

Slovenská technická univerzita
Fakulta informatiky a informačných technológií

Odovzdanie míľnik 1.

Tímový projekt MIAOW

Predmet:	Tímový projekt I.
Členovia tímu:	Bc. Martin Gazdík Bc. Timotej Gurka Bc. Marek Ďaďo Bc. Tamara Jantová Bc. Matej Blanárik Bc. Martin Bombala Bc. Nikola Stanimirovič
Vedúci práce:	Mgr. Monika Kováčová, PhD.
Akademický rok:	2020/2021

Slovenská technická univerzita
Fakulta informatiky a informačných technológií

Riadenie projektu

Tímový projekt MIAOW

Predmet:	Tímový projekt I.
Členovia tímu:	Bc. Martin Gazdík Bc. Timotej Gurka Bc. Marek Ďaďo Bc. Tamara Jantová Bc. Matej Blanárik Bc. Martin Bombala Bc. Nikola Stanimirovič
Vedúci práce:	Mgr. Monika Kováčová, PhD.
Akademický rok:	2020/2021

Obsah

1	Úvod	3
2	Roly členov tímu	3
3	Aplikácie manažmentov	5
3.1	Manažment iterácií	5
3.2	Manažment riadenia úloh	5
3.3	Manažment prehliadok kódu	5
3.4	Manažment verziovania kódu	5
3.5	Manažment dokumentovania	5
4	Sumarizácia šprintov	7
4.1	Šprint 1 - Singapura	7
4.2	Šprint 2 - Egyptian Mau	9
4.3	Šprint 3 - Snowshoe	10
5	Motivačný dokument	14
5.1	Predstavenie tímu	14
5.2	Technologické znalosti tímu	14
5.3	Motivácia - Téma č. 18 - FIFé International Cat Show [MIAOW]	15
5.4	Motivácia - Téma č. 6 - Transformácia priestorov na bezpečné a inteligentné miesta na prácu [SmartSpace]	16
5.5	Motivácia - Téma č. 16 – Inteligentný informačný systém zameraný na projektový manažment a automatizáciu procesu verejného obstarávania [iPP]	17
5.6	Príloha A - Zoradenie tém podľa priority	18
5.7	Príloha B - Rozvrh tímu	19
6	Metodiky	20
6.1	Definície dokončenia	20
6.2	Metodika popisu úloh	21
6.2.1	User Story	21
6.2.2	Bug	22
6.2.3	Epic	23
6.2.4	Feature	23
6.2.5	Impediment	24
6.2.6	Backlog item	24
6.2.7	Task	24
6.3	Metodika tímovej komunikácie	24
6.3.1	Komunikačné kanály	24
6.3.2	Emailová komunikácia	24
6.3.3	Komunikácia v kanáloch	25
6.3.4	Stretnutia	25
6.4	Metodika verziovania práce - kódu	25
6.4.1	Postup verziovania	25
6.5	Metodika testovania produktu	26
6.6	Metodika riadenia dokumentácie	26
6.7	Metodika Revízie zmien kódu	27
6.7.1	Autor	27

6.7.2	Recenzent	28
6.8	Metodika iterácií	28
6.8.1	1. stretnutie - zmena šprintu	28
6.8.2	2. stretnutie - práca na user stories	28
6.8.3	3. stretnutie - práca na user stories	28
6.8.4	4. stretnutie - retrospektíva	29
6.9	Metodika riadenia úloh	29
6.10	Životný proces USER STORY	29
6.10.1	Backlog	29
6.10.2	Ready	29
6.10.3	Develop	29
6.10.4	Blocked	29
6.10.5	Review	29
6.10.6	Test	29
6.10.7	Done	29

1 Úvod

Tento dokument sa zaoberá organizáciou tímu. Poskytuje informácie k riadeniu projektu a tímu. Okrem iného rieši pracovné procesy.

V našom tímovom projekte sa venujeme projektu MIAOW. Tento projekt je novo založeným projektom a zaoberá sa analýzou, návrhom, implementáciou, nasadením a testovaním webovej aplikácie, ktorá sprevádza celý proces manažmentu výstavy mačiek.

Hlavným cieľom tímového projektu je naučiť sa pracovať vo väčšej skupine ľudí (tíme) a správne sa riadiť. Zároveň je naším cieľom naučiť sa pracovať s nástrojmi na podporu manažmentu tímov.

Vedúcim projektu je Mgr. Monika Kováčová, PhD., ktorá nás má usmerňovať pri práci a pomáhať nám pri tvorbe úloh alebo problémoch ktoré sa vyskytnú počas implementácie. Vedúca tímu zároveň plní aj úlohu vlastníka produktu, čo znamená, že nám zadáva funkčné požiadavky na daný produkt.

Dokument tiež obsahuje roly členov tímu a aplikácie manažmentov. Ďalej pokračuje opisom šprintov a exportom jednotlivých úloh, ktoré boli analyzované alebo implementované v danom šprinte. Obsahuje aj metodiky, ktoré sú počas vývoja systému používané.

2 Roly členov tímu

Mgr. Monika Kováčová, PhD.

- Vedúca tímu
- Vlastník produktu
- Definuje víziu projektu

Bc. Martin Gazdík

- Scrum Master
- Solution Architect
- Zabezpečuje infraštruktúru

Bc. Timotej Gurka

- Manažér kvality
- Správca webového sídla
- UX/UI Návrhár

Bc. Marek Ďaďo

- Frontend Developer
- Manažér kvality kódu

Bc. Tamara Jantová

- Frontend Developerka
- Testerka

Bc. Matej Blanárik

- Backend Developer
- Zodpovedný za UX/UI

Bc. Martin Bombala

- Backend Developer
- Solution Architect

Bc. Nikola Stanimirovič

- Manažér vývoja
- Developer

Podiel práce na dokumentácii k riadeniu projektu	
Názov sekcie	Autor
Úvod	Martin Gazdík
Roly členov tímu	Martin Gazdík
Aplikácie manažmentov	Martin Gazdík
Sumarizácia šprintov	Timotej Gurka
Motivačný dokument	Tím
Metodiky	Martin Gazdík, Timotej Gurka
Webové sídlo	Martin Gazdík, Timotej Gurka

3 Aplikácie manažmentov

3.1 Manažment iterácií

Práca na tímovom projekte je rozdelená do šprintov - dvojtýždňových iterácií. Šprinty realizujeme podľa agilnej metodiky Scrum. Každý šprint začína plánovaním, pokračuje priebežnou prácou na vybraných úlohách a následné vyhodnotenie - product review a retrospektíva na zlepšenie procesov v tíme. Dôležitou súčasťou Scrumu sú pravidelné stretnutia všetkých členov tímu, ktoré realizujeme každú stredu s product ownerom (s vedúcou tímu) a každý pondelok bez nej.

- Metodika riadenia iterácií

3.2 Manažment riadenia úloh

Na manažment riadenia vývoja produktu ako aj riadenia tímu používame nástroj Azure DevOps. Ten nám umožňuje udržiavať zoznam používateľských príbehov (ďalej len ako US - User Story), ktoré predstavujú pridanú funkcionálnosť pre vlastníka produktu. Používateľské príbehy môžu obsahovať svoje podúlohy - tzv. tasky. Ďalej môžu byť US priradené do jednotlivých funkčných požiadaviek - features a epicov. Nástroj nám zároveň pomáha plánovať, riadiť a vyhodnocovať vývoj produktu ako aj prácu v tíme.

- Metodika riadenia úloh
- Ako popisovať položky v Azure DevOps

3.3 Manažment prehliadok kódu

Prehliadky kódu vykonávame v samostatnom module GIT Azure DevOps v časti Pull Requests. Každá nová vetva, ktorá sa má zlúčiť do develop vetvy (pričom vetva develop je hlavná vývojová vetva), musí prejsť cez prehliadku kódu inými, najmenej dvoma členmi tímu. Pri prehliadke kódu je dôležité dbať na kvalitu a správnosť kódu, ako aj na to, či sú zmeny zapracované v dokumentácii. Pri každej prehliadke sa autor žiadosti ako aj schvalovateľ riadia príslušnou metodikou.

- Ako vykonať code review

3.4 Manažment verziovania kódu

Pri vývoji je používaný verziovací systém git a projekt je uložený vo vzdialenom repozitári v systéme Azure DevOps. Spôsob verziovania je bližšie opísaný v metodike na verziovanie kódu a prácu s gitom, ktorá sa nachádza v prílohe tohto dokumentu. V tejto metodike sú tiež popísané jednotlivé typy vetiev, ktoré používame, ale aj to ako ich pomenovať a písať súhrnnú správu potvrdenia - commitu.

- Ako verziovať kód a pracovať s gitom

3.5 Manažment dokumentovania

Dokumentáciu momentálne udržiavame v dvoch repozitároch - oddelene pre backendovú ako aj frontendovú časť projektu, nakoľko ide o dva celkom rozličné prvky. Dbáme na to, aby bola dokumentácia pravidelne aktualizovaná. Používateľské príručky pripravujeme priebežne. Nakoľko aktuálne projekt nie je ešte v stave publikácie pre používateľov, predpríprava príručiek aktuálne nemá zmysel. Pri písaní dokumentácie sa riadime príslušnou metodikou.

- Ako písať dokumentáciu

4 Sumarizácia šprintov

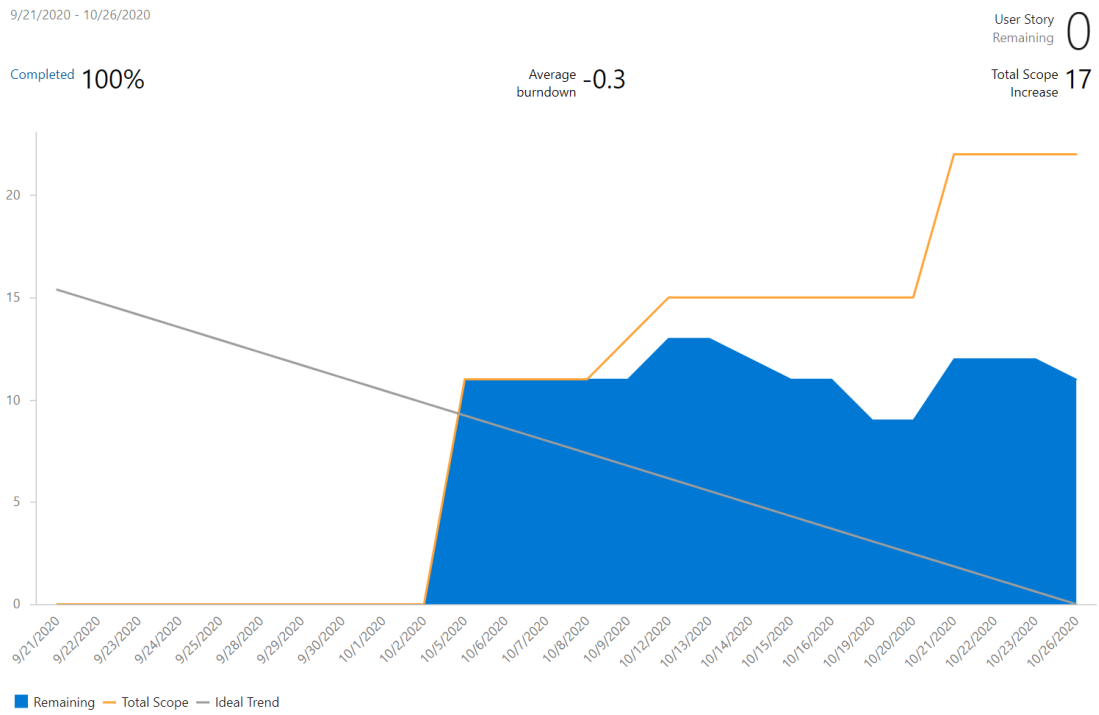
4.1 Šprint 1 - Singapura

Trvanie šprintu: 12.10.-26.10. 2020 Doručené story pointy: 66.5/74.5

V nultom a prvom šprinte sme sa zamerali na formovanie tímu a zadania projektu. Vytvorili sme tímový web, snažili sa na hrubo pochopiť doménu, v ktorej sa pohybuje zadanie - výstava mačiek na svetovej úrovni. Definovali sme si metodiky, ktoré budú smerovať našu ďalšiu spoluprácu, zoznámili sme sa so scrumom ako takým - na začiatku plánovania sme vytvorili back log, ktorý sme následne konzultovali s vedúcou tímu.

ID	Work Item Type	Title	Assigned To	State	Tags	Story Points
1	User Story	eslint Inicializácia	Marek Ďaďo <xdado@stuba.sk>	Closed	Frontend	2.5
2	User Story	Výber template	Tamara Jantová <xjantova@stuba.sk>	Closed	Frontend	2
3	User Story	Backend inicializácia	Martin Bombala <xbombalam@stuba.sk>	Closed	Backend	9.5
4	User Story	Backend Swagger	Martin Bombala <xbombalam@stuba.sk>	Closed	Backend	5
5	User Story	Setup Devops + repozitár	Martin Gazdik <xgazdikm2@stuba.sk>	Closed	InfraWork	3
6	User Story	Spísanie OneNote pre konvencie, pravidlá, etc.		Closed	Management	2.5
7	User Story	Inicializácia projektu Frontend	Marek Ďaďo <xdado@stuba.sk>	Closed	Frontend	14
8	User Story	Event Controller	Martin Bombala <xbombalam@stuba.sk>	Closed	Backend	2
9	User Story	Event grid	Tamara Jantová <xjantova@stuba.sk>	Closed	Frontend	7
10	User Story	WebSite	Martin Gazdik <xgazdikm2@stuba.sk>	Closed	InfraWork; Management	8
11	User Story	Pripraviť zoznam US	Timotej Gurka <xgurka@stuba.sk>	Closed	Management	2.5
12	User Story	Priprava VPN	Martin Gazdik <xgazdikm2@stuba.sk>	Closed	InfraWork	2.5
13	User Story	Priprava virtualky	Martin Gazdik <xgazdikm2@stuba.sk>	Closed	InfraWork	4
14	User Story	Upload file grid	Marek Ďaďo <xdado@stuba.sk>	Active	Frontend	8
15	User Story	Nastavenie Devops Git		Closed		2
16	Bug	Nejde git userom, ktorí nie su Basic (stakeholders)	Martin Gazdik <xgazdikm2@stuba.sk>	Closed	Urgent	
17	Task	Public Frontend inicializacia	Marek Ďaďo <xdado@stuba.sk>	Closed		
18	Task	Private frontend inicializacia	Tamara Jantová <xjantova@stuba.sk>	New		

Obr. 1: Export úloh z TFS pre šprint 1



Obr. 2: Burndown chart pre šprint 1

START	STOP	CONTINUE
Keď sľúbim, že budem dostupný, tak naozaj som dostupný - byť dochvilný	Niekedy sa až moc veľa rozprávame o nepodstatných veciach	Dopodrobna rozoberať danú problematiku
Viac sa baviť o tom ako má vyzeráť dátový model	Komplikovať veci	Rozprávať sa o možných riešeniach
Viac komunikovať, keď je nejaký problém	Nech všetko nestojí iba na jednom človeku	Komunikovať
Vpisovať aktualizácie do user stories	Dlho nám trvá ohodnocovanie taskov	Dokončiť web
Byť viac motivovaní		Vedieť vyjadriť svoj názor aj na calloch – možno aj za cenu, že skočím niekomu do reči
Tímové pivo (keď sa trochu ukludní covid)		

Obr. 3: Retrospektíva pre šprint 1

4.2 Šprint 2 - Egyptian Mau

Trvanie šprintu: 26.10.-09.11. 2020 Doručené story pointy: 39/72 Šprint začal tvorbou dema, aby sme boli schopní vhodne vybrať technológie, v ktorých sa bude na projekte pracovať. Vďaka vytvoreniu drobných projektov sme boli schopní zistiť, ktoré technológie budú kompatibilné a s ktorými sa nám dobre pracuje. Nakoľko je tímový projekt prvenstvom nielen pre nás ako tím, ale aj pre našu pani vedúcu, bolo na začiatku ťažké definovať zadanie a rozsah implementácie. V tomto pozostával problém tohoto šprintu, ktorý okrem iného odhalil aj množstvo vznikajúcich problémov v tíme - išlo najmä o nejasné rozdelenie kompetencií, chýbajúce metodiky, ktoré by jasne definovali, kto a kedy má osloviť kolegu, ako aj problematiku členenie tímu medzi backend, frontend a management, pričom komunikácia v segmente fungovala dobre, avšak komunikácia napriek segmentami častokrát zlyhávala. Napriek týmto problémom sa nám nakoniec podarilo ujasniť si pravidlá a metodiky, rozhodnúť sa pre technológie ako aj definovať kompetencie v tíme.

ID	Work Item Type	Title	Assigned To	State	Tags	Story Points
20	Task	Navigation bar	Marek Dado <xdado@stuba.sk>	Closed	Frontend	
21	Task	Refactoring usporiadania API a stromovej struktury	Martin Bombala <xbombalam@stuba.sk>	New	Backend	
22	User Story	EMS list		New		
24	Task	preštudovať štruktúru xls dokumentu		New		
25	Task	vymyslieť ako importovať z xls do MYSQL		New		
26	Task	entita v databáze		New		
27	Task	reálne uskutočniť import		New		
28	Task	skontrolovať náhodným porovnaním, či je naimportované skutočne všetko		New		
29	Task	urobiť vo front ende formulár na zadávanie položiek, ktoré nie sú v štandardnom ems liste (priklad, nc		New		
30	Task	urobiť formulár na možnosť zadania slovenskej mutácie pre vybrané farby		New		
31	User Story	Entity v databáze		New		
32	Task	entita plemiená		New		
33	Task	entita súťažné triedy (názov + popis)		New		
34	Task	entita - platby		New		
35	Task	entita - posudzovateľ		New		
36	Task	entita mačka		New		
37	User Story	Prihláška		New		
38	Task	základný formulár na webe		New		
39	Task	formulár na báze prihláška - editovateľná verzia		New		
40	Task	kontrola group		New		
41	Task	kontrola súťažných tried		New		
42	Task	formulár pre prihlášku vrhu		New		
43	Task	formulár pre control class (13)		New		
44	Task	voľiteľné súťaže		New		
45	Task	križová kontrola zaslaných prihlášok + sumarizácia prihlášok		New		
46	User Story	Testovacia výstava		New		
47	Task	vytvoríť testovací event		New		
48	Task	nasypať do neho údaje o mačkách z výstavy Marec 2019		New		
49	Task	prepojiť na event EMS tabuľky, grupy, classy... a ostatné obslužné tabuľky		New		
50	Task	skontrolovať, či funkcionality z US 37 fungujú na reálnych dátach		New		
51	User Story	Printové dokumenty		New		
52	Task	technologia generovania výstupov		New		
53	Task	report o stave mačiek pred výstavou		New		
54	Task	Zoznam mačiek pre veterinára		New		
55	Task	Zoznam mačiek pre platby		New		
56	Task	čistý dokument faktúra		New		
57	Task	vyplnený dokument faktúra		New		
58	Task	Zoznam tabulek na kletky		New		
59	Task	čistý dokument posudzovací formulár		New		
60	Task	vyplnený dokument posudzovací formulár pre každú súťaž samostatne	Monika Kovacova <monika.kovacova@stuba.sk>	New		
61	Task	Zoznam mačiek pritomných na výstave / zoznam absentov		New		
62	User Story	Printové dokumenty po vytvorení katalógu		New		
63	User Story	Vytvorenie katalógu		New		
64	Task	katalóg pre kategóriu 1		New		
65	Task	katalóg pre kategóriu 2		New		
66	Task	katalóg pre kategóriu 3		New		
67	Task	katalóg pre kategóriu 4		New		
68	Task	katalóg pre kategóriu House cat		New		
69	Task	Zoznam vystavovateľov		New		
70	Task	Dotatky ku katalógu		New		
71	Task	Vytvoríť grafický návrh ako máť usporiadané údaje o mačke	Martin Gazdik <kgazdikm2@stuba.sk>	New		
72	Task	Postprocessing katalogu ? automatizácia		New		
75	Task	Ukladanie hesla mimo plaintextovej formy		New		
76	Task	JWT Tokenizacia		New		
77	Task	Implementacia volanie autorizacie, vyprsanie session		New		
78	Task	Uprava routa na role používateľov		New		
79	User Story	Setup DEV prostredia	Martin Gazdik <kgazdikm2@stuba.sk>	New	InfraWork; Management	10
80	User Story	Spracovanie EMS listu	Nikola Stanimirovic <xstanimirovic@stuba.sk>	Closed	Backend	19
81	User Story	Datagrid EMS	Tamara Jantová <xjantova@stuba.sk>	Active	Frontend	7
82	User Story	OData EMS	Martin Bombala <xbombalam@stuba.sk>	Closed	Backend	8
83	User Story	Registration admin grid	Tamara Jantová <xjantova@stuba.sk>	New	Frontend	7
84	User Story	Registration customer grid	Marek Dado <xdado@stuba.sk>	Active	Frontend	16
86	User Story	Backend refactoring	Martin Bombala <xbombalam@stuba.sk>	Closed	Backend	5

Obr. 4: Export úloh z TFS pre šprint 2

Vzhľadom na riešenie dôležitých problémov počas konca šprintu 2, sme sa rozhodli čas venovať efektívnejšiemu a podrobnejšiemu plánovaniu, ako aj dopracovaniu a nastaveniu kompetencií v tíme. Práve z týchto dôvodov sme počas šprintu 2 nepraktizovali aktivitu Start+Stop+Continue. Z uvedeného dôvodu nepridávame jej výstup.

4.3 Šprint 3 - Snowshoe

Trvanie šprintu: 09.11.-23.11. 2019 Doručené story pointy: 79,5/121

Na začiatku šprintu sa zvolal meeting celého tímu nad rámec štandardných stretnutí. Dohodli sme sa na prepracovaní dátového modelu tak, aby vyhovoval členom tímu naprieč segmenty backend i frontend, formovali sa presné ciele šprintu

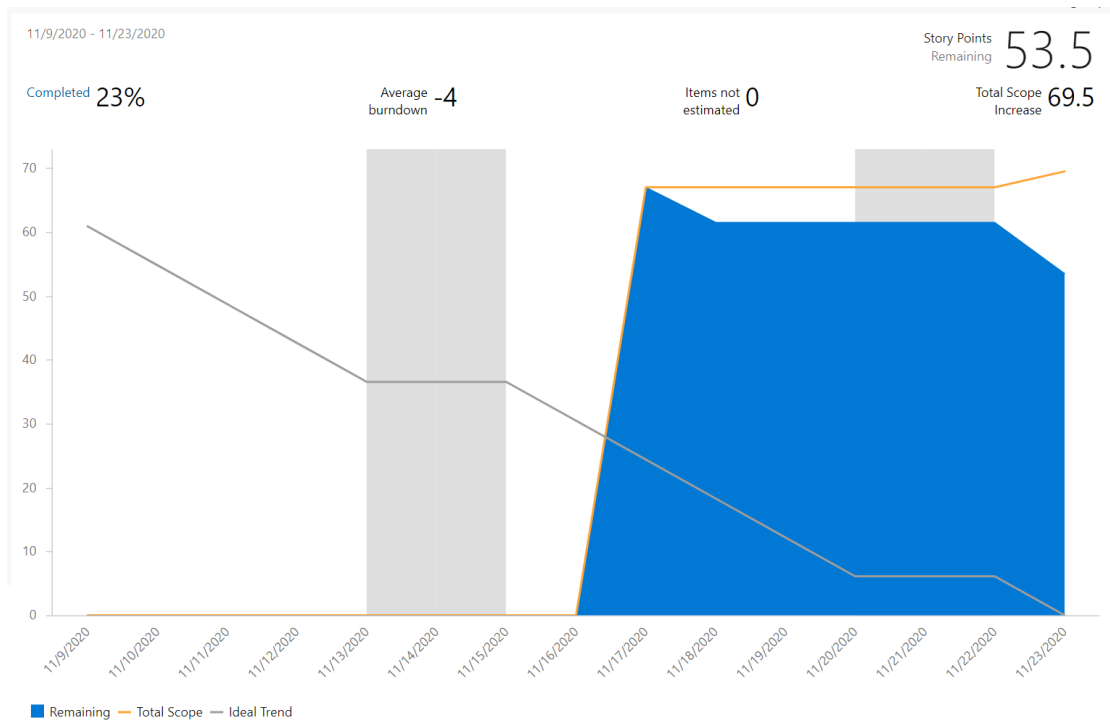
- prepracovanie dátového modelu
- import demo dát
- zapracovanie kontrolérov pre všetky navrhnuté entity

- nahrávanie súboru EMS listu z frontendu - používateľskej aplikácie
- automatické spracovanie EMS listu z nahratého xlsx súboru do databázy - kľúčová funkcionálna šprintu

Následne konštatujeme, že sa nám podarilo splniť všetky ciele tohto šprintu.

ID	Work Item Type	Title	Assigned To	State	Tags	Story Points
87	Task	Date-picker	Marek Ďaďo <xdado@stuba.sk>	Closed		
88	Task	Pagination	Tamara Jantová <xjantova@stuba.sk>	Closed		
89	Task	upload and store EMS file		New		
90	Task	User registration form	Tamara Jantová <xjantova@stuba.sk>	Closed		
91	Task	Parents form	Marek Ďaďo <xdado@stuba.sk>	Closed		
92	Task	Additional info form	Marek Ďaďo <xdado@stuba.sk>	Closed		
93	Task	Prihlaska na vystavu		New		
94	Task	Zoznam požiadaviek pre rozdelenie kliek		New		
95	Task	Karticky na kliek		New		
96	Task	Vseobecny zoznam maciek		New		
97	Task	Zoznam na obalky		New		
98	Task	nalepky pre vystavovateľa		New		
99	Bug	Cors headers not allowed	Nikola Stanimirovic <xstanimirovic@stuba.sk>	Closed		
100	User Story	DB Entities creation	Matej Blarík <xblarikm1@stuba.sk>	Closed		4
101	User Story	Demo data SQL insert	Matej Blarík <xblarikm1@stuba.sk>	Closed		4
102	User Story	Controllers creation	Martin Bombala <xbombalam@stuba.sk>	Closed		8
103	User Story	Uprava Spracovania EMS listu	Nikola Stanimirovic <xstanimirovic@stuba.sk>	Active		5.5
104	User Story	Backend tvorba PDF		New	Backend	19
105	Task	DevExtreme Component for EMS Data		Closed		
107	User Story	Projektová dokumentácia	Timotej Gurka <xgurka@stuba.sk>	Active		15
108	User Story	Spracovanie Big Picture		New		15
109	User Story	Dokumentácia - online		New		16
110	User Story	Uprava ciselnikov		New	Frontend	6
111	User Story	JS - Export gridu do PDF		New	Frontend	6
112	User Story	Backend - príprava katalogu		New	Backend	22.5

Obr. 5: Export úloh z TFS pre šprint 2



Obr. 6: Burndown chart pre sprint 3

START	STOP	CONTINUE
Dávať si vzájomne vedieť, keď sa niečo dorobí medzi FE a BE, aby sme vedeli na čom môžeme pokračovať, čo môžeme dokončiť	Útoky medzi FE a BE - blokuje sa často a potom vznikajú také veci, že „čakáme na BE“. Šprint potom vyzerá tak, že BE robí prvú polku šprintu a druhú polku robí FE.	Lepšia komunikácia, v ktorej treba pokračovať
Písať lepšie zápisy zo stretnutí	Mať dlhé <u>meetingy</u>	Byť k sebe úprimní
Byrokratickým veciam sa venovať priebežne		Pokračovať v robení viac jednoduchých vecí, ako 1 komplexnú
Zlepšiť komunikáciu medzi BE a FE		Komunikovať
Dávať si na začiatku šprintu jasné ciele		Mať dobrú náladu
Začať si dávať <u>deadliny</u> na funkcionálne prvky		Navrhovať pred implementáciou - ako aj v prípade dátového modelu
Viac komunikovať s vedúcou		
Oveľa zvýšiť intenzitu komunikácie		
Treba zvýšiť našu efektivitu		
Robiť <u>rebase</u> , zdefinovať si konvencie na Gite, mať tam lineárnu históriu		

Obr. 7: Retrospektíva pre šprint 3

5 Motivačný dokument

5.1 Predstavenie tímu

Nakoľko sme v tomto zoskupení mali spoločné predmety, prednášky, cvičenia a často aj rôzne zadania, môžeme konštatovať, že sme zohraný tím, poznáme sa, vieme čo očakávať od jednotlivcov a teda aj od tímu ako celku. Ako tím sa prezentujeme najmä týmito vlastnosťami:

- Súťaživosť – usilujeme sa o dosiahnutie toho, čo by sa ostatným nemuselo podariť
- Agilita – dokážeme sa pomerne rýchlo prispôbiť zmenám v zadaní, rýchlo sa orientujeme v rôznych témach
- „Good enough is not enough” – výsledok musí byť viac ako len dobrý
- Multifunkčnosť – znalosťami a skúsenosťami s rôznymi technológiami sa líšime – koniec koncov to však považujeme za výhodu
- Komunikatívnosť – pretože to považujeme za základ
- Schopnosť preskúmať problematiku a vhodne navrhnúť architektúru ako aj UX/UI
- Tím zameraný najmä na vývoj webových aplikácií pôsobiaci I vo svete IoT
- Bohaté pracovné skúsenosti naprieč tímom = schopnosť spolupráce v tíme

5.2 Technologické znalosti tímu

Nižšie popisujeme všetky technológie, nástroje, programovacie jazyky a techniky vývoja, s ktorými si trúfame pracovať a máme s nimi skúsenosti:

- Preferujeme Scrum metódu vývoja
- Manažment/verzia/problem management nástroje – Git, DevOps, Jira, Jazz, BitBucket
- Databázové technológie – MSSQL, MySQL, Postgres, MongoDB
- Testovanie – Selenium, Cypress
- Backend – C#, Java, Python, node.js, .NET
- Frontend – JavaScript, Angular, React, Typescript, Devextreme, Bootstrap
- Vývoj mobilných aplikácií – native - android, react native, flutter
- Docker, Kubernetes, NGINX, IIS, Entity Framework, linux, bash, perl, basic
- Skúsenosti v oblasti IoT a energetike
- Skúsenosti s návrhmi UX/UI

5.3 Motivácia - Téma č. 18 - FIFé International Cat Show [MIAOW]

Ponúkaná téma nás zaujala najmä z dôvodu, že ide o projekt, u ktorého vidíme možnosť nabrať nové skúsenosti a poznatky s ich následným využívaním aj po jeho skončení. Azda najväčším plusom je fakt, že výsledný produkt sa bude reálne využívať a nejde iba o zadanie v teoretickej rovine. Veríme, že naše spôsoby vývoja a komunikácie dokážu zabezpečiť, že z vytvárania adekvátne kvalitného a atraktívneho informačného systému budú profitovať obe strany. Medzinárodná výstava mačiek znie ako zaujímavá, avšak nie najbežnejšia téma, a práve o to väčší záujem v nás vzbudzuje. Z jej opisu sme nadobudli pocit, že na komunikácii bude záležať nielen nám, ale aj zadávateľovi, čo radi uvítame.

O túto tému máme prioritný záujem najmä preto, lebo veríme, že naše skúsenosti z oblasti vývoja webových aplikácií dokážu zabezpečiť kvalitný výsledný produkt, ktorý bude spĺňať všetky požiadavky zadávateľa a po vzájomnej dohode dokážeme pridať aj niečo z nášho „repertoára“.

Členovia nášho tímu sa s uvádzanými technológiami už v minulosti stretli, respektíve väčšia časť sa nimi zaoberá aj v práci, či mimoškolských aktivitách. Ako sme spomínali, všetci sa navzájom dobre poznáme a vieme tak, čo môžeme jeden od druhého očakávať. Nakoľko sme „odborový mix“, každý z nás dokáže do tímu priniesť niečo iné, v čom vidíme veľkú výhodu. S výberom odporúčaných technológií silno sympatizujeme a už teraz máme viaceré predstavy, ako projekt zrealizovať, aj keď máme samozrejme viacero otázok, ktorými by sme chceli zadávateľa zahrnúť.

Príjemným bonusom je, že možným vyústením projektu je z dlhodobého hľadiska aj vlastný startup, ale predovšetkým, že by nám bola poskytnutá šanca na reálne overenie funkčnosti produktu v praxi.

Sumár:

- Tím zameraný na vývoj webových aplikácií
- Praktické skúsenosti s komunikáciou so zadávateľom
- Skúsenosti z viacerých odborov a oblastí
- Skúsenosti s vývojom podobných projektov
- Nadšenie pre uvádzané technológie
- Tím, ktorý sa navzájom dobre pozná a vie, čo možno od jednotlivých členov očakávať

5.4 Motivácia - Téma č. 6 - Transformácia priestorov na bezpečné a inteligentné miesta na prácu [SmartSpace]

Táto téma nás oslovila, pretože je aktuálna a znalosti na ňu potrebné sa prekrývajú sa s našimi mimoškolskými skúsenosťami a záujmami. Páči sa nám, že cieľom ponúkanej témy je modulárne riešenie, ktoré môže byť v blízkej budúcnosti využívané ako štandard pre kancelárske priestory.

Čo sa týka IoT a technológií s ním spojenými, s tými máme mnoho skúseností primárne z praxe – z reálneho prostredia a implementácií u zákazníkov. Máme bohaté skúsenosti s návrhom a výrobou či úpravou hardvéru, návrhu a inštalácie infraštruktúry až po odovzdanie kompletného projektu. Čo sa týka len technológií, uvádzame aspoň krátky výber z nich :

- NB-IoT
- LoRaWAN
- Sigfox
- MODBUS TCP
- MODBUS RTU (RS485)
- OneWire
- SCADA
- MQTT
- IQRf
- Mervis

Boli by sme radi, ak by sme dostali možnosť zapojiť naše skúsenosti a pomôcť s implementáciou tejto témy.

Sumár:

- Máme skúsenosti s tvorbou web aplikácií
- Máme bohaté skúsenosti s IoT
- 6/7 členov tímu sú bakalári z odboru Internetové technológie

5.5 Motivácia - Téma č. 16 – Inteligentný informačný systém zameraný na projektový manažment a automatizáciu procesu verejného obstarávania [iPP]

Rozhodli sme umiestniť túto tému do TOP 3, nakoľko ide opäť o zaujímavý a ambiciózny projekt, ktorý je zameraný na webové technológie. Keďže sme tím, ktorý sa zameriava najmä na webové aplikácie, portály, systémy, či riešenia, myslíme si, že táto téma by nám sadla a vedeli by sme splniť očakávania zadávateľa.

Máme radi procesy, ich návrh a vizualizáciu, ako aj zobrazovanie čísel v zrozumiteľnej podobe a ďalšiu prácu s nimi. Komunikácia so zadávateľom je jedna z našich silnejších stránok, pretože to viacerí z nás zažili aj v mimoškolskom prostredí. User design je pre nás dôležitý, pretože veci by mali fungovať, no mali by sa aj ľahko ovládať.

Sumár:

- Skúsenosť s tvorbou reálnych projektov u zákazníkov
- Praktické skúsenosti s komunikáciou so zadávateľom
- Skúsenosti z viacerých odborov a oblastí
- Skúsenosti s vývojom podobných projektov
- Tím, ktorý sa navzájom dobre pozná a vie, čo možno od jednotlivých členov očakávať

5.6 Príloha A - Zoradenie tém podľa priority

Poradie	Číslo témy	Názov témy
1	18	FIFé International Cat Show [MIAOW]
2	6	Transformácia priestorov na bezpečné a inteligentné miesta na prácu [SmartSpace]
3	16	Inteligentný informačný systém zameraný na projektový manažment a automatizáciu procesu verejného obstarávania [iPP]
4	5	Educational and coworking driven orchestration portal [EDUCO]
5	9	Vzdialené monitorovanie zdravotného stavu človeka pomocou E-Health
6	12	Safety panel a spätná analýza údajov pre vývoj autonómneho vozidla [avPANEL]
7	2	Webové IDE pre ASIC [ASICDE]
8	4	Educational Content Engineering Hub - Databáza otázok, odpovedí, úloh a riešení [ECEH-DU]
9	10	Game-chain: Ako bezpečne vymieňať herné účty
10	13	Korekcia dynamických vlastností virtuálnych modelov komponentov vozidiel [CarComponents]
11	14	Platforma pre sledovanie dodávateľského reťazca s využitím technológie blockchain [S-Chain]
12	1	Blockchain platobné brány [BlockPay]

5.7 Príloha B - Rozvrh tímu

PRIORITY KEY:

Cvičenie	TeamMeeting	Prednáška
----------	-------------	-----------

ČAS	PONDELOK	UTOROK	STREDA	ŠTVRTOK	PIATOK
7:00					
7:30					
8:00		Ďaďo/Gurka	Bombala/Gazdík/Blanárík	Gazdík/Blanárík	Prednášky
8:30		Ďaďo/Gurka	Bombala/Gazdík/Blanárík	Gazdík/Blanárík	Prednášky
9:00	TeamMeeting	Ďaďo/Gurka	Bombala/Gazdík/Blanárík	Gazdík/Blanárík	Prednášky
9:30	TeamMeeting	Ďaďo/Gurka	Bombala/Gazdík/Blanárík	Gazdík/Blanárík	Prednášky
10:00	TeamMeeting		Ďaďo/Gurka		Prednášky
10:30	TeamMeeting		Ďaďo/Gurka		Prednášky
11:00	TeamMeeting	Ďaďo	Ďaďo/Gurka		
11:30	TeamMeeting	Ďaďo	Ďaďo/Gurka		
12:00	Ďaďo/Gurka	Ďaďo	Gurka	Stanimirovič/Jantová	
12:30	Gurka	Ďaďo	Gurka	Stanimirovič/Jantová	
13:00	Gurka		Prednášky	Stanimirovič/Jantová	
13:30	Gurka		Prednášky	Stanimirovič/Jantová	
14:00	Stanimirovič	Prednášky	Prednášky	Jantová	
14:30	Stanimirovič	Prednášky	Prednášky	Jantová	
15:00	Stanimirovič	Prednášky	Bombala/Blanárík	Jantová	
15:30	Stanimirovič	Prednášky	Bombala/Blanárík	Jantová	
16:00	Bombala	Prednášky	Bombala/Blanárík	Jantová	
16:30	Bombala	Prednášky	Bombala/Blanárík	Jantová	
17:00	Bombala	Prednášky	Stanimirovič/Jantová	Jantová	
17:30	Bombala	Prednášky	Stanimirovič/Jantová	Jantová	
18:00	Ďaďo/Stanimirovič/Gazdík/Gurka	Prednášky	TeamMeeting	Blanárík - práca	
18:30	Ďaďo/Stanimirovič/Gazdík/Gurka	Prednášky	TeamMeeting	Jantová - práca	
19:00	Ďaďo/Stanimirovič/Gazdík/Gurka		TeamMeeting	Ďaďo - práca	
19:30	Ďaďo/Stanimirovič/Gazdík/Gurka		TeamMeeting	Gazdík - práca	
20:00			TeamMeeting	Bombala - práca	
20:30			TeamMeeting	Stanimirovič - práca	
21:00					
21:30					

6 Metodiky

6.1 Definície dokončenia

User Story

- Má zapísaný názov, popis a akceptačné kritériá podľa príslušnej metodiky
- Prešla aktivitou Scrum Poker - dostala pridelené Story Pointy
- V prípade priradených entít typu Item sú dokončené všetky takéto entity
- V prípade implementačných úloh je schválený pull request

Sprint

- Všetky úlohy v šprinte spĺňajú definíciu dokončenia
- Pre prípadné nedokončené úlohy musí existovať výnimka majiteľom produktu
- Projekt je nasadený - akceptovateľná je aj ukážka verzie produktu majiteľovi produktu
- Backlog je aktualizovaný
- Šprint je zdokumentovaný, prebieha retrospektíva
- Šprint je označený ako úspešný product ownerom

Web

- Sídlo je verejne prístupné cez adresu požadovanú od vedenia predmetu
- Na webovom sídle sú základné informácie o projekte a tíme
- Na webovom sídle sa nachádza aspoň jeden kontaktný údaj na tím
- Na webe sú všetky potrebné súbory (retrospektívy, šprint reporty), je možné ich prezerat alebo stiahnuť

Analytická úloha

- Výsledok analýzy je prednesený minimálne v kanáli "Internal Development" (pozri metodiku komunikácie), prípadne v kanáli "General"
- Existuje písomný záznam analýzy - report či dokumentácia, ideálne vo formáte PDF
- Daný písomný záznam je nahratý ako príloha úlohe v systéme Azure DevOps
- Daný písomný záznam je nahratý v príslušnej zložke adresárovej štruktúry v systéme Sharepoint - resp. MS Teams

Implementačná úloha

- Vytvorená vlastná vetva s vhodnou konvenciou
- Napísaný kód pokrývajúci zadanie úlohy
- Otestovaná funkcionality - aspoň lokálne - **JE PRÍSNE ZAKÁZANÉ ODOVZDAŤ DO DEVELOP VETVY NEFUNKČNÉ DIELO, PRÍPADNE POŽIADAŤ O PULL REQUEST PRE VETVU S NEFUNKČNÝM DIELOM**
- Úloha spĺňa akceptačné kritériá

- Revízia kódu prebehla úspešne
- Kód je vsunutý do vetvy "develop"
- Kód je zdokumentovaný na miestach, kde to je potrebné (ako uzná za vhodné developer / schvaľovateľ)
- Ak je to vhodné/potrebné je aktualizovaná prípadná používateľská príručka

Infraštruktúrálna úloha

- Pri CI/CD úlohách je viacnásobne overená funkčnosť pipeline - build/release
- V prípade väčšieho zásahu do infraštruktúry, sú informovaní všetci členovia tímu cez kanál "General"
- V prípade výpadku služieb sú taktiež informovaní všetci členovia cez kanál "General"
- Pri úlohách s úpravou nastavení, ktoré už boli referencované v dokumentácií, je príslušná dokumentácia či príručka aktualizovaná - táto úloha môže byť pretvorená na item príslušnej US

Dokumentačná úloha

- Dokumentácia je schválená aspoň 1 ďalším členom tímu
- V prípade tvorby novej dokumentácie je táto nahratá na príslušné miesto - webové sídlo, MD file v Azure DevOps, či priečinok stromovej štruktúry v systéme Sharepoint, resp. MS Teams

6.2 Metodika popisu úloh

Poznámka: Nasledujúce podkapitoly môžu obsahovať anglické názvy niektorých elementov (sú prebraté zo systému pre riadenie projektu DevOps) a to z dôvodu jednoznačnej identifikácie jednotlivých polí pri zadávaní popisu úloh

6.2.1 User Story

1. Názov

- Názov je výstižný
- Názov je v anglickom jazyku

2. Popis

- Popis je v slovenskom jazyku, diakritika nie je nutná
- V popise je napísaná podstata úlohy, ak je potrebné aj dôvod implementácie.
- Dôležité je vpisovanie jednotlivých bodov, ktoré je potrebné podniknúť v rámci prác na tejto úlohe.

3. Akceptačné kritériá

- Akceptačné kritériá sú zadané bodovo v slovenskom jazyku, diakritika nie je nutná
- Každé z kritérií musí byť merateľné a jednoznačne dokázateľné
- Pri implementačných úlohách je jedno z nevyhnutných kritérií schválenie Pull-Requestu

4. Tag

- V prípade priradenia úlohy do segmentu je nutné priradiť Tag Frontend/Backend, etc.
- Musí byť zvolený najmenej jeden takýto Tag

6.2.2 Bug

1. Title

- Názov je výstižný
- Názov je v anglickom jazyku

2. Repro steps

- obsahuje menovité kroky na reprodukciu bugu či problému
- kroky sú celkom jednoznačné až po úroveň "klikni na tlačidlo"
- kroky sú v poradí, v ktorom bol problém simulovaný či objavený

3. System info

- obsahuje základné informácie o systéme / zariadení, kde sa udiala chyba
- obsahuje typ zariadenia - Notebook, PC, Tablet, Mobilný, telefón a pod.
- obsahuje typ operačného systému
- je potrebné uviesť aj prehliadač či program na ktorom sa podarilo chybu nasimulovať
- vzhľadom na určité špecifiká tohoto projektu je nutné uviesť v prípade notebookov, či majú dotykovú obrazovku

4. Attachments

- obsahuje videá, snímky obrazovky, či súbory s ktorými sa podarilo nasimulovať chybu
- v prípade snímky či záznamu obrazovky sa vyznačení problémové miesto

5. Resolved Reason

- zadáva riešiteľ
- smie byť vyplnené až po dokončení riešenia

6. Story Points

- riešiteľ nevyplní
- hodnoteniu odstraňovania bugov sa venuje metodická časť riadenia projektu a story pointy pre túto časť sa zadávajú dodatočne !

7. Priority *nadobúda iba nasledovné hodnoty

- 1 - Produkt sa nedá odoslať bez úspešného vyriešenia pracovného predmetu a mal by sa riešiť čo najskôr.
- 2 - Produkt sa nedá odoslať bez úspešného vyriešenia predmetu práce, nemusí sa však okamžite riešiť.

- 3 - Vyriešenie pracovnej položky je voliteľné na základe zdrojov, času a rizika

8. Severity *nadobúda iba nasledovné hodnoty

- 1 - Kritická: Musí sa opraviť. Porucha, ktorá spôsobuje ukončenie jedného alebo viacerých komponentov systému alebo celého systému alebo spôsobuje rozsiahle poškodenie údajov. Neexistujú žiadne prijateľné alternatívne metódy na dosiahnutie požadovaných výsledkov.
- 2 - Vysoká: Zvážte opravu. Porucha, ktorá spôsobuje ukončenie jedného alebo viacerých komponentov systému alebo celého systému alebo spôsobuje rozsiahle poškodenie údajov. Existuje však prijateľná alternatívna metóda nedosiahnutie požadovaných výsledkov.
- 3 - Stredná: Porucha, ktorá spôsobuje, že systém poskytuje nesprávne, neúplné alebo nekonzistentné výsledky.
- 4 - Nízka: Menší alebo kozmetický defekt, ktorý má prijateľné riešenie na dosiahnutie požadovaných výsledkov.

6.2.3 Epic

Nasledovná sekcia môže obsahovať už spomenuté parametre z vyššie uvedených sekcií. Preto budú opísané iba prijaté prípadné zmeny v popise, oproti predošlému zneniu.

- DESCRIPTION
- ACCEPTANCE CRITERIA - definuje skôr funkcionálne požiadavky
- STATUS START DATE - začiatok prác
- STATUS TARGET DATE - predpokladaný koniec prác
- PRIORITY - subjektívne, nadobúda hodnoty 1-4
- EFFORT - relatívny odhad množstva práce potrebnej na dokončenie epicu
- BUSINESS VALUE - priorita, ktorá zachytáva relatívnu hodnotu epicu v porovnaní s ostatnými položkami rovnakého typu. Čím vyššie číslo, tým vyššia je biznis hodnota.
- TIME CRITICALITY - Subjektívna merná jednotka, ktorá zachytáva, ako sa biznis hodnota epicu v priebehu času znižuje. Vyššie hodnoty znamenajú, že epic je zo svojej podstaty časovo kritickejší ako položky s nižšími hodnotami - nepovinné pole
- VALUE AREA - zákaznícka hodnota *nadobúda nasledovné hodnoty
 - Architectural : technické služby na implementáciu obchodných funkcií, ktoré poskytujú riešenie
 - Business : služby, ktoré naplňajú potreby zákazníkov alebo zainteresovaných strán a ktoré priamo prinášajú hodnotu pre zákazníka na podporu podnikania

6.2.4 Feature

Rovnaká štruktúra a pravidlá ako pri epicu (Vid' popis epicu)

6.2.5 Impediment

Impedimenty (prekážky) predstavujú neplánované činnosti. Ich riešenie si vyžaduje viac práce nad rámec toho, čo sa sleduje podľa skutočných požiadaviek

- DESCRIPTION - predstavuje popis problému a všetky závislosti
- RESOLUTION - ponúka riešenia, nahrádza akceptačné kritéria (viď User Story)
- PRIORITY - Subjektívne hodnotenie - nadobúda hodnoty 1 - 4

6.2.6 Backlog item

Rovnaká štruktúra a pravidlá ako pri epicu (Viď popis epicu).

6.2.7 Task

Ide o child element vlastnený User Story - Task je teda vždy priradený konkrétnej User Story. Zachytáva teda čiastkovú implementačnú či inú parciálnu prácu na User Story.

Nasledovná sekcia môže obsahovať už spomenuté parametre z vyššie uvedených sekcií. Preto budú opísané iba prijaté prípadné zmeny v popise, oproti predošlému zneniu.

- REMAINING WORK - V prípade ukončenia prác na danom tasku sa ihneď zaznačí odhad zostávajúcej práce vo formáte hod:min
- BLOCKED - nadobúda hodnoty ÁNO/NIE DÔLEŽITÉ ! V PRÍPADE OZNAČENIA TASKU AKO BLOKOVANÝ, JE POTREBNÉ IHNEĎ NAPÍSAŤ DO KOMENTÁRU DÔVOD BLOKÁCIE, PRÍPADNE OKREM TOHO NAMAPOVAŤ VZŤAH S ITEMOM, KTORÝ SÚVISÍ S BLOKÁCIU !

6.3 Metodika tímovej komunikácie

Na internú komunikáciu v tíme a s vedúcou tímu používame nástroj MS Teams. Nástroj umožňuje rozdeliť témy komunikácie na prvky v kanáloch. Nástroj poskytuje dobré vyhľadávanie v histórii a taktiež integrácie iných nástrojov. Nástroj zároveň vďaka integrácii na Sharepoint slúži ako úložisko pre zápisky zo stretnutí a iné dokumenty pre našu internú organizáciu.

6.3.1 Komunikačné kanály

- General - Spoločné stretnutia s vedúcou, riešenie problémov, výstupy, dokumenty, podklady
- Frontend *PRIVATE* - slúži na pridávanie odkazov a zaujímavosti segmentu Backend
- Backend *PRIVATE* - slúži na pridávanie odkazov a zaujímavosti segmentu Backend
- Internal Development - pondelkové stretnutia, interné stretnutia

6.3.2 Emailová komunikácia

Ako tím máme vytvorený e-mailový alias, ktorého správy sú presmerované do univerzitných e-mailových schránok jednotlivých členov vývojového tímu, mimo vedúcej Tímového projektu. Prostredníctvom e-mailu komunikujeme výhradne s tretími stranami, ktoré sú zapojené do procesu vývoja projektu (resp. za predpokladu, že takýto vzťah vznikne). Za pravidelné sledovanie a odpovedanie na prijaté správy je zodpovedný manažér komunikácie.

E-Mailová adresa: team@miaow.sk

6.3.3 Komunikácia v kanáloch

- Nové kanály môže zakladať iba vlastník tímu - Scrum master, ostatní členovia môžu o pridanie požiadať, kanál vytvára Scrum Master na základe posúdenia žiadosti
- Člen príslušného kanála vo vláknoch komunikuje výhradne o téme, na ktorú je kanál určený a to stručne a jasne

6.3.4 Stretnutia

Vzhľadom na pandemickú situáciu SARS-CoV-2 (COVID-19) prebiehajú všetky stretnutia výlučne online - cez nástroj MS Teams.

- Každé stretnutie vedie Scrum-Master - určuje najmä tému a smer komunikácie
- V prípade, že Scrum-Master doručí pred začiatkom alebo tesne po začiatku stretnutia agendu, sú členovia povinní túto zrevidovať, prípadne podať návrh o jej doplnenie. Následne zmeniť agendu stretnutia môže len Scrum-Master.
- Na prerušenie komunikačného toku využije člen tímu funkciu "zdvihnutie ruky ak nie je vypočutý, je možné slovne komunikáciu prerušiť - vstúpiť do komunikácie
- V prípade nepripojenia člena do 15 minút od začiatku stretnutia je považovaný za neprítomného. Túto skutočnosť nahlasuje Scrum Master vedeniu projektu / predmetu. V prípade neprítomnosti Scrum Mastra nahlasuje túto udalosť ktorýkoľvek člen tímu
- V prípade meškania, či ospravedlnenej neúčasti skutočnosť člen nahlási v kanáli General / Internal Development, prípadne inou cestou.
- Ak nie je určená iná agenda, či nie je Scrum Master prítomný, nevyhnutný rozsah stretnutia je Standup, prípadne pri začiatku/konci šprintu aj product review, retrospektíva s činnosťou Start, Stop, Continue, ako aj Plánovanie nasledujúceho šprintu

6.4 Metodika verziovania práce - kódu

Každý člen používa jediný GIT repozitár v systéme Azure Devops.

PRENOS KÓDU DO OSOBNÝCH REPOZITÁROV JE ZAKÁZANÝ !

Git repozitár má 2 hlavné a n vedľajších - vývojových developerských vetiev. Hlavné vetvy:

- master - produkčná verzia - pull request pri doručení otestovanej verzie produktu - slúži ako UAT - User Acceptance Test vetva
- develop - funkčná verzia najnovšie pridávanej funkcionality, práve testovaná, neskôr bude spojená s master vetvou - slúži ako TEST

UPOZORNENIE: PULL REQUEST DO VETVY "master" JE MOŽNÝ VÝHRADNE LEN Z VETVY DEVELOP A TO PO OTESTOVANÍ DANEJ VERZIE !

6.4.1 Postup verziovania

Nasledovné kroky tvoria životný cyklus vývoja úlohy

1. Ak developer nemá inicializovaný projekt, riadi sa dokumentáciou inicializácie - klonuje vetvu "develop". v prípade, že má developer projekt už inicializovaný z predošlého vývoja, aktualizuje si verziu projektu z vetvy "develop"
2. developer si vytvorí lokálnu vetvu, ktorú pomenuje v nasledovnej konvencii: [AISmeno]-[cisloulohy] a teda napríklad xgazdikm2-007 (Teda autor Úlohy je Martin Gazdík, pracuje na UC 007)

UPOZORNENIE ! ČÍSLO ZA MENOM NIKDY NEZNAČÍ ČÍSLO TASKU, ALE CELEJ USER STORY !

3. po dokončení úlohy developer vytvára commit, kde vhodne popíše odpracovaný progres - týmto krokom sa z jeho lokálnej vetvy stáva aj vzdialená !
4. Developer vytvára Pull Request kde spíše progres na úlohe - v anglickom jazyku a bodovo
5. Developer informuje členov segmentu o vytvorení Pull Requestu - cez príslušný kanál v aplikácii MS Teams
6. Nasleduje schvaľovacie / pripomienkovacie konanie
7. V prípade pripomienok developer dopracuje DO ROVNAKEJ VETVY pripomienky posudzovateľov a aktualizuje svoj commit
8. Pull request je schválený, obsah dopracovanej funkcionality vchádza do hlavnej vetvy

6.5 Metodika testovania produktu

Na základe predošlej diskusie, no najmä vzhľadom na fakt, že tento tím vytvára celkom nový produkt bolo rozhodnuté o nepísaní tzv. Unit testov.

Testovanie teda prebieha tradičnou metódou, na ktoré môžu no nemusia byť napísané testovacie scenáre.

- Testuje vždy taký člen tímu, ktorý sa nepodieľal na vývoji danej funkcionality, ani funkcionality s tým súvisiacej.
- K testu dochádza minimálne pri vydaní novej verzie produktu - pri prechode z vetvy "develop" do vetvy "master"
- Výstupom testu je stručná správa v systéme DevOps, prípadne MS Teams, ktorá opisuje testované scenáre ako aj výsledok testu.
- V prípade nedostatkov daný tester bezodkladne založí v systéme DevOps Bug - podľa metodiky.
- Pri najbližšom Standupe prezentuje tester výsledky testov.

6.6 Metodika riadenia dokumentácie

Dokumentácie členíme na nasledovné

1. Metodiky - LaTeX
2. Používateľské príručky - LaTeX
3. Inštalčné príručky - LaTeX
4. Dokumentácia kódu - MD

5. Dokumentácia k infraštruktúre - MD

V závislosti od typu dokumentácie platia vo všeobecnosti nasledovné pravidlá

- Názov súboru je v jazyku dokumentu
- Názov nikdy neobsahuje medzery ani diakritiku, preferovaný znak je "_"pred "
- Dokumentácia musí byť v primeranej úrovni podrobnosti

Mimo pravidiel, platia Dokumentačné FUP (Friendly User Policy):

- Nesnažiť sa nasilu písať viac
- Písať tak, aby porozumel aj človek, ktorý sa na to pozrie prvý krát
- Používať diakritiku a interpunkciu (mimo kódu)

6.7 Metodika Revízie zmien kódu

Revízia kódu je kľúčová pre správnosť fungovania finálneho produktu.

6.7.1 Autor

1. Po vytvorení vetvy je nutné vytvoriť Pull Request, popis musí jasne zachytiť zmeny, je možné použiť aj existujúcu commit správu, avšak len v prípade že stručne zachytáva všetky zmeny a to v anglickom jazyku
2. Commit sa musí linkovať s taskom / User Story s ktorým / ktorou súvisí ! S User Story sa linkuje taký commit, ktorý sa po schválení považuje za hotový
3. Autor vyberie recenzentov a priradí ich k danému Pull Requestu a to nasledovne:
 - Seba samého ako required
 - Aspoň jedného ďalšieho člena segmentu, ktorého sa úloha týka ako required
 - Iného člena / členov segmentu ktorého sa úloha týka ako optional
4. Ak si je autor svojou prácou istý, lokálne otestoval zmeny ako aj to, že neznefunkčnil ostatnú prácu, vykoná tzv. Self Approve - Schvaľuje za seba svoj pull request, čím dáva najavo, že takýto Pull Request je pripravený na posudok
5. Autor aktivuje funkciu "Auto Complete" pričom zvolí aj možnosť automatického odstránenia vetvy po schválení
6. Autor bezodkladne po vytvorení Pull Requestu notifikuje tím, minimálne na úrovni segmentu prostredníctvom systému MS Teams v príslušnom kanáli, pričom tých recenzentov, ktorých nastavil vždy aj označí, aby zabezpečil ich okamžitú notifikáciu (notifikácia v chate tímový projekt na facebooku je vítaná, nie však povinná)
7. Autor pravidelne kontroluje stav svojho Pull Requestu - v prípade komentárov:
 - Ak je komentár výčitkou, zapracuje všetky takéto výčitky ju a opäť notifikuje tím
 - Ak je komentár návrhom, mal by ho len prijať v systéme DevOps
 - Ak je komentár len poznámka, označí ju autor ako "Resolved to platí aj pre iné nedefinované komentáre, ktoré nie je nutné ďalej zapracovať

8. Po prijatí Pull Requestu je autor mailovo notifikovaný a jeho úloha presunutá ako dokončená

6.7.2 Recenzent

1. Recenzent pravidelne sleduje nové - aktívne Pull Requesty a to s vyššou prioritou ako vlastný vývoj - to by mohlo byť považované za brzdenie tímového vývoja produktu !
2. Recenzent si naštuduje, o čo v danej úlohe ide, zisťuje, či sú splnené akceptačné kritériá úlohy
3. Recenzent sleduje záložku "Files"v danom Pull Requeste, kde vidí aktuálne zmeny oproti predošlej verzii vetvy "develop"
4. V prípade komentárov klikne recenzent na riadok, ktorý chce okomentovať a napíše komentár, prípadne označí znakom @ osobu, ktorej sa komentár týka. Následne označí typ komentáru (či je potrebné ho zapracovať)
5. V prípade drobných návrhov v kóde (optimalizácia na pár riadkov, zbytočné medzery, malé veľké písmeno, atď) Označí recenzent uvedené riadky kódu a stlačí symbol žiarovky. Prepíše kód na návrh a návrh odošle. Autor tak nemusí danú zmenu robiť lokálne, ale vie ju jednoducho prijať.
6. Recenzent je povinný si danú vetvu stiahnuť a funkcionality otestovať. INAK NESMIE SCHVÁLIŤ DANÚ ZMENU

6.8 Metodika iterácií

Iterácie tvoria v prvom rade šprinty, ktoré trvajú 14 dní a ich počiatočné a koncové dátumy sú výhradne v správe vedenia predmetu. Pre každý šprint máme naplánované 4 stretnutia:

6.8.1 1. stretnutie - zmena šprintu

Na začiatku tohto stretnutia spoločne ukončíme predchádzajúci šprint. Urobíme product review a predvedieme to, čo sa nám podarilo vytvoriť za tento šprint. Prejdeme si s vedúcou user stories, ktoré sme mali naplánované a ona vyhodnotí či sme splnili definition of done pre každý story. Po akceptovaní user stories vedúcou tímu môžeme daný šprint uzavrieť. Následne prechádzame ku plánovaniu nového šprintu. Spoločne špecifikujeme user stories, ktoré pridáme sú súčasťou šprint backlogu. Pred naplánovaním ohodnotíme s pomocou scrum poker každú user story a priradíme im výsledné story pointy. Každú user story si rozdelíme na menšie tasky.

6.8.2 2. stretnutie - práca na user stories

Počas tohto stretnutia pracujeme na user stories, ktoré nám boli pridelené a riešime problémy, na ktoré sme narazili.

6.8.3 3. stretnutie - práca na user stories

Na tomto stretnutí pokračujeme v práci na user stories pričom je dôležité, aby každá story by mala byť aspoň začatá. Riešime ďalšie vzniknuté problémy podľa dôležitosti.

6.8.4 4. stretnutie - retrospektíva

Na poslednom stretnutí tejto iterácie vedie scrum master retrospektívu, kedy každý člen tímu má vyjadriť svoj názor. Zameriavame sa na to, čo by sme mali začať robiť, čo by sme mali prestať robiť a v čom by sme mali pokračovať. Odpovede si zapisujeme a na konci budúceho šprintu zisťujeme či sa nám ich poradilo naplniť. Ďalej pokračujeme prácou na pridelených stories.

6.9 Metodika riadenia úloh

Na riadenie úloh používame nástroj Azure DevOps.

6.10 Životný proces USER STORY

6.10.1 Backlog

Po identifikovaní sa user story do backlogu. User stories, ktoré sa tu nachádzajú nemusia mať opis, nie sú rozdelené na tasky a ani nie je ohodnotená ich zložitosť. User stories, ktoré sú v stave *Backlog* nám slúžia ako zdroj nápadov a funkcionality, ktorú by sme chceli v budúcnosti v projekte mať.

6.10.2 Ready

User stories, ktoré sú naplanované na aktuálny šprint sa presúvajú do stavu *Ready*. Majú už pridaný opis, ohodnotenie story pointami a majú pridelené jednotlivé tasky.

6.10.3 Develop

Akonáhle sa začne na user story pracovať, tak sa presúva do stavu *Develop*.

6.10.4 Blocked

Do stavu *Blocked* sa user story dostane, keď nie je možné na nej ďalej pracovať, lebo určitá funkcionality z inej user story nie je ešte hotová. V tomto stave sa nachádza dovtedy kým nie je dôvod jej zablokovania odstránený.

6.10.5 Review

User story sa do tohto stavu dostane, keď je práca na nej ukončená. V stave Review zostáva až kým nie vytvorený pull request na ňu schválený.

6.10.6 Test

Do stavu *Test* sa user story dostane ak je hotová jej funkcionality, ale je ju ešte potrebné otestovať. V tomto stave zostáva až dovtedy kým nie sú všetky testy vykonané.

6.10.7 Done

V prípade schválenia pull requestu sa user story automaticky presunie do stavu *Done*. Tento stav je konečný.

Slovenská technická univerzita
Fakulta informatiky a informačných technológií

Inžinierske dielo

Tímový projekt MIAOW

Predmet:	Tímový projekt I.
Členovia tímu:	Bc. Martin Gazdík Bc. Timotej Gurka Bc. Marek Ďaďo Bc. Tamara Jantová Bc. Matej Blanárik Bc. Martin Bombala Bc. Nikola Stanimirovič
Vedúci práce:	Mgr. Monika Kováčová, PhD.
Akademický rok:	2020/2021

Obsah

1	O projekte MIAOW	2
1.1	Úvod	2
1.2	Pôvodné zadanie projektu	2
1.3	Globálne ciele projektu na zimný semester	3
1.4	Celkový pohľad na systém	3
1.4.1	Základná architektúra	4
1.5	Technológie	5
2	Infraštruktúra	6
3	Príručky	6
3.1	Príručka pre segment Frontend	6
3.1.1	Node	6
3.1.2	ESLint	6
3.1.3	Development Server	6
3.2	Príručka pre segment Backend	6
3.2.1	Mongo	7
3.2.2	Postgres	7
3.2.3	DataGrip	7
3.2.4	Django	7
4	Moduly systému	9
4.1	Prezentačná vrstva	11
4.1.1	Modul User Registration	11
4.1.2	Modul Competition Registration	11
4.1.3	Modul Judges Manager	11
4.1.4	Modul Registration Manager	11
4.1.5	Modul LOVs Manager	11
4.1.6	Modul EMS Manager	11
4.1.7	Modul EMS Upload	11
4.2	Aplikačná vrstva	12
4.3	Databázová vrstva	12

Podiel práce na dokumentácii k inžinierskemu dielu	
Názov sekcie	Autor
O projekte	Martin Gazdík, Tamara Jantová
Infraštruktúra	Martin Gazdík
Príručky	Tamara Jantová, Marek Ďaďo, Martin Bombala, Matej Blanárik
Moduly systému	Nikola Stanimirovič

1 O projekte MIAOW

Táto kapitola obsahuje úvod do projektu, definované globálne ciele pre projekt, celkový pohľad na systém týkajúci sa architektúry a opis všetkých technológií, ktoré boli použité.

1.1 Úvod

Projekt MIAOW je zameraný na analýzu, návrh a implementáciu kompletného systému manažmentu výstavy. Má pomáhať predovšetkým organizátorom výstavy, kde riešia množstvo úloh, ktoré môžu byť prenesené do sveta webovej aplikácie. Rovnako však vnímame možnosť postupného nahradenia množstva tlačových dokumentov a zjednodušenie procesov naprieč výstavou a ich integráciu do takejto aplikácie.

1.2 Pôvodné zadanie projektu

Medzinárodná výstava mačiek (MVM) nie je len o mačkách. Je to v skutočnosti zložitý živý organizmus, ktorý pripravujete “na život” ako organizátor niekoľko mesiacov a potom sa vám na jednom mieste, v jednom čase objaví 200 – 1500 mačiek (z obvykle okolo 15-25 štátov), 150 – 1000 vystavovateľov (väčšina ovláda tak maximálne materinský jazyk a trochu angličtinu), 7 – 30 posudzovateľov a to všetko začne a skončí v intervale 12 tich hodín. Aby ste to ako organizátor v zdraví prežil, potrebujete skutočne inteligentný informačný systém a vytvoriť takýto systém je primárny cieľ tohto projektu.

Medzinárodná výstava mačiek má presne definované pravidlá – Show Rules, Show Organisers Guide (http://fifeweb.org/wp/lib/lib_current.php), ktoré vymedzujú rámec povinných a odporúčaných činností. Pravidlá aj činnosti budú partnerom podrobne vysvetlené vrátane časových súsledností a best practises. Vytvorený informačný systém musí tieto pravidlá dodržať v plnom rozsahu.

Realizácia projektu si bude vyžadovať komplexný projektový manažment aj na strane developer-ského teamu. Výhodou je fakt, že „zadávateľ“ má presnú predstavu o tom, aké úlohy musí byť systém schopný zabezpečiť. Team bude potrebovať zvládnuť celý proces dodávania softvérového riešenia na mieru od analýzy požiadaviek a definície špecifikácie, nastudovania problematiky, High-level aj Low-level designu, vrátane „kódovania“ (zero-coding), testovania a nasadenia aplikácie až po ostré spustenie do prevádzky. K dispozícii budú reálne raw dáta.

Projekt musí koordinovať časové hladiny: činnosti pred začatím výstavy, v deň výstavy pred začatím posudzovania, počas vlastných súťaží, počas Best IN SHOW (vrchol výstavy) a rovnako aj po výstave.

Projekt musí koordinovať personálne hladiny: činnosti vykonávané organizačným tímom, veterinárom, riaditeľom výstavy, posudzovateľom, stewardom, vystavovateľmi. Len návštevníci a mačky nebudú mať žiadne špeciálne hladiny.

Projekt musí koordinovať činnosti výstavného výboru (dokumentácia v súvislosti s výstavou, posudzovaním, súťažami), posudzovateľa (dokumentácia hodnotenia mačky) – ideálne mobilná aplikácia (aby hodnotenie mačky zodpovedalo pravidlám FIFÉ), vystavovateľa – ideálne mobilná aplikácia (aby doniesol na stôl posudzovateľovi tú správnu mačku a v tom čase, keď to posudzovateľ očakáva bez časového oneskorenia), činnosti stewardov pri realizácii BIS.

1.3 Globálne ciele projektu na zimný semester

Nakoľko ide o celkom nový projekt, cieľom na zimný semester je najmä detailne pochopiť, analyzovať a zdokumentovať aktuálny proces výstavy a to najmä:

1. Aktérov, ktorí vstupujú do procesu výstavy
2. Jednotlivé procesy, ktoré prebiehajú pred, počas a po výstave
3. Závislosti jednotlivých entít a procesov
4. Pridanú hodnotu digitalizácie čiastkových procesov
5. Pravidlá výstavy či procesy, ktoré musia byť zachované a nie je možné ich navrhnúť inak či zahrnúť do procesu digitalizácie

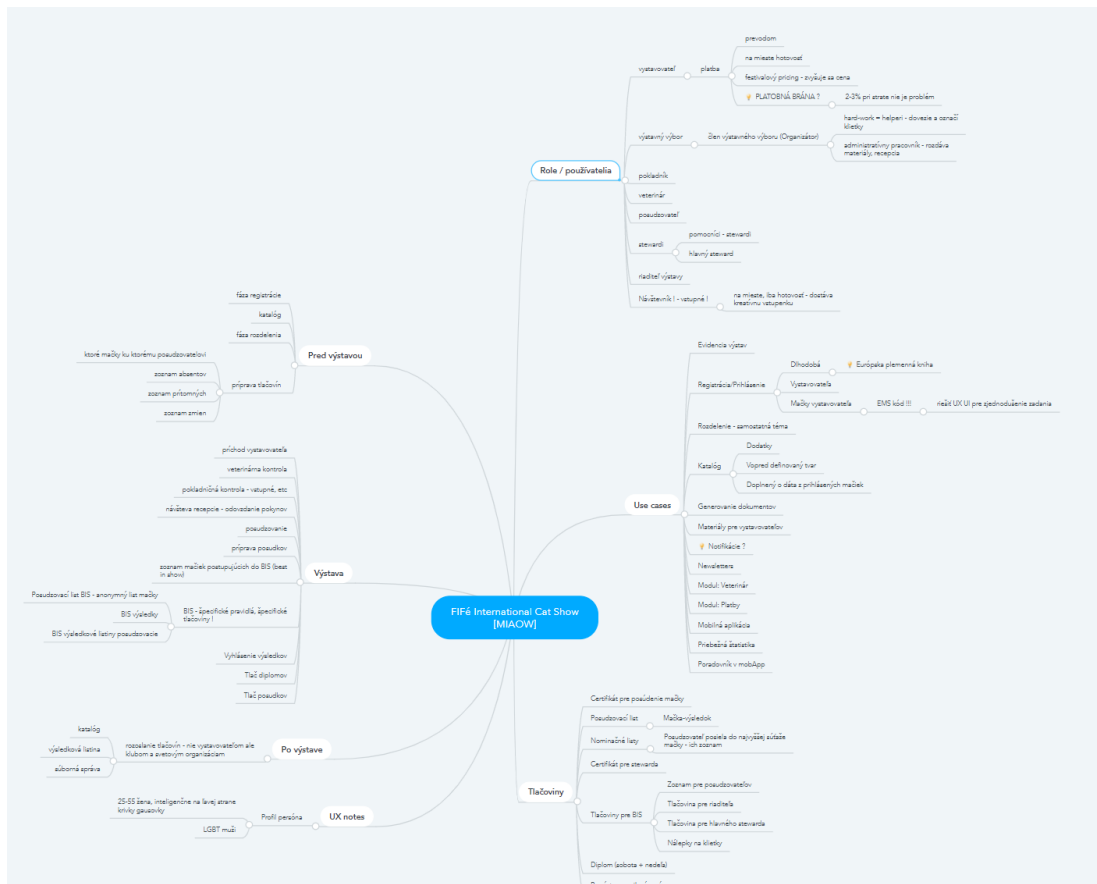
Následne navrhnúť koncept systému, vrátane modulov - nosných častí systému, dátový model a architektúru takejto aplikácie, analyzovať a vybrať technológie, ktoré sú vhodné na jej vývoj a implementovať prvú časť takéhoto systému.

Na konci semestra by sme radi prezentovali čiastkovú funkčnosť registrácie na výstavu, ktorú by sme radi doručili aktérom, najmä vystavovateľom, aby sme získali relevantnú spätnú väzbu na novo-vznikajúci proces registrácie.

1.4 Celkový pohľad na systém

Kapitola obsahuje opis základnej architektúry spolu s obrázkom, na ktorom je možné vidieť všetky komponenty.

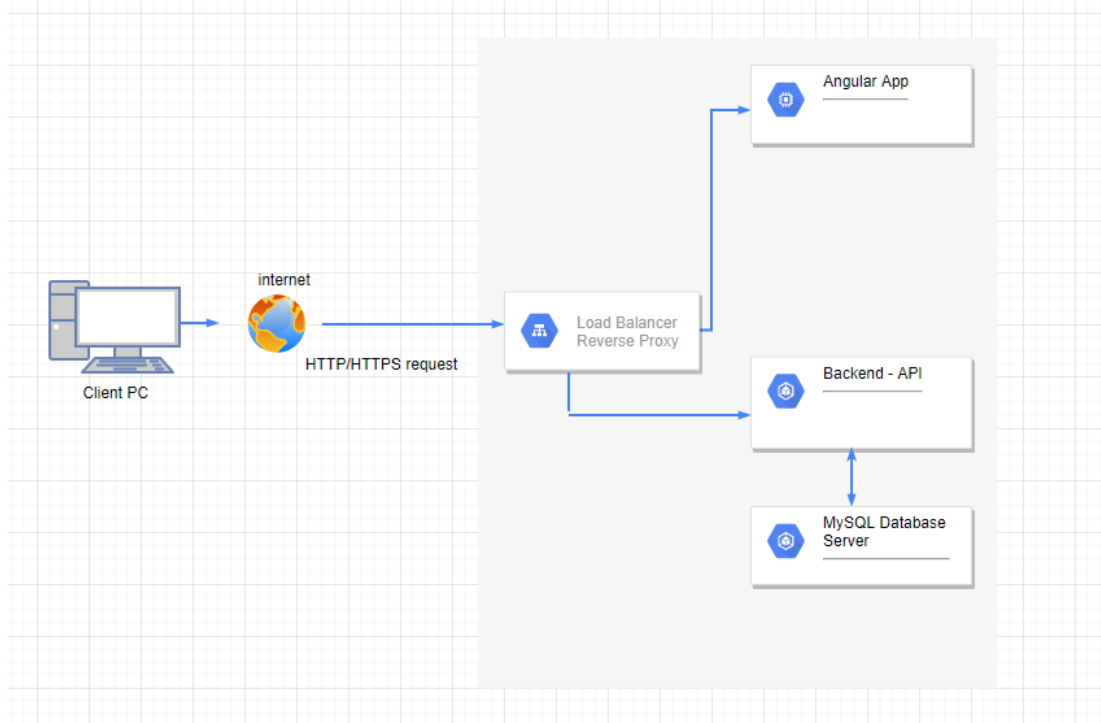
Pre lepšie pochopenie požiadaviek sme vytvorili mind mapu, ktorá združuje základné pojmy a poznatky o aktuálnom procese, ktorý budeme digitalizovať.



Obr. 1: Mind mapa produktu

1.4.1 Základná architektúra

Na obrázku nižšie je možné vidieť aktuálnu verziu hrubého návrhu základnej architektúry projektu.



Obr. 2: Základný návrh architektúry systému

1.5 Technológie

Po úvodnej analýze, sme sa rozhodli využívať nasledovné technológie:

- Angular**
 Na vytvorenie koncovej klientskej webovej aplikácie používame Angular - open-source framework na vývoj webových aplikácií s využitím programovacieho jazyka TypeScript
- Devextreme**
 Pre uľahčenie no najmä výrazné urýchlenie vývoja sme sa rozhodli používať komponenty do Angularu - Devextreme, ktoré sú pre nekomerčný vývoj zdarma a poskytujú množstvo komponentov, ako dátové gridy, formuláre, grafy a podobne.
- Django**
 Na tvorbu tzv. kontroléra pre jednotlivé entity, postavenom na REST API, ďalej len ako Backend - využívame Django - webový framework postavený na programovacom jazyku python.
- Swagger**
 Pre udržanie dokumentácie backendu priamo z kódu sme inicializovali nástroj na dokumentáciu vytváraných RESTových služieb Swagger UI.
- PostgreSQL**
 Ako dátový sklad sme sa rozhodli využiť SQL Databázu, konkrétne Postgres.

2 Infraštruktúra

3 Príručky

V nasledujúcej časti prikkladáme základné príručky určené pre vývojárov vznikajúceho systému

3.1 Príručka pre segment Frontend

3.1.1 Node

V projekte by sme mali používať Node verziu 14.6.0. Stačí ju mať v systéme nainštalovanú a pri prepnutí sa do priečinka zadať príkaz `npm use`. Tento príkaz vytiahne verziu zo súboru `.nvmrc`.

Keď sa nachádzame v konkrétnom priečinku, stačí zadať príkaz `npm install`, čím sa nainštalujú všetky potrebné moduly.

3.1.2 ESLint

ESLint nám analyzuje kód a stará sa o jeho úpravu za účelom jednotnosti a ucelenosti pri dodržiavaní vopred dohodnutých pravidiel a konvencií. Nakoľko sa nachádza v súbore `package.json` (v `AdministratorWebApp` aj `CustomerWebApp`), stačí spustiť príkaz `npm install` a nainštaluje sa spoločne s ostatnými modulmi.

Pokiaľ chceme, aby ESLint upravoval kód po uložení súborov v prostredí WebStorm, treba postupovať nasledovne:

1. Preferences -> Languages & Frameworks -> JavaScript -> Code Quality Tools -> ESLint
2. Treba zaškrtnúť Manual ESLint configuration
3. ESLint package: Detect package and configuration file from the nearest package.json
4. Zaškrtnúť Run eslint -fix on save
5. Kliknúť na Apply

V prípade VS Code je potrebné mať nainštalované rozšírenie s názvom ESLint. Predvolené nastavenie spôsobuje, že na nesprávne zapísaný kód je programátor síce upozornený, ale po uložení ESLint automaticky kód neupravuje. Toto nastavenie je možné zmeniť postupom na stránke: <https://marketplace.visualstudio.com/items?itemName=dbaeumer.vscode-eslint>.

3.1.3 Development Server

Zadanie príkazu `ng serve`, na `http://localhost:4200/` môžeme po zadaní príkazu vidieť aktuálnu bežiacu aplikáciu. Aplikácia sa sama znovu načíta pri akejkoľvek zmene zdrojových súborov.

3.2 Príručka pre segment Backend

Nasledovné podkapitoly obsahujú kroky pre inicializáciu danej technológie. Pre zjednodušenie pochopenia vykonávania jednotlivých krokov sme sa rozhodli ponechať príručku pre inicializáciu backend segmentu v **anglickom jazyku**.

3.2.1 Mongo

1. Install mongo
2. Open MongoDB Compass
3. Create Database cat-changes
4. Create Collections: changes

3.2.2 Postgres

1. Install pgAdmin
2. Install Postgres
3. Open pgAdmin
4. Create user
 - Username: cat-user
 - Password: cat-user
5. Create database
 - Name: cat-db
 - Add user: cat-user
 - Set privileges: all
6. Name: cat-db
7. Add user: cat-user
8. Set privileges: all

3.2.3 DataGrip

1. Install DataGrip (or preferred db viewer)
2. Connect to both databases

3.2.4 Django

1. Cd <repo pathname>
2. django-admin startproject backend
3. Cd backend
4. Install requirements - pip install -r requirements.txt (prípadne python3 -m pip install -r requirements.txt)
5. Add databases to settings.py (obrázok)

```

DATABASES = {
    'default': {
        'ENGINE': 'django.db.backends.postgresql_psycopg2',
        'NAME': 'cat-db',
        'USER': 'cat-user',
        'PASSWORD': 'cat-user',
        'HOST': 'localhost',
        'PORT': '5432'
    },
    'changes': {
        'ENGINE': 'djongo',
        'NAME': 'cat-changes',
        'HOST': 'localhost',
        'PORT': '27017'
    }
}

```

Obr. 3: *settings.py*

6. Run commands:
 - python manage.py makemigrations
 - python manage.py migrate
7. Run django app
 - python manage.py runserver

Požiadavky pre projekt:

1. Django==3.0.5
2. drf_yasg==1.17.1
3. djangorestframework==3.11.0
4. django-rest-swagger==2.2.0
5. psycopg2==2.8.5
6. djongo==1.3.3
7. pandas==1.1.4
8. xlrd==1.2.0
9. openpyxl==3.0.5
10. django-cors-headers==3.5.0

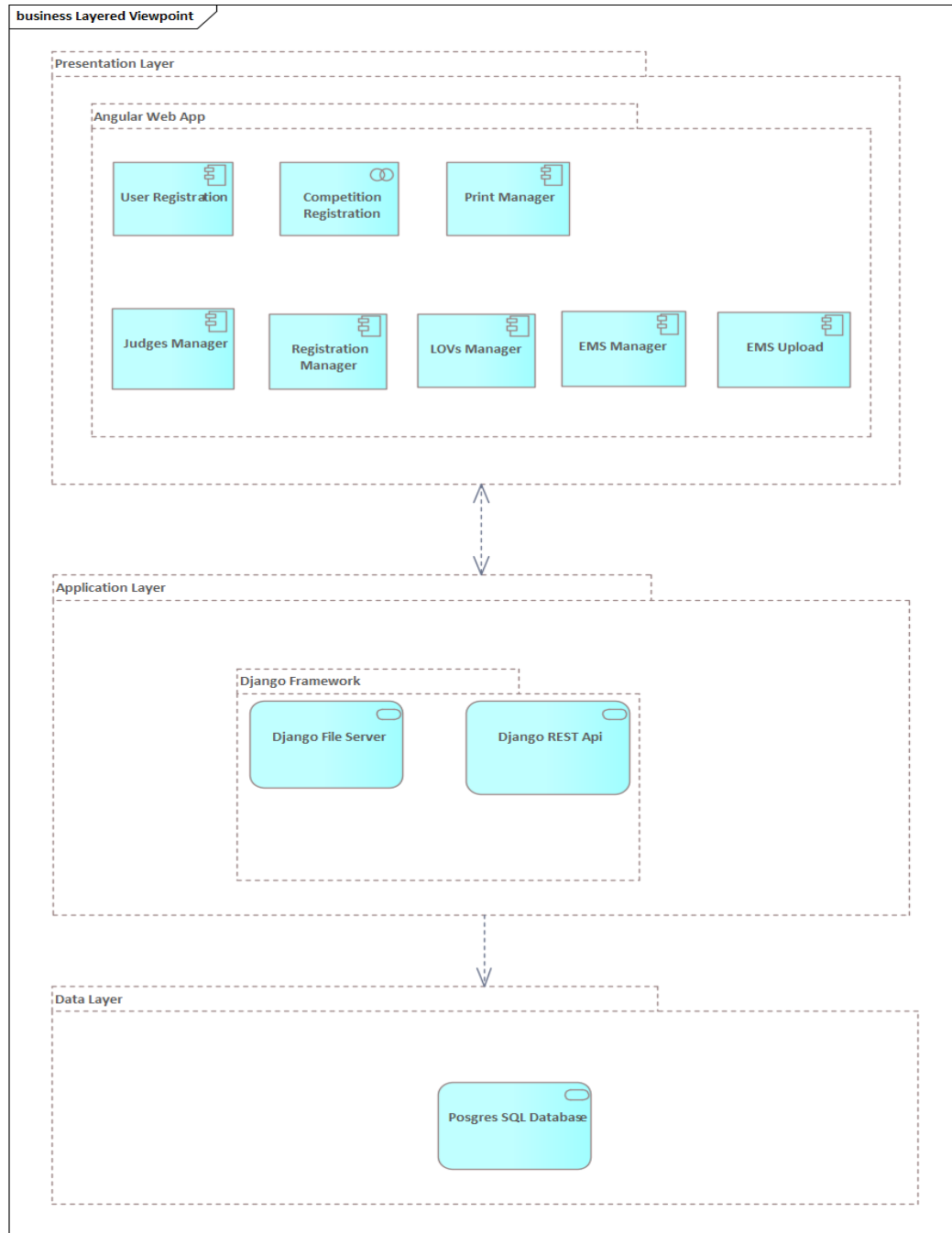
4 Moduly systému

Táto kapitola opisuje jednotlivé moduly systému a je členená na tri podkapitoly, ktoré korešpondujú s jednotlivými vrstvami systému.

Jednotlivé vrstvy systému:

- Prezentačná vrstva
- Aplikačná vrstva
- Databázová vrstva

Na obrázku nižšie je vidno jednotlivé vrstvy systému spolu s modulmi, ktoré sa na týchto vrstvách nachádzajú.



Obr. 4: Diagram modulov systému

4.1 Prezentačná vrstva

Prezentačnú vrstvu systému tvorí webová Angular aplikácia prostredníctvom, ktorej používateľ interaguje so systémom.

4.1.1 Modul User Registration

Prostredníctvom tohto modulu je umožnené sa zaregistrovať novému používateľovi do systému. Vyplnené údaje, ktoré používateľ do systému zadá sú následne prostredníctvom API volania spracované a uložené do Databázy.

4.1.2 Modul Competition Registration

Prostredníctvom tohto modulu je umožnené sa prihlásiť na konkrétnu súťaž a zaregistrovať na ňu aj svoje mačky. Tento modul umožní vyplniť údaje o mačke do formulára, ktorý obsahuje validácie jednotlivých polí aj kódu EMS. Pri chybnom zadaní údajov je používateľ notifikovaný, validuje sa aj samotný EMS kód.

4.1.3 Modul Judges Manager

Prostredníctvom tohto modulu je umožnené pridávať, prípadne odoberať jednotlivých posudzovateľov do súťaže. Rovnako je možné validne vytvoreným posudzovateľom odobrať, prípadne priradiť nejakú už registrovanú mačku na posúdenie.

4.1.4 Modul Registration Manager

Prostredníctvom tohto modulu je umožnené organizátorovi súťaže upravovať jednotlivé registrácie ešte aj v deň konania súťaže. Takto upravená registrácia je uložená do databázy.

4.1.5 Modul LOVs Manager

Prostredníctvom tohto modulu je umožnené si prezeráť a editovať všetky databázové číselníky za účelom flexibilnejšieho reagovania na prípadné potreby rozšírenia nejakého z číselníkov.

4.1.6 Modul EMS Manager

Prostredníctvom tohto modulu je umožnené si prezeráť a editovať jednotlivé EMS záznamy nakoľko sa môže stať, že sa potrebný záznam v importovanom súbore s EMS kódmi nenachádzal.

Rovnako sa tento manažér využíva na kontrolu priebehu a výsledného importu súboru s EMS kódmi. Systémový administrátor, tak môže rýchlo skontrolovať, či daný import súboru s EMS kódmi prebehol úspešne.

4.1.7 Modul EMS Upload

Prostredníctvom tohto modulu je umožnené nahrať validný súbor typu xlsx, obsahujúci EMS kódy. Tento súbor je následne parsovaný za účelom získania potrebných dát, nakoľko formát súboru nespĺňa požiadavky na uloženie do relačnej databázy.

Získané dáta so súboru sú následne uložené do databázy.

4.2 Aplikačná vrstva

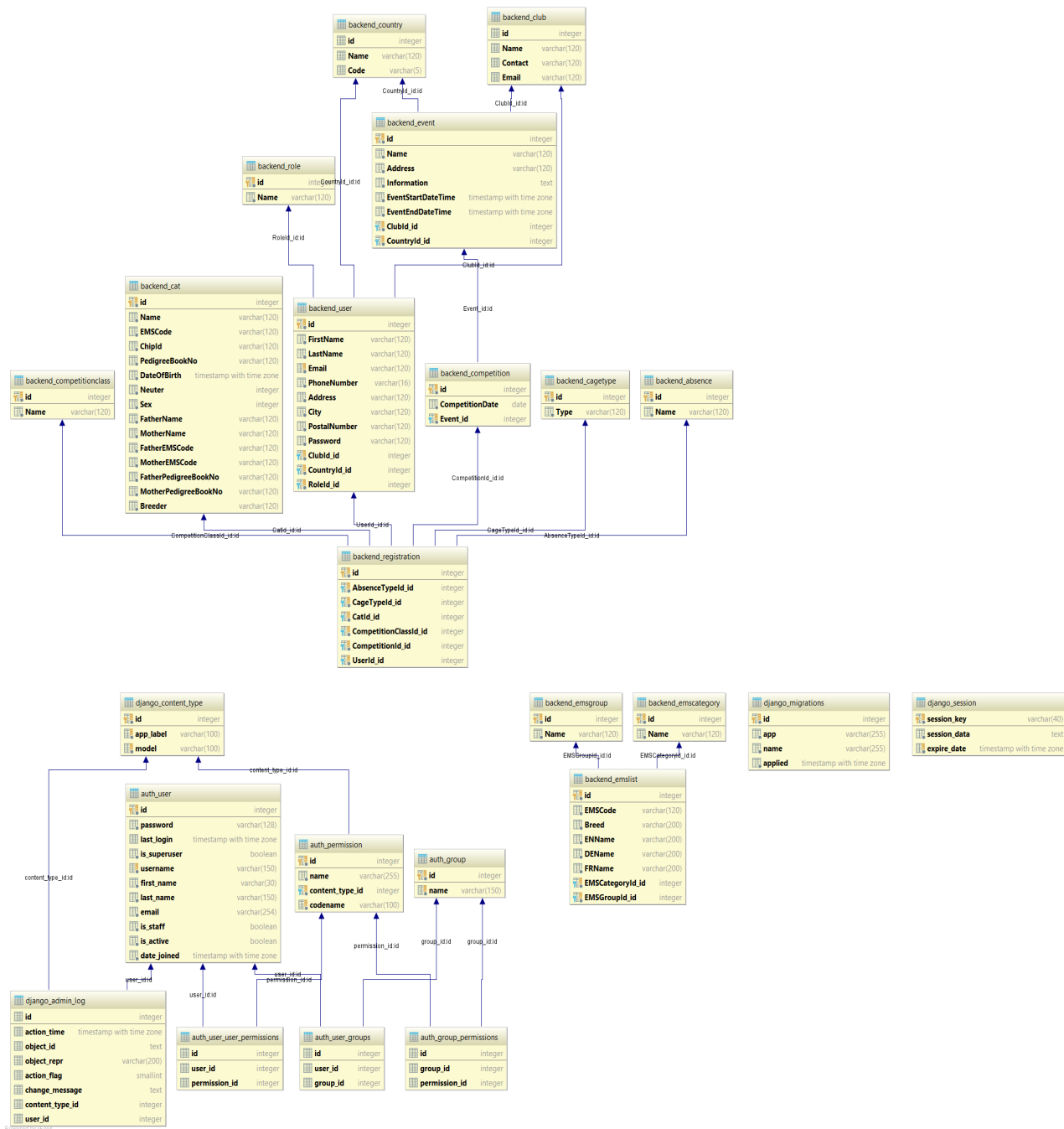
Aplikačnú vrstvu systému tvorí Django Framework, ktorý plní úlohu File servera a poskytuje REST Api na komunikáciu s webovou Angular aplikáciou.

Dokumentáciu k aplikačnej vrstve v tomto dokumente neuvádzame nakoľko je na serveri nainštalovaný a nakonfigurovaný Swagger.

4.3 Databázová vrstva

Databázovú vrstvu systému tvorí Postgres SQL Databáza, ktorá slúži na ukladanie jednotlivých entít systému.

Na obrázku nižšie sa nachádza dátový model systému.



Obr. 5: Dátový model