

Riadenie projektového tímu

1. Úvod

Tento dokument obsahuje informácie a aplikáciu metodík pri riadení tímu počas nášho tímového projektu. Práca na tímovom projekte trvala dokopy 2 semestre s prázdninovou prestávkou medzi nimi. Ďalej budem náš tímový projekt nazývať „Vanetlab“. V tomto dokumente je spísané ako sme využili Agile metodiku na našu prácu, ako sme si rozdeľovali úlohy podľa rolí v tíme a ako prebiehali šprinty. Na koniec zakončím dokument finálnou retrospektívou za zimný aj letný semester. Tento dokument neobsahuje nijaké informáciu k vývoju ani funkcionalite Vanetlabu. Pre tieto informácie, prosím pozrite dokumentáciu.

2. Role členov tímu a podiel práce

Na našom tímovom projekte sa podieľali šiesti členovia tímu. Každý z nás mal iné silné stránky a iné preferencie pri vývoji, takže sa nám po pár týždňoch vykryštalizovali semi-stabilné role a zodpovednosti. Nasledujú popisy našich samostatných rolí v tíme a našich zodpovedností.

Pre manažment tímu sme používali nástroj Azure DevOps. Tento nástroj nám umožnil organizovať a kontrolovať šprinty, prideliť jednotlivé úlohy členom tímu a zároveň pomohol veľmi pri písaní reflexii šprintov. Pre viac informácií o našej práci s Azure DevOps, prosím pozrite metodiku SCRUM.

Richard Andrášik

Richard mal rolu SCRUM mastera v tíme, čiže jeho hlavnou zodpovednosťou bolo dohliadať aby šprinty prebiehali hladko. Úlohou SCRUM mastera je zahajovať a ukončovať šprinty, rozdeľovať nepridelené úlohy a dbať na to aby šprinty skončili so všetkými úlohami úspešne dokončenými. Ako SCRUM master mal tiež za úlohu písať reflexie šprintov.

Mal tiež úlohu hlavného administrátora, ktorá zahŕňala písanie zápisníc na všetkých pravidelných týždňových stretnutí. Na začiatku projektu spísal veľké množstvo potrebnej analýzy. Pri dokumentácii projektu mal za úlohu napísať tento dokument o riadení projektového tímu. Na Vanetlabe nevyvíjal nové funkcionality, skôr opravoval chyby, čo sa nikomu inému nechceli.

Andrej Dubovský

Andrej bol náš hlavný front-end developer. Už od začiatku projektu pracoval na prelome vizuálnej a funkčnej stránky vo Vanetlabe. Ak by ste vybrali náhodnú funkcionálnosť, čo vidíte vo Vanetlabe, je veľká šanca, že v nej mal prsty Andrej.

V prvej časti projektu spracoval analýzu nástroja SUMO. Neskôr využil túto vedomosť integráciou NS-3 a SUMO vo Vanetlabe. Pri vývoji bol známy aj programovaním si funkcionálnosť jednu za druhou a nechávanie jeho chýb na niekoho iného. Keď bol vyzvaný dotiahnuť si svoju funkcionálnosť do konca, stal sa validačný incident, čo trval tri šprinty.

Vojtech Fudaly

Vojtech bol v našom tíme známy ako „Jack of all trades“. Pridával malé ale dôležité komponenty do Vanetlabu snáď v každej časti vývoja. Veľkú časť projektu, keď nepracoval na komponentoch, pomáhal pri vývoji väčších častí projektu ostatným členom tímu. Časť jeho kompetencie bolo aj dešifrovať a opravovať kód po Andrejovi.

Na začiatku projektu bol hlavný vývojár a údržbár našej tímovej stránky. Jeho úlohy sa však časom presunuli na vývoj čoraz komplikovanejších komponentov pre front-end aplikácie. Na konci projektu pomáhal spísať časť v dokumentácii o funkcionálnosti aplikácie.

Roman Grom

Roman mal pri celom vývoji pomocnú rolu. Od začiatku, kde pomáhal s analýzou nástroja SUMO, cez pomoci pri práci na tímovej stránke, pomoci vytvárania rozsiahlych komponentov do front-endu aplikácie po pomoc pri písaní dokumentácie. Pri práci sa zaboril do väčšiny častí projektu, čo sa odzrkadlilo na kvalite a konzistencii naprieč modulmi.

Najväčšiu samostatnú zásluhu má tiež pri front-ende Vanetlabu. Vytvoril a ďalej udržiaval okolité komponenty, ktoré sa netýkali hlavnej funkcionálnosti aplikácie. Ku koncu projektu urobil množstvo vizuálnych zmien, čo zlepšilo vzhľad Vanetlabu.

Samuel Kačeriak

Samuel bol hlavný back-end developer tímu. Sám vie zodpovedať za back-end Vanetlabu, ktorý pozná lepšie ako vlastnú dlaň. Je tiež autorom našej analýzy SDN a autorom Git metodiky. Často bral na svoju zodpovednosť rôzne zložité úlohy, ktoré si vyžadovali viacerých pomocných programátorov z nášho tímu.

V neskorších častiach projektu pracoval aj na front-ende, kde ošetroval chyby a vytváral riešenia pre okrajové situácie alebo scenáre. Samuel tiež urobil najväčšiu časť testovania, keďže na väčšinu chýb prišiel on.

Ondrej Martinka

Ondrej je ako lepidlo nášho tímu, keby lepidlo vedelo celý projekt urobiť samé. Neraz sa stalo, že sme potrebovali komponent a zistili sme, že ho Ondrej už urobil. Ondrej bol náš neformálny leader tímu lebo ako jediný vedel, čo prene projektu chýba a ako ho môžeme ďalej vyvíjať. Jeho rola sa dá popísať iba ako full-stack developer.

Okrem neskutočne rozsiahlej časti na back-ende projektu mal za úlohu využitie NS-3 v projekte. Od prototypov po finálnu verziu dohliadal na celý vývoj aplikácie. Chyby pri práci neriešil, tie nechával na svojich podriadených aby na nich strácali čas. Na konci projektu napísal obrovskú časť dokumentácie keďže vedel najlepšie ako Vanetlab funguje.

3. Aplikácie manažmentov

Pri tomto projekte boli použité 3 stupne manažmentu.

Prvý od vedúceho projektu, ktorý podobne ako zákazník, mal zoznam požiadaviek, ktoré sme počas projektu museli splniť a ku ktorým sa nás snažil mierne viesť. Náš vedúci projektu do vývoja projektu ani od rolí členov tímu nezasahoval. Na tejto úrovni manažmentu úplne stačilo aby boli požiadavky pri konci projektu splnené.

Druhý stupeň manažmentu bol od SCRUM mastera, ktorý vyberal a rozdeľoval úlohy členom tímu. Ako tu už bolo spomenuté, SCRUM master sa riadil hlavne podľa SCRUM metodiky, ktorá určuje našu prácu s Agile frameworkom. Na tomto stupni manažmentu SCRUM master rozdeľuje veľké celky backlogovaných úloh na menšie časti, ktoré prideluje všetkým členom tímu podľa osobných kompetencií. Počas projektu sme mali 10 dvoj týždňových šprintov na začiatku ktorých sme si napísali úlohy, čo potrebujeme urobiť, ohraničili rozsah šprintu a vymedzili rozsah úloh.

Tretí, posledný a najmenej formálny stupeň manažmentu pochádzal od team leadera, ktorým bol najčastejšie Ondrej. Toto bol neštruktúrovaný manažment na nízkej úrovni, ktorý zahŕňal pridávanie priorít úlohám a snahou zvýšenia efektivity ústnou negatívnou motiváciou. Na tejto úrovni riadenia bola taktiež zapojená Git metodika. Vo finálnom dôsledku na tejto úrovni aj team leader vedel ovládať kto s kým bude pracovať v prípade, že niektoré úlohy trvajú dlhšie ako sa očakávalo.

4. Sumarizácie šprintov

Všetky naše šprinty boli sumarizované našim SCRUM masterom Richardom v jednotlivých reflexiách šprintov. Tieto reflexie je možné nájsť na našej tímovej stránke. Obsahujú popisy k úlohám členov tímu počas každého šprintu. Každá reflexia obsahuje aj sumarizáciu očakávaní pri začatí šprintu, hodnotenie pri strede šprintu a finálna reflexia na konci šprintu. Každá predchádzajúca reflexia šprintu nám nejako pomohla organizovať nasledujúci šprint. Mali sme celkovo 10 šprintov: 5 cez zimný semester, 5 cez letný semester.

5. Globálna retrospektíva

Zimný semester

Cieľom prvého semestra tímového projektu bola analýza problému a možných riešení, vytvorenie tímovej stránky a prvý prototyp aplikácie Vanetlab. Celú analýzu sa nám vďaka jedinečnému tímovému úsiliu podarilo za prvý sprint. Pokračovali sme súbežne prototypovaním aplikácie a prácou na tímovej stránke. Do tretieho šprintu sme mali rozbehanú tímovú stránku a nespojené, samostatné prototypy back-endu a front-endu aplikácie. Tieto prototypy ešte nepoužívali reálne dáta, ale odvíjali sme sa od nich. Mali sme tiež už drôtené modely aplikácie, ku ktorým sme ďalej mohli smerovať.

Pri posledných dvoch šprintoch sme rýchlo pridávali funkcionality aplikácie aby bol finálny prototyp funkčný. Na konci priateho sprintu, sme mali hotovú základnú funkcionality aplikácie, aj keď tam bolo veľké množstvo chýb. Celkovo sme išli správne podľa časového plánu a dokončili všetko, čo sme chceli včas.

Letný semester

Keďže z prvého semestra sme už mali prototyp aplikácie so základnou funkcionality, cieľom práce v tomto semestri bolo funkcionality rozšíriť o všetky rozšírené funkcionality. Najväčšie množstvo času práce cez tento semester zabralo opravovanie chýb. Samotné to rozširovanie funkcionality aplikácie nebolo zložité, skôr bol problém s optimalizáciou. V tomto semestri sme mali aj problémy s tímovou víziou toho ako mala finálna aplikácia vyzeráť. Takže sme boli v priebehu práce spomalení na viacerých častiach.

Začali sme šiestym šprintom prerábaním slabých častí prototypu, ktoré sme na konci prvého semestra len tak zbúchali dokopy. V siedmom až deviatom semestri sme hlavne opravovali chyby, ktoré sme objavovali pri pridávaní nových funkcionality. V desiatom a finálnom šprinte sme už písali dokumentáciu.

Na začiatku semestra sme vyzerali dobre na to časovo. Neuvedomili sme si však, že posledných 10% práce zaberie 80% času. Preto nám zostalo trochu práce, čo sme museli dokončiť aj po skončení posledného šprintu. Táto práca nespádala do parametrov žiadneho šprintu a zahŕňala úlohy ako formátovanie dokumentácie alebo písanie dokumentu o riadení projektového tímu.