

Instrukcja obsługi GitHub Desktop

Aleksandra Snopek

18 stycznia 2026

Spis treści

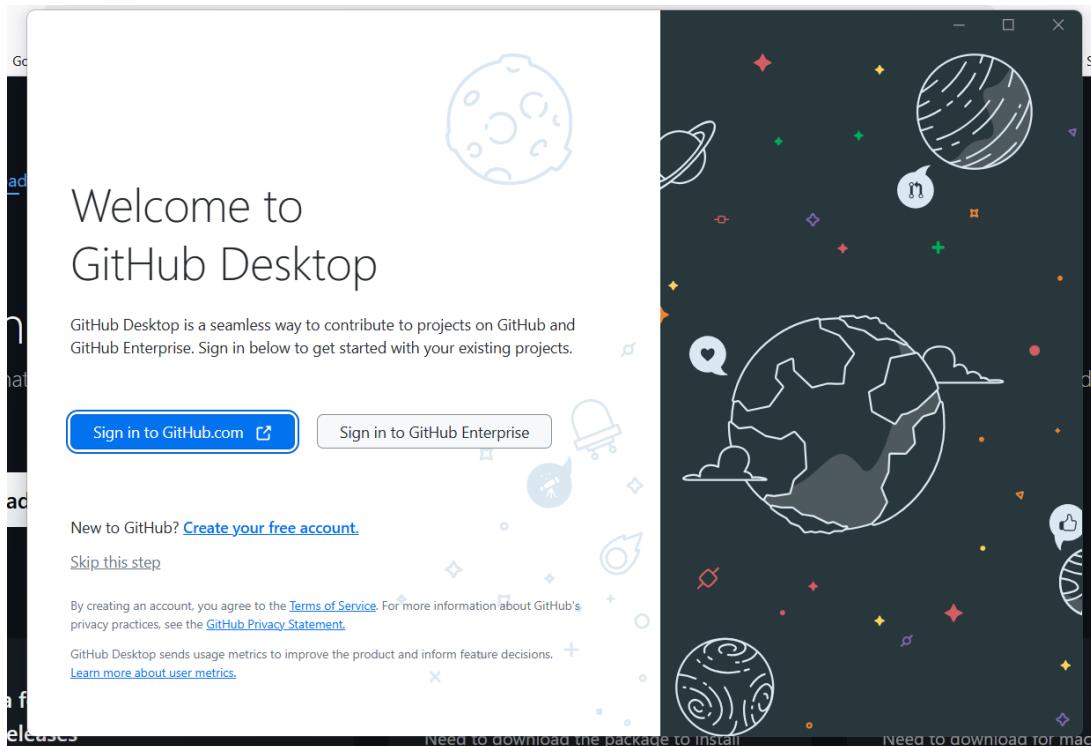
1	Wprowadzenie	2
1.1	Logowanie do GitHub Desktop	2
1.2	Klonowanie repozytorium	2
1.3	Widok repozytorium po klonowaniu	3
1.4	Wykrywanie zmian w plikach	4
1.5	Tworzenie commita	5
1.6	Historia commitów	6
1.7	Wysyłanie zmian do GitHuba	6
1.8	Tworzenie Pull Request	7
2	Podsumowanie	7

1 Wprowadzenie

Niniejsza instrukcja opisuje szczegółowo obsługę aplikacji GitHub Desktop. Przedstawia proces logowania, klonowania repozytorium, wprowadzania zmian, tworzenia commitów, synchronizacji z repozytorium zdalnym oraz tworzenia Pull Request.

1.1 Logowanie do GitHub Desktop

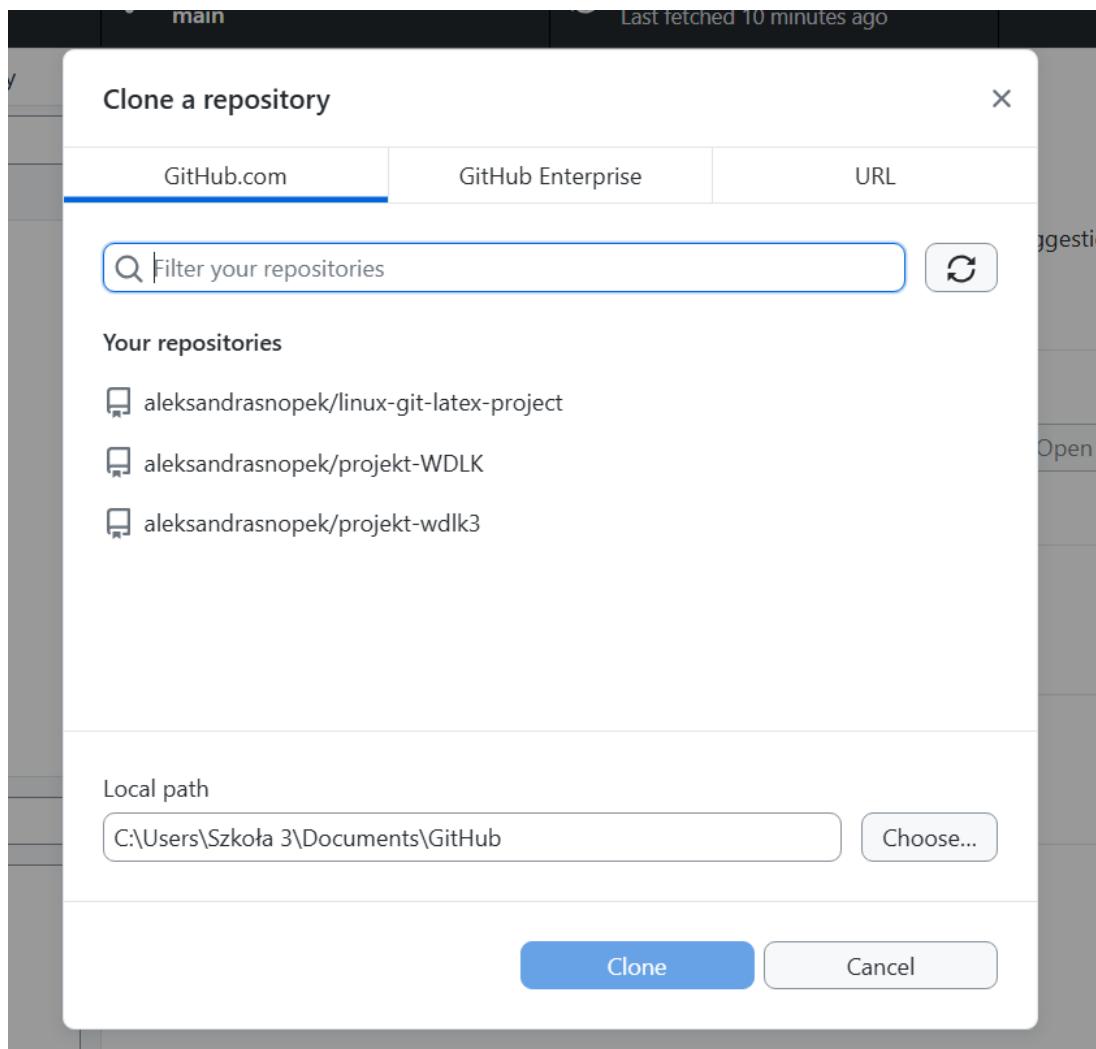
Po uruchomieniu aplikacji GitHub Desktop użytkownik musi zalogować się na swoje konto GitHub. Proces logowania polega na autoryzacji aplikacji poprzez przeglądarkę internetową. Po poprawnym zalogowaniu aplikacja uzyskuje dostęp do repozytoriów użytkownika.



Rysunek 1: Logowanie do aplikacji GitHub Desktop

1.2 Klonowanie repozytorium

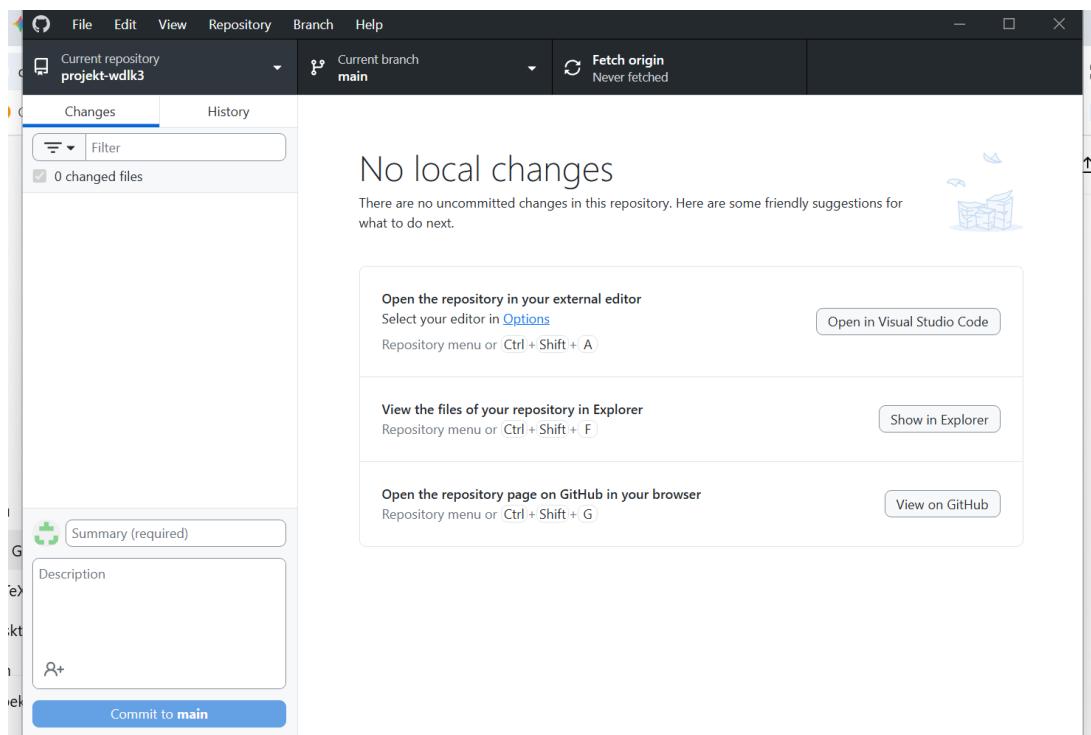
Po zalogowaniu użytkownik może sklonować istniejące repozytorium z platformy GitHub. W tym celu należy wybrać opcję *Clone a repository* oraz wskazać repozytorium, które ma zostać pobrane na komputer lokalny.



Rysunek 2: Okno klonowania repozytorium

1.3 Widok repozytorium po klonowaniu

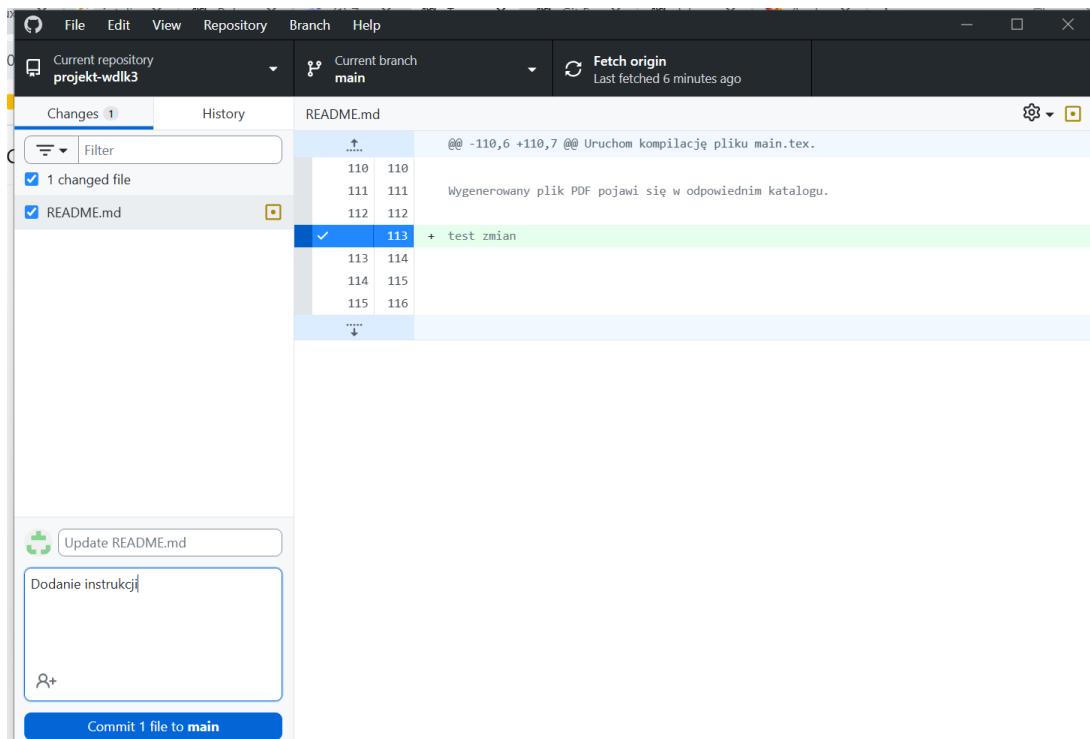
Po zakończeniu procesu klonowania aplikacja wyświetla główny widok repozytorium. Użytkownik może w nim zobaczyć aktualną gałąź, listę plików oraz historię zmian.



Rysunek 3: Widok repozytorium po sklonowaniu

1.4 Wykrywanie zmian w plikach

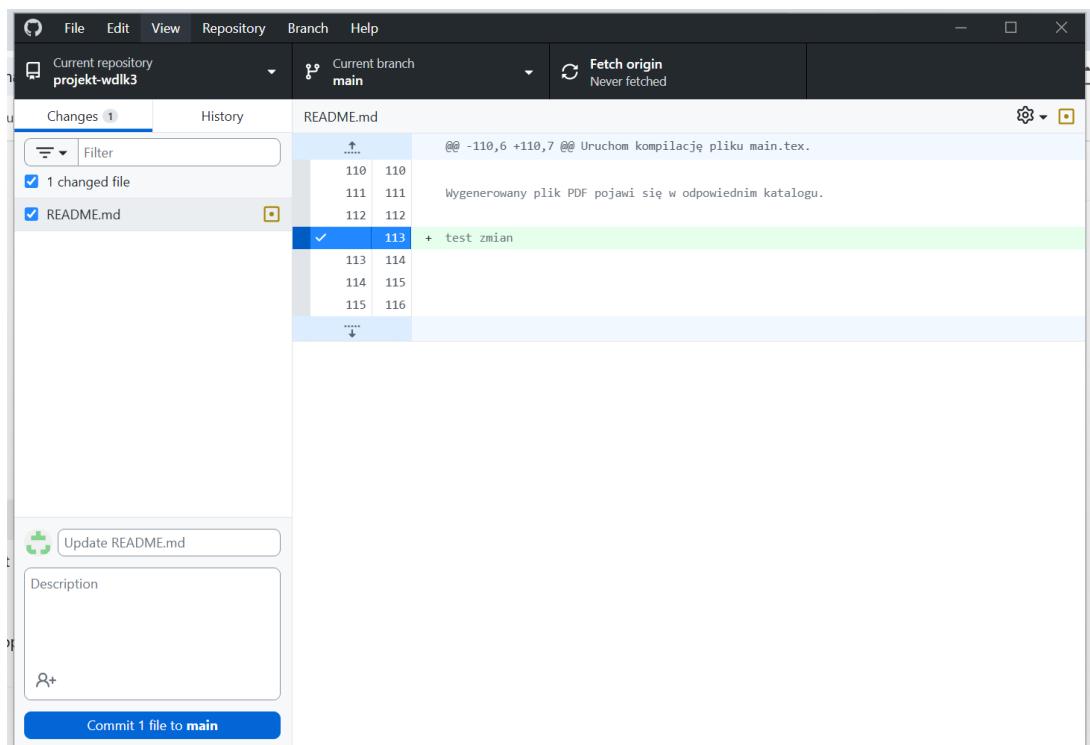
Po wprowadzeniu zmian w plikach projektu aplikacja GitHub Desktop automatycznie je wykrywa i prezentuje na liście zmian. Użytkownik może przejrzeć zmodyfikowane pliki przed wykonaniem commita.



Rysunek 4: Wykryte zmiany w plikach

1.5 Tworzenie commita

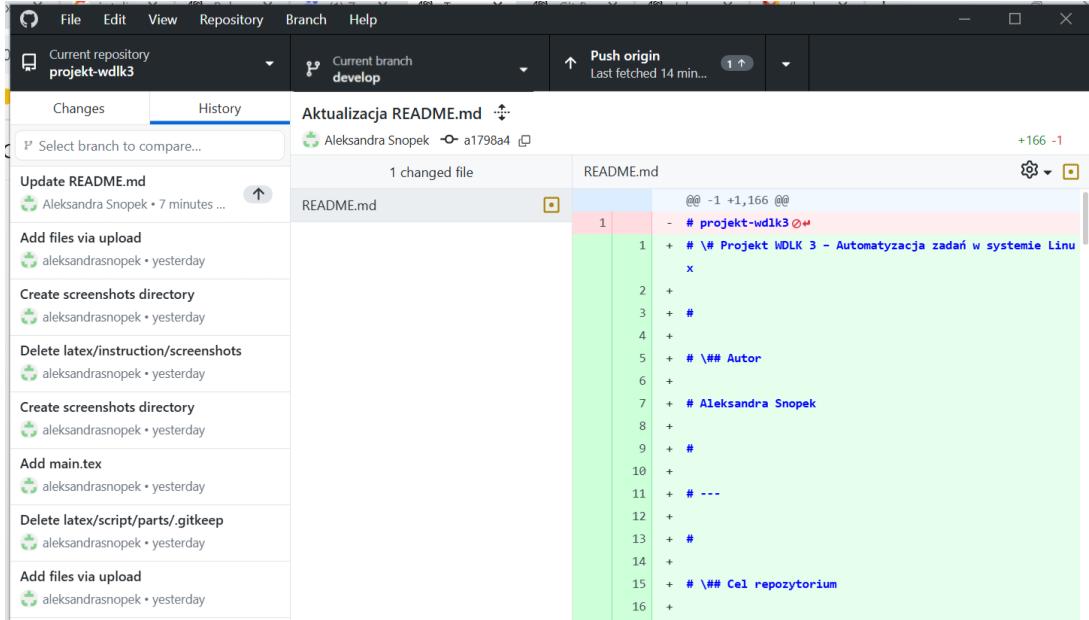
Aby zapisać zmiany w lokalnym repozytorium, użytkownik tworzy commit. W tym celu należy podać krótki opis zmian (commit message) oraz zatwierdzić operację.



Rysunek 5: Tworzenie commita w GitHub Desktop

1.6 Historia commitów

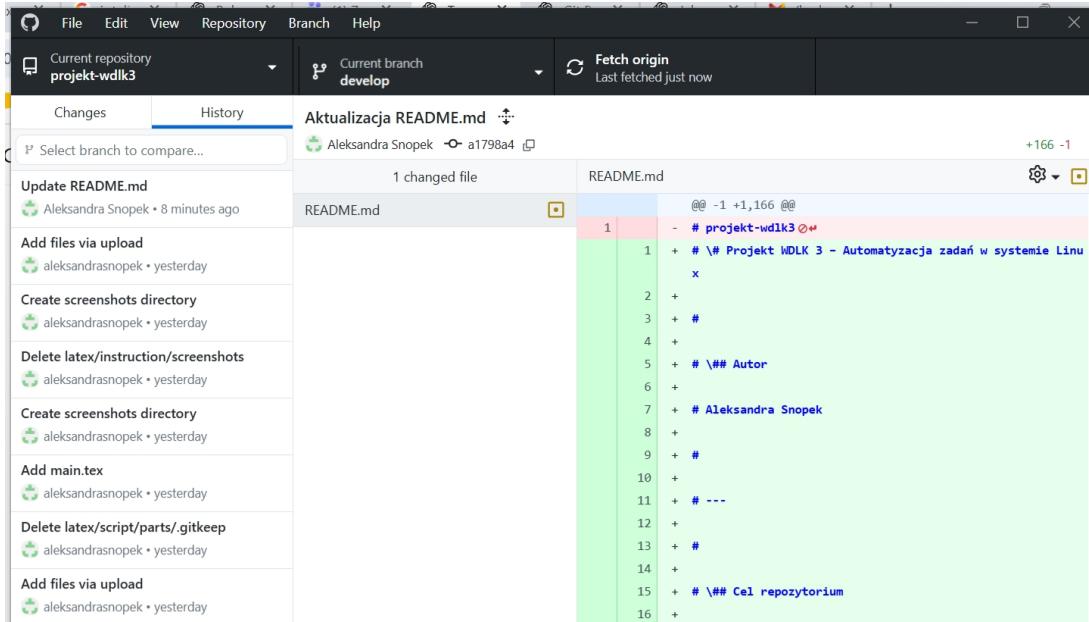
Aplikacja GitHub Desktop umożliwia przeglądanie historii commitów, co pozwala na analizę wprowadzonych zmian oraz identyfikację autorów poszczególnych operacji.



Rysunek 6: Historia commitów

1.7 Wysyłanie zmian do GitHuba

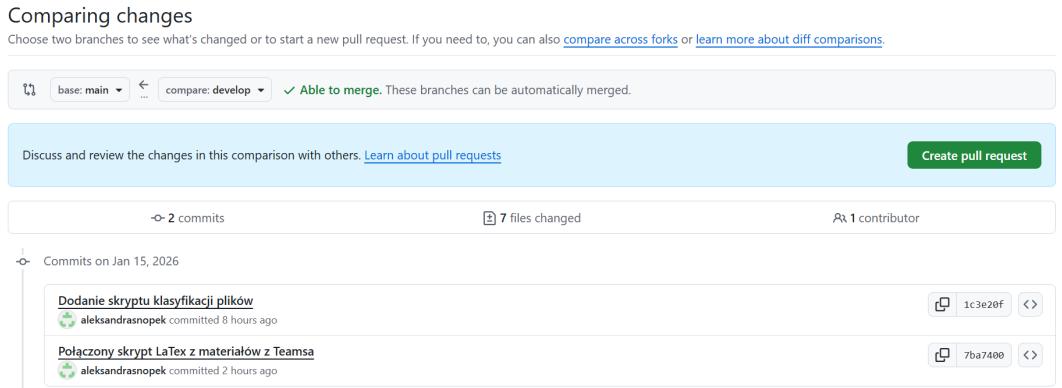
Po wykonaniu commitów zmiany mogą zostać wysłane do zdalnego repozytorium na platformie GitHub za pomocą operacji *Push*. Synchronizuje to lokalne zmiany z repozytorium zdalnym.



Rysunek 7: Wysyłanie zmian do repozytorium zdalnego

1.8 Tworzenie Pull Request

Pull Request służy do zgłoszenia zmian do połączenia z inną gałęzią repozytorium, najczęściej z gałęzią `develop` do `main`. Proces ten umożliwia przegląd zmian przed ich ostatecznym scaleniem.



Rysunek 8: Tworzenie Pull Request

2 Podsumowanie

GitHub Desktop umożliwia intuicyjną pracę z systemem kontroli wersji Git. Dzięki przejrzystemu interfejsowi użytkownik może efektywnie zarządzać repozytoriami bez konieczności korzystania z linii poleceń.