

Metodika prehliadok kódu



Metodika prehliadok kódu

Tím č. 03 FireAnt

Vedúci tímu: Branislav Pecher, Ivan Srba

Autor metodiky: [Miroslav Sumega](#)

Dátum vytvorenia: 10.10.2019

Poslednú zmenu vykonal: [Miroslav Sumega](#)

Dátum poslednej zmeny: 12.11.2019

Táto metodika popisuje pravidlá týkajúce sa prehliadok kódu.

Organizácia prehliadok kódu

Aby nevznikol chaos pri prehliadkach kódu, je potrebné dodržiavať nasledovné pravidlá:

- Prehliadka kódu sa vykonáva pre každý pull request.
- Na začiatku každého šprintu sa určí, kto bude vykonávať prehliadku kódu pre každý potenciálny pull request.
- Pri zložitejších funkcionalitách vykoná prehliadku kódu osoba, ktorá má dostatočné množstvo skúseností s danou oblasťou.
- Pri jednoduchších funkcionalitách môže vykonať prehliadku kódu aj osoba s malými, prípadne žiadnymi skúsenosťami s danou oblasťou (zabezpečí sa tak rozširovanie znalostí v tíme).
- Ak bude pull request vrátený alebo feature nebude v šprinte úspešne dokončená, osoba priradená na prehliadku kódu bude opätovne priradená na prehliadku kódu pre daný pull request/feature.
- Každý šprint sa budú meniť páry reviewer - autor kódu. Toto pravidlo neplatí pre prípady, keď bol vrátený pull request alebo nedokončená feature.

Vykonanie prehliadky kódu

Pri vykonávaní prehliadok kódu sa riadime nižšie uvedenými pravidlami:

- Ak niečomu nerozumiem, spýtam sa autora kódu a navrhнем pridenie/upravenie komentára.
- Snažím sa nájsť prípady, ktoré nie sú v kóde ošetrené a ak ich nájdem, navrhнем zmenu.
- Ak nájdem často opakujúci sa kód, navrhнем agregáciu do funkcie/metódy.

- Ak viem zlepšiť kód, navrhmem zmenu.
- Upozorním autora, ak kód nespĺňa metodiku štýlu písania kódu (pre každý jazyk samostatná metodika).
- Ak kód implementuje funkcionality dostupné v už existujúcich balíčkoch a táto implementácia nie je odôvodnená, navrhmem zmenu alebo požiadam o prídanie odôvodnenia implementácie.
- Ak je funkcia/metóda príliš veľká a spája viac funkcionalít, ktoré logicky nepatria do jednej funkcie/metódy a je možnosť vybrať ich do samostatných funkcií/metód, navrhmem zmenu.

Po implementácii úprav vyplývajúcich z prehliadky kódu upozorním reviewera, aby upravené časti opätovne skontroloval.

Hodnotenie kódu

Hodnotenie kódu sa píše, keď má reviewer ku kódu pripomienky. Ak nie sú žiadne pripomienky, nie je potrebné písať hodnotenie kódu.

Hodnotenie kódu sa píše výlučne v anglickom jazyku a do pull requestu v ktorom je kód, ktorý sa hodnotí.

Autor kódu by mal dbať na to, že **kritika kódu nie je kritikou autora kódu**. Taktiež by mal dbať na to, že častokrát **subjektívna hodnota kódu môže byť vyššia ako objektívna hodnota**.

Pri písaní hodnotení kódu je dôležité dodržiavať nasledujúce pravidlá, aby nevznikli konflikty medzi recenzentom a autorom kódu:

- Správy sa píše výhradne v prvej osobe jednotného čísla, napr. "I do not understand this code" namiesto "You wrote obscure code".
- Píše sa o kóde, nie o autorovi. Píšeme "This code is not effective" a nie "You wrote ineffective code".
- Je vhodnejšie správy písať formou otázok, napr. "Would not it be better if...?" namiesto "I think it would be better if...".
- Ak je potrebné vyjadriť sa na adresu autora, je vhodnejšie sa zamerať na správanie než na charakterové črty.

Autor kódu by mal dbať na to, že navrhnutá zmena nemusí nutne viesť k lepšiemu kódu a teda že môže byť v takom prípade odmietnutá.

Dôležité je v primeranej miere pochváliť.

Zaujímavé čítanie obsahujúce väčšinu predošlých bodov je na nasledovných odkazoch:

Hodnotenie kódu tak aby som nikoho neurazil: <https://phauer.com/2018/code-review-guidelines/#tl-dr>

Prečo, ako, kedy a čo hodnotiť: <https://medium.com/palantir/code-review-best-practices-19e02780015f>