



Czym jest Git i dlaczego warto go używać?



Krótki przewodnik po systemie
kontroli wersji



Czym jest Git?

- Git to system kontroli wersji (VCS - Version Control System).
- Umożliwia śledzenie zmian w kodzie i zarządzanie nim.
- Stworzony przez Linusa Torvaldsa w 2005 roku.
- Jest rozproszony, co oznacza, że każdy użytkownik ma pełną kopię repozytorium.

Dlaczego Git?

1. Śledzenie zmian w plikach:
 - Co zostało zmienione, kiedy i przez kogo.
2. Praca zespołowa:
 - Wiele osób może pracować jednocześnie nad tym samym projektem.
3. Bezpieczeństwo:
 - Historia zmian jest przechowywana lokalnie i zdalnie.
4. Eksperymentowanie:
 - Gałęzie pozwalają na testowanie nowych funkcji bez ryzyka.
5. Rozwiązywanie konfliktów:
 - Ułatwia integrację zmian od różnych członków zespołu.

Jak działa Git?

1. Repository (repozytorium):
 - Miejsce, gdzie przechowywany jest kod i historia zmian.
2. Commit:
 - "Zrzut ekranu" zmian w kodzie.
3. Branch (gałąź):
 - Oddzielne "ścieżki" do pracy nad różnymi funkcjami.
4. Merge:
 - Łączenie zmian z różnych gałęzi.

Zalety Gita

1. Elastyczność:
 - Możliwość cofania się do dowolnej wersji kodu.
2. Praca offline:
 - Wszystkie operacje można wykonywać bez połączenia z internetem.
3. Popularność:
 - Standard w branży IT.
4. Integracja z narzędziami:
 - GitHub, GitLab, Bitbucket do współpracy i hostowania kodu.
5. Wsparcie automatyzacji:
 - CI/CD do testowania i wdrażania kodu.

Przykłady zastosowań Gita

1. Rozwijanie aplikacji mobilnych:

- Student A pracuje nad interfejsem, a student B nad logiką aplikacji. Dzięki Git mogą łączyć zmiany.

2. Praca nad projektem zespołowym:

- Każdy ma swoją gałąź na nowe funkcje i na końcu je łączy.
 - i. Bonus: po 3 dniach pracy nad daną funkcją, menedżer postanawia przerzucić nas do innego projektu “na chwilę” - możemy zostawić rozgrzebaną gałąź i wrócić do niej później.

3. Pisanie pracy dyplomowej:

- Śledzenie wersji tekstu, cofanie się do starszych wersji (w przypadku pisania pracy offline, nie w chmurze).

4. Tworzenie strony internetowej:

- Poprawianie błędów bez wpływania na działającą wersję strony.

Jak zacząć z Gitem?

1. Instalacja Gita

- a. Można pobrać z git-scm.com - w pracowni już zainstalowany.

2. Podstawowe komendy:

- a. `git init` - tworzy repozytorium.
- b. `git add` - dodaje pliki do śledzenia.
- c. `git commit` - zapisuje zmiany.
- d. `git push` - wysyła zmiany na zdalne repozytorium.

3. Stwórz konto na GitHub/GitLab/BitBucket/etc..

4. Zadanka dostępne na tinyurl.com/PUgit2024