Facultatea Calculatoare, Informatica si Microelectronica

Universitatea Tehnica a Moldovei

Medii Interactive de Dezvoltare a Produselor Soft

Lucrarea de laborator#2

Mediul GitHub

lector asistent:

Autor: Irina Cojanu

Tincu Maxim lector superior:

Svetlana Cojocaru

2016

Laboratory work #2

1. Scopul lucrarii de laborator

Familiarizarea cu mediul GitHub

1. Obiective

Intelegerea si folosirea CLI (basic level)  
Administrarea remote a masinilor linux machine folosind SSH (remote code editing)  
Version Control Systems (git || mercurial || svn)  
Compileaza codul C/C++/Java/Python prin intermediul CLI, folosind compilatoarele gcc/g++/javac/python

1. Laboratory work implementation

3.1Tasks and Points

Familiarizarea cu mediul GitHub

3.2 Analiza lucrarii de laborator

GITHUB LINK - <https://github.com/irmate210/MIDPS>  
  
Instalarea GitHub pe calculator. Inregistrea unui profil pe GiHub. Crearea unui repositoriu. Intelegerea modului de lucru cu GitHub.

Concluzie:  
 GitHub este un proiect initiat de Google si este un spatiu de gazduire pentru proiecte open source. Am intilnit mai multi programatori de inalta calificare si fiecare dintre ei foloseau GitHub. Din aceasta cauza am hotarit sa vad ce este GitHub si cum el funcioneaza.

De la inceput am intrat pe GitHub.com, am scos si am instalat GitHub pe calculatorul meu personal si mi-am facut un profil. Dupa care urmeaza crearea repositoriului. Pentru aceasta am apasat butonul New Repository, i-am dat denumirea si descriptia. L-am facut public si am selectat Initialize this repository with a README.

Dupa aceasta a urmat procedura crearii unui branch. De la inceput repositoriul nostru are un singur branch numit master care este considerat primul branch sau defaul branch. Noi folosim branch-urile ca sa experimentam si sa facem editari inainte sa la commitem in master. Fiecare dintre noi a avut situatii cind a facem un document, il numim – FinalWork, si pe urma mai facem citeva documente cu mici schimbari ca exemplu – FinalWork1, FinalWork2Done, FinalWork\_Final. Branchiurile ne permit sa facem aceleasi lucruri in repositoriul GitHub. Pentru a crea un branch nou, apasam brach:master, scriem denumirea unui branch nou (exemplu Maxim) si tastam Create branch. Acum noi avem 2 branch-uri master si Maxim.

Urmatoarea etapa a fost realizarea unui commit. Mesajele commit salveaza in istorie editarile facute de dvs. pentru ca pe urma sa intelegi de ce si ce ati facut.

In etapa finala trebuie ca toate schimbarile dvs. sa le includem intr-un loc aparte. Pentru aceasta folosim Merge si Pull Request. Apasam buton verde Merge pull request si unim toate schimbarile in master. Confirmam merge. Dupa aceasta se poate de sters branchiul nostru (Maxim) pentru ca totul a fost unit/transferat pe master.