**Imię i nazwisko:** Julia Żychowska  
**Kierunek:** Informatyka Stosowana (NS)  
**Nr. indeksu:** 307699  
**Przedmiot:** Metodyki DevOps  
**Sprawozdanie:** 01  
**Grupa:** GĆL03

**Sprawozdanie**

1. **Zainstaluj klienta Git i obsługę kluczy SSH**Instalowanie gita, używając polecenia: Obraz zawierający tekst

   Opis wygenerowany automatycznieObraz zawierający tekst

   Opis wygenerowany automatycznie

Obraz zawierający tekst

Opis wygenerowany automatycznie

1. **Sklonuj repozytorium** [**https://github.com/InzynieriaOprogramowaniaAGH/MDO2022**](https://github.com/InzynieriaOprogramowaniaAGH/MDO2022)**za pomocą HTTPS**Sklonowałam repozytorium za pomocą HTTPS używając polecenia git clone i adresu https:  
   Obraz zawierający tekst

   Opis wygenerowany automatycznie Obraz zawierający tekst

   Opis wygenerowany automatycznie
2. **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**Początkowo tworzyłam klucze SSH – DSA, poleceniem: **Obraz zawierający tekst, zrzut ekranu, sprzęt elektroniczny

  Opis wygenerowany automatycznie**  
    
  **Obraz zawierający tekst, zrzut ekranu, sprzęt elektroniczny

  Opis wygenerowany automatycznie**

które nie chciały się podpiąć do Github’a  


W następnej próbie utworzyłam 2 klucze ED25519, poleceniem: Obraz zawierający tekst

Opis wygenerowany automatycznie, przy których już nie było problemu. Pierwszy klucz nazwałam klucz1 i jest on bez hasła:

Obraz zawierający tekst

Opis wygenerowany automatycznie

Drugi klucz nazwałam klucz2 i jest już on z hasłem:  
Obraz zawierający tekst, zrzut ekranu, sprzęt elektroniczny

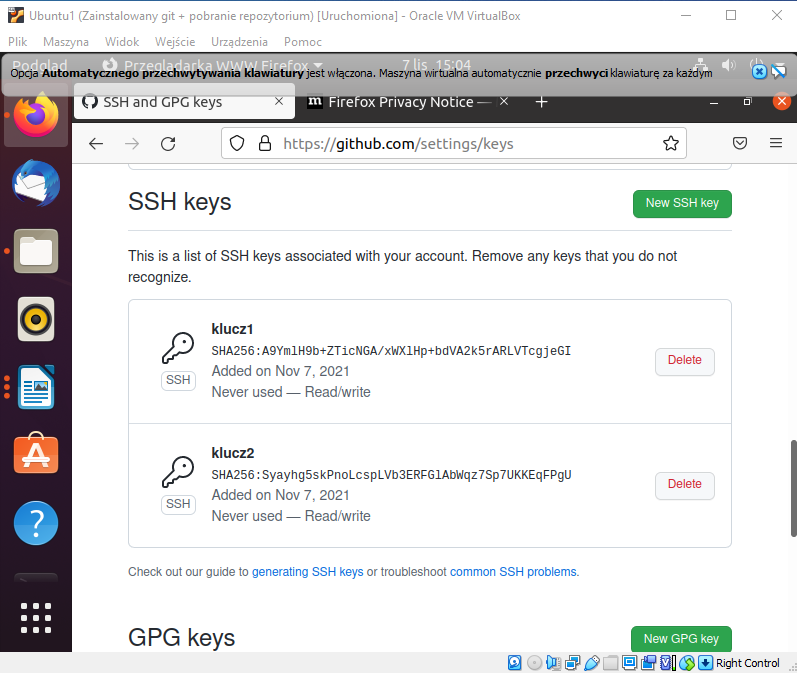
Opis wygenerowany automatycznie

* **Skonfiguruj klucz SSH jako metodę dostępu**

Dodałam dwa utworzone klucze do konta gitHub:  
**Obraz zawierający tekst

Opis wygenerowany automatycznie**

W polu key wpisałam utworzony klucz publiczny. Czynność potwórzyłam dla klucza1 i dla klucz2. Co dało następujący efekt:



* **Sklonuj repozytorium z wykorzystaniem protokołu SSH**Rozpoczełam od uruchomienia ssh-agenta poleceniem Obraz zawierający tekst, zrzut ekranu, sprzęt elektroniczny, komputer

  Opis wygenerowany automatycznie

Obraz zawierający tekst, zrzut ekranu, sprzęt elektroniczny, komputer

Opis wygenerowany automatycznienastępnie dodałam klucz prywatny do agenta poleceniem:  
Obraz zawierający tekst, zrzut ekranu, sprzęt elektroniczny

Opis wygenerowany automatycznie

Obraz zawierający tekst, zrzut ekranu, sprzęt elektroniczny

Opis wygenerowany automatycznie

i sklonowałam repozytorium za pomocą protokołu SSH: [git@github.com:InzynieriaOprogramowaniaAGH/MDO2022.git](mailto:git@github.com:InzynieriaOprogramowaniaAGH/MDO2022.git), dla pierwszego klucza, poleceniem:

Obraz zawierający tekst

Opis wygenerowany automatycznie

Obraz zawierający tekst

Opis wygenerowany automatycznieSprawdziłam użycie klucza w gitHub i wyszło, że klucz1 został wykorzystany:  
Obraz zawierający tekst

Opis wygenerowany automatycznie  
  
Następnie przeszłam do sprawdzenia drugiego klucza (klucz2 – z hasłem). Analogicznie jak przy kluczu1 uruchomiłam agenta i dodałam klucz2. Jedyną różnicą było to, ze przy dodawaniu poprosił mnie o wcześniej przypisane hasło:  
Obraz zawierający tekst

Opis wygenerowany automatycznie  
  
Po tej czynności pobrałam repozytorium ponownie:Obraz zawierający tekst

Opis wygenerowany automatycznie  
  
Przed kolejnym pobraniem repozytorium usuwałam poprzednie poleceniem: rm -rf MDO2022.  
  
  
  
  
  
  
  
  
  
Sprawdzając wykorzystanie klucza okazało się, że drugi również został wykorzystany:  
Obraz zawierający tekst, zrzut ekranu, parking

Opis wygenerowany automatycznie

1. **Przełącz się na gałąź swojej grupy**

Polceniem git checkout GCL03 przełączyłam się na gałąź GCL03Obraz zawierający tekst, ekran

Opis wygenerowany automatycznie

1. **Utwórz gałąź o nazwie "inicjały & nr indeksu" np. KD232144**Poleceniem git checkout -b JZ307699, utworzyłam nową gałąź i od razu się na nią przełączyłam:

Obraz zawierający tekst, ekran

Opis wygenerowany automatycznie

Tutaj zrobiony punkt 4 i 5 w ubuntu:

Obraz zawierający tekst

Opis wygenerowany automatycznie

1. **Rozpocznij pracę na nowej gałęzi**

* **W katalogu właściwym dla grupy utwórz nowy katalog, także o nazwie "inicjały & nr indeksu" np. KD232144**W katalogu MDO2022/GCL/03 utworzyłam poleceniem mkdir JZ307699 katalog:  **Obraz zawierający tekst

  Opis wygenerowany automatycznie**
* **W nowym katalogu dodaj plik ze sprawozdaniem**W katalogu JZ307699 wrzuciłam aktualne sprawozdanie

**Obraz zawierający tekst

Opis wygenerowany automatycznie**

* **Dodaj zrzuty ekranu**

Zrzuty ekranu umieściłam tutaj.

* **Wyślij zmiany do zdalnego źródła**

Dodałam plik „Sprawko” poleceniem git add Sprawko.docx, następnie zrobiłam commita, git commit -m „Sprawozdanie”. Terminal poprosił mnie o podanie maila, który powie mu kim jestem. Następnie powtórzyłam wykonanie commita i wysłałam zmiany do zdalnego źródła poleceniem git push origin JZ307699.

Obraz zawierający tekst

Opis wygenerowany automatycznie

* **Spróbuj wciągnąć swoją gałąź do gałęzi grupowej**Poleceniem git merge JZ307699 scaliłam zmiany wykonane w gałęzi JZ307699 z gałęzią bieżącą GCL03 (poleceniem git checkout GCL03 ustawiłam się na tej gałęzi). Następnie probówałam wysłać zmiany do źródła, które wyświetliły błąd:

Obraz zawierający tekst

Opis wygenerowany automatycznie

* **Zaktualizuj sprawozdanie i zrzuty o ten krok i wyślij aktualizację do zdalnego źródła (na swojej gałęzi)**
* **Oznacz tagiem ostatni commit i wypchnij go na zdalną gałąź**
* **Ustal hook, który będzie sprawdzał, czy wiadomość z commitem zawiera nazwę przedmiotu**
* **W jaki sposób stworzyć hook, który będzie ustawiał prefiks wiadomości commitu tak, by miał nazwę przedmiotu?**

1. **Wystaw Pull Request do gałęzi grupowej**
2. **Zgłoś zadanie (Teams assignment)**