

Wstęp

React Native to framework stworzony przez Facebooka, który umożliwia tworzenie aplikacji mobilnych na platformy **iOS** i **Android** przy użyciu tego samego kodu w języku **JavaScript**. Framework pozwala na budowanie natywnych aplikacji mobilnych, które wykorzystują natywne komponenty systemowe, dzięki czemu osiąga się wydajność bliską aplikacjom napisanym w językach natywnych, takich jak Swift (iOS) czy Kotlin (Android).

Główne Cechy React Native

1. **Cross-Platform Development (Tworzenie międzyplatformowe):**
 - React Native pozwala tworzyć aplikacje na **iOS** i **Android** przy użyciu jednego kodu źródłowego.
 - Zmniejsza to czas i koszty potrzebne do budowy aplikacji na dwie różne platformy.
2. **Natywny wygląd i wydajność:**
 - Aplikacje napisane w React Native korzystają z natywnych komponentów systemu operacyjnego, dzięki czemu interfejs wygląda i działa jak aplikacja natywna.
 - Wykorzystanie **mostu JavaScript-to-Native** zapewnia szybki dostęp do funkcji systemowych.
3. **Modularna architektura:**
 - React Native pozwala na podział aplikacji na mniejsze moduły, co ułatwia iteracyjne tworzenie aplikacji i umożliwia pracę wielu zespołów równocześnie.
4. **Hot Reloading:**
 - Umożliwia natychmiastowe podglądy zmian w aplikacji w czasie rzeczywistym, co znacznie przyspiesza proces programowania i debugowania.
5. **Rozległy ekosystem bibliotek i społeczności:**
 - Duża społeczność zapewnia bogaty wybór gotowych bibliotek, takich jak nawigacja (React Navigation), zarządzanie stanem (Redux, Zustand) czy integracja z funkcjami urządzenia (kamera, GPS).

Motywacja

Tworzenie aplikacji Habit Tracker w React Native jest uzasadnione ze względu na kilka kluczowych zalet:

1. Oszczędność czasu i zasobów

- Habit Tracker musi działać na **iOS** i **Android**, aby dotrzeć do szerokiej grupy użytkowników.
- Dzięki jednemu kodowi źródłowemu z React Native redukujemy czas i koszty tworzenia aplikacji, ponieważ nie trzeba tworzyć oddzielnych aplikacji natywnych.

2. Łatwa iteracja funkcjonalności

- Przy projektach takich jak Habit Tracker, które ewoluują w oparciu o feedback użytkowników, **hot reloading** i szybki cykl wdrożeń w React Native pozwalają szybko wprowadzać zmiany i ulepszenia.

3. Intuicyjne UI i UX

- Habit Tracker wymaga prostego i responsywnego interfejsu użytkownika, w którym użytkownik łatwo dodaje nawyki, przegląda statystyki i ustawia przypomnienia.
- React Native, dzięki swojej kompatybilności z natywnymi komponentami, umożliwia tworzenie interfejsów, które są szybkie i intuicyjne.

4. Integracja z funkcjami urządzenia

- Funkcje takie jak **przypomnienia push**, **powiadomienia lokalne**, **integracja z kalendarzem**, czy dostęp do czujników (np. krokomierz) są łatwo implementowalne za pomocą bibliotek React Native, takich jak:
 - `react-native-push-notifications` (powiadomienia push),
 - `react-native-calendar-events` (integracja z kalendarzem).

5. Skalowalność

- Habit Tracker może zacząć jako prosta aplikacja z podstawowymi funkcjami, a następnie rozwinąć się o zaawansowane opcje, takie jak personalizacja powiadomień, funkcje społecznościowe czy integracja z zewnętrznymi aplikacjami. React Native jest wystarczająco elastyczny, aby wspierać takie zmiany.

6. Rozbudowana społeczność i wsparcie

- Popularność React Native oznacza, że programiści mogą łatwo znaleźć rozwiązania problemów i korzystać z gotowych bibliotek dla funkcji habit trackera, np. wizualizacji statystyk (`react-native-chart-kit`).