

## 2.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 trochu 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 posiela requesty, aby sa identifikovala musí použiť API key. Marekova aplikácia by zachytila volanie a vrátila hocikaký text. Funkcia už existuje – my potrebujeme aby sa prepojila s nami (vložíme tam URL alef-u). 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 konfiguráku 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 merge 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 trochu skorej.

V TFS nový repozitár na ALEF treba vytvoriť.

## Code review

- pozrieť aký kód bol pridaný vo Visual(prave tlačítka COMPARE)
- to čo je spravené či to robí to čo má robiť korektnosť a správnosť
- či to je spravené rozumne (konštanty, veci v konfiguráku, 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 Visualku , 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.