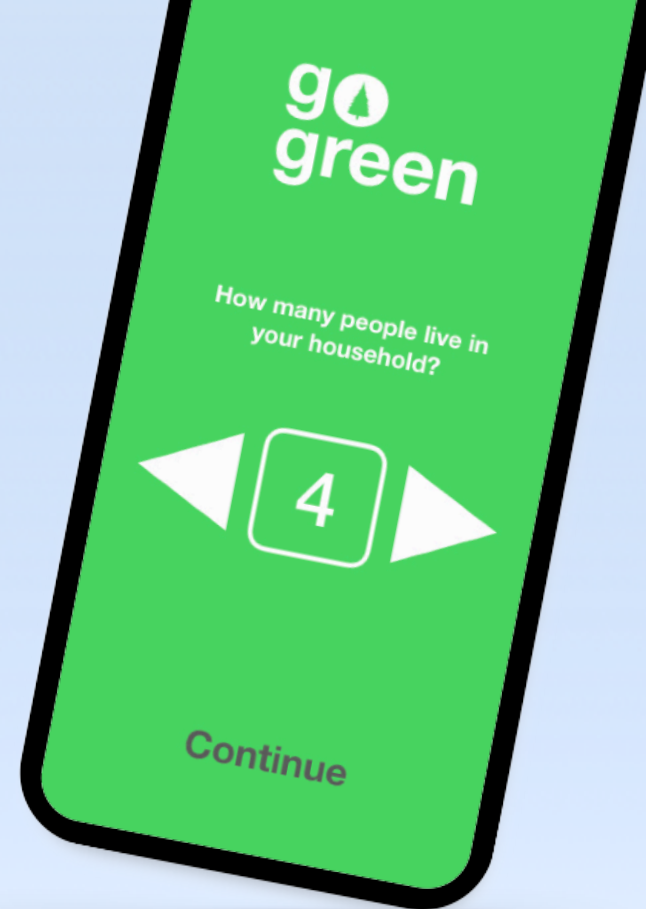




Swift Coding Club



Xcode

Projekty polegające na tworzeniu aplikacji

Projekty

1	SinglePhoto	3
2	QuestionBot	4
3	ChatBot	5
4	ColorMix	6
5	ElementsQuiz	7
6	AnimalSounds	8
7	RockPaperScissors	9
8	MemeMaker	10

Witaj w Swift Coding Club!

Dołączając do klubu programistów, rozpoczynasz przygodę, podczas której stworzysz swoje pierwsze aplikacje. Nie przejmuj się, jeśli dopiero zaczynasz naukę programowania. Zaprezentowane projekty pozwolą Ci zapoznać się z podstawowymi pojęciami związanymi z programowaniem i tworzeniem aplikacji. Każdy projekt opiera się na poprzednim i skupia wokół innych aspektów aplikacji. Dzięki temu pod koniec będziesz posiadać umiejętności pozwalające stworzyć Twoją własną aplikację.

Realizacja projektów wymaga kursu [Intro to App Development with Swift](#). Upewnij się, że spełniasz wymagania wstępne każdego z nich.



SinglePhoto

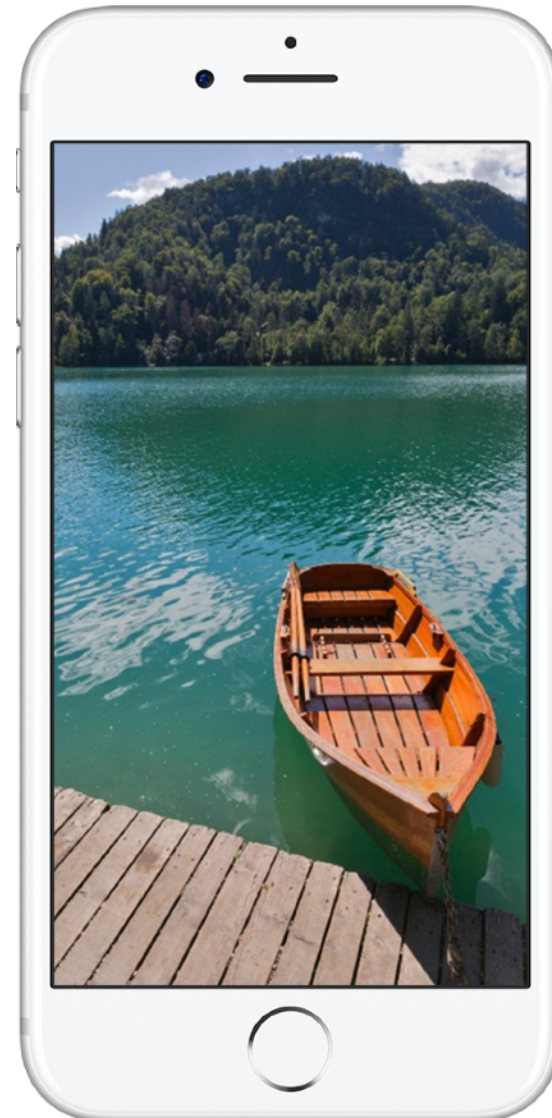
1

Wymagania wstępne: brak

Czas: 1,5 godziny

Przejdź: [Lekcja 5](#): Pierwsza aplikacja

Zobacz, jak łatwo można zbudować swoją pierwszą aplikację. SinglePhoto nie tylko wyświetla treść — na przykład zdjęcie — ale i reaguje po stuknięciu ekranu przez użytkownika. To prosty rodzaj interakcji, który stanowi fundament wielu aplikacji. Z pewnością przyda się podczas zdobywania kolejnych umiejętności związanych z programowaniem i tworzeniem aplikacji. W ramach tego projektu zapoznasz się także z narzędziami Xcode, Interface Builder i Simulator oraz dowiesz, jak je wykorzystywać do budowania własnych aplikacji.



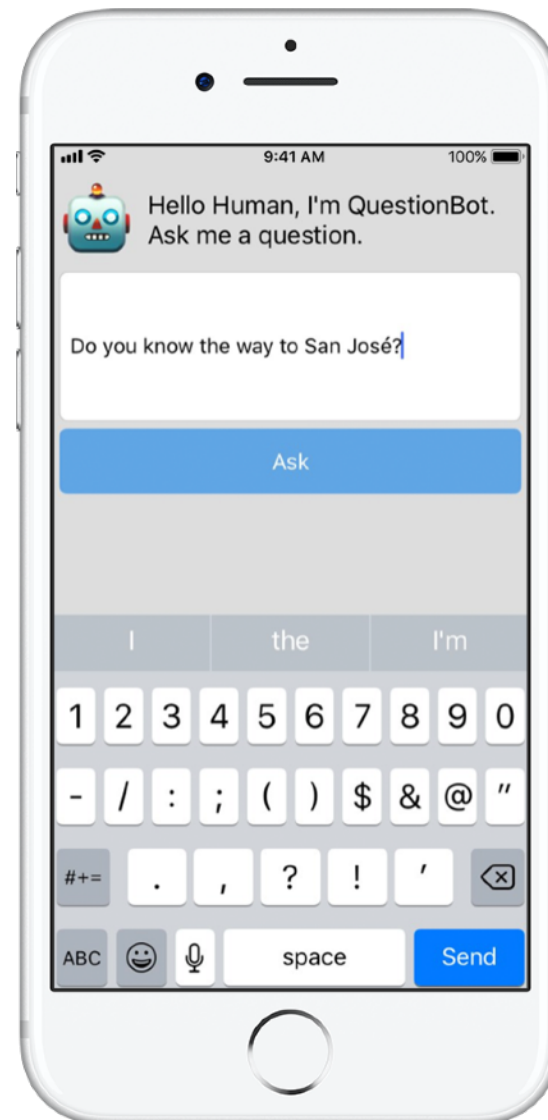
Wymagania wstępne:

- Aplikacja SinglePhoto
- Lekcja 2: Nazwy i identyfikatory
- Lekcja 3: Ciągi znaków
- Lekcja 6: Funkcje
- Lekcja 10: Parametry i wyniki
- Lekcja 11: Podejmowanie decyzji

Czas: 6,5 godziny

Przejdź: [Lekcja 13: QuestionBot](#)

Korzystasz czasem z aplikacji quizowych albo zastanawiasz się, jak działa Siri? Wiele aplikacji funkcjonuje, jakby miały własny „mózg”. Zbudujesz aplikację QuestionBot, w której inteligentny bot udziela adekwatnych odpowiedzi na różne pytania. Ale najpierw dowiesz się między innymi, jak zapisywać wartości w postaci stałych, przekładać tekst na kod czy określać dane wejściowe i wyjściowe. Realizacja tego projektu pozwoli Ci lepiej zrozumieć działanie aplikacji i logikę stojącą za jej „inteligencją”.



Wymagania wstępne:

- Aplikacja QuestionBot
- Lekcja 14: Tablice i pętle
- Lekcja 15: Definiowanie struktur

Czas: 3,5 godziny

Przejdź: [Lekcja 16: QuestionBot 2](#)

Czas sprawić, by aplikacja QuestionBot działała jeszcze inteligentniej. Na pewno będziesz z niej korzystać wiele razy, tak więc przyda się rozszerzyć ją o możliwość „zapamiętywania” poprzednich interakcji. ChatBot to udoskonalona wersja aplikacji QuestionBot, która zapisuje historię wiadomości wymienianych przez użytkownika z botem. Aplikacja ChatBot jest już po części zbudowana, dzięki czemu możesz skoncentrować się na zdobywaniu różnych umiejętności wymaganych do stworzenia tych elementów, które pozwolą na śledzenie konwersacji.

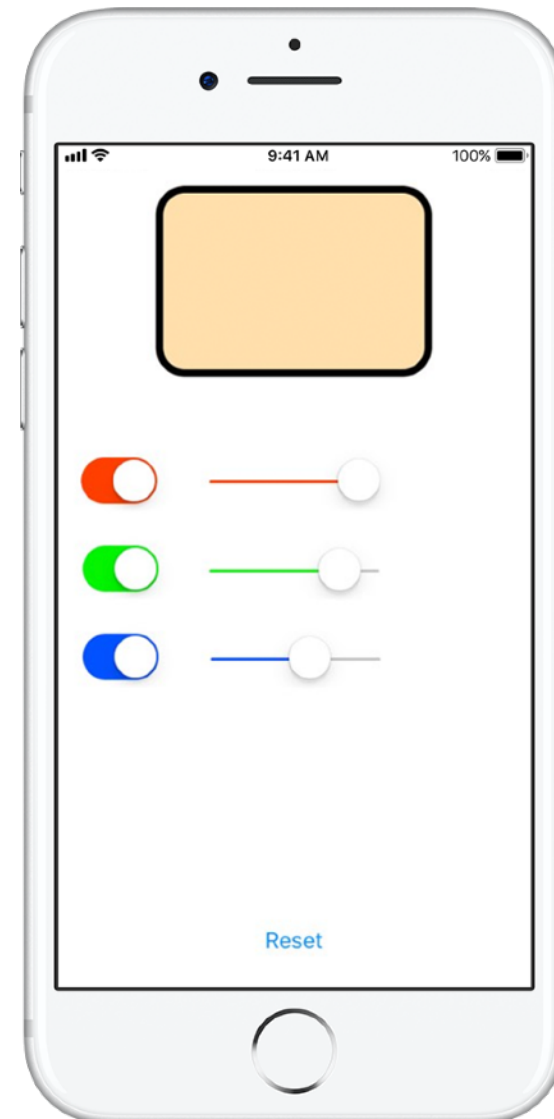


Wymagania wstępne: aplikacja ChatBot

Czas: 1,5 godziny

Przejdź: [Lekcja 17](#): Działania i rezultaty

Zastanów się nad interfejsem użytkownika (User Interface — UI). Jak dotąd udało Ci się zbudować aplikacje, w których można stukać przyciski i wprowadzać tekst. Tworząc aplikację ColorMix, nauczysz się umieszczać przełączniki i suwaki w interfejsie użytkownika. Co więcej, dowiesz się, jak połączyć te wizualne elementy interfejsu z językiem Swift, tak aby działały w oczekiwany sposób. W efekcie Twojej pracy zbudujesz aplikację ColorMix, która generuje różne kolory, mieszając czerwień, zieleń i błękit.

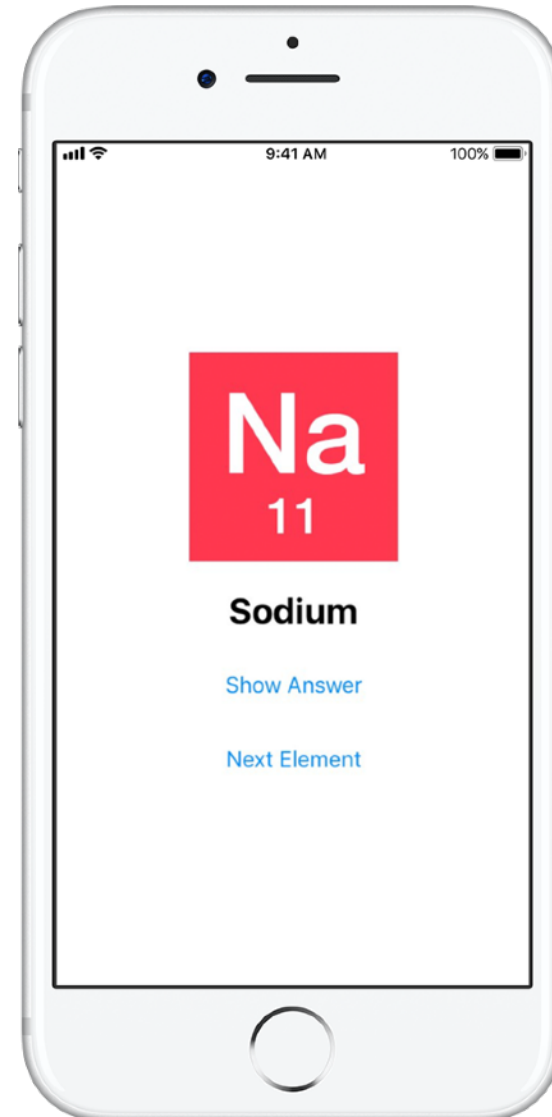


Wymagania wstępne: aplikacja ColorMix

Czas: 2 godziny

Przejdź: [Lekcja 18](#): Adaptacyjne interfejsy użytkownika

Większość osób pobiera aplikacje w celu rozwiązania konkretnego problemu — na przykład uporządkowania plików, rozliczenia finansów czy uzyskania wskazówek. Zbuduj aplikację ElementsQuiz, która pomaga uczniom w nauce układu okresowego pierwiastków. Wykorzystasz widoki stosu do łatwego umieszczenia wielu widoków naraz na jednym ekranie. Będą one dostosowywać się do różnego rozmiaru ekranu.

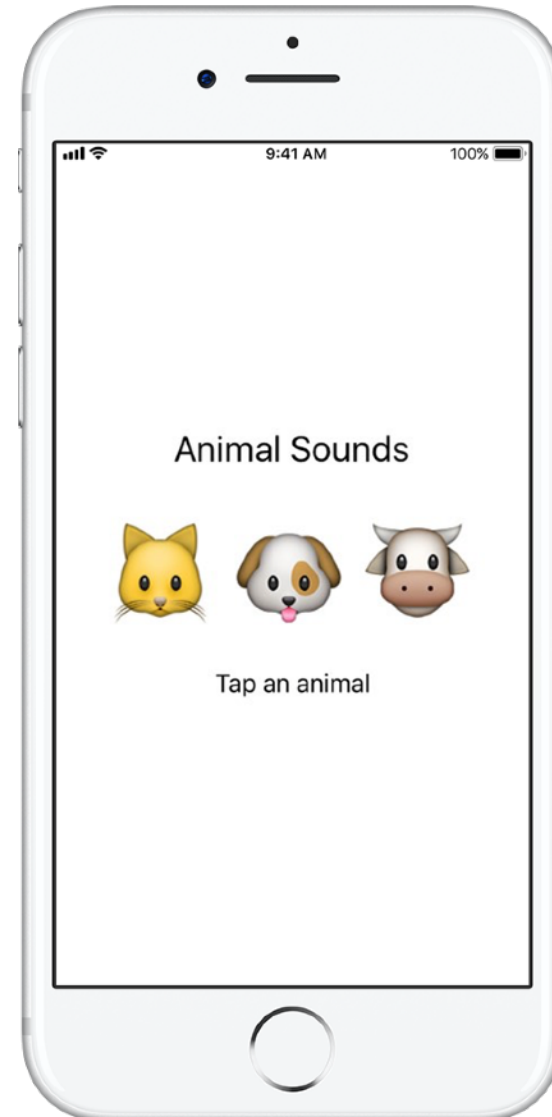


Wymagania wstępne: aplikacja ElementsQuiz

Czas: 1 godzina

Przejdź: [Lekcja 18](#): Adaptacyjne interfejsy użytkownika

W miarę jak aplikacje stają się coraz bardziej złożone, z pewnością zastanawiasz się „Czy cały ten kod trzeba pisać samodzielnie? Czy nikt nie napisał go wcześniej?”. Podczas pracy nad aplikacją AnimalSounds wykorzystasz wcześniej stworzony kod, który odpowiada za odtwarzanie dźwięków, i nauczysz się go z nią integrować. Następnie użyjesz widoków stosu w innych widokach stosu, aby zbudować najbardziej zaawansowany interfejs użytkownika ze wszystkich dotychczasowych.



RockPaperScissors

7

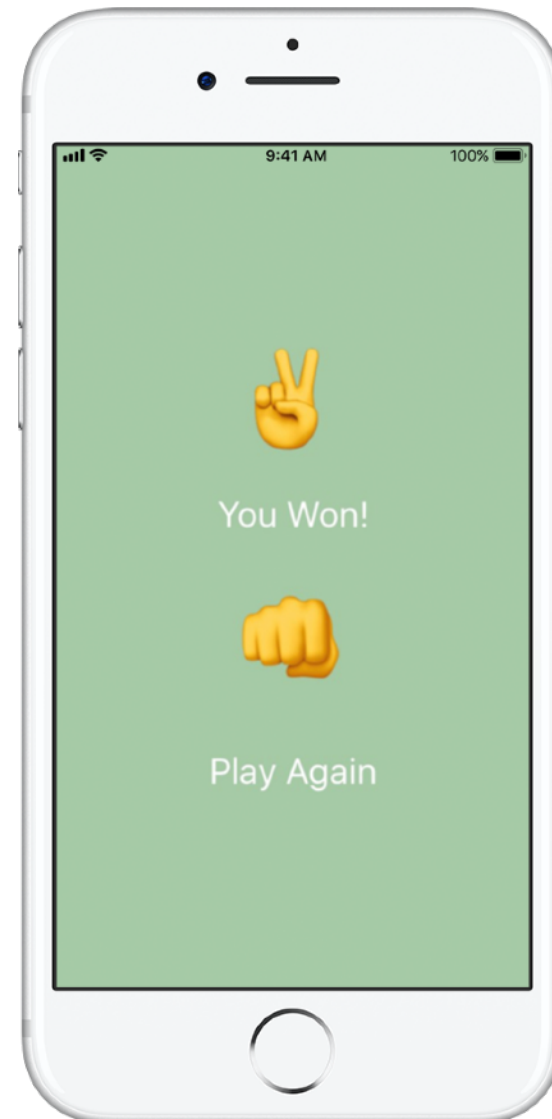
Wymagania wstępne:

- Aplikacja AnimalSounds
- Lekcja 19: Listy

Czas: 1,5 godziny

Przejdź: [Lekcja 20](#): Projekt końcowy

Użytkownikowi często trzeba udostępnić listę opcji do wyboru, na przykład wskazówek podróży czy rodzajów kuchni. Podczas realizacji tego projektu nauczysz się tworzyć trzy unikatowe opcje niezbędne do gry w Kamień, papier, nożyce. Dzięki temu użytkownik będzie mógł bez końca mierzyć się z komputerem. Ponadto dowiesz się, jak generować losowe liczby, tak aby umożliwić komputerowi losowy wybór opcji.



Wymagania wstępne:

- aplikacja AnimalSounds
- Lekcja 19: Listy

Czas: 1,5 godziny

Przejdź: [Lekcja 20](#): Projekt końcowy

Mem internetowy to zazwyczaj nieruchomy obraz z tekstem w górnej lub dolnej części. Do zbudowania aplikacji MemeMaker wykorzystasz segmentowane elementy sterujące, które pozwalają wyświetlać różne napisy nad i pod obrazem. A ponieważ te elementy sterujące działają niezależnie, możesz wybierać i dopasowywać napisy, tworząc swoje własne zestawienia.

