

SLOVENSKÁ TECHNICKÁ UNIVERZITA V BRATISLAVE
Fakulta informatiky a informačných technológií
Ilkovičova 2, 842 16 Bratislava 4

Dokumentácia k riadeniu

Etapa 2 – letný semester

Tímový projekt
2018/2019

Tím č. 04 : Fastar

Bc. Michaela Balážová
Bc. Kamil Janeček
Bc. Tomáš Jendrejčák
Bc. Matúš Kalafut
Bc. Matej Končál
Bc. Michal Maňak
Bc. Ján Vnenčák

Vedúci projektu: Ing. Kamil Burda

Dátum poslednej úpravy: 10.05.2019

Obsah

1. Úvod.....	4
2. Roly členov tímu a podiel práce	5
2.1. Opis pridelených rolí.....	6
2.2. Podiel práce.....	6
3. Aplikácie manažmentov.....	8
4. Sumarizácie šprintov.....	9
4.1. Šprint č. 1 (09.10.2018 - 23.10.2018).....	9
4.1.1. Retrospektíva šprintu č. 1	9
4.2. Šprint č. 2 (23.10.2018 - 06.11.2018).....	10
4.2.1. Retrospektíva šprintu č. 2	10
4.3. Šprint č. 3 (06.11.2018 - 20.11.2018).....	11
4.3.1. Retrospektíva šprintu č. 3	11
4.4. Šprint č. 4 (20.11.2018 - 04.12.2018).....	12
4.4.1. Retrospektíva šprintu č. 4	12
4.5. Šprint č. 5 (04.12.2018 - 18.12.2018).....	13
4.5.1. Retrospektíva šprintu č. 5	13
4.6. Šprint č. x (18.12.2018 - 12.02.2019).....	13
4.7. Šprint č. 6 (12.02.2019 - 26.02.2019).....	14
4.7.1. Retrospektíva šprintu č. 6	15
4.8. Šprint č. 7 (26.02.2019 - 12.03.2019).....	15
4.8.1. Retrospektíva šprintu č. 7	16
4.9. Šprint č. 8 (12.03.2019 - 26.03.2019).....	16
4.9.1. Retrospektíva šprintu č. 8	17
4.10. Šprint č. 9 (26.03.2019 - 09.04.2019).....	17
4.10.1. Retrospektíva šprintu č. 9	18
4.11. Šprint č. 10 (09.04.2019 - 23.04.2019).....	18
4.11.1. Retrospektíva šprintu č. 10	19
4.12. Šprint č. 11 (23.04.2019 - 07.05.2019).....	19
4.12.1. Retrospektíva šprintu č. 11	20
5. Globálna retrospektíva	21
6. Metodiky.....	22
6.1. Metodika pre prehliadku zdrojového kódu	22
6.1.1. Prostredie	22

6.2. Metodika pre písanie zdrojového kódu.....	22
6.2.1. Všeobecné konvencie.....	22
6.2.2. Konvencie pre Python.....	22
6.2.3. Konvencie pre Javascript	23
6.3. Metodika pre verziovanie zdrojového kódu	23
6.4. Metodika pre nástroj JIRA.....	25
6.4.1. Úlohy.....	25
6.4.2. Roly.....	25
6.4.3. Vytváranie úloh.....	25
6.4.4. Stavby	26
6.4.5 Akceptovanie user story Product ownerom	27
6.5. Metodika ku dokumentácií	28
6.5.1. Formálna úprava	28
6.5.2. Jazyk	29
6.5.3. Tvorba retrospektívy šprintu.....	29
6.5.4. Dokumenty.....	29
6.5.5. Technická dokumentácia.....	29
6.6. Metodika Definition of ready.....	30
6.7. Metodika Definition of done.....	30
6.8. Metodika plánovania.....	30
6.9. Metodika komunikácie.....	31
6.9.1. Vytvorenie nového kanálu	31
6.9.2. Existujúce kanály komunikácie	31
7. Export evidencie úloh	33
8. Webové sídlo tímu	34

1. Úvod

V dokumentácii riadenia zachytávame ako náš tím funguje, opisujeme samotný tím, rozdelenie úloh medzi členov tímu a celkové riadenie v rámci predmetu Tímový projekt. Významná časť tejto dokumentácie je venovaná aj používaným metodikám, ktoré sme si spoločne vytvorili a ktoré na projekte používame. Opisujeme tiež už realizované šprinty, popis jednotlivých úloh v šprinte, sumarizáciu šprintu a retrospektívu. Ako dôkaz pre našu vykonanú prácu slúžia exporty z nástroja JIRA.

2. Roly členov tímu a podiel práce

Náš tím sa skladá zo siedmich členov. Každý člen tímu má zodpovednosť za určitú oblasť pri vývoji produktu a pri úlohách týkajúcich sa manažmentu.

Bc. Matej Končál

- systémový administrátor
- testovanie
- machine learning expert

Bc. Matúš Kalafut

- vývojár
- machine learning expert
- expert na vizualizačné prostriedky (Bokeh)

Bc. Michal Maňak

- scrum master
- machine learning expert

Bc. Kamil Janeček

- systémový administrátor
- testovanie
- kvalita kódu
- verziovanie

Bc. Tomáš Jendrejčák

- machine learning expert
- testovanie

Bc. Michaela Balážová

- dokumentácia
- machine learning expert
- junior expert na vizualizačné prostriedky (Bokeh)

Bc. Ján Vnenčák

- webové sídlo tímu
- logger

2.1. Opis pridelených rolí

Systemový administrátor - spravuje a konfiguruje serverovú časť aplikácie

Testovanie - odborník na vytváranie testov

Machine learning expert - implementácia machine learning modulu

Vývojár - hlavný vývojár tímu, implementácia vo viacerých moduloch projektu

Kvalita kódu - sledovanie a dodržiavanie kvality vytvoreného kódu

Verziovanie - dohliadanie na správne verziovanie

Dokumentácia - tvorba dokumentácie, jej úprava

Webové sídlo tímu - implementácia a spravovanie webovej stránky tímu

Expert na vizualizačné prostriedky - tvorba funkcií pre vizualizovanie dát

2.2. Podiel práce

V rámci podielu prác pri moduloch rátame celkový strávený čas, či už je to implementácia, návrh, testovanie alebo code review. Taktiež pri práci na moduloch zahŕňame aj technickú dokumentáciu, ktorá sa píše spolu so zdrojovým kódom.

Pri stĺpci iné zahŕňame aktivity, ktoré bolo potrebné vykonať buď pre nastavenie nástrojov alebo vytvorenie a správu webového sídla tímu.

Meno člena tímu	server modul	machine learning	logger modul	dokumentácia	iné (webstránka, JIRA)
Bc. Matej Končál	25%	24%	5%	6%	20%
Bc. Matúš Kalafut	8%	19%	5%	8%	5%
Bc. Michal Maňak	8%	24%	5%	6%	10%
Bc. Kamil Janeček	35%	5%	5%	8%	25%
Bc. Tomáš Jendrejčák	8%	19%	5%	6%	5%

Bc. Michaela Balážová	8%	9%	5%	60%	5%
Bc. Ján Vnenčák	8%	0%	70%	6%	30%

3. Aplikácie manažmentov

Do procesu tvorby metodík sa zapojil každý člen tímu, každý mal na starosti vypracovanie jednej z nich. Tvorba metodiky spočívala najprv zo spoločnej konverzácie a diskusie ohľadne vzniknutých problémov, až kým sa celý tím nedohodol na spoločných riešeniach a následne bolo realizované jej vypracovanie.

Väčšina metodík sa zatiaľ počas práce na projekte nemenili - napr. verziovanie, kde bol dohodnutý formát ako majú vyzerat' rôzne vetvy a commit správy v nich. Pri metodike ohľadne dokumentácie bolo dohodnuté, že pre lepšiu prístupnosť širšej verejnosti sa dokumentácia k inžinierskemu dielu bude vypracovávať aj v anglickom jazyku. V metodike ohľadne nástroja JIRA boli upresnené prechody medzi stavmi pre rôzne úrovne (story, sub-task).

V niektorých šprintoch sú zaradené samostatne stojace tasky, ktoré nepatria pod žiadnu *story* a tým pádom nemajú odhadnuté *story points*. V tomto prípade sa jedná o funkcionality, ktorá musí byť realizovaná pre úspešné dokončenie šprintu. Sú to úlohy, ktoré nevieme namapovať pre žiadnu *story*, resp. neposkytujú priamu funkcionality pre používateľa - napr. vytvorenie demo ukážky pre PO alebo naštudovanie problematiky, ktorá sa bude realizovať v nasledujúcom šprinte.

Tiež sme si pridali do metodiky plánovania, že každá *story*, ktorá vstupuje do nového šprintu musí mať priradenú osobu, ktorá je za ňu zodpovedná.

4. Sumarizácie šprintov

4.1. Šprint č. 1 (09.10.2018 - 23.10.2018)

V prvom šprinte sme sa zamerali na úlohy týkajúce sa manažmentu a vytvorenie prvotného loggeru pre počítačovú myš. Vytvorili sme Big Picture projektu, kde sme opísali náš projekt, definovali sme si ciele, ktoré by chceli naplniť na konci zimného semestra a vízie, ktoré by sme chceli realizovať ďalej v letnom semestri. Tento dokument opisuje aj navrhovanú architektúru systému. Určili sme si metodiky, ktoré používame pri manažmente projektu, tiež sú bližšie popísané v kapitole 6. Na metodikách spracoval každý člen tímu.

Implementovali sme základný logger a ukladanie dát z myši na webe. Pre udalosti z myši sme zachytávali súradnice (x, y), typ udalosti a časovú známku. Nastavili sme aj databázový server, kde sme ukladali testovacie dáta.

Typ	Popis úlohy	Story points	Realizovatelia
Story	Logovanie surových dát z počítačovej myši	13	Ján Vnenčák , Kamil Janeček
Story	Ukladanie logovaných dát	13	Kamil Janeček , Matej Končál
Managment story	Big picture	13	Michaela Balážová , Michal Manak
Managment story	Metodiky	5	všetci členovia tímu

4.1.1. Retrospektíva šprintu č. 1

Čo sme urobili dobre?

- Stihli sme všetko, čo sme mali naplánované
- Rovnomerné rozdelenie úloh

Čo sme mohli urobiť lepšie?

- Vylepšiť časové plánovanie
- Vylepšiť remote komunikáciu o úlohách
- Zmierniť automatické notifikovanie o činnostiach (JIRA, Confluence)

Akcie

- Vyriešiť automatické notifikovanie - spoločné stretnutie a diskusia
- Pozrieť sa znova na metodiky, po komentároch

4.2. Šprint č. 2 (23.10.2018 - 06.11.2018)

V druhom šprinte sme prepojili vytvorený logger so serverom, teda logované dáta boli posiadané na priamo na server a ukladané do databázy. Vytvorili sme rozhranie loggeru pre možnosť logovania vlastných (custom) udalostí. Pomocou vytvoreného machine learning modul sme rozdelili používateľské dáta na jednotlivé segmenty a realizovali sme výpočet stĺpcov s rýchlostnými metrikami.

V tomto šprinte nám nastala situácia, kedy pri prezentácii demo ukážky PO nebol kompletne spokojný s výsledkom, chyba nastala v komunikácii a v nekompletnej špecifikácii. V tomto prípade sme postupovali podľa postupu, ktorý je opísaný v metodiky pre nástroj JIRA - teda prípad, kedy PO čiastočne akceptuje Story (odpočítanie nového odhadu chýbajúcej funkcionality).

Typ	Popis úlohy	Story points	Realizovatelia
Story	Pokročilé logovanie surových dát z počítačovej myši	2	Ján Vnenčák , Kamil Janeček
Story	Logovanie vlastných eventov	8	Ján Vnenčák , Kamil Janeček , Michal Manak , Matúš Kalafut
Story	Rozdelenie surových dát do časových intervalov a segmentov	12	Matej Končál , Michal Manak
Story	Výpočet rýchlostných stĺpcov zo surových dát	21 → 18	Michaela Balážová , Matej Končál , Michal Manak , Tomáš Jendrejčák
Story	Integrácia loggeru spolu so serverom	8	Matej Končál , Kamil Janeček , Matúš Kalafut , Ján Vnenčák
Task	Príprava jupyter notebooku pre demo ukážku	-	Matúš Kalafut , Matej Končál

4.2.1. Retrospektíva šprintu č. 2

Čo sme urobili dobre?

- Aj napriek študijnému voľnu sme dodržali pracovnú hygienu
- Pekne vypracovaná demo ukážka

Čo sme mohli urobiť lepšie?

- Zlepšiť komunikáciu s PO pri definovaní stories
- Vyhradiť viac času na plánovanie
- Zlepšiť spôsob spúšťania demo ukážky

- Zrýchliť ukončenie šprintu (stop diskusie po 5 minútach)

Akcie

- [Ján Vnenčák](#) Sprístupniť logger modul
- Stop diskusie po 5 minútach

4.3. Šprint č. 3 (06.11.2018 - 20.11.2018)

V treťom šprinte sme pridali možnosť sťahovania dát vo formáte .csv, zo stĺpcov s rýchlostnými metrikami sme extrahovali črty, ktoré budú ďalej použité na klasifikáciu. Vytvorili sme prvotný klasifikátor, ktorý bude binárne predikovať, či sa jedná o autentifikovaného používateľa. Pre ďalšiu fázu projektu sme vykonali analýzu prístupných senzorov v mobilných zariadeniach.

Typ	Popis úlohy	Story points	Realizovatelia
Story	Extrahovanie črt zo segmentov	21	Matej Končál , Tomáš Jendrejčák , Matúš Kalafut
Story	Nastavenie dát na jednotný formát	3	Michal Manak
Story	Výpočet rýchlostných stĺpcov zo surových dát	3 (21-18)	Michal Manak
Story	Generovanie dátovej sady zo získaných dát	8	Matej Končál , Matúš Kalafut , Kamil Janeček
Story	Autentifikácia používateľa	13	Matej Končál , Tomáš Jendrejčák , Matúš Kalafut
Managment story	Vytvorenie dokumentácie	8	Michaela Balážová
Task	Zmena implementácie demo ukážky	-	Ján Vnenčák
Task	Prehľad logovacích možností pre mobilné zariadenia	-	Ján Vnenčák

4.3.1. Retrospektíva šprintu č. 3

Čo sme urobili dobre?

- Implementovaná základná klasifikácia
- Dobrý code review

Čo sme mohli urobiť lepšie?

- Pripraviť sa na retrospektívu
- Stav v JIRE meniť pred začiatkom akcií
- Zlepšiť odozvu pri komunikácii ohľadom dokumentácie
- Stories by mohli mať svojich zodpovedných vedúcich

Akcie

- Pozrieť videá o retrospektíve
- Určiť stories vedúcich pri plánovaní
- Naškálovať vhodne story pointy

4.4. Šprint č. 4 (20.11.2018 - 04.12.2018)

V tomto šprinte sme pridali loggovanie udalostí aj z mobilného zariadenia, začali sme s vizualizáciou kriviek pre metriky FAR, FRR a EER. Spravili sme integráciu fáz spracovania dát do pipeline a škálovateľnosť modelu počtom používateľov.

Typ	Popis úlohy	Story points	Realizovatelia
Story	Vykreslenie kriviek pre FAR, FRR a EER	8	Tomáš Jendrejčák , Michal Manak
Story	Integrácia fáz spracovania dát do pipeline	13	Matej Končál Matúš Kalafut
Story	Škálovateľnosť modelu počtom používateľov	8	Michal Manak , Matej Končál
Story	Logovanie surových dát z mobilných zariadení	8	Ján Vnenčák , Kamil Janeček
Managment story	Vytvorenie dokumentácie	1	Michaela Balážová

4.4.1. Retrospektíva šprintu č. 4

Čo sme urobili dobre?

- Párové programovanie

Čo sme mohli urobiť lepšie?

- Robiť priebežne
- Interaktívnejšia komunikácia na stretnutiach s vedúcim

Akcie

- Pridať stand-up na Slacku v piatky

4.5. Šprint č. 5 (04.12.2018 - 18.12.2018)

Cieľom tohto šprintu je vytvoriť scenár pre demo, ktoré budeme implementovať v nasledujúcich mesiacoch. Budeme pracovať na delení dát do segmentov a na voľbe frekvencií pri logovaní mobilných zariadení.

Typ	Popis úlohy	Story points	Realizovatelia
Story	Možnosť vybrania frekvencie logovania z mobilných zariadení	8	Ján Vnencák
Story	Aplikácia preddefinovaných kritérií pre delenie surových dát na segmenty	5	Michal Manak
Managment story	Vytvorenie scenáru pre demo	8	všetci členovia tímu
Managment story	Vytvorenie finálnej dokumentácie	5	Michal Manak , Michaela Balážová , Ján Vnencák
Managment story	Prezentácia na predmet MTS	5	všetci členovia tímu

4.5.1. Retrospektíva šprintu č. 5

Čo sme urobili dobre?

- Mali sme stretnutie navyše

Čo sme mohli urobiť lepšie?

- Zčať skôr code review

Akcie

- Priradovať reviewerov na dokončené subtasky

4.6. Šprint č. x (18.12.2018 - 12.02.2019)

Tento šprint sa realizoval počas akademických prázdnin, jedným z jeho cieľov bolo premyslieť a pracovať na demo verzii aplikácie. Riešili sme aj technický dlh, čiže sme sa snažili refaktorovať jednotlivé moduly a upraviť ich štruktúru. Počas tohto šprintu sa nám vyskytli jednotlivé úlohy (napr. Potrebná akcia - obnova šifrovacieho certifikátu) , ktoré sme museli dodatočne pridať.

Tento šprint sme aj preto označili inak ako všetky ostatné, nakoľko sme jednotlivé úlohy neodhadovali, keďže trval netradične dlhé časové obdobie. Zrealizovali sme aj osobné stretnutie všetkých členov tímu, kde sme diskutovali najmä o postupnosti jednotlivých akcií vrámci demo verzie aplikácie.

Typ	Popis úlohy	Story points	Realizovatelia
Story	Zozbieraj dáta pre autentifikáciu používateľa	-	Kamil Janeček Ján Vnenčák Michal Manak
Task	Docker-compose - pridaj siete	-	Kamil Janeček
Task	Potrebná akcia - obnova šifrovacieho certifikátu	-	Kamil Janeček Matej Končál
Managment story	Technický dlh - server modul	-	Kamil Janeček
Managment story	Technický dlh - logger modul	-	Ján Vnenčák
Managment story	Technický dlh - machine learning modul	-	Kamil Janeček Matej Končál Michaela Balážová
Managment story	Balíkovanie modulov	-	Kamil Janeček Matej Končál
Managment story	Ukladanie dát pre autentifikáciu používateľa	-	Matej Končál Kamil Janeček Matúš Kalafut Michal Manak
Managment story	Vytvorenie scenáru pre demo	-	všetci členovia tímu

4.7. Šprint č. 6 (12.02.2019 - 26.02.2019)

Cieľom tohto šprintu bolo spustiť (ang. release) logger balík, spustiť serverovú aplikáciu, vylepšiť demo verziu aplikácie, ktorú sme si načrtli v predošlom šprinte a implementovať nové vizualizácie. Dané úlohy sme čiastočne splnili, úspešne sa nám podarilo spustiť logger. Pridali sme si [nojira] tag, ktorý nám pomohol efektívne pracovať na jednotlivých úlohách bez vytvorenia Jira úlohy. Na konci tohto šprintu sme v konečnom dôsledku dospeli k lepšej infraštruktúre. Podarilo sa nám zrealizovať aj plánované vizualizácie - DET, ROC krivky.

Typ	Popis úlohy	Story points	Realizovatelia
Story	Vykresliť ROC krivku	5	Tomáš Jendrejčák Michaela Balážová

Story	Vykresliť DET krivku	8	Tomáš Jendrejčák Michaela Balážová
Managment story	Balíkovanie modulov	3	Kamil Janeček Matej Končál
Managment story	Technický dlh - logger modul	8	Ján Vnenčák
Managment story	Ukladanie dát pre autentifikáciu používateľa	8	Matej Končál Kamil Janeček Matúš Kalafut Michal Manak
Managment story	Autentifikácia používateľa	13	Matej Končál Michal Manak
Managment story	Vytvorenie scenáru pre demo	3	všetci členovia tímu

4.7.1. Retrospektíva šprintu č. 6

Čo sme urobili dobre?

- Release loggera
- NoJira tag - zvýšenie efektivity
- Použiteľná infraštruktúra

Čo sme mohli urobiť lepšie?

- Zčať skôr code review

Akcie

- Párové programovanie (silno špecifické stories) - review navzájom (napr. ROC, DET krivky)

4.8. Šprint č. 7 (26.02.2019 - 12.03.2019)

V tomto šprinte sa pracovalo na vizualizácií, či už dát z myši alebo dokončovaní úlohy z predošlého šprintu (DET krivka). Začali sme logovať metadáta zo zariadenia a venovali sme sa aj fitrovaniu dát. V porovnaní s predošlým šprintom sa nám podarilo dokončiť všetko na čas a taktiež všetko bolo akceptované PO.

Typ	Popis úlohy	Story points	Realizovatelia
Story	Logovanie metadát zariadenia	3	Ján Vnenčák
Story	Vykresliť DET krivku	3	Michaela Balážová

Story	Existujúce odvodené stĺpce spraviť škálovateľné na počet osí	8	Tomáš Jendrejčák
Story	Vizualizovať dáta z myši	13	Michal Manak Matúš Kalafut
Story	Pokročilé filtrovanie dát	8	Kamil Janeček
Task	Možnosti pre unikátne ID zariadenia	-	Michal Manak Ján Vnenčák

4.8.1. Retrospektíva šprintu č. 7

Čo sme urobili dobre?

- Priebežná práca (dokonca aj demo)
- Všetko naplánované sme stihli

Čo sme mohli urobiť lepšie?

- Plánovať viac do budúcnosti (aj 2 šprinty)

4.9. Šprint č. 8 (12.03.2019 - 26.03.2019)

Tento šprint sa nám nepodarilo vhodne naplánovať, nakoľko sme počas práce na ňom zistili, že medzi úlohami boli dosť výrazné závislosti. Spôsobila to aj úloha refaktoru machine learning modulu, keďže bola dostatočne rozsiahla a všetky úlohy ohľadne tohto modulu boli na nej závislé. Jednu úlohu (Autentifikácia používateľa) sme zaradili do šprintu, ale nezačala sa na nej práca z časových dôvodov. Avšak aj napriek týmto problémom bola väčšina úloh načas a správne dokončená, čiže aj napriek komplikáciám celkovo šprint dopadol dobre.

Typ	Popis úlohy	Story points	Realizovatelia
Story	Vizualizovať dáta z mobilu	13	Matúš Kalafut
Story	Opraviť počítanie v stĺpcoch	-	Tomáš Jendrejčák
Story	Mobilné dáta po spracovaní	8	Ján Vnenčák
Managment story	Refaktor machine learning modulu	8	Michal Manak Matej Končál Matúš Kalafut
Managment story	Zjednotiť návratové hodnoty v machine learning module	5	Matej Končál Tomáš Jendrejčák
Managment story	Autentifikácia používateľa	13	Matej Končál Michal Manak

Management story	Vytvorenie scenáru pre demo	3	Michaela Balážová Ján Vnenčák Michal Manak
Task	Aktualizovať Jira metodológiu	-	Kamil Janeček
Task	Prehľad ohľadne možností vizualizácie pre 3D dáta z leap motion	-	Kamil Janeček

4.9.1. Retrospektíva šprintu č. 8

Čo sme urobili dobre?

- aj vzhľadom na veľké závislosti medzi úlohami dopadol šprint dobre

Čo sme mohli urobiť lepšie?

- zle sme si naplánovali šprint (veľké závislosti)
 - mysleli sme si, že to bude zvládnuteľné, ale nebolo
- jedna úloha nebola ani začatá (Autentifikácia používateľa)
- stále hľadáme staré demo verzie, ktoré sme prezentovali - lebo sa neukladajú
- niektorí členovia tímu stále nemali očakávané nastavený config toolov ([Tomáš Jendrejčák](#))

Akcie

- Ukladať všetky demo verzie na prezentovanie pri konkrétnych taskoch (napr. v JIRA, prípadne vo WIKI)
- Skontrolovať konfigurácie gitu (autocrf) a dopísať do metodiky ([Kamil Janeček](#))

4.10. Šprint č. 9 (26.03.2019 - 09.04.2019)

V tomto šprinte sa nám podarilo pripraviť sa na pipeline pre ďalší šprint, čo bol výrazný krok. Ďalej sme riešili najmä vizualizácie, kde aj napriek poučeniu z minulého šprintu sme našli počas práce závislosti medzi jednotlivými úlohami. Plusom bolo, že sa nám aj napriek problémom podarilo všetko úspešne dokončiť. A tiež sa nám podarilo realizovať obidve akcie z minulého šprintu - aktualizovanie metodiky a vytvorenie úložiska na WIKI pre ukladanie vytvorených demo verzií alebo prezentačných jupyter notebookov.

Typ	Popis úlohy	Story points	Realizovatelia
Story	Zvýrazniť segmenty v surových dátach	13	Matúš Kalafut
Story	Pridať vlastné udalosti do vizualizovaných dát	5	Michal Manak

Story	Integrácia fáz spracovania dát do pipeline	1	Kamil Janeček Matej Končál
Story	Zobraziť metriky na bodoch krivky	3	Tomáš Jendrejčák
Story	Zobraziť bod EER na krivkách	1	Tomáš Jendrejčák
Story	Pridať obrázky pozadia do vizualizácií	2	Michaela Balážová
Story	Vizualizovanie vlastných stĺpcov	5	Michaela Balážová
Managment story	Vytvoriť sklearn kompatibilné transformátory pre behlearn funkcie	13	Matej Končál
Task	Inštalovať Jupyter notebook na TP-VM	-	Kamil Janeček Matej Končál
Task	Negeneruje sa dokumentácia ku machine learning modulu	-	Matej Končál
Bug	Gunicorn worker vyprší čas keď sa snaží stiahnuť príliš veľkú dátovú sadu	-	Kamil Janeček Matej Končál

4.10.1. Retrospektíva šprintu č. 9

Čo sme urobili dobre?

- splnili sme akcie definované v predchádzajúcej retrospektíve
- sme pripravení na pipeline

Čo sme mohli urobiť lepšie?

- nekonzistencie v názvoch v rôznych častiach modulov
- mali sme veľké závislosti medzi rôznymi story (žiadne zlepšenie od minulého šprintu)
- nerovnomerne rozdelená práca

4.11. Šprint č. 10 (09.04.2019 - 23.04.2019)

V tomto šprinte sa nám úspešne podarilo dokončiť pipeline. Potrebovali sme zjednotiť názvy stĺpcov na celej platforme, čiže túto zmenu sme implementovali hneď na najbližšom stretnutí, keďže všetky úlohy záviseli od tejto jednej. Začali sme refaktor vizualizačných implementácií (dáta z myši, z mobilu, vlastné stĺpce...), keďže sme pridávali viacero rôznych funkcionalít a bolo potrebné mať celistvejšiu a prehľadnejšiu implementáciu. Tu sme zistili, že jednotlivé podúlohy medzi sebou súvisia a viacerí členovia tímu museli čakať na iných, kým sa dokončila úloha. Aj z tohto dôvodu sa nestihli dokončiť všetky podúlohy a časť musela byť prenesená do ďalšieho šprintu.

Typ	Popis úlohy	Story points	Realizovatelia
Story	Rozdeľte vzorky na jedného používateľa a nie na označenie triedy	8	Ján Vnenčák
Managment story	Refaktor aktuálnej vizualizačnej implementácie	21	Kamil Janeček Matej Končál Tomáš Jendrejčák Matúš Kalafut Michaela Balážová
Managment story	Autentifikácia používateľa	8	Matej Končál Michal Manak
Task	Zjednotiť pomenovanie stĺpcov na celej platforme	-	Matej Končál Ján Vnenčák

4.11.1. Retrospektíva šprintu č. 10

Čo sme urobili dobre?

- Konečne sme vytvorili pipeline.
- Blocker task sme vyriešili hneď na najbližšom stretnutí presne podľa plánu.

Čo sme mohli urobiť lepšie?

- Nemuseli by sme mať závislosti medzi subtaskami.
- Niektorí polovicu šprintu nič nerobil - kvôli čakaniu na ostatné úlohy.
- Riešiť veci viac verejne - niektorí členovia tímu nevedia o problémoch iných členov. (cc [Tomáš Jendrejčák](#))

Akcie

- Určiť si deadlines pre subtasky.
- Využiť verejné kanály na slacku, nie súkromné správy.

4.12. Šprint č. 11 (23.04.2019 - 07.05.2019)

V tomto šprinte sme riešili testovanie používateľom, ktoré sme pripravovali a neskôr aj realizovali. Taktiež sme sa venovali aj úlohám ohľadne refaktoru vizualizačnej implementácie, ktoré sme si preniesli z predošlého šprintu. A finalizovali sme dokumentáciu, či už generovanú z implementácie alebo formálnu k odovzdaniu projektu. Podarilo sa nám dokončiť všetky úlohy na čas a taktiež sme aplikovali akciu z minulého šprintu a nastavili sme si deadline pre úlohy spojené s testovaním, ktorý sme dodržali.

Typ	Popis úlohy	Story points	Realizovatelia
-----	-------------	--------------	----------------

Story	Pridať obrázok pozadia do vizualizácií	3	Michaela Balážová Matej Končál
Managment story	Refaktor aktuálnej vizualizačnej implementácie	13	Michaela Balážová Matej Končál
Managment story	Finalizovanie dokumentácie pre všetky moduly	13	Michaela Balážová
Managment story	Prípraviť testovanie používateľom	13	Matej Končál Matúš Kalafut Michal Manak Tomáš Jendrejčák
Managment story	Prípraviť demo server	5	Matej Končál Kamil Janeček

4.12.1. Retrospektíva šprintu č. 11

Čo sme urobili dobre?

- Stihli sme deadline definovaný na prvú polovicu šprintu.

Čo sme mohli urobiť lepšie?

- Nedostupnosť na tímovom stretnutí ohlásiť hlavne tímu.

5. Globálna retrospektíva

Absolvovali sme 11 šprintov za uplynulé 2 semestre. Náš projekt poskytuje funkčný logger pre udalosti z počítačovej myši a taktiež pre udalosti z mobilného zariadenia, ktoré ukladá do databázy na server. Poskytuje tiež predspracovanie dát a vykonáva jednoduchú klasifikáciu. Machine learning modul poskytuje aj vizualizačné funkcie pre zobrazenie dát z počítačovej myši a taktiež dát z mobilného zariadenia.

Podarilo sa nám splniť niektoré z cieľov, ktoré sme si stanovili na konci zimného semestra, aj keď sme sa vybrali trochu iným smerom, ako sme prvotne plánovali. Rozvinuli sme najmä vizualizácie a pracovali sme na pipeline.

Identifikovali sme zopár problémov počas projektu, niektoré z nich sa nám podarilo vyriešiť.

- Zvyčajne sme mali problém s časom na stretnutiach, kde sme uzatvárali aj otvárali nový šprint. Tento problém sme vyriešili priebežným plánovaním na stretnutiach v strede šprintu. Tento krok nám značne pomohol pri lepšej organizácii a efektívite na stretnutiach.
- Zabúdalo sa na menenie stavov stories v Jire. Tento problém sme vyriešili pridelením zodpovedných vedúcich na úlohy pri otváraní nového šprintu. Tento spôsob sa nám v letnom semestri osvedčil a úlohy v Jire sme udržiavali v aktuálnom stave.
- Veľké množstvo automatizovaných správ na základe diskusie sme vyriešili každý individuálne podľa potreby.
- Zaviedli sme pripomienky cez nástroj Slack pre experty Jiry na webové sídlo, na ktoré sa predtým zabúdalo.
- Problémy, ktoré zatiaľ neboli jasne vyriešené sú ohľadne retrospektív šprintov, kde máme problém so zbieraním dát od členov tímu. S týmto problémom sme pracovali a čím ďalej sa zlepšujeme.

6. Metodiky

6.1. Metodika pre prehliadku zdrojového kódu

(Tomáš Jendrejčák, Matej Končál)

V tíme nemáme dopredu určených reviewerov. Dokončené tasky čakajúce na review sa nachádzajú v stave READY FOR REVIEW. Člen tímu, ktorý svoj task už dokončil, alebo na ňom práve nepracuje, si môže tento task vziať na review presunutím tasku do stavu IN REVIEW. V prípade, že reviewer počas code review neobjavil žiadne nedostatky, presunie task do stavu DONE. V opačnom prípade reviewer pridá komentár k chybnjej časti kódu. Vývojár buď zapracuje požadované zmeny alebo reaguje na komentár. Tento proces sa iteruje, kým reviewer neakceptuje kód.

6.1.1. Prostredie

Code review sa vykonáva v prostredí [GitLabu](#). V GitLabe si reviewer zobrazí otvorené merge requesty a zvolí merge request pre reviewovanú story a zobrazí zmeny. V prípade potreby pridania komentára pridáva komentár priamo k príslušnému riadku zdrojového kódu. Ak boli všetky pripomienky ku kódu zapracované, reviewer schváli merge request.

6.2. Metodika pre písanie zdrojového kódu

(Kamil Janeček, Ján Vnenčák)

Pri vyvíjaní nášho produktu využívame niekoľko programovacích jazykov. Konkrétne sa jedná o:

- Python
- Javascript

6.2.1. Všeobecné konvencie

- Zdrojový kód píšeme v angličtine
- Komentáre píšeme v angličtine
- Kód musí byť pokrytý jednotkovými testami. Za tieto testy zodpovedá dodávateľ funkcionality (osoba, ktorá otvorila merge request).

6.2.2. Konvencie pre Python

- Všetok kód je formátovaný podľa [PEP 8](#)
 - tento stav je vynútený pomocou nástroja [flake8](#), ktorý sa spúšťa pri každom commite do git repozitára. Ak táto kontrola neskončí úspešne nebude možné tento kód prijať.

- Dodržiavame best-practices pre použité knižnice a frameworky ([Flask](#), [pytest](#))
- V prípade potreby rozdelenia riadku s binárnym operátorom (napr. +, -) sa tento riadok delí pred operátorom (operátor sa presúva na ďalší riadok)

6.2.3. Konvencie pre Javascript

- Dodržiavame konvencie písania kódu podľa [Google štandardu](#)
 - na kontrolu konvencií používame nástroj [eslint](#)
 - pri commite do git repozitára sa v rámci pipeline je spustená automatická kontrola dohodnutých konvencií

6.3. Metodika pre verziovanie zdrojového kódu

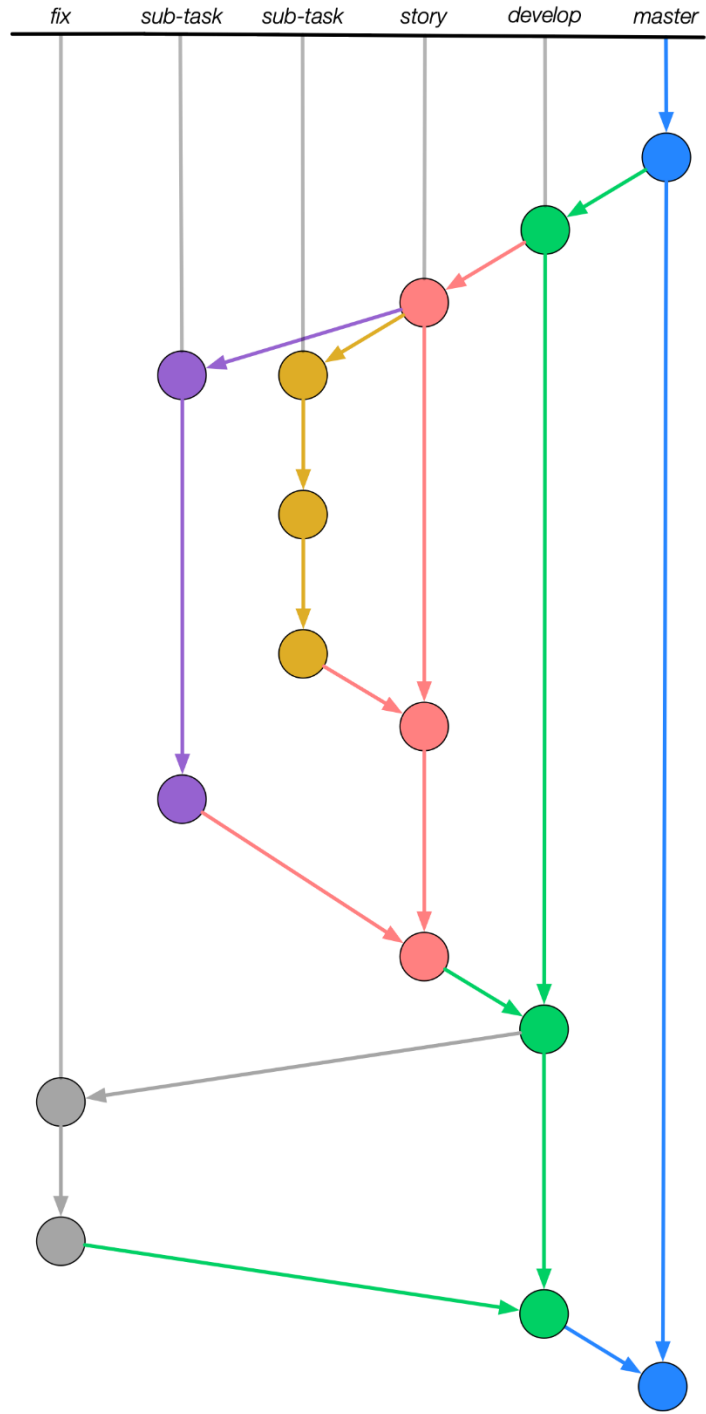
(*Michal Maňak, Matúš Kalafut*)

Na verziovanie používame nástroj git. Používame 4 úrovne vetiev - *master*, *develop*, *story*, *sub-task*.

Každý repozitár obsahuje 2 hlavné vetvy - *master* a *develop*. *Master* vetva predstavuje produkt v stave po šprinte. Vetva *develop* slúži na pridávanie funkcionality z user stories počas šprintu. Po uzavretí šprintu sa mergeje do *master* vetvy.

Každá *story* sa vyvíja vo svojej vetve, odvodenej z *develop* vetvy. *Sub-tasky* sa vyvíjajú v osobitnej vetve odvodenej od danej *story* vetvy. *Story* a *sub-task* branche pomenúvame podľa labelu story alebo sub-tasku z Jiry a jeho názvu (napr. BEL-21 - názov).

Commit správy musia obsahovať label sub-tasku na začiatku (príklad - BEL-44 - basic mouse events logging works). Po dokončení celého sub-tasku sa otvorí merge request do *story* branche. Vtedy sa vykonáva code review na merge requeste. Následne sa *sub-task* branch squashne do jedného commitu a mergeje do *story* vetvy. Ak je celá funkcionality v rámci *story* vetvy hotová, mergeje sa do *develop* vetvy.



Obr. č. 1 : Grafické znázornenie vytvoreného flow-u

6.4. Metodika pre nástroj JIRA

(Matúš Kalafut)

V našom projekte používame nástroj JIRA pre agilné zaznamenávanie úloh vykonávaných na projekte. Tento nástroj priamo umožňuje vytváranie backlogu, user stories a šprintov.

6.4.1. Úlohy

Úlohy sú usporiadané v 3-úrovňovej hierarchii:

- epic story
- story
- sub-task

Epic story - najväčši celok, ktorý zastrešuje stories tematicky patriace pod danú epic story.

User story - jedna z vlastností, ktorú má výsledný produkt obsahovať. Definuje product owner.

Managment story - slúži pre interné záležitosti v tíme a nesúvisí priamo s produktom.

Sub-task - konkrétna úloha, ktorú je potrebné splniť pre naplnenie user alebo managment story.

6.4.2. Roly

Riešiteľ

- tvorí riešenie úlohy, ktorú má pridelenú.

Watcher

- osoba, ktorá je akýmkoľvek spôsobom zapojená do riešenia úlohy alebo sa úloha týka tej časti vývoja, na ktorej spolupracujú.

6.4.3. Vytváranie úloh

Práca, ktorá je vykonávaná na projekte, musí mať vytvorený príslušný sub-task. Takto dosiahneme prehľad o celkovej práci a práci jednotlivých členov tímu.

Pri vytváraní sub-tasku je potrebné do popisu zadať popis úlohy, zdroje, ktoré by mohli pomôcť pri riešení a postrehy jednotlivých členov tímu. Ďalej sa pre story zadefinujú akceptačné kritéria, aby riešiteľ aj následne recenzent presne vedeli, čo má daná úloha spĺňať. Sub-tasku sa nastaví priorita, pridajú sa prílohy, napríklad poznámky k sub-tasku zo stretnutia, na ktorom sa daný sub-task rozoberal. Sub-tasku sa priradí šprint, do ktorého patrí a prípadne sa priradí závislosť na iný sub-task.

K úlohám je potrebné priradiť watchers, aby boli kompetentné osoby upozorňované na všetku aktivitu týkajúcu sa sub-tasku.

6.4.4. Stavy

Story aj sub-task prechádza tými istými stavmi, avšak pre každý typ to znamená niečo málo odlišné.

Story má vždy jeden zo 6 možných stavov:

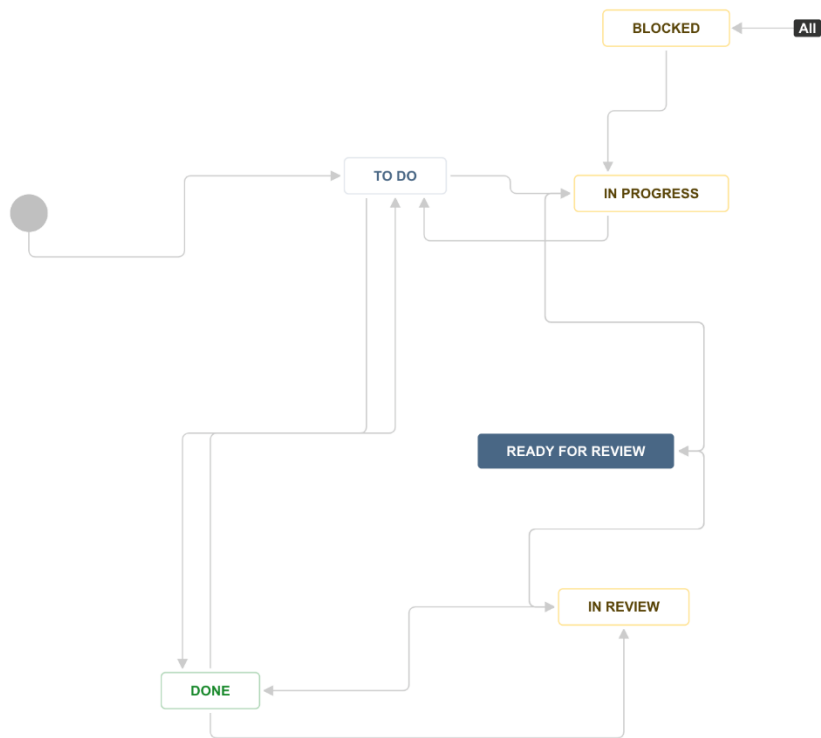
- TO DO - úloha, ktorú je potrebné vyriešiť a nezačala práca na nej. V tejto fáze sa vytvárajú sub-tasks, ktoré predstavujú menšiu granularitu práce na *story*.
- IN PROGRESS - na úlohe sa aktuálne pracuje, sú vytvorené *sub-tasks*, ktoré je potrebné realizovať v rámci danej *story*. *Sub-tasks* môžu byť priradené rôznym osobám.
- READY FOR REVIEW - bola dokončená práca na tejto úlohe, dokončili sa všetky *sub-tasks*, ktoré boli vytvorené k *story*. Úloha čaká na kontrolu.
- IN REVIEW - úloha prebieha kontrolou správnosti, väčšinou formou demo ukážky, ktoré sa prezentuje Product Ownerovi (PO).
- DONE - úloha bola skontrolovaná PO a spĺňa jeho kritériá.
- BLOCKED - do tohto stavu dáme úlohu, na ktorej nevieme pracovať, pretože nie je dokončená iná úloha, od ktorej je závislá.

Sub-task má vždy jeden zo 6 možných stavov:

- TO DO - úloha, ktorú je potrebné vyriešiť a nikto na nej práve nepracuje.
- IN PROGRESS - úlohu si prebrala konkrétna osoba a pracuje na nej.
- READY FOR REVIEW - osoba, zodpovedná za úlohu dokončila svoju prácu. Úloha čaká na kontrolu.
- IN REVIEW - úloha prebieha kontrolou správnosti a či spĺňa definition of done.
- DONE - úloha bola skontrolovaná a spĺňa definition of done.
- BLOCKED - do tohto stavu dáme úlohu, na ktorej nevieme pracovať, pretože nie je dokončená iná úloha, od ktorej je závislá.

Na obrázku č. 1 je znázornený proces, ako sa môžu jednotlivé stavy medzi sebou meniť.

Do stavu BLOCKED môžu prejsť všetky úlohy zo všetkých stavov.



Obr. č. 2 : Grafické znázornenie stavov a prechodov medzi nimi

6.4.5 Akceptovanie user story Product ownerom

1. Tím vytvorí demo s požadovanou funkcionalitou, ktorá bola špecifikovaná v *story*.
2. Product owner (PO) presunie *story* zo stavu READY FOR REVIEW na stav IN REVIEW pred odskúšaním demo ukážky.
3. PO odskúša demo.
4. Rozhodnutie PO:
 1. Ak je demo v poriadku a PO nemá žiadne výhrady, presunie ho do stavu DONE a táto *story* sa považuje za vyriešenú.
 2. Ak časť funkcionality nie je v poriadku:
 1. PO presunie aktuálnu *story* do stavu DONE
 2. Tím odhadne chýbajúcu funkcionalitu *story pointami* a odpočíta nové *story points* od pôvodného odhadu.
 3. Chýbajúcu časť pridá s novým odhadom do ďalšieho šprintu.
 3. Ak demo nebolo vôbec vyhovujúce, PO vráti tímu *story* na prerobenie (z IN REVIEW sa potrebujeme chceme dostať do stavu TO DO):
 1. Zo stavu IN REVIEW sa pomocou "Stop QA" dostaneme naspäť do stavu READY FOR REVIEW.
 2. Zo stavu READY FOR REVIEW sa pomocou "Not Completed" dostaneme do stavu IN PROGRESS.

3. Zo stavu IN PROGRESS sa pomocou "Stop Work" dostaneme do stavu TO DO.

6.5. Metodika ku dokumentácií

(*Michaela Balážová*)

Dokumentáciu priebežne tvoríme v nástroji Confluence, kde sú zoskupené všetky vytvorené dokumenty k projektu. Sú rozčlenené hierarchicky podľa kategórií a každá kategória obsahuje články s výstižným nadpisom, ktoré už konkrétne opisujú daný problém. Do tohto priestoru má prístup celý tím, spolu s vedúcim projektu. Dokumentáciu tvorí najmä určený dokumentarista v tíme, avšak vždy na viacerých častiach spolupracujú viacerí členovia tímu.

Pri každom priebežnom kontrolnom bode tieto dokumenty zoskupíme, upravíme ich do požadovanej formy a zverejníme ich na našom tímovom webe na adrese - <https://team04-18.studenti.fiit.stuba.sk/>, vo formáte pdf.

6.5.1. Formálna úprava

Nasledovné pravidlá platia pre formálne dokumenty, ktoré sa objavajú aj na našom webovom sídle. Zoznam týchto dokumentov je vypísaný nižšie v podkapitole [Dokumenty](#).

Štruktúra dokumentu

Každý dokument obsahuje:

- úvodnú stranu
- obsah dokumentu
- úvod - obsahuje bližší popis dokumentu
- samotný text

Úvodná strana

Úvodná strana obsahuje:

- názov univerzity, fakulty a jej adresu
- názov dokumentu
- názov tímu
- vedúceho projektu
- zoznam členov tímu (mená zoradené podľa abecedy)
- dátum poslednej úpravy

Obrázky a tabuľky

Každý obrázok alebo tabuľka majú popis vo formáte - napr. "*Obr. č. 1 - popis obrázka*" alebo "*Tabuľka č. 1 - popis tabuľky*". Tento text je zvýraznený kurzívou.

Číslovanie strán

Každá strana v dokumente je číslovaná okrem úvodnej strany.

6.5.2. Jazyk

Dokumentácia je napísaná v slovenskom jazyku. Niektoré dokumenty budú v anglickom jazyku, aby boli prístupné širokej verejnosti.

6.5.3. Tvorba retrospektívy šprintu

Na konci každého šprintu tvoríme dokument, obsahujúci retrospektívu šprintu.

- názov šprintu - tvar : Retrospektíva (x. šprint)
- dátum trvania šprintu
- účastníci
- čo sme urobili dobre?
- čo sme mohli urobiť lepšie?
- akcie pre ďalší šprint

6.5.4. Dokumenty

- inštalačná príručka
- používateľská príručka
- dokumentácia k riadeniu
 - metodiky
- dokumentácia k inžinierskemu dielu (aj v anglickom jazyku)
 - big picture
 - technická dokumentácia

6.5.5. Technická dokumentácia

Technická dokumentácia sa tvorí priebežne počas implementácie jednotlivých modulov spôsobom vhodným pre danú platformu (napr. Python Docstring, OpenApi). Následne sa tieto dokumentácie spracujú špeciálnymi nástrojmi, ktorých výsledkom je verejne dostupná webová podoba. Tieto dokumentácie sú dostupné na nasledovných miestach:

- [server modul](#)
- [logger modul](#)
- [machine learning modul](#)

6.6. Metodika Definition of ready

(Michal Maňak, Tomáš Jendrejčák)

- akceptačné kritéria musia byť definované a overiteľné
- story musí mať dostatočný opis
 - tím si musí byť istý svojou schopnosťou splniť story
 - poznať oblasť a problém
 - poznať technológie, ktoré budú použité, prípadne mať priestor pre ich naštudovanie
 - tím musí mať dostatočné časové kapacity pre splnenie úlohy
 - mala by obsahovať aspoň jeden scenár využitia funkcionality
- story musí byť odhadnutá story pointami
- story musí mať veľkosť primeranú na to, aby sa dala vykonať v jednom šprinte

6.7. Metodika Definition of done

(Michal Maňak, Tomáš Jendrejčák)

user story

- všetky sub-tasky danej story musia byť dokončené
- akceptačné kritéria musia byť splnené
- story schválená product ownerom
 - po predvedení demo ukážky a prejdení scenárov

managment story

- všetky sub-tasky musia byť dokončené
- dokončené všetky úlohy, ktoré boli definované v popise
- musí sa nachádzať v stave DONE

epic story

- všetky user stories, ktoré boli zahrnuté v epic story:
 - sú dokončené
 - schválené Product ownerom

6.8. Metodika plánovania

(Michaela Balážová, Michal Maňak)

Na našom projekte fungujeme pomocou Scrumu, ktorý máme upravený pre fungovanie na predmete Tímový projekt. Realizujeme 2-týždňové šprinty.

Stretnutia s product ownerom realizujeme každý utorok počas 3 hodín. Na každom stretnutí robíme "daily stand-up" (avšak na týždennej báze), kde diskutujeme ako každý člen tímu od posledného stretnutia pokročil a prípadne aké má problémy. Na tomto stretnutí otvárame aj uzatvárame šprint.

Čo sa týka plánovania šprintu, na začiatku šprintu si vyberieme *stories*, ktoré budeme realizovať v danom šprinte. Každá *story*, ktorá sa zahrnie do šprintu musí mať zodpovednú osobu, ktorá je zodpovedná za túto *story* (vytváranie *sub-taskov*, ich pridelenie...). V tomto kroku je potrebné vedieť aj ktoré *stories* sú od seba závislé, aby sme vedeli rozumne rozdeľovať prácu.

Jednotlivé *story* pri začínaní šprintu odhadujeme pomocou *story points*. Zvyčajne do jedného šprintu zahrnieme úlohy, ktoré majú v súčte 40-50 *story points*. Do šprintu sa môže dostať aj dostatočne veľká úloha, v našom ponímaní je to napr. 21 *story points*. Pri uzatváraní šprintu realizujeme retrospektívu šprintu, kde prebieha diskusia o tom, čo sme spravili dobre alebo zle v aktuálnom šprinte a plánujeme ďalšie akcie. Backlog priebežne upravujeme (zvyčajne na stretnutí v strede šprintu alebo pred začatím ďalšieho šprintu) - zoradíme ho podľa priority, pridávame *stories* a bližšie špecifikujeme nejasnosti s PO.

Tímové 3-hodinové stretnutia realizujeme každý pondelok, kde musí byť prítomný každý člen tímu. Toto stretnutie využívame najmä na riešenie spoločných problémov a diskusiu v tíme. Každý člen tímu pracuje na pridelených úlohách samostatne mimo tohto časového priestoru.

6.9. Metodika komunikácie

(*Michaela Balážová*)

Tím komunikuje pomocou nástroja Slack, kde má každý používateľ nastavené upozornenia podľa vlastného uváženia.

6.9.1. Vytvorenie nového kanálu

Pokiaľ sa deteguje oblasť, pre ktorú je potrebné vytvoriť nový komunikačný kanál, používateľ ho vytvorí. Pri inicializácii nového kanálu je potrebné ho správne a vecne pomenovať, prípadne sa môže pridať popis tohto kanálu. Kanál môže byť buď verejný, ktorý vidia všetci prizvaní používatelia v pracovnom prostredí (angl. workspace) alebo môže byť privátny, kde sa pozvú len vybraní členovia.

6.9.2. Existujúce kanály komunikácie

Komunikáciu máme rozdelenú medzi rôzne kanály (angl. channels). Každý kanál má svoj vlastný účel a komunikácia v ňom je tematická podľa názvu a účelu.

- logger-web
- ml-module
- server

Každý implementačný modul v našom produkte má vlastný kanál, kde sa posielajú správy z nástroja Gitlab a komunikuje sa ohľadne daného modulu a riešených problémov.

- Jira

Kanál Jira slúži najmä na sledovanie aktivity v nástroji Jira. Posielajú sa tu správy ohľadne vytvorenia úloh, zmeny stavu alebo nových komentárov k úlohám.

- webpage

V tomto kanáli sa generujú správy z nástroja Gitlab a pridávajú sa súbory, ktoré je potrebné pridať na naše webové sídlo (retrospektívy, dokumentácia, export z Jiry).

- random
- general
- stretnutia

Kanál *random* sa používa pre voľnú komunikáciu medzi členmi tímu. Pre všeobecné správy, ktoré nezapadajú do žiadneho z kanálov existuje kanál *general*. Kanál *stretnutia* sa využíva pre poznámky a obrázky zo spoločných spoločných stretnutí tímu. V prípade stand-upu, ktorý sa nemôže udiť fyzicky na fakulte, realizuje sa práve v tomto kanáli.

7. Export evidencie úloh

Export evidencie úloh z JIRY je poskytnutý na našom webovom sídle.

8. Webové sídlo tímu

<https://team04-18.studenti.fiit.stuba.sk/>