

Slovenská technická univerzita

Fakulta informatiky a informačných technológií

Ilkovičova 3, 812 19 Bratislava

Databázové systémy

Projekt

Matúš Barabás

Cvičiaci: Ing. Eduard Kuric
Študijný odbor: Informatika
Ročník: 2. Bc.
Akademický rok: 2015/2016

Obsah

1.	Znenie zadania.....	3
2.	Špecifikácia scenárov a opis ich implementácie	3
2.1	Trieda Account_controller.....	3
2.2	Trieda All_matches_controller	3
2.3	Trieda Goalies_controller.....	4
2.4	Trieda Matches_controller.....	4
2.5	Trieda Players_controller	4
2.6	Trieda Teams_controller	5
2.7	Trieda Trade_controller.....	5
3.	Diagram logického dátového modelu.....	6
4.	Diagram fyzického dátového modelu.....	7
5.	Návrh riešenia/Implementácia.....	8
5.1	Programové prostredie	8
5.1.1	GUI základnej obrazovky	9
5.1.2	GUI Players	9
5.1.3	GUI Goalies.....	10
5.1.4	GUI Teams	10
5.1.5	GUI Matches a All matches	11
5.1.6	GUI Account	11
5.2	Návrhové rozhodnutia	12
6.	Záver/Zhodnotenie	12
	Zdroje	13

1. Znenie zadania

Vo vami zvolenom prostredí vytvorte databázovú aplikáciu, **ktorá komplexne rieši minimálne 6 scenárov** vo vami zvolenej doméne. Presný rozsah a konkretizáciu scenárov si dohodnete s Vaším cvičiacim na cvičení. Aplikáciu vytvoríte v dvoch iteráciách. V prvej iterácii, postavenej nad relačnou databázou, musí aplikácia realizovať tieto všeobecné scenáre:

- Vytvorenie nového záznamu,
- Aktualizácia existujúceho záznamu,
- Vymazanie záznamu,
- Zobrazenie prehľadu viacerých záznamov (spolu vybranou základnou štatistikou),
- Zobrazenie konkrétneho záznamu,
- Filtrovanie záznamov spĺňajúcich určité kritériá zadané používateľom.

Aplikácia môže mať konzolové alebo grafické rozhranie. Je dôležité aby scenáre boli realizované realisticky - teda aby aplikácia (a teda aj jej používateľské rozhranie) naozaj poskytovala časť funkcionality tak, ako by ju očakával zákazník v danej doméne.

2. Špecifikácia scenárov a opis ich implementácie

2.1 Trieda *Account_controller*

- Zobrazenie všetkých záznamov z tabuľky *betting_events* z databázy spolu s kurzami zoradených podľa dátumu stlačením tlačidla *SHOW BETTING EVENTS*. Použitím JOINOV a cudzích kľúčov som sa dostal z tabuľky *betting_events* až k tabuľke *teams*, z ktorých som si vybral potrebné názvy tímov.
- Výpočet eventuálnej výhry na základe kurzu z tabuľky *betting_events* a stavených peňazí. Z tabuľky *betting_events* si najprv vyberiem na základe *id* zadaného používateľom kurz, ktorý vynásobím sumou peňazí, ktoré si taktiež zadáva používateľ.
- Pridanie stávkovej príležitosti do tabuľky *account* na základe *id*, ktoré zadá používateľ, z tabuľky *betting_events*.
- Zobrazenie všetkých záznamov tabuľky *account*. Použitý JOIN a cudzie kľúče, aby som sa dostal ku skratkám tímov v tabuľke *teams*.
- Aktualizácia údajov v tabuľke *account* – označenie zápasu podľa *id* ako vyhraný, prehraný alebo zrušený a na základe tejto hodnoty aktualizácia údajov v stĺpcoch *profit*, *loss* a *success*. Použitie CASE, to znamená, ak napríklad zápas bol označený ako WIN, do stĺpca *profit* sa pridá rozdiel možnej výhry a stavenej sumy, do stĺpca *loss* pridá 0 a do stĺpca *success* pridá hodnotu 1.
- Zobrazenie základnej štatistiky účtu – aktuálna suma na účte, zisk/strata, úspešnosť, počet vyhraných zápasov, počet prehraných a počet zrušených zápasov z tabuľky *account*.
- Vymazanie stávkovej príležitosti z tabuľky *account* na základe *id* zadanej používateľom.

2.2 Trieda *All_matches_controller*

- Zobrazenie všetkých záznamov o zápasoch odohraných medzi jednotlivými tímami alebo zobrazenie vybraných zápasov filtrovaných na základe tímu zadaného používateľom z tabuľky *games_14_15*. Použitý cudzí kľúč a JOIN na tabuľku *teams*, z ktorej sa vyberá meno daného domáceho a hosťujúceho tímu.

2.3 Trieda *Goalies_controller*

Primárnou tabuľkou pre všetky tieto scenáre je tabuľka *goalies*.

- Zobrazenie 10 najlepších brankárov na základe percentuálnej úspešnosti zákrokov. Podmienkou je, aby mali viac ako 20 odohraných zápasov. Použitý SELECT, ktorý vyberá potrebné stĺpce, JOIN na tabuľku *teams*, z ktorej sa ako obvykle vyberá skratka tímu, v ktorom pôsobí brankár. Všetko je to zoradené podľa percentuálnej úspešnosti zákrokov vzostupne.
- Zobrazenie jedného brankára, na ktorého bolo vyslaných najviac striel. Podobným spôsobom ako predchádzajúci scenár, použitá agregáčna funkcia max.
- Zobrazenie jedného brankára, ktorý má na svojom konte najviac víťazstiev.
- Zobrazenie priemerného veku brankárov. Použitá agregáčna funkcia priemeru na stĺpec *age*. Zobrazenie výsledku do Label.
- Zobrazenie priemerného počtu striel na všetkých brankárov.
- Zobrazenie priemerného počtu zásahov všetkých brankárov.
- Zobrazenie priemerného počtu trestných minút na jedného brankára.
- Zobrazenie celkového súčtu všetkých striel, ktoré smerovali na všetkých brankárov. Použitá agregáčna funkcia súčtu.
- Zobrazenie celkového súčtu všetkých zásahov všetkých brankárov.
- Zobrazenie celkového súčtu všetkých trestných minút všetkých brankárov.
- Zobrazenie všetkých brankárov.
- Zobrazenie brankárov na základe tímu zadaného používateľom.
- Zobrazenie konkrétneho brankára vyhladaného v databáze podľa jeho mena zadaného používateľom.
- Vymazanie konkrétneho brankára z databázy na základe mena, ktoré musí používateľ zadať do systému.

2.4 Trieda *Matches_controller*

- Pridanie záznamu – zápasu do tabuľky *games_14_15* na základe hodnôt zadaných používateľom. INSERT, v ktorom je použitý SUBSELECT, ktorý z tabuľky *teams* vyberá *id* tímu, ktorého názov zadal používateľ do prideleného TextFieldu.
- Vloženie záznamov do dvoch tabuliek. Najprv vloženie záznamu do tabuľky *games_14_15* na základe hodnôt zadaných do TextFieldov, a na základe *id* z tejto tabuľky pridanie záznamu spolu s ďalšími informáciami a výške kurzov zadaných používateľom do tabuľky *betting_events*. Použitý SUBSELECT.

2.5 Trieda *Players_controller*

- Zobrazenie hráčov, ktorí prestúpili z tímu do iného tímu spolu s dátumami odkedy a dokedy v danom tíme pôsobili. SELECT, ktorý použije JOIN na tabuľku *teams*, z ktorej si vyberie skratku tímu a JOIN na tabuľku *players_14_15*, z ktorej vyberá meno daného hráča.
- Zobrazenie prehľadu všetkých hráčov v databáze.
- Zobrazenie hráčov na základe tímu zadaného používateľom.
- Zobrazenie 10 najlepších hráčov na základe celkového získaného počtu bodov.
- Zobrazenie 10 najlepších útočníkov na základe získaných bodov. Títo hráči sa hľadajú na základe stĺpca *pos*, kde sa musí nachádzať buď C, RW alebo LW.
- Zobrazenie 10 najlepších obrancov na základe získaných bodov. V stĺpci *pos* musí byť D.
- Zobrazenie hráča, ktorý strelil najviac gólov. Použitá agregáčna funkcia max.
- Zobrazenie hráča, ktorý má na svojom konte najviac asistencií. Použitá agregáčna funkcia max.

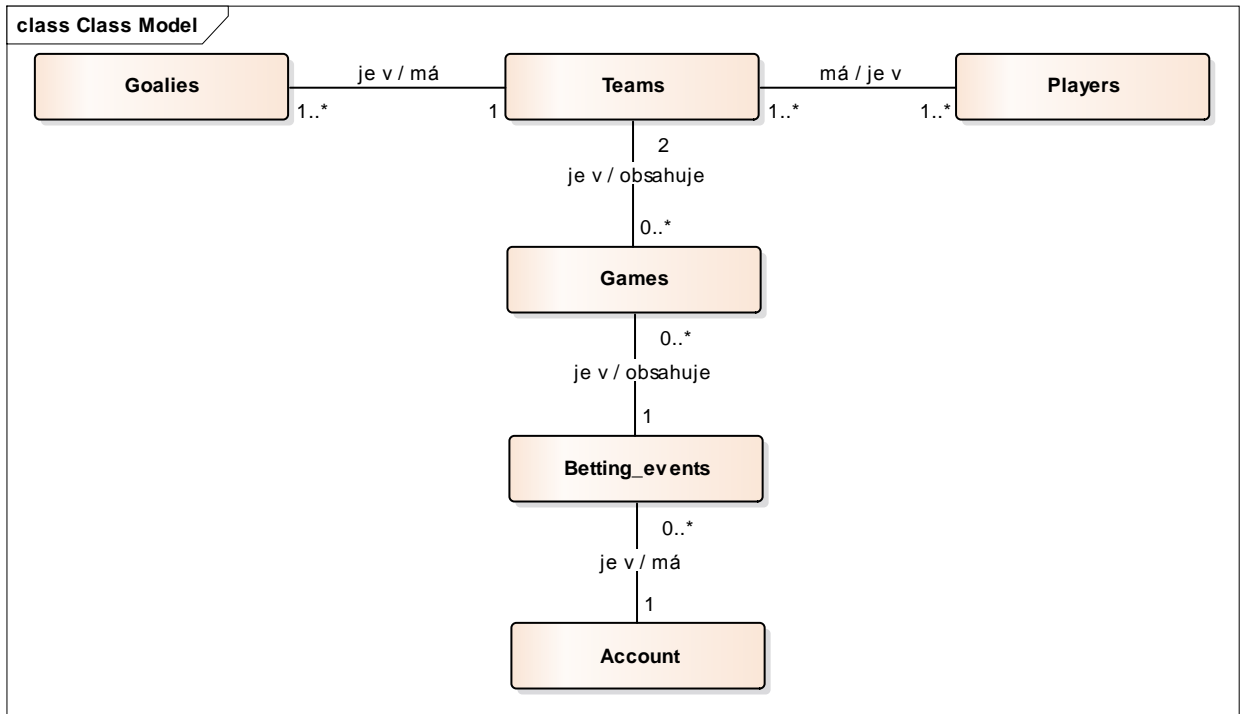
2.6 Trieda Teams_controller

- Zobrazenie prehľadu všetkých tímov z tabuľky *teams*. Vyberám celú tabuľku, dáta z nej uložíam do ObservableListu a zobrazím v príslušnej tabuľke.
- Pridanie nového tímu do databázy. INSERT na základe údajov, ktoré používateľ zadá do príslušných TextFieldov.

2.7 Trieda Trade_controller

- Aktualizácia záznamu – vloženie dátumu, ktorým sa končí pôsobenie hráča v danom tíme do príslušného stĺpca v tabuľke *player_in_team* a na základe tejto udalosti pridanie nového záznamu obsahujúci nový tím a dátum začiatku pôsobenia v tomto tíme do tej istej tabuľky.

3. Diagram logického dátového modelu

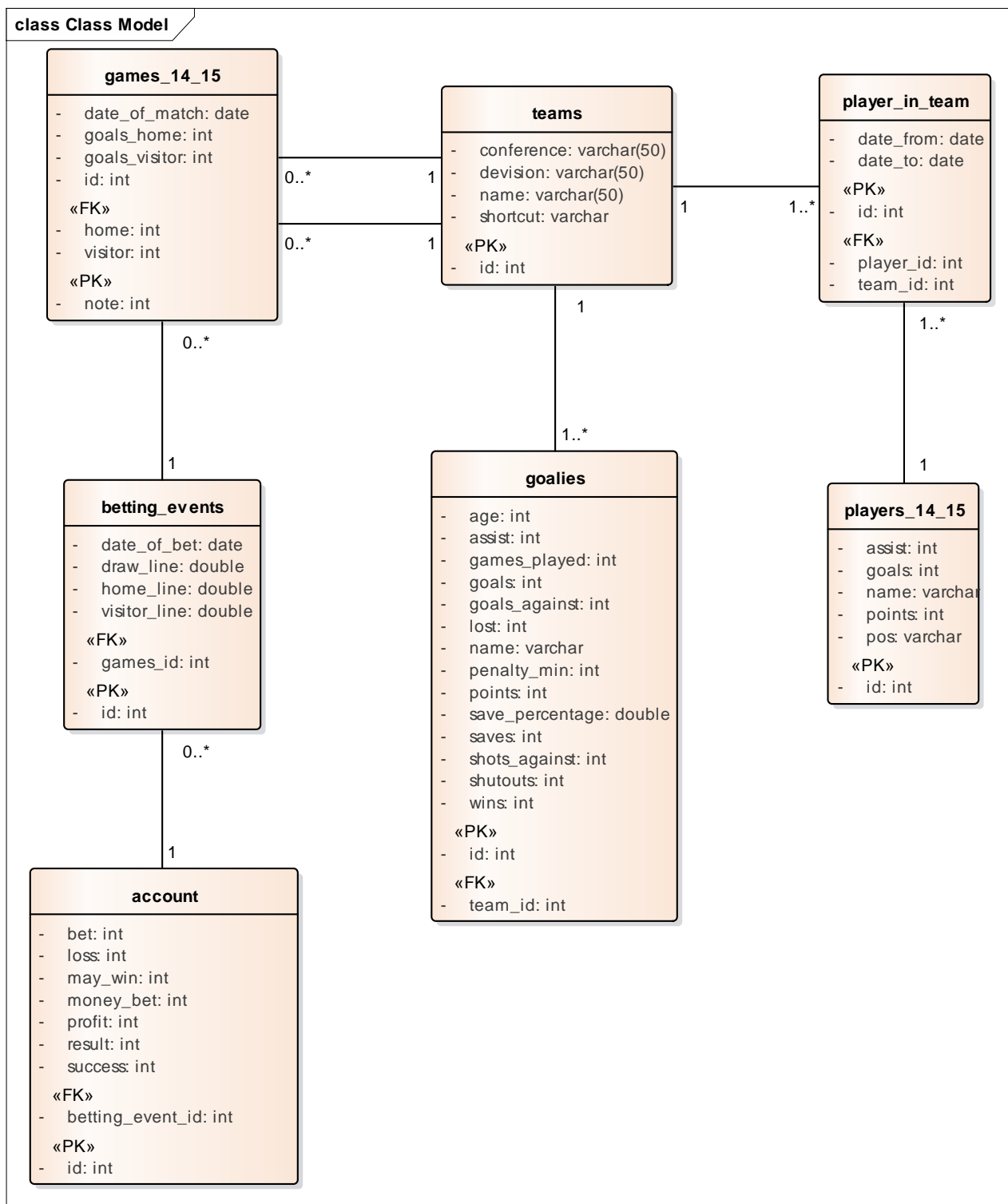


Obrázok 1 Diagram logického dátového modelu

Diagram logického dátového modelu obsahuje názvy entít a jednotlivé vzťahy medzi entitami. Tento model obsahuje taktiež základnú logiku fungovania aplikácie, návrh na názvy a počet tabuliek v databáze. Je to zatiaľ však iba všeobecná rovina návrhu, pretože je potrebné vyriešiť niekoľko problémov vychádzajúcich z daného modelu. Ide najmä o rozdelenie vzťahu m*n pridaním väzobnej entity, ale aj doplnenie potrebných atribútov a primárnych a cudzích kľúčov. Vzťahy jednotlivých entít:

- Brankár je v 1 tíme.
- Tím má viac brankárov.
- Jeden hráč môže byť vo viacerých tímoch.
- Tím má viacero hráčov.
- Jeden zápas obsahuje 2 tímy.
- Tím je v / vyskytuje sa vo viacerých zápasoch.
- Jeden zápas je len jednou stávkovou udalosťou.
- Stávkové udalosti obsahujú viacero zápasov.
- Jedna stávková udalosť môže byť len v jednom účte.
- Účet môže mať viacero stávkových udalostí.

4. Diagram fyzického datového modelu



Obrázok 2 Diagram fyzického datového modelu

Diagram fyzického dátového modelu je už komplexnejším zobrazením, ako bude aplikácia fungovať. Obsahuje už názvy jednotlivých tabuliek, tak ako sú uložené v databáze, atribúty jednotlivých tabuliek, primárne a cudzie kľúče, väzobné tabuľky a vzťahy medzi entitami.

Tabuľka *account* obsahuje dáta o stávkovej udalosti, keďže sa cudzím kľúčom odkazuje na tabuľku *betting_events*. Obsahuje teda stávku na daný zápas, možnú výhru, zisk a stratu z daného zápasu, úspešnosť atď. Slúži teda na evidenciu informácií o stávkovom účte.

Tabuľka *betting_events* obsahuje informácie o stávkovej udalosti, to znamená mená oboch tímov a kurzy na domáci, hosťujúci tím alebo remízu.

Tabuľka *games_14_15* obsahuje všetky odohrané zápasy, názvy tímov, počet gólov jednotlivých tímov, dátum zápasu a poznámku v prípade, že daný zápas skončil po predĺžení alebo po nájazdoch.

Tabuľka *teams* obsahuje názov tímu, konferenciu a názov divízie, v ktorej je umiestnený.

Tabuľka *goalies* obsahuje informácie o všetkých brankároch. Ide najmä o meno brankára, vek, počet odohraných zápasov, počet víťazstiev, prehier, inkasovaných gólov, informáciu o strelách ktoré smerovali na daného brankára, o počte zásahov, úspešnosti zásahov a ďalšie.

Tabuľka *player_in_team* je väzobnou tabuľkou. Umožňuje uschovanie informácií o prestupoch jednotlivých hráčov z jedného tímu do iného. Obsahuje informáciu o dátume od začiatku a dátumu konca pôsobenie v danom tíme reprezentovanom ako cudzí kľúč na tabuľku *teams*, v ktorej sa nachádza už konkrétny názov tímu.

Tabuľka *players_14_15* je úložiskom pre všetkých hráčov. Obsahuje informáciu o mene, pozícii, počte gólov, asistencií a celkový počet získaných bodov.

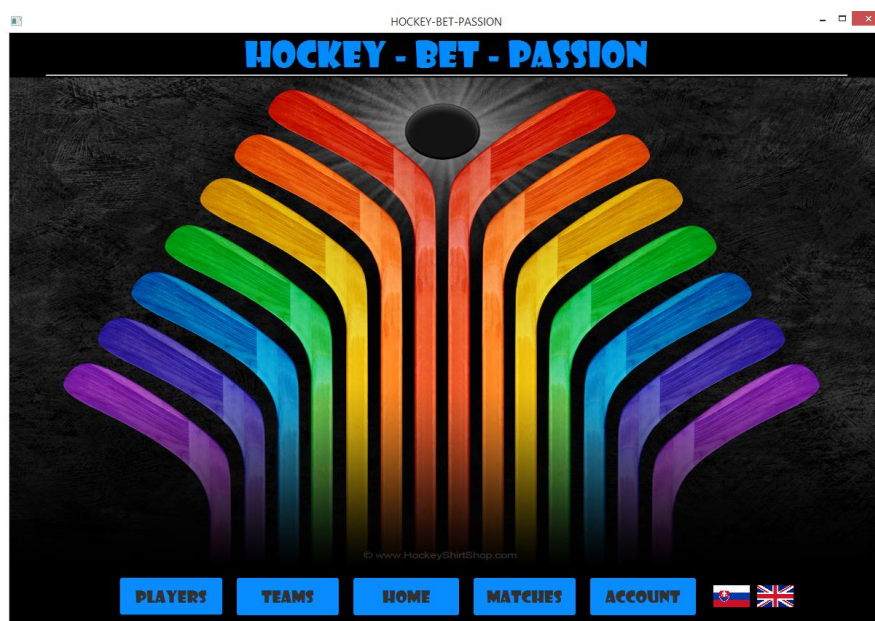
5. Návrh riešenia/Implementácia

Hlavnou a nosnou myšlienkou môjho projektu sú hokejové zápasy v spojení so stávkovou kanceláriou. V časti, ktorá sa týka hokejových zápasov, moja aplikácia zaznamenáva celé dianie, to znamená všetky tímy rozdelené do divízií a konferencií, jednotlivých hráčov daných tímov spolu s ich základnou štatistikou, brankárov taktiež spolu s ich štatistikou. Tieto zápasy môžu byť pridané aj ako stávkové príležitosti. Tie tvoria druhú časť mojej aplikácie, do ktorej patrí zaznamenávanie stavených zápasov, výsledky, či bola daná stávka úspešná, neúspešná alebo zápas bol zrušený, počítanie eventuálnej výhry a taktiež štatistika stávkového účtu, kde sa zobrazuje suma na účte, zisk, úspešnosť tipov atď.

5.1 Programové prostredie

Zadanie je implementované v programovacom jazyku Java, v prostredí Eclipse s využitím databázy PostgreSQL. Je použitá taktiež technológia Hibernate, ktorá uľahčuje prácu s transakciami a umožňuje mapovanie objektov, čo som však v mojom projekte nepoužil. Na vytvorenie používateľského rozhrania som použil JavaFX a nástroj SceneBuilder, ktorý mi po vytvorení obrazoviek vygeneroval xml súbory. Pracoval som aj s jazykom CSS, pomocou ktorého som si upravil návrh a napr. zmenu farby tlačidla pri prechode nad ním alebo jeho stlačenia.

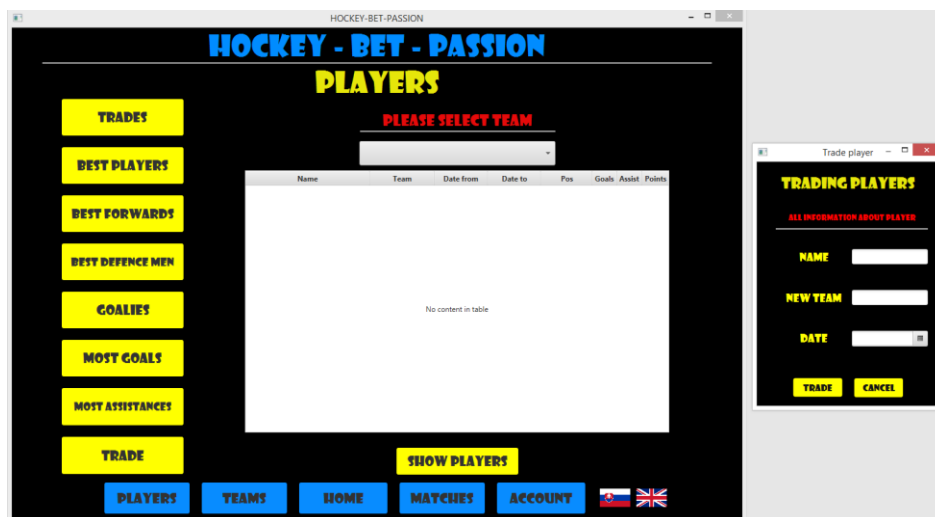
5.1.1 GUI základnej obrazovky



Obrázok 3 Základná obrazovka

Základná obrazovka obsahuje 5 tlačidiel, ktoré umožňujú prepínanie medzi jednotlivými obrazovkami. Obsahuje taktiež výber jazyka, ten však funguje iba pre toto základné okno. Je implementované ako BorderPane a ostatné okná sa umiestňujú do jeho strednej časti.

5.1.2 GUI Players



Obrázok 4 Players obrazovka

Obrazovka Players zobrazuje základnú štatistiku o hráčoch a samotných hráčov z jednotlivých tímov. Okno Trading players umožňuje prestup hráča z tímu do tímu.

5.1.3 GUI Goalies

GOALIES STATISTICS

TABLE OF ALL GOALIES

Name	Age	Team	Games	Wins	Lost	Goals ag.	Shots	Saves	Saves perc.	Shutouts	G	A	P	PM
Jake Allen	24	STL	37	22	7	79	909	830	0.913	4	0	2	2	2
Frederik Andersen	25	ANA	54	35	12	123	1436	1313	0.914	3	0	3	3	4
Craig Anderson	33	OTT	35	14	13	87	1134	1047	0.923	3	0	0	0	4
Richard Bachman	27	EDM	6	3	2	15	168	153	0.911	1	0	0	0	0
Niklas Backstrom	36	MIN	19	5	7	51	452	401	0.887	0	0	0	0	2
Jonathan Bernier	26	TOR	58	21	28	152	1735	1583	0.912	2	0	1	1	0
Reto Berra	28	COL	18	5	4	33	403	370	0.918	1	0	0	0	0
Ben Bishop	28	TBL	62	40	13	136	1620	1484	0.916	4	0	4	4	4
Sergei Bobrovsky	26	CBJ	51	30	17	134	1632	1498	0.918	2	0	2	2	4
Martin Brodeur	42	STL	7	3	3	17	169	152	0.899	1	0	0	0	0
Laurent Brossoit	21	EDM	1	0	1	2	51	49	0.961	0	0	0	0	0
Ilya Bryzgalov	34	ANA	8	1	4	23	150	127	0.847	0	0	2	2	0
Tyler Bunz	22	EDM	1	0	0	3	12	9	0.75	0	0	0	0	0
Corey Crawford	30	CHI	57	32	20	126	1661	1535	0.924	2	0	1	1	8
John Curry	30	MIN	2	0	0	5	25	20	0.8	0	0	0	0	0

NAME AGE TEAM **DELETE** **SHOW** **CANCEL**

Obrázok 5 Goalies obrazovka

Obrazovka Goalies zobrazuje všetkých brankárov spolu ich štatistikami.

5.1.4 GUI Teams

HOCKEY - BET - PASSION TEAMS

TABLE OF ALL TEAMS

Name	Conference	Division
No content in table		

ADD OR DELETE TEAM NHL

TEAM NAME

CONFERENCE

DEVISION

ADD

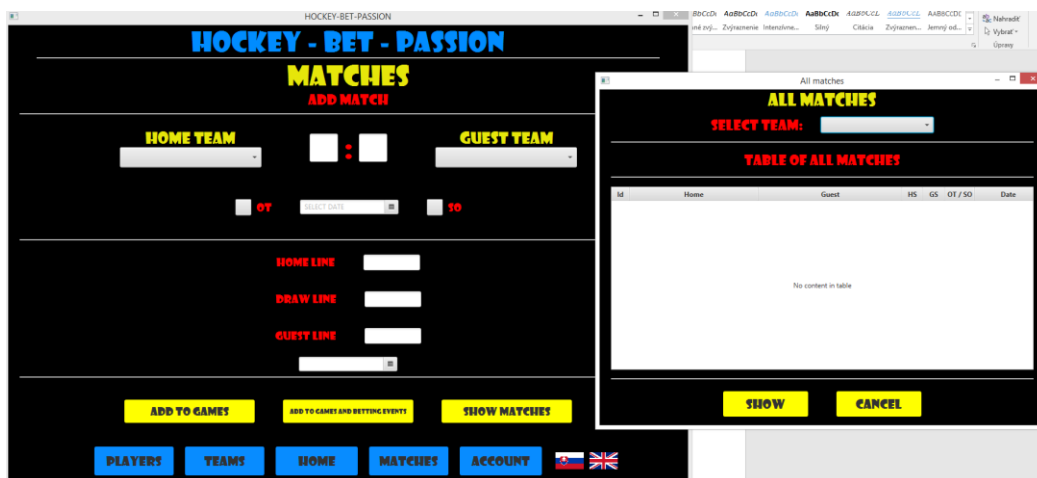
SHOW TEAMS

PLAYERS **TEAMS** **HOME** **MATCHES** **ACCOUNT**

Obrázok 6 Teams obrazovka

Obrazovka Teams slúži na zobrazenie tímov a taktiež pridávanie nového tímu do databázy.

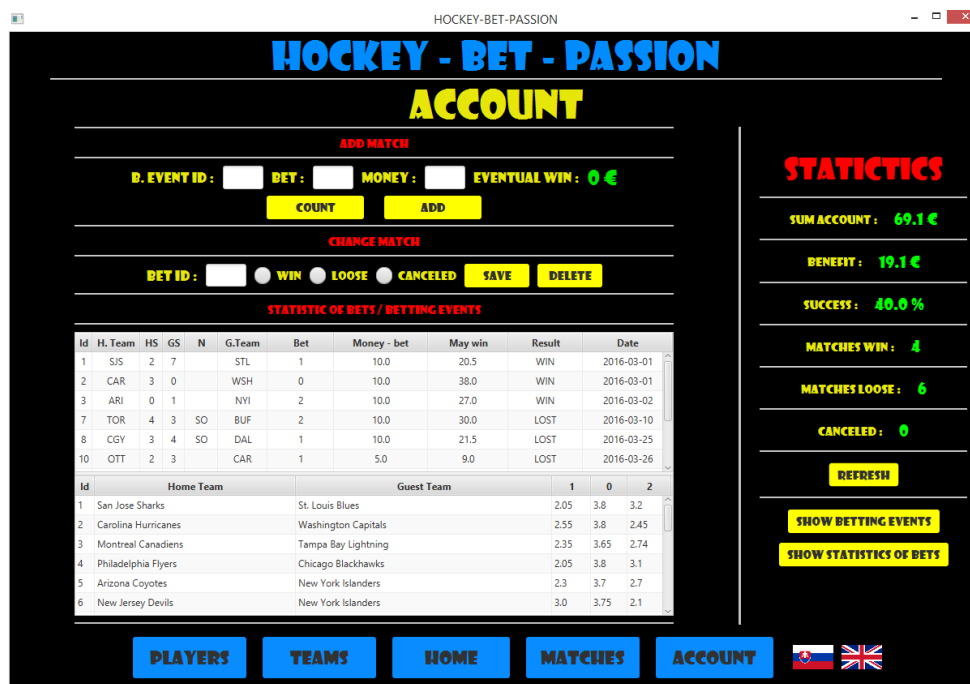
5.1.5 GUI Matches a All matches



Obrázok 7 Matches a All matches obrazovka

Obrazovka Matches slúži na pridávanie zápasov do databázy ale aj do tabuľky a stávkových udalostiach a All matches na zobrazenie zápasov.

5.1.6 GUI Account



Obrázok 8 Account obrazovka

Obrazovka Account slúži na správu stavených zápasov, ich štatistiku, zobrazenie aktuálnej sumy na účte, zisku, úspešnosti a iných štatistik.

5.2 Návrhové rozhodnutia

Každá z týchto obrazoviek je priradená k jednej triede v zdrojovom kóde. Ide o triedu, ktorá je pre každú obrazovku controllerom. V tejto triede sa vykonáva všetka potrebná funkcionálna týkajúca sa danej obrazovky, to znamená všetky metódy na riadenie tlačidiel, zobrazovanie dát do tabuliek atď.

Dáta, ktoré sa zobrazujú v jednotlivých tabuľkách majú taktiež priradenú svoju triedu. Dá sa teda povedať, že každá tabuľka je v zdrojovom kóde reprezentovaná triedou. Napríklad tabuľka tímov v obrazovke Teams. Dáta, ktoré sú do tejto tabuľky dané sú súčasťou ObservableListu, ktorý je typu triedy Teams. Táto trieda je iba akousi reprezentáciou dát, pretože slúži pre naplnenie tohto ObservableListu, ktorý je neskôr do tabuľky vložený.

Pripojenie databázy je riešené za pomoci technológie Hibernate v balíku Util a triede HibernateUtil, ktorá má ako zdroj dát konfiguračný súbor, v ktorom sa nachádzajú všetky potrebné informácie ako názov databázy, meno, heslo. Táto trieda vytvorí nový SessionFactory z konfiguračného súboru.

Po spustení aplikácie sa najprv otvorí nová Session a až potom nasleduje spustenie domovského okna. Po zatvorení okna – ukončení aplikácie sa táto Session zavrie.

K databáze pristupujem pomocou SQL query. To znamená pred každým zásahom do databázy si vytvorím SQL query, v ktorom sa nachádza potrebný SQL kód. Dáta, ktoré zobrazujem v tabuľkách si vytiahnem pomocou SQL query, potom ich transformujem a uložím ako objekty do listu. Prechádzam týmto listom až do konca, mením objekty na mapy a pomocou kľúča, ktorým je názov stĺpca vyberám konkrétne potrebné dáta.

Transakcie využívam pri všetkých INSERT, DELETE a UPDATE, ale hlavne pri práci s dvomi tabuľkami, kde ak sa vykoná len jeden z dvoch insertov, tak transakcia sa neukončí, naopak zmeny do databázy sa nevykonajú, budú vrátené.

6. Záver/Zhodnotenie

Myslím, že som naplnil potrebné požiadavky na projekt. Aplikácia má intuitívne grafické prostredie, scenáre sú realizované realisticky, poskytujú používateľovi funkcionálnu. Splnil som všetky potrebné scenáre, použil som transakcie a prácu s viacerými tabuľkami.

Zdroje

Dostupné na Internetu: <http://dbs-fiit.github.io/>

Dostupné na Internetu: <http://hibernate.org/>