

## Príloha B: Zápisy zo stretnutí

---

### 1 Prvé stretnutie

- Stretnutie otvára Robo a oznamuje nám, že budúci týždeň bude musieť byť stretnutie v iný deň, pretože bude odcestovaný. Skúsi, či si ešte pamätá naše mená.
- Robo chce vidieť náš plagát, Martin Svrček mu ho ukáže na notebooku a Robo je spokojný, hovorí, že môže byť a aj názov tímu Focus sa mu páči.
- Začínajú sa riešiť organizačné veci, Robo navrhuje nech máme nejaké centrálné úložisko, kde si budeme ukladať dokumenty. Šimon odpovedá, že už sme si spravili Google drive. Robo povie, že môže byť a nech ho tam pridáme.
- Robo sa pýta ako okrem emailu medzi sebou komunikujeme, pretože to nestačí. Vie, že máme aj Facebook-skupinu ale radšej by bol keby to bolo niečo ako instant messaging, či niečo také nepoznáme z práce? Šimon navrhuje heapchat.
- Robo nám začína vysvetľovať jednotlivé funkcie rolí:
- Naš tím by mal mať obsadené nasledujúce role: Manažér komunikácie, Manažér dokumentácie, Vedúci tímu, Zástupca tímu, Manažér kvality: 2 ľudia môžu byť, jeden bude dozerat' nato ako sa budú robiť testy – pomocou tfs-ka, druhý bude mať na starosti code-review, to znamená, že každý kód budú vidieť dvaja ľudia a až potom pôjde na schválenie k Robovi. Tento proces by sa mal zachytiť v nejakej metodike manažérom kvality.  
Manažér pre code versioning – napríklad v Gite, napísať aj postup ako to bude fungovať.  
Tieto metodiky môžeme využiť potom aj v manažmente.  
Manažér plánovania: v podstate sa plánovanie robí spoločne, najprv sa stanovia featury, tie sa rozbijú na user stories a tie na tasky. Plánovanie sa robí pomocou metódy pokru, kedy spoločne prídeme nato koľko story pointov obsahuje task.
- Manažér rizik a rozsahu – bude robiť aj manažér plánovania.
- Martin Svrček sa opýtal, kto sú naši zákazníci? Robo odpovedal, že máme možnosti:
  - možnosť, že to budú členovia fakulty - napríklad profesorka Bieliková, Michal Barla. Preto sa Robo pýta, či máme záujem o stretnutie s nimi, kde by sme sa dohodli o našich a ich predstavách, aby sme nezačali robiť niečo, čo oni nechceli.
  - plus sú to ľudia, ktorí používajú ux-lab, nejakí diplomanti a mohli by používať náš produkt
  - Robova funkcia je product owner.
- **Rozdelenie rolí:**
  - Vedúci tímu – Petet Kiš
  - Zástupca – Matej Kucek
  - Manažér komunikácie – Peter Kiš
  - Manažér kvality a integrácie – kvalita testovania – Peter Kušnír
    - integrácia – Matej Kucek
    - Git- Šimon Váliček
    - code review – Marek Šulek

Manažér plánovania a rozsahu- Martin Svrček

Manažér dokumentácie - Viktória Lovasová

- Dohodnutie sa na stretnutí so zákazníkmi, či máme záujem? Súhlasíme, Peter Kiš napíše prvý mail.
- Robo sa pýta, či chceme ísť na TP cup? Dve strany treba napísať a poslať do 28.10.2014. Prezentácia bude na iit src – tam nejaké 4 strany po anglicky napísať. Hodnotenie bude na základe prezentácie a aj na základe výstupnej správy. Porota je z praxe a môžeme dostať spätnú väzbu. Záverom je, že sa ešte rozhodneme.
- Prichádza odborný praktikant Mišo a začína nám vysvetľovať projekt, na ktorý my nadväzujeme. Má tri komponenty: informačný systém, chrom-addon, firefox, dve zariadenia na sledovanie pohľadu. Server – obsahuje dve databázy: nosql ravenDB v c# písaná na framework .NET, podporuje JSONy tie sa posielajú, druhá je MSSQL: obsahuje dáta o používateľoch. Eyetracker posielá súradnice očí na server a takisto identifikuje element, ďalej aj focused application, kde sa nachádza kurzor, pracuje sa aj na tom kde sú kliky, takisto aj timestamps. Komunikácia je zabezpečená cez REST operácie. Zvonku je nedostupný gaze data service. Najvyššia úroveň:
  - Prepojenie cez api s aplikáciou ALEF:  
Alef bude komunikovať s našou aplikáciou, bude môcť robiť nejaké akcie: reagovať na základe pohľadu a myši a odporúčať. Toto je najdôležitejšie, lebo to je použiteľné, nahradíme ALEF dáme tam niečo iné a bude to fungovať.
  - Štatistiky a vizualizácia
  - Podpora priebehu samotného testovania  
Zadať pokyny, má nižšiu prioritu ako ostatné dve.

### **Prepojenie cez api s aplikáciou ALEF**

Prebieha diskusia o features. Autentifikácia – posielanie v requeste apikey. ALEF si musí uložiť apikey, aby ho mohol posielat'. Potrebne získať aké sú zdefinované objects of interests. Get pre aoi koľkokrát sa tam používateľ pozrel, ako dlho sa tam pozeral, nie len kde sa pozerá, momentálne, xpath tiež môže vracat' kam sa pozerá. Šimon to celé zrušil lebo to je zlé pre biznis model. Preto sa bude posielat' len Get fix count, fix time -> rel, absolut. Uložiť v alefe apikey treba spraviť, na jeho strane nič nie je. V requeste posielam apikye, project\_id, session\_id, user\_id.

Dohodli sme sa na nasledovných Features:

**Features:** Ukladanie Apikey v ALEFe

Vytvorenie projektu

Vytvorenie sedenia

Pridanie používateľov k projektu

Získavanie dat - aoi agregacie-pohľad,mys

Upozornenie/Prispôbenie obsahu(odporúčanie)

Štatistika/Vizualizácia

Robo navrhuje aj aby sme to dotiahli až na odporúčania v ALEFe.

### **Štatistiky a vizualizácia**

**Features:**

**Poskytnutie prehľadu** – koľko tam je toho nazbieraného, project, session, user,  
- počet, čas

## Počítanie agregácií – nejaké predrátavanie by tam malo byť

(Vysvetlenie): ak sa niekam pozeral 1 s tak mi vytvor anotáciu(značku) viem si nastaviť aj že 5 sekúnd napríklad dá sa to nastaviť cez rozhranie. Mám objekty A1,A2,A3,..... kam sa pozeral a koľko. Takže ak už robím agregácie tak z týchto anotácií. Takže aj pri prehľade sa dá z anotácií robiť aj agregácie. Priebežne aktualizovať anotácie, zvyšovať počet pozretí aby som mala predpripravené keď príde request. Teda mať tam nejaké countre .

Počet fixácií

Zmena v čase

Trvanie -> relativne

->abolutne

Zachytávanie obrazovky-

(Vysvetlenie): robiť cez screenshot? Alebo video? Alebo stiahnuť stránku obraz(Mirror)

Heatmapy

Graf- poradie kam používateľ šiel

Zoom

Analýza pozerania na video

Šimon navrhuje personalizovanú reklamu – Robo hovorí, že to je super point na prezentáciu do TP cupu, nie len na výučbový systém ale aj na reklamu a bol by tam aj biznis model lebo že aj minulý rok sa sťažovali, že neboli stavané tak, že majú víziu posunúť to ďalej.

Minimálne prezentovať aj keď nie realizovať.

Ďalej nasledovalo plánovanie úloh do najbližšieho šprintu. Nabudúce si povieme v akom stave sú úlohy, na konci je retrospektíva, čo by sme mali zmeniť, čo bolo dobre. Potom je plánovanie nového šprintu.

### Hotovo je keď:

Ten, kto implementuje hneď spraví aj testy, najprv by mal byť test a až potom implementácia, keď všetky testy prešli, kód spĺňa špecifikáciu, potom to preje k druhej osobe, ktorá bude zodpovedná za code review, dá mu feedback, ak je to ok tak to ide na kontrolu Robovi. Ako sa presne bude robiť code review si povieme nabudúce, podobne ako s gitom to je je tam strom kde viem pridávať komentáre potom do kódu. Peter Kiš sa pýta, či bude code review robiť stále ten istý alebo môžu byť aj viacerí? Robo hovorí, že to môže byť jeden, ktorý bude potom implementovať menej a je dobrý v .NETE alebo budú dvojčky a jeden implementovať a druhý code review robiť alebo pri každej úlohe sa určí kto robí review. Robo hovorí nech je úloh v šprinte menej ale aby sa to stihlo spraviť. Už po prvom týždni to má ísť ku code reviewerovi. K Robovi to môže ísť najneskôr na dokončení šprintu.

### Úlohy:

- ALEF- vytvoriť funkciu v rámci gaze api, ktorý si alef vypýta apikey a tá funkcia v api vráti len obyčajné ok, nejaká jednoduchá komunikácia so slovom ok, a v alefe sa zobrazí modálne okno ok. User story: ALEF chce komunikovať cez api.
- Rozdelenie úloh - tasky:

- ALEF: Forknúť si git repozitár nad našou vlastnou verziou alefu, Rozbehať alef, Volanie cez api, Zobrazenie výsledkov, Testy.
  - api: Vytvoriť api kľúč pre ALEF - cez rozhranie spraviť. Vytvoriť volanie Support: rozbehať a nastaviť con. integration cez tfs
  - stránka tímu- jednoduchá, statická, vždy tam treba dať zápisnicu a dokumenty a plán
  - Zápisnica – všetky veci majú byť v tfs, teda aj zápisnica a exportovať sa vždy bude zápisnica z tfs
  - Budúce stretnutie bude viesť Viktória Lovasová, pretože písala túto dokumentáciu, povieme si čo sme mali spraviť.
  - Štúdium – ako sa pracuje s gitom, .net, prostredie (VS), Ruby on Rails, pozrieť dokumentáciu z min roku ako je zdokumentované api a databáza. Tieto úlohy spraví každý. Každý, za čo je zodpovedný začať nízko úrovňovo písať metodiku. Testovanie, integrácia a git je dôležité.
- Šimon sa pýta kedy bude budúce stretnutie. Bude vo štvrtok. Presný čas nám ešte Robo dá vedieť. Dohadovanie, kto bude čo robiť: stránka – do tfska nahodiť kto je zodpovedný, za task je zodpovedný vždy jeden, stránku robí Martin Svrček a Peter Kušnir. Apikey spraviť pre ALEF- Šimon,
  - Api vytvoriť funkciu – Marek Šulek, codeReview: Peter Kiš.
  - Nastavovanie v tfs – Matej Kucek
  - ALEF – Viki a Robo , code review Martin Svrček
  - Ukončenie stretnutia.

## 2 Druhé stretnutie

- Viki spravila funkciu na preposielanie z ALEFu do facebooku, ale nefunguje zatiaľ funkcia na strane Gaze-u.
- Peter Kušnir spravil stránku. Robo -> Big like. Treba prerobiť plán aby tam boli všetky hlavné úlohy a aj plán na semester s termínmi odovzdávania.
- Martin - - Plánovanie sa robí na konci šprintu ako retrospektíva plus ohodnocovanie obtiažnosti úloh.
- Matej - - Tasky v TFS by sa mali podrobnejšie trošku opísať. Nastaviť si notifikácie o úlohách. Ukladať si aj čas, ktorý sme venovali robote. Napísať aj keď sme riešili bugy a nestihli sme spraviť nejaký task.
- Marek - - Komunikácia – ALEF posielala requesty, aby sa identifikovala musí použiť API key. Marekova aplikácia by zachytila volanie a vrátila hocijaký text. Funkcia už existuje – my potrebujeme aby sa prepojila s nami (vložíme tam URL alefu). Marekova API musí skontrolovať či ten prichádzajúci API kľúč je dobrý. Pri testoch musíme riešiť keď sa neprijme správny API kľúč. + API key v alefe hodiť do konfigúrujúceho nejakého.
- Peter Kiš – vyriešil komunikačný kanál, dohodol stretnutie so zákazníkmi.

- Šimon pomáhal Marekovi generovať API key --- ďalšie úlohy: nahodiť stránku na web + vyriešiť API.

### Ďalšie poznámky

Pozrieť si ako sa testuje v Railsoch aj v C# a RSpec

Po každom šprinte by mala byť každá funkcia otestovaná a nasadená.

Všetko čo robíme by sa malo nasadzovať na našu vetvu a tam to bude bežať a naša práca sa mernie s prácou praktikantov až na konci.

Metodika na integráciu – pozrieť TFS integration.

Treba mať do konca šprintu spravené aj testy a spraviť aj review kódu. Aby sa stihol code review tak musí byť funkcia hotová trošku skorej.

V TFS nový repozitár na ALEF treba vytvoriť.

### Code review:

- pozrieť aký kód bol pridaný vo Visual(prave tlačidlo COMPARE)
- to čo je spravené či to robí to čo má robiť korektnosť a správnosť
- či to je spravené rozumne (konštanty, veci v konfigurácii, premenné) , či by som bol ja spokojný s kódom, zatiaľ to robiť cez komentáre ku tomu čo je zle alebo čo treba zmeniť
- podpora pre codereview v TFS možno vo Visuallku , ale zatiaľ iba cez komentáre.
- Ten kto píše kód píše aj testy ale codereviewer následne kontroluje aj testy
- Robiť aj testy na výskyt bugu. – Toto môže robiť code reviewer a potom povie tomu kto to naprogramoval, že na tomto to môže padať.
- Každý by si mal pozrieť ako sa testuje.

### Budúce stretnutie – okolo 3 hodiny

- Rozdeliť si nové zadanie úloh pre tím do budúceho cvičenia podľa upravenej prezentácie z prednášky TP.
- Rozhodnúť sa ako to bude s TPcupom.
- Vyhodnotenie šprintu – či to čo sme spravili funguje a či je to správne.
- Povieme čo bolo dobre a čo treba zmeniť.
- Continual Integration – keď ja niečo pushnem tak sa automaticky rozozná zmena a automaticky spustí všetky testy a ak je všetko zelené tak sa to nasadí ale ak nie je všetko zelené tak sa to musí opraviť --- lebo sa môže stať, že našou prácou sme mohli rozbiť alebo ovplyvniť niečo iné a Continual Integration tieto veci odhalí.
- Ak sa odhalí bug v robote predchádzajúceho tímu (lebo ten nemá spravené testy) tak treba spraviť test ktorý bude simulovať tieto bugy.
- Treba robiť refactoring kódu – aj odreportovať, že sme to spravili.
- Vygenerovanie reportu úloh z TFS.
- Product Backlog – treba si ho dopĺňať popri práci – ale nezasahovať do sprint backlogu počas sprintu. Môžeme pridávať úlohy kedykoľvek do product backlogu.

## 3 Tretie stretnutie

8:30 dostavili sa všetci. Môžeme začať.

V TFSku môžeme mať viacero repozitárov kvôli ALEFu, dá sa meniť stav tasku commitmi (aj ho zavrieť). O detailoch ešte príde mail.

### **Zhodnotenie týždňa**

Viki: zavolala funkciu z API, ale nemá ešte hotové testy. Robo to kontroluje chmúravým pohľadom (nejak dlhšie to bežalo), ale nakoniec je spokojný.

Mato Svrček: robil code review Viki, kontroloval zdrojáky aj konfigurák, Robo má nejaké výhrady k názvu controlleru (do budúca funkcionalitu umiestňovať do libky nie do controllera).

Marek: vytvoril nový projekt (solution) na testovanie API, lebo starý kód nebol použiteľný v tomto smere. S pomocou Šimona to nejako spravili, aby to šlo zavolať. Code review -> Peťo Kiš (nebolo veľmi čo opravovať), testy sa nestihli.

Matej: venoval sa nastaveniu continuous integration (Šimon že to donastaví), spravil nastavenie notifikácií, čo sa týka reportov tak k tomu sa mu nepodarilo nič nájsť. Robo k tomu niečo našiel, tak sa na to treba pozrieť. Report -> kto čo robí, v akom stave sú úlohy a čo sa bude riešiť.

Peto Kušník: dokončenie stránky (plán), task: dorobiť features na vyššej úrovni (termíny odovzdávania, vysokoúrovňový pohľad na veci čo ideme robiť; momentálny stav nie je úplne vyhovujúci, plán je náš backlog, čiže len termíny a fičúrky)

Potrebuje dohodnúť framework na testovanie pre C# (Šimon navrhuje Nspec, Peťo Kušník Nunit), Rails je dohodnutý na Rspec.

Došiel Mišo, Šimon sa pýta ohľadom deployovania na server, oni to robia ručne; otázky na API: Šimon rozhodol, že je to nepoužiteľné tak ideme robiť API odznova. Čo sa týka stránky Šimon vybaví novú doménu v priebehu tohto týždňa, aby sa dali stránky odlišovať (stará a nová); chce tam nahodiť ftp.

Záver zo včera: hlavné veci pre tímak sú spraviť základné vizualizácie, začať zbierať viac dát (pohľad, myš, video); chystá sa projekt z UX labu, majú termín, aby bol sfunkčnený lab, možno v budúcnosti by sa dalo zohnať aj lepšie železo; ďalej sa bude riešiť integrácia s ALEFom, tiež boli rôzne nápady, posielalo by to aj nám aj klientovi (APIčku by bolo treba dorobiť server). Posledná vec je poskytovanie údajov učiteľom o tom, ktoré časti sú čítané apod. Mišo odchádza. Ak sa nájde niečo ďalšie, tak máme poslať mail.

Robo navrhol uzavrieť šprint. Každý týždeň by mal byť report a na konci šprintu by mala byť retrospektíva. Celkom zaujímavý spôsob ako to spraviť je, že povieme, čo z toho čo sme spravili už nerobiť, v čom pokračovať a čo začať robiť nové.

Začína Maťo Svrček. Navrhuje, aby sme ako tím lepšie pochopili, ako to celé funguje a navrhuje zintegrovat tam všetkých členov. Bolo dobré, že sa nejako začalo, a že sa aj niečo spravilo. Navrhuje analýzu kódu, aby sme sa ako členovia lepšie oboznámili s fungovaním kódu.

Viki: nerobili sa testy a malo by sa to zmeniť. Páčilo sa jej, že sme spolupracovali na taskoch. Peto Kiš: páčilo sa mu, že sme zorganizovali stretnutie so zákazníkmi, Šimon sa ukázal ako "ťahúň".

Maťo navrhuje ešte stanoviť si jasné metodiky.

Matej: páči sa mu atmosféra v tíme, a to že sme sa chválili :). Čo mu vadí, že by mohla byť väčšia aktivita na TFSku, písať si hodiny, čo sa týka vizuálka, tak viac sa oboznámiť s projektom.

Šimon navrhuje kódorské stretnutia, kde by sme spoločne kódili.

Peťo Kušnir: páčila sa mu tímová práca a navrhuje pracovať na reálnych, aj keď jednoduchších taskoch.

Marek: oceňuje prácu v tíme a pomoc od Šimona. Negatívne hodnotí tímovú komunikáciu cez HipChat a odporúča označovať ľudí v komentoch a rýchlejšiu spätnú väzbu. Ešte spomenul, že veľa času strávil na aktuálnom tasku a veľmi mu to nešlo. Navrhuje začať makat' na ľahších, ale reálnych veciach.

Šimon: ++ ľudia v tíme a spolupráca, navrhuje začať makat' na reálnych veciach.

Robo: ++ že sme dokázali pracovať spolu po stretnutí s prof. Bielikovou, ale máme sa vyvarovať problémom s termínmi (stránka). Treba identifikovať potencionálny problém, aby sme si vedeli vytvoriť časovú rezervu.

Robo navrhuje spraviť lean canvas pre náš tímový projekt do najbližšieho stretnutia a nezabudnúť spraviť veci ešte vrámci tohto šprintu (aktualizovať web,...).

## **Uzatvorenie šprintu**

Metodiky:

Vypracovať a hodiť do google docu kvôli pripomienkovaniu (má to byť tímová verzia, aby s ňou všetci súhlasili)

Založiť dokument (základ pre tímovú dokumentáciu) a na začiatok spraviť analýzu súčasného stavu (tiež ako google doc), aby sa rozobrala súčasná situácia a kvôli lepšiemu pochopeniu fungovania kódu (na úrovni tried)

Rozprava o paneloch: vytvorenie prezentácie o tom ako funguje produkt, ale aj o tíme, ako fungujeme v tíme, ako by to mohlo byť...

Roly:

Plánovanie pre tím a jednotlivých členov tímu: Maťo Svrček (metodika), inak prevažne spoločne v rámci tímu

Vyhodnocovanie plnenia plánu a návrh úprav: v priebehu a na konci šprintu priradiť taskom hodnoty, vytvoriť burndown chart => Maťo Svrček

Organizácia komunikácie v tíme: Peťo Kiš

Udržiavanie informácií o stave projektu: TFSko -> Matej, stránka aktuality -> Peťo Kušnir

Evidencia úloh: TFSko = Matej

Identifikovanie a riadenie rizík: odtrhnuté od reality pri tímakoch; identifikácia rizík, ohodnotenie pravdepodobnosťou, že nastane, cenou škody a navrhnuť riešenia, či im budem predchádzať, alebo ich budem ignorovať; vytvorenie nejakého plánovania akcií tímu (mal by sa na to niekto pozrieť a spísať to - prihlásil sa Maťo Svrček)

Zabezpečenie efektívneho znovupoužitia:

Monitorovanie prehliadky vytváraného výsledku: Robo posielal link na code review, Marek má pozrieť prezentáciu z prednášky o code review a vytvoriť projekt v Perconik-u už v tomto

šprinte, aby sa robil code review v tomto systéme. Aby to do budúceho týždňa bolo nastavené.

Vyhodnocovanie testov: každý píše testy sám

Identifikácia riadenia chýb v softvéri: ako zalogovať objavený bug, kto ho má riešiť, atd. Spíše Viki

Integrácia softvéru: viac menej hotové

Manažment verzií: Git, ako to používať, ako písať commit správy, ako sa dá automaticky uzatvárať úlohy a priradovať taskom commity

Požiadavky na zmenu: návrh na zmenu, ako vyhodnotiť požiadavku, ako ju zapracovať,...

TP cup:

národ je rozdelený, prebytočná robota za malý zisk.

Rozhoduje kvalita produktu, kvalita prezentácie, použiteľnosť v reálnom živote, biznis plán.

V zimnom semestri: prihláška, v letnom v marci 4 strany + 2 stranový abstrakt po anglicky, vytvoriť poster na IIT SRC a odprezentovať to, napísať na blog a predstaviť sa ako tím, spraviť video prezentujúce tím (60 sec), finálna prezentácia (povinná aj tak) + ak postúpime do finále tak 1 prezentácia navyše.

Čo nám to dá: skúsenosť súťažiť s niekým, podobné ako startup súťaže, prezentovať nápad a víziu.

Robo by bol za, máme sa rozhodnúť do večera že ako.

Šimon spravil veľmi dobré vysvetlenie aktuálneho stavu projektu s víziou na nasledujúce obdobie.

Budúce stretnutie (backlog rooming).

Rozdelenie úloh na nasledujúci šprint

AGREGÁCIE:

- počítanie agregácií z anotovaných dát -> vytvoriť tabuľku v MS SQL, rozšíriť workera (pri pridaní anotácie nech sa inkrementuje príslušný counter) - [zdokumentovanie fungovania Raven DB a aj workera ]

API

- zadefinovanie volaní API -> identifikovanie základných call-ov (cez JSON) [vytvoriť projekt, vytvoriť session {v podstate podobné veci aké sú dostupné cez web}]
- autentifikácia cez API kľúč (cez databázu)
- volanie z ALEFu (skryť do libky), premenovať controller
- administrácia -> sledovanie pohľadu

REAL TIME

- add-on (Chrome) -> Publisher -> Event(x,y) OR Event(XPath)
- ALEF -> Subscriber (JavaScript) [format Publisher-Subscriber] ->handler (odozva ALEFu), AJAX
- Publikovanie Eventov -> JQuery \$('event').ontrigger(...); pozrieť BackboneJS framework; pridanie GazeTrackingController-u, ktorý by obsluhoval eventy ktoré prídu z JQuery
- počúvanie eventov cez backboneJS (spojiť so zvýrazňovaním elementov)

ODOZVA ALEFU

- zvýrazňovanie elementu v ALEFe, na ktorí sa užívateľ pozerá



## Ohodnocovanie

úloh:

Autentifikácia cez API kľúč (20) - Marek

Volanie z ALEFu (3) - Viki

Administrácia (sledovanie pohľadu) (3) - Matej

Zadefinovanie volaní API (5) - Šimon

Agregácie (20) - Peťo Kiš

Publikovanie eventov (13) - Peťo Kušnir (pozrieť si dokumentáciu k addonu a skúsiť spísať dokumentáciu, aby sa vedelo do budúcnosti)

Počúvanie eventov + zvýraznenie (13) - Maťo Svrček

Doplniť odhady časov (najlepšie ešte dnes).

Všetci sú už hladní a je tu zima. Je 12:15. Končíme a ideme jesť.

## 4 Štvrté stretnutie

**Peto Kušnir:** spravil publikovanie eventov, nepodarilo sa nastaviť extension. Identifikoval si miesto. Nestihol dokončiť úlohu, bol v tom problém. Dnes to dokončí.

Treba spraviť dokumentáciu ku každému tasku, aby ostatní vedeli sa v tom orientovať.

**Viktoria** – nevedela presne ako na to. Nepodarilo sa jej úplne dokončiť task.

**Martin Svrček** - Mal robiť zvýraznenie eventov. Podarilo sa mu to. Robo mal ešte pripomienky k tomu. Urobil analýzu testovania.

**Peto Kiš** - nepodarilo sa mu úplne dokončiť task. Odhaduje to ešte na dosť dlho.

**Mato Kucek** - Podarilo sa mu splniť task. Najviac sa mu zabralo rozbehať ALEF.

**Marek Šulek** - Air spec treba. Dokončil task ale nespravil testy.

**Šimon** - Spolupracoval na tasku s Marekom, čiastočne zadefinoval API, dnes to dorobí.

### Zhrnutie šprintu

Peto Kušnir - Navrhuje stretnutia kde sa spoločne programovalo.

Viki - tiež je za spoločné programovanie

Martin - Spokojný.

Peto - Jeho uloha bola, zle odhadnutá, venoval jej veľa času, potrebuje spolupracovníka.

Jeho uloha bola pre viacerých ľudí.

Matej - Robilo sa mu dobre lebo dostal dobrý support.

Robo - Vela času stále zaberá informácie, treba si zdieľať informácie.

Marek - Nespokojný s efektivitou svojho programovania,

Šimon ukazoval navrhnuté **API**. Requesty budú vo formate post. Responses vo forme JSON.

Agregácie - Peťo navrhol ako vylepšiť databázu. Dohadovali sme sa ako bude navrhnutá databáza.

Real time komunikácia -

Alef

### Pridelenie uloh

- Marek - Pridanie funkcionality do API, support od Šimona - 13 story points

- Šimon - Dozerat a pomoct nad marekom a Petom.
- Martin - Analyzovanie otazkovaca v ALEFE - 13 points
- Matej - Manazment a vytvorenie gazetracking projektu v ALEFE, volat funkcie API
- Peto - Analyza
- Dokumentaciju, musi kazdy spisat.

## 5 Piate stretnutie

Peter kiš - robili s Matejom dopyty. Vytváral testovacie data, Agregacne dopyty

Robov komentar - otestovat z pohladu priepustnosti, kde su limity riesenia, skusit si cez simulator aspon od jedneho cloveka ci to bude stihat. Musite to niekde nasadit.

Matej Kucek - Nevie sa pripojit, Nedostal sa k analyze redisu, Robil agregacne dopyty

Robov komentar - heslo je nepoviem, spravit si novy ucet cez ktory to budeme testovat, Redis nie je uz relevantny,

Peto Kusmir - dorobit ulohu, zacal so serverovou castou, nestihol vsetko

Robov komentar - ked si nie sme isty treba dat komentar, mozme sa poradit s minulymi vyvojarmi, zakomentovane kody su uz asi nepotrebne, treba ich mazat a komentovat na urovni

Simon Valicek - pokracuje v praci, treba doriesit deploy a stranku, skusi to dnes pripomenut,

Robov komentar - treba to doriesit, sprístupnit stranku,

Marek Sulek - dorobil funkcie z minuleho tyzdna, pripravil si funkcie,

Robov komentar - oddelit zobrazenie od kontrolera, pytal sa ohladne code review ci su vsetci v perconiku.

Martin Svrcek - dorobil view, opisal svoje problemy

Viktoria Lovasova - spravila manazovanie projektu gaze tracking, ale este to nekomunikuje a je to nekonzistentne jazykovo.

Robo rozpravoval o buducnosti alefu.

Dalej rozprava o obsahu dokumentacie, akym cinnostiam sa manazer venoval, podiel prace vyzera ako jedna tabulka kde su vyjadreny pomer kto sa ktorej kapitole venoval. Aplikácie manažmentu tam opiseme jednotlivé cinnosti, v stručnosti zhodnotit a odkázat sa na príslušnú metodiku. Sumarizacia sprintov - burndown chart, okomentovany preco tak vyzera, opisat ak ofungovalo planovanie a priebeh. Dalej zhodnotenie co bolo dobre co by sme chceli stihnout.

Diskusia o metodikach.

## 6 Šieste stretnutie

Peter Kušník:

-moderoval stretnutie

Peter Kiš:

- analyzoval, ako aplikacia komunikuje s RavenDB
  - implementacia je neprehladna, nepouziva orm ale linq
  - obsahuje vela zakomentovaneho kodu-nvieme, ci je nefunkcny alebo nepotrebný

Robo:

- v pripade potreby je nutne urobit markup existujuceho kodu
- pri implementacii noveho kodu treba kod komentovat a generovat z neho dokumentaciu
- treba pouzivat db migracie

Matej:

- nastavoval query v tfs
- dorobit do alefu nove view v administracii
  - rozbehal si alef
  - nema v nom testovacie data

Robo:

- treba sa pridať do redminu alefu
- vie pomocť s vysvetlenim implementacie alefu

Marek:

- studoval rest sluzby

Robo:

- treba si pozriet ako sa v sucasnosti realizuju dopyty na db

Viki:

- upravila api call z alefu
- vytvorila testy, ale ich funkcnost je zavisla na implementacii api
- este mala napisat metodiku testovania, co nestihla

Robo:

- testy by nemali priamo volat api ale mali by sa napisat mocky
- toto iste treba aplikovat pri testovani funkcii, ktore vyuzivaju model
- treba si pozret dokumentaciu k Rspec
- pri zistení chyby pri testovani je dobre vytvorit novy task v tfs

Martin:

- robil analyzu implementacie zachytavania eventov zverejnených browser extensionom pomocou backbone.js
- pisal metodiku riadenia scrum

Robo:

- treba si pozriet repozitar alefu, kde je mozne vidiet pouzitie backbone.js
  - backbone.js vyuziva model, ktory komunikuje so serverom
  - view, ktory zobrazuje data

Šimon:

- nova domena
- aliasy + deploymenty pre novu domenu
- vytvorenie kopie db
- rozpracovanie api

Robo:

- v db nie su az tak hodnotne data
- specifikaciu api je nutne stihnuc dokoncit do konca sprintu
- v 9. tyzdni sa odovzdava priebezna dokumentacia k projektu, riadeniu a big picture
  - big picture
    - opis povodnej myslienky sluzby
  - architektura
    - moduly
    - prepojenie
    - komunikacia
  - vystupy analyzy jednotlivych modulov
  - sucastou dokumentacie vysledky testov modulov
  - dokumentacia riadenia
    - big picture
    - role clenov a podiel prace
      - zodpovednosti na systeme jednotlivych clenov
    - aplikacie manazmentov
    - opis jednotlivych sprintov
- na testovanie budeme pouzivat Nspec
- v ramci tp cupu budu k dispozicii mentoringy
  - prezentacia/business
- treba vytvorit a vytlacit lean canvas
- na code review treba pouzivat perconik
  - Robo sa ozve Karolovi Rastocnemu, aby nam v nom vytvoril projekt

## 7 Siedme stretnutie

Viktoria Lovasova

naimplementovane manazovanie projektov  
testy nie su

Peter Kis

stihli sa agregovane dopyty, je to otestovane  
zacal sa robit manazer, ktory zbiera data a vytvara agregacie  
testovat cez simulator, ale on nie je prepojeny s anotaciami

Peter Kusnir

vytvoril dokumenty na zdokumentovany kod  
treba otestovat nejako

Simon

bojoval so serverom, pisal na fakultny mail ale neodpisali  
nemal iny task

Marek Sulek

spravil tasky localne  
nema testy  
ani code review

Martin Svrek

pocuvanie v alefe javascript  
otestovanie  
analyza otazkovaca

## **Retrospektiva**

Robo

dufas ze, na konci kazdeho sprintu bude vzdy nasadenie testy sa spustia  
koniec sprintu a tasky nie su uplne skoncene  
snazit sa uzavriet  
hlavne testy

Martin Svrcek

rozdelovat tasky uz na stretnuti, nie len user story  
nabuduci tyzden v stredu prejdeme si dalsie user stories a ohodnotime to aj na dalsom  
stretnuti uz len vybrat a rozbit na tasky a takto by sme to stihali

Marek Sulek

viazne komunikacia

## **Zaciatok 4. sprintu**

Planovanie

### API

Sessions

Users: previazat usera so systemom druhym, vytvorit usra a zapamat idcko

AOI

Agregacie

get fix count, fix duration

### AGREGACIE

prerobit tabulku v casti statistics  
a zobrazi pre tychto pouzivatelov taketo pocty  
kludne aj viac tabuliek  
nemusi to byt user friendly

Vytvaranie agreacii

### JS KNIZNICE

evaluation questions  
bud volanie z kodu  
alebo definovat v administracii custom pravidla  
urobit trigger cez tu otazku

### AOI

URL

ako to funguje, ignorujem vsetko za / napr ze dam tam \* - zanalyzovat ci to tak  
funguje, ak ano ci to anotacny worker zohladnuje

xpath

nevratit len jeden ale viac

## 8 ôsme stretnutie

Simon:

dospecifikovanie API, pretrvávajúce problémy s pridelením domeny

Peter Kusnir:

vytvoril základnu kostru API volaní pre oblasti zaujmu, otázky ohľadom typu navratových hodnôt, potrebuje ešte dospecifikovať niektoré volania

Marek:

Začal robiť na API funkciách pre správu užívateľov. Upravil celkovú štruktúru API a zmenil navratové hodnoty funkcií na DTO príslušného modelu + http status kód, pomáhal

Simon napísal aj testy na niektoré vytvorené funkcie.

Viki:

Malá robiť API funkcie pre sessiony. Vytvorila základné funkcie podľa špecifikácie, pomáhal Marek.

Mato Svrcek:

Robí na analýze a implementácii otázkovacieho v Alefe a jeho prepojenie so systémom cez API. Podarilo sa mu zdokumentovať otázkovacie v Alefe, vytvoril o tom dokumentáciu a začal s implementáciou.

Matej Kucek:

Mal spraviť zobrazovanie agregácie vo webovej aplikácii. Mal problémy s testovaním kódu keďže to u seba nevedel rozbehať. Dosiel Miso a pomohol mu s tým a aj rozprával o tom ako testovať aplikáciu.

Peto Kis:

Podarilo sa mu dokončiť počítanie agregácie.

## 9 Deviate stretnutie

Viki:

Dorobila API funkcie pre správu sedení, nestihla napísať testy, kód skontroloval Marek v rámci pair programming-u

Matej:

Podarilo sa mu rozbehať webovú aplikáciu u seba aj keď musel následne riešiť nejaké problémy s prihlásením. Ako riešenie si musel rozbehať u seba MySQL databázu v ktorej sú záznamy o užívateľoch a preto sa mu nepodarilo úplne dokončiť vizualizovanie agregácie na webe

Peto Kis:

Dokončenie práce na počítaní agregácie, spísanie dokumentácie a testovanie prípustnosti

Peto Kusnir:

Doplnil funkcie do API pre správu oblasti zaujmu, zmenil navratové hodnoty funkcií podľa dohodnutého štandardu, začal na testoch ale nestihol dokončiť

Marek:

Dokoncil API funkcie pre spravu userov, napisal testy a menil celkovu strukturu projektu API. Pomahal Viki s implementaciou.

Mato Svrcek:

Implementacia generovania otazky na zaklade prijateho eventuu.

Simon:

Trosku s meskanim ale predsa, vecne problemy so serverom, dospecifikoval API volania, stretol sa s Petrom Lackom kvoli domene, aj tak sa s tym velmi nepodarilo pohnut

### **Semestralna retrospektiva:**

Podarilo sa spravit pocitanie agregacii a zaklad pre API, poopravovali sa niektore chyby v kode, robilo sa na prepojeni s Alefom, od ktoreho sa kvoli buducim zmenam nakoniec upustilo, tim sa uchytil v teme, komunikacia bola na dobrej urovni ale treba sa vyvarovat situaciam kedy sa tyzden kvoli nejakej chybe niekto nevie pohnut a tim sa o tom dozvie az na stretnuti, zlepisit pracu v kolektive, stretnutia aj mimo vyhradeny rozvrh a riesenie problemov spolocne, zlepisit pisanie testov a vykonavanie prehliadok kodu, neriesit ulohy na poslednu chvilu, kladne je hodnotena ustretovost predosleho timu a ochota pomoct a poradit pri vzniknutych chybach alebo nasadzovani a testovani aplikacie.

Planovanie posledneho tyzdnoveho sprintu:

Dokoncenie rozrobenych uloh z predchadzajucich sprintov, doplnenie testov, spisanie dokumentacie riadenia a dokumentacie o inzinierskom diele a nasadenie celeho systemu na server.

## **10 Stretnutie so zákazníkmi**

- Prepojenie s ALEFom , Záznam z kamery – problémom je synchronizácia, spôsob odosielania , Treba dbať na čo sa používa eye-tracker.
- Biznis problém
- MB – Posielat' dáta klientovi, (Šimon – „prídeme s tým o biznis“) -> MB - neprídeme s tým o biznis, lebo to môžeme nejako splatniť -> ??.
- Ak to budeme posielat' na server tak to nebude škálovateľné -> ak bude 200 aplikácií s 200 používateľmi tak všetko bude tiecť na náš server a z neho a bude to problém. Ideou bolo aby tá druhá aplikácia nemusela robiť cez svoju databázu nejaké veci. Skôr investovať do toho aby to mohol klient použiť.
- MLabaj - Ani to nemusí ísť vôbec nikam a môže sa to v javascripte spracovať. Javascript bude čakať na event a ak sa niečo bude sledovať (nejaký pohľad) tak bude reagovať nejako.

SOFTEC-FIIT\_projekt

- MBielikova - Pozor na konkurenciu so Softecom -> s nimi sa bude robiť projekt na UX lab – budú tam ľudia aj od nás. Nemôžeme replikovať to čo je v požiadavkách na tento projekt.

- Minulý projekt robil spracovanie s webom -> ak by sme to dotiahli tak sa nebijeme so Softecom.
- Pridanie poznámok k priebehu experimentov -> toto sa budfe riešiť so Softecom.
- Kvalitatívne testovanie – napr. zadávanie úloh a pod. – máme toto vôbec riešiť ? – skôr nie, lebo to sa rieši so Softecom. Toto ani nie je potrebné , skôr sa to robí ako súčasť webu.

#### Vizualizácia

- aj pre jedného aj pre viac používateľov – nie on the fly ale prepočítané vopred
- heat mapy a obrazovka za tým, za grafom
- Vizualizácia pre učiteľa – **dôležité pre zákazníkov !!!** -> na základe toho by vedel že ktoré časti textu sú zlé. – systém by vedel pomôcť učiteľovi na základe získaných údajov zo sledovania – toto by bolo veľmi dôležité.

#### ALEF integration

- MB: dáta sa dostanú do ALEFu – a priamo v ALeFe vymyslieť scenár súvisiaci s pozorovaním
- ML: definovanie pravidiel (serverové, klientské) keď sa človek pozrie na LO tak sa ho spýtame prečo tam pozerá -> toto už funguje !!!
- Zamerať sa na webové aplikácie.
- Analýza portálu -> toto bola 1naj stránka a a na toto sa nepozeralo.

#### Zhrnutie

- Ako urobiť vyzbierané dáta užitočné – vizualizovať ich a prepojiť ich s aplikáciami.
- Orientovať sa na pomoc učiteľovi.