnosť štatistík a rozšírených štatistík.

Logo	Name	Games played	Wins	Overtime wins	Ties	Overtime losses	Losses	Goals	Goals against	Goal difference	Assists	Penalty minutes
	Legion	6	0	0	0	1	5	10	24	-14	7	8
	Kraken Knights	6	0	1	0	0	5	13	21	-8	12	8
	Panthers	6	1	0	0	0	5	9	23	-14	9	22
	Sharks	6	1	2	0	2	1	13	11	2	10	24
	Red Bombs	6	1	0	0	2	3	15	21	-6	11	28
	Lion Kings	6	1	0	0	1	4	10	23	-13	9	4
	Šahy Flamingos	6	2	1	0	0	3	17	20	-3	17	22
	Šahy Flamingos B	6	2	2	0	0	2	16	16	0	15	2
	KD	6	3	1	0	0	2	19	16	3	15	20
	White Lotus	6	3	1	0	1	1	15	13	2	10	12

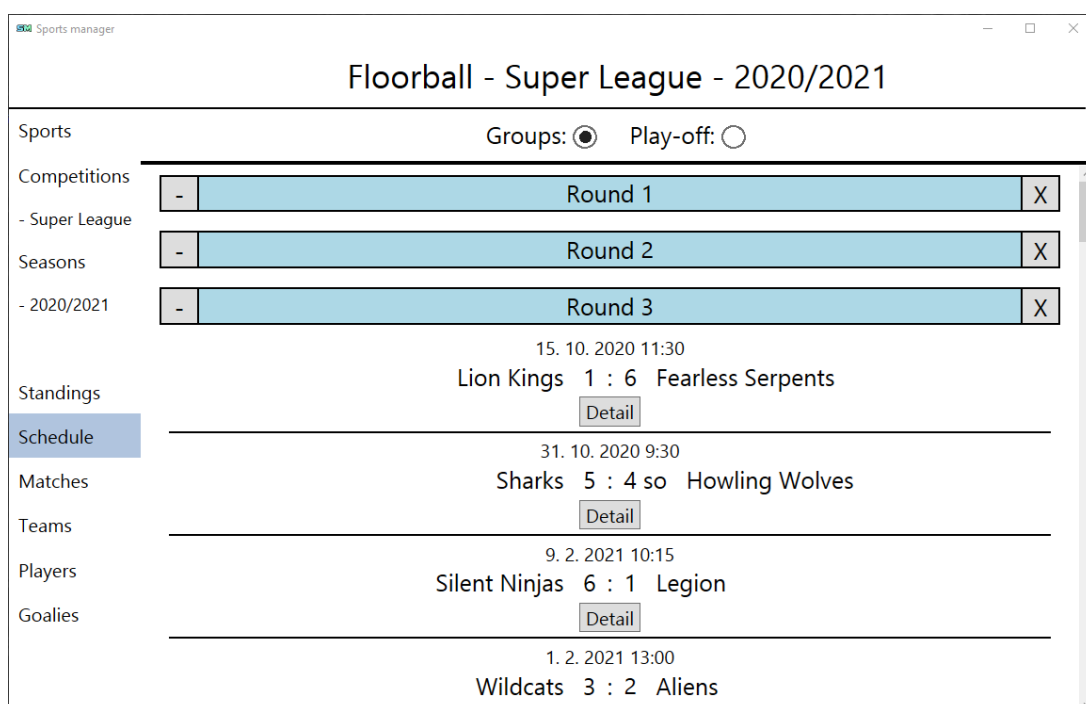
Obrázok 11: Zoznam tímov

Tlačidlami “Export PDF” a “XLSX” sa tabuľka exportuje vo formáte PDF alebo XLSX a uloží na Vami zvolenom mieste. Vyplnením políčka “Top:” sa exportuje iba udaný počet prvých riadkov. Ak je políčko nevyplnené, exportujú sa všetky riadky. Dvojitým kliknutím na niektorý z riadkov sa navigujete na detailný pohľad daného objektu.

## Rozpis zápasov

Po výbere sezóny sa v navigačnej lište zobrazí možnosť “Schedule”, cez ktorú sa dá navigovať na pohľad rozpisu zápasov. V hornej časti okna je možné prepínať sa medzi časťami sezóny zaškrtavajúcimi políčkami “Groups” (skupinová časť), “Qualification” (kvalifikácia) a “Play-off”.

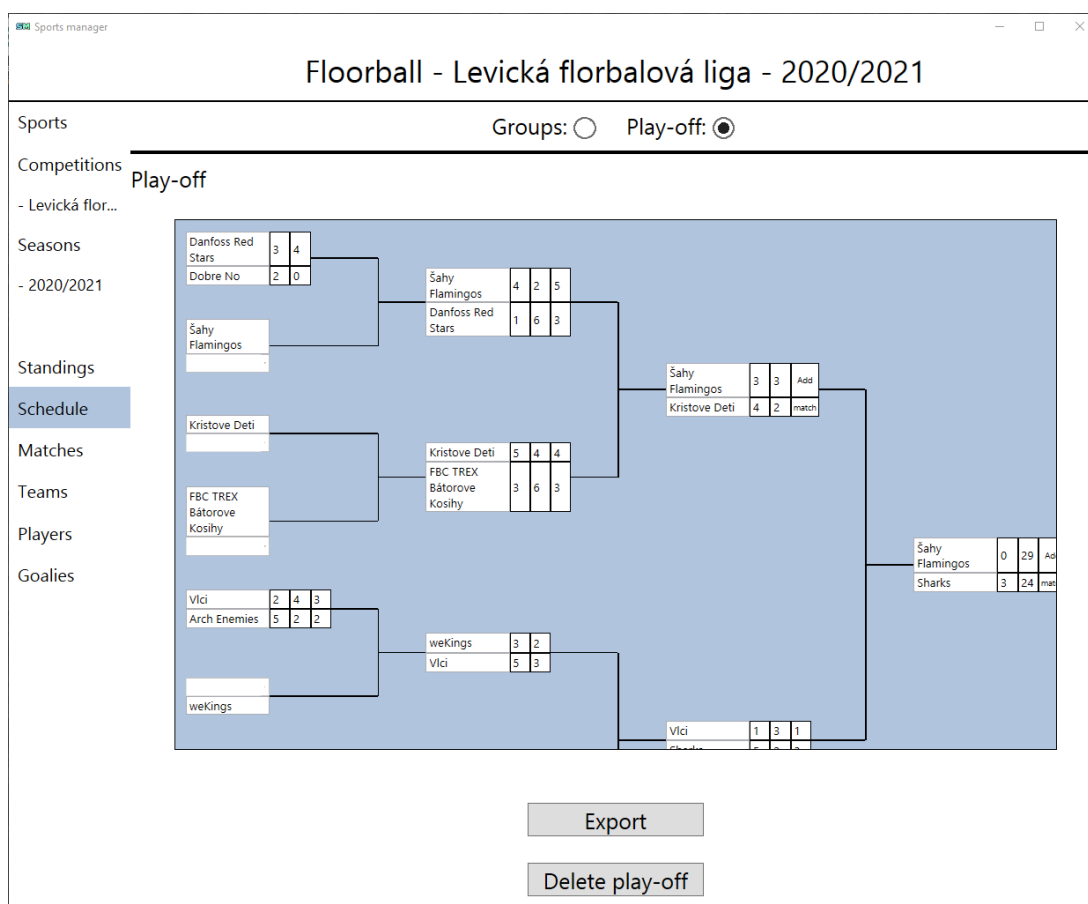
Pre skupinovú časť je možné pridávať kolá (“Add round”), do ktorých sa potom môžu pridávať zápasy (“Add match”). Každé kolo sa dá zabaliť a znova rozbaľiť tlačidlom “-” v ľavom rohu. Kolo aj so všetkými jeho zápasmi sa odstráni tlačidlom “X” na pravej strane. V rozbalenom kole vidieť jeho zápasy s časom stretnutia, výsledkom a súperiacimi tímami. Pod každým zápasom je tlačidlo “Detail”, ktorým sa je možné navigovať na detail daného zápasu.



Obrázok 12: Rozpis zápasov základnej časti

Pri kvalifikácii sa zobrazuje pavúk, ktorý je možné myšou posúvať a kolieskom na myši približovať a oddiaľovať. Po kliknutí na zápas sa zobrazí detail zápasu. Pridať nový zápas je možné kliknutím na tlačidlo “Add match” v danej sérii. Medzi jednotlivými pavúkmi kvalifikácie sa prepína pomocou šípok v ľavom a pravom hornom rohu.

V play-off časti sa taktiež zobrazuje pavúk ako pri kvalifikácii. V prípade, že play-off ešte nezačalo, zobrazuje sa tu iba tlačidlo “Start play-off”, ktorým sa play-off spustí a predošlá časť sezóny zablokuje na editáciu. Zaškrtnutím políčka “Seed competitors” sa pri štarte play-off súťažiaci automaticky nasadia, buď náhodne, alebo podľa výsledku predošlej časti sezóny. Tlačidlom “Delete play-off” je možné potom celé play-off zmazať aj so všetkými jeho zápasmi. Pavúka kvalifikácie aj play-off je možné exportovať do formátu PNG tlačidlom “Export” nachádzajúcim sa pod pavúkom.



Obrázok 13: Rozpis play-off

## Pohľad tímu

V pohľade tímu sa zobrazujú základné údaje o ňom ako logo, názov a ďalšie informácie. Pod nimi je zoznam súťaží, ktorých sa tím zúčastnil. Súťaž je možné rozbaľiť tlačidlom “-”. Po rozbaľení sa zobrazia sezóny danej súťaže, ktorých sa tím zúčastnil. Tie je rovnako možné rozbaľiť. Následne, po rozbaľení sezóny, môžeme

vidieť súpisťu tímu v danej sezóne, prípadne ju meniť. Hráča zo súpisťy odstránime tlačidlom “X”, pričom je to možné, iba ak sa hráč ešte nezúčastnil žiadneho zápasu.

Pod súpisťou je formulár na zmenu zápisu hráča a pridanie nového hráča. Vo formulári “Edit player” v zozname “Select player” vyberieme hráča zo súpisťy, v zozname “Position” vyberieme novú pozíciu a v poličku “Number” vyberieme nové číslo hráča. Tlačidlom “Edit player” upraveného hráča uložíme.

V časti “Existing” vyberieme už existujúceho hráča v danom športe, ktorého môžeme zapísať na súpisťu tlačidlom “Add player”. Jeho pozíciu a číslo nastavíme v časti “Add player”.

Floorball - Super League - 2020/2021			
Sports			
Competitions			
- Super League	<b>Super League</b>		-
Seasons			
- 2020/2021	<b>2020/2021</b>		-
Number	Name	Position	
2	Micah Desai	right wing	X
8	Sophie Goldman	center	X
14	Silas Weiner	goaltender	X
17	Beau Rollins	right wing	X
29	Angel Alexander	center	X
33	Caleb Hodge	defenseman	X
35	Luka Hoyle	goaltender	X
46	Leon Underwood	right wing	X
54	Theodore Puckett	left wing	X
57	Malachi Adkins	defenseman	X
66	Henry Barton	left wing	X
68	Roman Hewitt	right wing	X
72	Carter Alexander	left wing	X
73	Xander Clapp	left wing	X

Obrázok 14: Súpisťu tímu

## Pridanie nového hráča

Nový hráč sa pridáva priamo na súpisťu niektorého z tímov. Služi na to samostatný formulár “New” v pohľade tímu, ktorý je viditeľný po rozbalení konkrétnej súťaže

a sezóny tímu, do ktorej hráča chceme zapísať. Formulár pozostáva z nasledovných políčk:

- First Name – krstné meno
- Last Name – priezvisko
- Birthdate – dátum narodenia
- Gender – pohlavie
- Height – výška v centimetroch
- Weight – váha v kilogramoch
- Plays with – pravák / ľavák
- Citizenship – občianstvo
- Birthplace City – mesto narodenia
- Birthplace country – štát narodenia
- Info – ďalšie informácie
- Status – aktívny / neaktívny

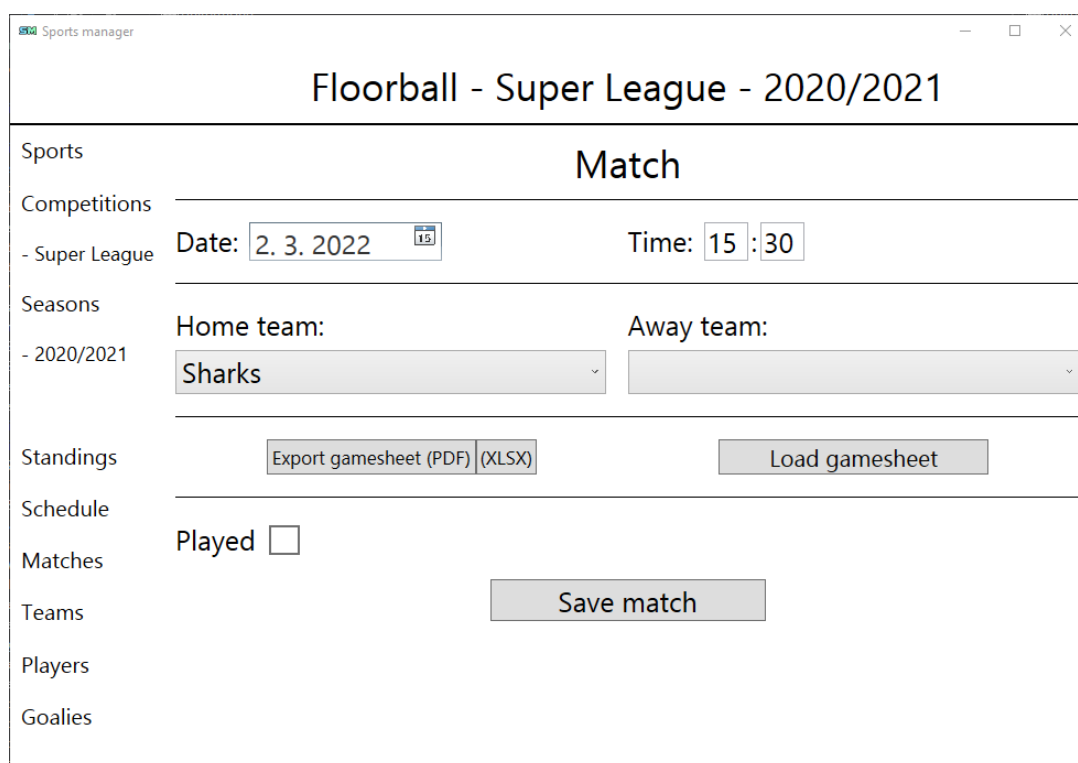
Pozíciu hráča a jeho číslo vyplníme v časti formulára “Add player” vyššie. Tlačidlom “Add player” hráča pridáme.

Floorball - Super League - 2020/2021	
<b>Edit player:</b>	
Number (1-99)	Position
Select player	<b>Edit player</b>
<b>Add player:</b>	
Number (1-99)	Position
<b>Existing:</b>	
Select player	<b>Add player</b>
<b>New:</b>	
First Name	Last Name
Birthdate	Gender
Height (cm)	Weight (kg)
Plays with	Citizenship
Birthplace City	Birthplace country
Info	
Status	<b>Add player</b>

Obrázok 15: Formulár pridania hráča

## Zápis zápasu

Formulár pre vyplnenie údajov o zápase začína položkami “Date” (dátum), “Time” (čas stretnutia), “Home team” a “Away team” (domáci tím a hostia). Pod nimi sa nachádza tlačidlo “Export gamesheet PDF/XLSX”, ktorým je možné stiahnuť si prázdny zápis o zápase s predvyplnenými súpiskami hráčov. Neskôr vyplnené údaje z naskenovaného zápisu, je možné tlačidlom “Load gamesheet” automaticky načítať do formulára. Táto akcia môže byť zablokovaná antivírusom. V takom prípade bude potrebné povoliť jej vykonanie.



The screenshot shows a web application window titled "Sports manager" with a sub-header "Floorball - Super League - 2020/2021". The main content area is titled "Match" and contains the following elements:

- Sports:** Match
- Competitions:** - Super League
- Date:** 2. 3. 2022 (with a calendar icon)
- Time:** 15:30
- Seasons:** - 2020/2021
- Home team:** Sharks (dropdown menu)
- Away team:** (empty dropdown menu)
- Buttons:** "Export gamesheet (PDF) (XLSX)" and "Load gamesheet"
- Schedule:** Played
- Matches:** Save match (button)
- Teams:**
- Players:**
- Goalies:**

Obrázok 16: Základný formulár zápisu zápasu

Zápas je možné uložiť aj bez vyplnenia údajov ako nadchádzajúci zápas. V takom prípade ostane políčko “Played” nevyplnené. Políčko “Forfeit” slúži na zaznamenanie kontumácie zápasu, pričom sa vyberie strana, v ktorej prospech, kontumácia nastala.

Ak je políčko “Played” zaškrtnuté, je možné vyplňať samotný záznam zápasu. Najskôr je potrebné vyplniť súpisky oboch tímov, inak hráči nebudú k dispozícii pre výber v nasledujúcich formulároch.

Floorball - Super League - 2020/2021

Home roster				Away roster			
Number	Name	Present		Number	Name	Present	
7	Luis Rodgers	<input checked="" type="checkbox"/>		2	Micah Desai	<input type="checkbox"/>	
43	Bryson McKenzie	<input checked="" type="checkbox"/>		8	Sophie Goldman	<input checked="" type="checkbox"/>	
45	Jason Moss	<input checked="" type="checkbox"/>		14	Silas Weiner	<input checked="" type="checkbox"/>	
47	Bentley Watts	<input type="checkbox"/>		17	Beau Rollins	<input type="checkbox"/>	
48	Tristan Hodge	<input checked="" type="checkbox"/>		29	Angel Alexander	<input checked="" type="checkbox"/>	
49	Enzo Sparks	<input type="checkbox"/>		33	Caleb Hodge	<input type="checkbox"/>	
55	Max Hoyle	<input checked="" type="checkbox"/>		35	Luka Hoyle	<input type="checkbox"/>	
62	Axel Sykes	<input checked="" type="checkbox"/>		46	Leon Underwood	<input checked="" type="checkbox"/>	
65	Quinn Abrams	<input type="checkbox"/>		54	Theodore Puckett	<input checked="" type="checkbox"/>	
67	Nathaniel Wilkerson	<input type="checkbox"/>		57	Malachi Adkins	<input checked="" type="checkbox"/>	
72	James Snyder	<input checked="" type="checkbox"/>		66	Henry Barton	<input checked="" type="checkbox"/>	
81	Alexander Middleton	<input type="checkbox"/>		68	Roman Hewitt	<input checked="" type="checkbox"/>	
87	Owen Justice	<input checked="" type="checkbox"/>		72	Carter Alexander	<input checked="" type="checkbox"/>	
95	Arthur Hobbs	<input checked="" type="checkbox"/>		73	Xander Clapp	<input type="checkbox"/>	
97	Connor Song	<input checked="" type="checkbox"/>		90	Miles Byrd	<input type="checkbox"/>	
				98	Jackson Glover	<input checked="" type="checkbox"/>	

Obrázok 17: Súpisový zoznam tímov v zápase

Pred vyplnením samotných udalostí v zápase, je potrebné udať počet periód zápasu, a ich dĺžku v minútach. Taktiež sa políčkami “Overtime” a “Shootout” zaznamená, či sa hralo predĺženie, alebo zápas rozhodol až rozstrel v samostatných nájazdoch. Potom nasleduje päť formulárov, pre každý typ udalostí zápasu, rozdelených podľa periód.

---

Periods:  Overtime

Period duration:  Shootout

---

Obrázok 18: Nastavenie periód zápasu

Prvý formulár “Goals” slúži na zaznamenanie gólov v zápase. Vyplnia sa nasledovné položky:

- Period time – čas strelenia gólu v perióde
- Side – skórujúci tím
- Scorer – číslo strelca gólu
- Assist – číslo nahrávača na gól (prázdne ak nahrávač nie je)
- Penalty shot – zaškrtnúť, ak gól padol z trestného strielania
- Own goal – zaškrtnúť v prípade vlastného gólu
- Delayed penalty – zaškrtnúť, ak gól padol počas avizovaného vylúčenia

V prípade trestného strielania, je potrebné údaj zaznamenať aj vo formulári “Penalty shots” pre trestné strielania.

### Goals

Period 1

Time	Side	Goal	Assist	Type	
5:17	Home	43	7	normal	<b>X</b>
8:46	Away	54	14	delayed penalty	<b>X</b>

Period time:  :       Side:

Scorer:       Assist:

Penalty shot       Own goal

Delayed penalty      

Obrázok 19: Formulár pre pridávanie gólov

Nasleduje formulár “Penalties” pre zaznamenanie vylúčení. Formulár obsahuje položky:

- Period time – čas udelenia vylúčenia v perióde
- Side – previnený tím
- Reason – dôvod vylúčenia
- Type – typ vylúčenia
- Player – číslo vylúčeného hráča

### Penalties

Period 1

Time	Side	Player	Reason	Type	
8:12	Away	98	Hooking	2-minute bench	<b>X</b>

Period time:  :       Side:

Reason:

Type:

Player:      

Obrázok 20: Formulár pre pridávanie vylúčení

Formulárom “Penalty shots” sa zaznamenávajú všetky trestné strielania. Je tu potrebné zaznamenať aj premenené trestné strielania. Vyplnia sa položky:

- Period time – čas v perióde
- Side – tím, ktorému bolo udelené trestné strielanie
- Player – číslo hráča, ktorý sa trestného strielania zúčastnil (nie brankár)
- Was goal – zaškrtnúť, ak bolo trestné strielanie premenené

## Penalty shots

Period 1

Time	Side	Player	Was goal	
4:10	Away	29	False	X

Period time:  :       Side:

Player:       Was goal

Obrázok 21: Formulár pre pridávanie trestných strielaní

“Goalkeeper shifts” je formulár pre zaznamenanie striedaní brankárov. Striedanie končí koncom periódy, výmenou brankára alebo odvolaním brankára. Vyplnia sa položky:

- Period start time – čas začiatku striedania v perióde
- Period end time – čas konca striedania v perióde
- Side – tím brankára
- Goaltender – číslo brankára

## Goaltender shifts

### Period 1

Start Time	End Time	Side	Player	
0:00	20:00	Home	55	X
0:00	19:13	Away	68	X

Period start time:  :

Period end time:  :

Side:

Goaltender:

Add

Obrázok 22: Formulár pre pridávanie striedaní brankárov

Time-outs sa zaznamenávajú vo formulári “Time-outs”. Tu je potrebné vyplniť čas, kedy sa time-out zobral (“Period time”), a tím, ktorý si ho zobral (“Side”).

### Overtime

Time	Side	
2:47	Home	X

Period time:  :

Side:

Add

Obrázok 23: Formulár pre pridávanie time-outov

Každá udalosť sa pridá tlačidlom “Add” v prislúchajúcom formulári. V prípade chyby v zázname sa tlačidlom “X”, vedľa udalosti, daný údaj odstráni.

V prípade rozstrelu v samostatných nájazdoch bude dostupný aj formulár “Shootout”. V poličku “Series” je potrebné nastaviť počet sérií. Jedna séria pozostáva z jedného pokusu domácich a jedného pokusu hostí. Potom sa pre každý pokus vyplnia údaje:

- Side – tím, ktorý útočí
- Player – číslo hráča útočiaceho tímu

- Goaltender – číslo brankára brániaceho sa tímu
- Was goal – zaškrtnúť, ak padol gól

Floorball - Super League - 2020/2021

Sports: FLOORBALL 0 - 0 Date: [ ]

Competitions: Add

- Super League

Seasons: Shootout

- 2020/2021 Series: 5

Serie	Side	Player	Goaltender	Was goal	
1	Away	42	64	<input checked="" type="checkbox"/>	X
1	Home	44	63	<input checked="" type="checkbox"/>	X
2	Away	53	64	<input type="checkbox"/>	X
2	Home	35	63	<input type="checkbox"/>	X
3	Away	84	64	<input type="checkbox"/>	X
3	Home	5	63	<input type="checkbox"/>	X
4	Away	98	64	<input checked="" type="checkbox"/>	X
4	Home	32	63	<input checked="" type="checkbox"/>	X
5	Away	32	64	<input type="checkbox"/>	X
5	Home	35	63	<input checked="" type="checkbox"/>	X

Process data

Check data

Obrázok 24: Formulár samostatných nájazdov

Ak sú všetky údaje vyplnené, je ich potrebné spracovať tlačidlom “Process data”. Súčasne sa tak aj skontroluje, či nedochádza k časovým kolíziám udalostí. Ak niektoré udalosti nastali v rovnaký čas, zobrazí sa vyskakovacie okno, ktoré na to upozorní. Zároveň sa zobrazí formulár s danými kolíziami, v ktorom bude možné určiť skutočné poradie udalostí v zápase pomocou šípok.

Process data

Period 1	0:00	Home	goaltender in	^	v
Period 1	0:00	Away	goaltender in	^	v
Period 1	5:17	Home	goal scored by 43	^	v
Period 1	5:17	Home	2-minute bench penalty for 87	^	v

Check data

Obrázok 25: Formulár pre určenie poradia kolidujúcich udalostí

Posledným krokom pred uložením zápasu je stlačenie tlačidla “Check data”, ktoré rovnako skontroluje časové kolízie, ale tentoraz s koncami vylúčení. Zobrazí sa formulár, v ktorom bude možné pomocou šípok určiť, či daná udalosť nastala ešte počas vylúčenia alebo až po jeho konci. Ak je udalosť na ľavej strane pod nápisom

“In”, tak nastala počas vylúčenia. Naopak, ak ostane na pravej strane pod nápisom “Out”, tak nastala až po konci vylúčenia.

Penalty from 5:17 to 7:17				
In			Out	
7:17	Away	goal scored by 29	<	>

Obrázok 26: Formulár pre upresnenie kolízií pri koncoch vylúčení

Po vyplnení všetkých údajov a upresnení poradia udalostí, je možné zápas uložiť tlačidlom “Save”.

## Detail zápasu

V pohľade detailu zápasu je k dispozícii prehľad skóre, súpisky tímov a výpis priebehu zápasu po periódach. V pravom hornom rohu sú tri tlačidlá. “Edit” slúži na úpravu zápasu, “Delete” na odstránenie zápasu a “Export” na exportáciu priebehu zápasu vo formáte PDF.

The screenshot shows the 'Detail zápasu' (Match Detail) view in the Sports Manager application. The match is titled 'Floorball - Super League - 2020/2021' and took place on 29. 12. 2020 at 8:15. The teams are Šahy Flamingos and Legion. The final score is 6 : 3, with a period breakdown of (3:2, 1:0, 2:1). The interface includes a sidebar with navigation options like 'Standings', 'Schedule', 'Matches', 'Teams', 'Players', and 'Goalies'. At the top right, there are buttons for 'Export', 'Edit', and 'Delete'. Below the team logos, the 'Rosters' section lists players for both teams.

Rosters	
6# Greyson Watkins (center)	2# Daniel Robertson (left wing)
12# Josiah Branch (right wing)	4# Timothy Davidson (defenseman)
16# Jeremiah Blanchard (defenseman)	14# Wesley Becker (center)
20# Justin Hallman (goalkeeper)	18# David Carter (left wing)

Obrázok 27: Detail zápasu

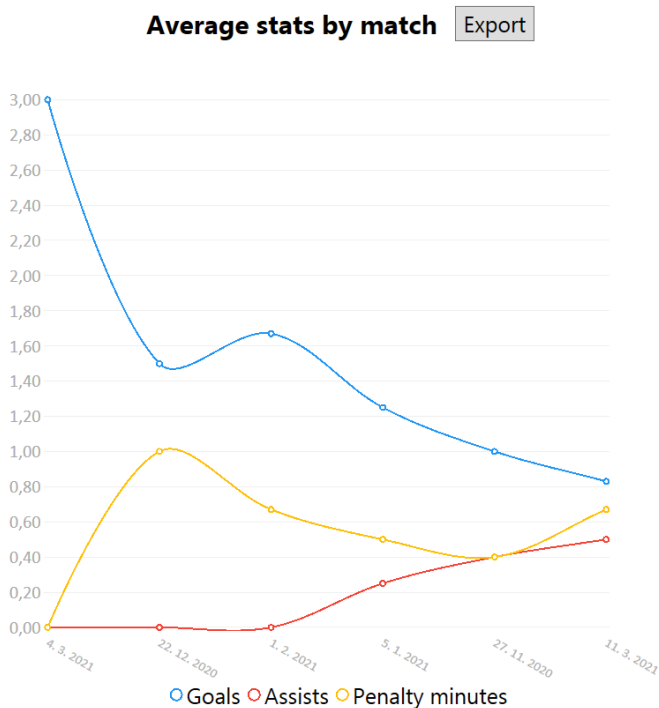
Floorball - Super League - 2020/2021

		Period 1	
Sports		Goaltender in - J. Holloway	0:00
Competitions	- Super League		0:00 Goaltender in - A. Lane
Seasons	- 2020/2021	GOAL - M. Whitaker (J. Blanchard)	2:02 GOAL - D. Robertson (A. Merritt)
Standings			4:01 GOAL - C. McNamara (J. Kennedy)
Schedule		GOAL - K. Moore (J. Holloway)	8:16
Matches		GOAL - J. Sharpe (J. Holloway)	8:59
Teams		Goaltender out - J. Holloway	10:00
Players			10:00 Goaltender out - A. Lane
Goalies			
		Period 2	
		Goaltender in - J. Holloway	0:00
			0:00 Goaltender in - A. Lane
		GOAL - M. Whitaker (J. Blanchard)	1:43 2-minute bench penalty - C. McNamara (High stick)
		Goaltender out - J. Holloway	9:39
			10:00
			10:00 Goaltender out - A. Lane
		Period 3	
		Goaltender in - J. Holloway	0:00
			0:00 Goaltender in - A. Lane
		GOAL - J. Branch (B. Barton)	0:03
			1:32 GOAL (PS) - A. Graham
		GOAL - B. Barton (J. Holloway)	1:51
		Goaltender out - J. Holloway	10:00
			10:00 Goaltender out - A. Lane

Obrázok 28: Sumár priebehu zápasu

## Detail hráča

V pohľade detailu hráča vidieť najskôr základné informácie hráča. Pod nimi sa nachádzajú tri grafy, ktoré znázorňujú výkonnosť hráča v jeho zápasoch, a to dosiahnuté góly, asistencie a vylúčenia. Prvý graf znázorňuje dané štatistiky v každom zápase. Druhý graf dané štatistiky akumuluje od prvého zápasu. V treťom grafe môžeme vidieť meniaci sa priemer dosiahnutých štatistík hráča počas zápasov. Grafy je možné exportovať vo formáte PNG tlačidlom “Export” pri názve grafu.



Obrázok 29: Graf zobrazujúci priemer štatistík hráča po zápasoch

Pod grafmi sa nachádza zoznam súťaží, ktorých sa hráč zúčastnil, najskôr ako hráč v poli, a potom ako brankár. Súťaž je možné rozbaľiť tlačidlom “-”, pre zobrazenie sezón. Po rozbaľení sezóny sa objaví zoznam zápasov, ktorých sa hráč zúčastnil. Pod každou súťažou a sezónou je výpis dosiahnutých štatistík hráča v danej súťaži a sezóne.

**Competitions**

As player:

Super League									
Games played	Goals	Assists	Points	Penalty minutes					
5	1	2	3	0					

2020/2021

Games played	Goals	Assists	Points	Penalty minutes					
5	1	2	3	0					

Part of season	Date	Time	Home	Score	Away	Goals	Assists	Penalty minutes		
Group	Thursday, 11. 3. 2021	10:30	KD	3 : 2	Fearless Serpents	5	5	6		
Group	Thursday, 28. 1. 2021	12:30	Legends	1 : 4	Fearless Serpents	5	5	0		
Group	Wednesday, 27. 1. 2021	11:00	Lion Kings	1 : 5	Fearless Serpents	6	6	0		
Group	Tuesday, 19. 1. 2021	11:15	Legends	1 : 0	Fearless Serpents	1	1	2		
Group	Thursday, 15. 10. 2020	11:30	Lion Kings	1 : 6	Fearless Serpents	7	5	2		

As goaltender:

Obrázok 30: Zoznam účasti hráča

## Editácia a odstánenie dát

V pohľade detailu konkrétneho objektu sa v pravom hornom rohu nachádzajú tlačidlá “Edit” na editáciu objektu, alebo “Delete” na jeho odstránenie. Pri editácii objektu sa zobrazí formulár podobný tomu, ktorým sa objekt vytvoril.

## Poradie v skupinách

Aktuálne poradie tímov je k dispozícii v pohľade dosiahnuteľnom cez záložku “Standings” v navigačnej lište. Navrchu sa nachádza položka “Winner” pre vyhlásenie víťaza sezóny. Výber je nutné potvrdiť tlačidlom “Declare”. Po tom, ako bude víťaz vyhlásený, sa akákoľvek editácia v sezóne zablokuje. Víťaza je možné zrušiť tlačidlom “Delete”.

Floorball - Super League - 2020/2021

Sports: Winner:  Declare

Competitions: After round: Round 8

- Super League

Seasons: - 2020/2021

Export (PDF) (XLSX)

Group A

Name	GP	W	WO	T	LO	L	G	GA	GD	PM	P
KD	6	3	1	0	0	2	19	16	3	20	11
Legends	6	3	1	0	0	2	11	11	0	20	11
Fearless Serpents	6	3	0	0	1	2	20	10	10	0	10
Lion Kings	6	1	0	0	1	4	10	23	-13	4	4

Group B

Name	GP	W	WO	T	LO	L	G	GA	GD	PM	P
Howling Wolves	6	4	0	0	2	0	20	8	12	6	14
Šahy Flamingos B	6	2	2	0	0	2	16	16	0	2	10
Sharks	6	1	2	0	2	1	13	11	2	24	9
Panthers	6	1	0	0	0	5	9	23	-14	22	3

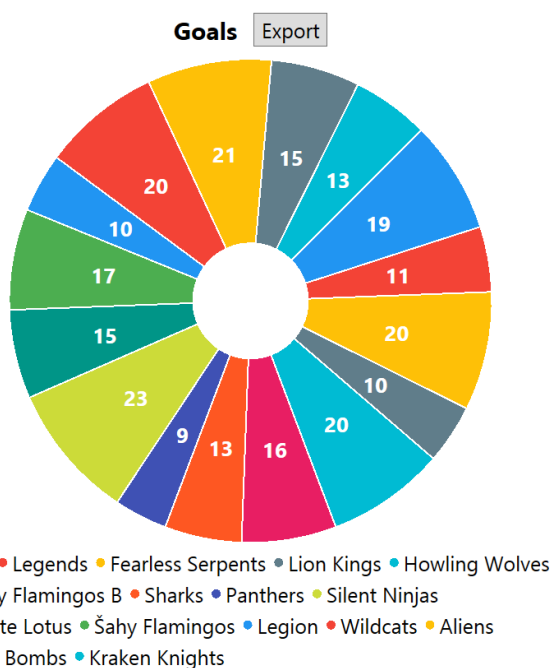
Group C

Name	GP	W	WO	T	LO	L	G	GA	GD	PM	P
Silent Ninjas	6	5	0	0	0	1	23	8	15	10	15
White Lotus	6	3	1	0	1	1	15	13	2	12	12

Obrázok 31: Aktuálne poradie tímov v sezóne podľa skupín

Potom sa v tomto pohľade zobrazujú tabuľky poradia skupín po danom kole. Kolo, po ktorom sa majú tabuľky zobrazit', je možné zvolit' položkou "After round" v ľavom hornom rohu. Tabuľky sa dajú exportovať vo formáte PDF alebo XLSX tlačidlami "Export (PDF) / (XLSX)".

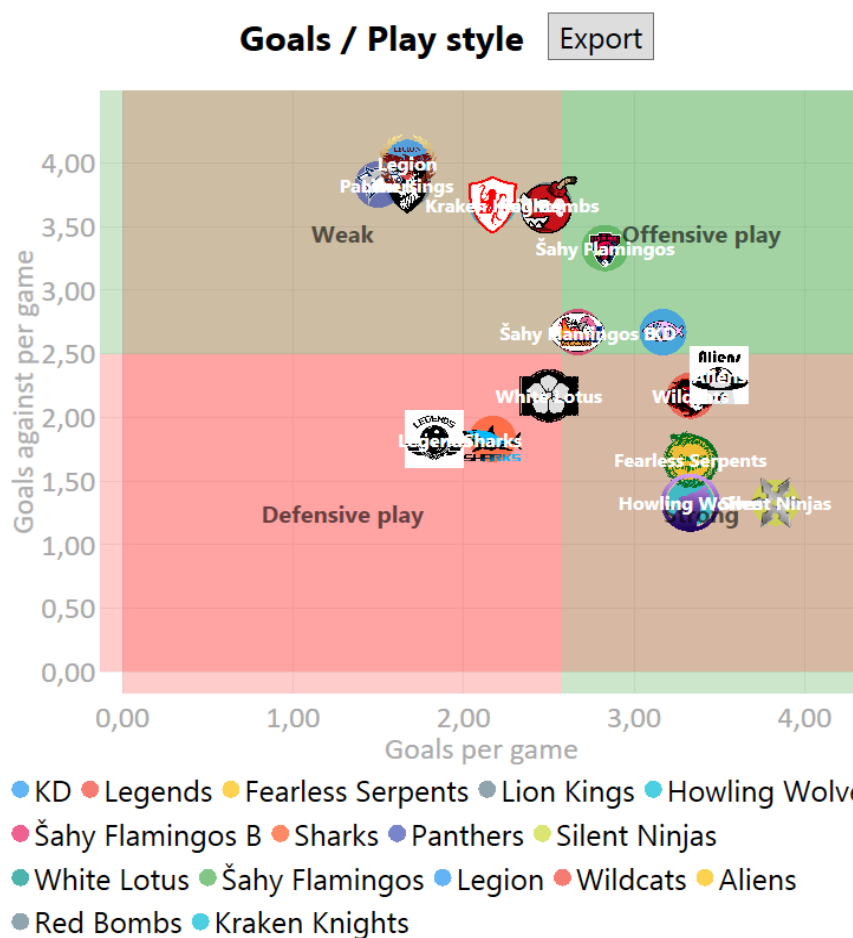
Pod tabuľkami sa nachádza šesť grafov. Prvé tri sú koláčové grafy, znázorňujúce podiel gólov, asistencií a trestných minút v základnej časti do daného kola.



Obrázok 32: Koláčový graf znázorňujúci podiel gólov tímov v sezóne

Ďalšie tri grafy sú bodové. Prvý z nich zobrazuje počet strelených a inkasovaných gólov tímov, čím sa tímy rozdelia do štyroch kvartálov grafu. Kvartály majú nasledovný význam, podľa ich názvu:

- Strong – tím strieľa viac gólov ako ich inkasuje, je to silný tím
- Weak – tím inkasuje viac gólov ako ich strieľa, je to slabý tím
- Offensive play – tím hrá otvorenú hru, veľa gólov strieľa ale aj inkasuje
- Defensive play – tím hrá defenzívnu hru, v jeho zápasoch padá málo gólov



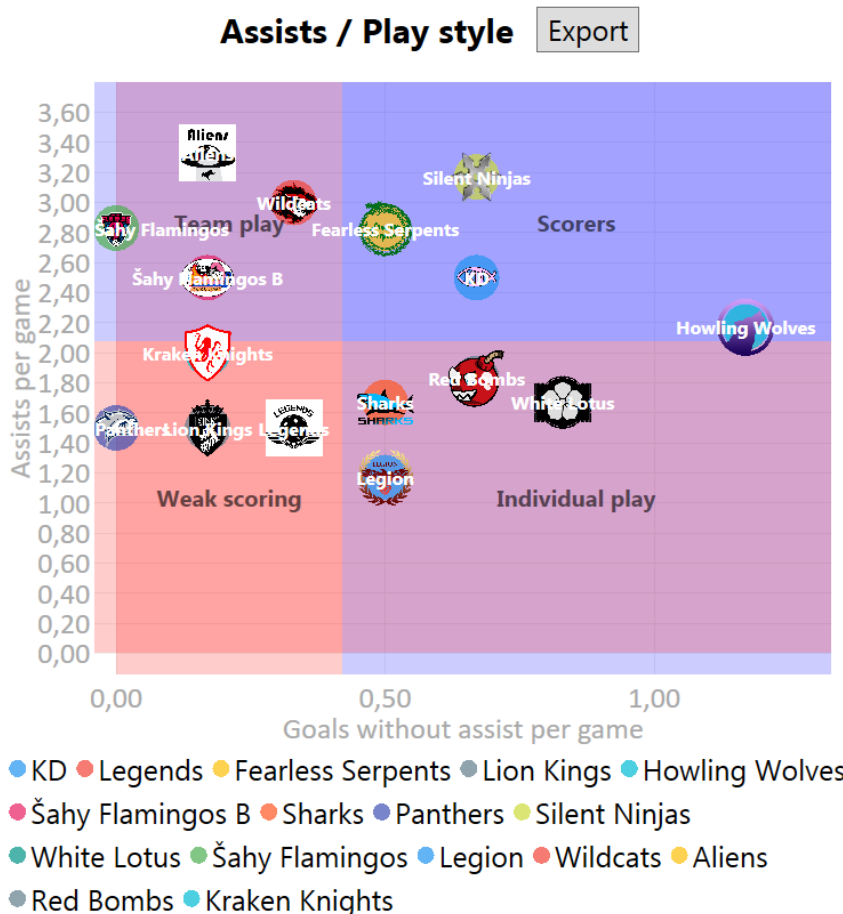
Obrázok 33: Graf zobrazujúci štýl hry tímu podľa strelených a inkasovaných gólov

Druhý graf zobrazuje počet gólov tímov skórovaných s asistenciou a bez nej.

Rozdelenie tímov do kvartálov grafu znamená:

- Team play – tím hrá tímovo, väčšina gólov padlo po asistencii, zakladanie útokov už z obrany, tímu chýbajú individuálni hráči
- Individual scoring – tím má silných individuálnych hráčov, väčšina gólov padá bez asistencie, pár hráčov ťahá celý tím

- Scorers – tím má silných individuálnych strelcov, ale zároveň hrá tímovú hru, strieľa nadpriemerný počet gólov
- Weak scoring – tím takmer vôbec neskóruje, nemá dobrý útok



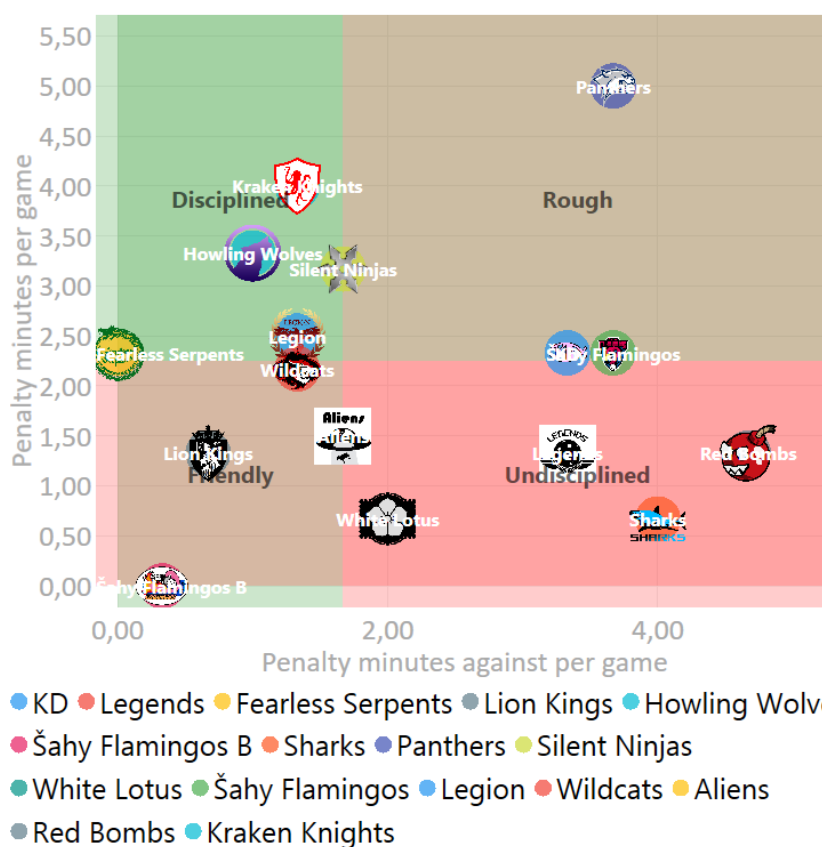
Obrázok 34: Graf zobrazujúci štýl hry tímu podľa počtu gólov s asistenciou a bez nej

Posledný tretí graf znázorňuje počet udelených trestných minút tímu a jeho súperovi. Z grafu možno podľa kvartálov vyčítať:

- Rough – tím hrá so svojím súperom tvrdú hru, v stretnutiach je veľký počet trestných minút na oboch stranách
- Friendly – tím hrá so svojím súperom čistú hru, v stretnutiach sa takmer nevyučuje
- Disciplined – tím sa nenechá vylúčiť, hrá disciplinovane, donúti súperu urobiť chyby, aby bol vylúčený
- Undisciplined – tím hrá nedisciplinovane, hrá špinavú hru, súperovi pravidelne ponúka výhodu vlastnými vylúčeniami

## Penalties / Play style

Export



Obrázok 35: Graf zobrazujúci štýl hry tímu podľa obdržaných trestných minút tímu a jeho súpera