

Slovenská technická univerzita

Fakulta informatiky a informačných technológií

Ilkovičova 2, 842 16 Bratislava 4

Metodika konvencie



BEHAWORKS

Ághová Adriana
Civáň Martin
Cukerová Zuzana
Kováč Matúš

Pilňan Matúš
Schmidt Andrej
Villár Viliam

GitHub

V rámci projektu vytvoreného na githube máme vytvorených niekoľko schránok (logger, stranka, logger-client, logger-python-client). Každá obsahuje hlavnú vetvu „master“ a vedľajšie vetvy, ktorých pomenovanie je konzistentné s pomenovaním tasku, ktorého implementáciu obsahujú. Tieto vedľajšie vetvy (obsahujúce správne naimplementované riešenie) sa následne spájajú do hlavnej vetvy (master). Po identifikácii konfliktu v súboroch je tento problém riešený priamo členom tímu, ktorý dané rozdiely zanalyzuje a následne vytvorí konzistentný súbor.

Stránka

1. dodržiavať reportovanie na týždennej báze (report musí byť dostupný na stránke ku dňu najbližšieho stretnutia)
 - a. opis prác vykonávaných na stretnutiach
 - b. pomenovanie reportov -> REPORT [ČÍSLO_STRETNUTIA]
 - c. report = vždy aktuálny šprint zoradený podľa statusov (Done ,In Progres, To Do)
2. Záložka v rámci dokumentov „Odobzdané súbory“
 - a. Pridávať všetky dokumenty odobzdané do IS
 - b. Pomenovávať podľa stránky predmetu
3. Záložka v rámci dokumentov „Metodiky“
 - a. Pridávať všetky existujúce metodiky pre prácu v tíme č. 12
 - b. Pomenováva podľa vzoru Metodika <jej_nazov>
4. Záložka v rámci dokumentov „Dokumentácia stretnutí“
 - a. Pridávať všetky reporty zo stretnutí
 - b. Pomenovať ako „Stretnutie č. (poradie stretnutia)“
 - c. Pomenovanie obsahuje taktiež názov šprintu a dátum stretnutia
5. Záložka v rámci dokumentov „Prílohy na stiahnutie“
 - a. Pridávať všetky grafické produkty tímu
 - b. Pomenovanie ako „(názov veci) tímu“

Programovanie

1. Python používať PEP 8 (<https://www.python.org/dev/peps/pep-0008/>)
2. Pred commit-om je potrebné v prostredí (Idea) nastaviť nasledujúce veci:
 - a. „Reformat code“ - umožňuje preformátovať kód podľa požiadaviek, ktoré sme si určili v nastaveniach štýlu kódu .
 - b. „Rearrange code“ - umožňuje rýchlo zmeniť usporiadanie výrazov, príkazov, typov členov a ďalších prvkov v našom kóde.
 - c. „Optimize imports“ - spôsob riešenia nevyužitého importu je použitie rýchlej opravy

-
- d. „Perform code analysis“ – umožňuje vytvorenie analýzy kódu
 - e. „Check TODO(Show All)“ - IntelliJ IDEA neustále skúma projekt, či neobsahuje komentáre v zdrojovom kóde, ktoré zodpovedajú konkrétnym vzorom TODO, a zobrazí ich v okne nástroja TODO
 - f. „Cleanup“