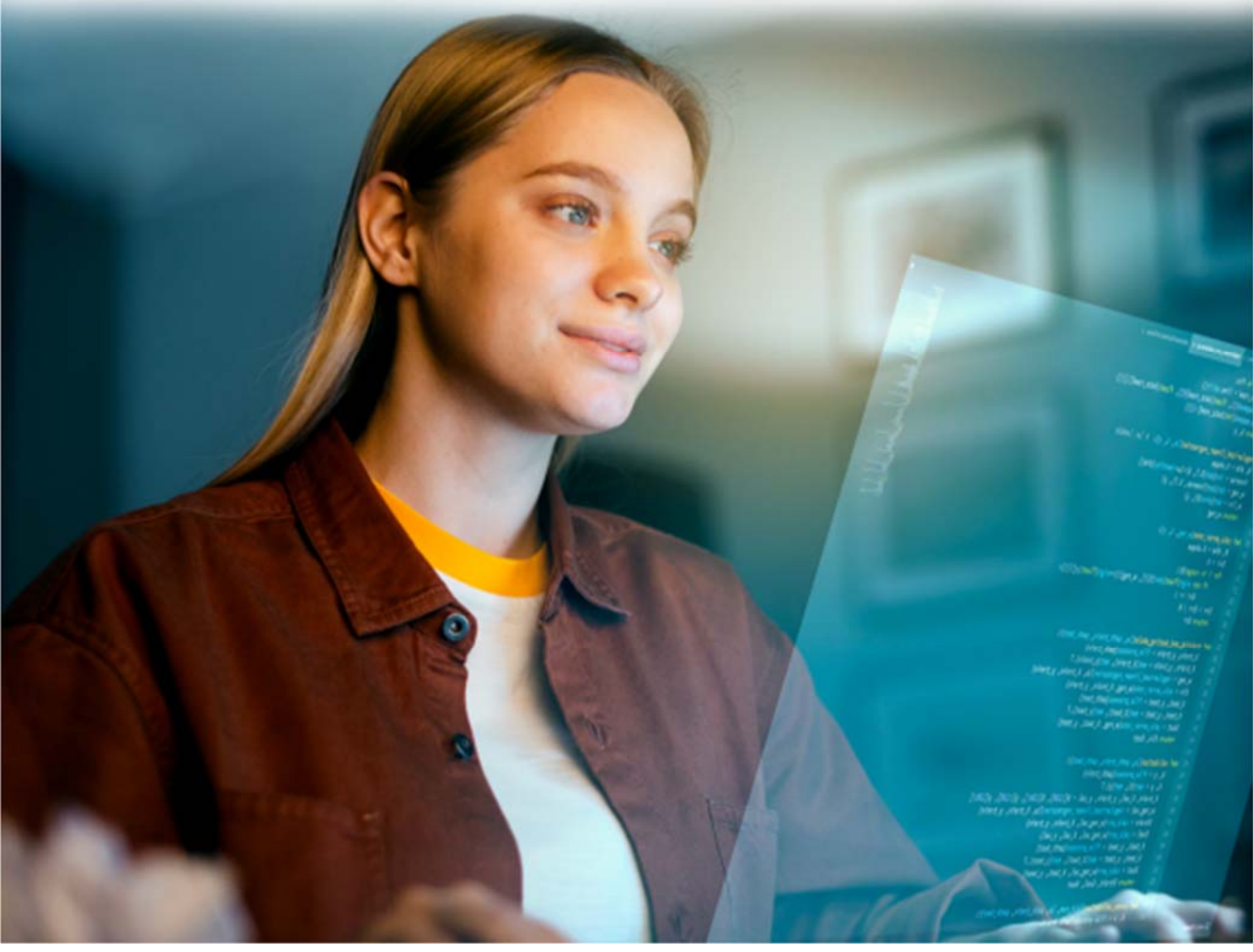


ROZDZIAŁ 3

SYSTEMY KONTROLI WERSJI



3 Systemy Kontroli Wersji

3.1 Wprowadzenie do Gita

Git to rozproszony system kontroli wersji, stworzony przez Linusa Torvaldsa w 2005 roku na potrzeby rozwoju jądra systemu Linux. Umożliwia:

- śledzenie zmian w plikach,
- cofanie się do wcześniejszych wersji projektu,
- równoczesną pracę wielu osób,
- synchronizację lokalnych zmian z repozytorium zdalnym,

Dla porównania: systemy takie jak SVN (Subversion) czy Mercurial działają w modelu scentralizowanym — cały projekt przechowywany jest na jednym głównym serwerze. Aby pobrać zmiany lub wysłać własne, programista musi mieć z nim stałe połączenie. Bez dostępu do serwera nie da się efektywnie pracować. Taki model nie sprzyja prywatnym eksperymentom — ściągasz najnowszą wersję (update), wprowadzasz zmiany i od razu je wysyłasz.

W Gicie każdy użytkownik posiada pełną kopię całego repozytorium, w tym całą historię zmian. Oznacza to, że większość operacji można wykonywać lokalnie, bez dostępu do internetu. Połączenie z internetem jest wymagane dopiero w momencie synchronizacji ze zdalnym repozytorium.

3.2 Historia Gita

Git powstał w 2005 roku, gdy projekt Linux Kernel potrzebował własnego, otwartego systemu do zarządzania wersjami, po utracie dostępu do poprzedniego narzędzia (BitKeeper) z powodu wprowadzenie opłat.

Autor szukał innego rozproszonego systemu kontroli wersji, opierając się na tych czterech kryteriach. Nie znalazł żadnego odpowiadającego tym wymaganiom.

- przykład CSV, czego nie robić,
- rozproszony,
- chroniony przed błędami w repozytorium (np. awaria dysku twardego),
- szybki,

Ze wszystkich dostępnych wtedy kontroli wersji po 2 kryteriach pozostał jedynie Monotone. Ten jednak był zbyt wolny w oczach Torvaldsa. Podjęto decyzję o stworzeniu nowego systemu kontroli wersji, który spełniałby wszystkie wymagania. Prace rozpoczęto 3 kwietnia 2005r. 16 czerwca Linux 2.6.12 był hostowany przez Gita. Jest on również przykładem że dobre projekty nie muszą długo powstawać - git powstał w 10 dni.

Od tamtej pory Git stał się standardem w świecie programistów, projektów open-source i firm IT. Git szybko zdobył popularność dzięki swojej wydajności, bezpieczeństwu i elastyczności. Na Gicie opierają się popularne platformy takie jak: GitHub, GitLab czy Bitbucket. Oprócz przechowywania kodu umożliwiając one współpracę zespołową, przeglądanie zmian (code review), automatyzację procesów (CI/CD) i zarządzanie projektami.

3.3 Instalacja i konfiguracja Gita

3.3.1 System Windows 11

W przeglądarce internetowej wpisz git-scm.com i naciśnij „Pobierz dla systemu Windows”



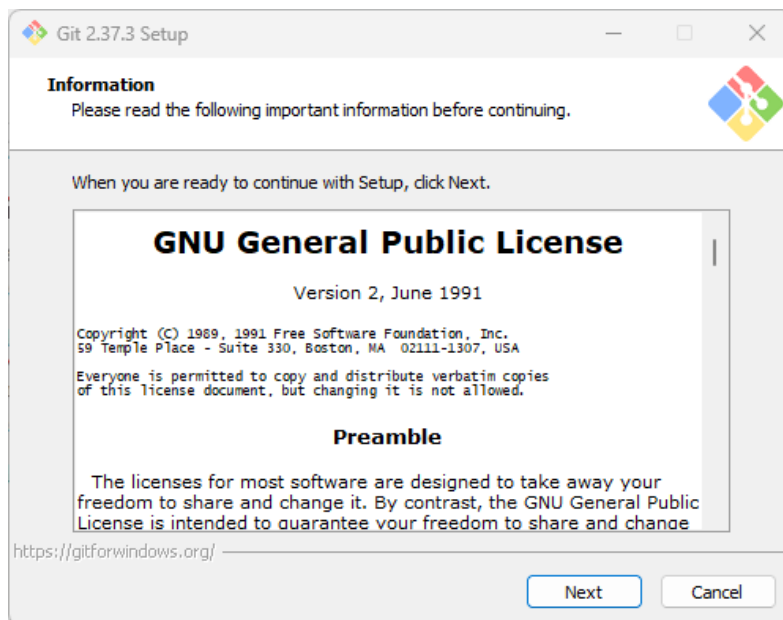
Rysunek 3.35 Pobieranie Gita z oficjalnej strony

Najnowsza wersja będzie widoczna w pierwszej linii. Naciskamy na „Kliknij tutaj, aby pobrać” rozpoczynając pobieranie.



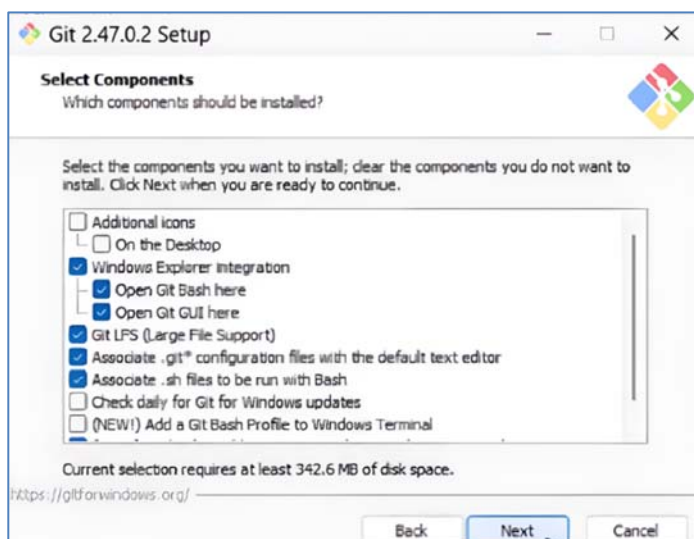
Rysunek 3.36 Rozpoczynanie pobierania Gita

Należy przeczytać licencje, nacisnąć „Next” oraz wybrać katalog w którym będzie zainstalowany Git po czym ponownie nacisnąć „Next”.



Rysunek 3.37 Licencja, którą należy przeczytać

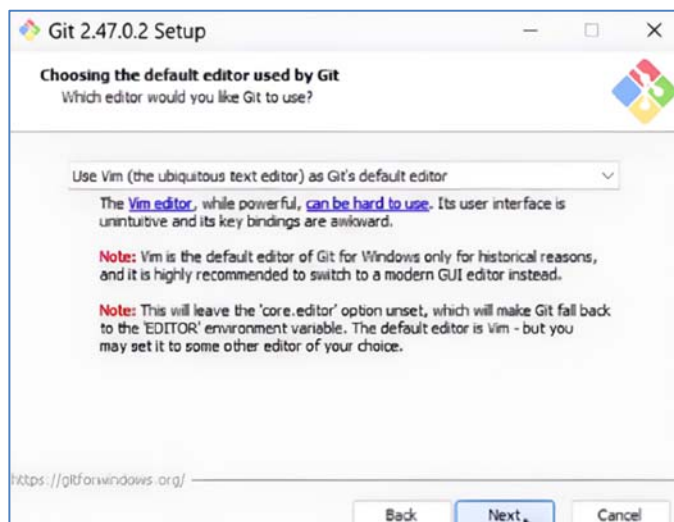
Kolejnym krokiem jest wybór dodatkowych komponentów, ja zostawiłam domyślne. Naciskamy „Next”.



Rysunek 3.38 Wybór dodatkowych komponentów

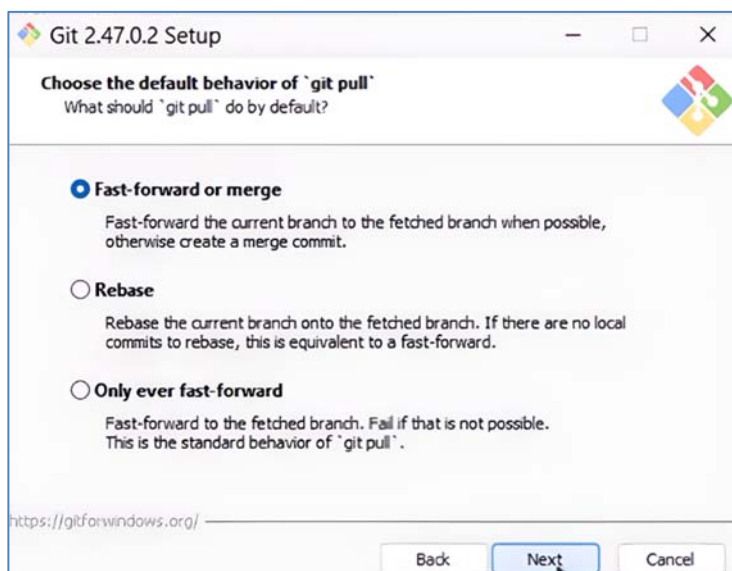
Pokazało nam się okno do wyboru folderu, potwierdziłam „Git” naciskając „Next”.

Następny krok jest istotny, ponieważ to tutaj wybierany jest domyślny edytor, który będzie się pokazywał np. kiedy nie podamy tytułu commita (zmian które wprowadziliśmy). Domyślny edytor można później zmienić. Osobiście lubię pracować z edytorem Vim, mimo że nie jest on z początku prosty w obsłudze. Po wyborze naciskamy „Next”.



Rysunek 3.39 Wybór domyślnego edytora w Gicie

Od lat główna gałąź na której będziemy pracować nazywa się „master”, jednak ze względu na to, że słowo „master” może kojarzyć się z „slave” co znaczy „niewolnik”, powoli główna gałąź nazywana jest „main”. Za pomocą komendy można później zmienić nazewnictwo głównej gałęzi. Zostawiłam opcję z „master” i taką nomenklaturą będę się głównie posługiwać. Naciskamy „Next”.



Rysunek 3.40 Wybór nazewnictwa głównej gałęzi

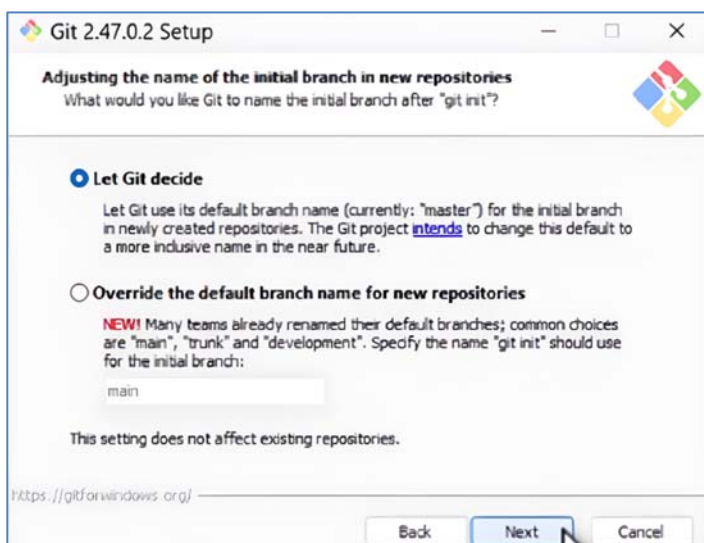
Następne pytanie dotyczy integracji Gita, co zostawiamy domyślne, aby móc otworzyć Gita z dowolnego miejsca na naszym komputerze, dalej „Next”.

Następne dwa pytania dotyczą bezpieczeństwa i połączenia z zewnętrznym repozytorium. Decyduję się zostawić obie domyślne opcje.

Kolejne pytanie dotyczy tego, w jaki sposób Git ma traktować koniec linii w plikach tekstowych. Pozostawiam domyślne.

W tym oknie decydujemy o rodzaju emulacji terminala, nic nie zmieniałam, naciskamy „Next”.

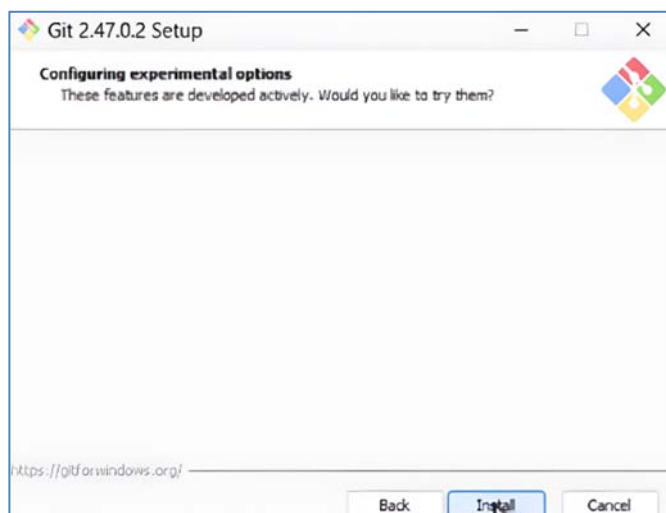
Następnie decydujemy o domyślnym sposobie wypychania zmian do zewnętrznego repozytorium. Do zrozumienia opcji wyboru potrzebne jest zapoznanie się z różnicą między git merge i git rebase, czemu poświęcam osobny podrozdział. Zostawiłam domyślne „Fast-forward or merge”.



Rysunek 3.41 Wybór domyślnego wypychania zmian

Ostatnie pytanie dotyczy optymalizacji, co również pozostawiłam domyślne. Naciskamy „Next”

Naciskamy „Install” kończąc konfigurowanie Gita, zaczynając jego instalację.



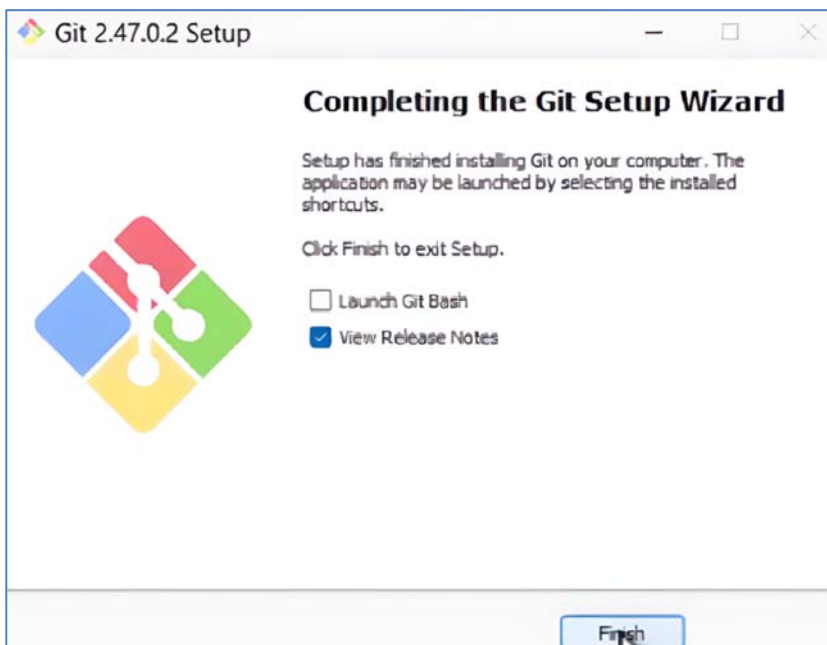
Rysunek 3.42 Zakończenie konfiguracji poprzez naciśnięcie przycisku „Install”

Po zakończeniu pobierania, naciskamy „Finish”.

W kolejnym kroku należy uruchomić terminal, aby upewnić się, że git został poprawnie zainstalowany – na przykład wpisując komendę **git --version**, po poprawnym wykonaniu tej komendy powinna się pokazać aktualna wersja gita zainstalowana na komputerze

```
C:\Users\kinga>git --version  
git version 2.42.0.windows.2
```

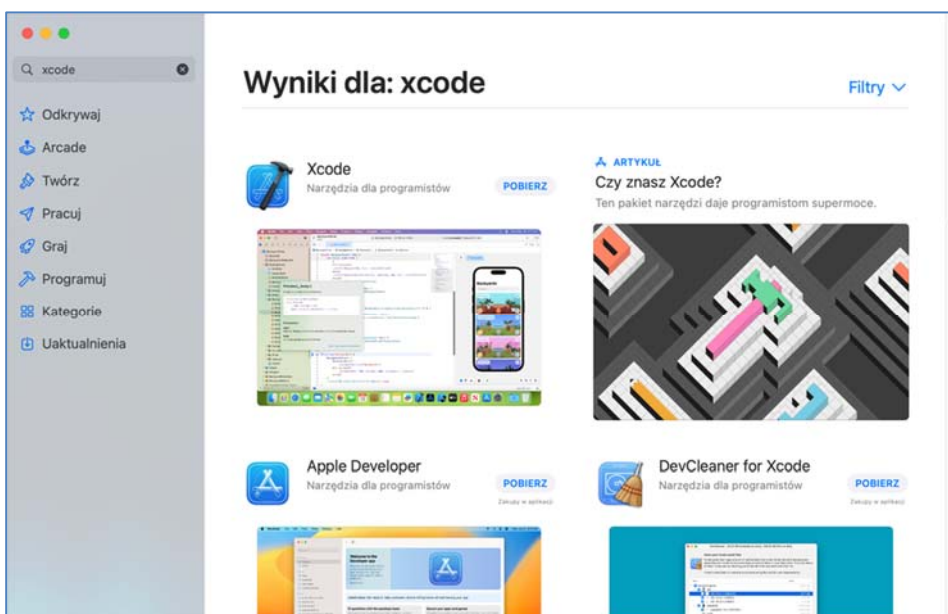
Rysunek 3.43 Moja aktualna wersja Gita



Rysunek 3.44 Zakończenie instalacji poprzez naciśnięcie przycisku „Finish”

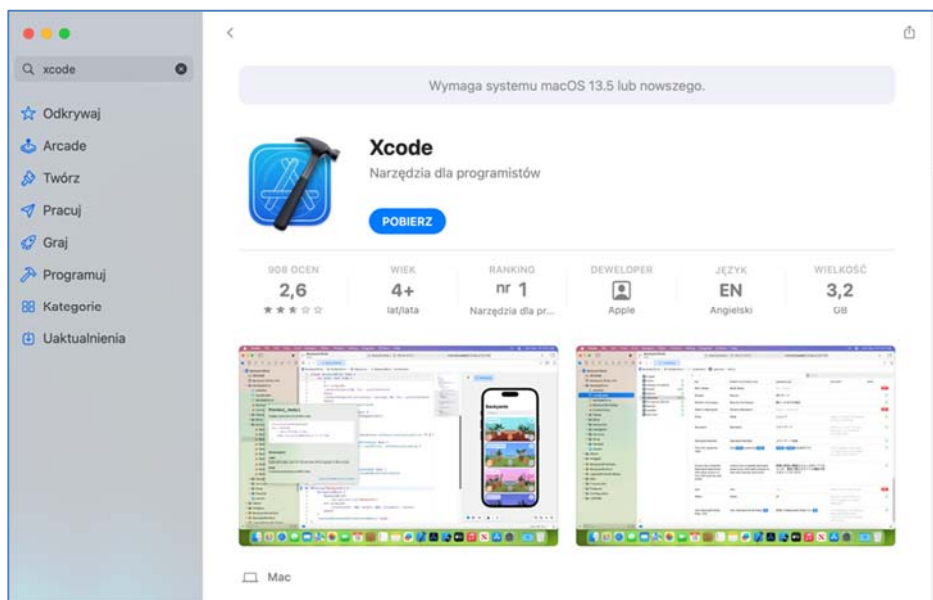
3.3.2 System MacOS

Najłatwiejszym sposobem zainstalowania Gita jest zainstalowanie narzędzia Xcode, które jest dostarczane między innymi z Git. Xcode możesz pobrać bezpośrednio z AppStore. Najlepiej po prostu je wyszukać:



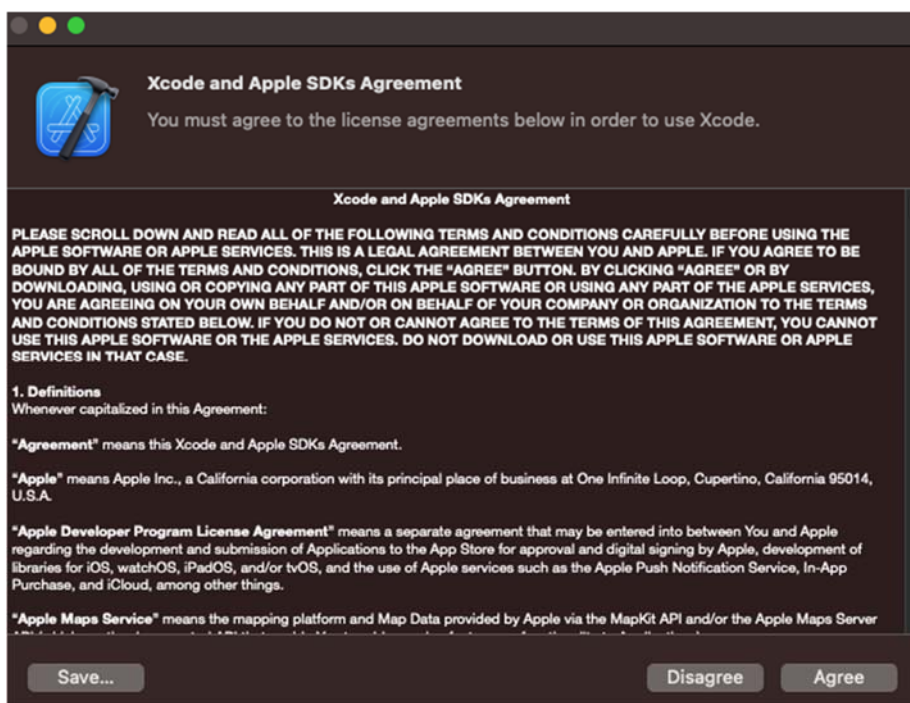
Rysunek 3.45 Wyszukiwanie Xcode w AppleStore

Aby zainstalować Xcode na swoim komputerze wystarczy kliknąć przycisk „Pobierz”, a aplikacja pobierze się i automatycznie rozpocznie się proces instalacji.



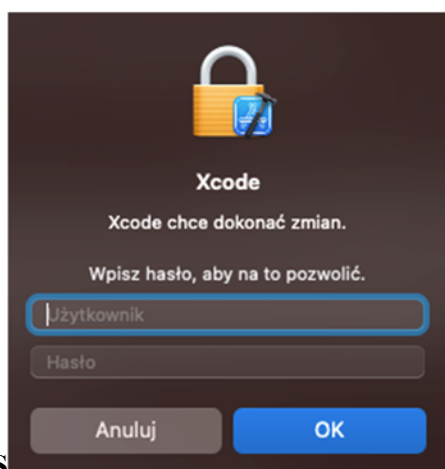
Rysunek 3.46 Rozpoczęcie pobierania poprzez naciśnięcie przycisku „pobierz”

Podczas instalacji oprogramowania Xcode należy zaakceptować warunki korzystania z tego programu.



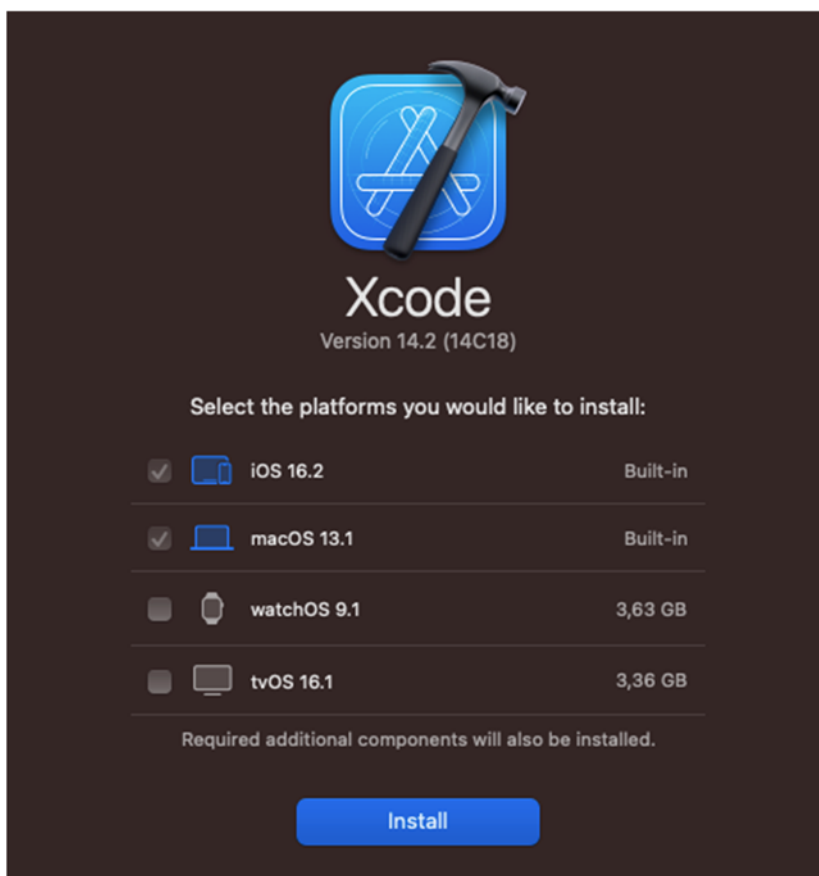
Rysunek 3.47 Akceptowanie warunków korzystania z programu

Program Xcode będzie prosił o pozwolenie na dokonywanie zmian. Należy podać hasło, aby na to pozwolić.



Rysunek 3.48 Prośba o pozwolenie na dokonywanie zmian

Po zainstalowaniu środowiska Xcode zostanie przeprowadzona konfiguracja wstępna, podczas której użytkownik musi zainstalować wtyczki, które będą pomocne przy implementacji aplikacji na konkretne urządzenia Apple. Z racji tego, że instalujemy Xcode'a tylko dlatego, że potrzebujemy tylko Gita, nie ma sensu wybierać coś więcej niż opcje oznaczone jako „Built-in”. W kolejnym kroku należy kliknąć przycisk „Install”.



Rysunek 3.49 Pozostawienie domyślnie instalowanych wtyczek

W kolejnym kroku zostanie automatycznie uruchomione okno, w którym można stworzyć nowy projekt Xcode. Oznacza to, że Xcode wraz z dodatkami został poprawnie zainstalowany.