Inżynieria oprogramowania Lab1 Piotr Jałocha 294251

1. Zainstaluj klienta Git i obsługę kluczy SSH

Zainstalowałem najnowszą wersję klienta GIT wraz z obsługą kluczy SSH przez instalator

<Instalacja.png>

1. Sklonuj repozytorium <https://github.com/InzynieriaOprogramowaniaAGH/MDO2022> za pomocą HTTPS

Sklonowałem repozytorium za pomocą HTTPS

<klonowanieHTTPS.png>

1. Upewnij się w kwestii dostępu do repozytorium jako uczestnik i sklonuj je za pomocą utworzonego klucza SSH
   * Utwórz dwa klucze SSH, inne niż RSA, w tym co najmniej jeden zabezpieczony hasłem

Utworzenie pierwszego klucza SSH zabezpieczonego hasłem

<ssh1.png>

Utworzenie drugiego klucza SSH zabezpieczonego hasłem

<ssh2.png>

* + Skonfiguruj klucz SSH jako metodę dostępu

Dodanie kluczy do ssh-agent

<ssh-agent.png>

Dodanie klucza do konta GitHub

<dostep-ssh.png>

* + Sklonuj repozytorium z wykorzystaniem protokołu SSH

Klonowanie repozytorium za pomocą SSH

<klonowanie-ssh.png>

1. Przełącz się na gałąź swojej grupy

Przełączenie na gałąź grupy laboratoryjnej 1

<przelaczenia-GCL01.png>

1. Utwórz gałąź o nazwie "inicjały & nr indeksu" np. KD232144

Utworzenie gałęzi

<tworzenie-galezi.png>

1. Rozpocznij pracę na nowej gałęzi
   * W katalogu właściwym dla grupy utwórz nowy katalog, także o nazwie "inicjały & nr indeksu" np. KD232144

<tworzenie-katalogu.png>

* + W nowym katalogu dodaj plik ze sprawozdaniem

<dodanie-sprawozdania.png>

* + Dodaj zrzuty ekranu

<dodanie-zdjec.png>

* + Wyślij zmiany do zdalnego źródła

Dodajemy wszystkie nowe pliki

<dodawanie-plikow.png>

<commit.png>

<push.png>

* + Spróbuj wciągnąć swoją gałąź do gałęzi grupowej

Przechodzę na gałąź, która GCL01 przy użyciu git checkout GCL01.

Następnie używam polecenia git merge PJ294251

<wciąganie-galezi.png>

* + Zaktualizuj sprawozdanie i zrzuty o ten krok i wyślij aktualizację do zdalnego źródła (na swojej gałęzi)

Gałąź zabezpieczona, nie można zaktualizować sprawozdania i plików

<git-push-fail.png>

* + Oznacz tagiem ostatni commit i wypchnij go na zdalną gałąź

Dodawanie taga do commita

<git-tag.png>

* + Ustal hook, który będzie sprawdzał, czy wiadomość z commitem zawiera nazwę przedmiotu

Przechodzę w głównym folderze repozytorium do ukrytego katalogu .git, a następnie do katalogu hooks.

Tworzę plik commit-msg i nadaję mu uprawnienia do wykonywania.

<git-hook.png>

<missing-prefix.png>

Napisałem odpowiedni skrypt sprawdzający czy wiadomość zawiera odpowiedni ciąg znaków

<nano-commit-msg.png>

* + W jaki sposób stworzyć hook, który będzie *ustawiał* prefiks wiadomości commitu tak, by miał nazwę przedmiotu?

Modyfikuję poprzedni skrypt

<hook2.png>

1. Wystaw Pull Request do gałęzi grupowej

Przechodzę do zakładki Pull request a następnie klikam w Create pull request.

Później wybieram odpowiednie branche i zatwierdzam

<pull-request.png>

1. Zgłoś zadanie (Teams assignment)