

**Przetwarzanie mowy**

**Projektowanie interfejsów multidedialnych**

**Prowadzący:**

**dr Marek Jaszuk**

**Wykonawca:**

**Michał Dałomis**

**Rzeszów 2020 r.**

Spis treści

[Technologie 3](#_Toc31973912)

[JavaScript 3](#_Toc31973913)

[HTML 5](#_Toc31973914)

[Część praktyczna 7](#_Toc31973915)

[Założenia projektowe 7](#_Toc31973916)

[Web Speech API 7](#_Toc31973917)

[Tworzenie 7](#_Toc31973918)

[Ostateczny widok strony internetowej 10](#_Toc31973919)

# Technologie

### JavaScript

JavaScript jest dynamicznym językiem programowania .Najczęściej wykorzsytywany jest do tworzenia stron internetowych, których implementacje umożliwiają skryptowi po stronie klienta interakcję z użytkownikiem i pozwala na dynamiczne tworzenie stron.Warto dodać , że jest to interpretowany język programowania z funkcjami obiektowymi.

JavaScript został po raz pierwszy zaprezentowany jako LiveScript**,** ale Netscape zmienił nazwę na JavaScript. Rdzeń języka został osadzony w Netscape, Internet Explorer i innych przeglądarkach internetowych.

Specyfikacja [ECMA-262](http://www.ecma-international.org/publications/index.html) definiuje standardową wersję podstawowego języka JavaScript.

Podstawowe cechy języka JavaScript:

* JavaScript jest lekkim, interpretowanym językiem programowania;
* Zaprojektowany do tworzenia aplikacji sieciowych;
* Uzupełniający i zintegrowany z Javą;
* Uzupełniający i zintegrowany z HTML;
* Otwarty i wieloplatformowy.

**JavaScript client-side**

JavaScript client-side jest najczęsciej spotykaną formą tego języka. Skrypt powinien być zawarty w dokumencie HTML lub w odpowiednim do którego powinien się odwoływać, aby kod był widziany przez przeglądarkę.

Oznacza to, że strona internetowa nie musi być statycznym kodem HTML, ale może obejmować programy, które wchodzą w interakcje z użytkownikiem, kontrolują przeglądarkę i dynamicznie tworzą treść HTML.

Mechanizm JavaScript po stronie klienta zapewnia wiele korzyści w stosunku do tradycyjnych skryptów po stronie serwera CGI.Na przykład można użyć JavaScript, aby sprawdzić czy użytkownik wpisał prawidłowy adres e-mail w polu formularza.

Kod JavaScript jest wykonywany, gdy użytkownik prześle formularz i tylko wtedy, gdy wszystkie wpisy są prawidłowe, zostaną one przesłane do serwera WWW.

Skrypt JavaScript może być wykorzystywany do wychwytywania zdarzeń inicjowanych przez użytkownika takich jak kliknięcia przycisków, nawigacja linków i inne działania, które użytkownik inicjuje jawnie lub pośrednio.

**Zalety JavaScript**

**Mniej interakcji z serwerem** - Można zweryfikować dane wejściowe użytkownika przed wysłaniem strony na serwer. Oszczędza to ruch na serwerze, co oznacza mniejsze obciążenie serwera.

**Natychmiastowa informacja zwrotna dla odwiedzających** - nie muszą czekać na ponowne załadowanie strony, aby sprawdzić, czy zapomnieli coś wpisać.

**Zwiększona interaktywność**- można tworzyć interfejsy, które reagują, gdy użytkownik najedzie na nich myszką lub aktywuje je za pomocą klawiatury.

**Bogatsze interfejsy**- można użyć JavaScript, aby uwzględnić takie akcje jak przeciąganie i upuszczanie komponentów czy suwaków w celu zapewnienienia użytkownikom bogatego interfejsu witryny.

**Ograniczenia JavaScript**

JavaScript nie może być traktowany jako pełnoprawny język programowania. Brakuje

mu następujących, ważnych funkcji :

* JavaScript po stronie klienta nie pozwala na odczyt lub zapis plików. Zostało to zachowane ze względów bezpieczeństwa.
* JavaScript nie może być używany w aplikacjach sieciowych, ponieważ taka pomoc nie jest dostępna.
* JavaScript nie ma możliwości obsługi wielowątkowości lub obsługi wieloprocesorowej.

JavaScript jest lekkim, interpretowanym językiem programowania, który pozwala budować interaktywność na statycznych stronach HTML.

**Narzędzia programistyczne JavaScript**

Jedną z głównych zalet JavaScript jest to, że nie wymaga on drogich narzędzi programistycznych.Możliwe jest zacząć od prostego edytora tekstu, takiego jak notatnik. Ponieważ jest to język interpretowany w kontekście przeglądarki internetowej, nie jest konieczny zakup kompilatora.

Aby ułatwić korzystanie z tego produktu stworzono bardzo przydatne narzędzia do edycji JavaScript. Niektóre z nich są wymienione poniżej:

* **Microsoft FrontPage**- Microsoft opracował popularny edytor HTML o nazwie FrontPage. FrontPage zapewnia twórcom stron internetowych szereg narzędzi JavaScript, które pomagają w tworzeniu interaktywnych stron internetowych;
* **Macromedia Dreamweaver MX-**  Macromedia Dreamweaver MX to bardzo popularny edytor HTML i JavaScript wśród zaawansowanych programistów. Zapewnia kilka poręcznych, wstępnie skompilowanych komponentów JavaScript, dobrze integruje się z bazami danych i jest zgodny z nowymi standardami takimi jak XHTML i XML.
* **Macromedia HomeSite 5**- HomeSite 5 to popularny edytor HTML i JavaScript firmy Macromedia, którego można używać do skutecznego zarządzania osobistymi stronami internetowymi.

Aby dołączyć JavaScript do strony robi się to w podobnym stylu co CSS tylko dodając <script>

<script>

// Kod JavaScript

</script>

Aby umieścić kod JavaScript w osobnym pliku można to zrobić w następujący sposób:

**<script src="script.js"></script>**

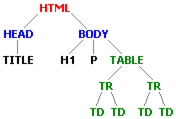
### HTML

Hipertekstowy język znaczników, wykorzystywany do tworzenia dokumentów hipertekstowych.

HTML pozwala opisać strukturę informacji zawartych wewnątrz strony internetowej, nadając odpowiednie znaczenie semantyczne poszczególnym fragmentom tekstu – formując hiperłącza, akapity, nagłówki, listy oraz osadza w tekście dokumentu obiekty plikowe np. multimedia , elementy baz danych.

HTML umożliwia określenie wyglądu dokumentu w przeglądarce internetowej. Do szczegółowego opisu formatowania akapitów, nagłówków, użytych czcionek i kolorów zalecane jest wykorzystywanie kaskadowych arkuszy stylów.

Język HTML składa się z kilku kluczowych komponentów:

* znaczników (i ich atrybutów),
* typów danych,
* referencji znakowych,
* odwołań w postaci encji,
* deklaracji typu dokumentu.

Struktura HTML z ukazaniem znaczników

**Budowa HTML**

**Przykładowe zrzuty ekranów kodu źródłowego HTML:**

1<html>

 2       <head>

 3               <title>...</title>

 4       </head>

 5       <body>

 6               <h1>...</h1>

 7               <p>...</p>

 8               <table>

 9                       <tr>

10                               <td>...</td>

11                               <td>...</td>

12                       </tr>

13                       <tr>

14                               <td>...</td>

15                               <td>...</td>

16                       </tr>

17               </table>

18       </body>

19</html>

**Prosty dokument**

1**<html>**

 2  **<head>**

 3  <title> ... page title ...</title>

 4  **</head>**

 5  **<body>**

 6  ... visible page content ...

 7  **</body>**

 8**</html>**

**Przykładowe atrybuty tekstu HTML**

|  |  |  |
| --- | --- | --- |
| background | [**BODY**](http://www.kurshtml.edu.pl/html/body,html.html) | **Obrazek w tle dokumentu** |
| bgcolor | [**BODY**](http://www.kurshtml.edu.pl/html/body,html.html) | **Kolor tła dokumentu** |
| bgcolor | [**TABLE**](http://www.kurshtml.edu.pl/html/table,html.html) | **Kolor tła komórek tabeli** |
| bgcolor | [TD](http://www.kurshtml.edu.pl/html/td,html.html)**,** [TH](http://www.kurshtml.edu.pl/html/th,html.html) | **Kolor tła komórek tabeli** |
| bgcolor | [**TR**](http://www.kurshtml.edu.pl/html/tr,html.html) | **Kolor tła wierszy tabeli** |

# Część praktyczna

### Założenia projektowe

Przygotowanie aplikacji webowej do rozpoznawania mowy. Aplikacja powinna dawać możliwość przetwarzania mowy na tekst na ekranie komputera, umożliwiając zaznaczanie pól czy wprowadzenie nawet jednej linijki tekstu .Brak ograniczeń w którym języku aplikacja powinna zostać napisana.

### Web Speech API

Nowy interfejs oparty na javascripcie, który umożliwia deweloperom aplikacji webowych tworzyć mechanizm rozpoznawania mowy we własnych projektach. Web Speech API pozwala nie tylko wyszukiwać za pomocą głosu, ale oferuje również wsparcie dla dyktowania, a nawet kontrolowania postaci w grach uruchamianych w przeglądarce.

Aplikacja stworzona w projekcie używa interfejsu Web Speech API do konwersji wypowiadanych słów na tekst. Istnieje prosty JavaScript API, który pozwala zintegrować rozpoznawanie mowy na dowolnej stronie internetowej.

API WebSpeech posiada dwie funkcje syntezy mowy : tekstu na mowę i rozpoznawania mowy.

### Tworzenie

Do ukończenia projektu niezbędne będzie użycie :

* edytora tekstu;
* przeglądarki internetowej.

Przechodząc do projektowania należy utworzyć strukturę strony HTML.

Utworzony kod wygląda następująco :



<button id='komenda'>Wydaj komendę</button>- służy do utworzenia pola "Wydaj komendę"

<input id='chkBox' type="checkbox"> TEST

<input id='chkBox2' type="checkbox"> TEST2

Służą odpowiednio do utworzenia przycisków "TEST", "TEST2".



Na powyższym zrzucie obrazu zostaje ukazane kodowanie, dzięki któremu w przeglądarce otwierają się pola: "Test", "Test2" oraz przygotowane do użycia "Wydaj komendę"

Następnym krokiem będzie utworzenie struktury JavaScriptAPI

Każdy string przekazy do API musi być kodowany w UTF-8.



Do rozpoznawania mowy zostały użyte odpowiednio:

var SpeechRecognition = SpeechRecognition || webkitSpeechRecognition;

var SpeechGrammarList = SpeechGrammarList || webkitSpeechGrammarList;

Aby rozpoznać język w jakim mówi użytkownik użyto :

recognition.grammars = speechRecognitionList;

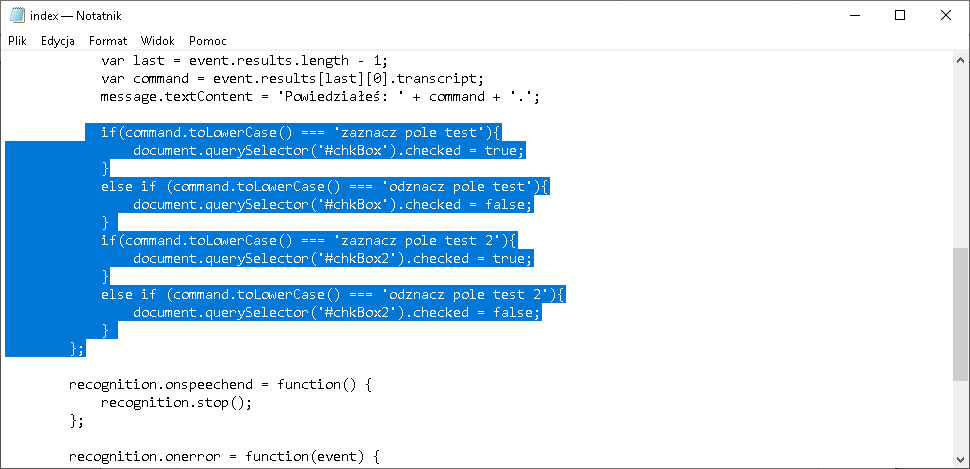
recognition.lang = 'pl-PL';

onresult - interfejs SpeechRecognition . Reprezentuje moduł obsługi zdarzeń, który zostanie uruchomiony, gdy usługa rozpoznawania mowy zwróci wynik.

recognition.onresult = function(event) {

Elementy typu checkbox są domyślnie renderowane jako pola, które są zaznaczone po aktywacji, tak jak w oficjalnym formularzu urzędowym przy zaznaczaniu np.długopisem.

W tym przypadku służa do zaznaczania pól : "Test" i "Test2"



onspeechend – interfejs SpeechRecognition - reprezentuje moduł obsługi zdarzeń, który zostanie uruchomiony, gdy mowa rozpoznawana przez system rozpoznawania mowy przestanie być wykrywana.

recognition.onspeechend = function() {

recognition.stop();

onerror - interfejs SpeechRecognition - reprezentuje moduł obsługi zdarzeń, który zostanie uruchomiony w przypadku wystąpienia błędu rozpoznawania mowy

recognition.onerror = function(event) {

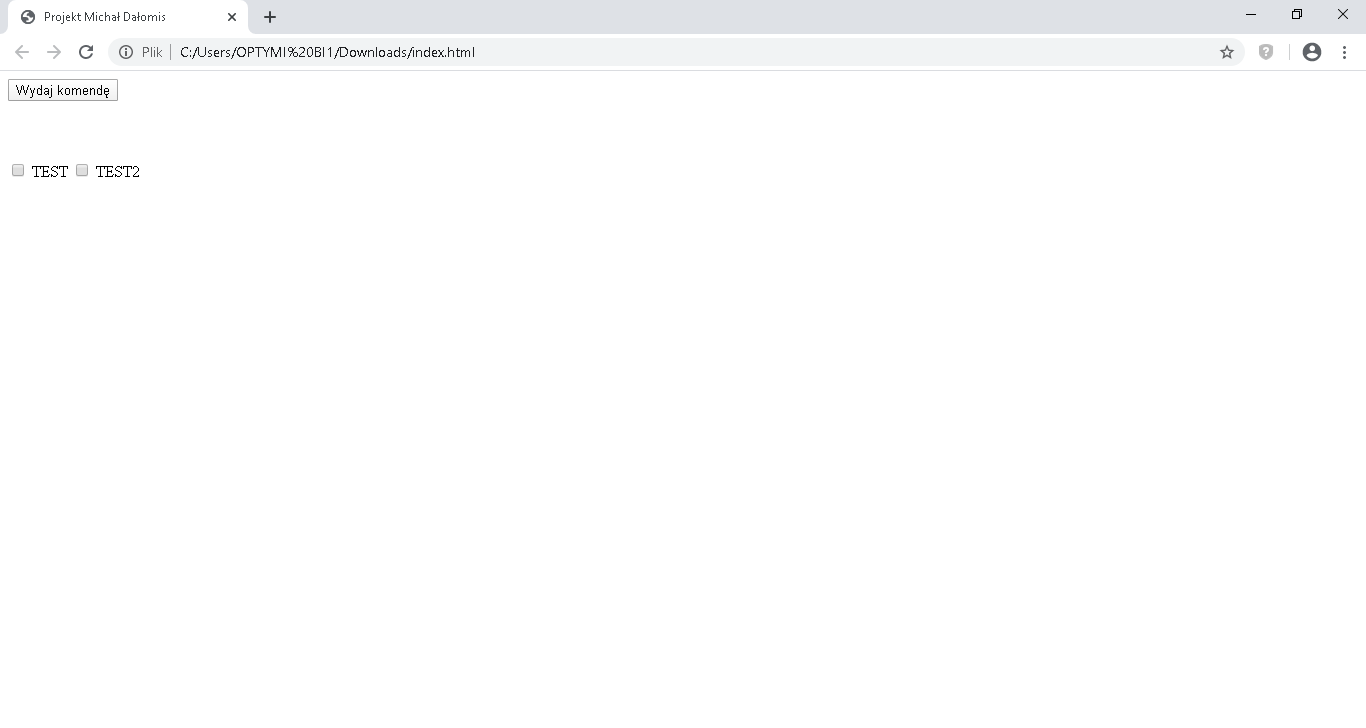
message.textContent = 'Błąd przy rozpoznawaniu: ' + event.error;

Funkcja document.querySelector zwraca pierwszy element wewnątrz dokumentu, który pasuje do podanego selektora lub grupy selektorów.

document.querySelector('#komenda').addEventListener('click', function(){

recognition.start();

### Ostateczny widok strony internetowej

****