

Metodika práce na serveri (Autor: scrum master, požiadavka vzišla zo skúsenosti s praxou)

Táto metodika je dôležitá z pohľadu vývoja a práce na produkte. Máme github repozitár, kde je uložený kód serveru. Vždy funkčná a najaktuálnejšia verzia sa nachádza v masteri.

Odkaz na github repozitár: https://github.com/cergina/secure_data_collection (privátny)

JIRA: <https://sonet-team.atlassian.net/jira/software/projects/DCS/boards/1> (prístup len členom tímu)

1. Získanie najaktuálnejšej verzie projektu cez projekt v IntelliJ a bytie v branchi master

Pokiaľ ide práca na novom tasku

Predpoklad: V JIRE je už task vytvorený. Má isté číslo, napr.: DCS-69.

- 2a. Vývojár si od najnovšej branchi master vytvorí novú svoju s názvom toho tasku. "DCS-69"
- 3a. Po dokončení vývoja napíše commit message, comitne, pushne
- 4a. Požiada o pull request do mastera
- 5a. Prebehne code review s určenými ľuďmi na túto činnosť
- 6a. Ak prebehne všetko ok, do mastera sa zlúči táto vetva. Ak nie, vráti sa späť do vývoja

Pokiaľ sa jedná o opravu bugu, ktorý sa ukázal

Predpoklad: Bug má svoje vlastné číslo napr.: DCS-70. Ale je nalinkovaný na task napr.: DCS-69

- 2b. Vývojár si vytvorí branchu z mastera (ak bol už zlúčený samozrejme master s orig. vetvou)
- 3b. Názov tejto branchi bude DCS-70_repaires-69
- 4b. Ak by nastal extrémny prípad prelinkovania, že vytvorený bug opravuje chybu, ktorá vyskočila po opravení iného bugu názov bude štýlom DCS-71_repairs-70-69
- 5b. Zvyšné kroky sú súmerné s prácou obyčajnou

Testovanie

Ak sa bude jednať o funkcionálnosť, ktorá sa dá otestovať, jej otestovanie prebehne prostredníctvom aplikácie POSTMAN a poslania príslušných POST, GET, ... požiadaviek. Človek, ktorý testuje niečo je plne zodpovedný za svoje konanie, pokiaľ by otestoval len jednu funkcionálnosť namiesto otestovania všetkých s tým spojených