

# Amazon Alexa (1)

## Tworzymy umiejętności Skills

W artykule zaprezentujemy sposób tworzenia „umiejętności” dla asystentki głosowej Alexa (skills). Wykonamy i opublikujemy umiejętność, dzięki której Alexa będzie symulowała losowanie kostką do gry w grach planszowych.

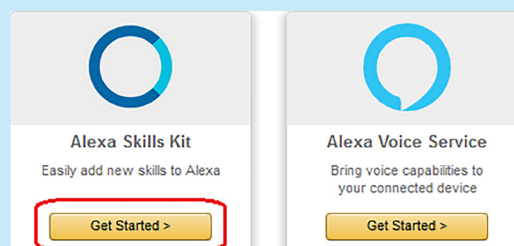
W momencie pisania artykułu liczba umiejętności obsługiwanych przez asystentkę głosową Alexa przekroczyła 25 tysięcy. Ciągłe powstają nowe umiejętności zwiększające funkcjonalność asystentki. Umiejętności tworzone są przez firmy, które integrują własne produkty i usługi z asystentką Alexa (samochody, sprzęt AGD, zakupy w sklepach internetowych, itd.). Umiejętności mogą być również tworzone na własne potrzeby przez użytkowników asystentki Alexa.

Dla ułatwienia tworzenia umiejętności firma Amazon przygotowała narzędzia dla developerów oraz liczne kursy i przewodniki. Centrum wiedzy o tworzeniu umiejętności jest portal <http://developer.amazon.com>. W portalu firma Amazon udostępnia szablony ułatwiające budowanie własnych umiejętności Skills. Opis i podstawowe funkcje szablonów prezentuje tabela 1. Przed rozpoczęciem tworzenia umiejętności dla asystentki Alexa należy zdecydować, z którego rodzaju szablonu będziemy korzystali. My użyjemy najbardziej ogólnego i dającego największe możliwości szablonu *custom skills*. Następnie, używając portalu <http://developer.amazon.com> zdefiniujemy interfejs głosowy użytkownika (*Voice User Interface*). Napiszemy kod programu do obsługi umiejętności (program w języku JavaScript). Uruchomimy

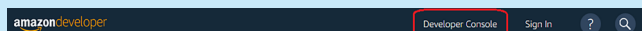
funkcję **Lambda** do obsługi programu (użycie portalu *Amazon Web Services*) oraz przeprowadzimy testy utworzonej umiejętności. Na zakończenie, opublikujemy utworzoną umiejętność w serwisie Amazon.

### Voice User Interface

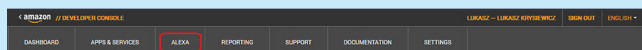
Interfejs głosowy użytkownika tworzymy z użyciem portalu developera. Uruchamiamy stronę internetową <http://developer.amazon.com>, wybieramy uruchomienie konsoli zarządzania i logujemy



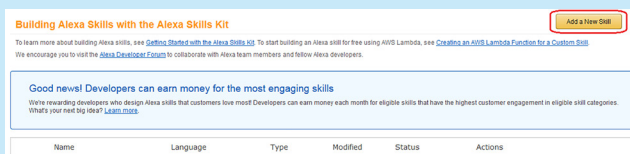
Rysunek 3. Amazon developer. Uruchomienie ASK



Rysunek 1. Amazon developer. Logowanie do serwisu



Rysunek 2. Amazon developer. Uruchomienie serwisu Alexa



Rysunek 4. Amazon developer. Dodanie umiejętności do konta użytkownika

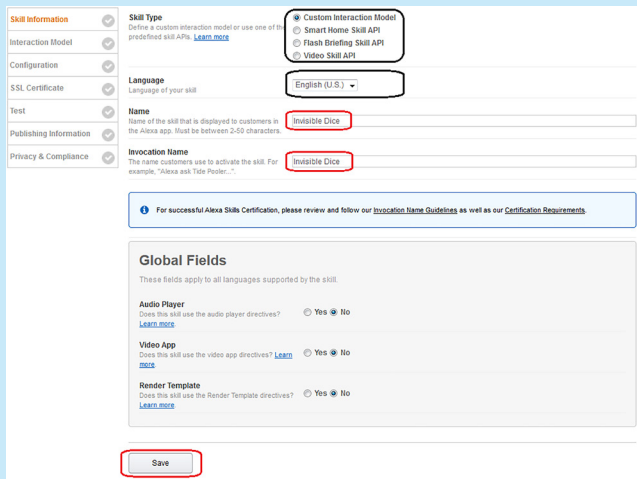
Tabela 1. Szablony do tworzenia umiejętności Skills

custom skills	home skills	flash briefing skills	video skills
opis funkcji szablonów			
obsługa dowolnego rodzaju polecenia np.: odpowiedź na zadane pytanie, integracja ze sklepami i serwisami internetowymi, gry interaktywne itd.	sterowanie urządzeniami inteligentnego domu np.: oświetlenie, termostat, itd.	obsługa serwisów informacyjnych	obsługa odtwarzania video np.: umożliwia konfigurację usług telewizyjnych
użytkownik może definiować własne polecenia oraz oprogramować kod ich obsługi	użytkownik może oprogramować polecenia zdefiniowane w szablonie		

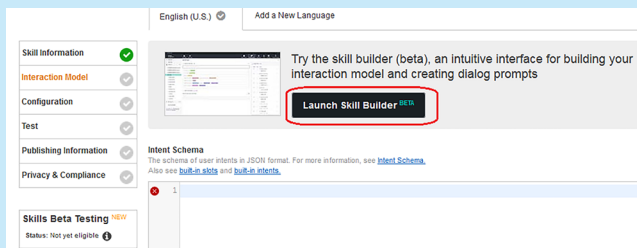
się do portalu (**rysunek 1**). Jeśli nie mamy konta w portalu developera, to należy się zarejestrować.

Po zalogowaniu się z dostępnych opcji wybieramy zakładkę Alexa (**rysunek 2**). Następnie mamy do wyboru dwa pluginy: *Alexa Skills Kit* oraz *Alexa Voice Service*. Ponieważ będziemy tworzyć umiejętność, to wybieramy *Alexa Skills Kit* (**rysunek 3**). Uruchamiane jest okno z informacjami o umiejętnościach przypisanych do konta użytkownika. My jeszcze nie mamy zdefiniowanych umiejętności i dane nie są uzupełnione. Żeby rozpocząć tworzenie umiejętności, z dostępnych opcji wybieramy *Add a New Skill* (**rysunek 4**). W przeglądarce jest uruchamiany kreator tworzenia umiejętności. W pierwszym kroku uzupełniamy informacje o tworzonej umiejętności. Wybieramy rodzaj umiejętności (szablon *Custom Interaction Model* – opcja domyślna) oraz podstawowy język, w którym umiejętność będzie obsługiwana (język angielski amerykański – opcja domyślna). Następnie, wprowadzamy nazwę umiejętności oraz komendę głosową do jej uruchomienia (maksymalnie do 3 słów). Po zakończeniu konfiguracji zatwierdzamy ustawienia (**rysunek 5**).

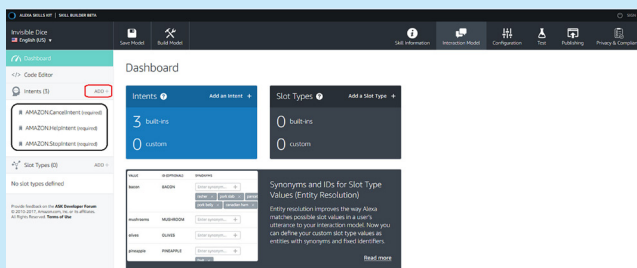
Prezentowane jest podsumowanie wprowadzonych ustawień a następnie jesteśmy przenoszeni do kolejnej zakładki kreatora. W zakładce *Interaction Model* będziemy konfigurować interfejs



**Rysunek 5. Amazon developer. Konfiguracja podstawowych informacji o umiejętności**



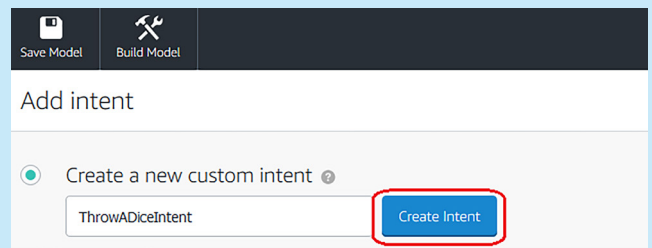
**Rysunek 6. Amazon developer. Interfejs głosowy użytkownika. Uruchomienie kreatora**



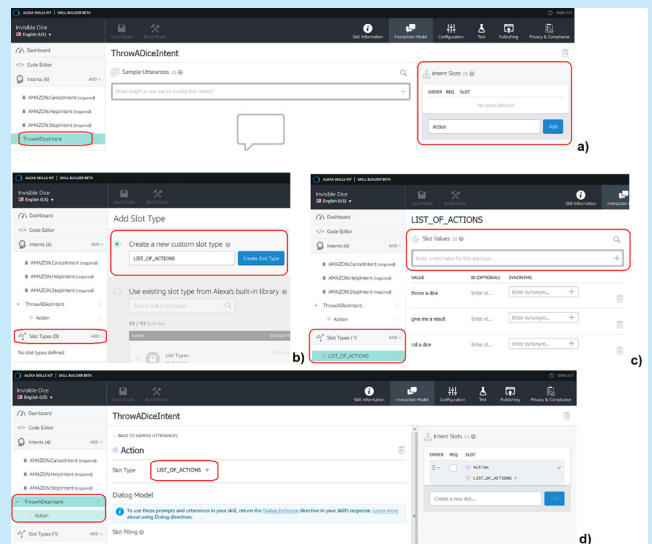
**Rysunek 7. Amazon developer. Interfejs głosowy użytkownika. Okno kreatora**

głosowy użytkownika. Budowa interfejsu sprowadza się do zdefiniowania: *Intents* (intencji tj. czynności, którą chcemy oprogramować), opcjonalnie *Slot Types* (typy danych używane do tworzenia wypowiedzi) oraz *Sample Utterances* (wypowiedzi obsługiwane przez asystentkę Alexa). Żeby ułatwić konfigurację interfejsu użytkownika firma Amazon przygotowała narzędzie *Skill Builder*. Oprogramowanie uruchamiamy bezpośrednio z zakładki *Interaction Model* (**rysunek 6**).

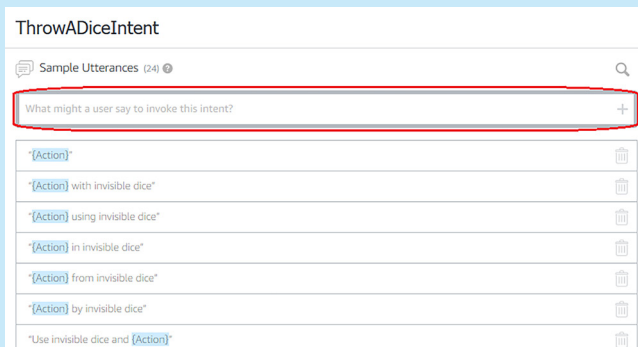
Po uruchomieniu mamy zdefiniowane intencje: *Cancel*, *Help*, *Stop*. Korzystając z funkcji *Add* w zakładce *Intents* możemy dodawać własne definicje (**rysunek 7**). Dodamy intencję, w której programujemy akcję rzutu kostką. W tym celu, z zakładki *Intents* wybieramy przycisk *Add*. Następnie wprowadzamy nazwę tworzonej intencji *ThrowADiceIntent* i zatwierdzamy wybór przyciskiem *Create Intent* (**rysunek 8**). W kolejnym kroku definiujemy komendy głosowe, które będą uruchamiały intencję. Żeby ułatwić budowanie wypowiedzi utworzymy nowy *Slot*. W naszym wypadku będzie to slot o nazwie *Action*, który użyjemy do zdefiniowania wypowiedzi określających akcję rzutu kostką. W tym celu w oknie kreatora zaznaczamy utworzoną intencję *ThrowADiceIntent* oraz korzystając z dostępnych funkcji kreatora tworzymy slot o nazwie *Action* (**rysunek 9a**). Następnie definiujemy nowy typ danych *LIST\_OF\_ACTIONS*, z którego będzie korzystał slot *Action* (**rysunek 9b**). Po utworzonego typu danych wprowadzamy wypowiedzi opisujące akcję rzutu kostką. W naszym przypadku są to frazy: *throw a dice*, *give me a result*, *roll a dice* (**rysunek 9c**). Gdy mamy już utworzony Slot oraz typ danych z wypowiedziami, to łączymy oba elementy ze sobą. Żeby to zrobić, w opcjach konfiguracyjnych slotu *Action* zaznaczamy użycie typu danych *LIST\_OF\_ACTIONS* (**rysunek 9d**).



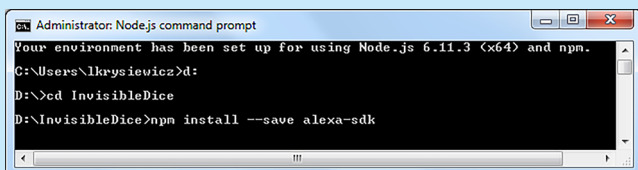
**Rysunek 8. Amazon developer. Interfejs głosowy użytkownika. Tworzymy intencje**



**Rysunek 9. Amazon developer. Interfejs głosowy użytkownika. Tworzymy slot. a) nowy slot Action b) nowy typ danych LIST\_OF\_ACTIONS c) definicja wartości dla typu danych LIST\_OF\_ACTIONS d) slot Action wybór typu danych LIST\_OF\_ACTIONS**

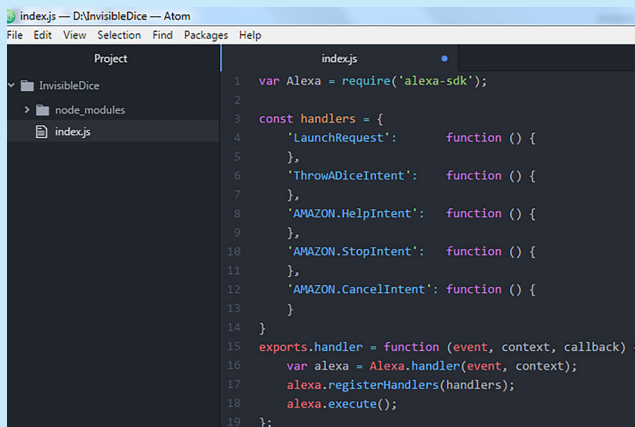


**Rysunek 10. Amazon developer. Interfejs głosowy użytkownika. Definicja wypowiedzi**



**Rysunek 11. Node.js Pobranie pakietu npm frameworka Alexa Skills Kit SDK**

Slot został utworzony i skonfigurowany. W kreatorze uruchamiany intencję *ThrowADiceIntent* (rys. 9a) oraz wprowadzamy przykłady wypowiedzi (rysunek 10). Definiujemy jak największą liczbę wypowiedzi. Korzystamy przy tym ze zdefiniowanego slotu (w materiałach dodatkowych dołączonych do projektu jest dostępny plik *IntentSchema.json* opisujący intencje wraz z zdefiniowanymi

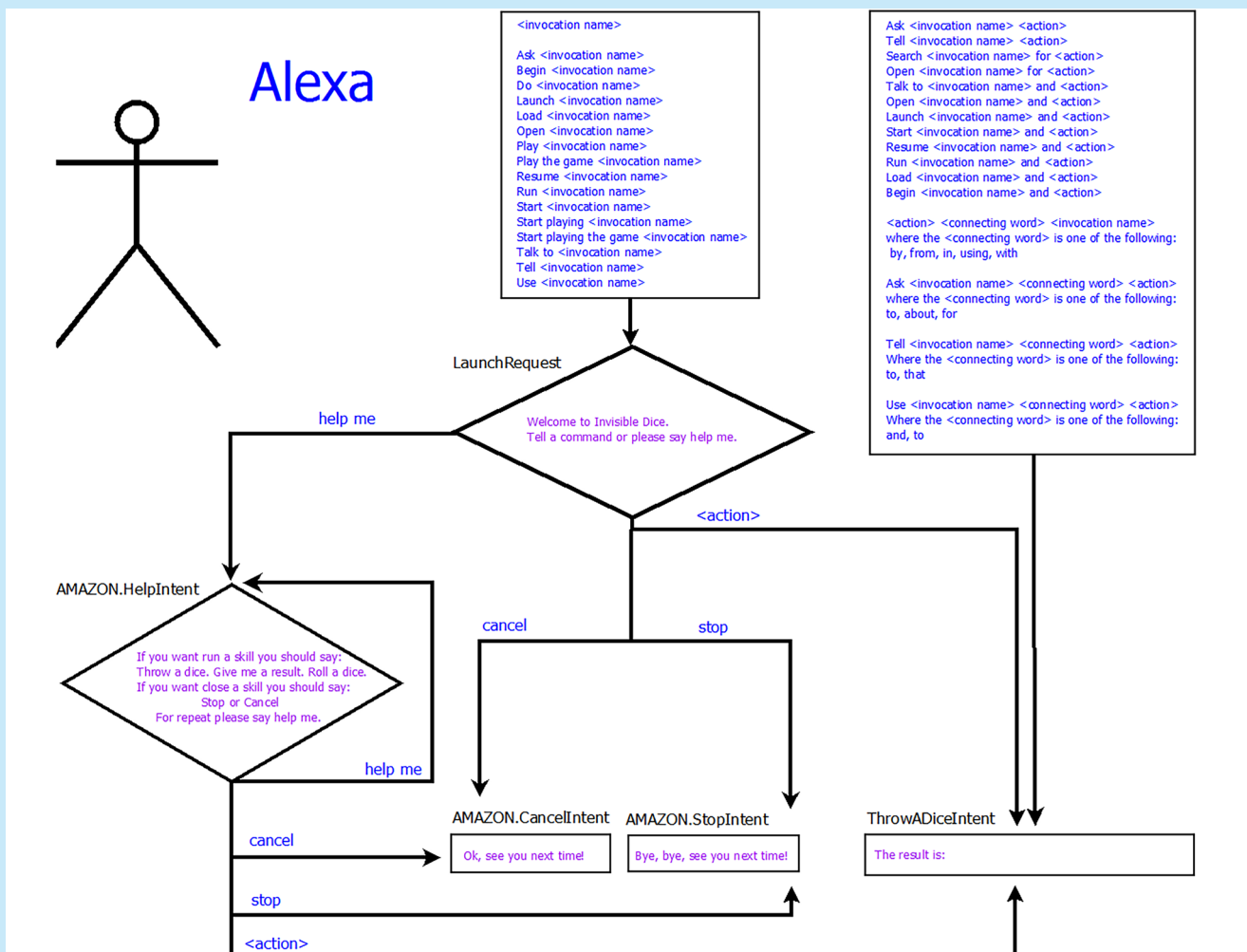


**Rysunek 12. Edytor Atom. Szablon programu do obsługi umiejętności**

wypowiedziami). Na zakończenie, używając przycisków *Save Model* oraz *Build Model* zapisujemy oraz kompilujemy utworzony interfejs głosowy. Teraz, jeśli wypowiemy do asystentki Alexa jedną ze zdefiniowanych wypowiedzi, automatycznie zostanie uruchomiony kod programu odpowiedzialny za obsługę przypisaną do wypowiedzi intencji. W dalszej części artykułu pokażemy, w jaki sposób napisać oprogramowanie do obsługi intencji.

### Alexa Skill Code

Żeby oprogramować obsługę umiejętności dla asystentki Alexa mamy do wyboru dwie opcje. Pierwsza, to napisanie własnego web serwisu (requests over HTTPS) i umieszczenie oprogramowania



**Rysunek 13. Invisible Dice. Diagram działania umiejętności**

na serwerze (własnym bądź hostingodawcy). Druga, to skorzystanie z funkcji Lambda udostępnianej na serwerach firmy Amazon. W naszym przypadku skorzystamy z funkcji Lambda. Kod programu dla funkcji Lambda napiszemy w JavaScript (pozostałe opcje to: Java, Python i C#). Dodatkowo skorzystamy z przygotowanego przez firmę Amazon frameworka *Alexa Skills Kit SDK*.

Najpierw przygotowujemy środowisko uruchomieniowe dla JavaScript: *Node.js*. Uruchamiamy stronę [nodejs.org](https://nodejs.org) dedykowaną dla projektu *Node.js* pobieramy, a następnie instalujemy oprogramowanie. Po zainstalowaniu środowiska uruchamiamy linię komend *Node.js command prompt* oraz przechodzimy do katalogu projektu (u nas to folder *InvisibleDice* utworzony na partycji D). Następnie, korzystając z wbudowanego w środowisko *Node.js* menedżera pakietów npm pobieramy framework *Alexa Skills Kit SDK*. Zestaw komend użytych do pobrania frameworka prezentuje **rysunek 11**.

Po zakończeniu pobierania w katalogu projektu zostanie utworzony folder *node\_modules* zawierający oprogramowanie frameworka *Alexa Skills Kit SDK for Node.js*. Dokumentacja techniczna frameworka jest dostępna na stronie <https://goo.gl/YmCW35>. Po zapoznaniu się z dokumentacją możemy przejść do tworzenia kodu programu dla umiejętności. Najpierw jednak będziemy musieli zainstalować edytor tekstu dla języka programowania *JavaScript*. W naszym przypadku wybraliśmy darmowy edytor Atom.

Edytor Atom pobieramy ze strony [atom.io](https://atom.io). Instalujemy a następnie uruchamiamy oprogramowanie. Po uruchomieniu edytora otwieramy folder *InvisibleDice* (*File* → *Open Folder*), w którym znajdują się pliki frameworka. Następnie tworzymy nowy plik (*File* → *New File*) i zapisujemy plik w folderze projektu (*File* → *Save*) nadając mu nazwę *index.js*. W pliku *index.js* umieszczamy kod programu do obsługi umiejętności. Na początku programu inicjujemy użycie frameworka *Alexa Skills Kit SDK for Node.js*. Następnie, piszemy kod programu do obsługi intencji. Na zakończenie rejestrujemy użycie intencji oraz uruchamiamy frameworka. Wygląd szablonu programu do obsługi umiejętności asystentki Alexa pokazano na **rysunku 12**.

W naszym przypadku do kodu źródłowego szablonu dodamy obsługę intencji: *Launch*, *Help*, *Stop*, *Cancel* oraz *ThrowADice*. Intencja *Launch* jest uruchamiana, gdy użytkownik wypowie jedną ze zdefiniowanych komend startowych (lista komend startowych została zamieszczona w komentarzu do kodu programu). Po uruchomieniu intencji *Launch* jest odtwarzany komunikat

```
Listing 1. Kod programu do obsługi umiejętności: Invisible Dice
var Alexa = require('alexa-sdk');

// Used when a user asks for help.
const helpMessage = "If you want run a skill you should say:
  Throw a dice. Give me a result. Roll a dice. " +
  "If you want close a skill you should say: Stop or Cancel." +
  "For repeat please say help me.";

// Used when a user opens skill.
const welcomeMessage = "Welcome to Invisible Dice. " +
  "Tell a command or please say help me.";

// Used when a user says cancel.
const cancelMessage = "Ok, see you next time!";

// Used when a user says stop.
const stopMessage = "Bye, bye, see you next time!";

// Used to randomise numbers
const t_numbers = [
  "one",
  "two",
  "three",
  "four",
  "five",
  "six"];

const handlers = {
  'LaunchRequest': function () {
    this.emit(':ask', welcomeMessage, welcomeMessage);
  },
  'ThrowADiceIntent': function () {
    var sNumber = t_numbers[Math.floor(Math.random() * t_numbers.length)];
    var sOutput = "The result is:" + " " + sNumber;

    this.emit(':tell', sOutput, sOutput);
  },
  'AMAZON.HelpIntent': function () {
    this.emit(':ask', helpMessage, helpMessage);
  },
  'AMAZON.StopIntent': function () {
    this.emit(':tell', stopMessage, stopMessage);
  },
  'AMAZON.CancelIntent': function () {
    this.emit(':tell', cancelMessage, cancelMessage);
  }
}

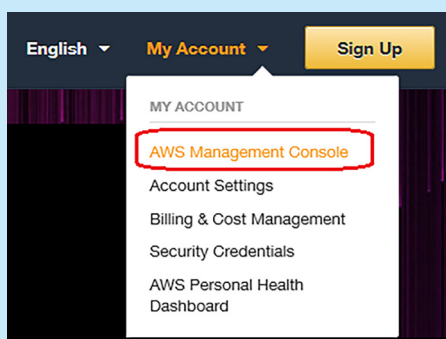
exports.handler = function (event, context, callback) {
  var alexa = Alexa.handler(event, context);
  alexa.registerHandlers(handlers);
  alexa.execute();
};
```

powitalny, a użytkownik jest pytany o dalsze instrukcje. Intencję *ThrowADice* (obsługa rzutu kostką) możemy uruchomić z intencji *Launch* (pytanie o instrukcję wydajemy polecenie rzutu kostką). Intencję możemy również uruchomić „bezpośrednio” wypowiadając jedną ze zdefiniowanych w interfejsie głosowym użytkownika fraz. Diagram pracy umiejętności niewidzialnej kostki do gry prezentuje **rysunek 13**. Kod programu umiejętności umieszczono na **listingu 1**.

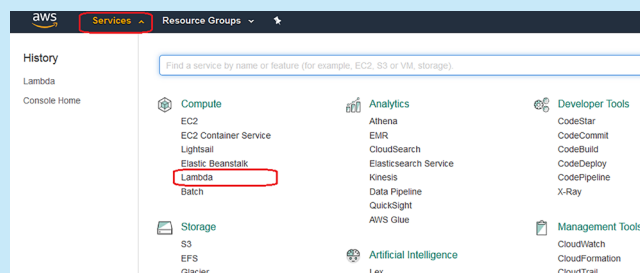
Na zakończenie kod programu (plik *index.js*) oraz folder zawierający framework *Alexa Skills Kit SDK* (folder *node\_modules*) kompresujemy do archiwum w formacie zip. Skompresowany plik użyjemy podczas konfiguracji funkcji Lambda.

## Lambda Function

Firma Amazon udostępnia na platformie AWS (*Amazon Web Services*) serwis funkcja Lambda. Jest to serwis pozwalający na żądanie użytkownika bądź innego serwisu na uruchomienie wykonania



Rysunek 14. Platforma AWS. Logowanie do konta użytkownika



Rysunek 15. Platforma AWS. Wybieramy serwis funkcji Lambda

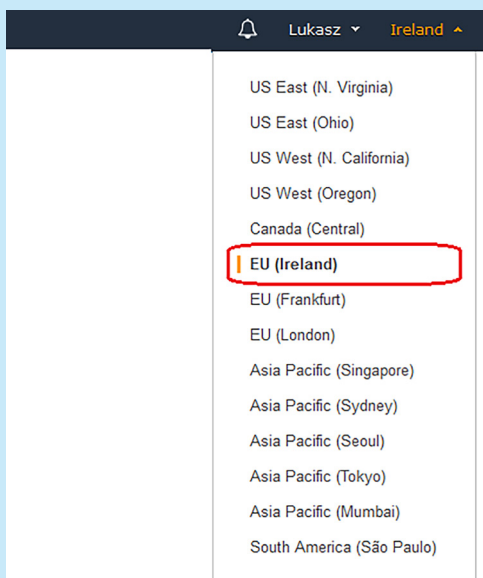
oprogramowania napisanego w jednym z języków: JavaScript, Java, C#, Python.

Żeby skorzystać z funkcji Lambda musimy posiadać konto użytkownika na platformie AWS. Do założenia konta jest potrzebny adres e-mail, telefon kontaktowy oraz karta kredytowa (firma Amazon używa danych z karty kredytowej do autoryzacji użytkowników platformy AWS). My założymy darmowe konto *AWS Free Tier*. Kolejne kroki tworzenia konta są następujące:

1. Uruchamiamy stronę <http://aws.amazon.com>.
2. Wybieramy opcję: *Create A Free Account*.
3. Wprowadzamy dane logowania do konta: nazwę użytkownika i adres e-mail i hasło.
4. Uzupełniamy dane kontaktowe: imię i nazwisko, adres oraz opcjonalnie nazwę firmy.
5. Podajemy informacje o karcie kredytowej (uwaga w celu autoryzacji z karty kredytowej zostanie pobrana opłata w wysokości 1 dolara amerykańskiego).
6. Potwierdzamy swoje „człowieczeństwo”. W formularzu na stronie konfiguratora podajemy numer telefonu i wybieramy opcję zadzwonić. Na stronie jest wyświetlany czterocyfrowy numer PIN. Po chwili na podany numer telefonu dzwoni automat z firmy AMAZON i prosi o wprowadzenie wyświetlonego numeru PIN.
7. Wybieramy poziom wsparcia technicznego. W naszym przypadku wybieramy darmową opcję podstawową *Basic (Free)*.
8. Zatwierdzamy utworzenie konta w portalu AWS.

W tym momencie zostanie utworzone darmowe konto *AWS Free Tier*. W ramach konta będziemy mieli dostęp do wszystkich serwisów platformy AWS. Jednak dostęp do serwisów będzie limitowany (ograniczony czas dostępu, limit na ilość zapytań, ograniczone zasoby, itp.). Po 12 miesiącach od założenia konta niektóre limity zostaną zwiększone. Konto będzie jednak nadal aktywne i serwisy będą dostępne. W wypadku funkcji Lambda w darmowej wersji konta jest ona dostępna z ograniczeniem 1 miliona zapytań na miesiąc (ponad 11 zapytań na sekundę). Taka liczba zapytań w zupełności wystarczy do pracy z asystentką Alexa. Szczegółowy opis darmowej licencji konta AWS jest dostępny pod adresem <http://aws.amazon.com/free>.

Po zarejestrowaniu konta na platformie AWS logujemy się do serwisu. Uruchamiamy stronę <http://aws.amazon.com> i z zakładki *My Account* wybieramy opcję *AWS Management Console* (rysunek 14). Następnie, podajemy nazwę użytkownika i hasło. Po zalogowaniu się uruchamiamy zakładkę *Services*, a następnie odnajdujemy

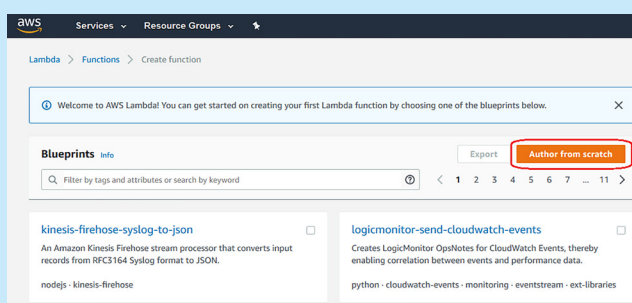


**Rysunek 16. Platforma AWS. Funkcja Lambda. Wybór regionu działania**

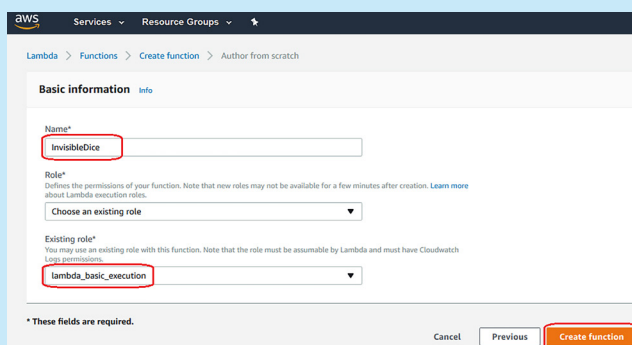
i wybieramy funkcję Lambda (rysunek 15). W kolejnym kroku wybieramy region geograficzny, w którym będą znajdowali się odbiorcy naszej funkcji. Do wyboru mamy dwa obszary: US East (N. Virginia) i EU (Ireland) (tylko w tych centrach danych funkcja Lambda obsługuje *Alexa Skills Kit*). W naszym przypadku wybieramy region przeznaczony dla Europy (rysunek 16). Następnie przechodzimy do tworzenia nowej funkcji Lambda. W oknie głównym serwisu AWS zaznaczamy opcję *Create a function* (rysunek 17). Uruchamiane jest okno, w którym są prezentowane gotowe szablony dla funkcji Lambda. My wybieramy opcję tworzenia funkcji od podstaw *Author from scratch* (rysunek 18). Wprowadzamy nazwę funkcji (*InvisibleDice*),



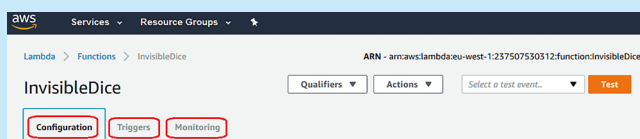
**Rysunek 17. Platforma AWS. Funkcja Lambda. Utwórz nową**



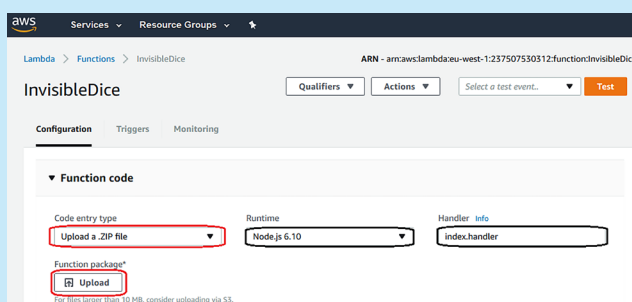
**Rysunek 18. Platforma AWS. Funkcja Lambda. Nowa funkcja. Definicja użytkownika**



**Rysunek 19. Platforma AWS. Funkcja Lambda. Nowa funkcja. Informacje podstawowe**

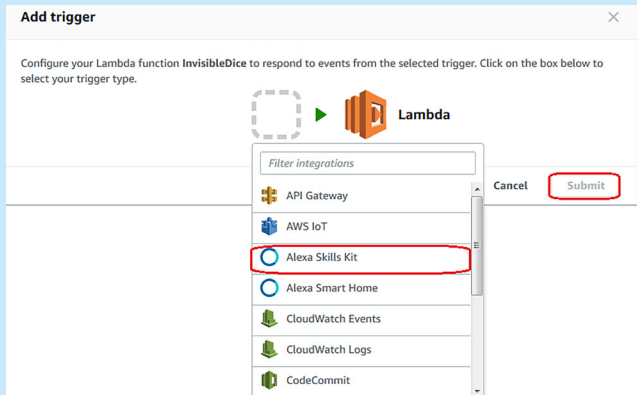


**Rysunek 20. Platforma AWS. Funkcja Lambda. Panel sterowania**

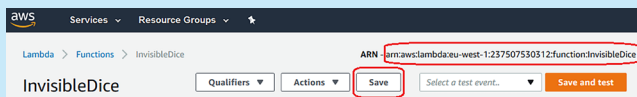


**Rysunek 21. Platforma AWS. Funkcja Lambda. Upload archiwum ZIP z kodem funkcji**

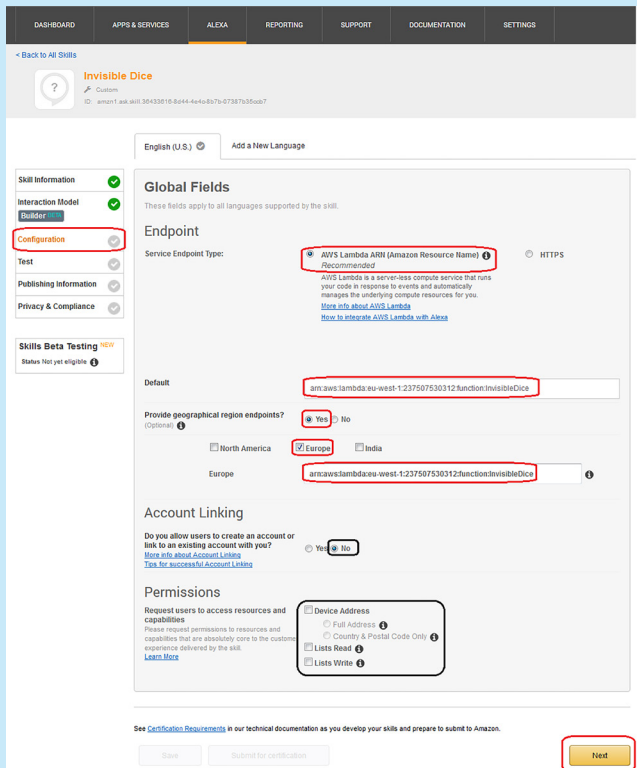
definiujemy uprawnienia (rola `lambda_basic_execution`) oraz zatwierdzamy utworzenie funkcji (rysunek 19). W systemie zostaje utworzona funkcja Lambda. W panelu głównym zarządzania funkcją dostępne są trzy zakładki: konfiguracja, wyzwalacze funkcji oraz monitoring działania funkcji (rysunek 20). Wybieramy zakładkę odpowiadającą za konfigurację, a następnie przechodzimy do pola zarządzania kodem funkcji Lambda. Domyślnie jest uruchomiony edytor kodu programu online. W naszym przypadku będziemy dołączać wcześniej napisany kod programu. Z listy rozwijalnej `Code entry type` wybieramy opcję `Upload a .ZIP file i dołączamy archiwum ZIP zawierające kod programu dla umiejętności InvisibleDice`. (rysunek 21). Po dołączeniu archiwum z kodem programu funkcji Lambda przechodzimy do zakładki `Triggers`. W zakładce zdefiniujemy serwis, który będzie wyzwaliał wykonanie funkcji Lambda. Z dostępnych opcji wybieramy



**Rysunek 22. Platforma AWS. Funkcja Lambda. Konfiguracja wyzwalacza funkcji**



**Rysunek 23. Platforma AWS. Funkcja Lambda. Zakończenie konfiguracji**



**Rysunek 24. Amazon developer. Konfiguracja umiejętności. Połączenie VUI z kodem programu**

serwis `Alexa Skills Kit` a następnie zatwierdzamy wybór (rysunek 22). Na zakończenie zapisujemy ustawienia funkcji Lambda oraz kopiujemy identyfikator utworzonej funkcji (rysunek 23). Dalszą konfigurację umiejętności niewidzialnej kostki do gry będziemy realizować w portalu `developer`, więc możemy wylogować się z platformy `AWS Amazon Web Services`.

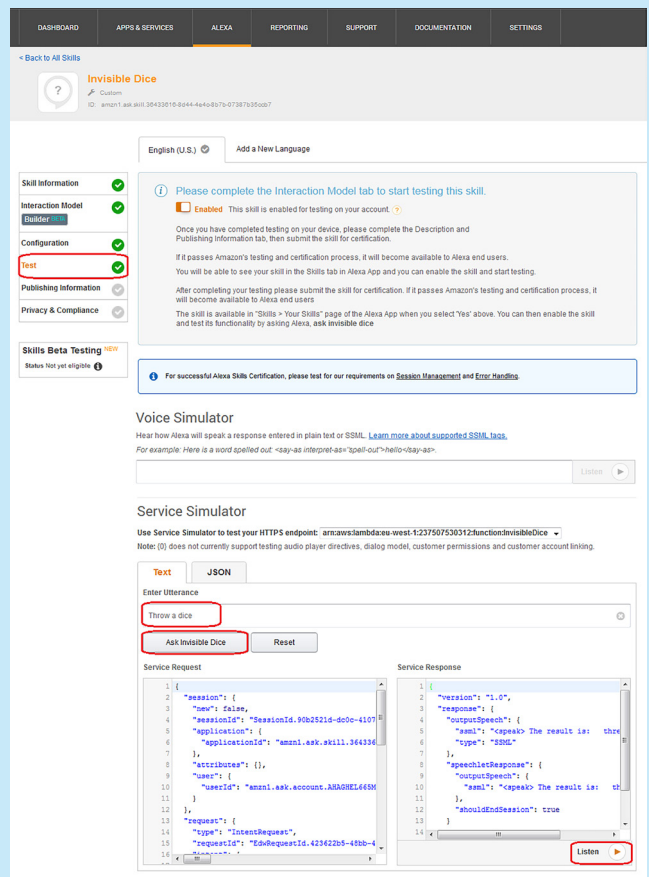
## Connect VUI to Code

Wykonaliśmy interfejs głosowy użytkownika `VUI (Voice User Interface)` oraz kod programu do obsługi umiejętności. Żeby uruchomić oprogramowanie skonfigurowaliśmy funkcję Lambda. Kolejnym krokiem w tworzeniu umiejętności jest połączenie interfejsu głosowego użytkownika z kodem programu do obsługi umiejętności.

W tym celu logujemy się do portalu `developer` <http://developer.amazon.com> oraz powtarzamy czynności zilustrowane na rys. 1, 2 i 3. Następnie z listy dostępnych umiejętności wybieramy `InvisibleDice` oraz zaznaczamy edycję umiejętności. Uruchamiany jest kreator tworzenia umiejętności. Z menu kreatora wybieramy opcję `Configuration`. Następnie, zaznaczamy typ serwisu obsługującego kod programu umiejętności. Z dostępnych opcji: funkcja Lambda oraz własny webserwis zaznaczamy funkcję Lambda. W kolejnym kroku w pole `Default` wprowadzamy identyfikator naszej funkcji. Dodatkowo włączamy opcję `rejonizacji funkcji Lambda`. Zaznaczamy obszar `Europy` oraz ponownie wprowadzamy identyfikator funkcji. Pozostałe opcje pozostawiamy z ustawieniami domyślnymi (rysunek 23). Na zakończenie zatwierdzamy ustawienia (przycisk `Next`).

## Testy

Przechodzimy do kolejnego etapu tworzenia umiejętności dla asystentki głosowej `Alexa`. Mamy już zdefiniowany interfejs głosowy użytkownika. Napisaaliśmy oprogramowanie do obsługi umiejętności. Połączyliśmy oba elementy ze sobą. Teraz przetestujemy działanie umiejętności. Testy wykonamy korzystając z portalu



**Rysunek 25. Amazon developer. Testy działania umiejętności**

developera (można skorzystać z platformy AWS i przeprowadzić testy działania funkcji Lambda). W oknie kreatora umiejętności wybieramy opcję *Test*. Dla każdej z utworzonych intencji wprowadzamy zdefiniowane wypowiedzi i weryfikujemy czy odpowiedź asystentki Alexa jest poprawna (rysunek 25). Po zakończeniu testów przechodzimy do publikacji umiejętności.

## Publikacja

Umiejętność niewidzialnej kostki (Invisible Dice) dla asystentki głosowej Alexa jest już gotowa. Teraz złożymy wniosek o certyfikację i publikację umiejętności. Na początku musimy uzupełnić informacje opisujące naszą umiejętność. W tym celu w oknie kreatora umiejętności wybieramy zakładkę *Publishing Information*. Ustawiamy kategorię umiejętności oraz jeśli jest to wymagane to dodatkowo wybieramy podkategorię. Opisujemy działanie umiejętności podając opis skrócony oraz rozszerzony. Podajemy przykłady wypowiedzi używanych do uruchomienia umiejętności. Dołączamy ikony, które zostaną użyte w aplikacji Amazon Alexa. Na zakończenie zapisujemy ustawienia i przechodzimy dalej (rysunek 26).

W kolejnym kroku przygotowania umiejętności do publikacji definiujemy politykę prywatności oraz zatwierdzamy regulamin firmy

Rysunek 26. Amazon developer. Publikacja umiejętności. Informacje podstawowe

Amazon. W tym celu w oknie kreatora umiejętności wybieramy zakładkę *Privacy & Compliance*. Następnie odpowiadamy na pytania:

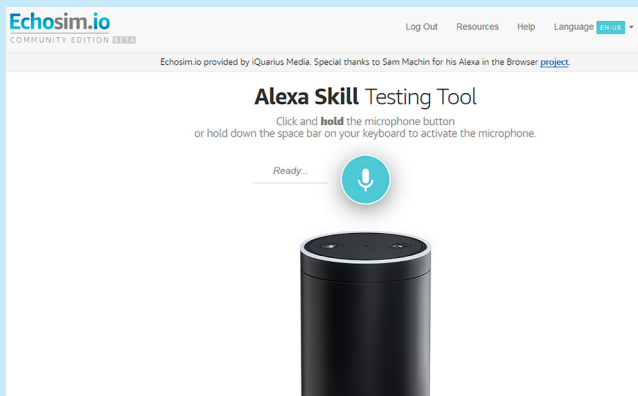
- Czy umiejętność pozwala na robienie zakupów i wydawanie „prawdziwych” pieniędzy?
- Czy umiejętność pozyskuje prywatne dane użytkowników?
- Czy umiejętność jest przeznaczona dla dzieci poniżej 13 roku życia?
- Czy umiejętność zawiera reklamy?

Po wprowadzeniu odpowiedzi zapisujemy ustawienia oraz wysyłamy umiejętność do certyfikacji (rysunek 27).

Rysunek 27. Amazon developer. Publikacja umiejętności. Polityka prywatności i regulaminy

Rysunek 28. Amazon developer. Publikacja umiejętności. Potwierdzenie przestania do certyfikacji

Rysunek 29. Amazon Alexa. Dodanie umiejętności do konta użytkownika



**Rysunek 30. Amazon Alexa. Uruchomienie umiejętności w symulatorze Echosim**

Przesłanie umiejętności do certyfikacji jest potwierdzone komunikatem systemowym (rysunek 28). Dodatkowo otrzymujemy wiadomość e-mail z informacjami opisującymi proces certyfikacji umiejętności. Wniosek o certyfikację umiejętności jest rozpatrywany przez inżynierów z firmy Amazon. Sprawdzana jest funkcjonalność umiejętności, zabezpieczenia, zgodność z polityką prywatności firmy Amazon. Przeprowadzane są testy interfejsu głosowego użytkownika (min.: testowane są komendy służące do uruchomienia umiejętności). Do czasu rozpatrzenia wniosku o certyfikację nie możemy edytować umiejętności (edycja jest możliwa, jeśli wycofamy wniosek). Odpowiedź z firmy Amazon otrzymujemy w czasie nie dłuższym niż 7 dni. W naszym wypadku wiadomość e-mail o pozytywnej weryfikacji otrzymaliśmy w ciągu kilku godzin od wysłania zgłoszenia. Pozytywne rozpatrzenie wniosku o certyfikację skutkuje automatyczną publikacją umiejętności.

## Uruchomienie

Nasza umiejętność została opublikowana i jest dostępna dla wszystkich użytkowników urządzeń z asystentką głosową Alexa! W naszym przypadku (właściciel umiejętności została automatycznie dodana do grupy umiejętności przypisanych do naszego konta. Pozostali użytkownicy, jeśli chcą skorzystać z naszej umiejętności powinni dodać ją do własnego konta w serwisie Alexa. W tym celu należy uruchomić konfigurator internetowy Amazon Alexa (<http://alexa.amazon.com>),

wybrać zakładkę *Skills*, odszukać umiejętność niewidzialnej kostki do gry (invisible dice) i zatwierdzić dodanie umiejętności do konta (rysunek 29). Teraz można zacząć używać umiejętności (patrz rys. 13) i grać w gry planszowe bez użycia kostki do gry. Życzę dobrej zabawy!

Jeśli użytkownik nie ma urządzenia z asystentką głosową Alexa, to może uruchomić umiejętność korzystając z symulatora *Echosim*. W tym celu należy otworzyć stronę internetową *echosim.io* i zalogować się do aplikacji używając danych konta Amazon (rysunek 30). Następnie, należy wcisnąć przycisk mikrofonu i wypowiedzieć polecenie do asystentki Alexa (np.: *Alexa, Open Invisible Dice*). Zachęcamy czytelników do uruchomienia i testów stworzonej w artykule umiejętności.

## Podsumowanie

Wykonaliśmy naszą pierwszą umiejętność dla asystentki głosowej Alexa. W kolejnym artykule rozbudujemy funkcjonalność umiejętności. Pokażemy, w jaki sposób uruchomić obsługę odtwarzania efektów dźwiękowych. Zdemonstrujemy też sposób wyświetlenia plansze z wynikami. Oprogramujemy nowe tryby pracy umiejętności.

**Łukasz Krysiwicz, EP**

REKLAMA

UlubionyKiosk.pl oferuje papierowe i elektroniczne wydania czasopism z najważniejszych segmentów rynku.

UlubionyKiosk.pl