

**Wydział Matematyki i Fizyki Stosowanej**

**Inżynieria i analiza danych**

**PROJEKT**

**Usługi sieciowe w biznesie**

**Gra kółko i krzyżyk - GIT**

Daniel Krzysik 166667

Rzeszów 2022

Spis treści

[1. Wstęp 3](#_Toc133101173)

[2. Git 3](#_Toc133101174)

[2.1 Git – podstawowe funkcje 3](#_Toc133101175)

[2.2 Instalacja Gita 3](#_Toc133101176)

[2.3 Konfiguracja gita 9](#_Toc133101177)

[3 Github 9](#_Toc133101178)

# 1. Wstęp

W dzisiejszych czasach programowanie stało się bardzo popularne, a wraz z nim także korzystanie z różnych narzędzi, takich jak git czy Github. Dlatego też postanowiłem nauczyć się tych narzędzi oraz wykorzystać je w swoim projekcie, którym jest gra kółko i krzyżyk w języku R.

Celem pracy jest stworzenie funkcjonalnej i poprawnie działającej gry, która umożliwi użytkownikom granie w kółko i krzyżyk na planszy 3x3. W ramach pracy zostanie omówiona koncepcja projektu oraz zastosowane metody, które pozwoliły na jego realizację. Opiszę także korzystanie z systemu kontroli wersji git oraz umieszczenie projektu na platformie Github.

Wierzę, że ta praca pozwoli mi nie tylko na rozwijanie swoich umiejętności programistycznych, ale także na poznanie nowych narzędzi, które pozwolą mi na jeszcze skuteczniejszą i efektywniejszą pracę.

# 2. Git

Co to jest Git i dlaczego cieszy się tak dużą popularnością? Ten system kontroli wersji znacznie usprawnia, a jednocześnie zabezpiecza codzienną pracę przy kodzie. Dzięki swojej prostocie i elastyczności może być wykorzystywany zarówno przy drobnych, jak i ogromnych projektach. Dlatego też jest używany przez programistów oraz grafików na całym świecie.

Odpowiadając w skrócie na pytanie, co to jest Git, należy powiedzieć, że to system kontroli wersji. Służy on więc do zarządzania historią kodu źródłowego. Jego funkcjonalność ma kilka podłoży. Między innymi sprawdza się tak dobrze, ponieważ:

* pozwala na jednoczesną pracę na tym samym kodzie przez kilka osób,
* umożliwia transferowanie i łączenie zmian z różnych branchy w jednym projekcie,
* pozwala na pracę offline we własnym repozytorium,
* jest szybki i wydajny.

Cechy te sprawiły, że Git szybko został doceniony w całej branży. Przechowywanie wersji, a także możliwość rozgałęziania kodu to niewątpliwie jego ogromne zalety.

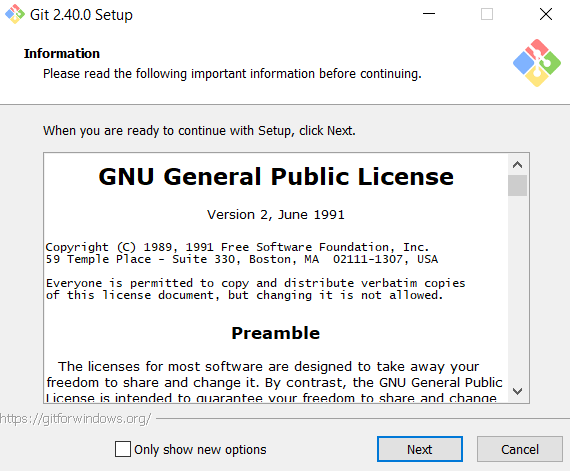
## 2.1 Git – podstawowe funkcje

Do najważniejszych pojęć należą:

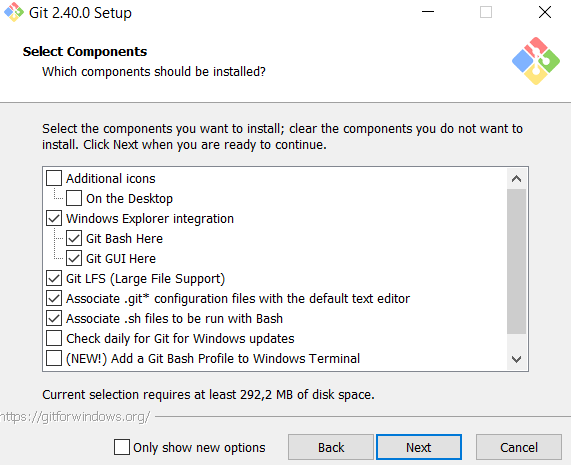
* SCM – Source Code Management, czyli po prostu kontrola wersji, a więc to, co zawdzięczamy systemowi Git;
* repozytorium – miejsce przechowywania całego projektu, wszystkich wersji kodu i historii wprowadzonych zmian;
* branch – pojedyncze odgałęzienie, jedna wersja, na której pracuje dany programista;
* clone – pozwala na skopiowanie kodu z repozytorium do własnej gałęzi;
* commit – przesyła dane z Twojej gałęzi do repozytorium;
* merge – łączy zmiany wprowadzone w poszczególnych branchach.

## 2.2 Instalacja Gita

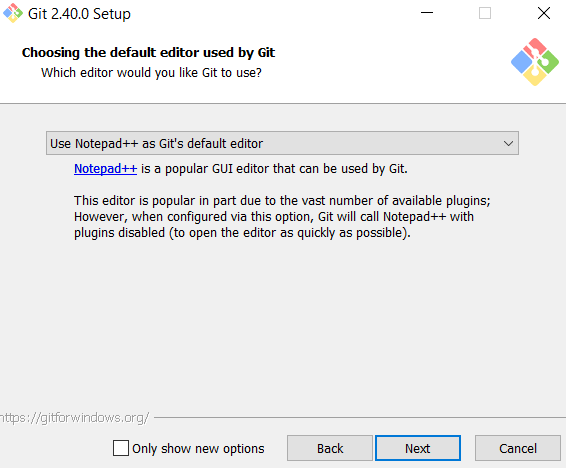
1. Pobierz instalator dla systemu Windows ze strony <https://git-scm.com/download/win>
2. Uruchom pobrany instalator
3. Zaakceptuj warunki licencji i kliknij przycisk "Next".



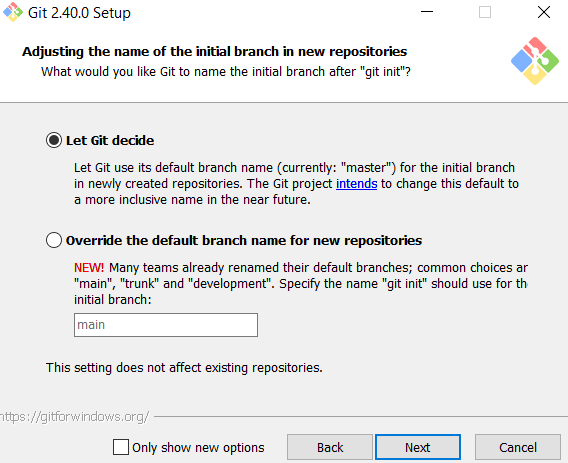
1. Wybierz składniki, które chcesz zainstalować. Domyślnie zaznaczona jest opcja "Git Bash Here", która pozwala na uruchomienie konsoli Git Bash z poziomu menu kontekstowego w Eksploratorze plików.



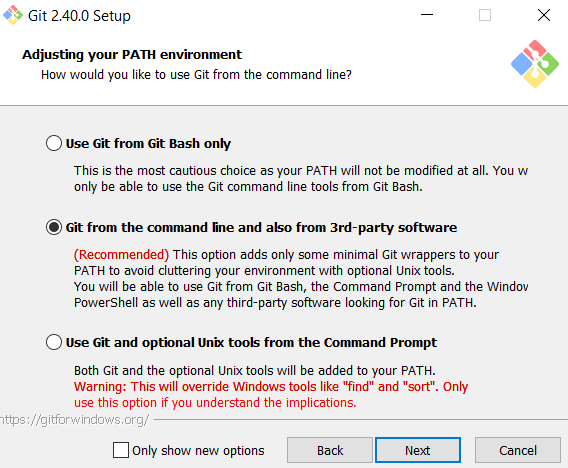
1. Wybierz edytor tekstu, który chcesz używać z Gitem. Domyślnie wybrany jest edytor Nano, ale możesz wybrać dowolny inny edytor, np. Notepad++ . Kliknij przycisk "Next".



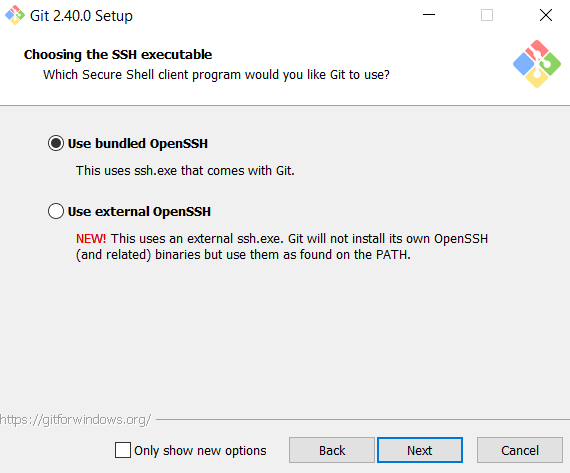
1. W tym oknie można wybrać, jak nazwę ma mieć domyślny branch w nowo tworzonych repozytoriach. Domyślnie Git proponuje nazwę "master", ale możesz wybrać również niestandardową nazwę. Jeśli wybierzesz niestandardową nazwę, pojawi się pole tekstowe, w którym możesz wpisać nazwę swojego wyboru.



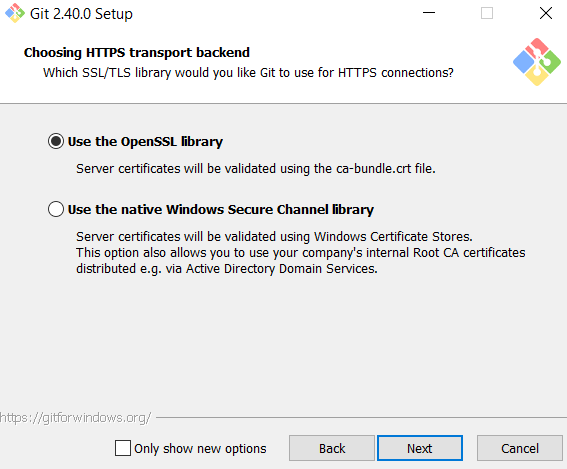
1. Poniżej decydujemy jak mocno zintegrowany z Windowsem ma być nasz git



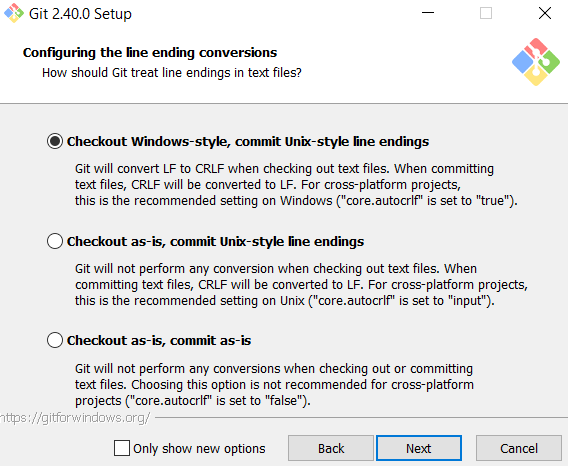
1. Poniższy krok instalacji Gita dotyczy wyboru klienta SSH, który będzie używany przez Gita podczas połączeń z zdalnymi serwerami.



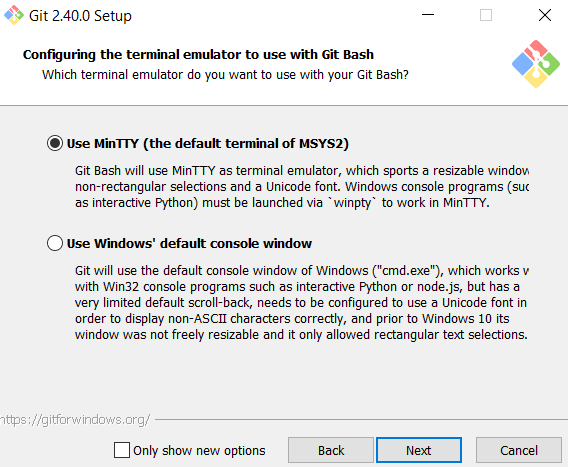
1. Ten krok instalacji Git'a dotyczy wyboru sposobu, w jaki Git będzie korzystał z protokołu HTTPS podczas łączenia się z zdalnym repozytorium.



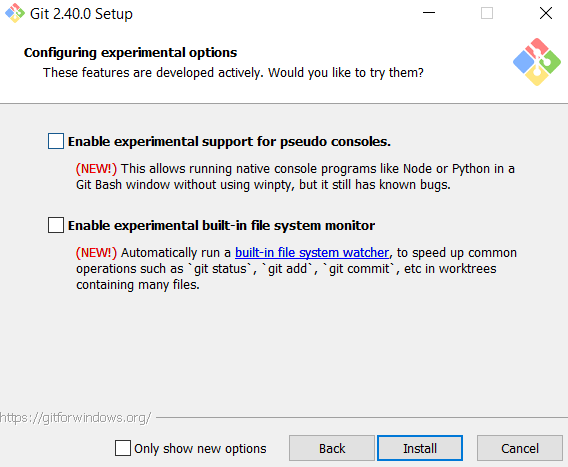
1. Konfiguracja konwersji końców linii.



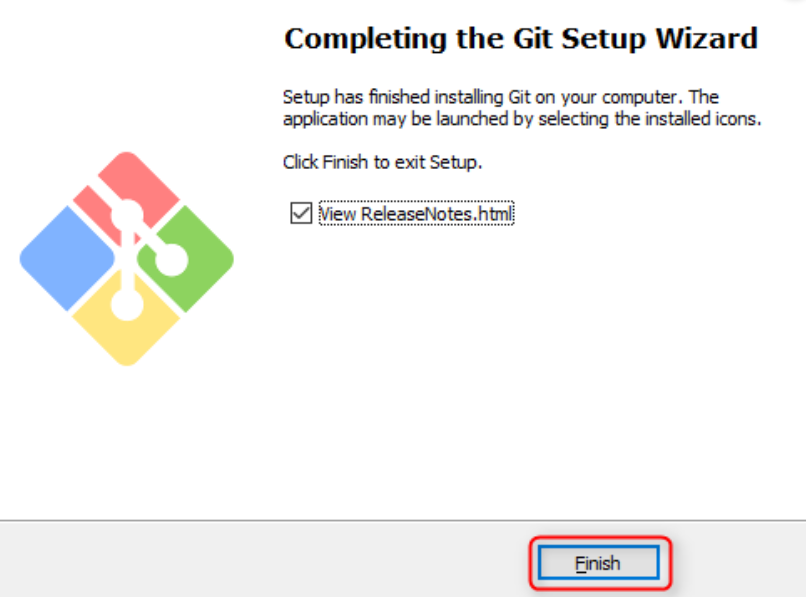
1. Krok "configurating the terminal emulator to use with git bash" podczas instalacji Git'a dotyczy wyboru preferowanego terminala do użytku z Git Bash.



1. Krok "configuring experimental options" podczas instalacji Git'a umożliwia skonfigurowanie eksperymentalnych opcji dla Gita. Te opcje są zazwyczaj dostępne tylko dla zaawansowanych użytkowników i nie są zalecane dla początkujących. Kliknij przycisk "Install", aby rozpocząć instalację.



1. Po zakończeniu instalacji kliknij przycisk "Finish", aby zakończyć proces instalacji.



## 2.3 Konfiguracja gita

Na początku musimy wstępnie skonfigurować naszą instalację Gita. Najważniejsze to ustawienie nazwy użytkownika i adresu e-mail. Dokonuje się tego wpisując poniższe polecenia:

git config –global user.name nasz\_login



git config –global user.mail nasz\_main@gmail.com



# 3 Github

GitHub to usługa hostingu umożliwiająca zarządzanie repozytoriami Git. Przy jego pomocy jesteśmy w stanie udostępnić swój kod w jednym miejscu dla wszystkich. Dzięki temu – w tym samym czasie – zapewniona jest możliwość aktywnej współpracy z pozostałymi członkami projektu. GitHub w przeciwieństwie do samego Gita działa w oparciu o chmurę. Każdy członek projektu może bez względu na szerokość geograficzną i sprzęt, na którym działa uzyskać zdalny dostęp do repozytorium Git (warunkiem jest dostęp do sieci).

# 4. Środowisko programistyczne R

## 4.1 Instalacja i konfiguracja R

## 4.2 Podstawy języka R

# 5. Tworzenie gry kółko i krzyżyk w języku R

## 5.1 Analiza problemu

## 5.2 Projektowanie struktury gry

## 5.3 Implementacja funkcji gry

## 5.4 Testowanie i uruchamianie gry

# 6. Wnioski

## 6.1 Podsumowanie projektu

## 6.2 Możliwość rozwoju projektu

# 7. Bibliografia

