



CHECK-MATES

Dokumentácia k inžinierskemu projektu

Tím č. 17: Check-mates

Vedúci tímu: Ing. Marián Šimko, PhD.

Členovia tímu: Bc. Igor Jurík, Bc. Matej Liskovec, Bc. Jaroslav Loebel, Bc. Martin Tamajka,
Bc. Peter Truchan, Bc. Ľubomír Vnenk

Akad. rok: 2014/2015

Obsah

Úvod	3
1 Globálne ciele ZS	4
1.1 Manažment tímu.....	4
1.2 Doména šachových turnajov	4
1.3 Technická stránka.....	4
1.4 Produkt	4
2 Celkový pohľad na systém.....	6
2.1 Architektúra súčasného stavu.....	7
2.2 Dátový model	7
2.3 Zoznam priložených elektronických dokumentov	8

Úvod

Tento dokument odráža priebežnú dokumentáciu k inžinierskemu dielu. Skladá sa z dvoch kapitol, ktoré sú zamerané na produkt. Prvá kapitola popisuje globálne ciele, ktoré si tím stanovil na zimný semester. Sú spísané v bodoch, tak ako sme si ich definovali počas semestra. Druhá kapitola stručne opisuje celkový náhľad na architektúru. Je tu opísaný súčasný stav architektúry produktu.

1 Globálne ciele ZS

Identifikovali sme nasledovné ciele, ktoré sme sa snažili zodpovedne naplniť.

1.1 Manažment tímu

- ☛ Odhadnúť výkonnosť tímu a jeho možnosti,
- ☛ spoznať schopnosti ostatných členov tímu,
- ☛ vytvoriť základ pre komunikáciu v tíme,
- ☛ vytvoriť si metodiky na jednotlivé činnosti a riadiť sa nimi,
- ☛ pravidelnými tímovými stretnutiami budovať tímového ducha a motiváciu,
- ☛ neustále zlepšovať a zefektívňovať komunikáciu v tíme,
- ☛ deliť úlohy v tíme rovnomerne a podľa schopností,
- ☛ neustále analyzovať riziká a minimalizovať ich dopad.

1.2 Doména šachových turnajov

- ☛ Osvojiť si procesy prebiehajúce v doméne
- ☛ Osvojiť si terminológiu z domény
- ☛ Nadviazať kontakty s potencionálnymi používateľmi z domény
- ☛ Preskúmať súčasný stav riešení v nami riešenej problematike
- ☛ Vyextrahovať potenciálnu cieľovú skupinu používateľov

1.3 Technická stránka

- ☛ Vybrať technológie vhodné na vývoj jednotlivých častí produktu
- ☛ Obozámieť sa (resp. pripomenúť si) vývoj v Ruby on Rails (táto technológia bola vybraná už vopred)
- ☛ Vytvoriť projekt a nasadiť naň verziovací systém
- ☛ úspešne nasadiť systém na produkčný server,
- ☛ implementovať nevyhnutné, nie vyčerpávajúce automatické testy,
- ☛ implementovať nevyhnutnú časť navrhnutého dátového modelu.

1.4 Produkt

- ☛ Určiť, aké funkcie by mal produkt poskytovať
- ☛ Určiť, z ktorých častí sa produkt bude skladať

1 Globálne ciele ZS

- Určiť MVP ako základ pre prvý prototyp
- implementovať tvorbu propozícií do funkčnej podoby,
- implementovať základ párovacieho Dutch systému so zreteľom na ďalšie rozšírenie

2 Celkový pohľad na systém

Frontend (klient) bude štandardne postavený na trojici HTML, CSS a Javascript, pričom na časti portálu, kde sa na stránke výraznejšie manipuluje s dátami použijeme SPA framework AngularJS pre zabezpečenie plynulého používateľského zážitku. Vzhľadom na to, že v súčasnosti sa rapídne zvyšuje počet používateľov, ktorí na internet prístupujú z mobilných zariadení, je nutné vytvoriť responzívny dizajn. Na to využijeme CSS framework Bootstrap od autorov Twitter. Zároveň, v neskorších fázach po implementácii základnej funkcionality, zvažujeme vývoj natívnych aplikácií pre najpoužívanejšie mobilné platformy (v prípade, že by to okolnosti vyžadovali).

Backendovú (serverovú) časť implementujeme v jazyku Ruby s použitím populárneho Ruby on Rails frameworku (RoR). Dôvodov prečo sme sa rozhodli pre tento jazyk je viacero:

- Väčšina členov tímu s ním majú skúsenosť
- ide o jazyk so stúpajúcou popularitou a s veľmi aktívnou komunitou,
- dynamická povaha interpretovaného jazyka je vhodná pre rapídny vývoj a rýchle
- prototypovanie - toto je dôležité kvôli tomu, že vývoj modulov nie je taký časovo náročný (a tým pádom drahý) a dovoľuje nám o niečo viac experimentovať
- RoR s pomocou Rspec je vhodný základ pre Test Driven Development a automatizované testy
- Komunita Ruby vytvorila a stále vytvára obrovské množstvo knižníc (nazývaných gems), ktoré zásadne uľahčujú vývoj zadarmo

Konceptuálne moduly produktu v našom prípade budú do istej miery odrážať používateľské príbehy veľkého rozsahu, takzvané epiky. Sú to logické celky, ktoré odrážajú veľké časti funkcionality významnej pre rôznych používateľov. Identifikovali sme šesť modulov:

- Vytváranie turnaja a propozícií
- Registrácia na turnaje a schvaľovanie hráčov
- Párovanie
- Evidencia hráčov a organizátorov
- Evidencia propozícií, turnajov a výsledkov
- Riadenie priebehu turnaja

2.1 Architektúra súčasného stavu

Základný architektonický vzor Model-View-Controller vyplýva z použitého Ruby on Rails frameworku. Pri jeho používaní kladieme dôraz na tenký controller - všetka biznis logika musí byť mimo neho a samotný controller slúži iba na to, aby získal dáta a spracoval ich do formy ľahko spracovateľnej pre view. Samotná biznis logika sa presúva do modelov a knižníc.

Pripravené je aj použitie frameworku AngularJS, ktorý je už v projekte integrovaný. Základná štruktúra a názvoslovie AngularJS inklinuje taktiež k Model-View-Controller vzoru, jeho použitie je však trochu odlišné. Model je viac menej iba rozhranie medzi serverom (kde sa Ruby on Rails controller mení na Restful službu) a logika a práca s dátami sa teda prirodzene presúva do controllera. Aj tu sa však budeme vyhýbať tučnému controlleru pomocou použitia direktív a služieb - konceptov ktoré v sú v AngularJS práve pre tento účel.

Aktuálna verzia prototypu sa nasadzuje na server pomocou gemu Capistrano, pričom dočasne je použitý Puma web server. Aplikácia v systéme beží ako služba.

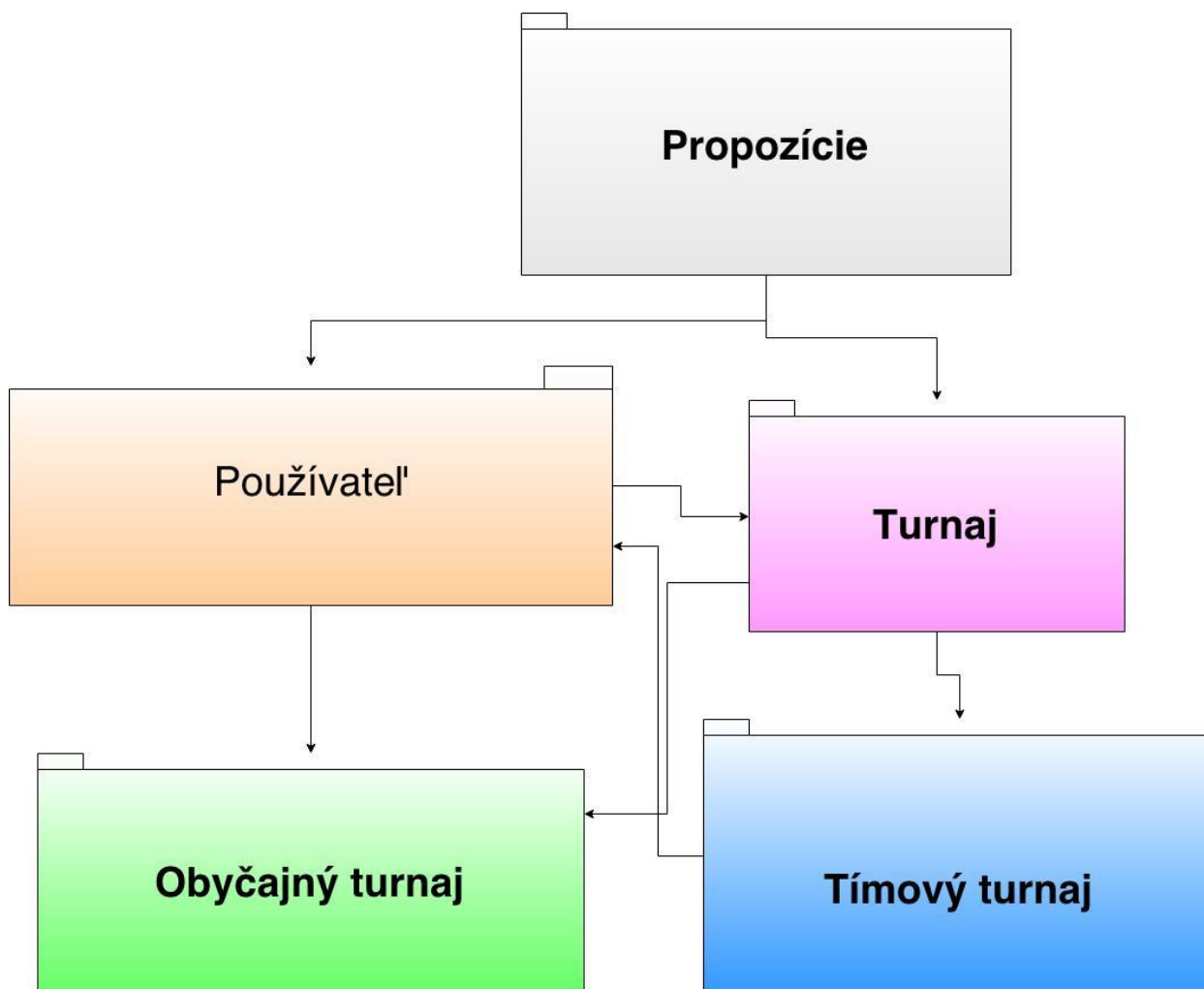
2.2 Dátový model

Dátový model je v prípade nášho projektu koncipovaný ako jedna relačná SQL databáza obsahujúca viacero tabuliek. Databázový stroj použitý na tento účel je PostgreSQL.

Na najvyššej úrovni sa možno na dátový model pozerat' z hľadiska toho, čo tá ktorá časť modeluje. Na tejto úrovni sme identifikovali tieto oblasti z biznis logiky začlenené do dátového modelu:

- propozície,
- turnaje,
- turnaje jednotlivcov,
- turnaje tímov,
- používatelia a hráči.

Tabuľky sú v E-R modeli zoskupené podľa týchto biznis entít, pričom každá z týchto entít je modelovaná väčším množstvom tabuliek. Na diagrame sú tieto zoskupenia znázornené za pomoci balíkov. V diagrame sú z dôvodov ochrany duševného vlastníctva zobrazené iba tieto balíky.



Obrázok 1 Vybraná časť dátového modelu

Tabuľky, ktoré sú spoločné pre viacero biznis entít, sú vyčlenené mimo balíkov. Zároveň existujú aj prepojenia medzi tabuľkami obsiahnutými v rôznych balíkoch. Takéto tabuľky neboli vyčlenené z balíkov, nakoľko s ostatnými tabuľkami v danom balíku tvoria jeden logický celok (napríklad tabuľka player).

2.3 Zoznam priložených elektronických dokumentov

Moduly_systému.pdf – obsahuje dokumentáciu k jednotlivým modulom systému