

Projektowanie i programowanie obiektowe

Wprowadzenie do pracowni programowania obiektowego

mgr inż. Krzysztof Rewak

28 września 2019

1 Plan zajęć

Przewidywany ogólny plan pierwszych zajęć laboratoryjnych:

- przedstawienie zasad zaliczenia przedmiotu;
- przedstawienie planu tematów i zajęć na cały semestr;
- przedstawienie sugerowanej literatury przedmiotu;
- wprowadzenie do systemu kontroli wersji Git.

2 System kontroli wersji

Należy utworzyć konto użytkownika na dowolnym systemie hostującym system kontroli wersji Git; zalecanym rozwiązaniem jest Bitbucket. Po zakończeniu procedury aktywacyjnej, należy utworzyć prywatne repozytorium o nazwie PP01 - Imię Nazwisko i udostępnić je prowadzącemu dla konta RewakK@pwsz.legnica.edu.pl.

2.1 Instrukcja korzystania z systemu kontroli wersji

Należy sprawdzić czy Git jest zainstalowany w systemie: najlepiej uruchomić Cmder i wpisać `git --version`. Jeżeli nie jest zainstalowany, należy zainstalować, najlepiej w pakiecie Laragon, który przyda się na następnych zajęciach: <https://sourceforge.net/projects/laragon/files/releases/4.0/laragon-full.exe>

2.1.1 Inicjalizacja katalogu repozytorium

Należy utworzyć katalog repozytorium, przykładowo w następujący sposób:

- `cd /` - przechodzi do głównego katalogu, tutaj `C:`
- `mkdir ppo` - tworzy katalog `ppo`
- `cd ppo` - przechodzi do katalogu `ppo`
- `mkdir 123456`, gdzie 123456 to numer indeksu - tworzy katalog 123456
- `cd 123456` - przechodzi do katalogu 123456

Polecenie `ls` pokaże istniejące w wybranej ścieżce pliki i katalogi.

2.1.2 Inicjalizacja repozytorium

Należy zainicjować repozytorium w następujący sposób (wewnątrz pustego katalogu projektu):

- `git init` - inicjuje repozytorium
- `git remote add origin https://bitbucket.org/krewak/pwsz.git` - dodaje połączenie do serwera; oczywiście należy podać adres do swojego repozytorium
- `git pull origin master` - ściąga wszystkie ewentualne zmiany
- `git status` - pokazuje stan repozytorium

2.1.3 Wprowadzanie zmian do repozytorium

Wprowadzanie zmian można przeprowadzić następująco:

- `touch readme.md` - tworzy plik
- proszę wpisać w utworzonym pliku swój numer indeksu, email studencki, adres repozytorium oraz imię i nazwisko
- `git add .` - (kropka po słowie `add` oznacza, że wybieramy wszystkie pliki) dodaje do repozytorium wszystkie pliki
- `git status` - pokazuje stan repozytorium
- `git commit -am "readme"` - dodaje commit o tytule `readme`
- `git push origin master` - przesyła commity na serwer

Przed dodaniem plików do repozytorium warto dodać w katalogu głównym plik `.gitignore`, którego treścią będzie jednolinijkowiec `*.exe`. Uniemożliwi to dodawanie plików wykonawczych do repozytorium, co drastycznie zmniejszy jego wielkość.

3 Zadanie do wykonania

Należy do repozytorium dodać katalog `lab01` z plikiem `lab01.txt`. Plik powinien zawierać listę komend potrzebnych do zainicjowania repozytorium, pobrania zmian z serwera oraz wgrania nowych zmian.

Literatura

- [1] Jerzy Grębosz. *Symfonia C++ Standard*. Wydawnictwo Edition 2000, 2008.
- [2] Robert C. Martin. *Czysty kod. Podręcznik dobrego programisty*. Wydawnictwo Helion, 2010.