



experiências  
interativas com  
Web Speech API



/luisleao

# experiências interativas com Web Speech API



/luisleao







garoa.net.br

+GDG São Paulo

revista iMasters



/luisleao

# reconhecimento de fala

como era antes?

e agora?

demonstrações



/luisleao

como era antes?

2011



# 2011: atributo html

<input x-webkit-speech>

basta adicionar "**x-webkit-speech**" em  
qualquer tag "input" de texto que o ícone do  
microfone aparece

# 2011: atributo html

<input x-webkit-speech>

basta adicionar "**x-webkit-speech**" em  
qualquer tag "input" de texto que o ícone do  
microfone aparece

# 2011: atributo html

<input x-webkit-speech>

basta adicionar "**x-webkit-speech**" em  
qualquer tag "input" de texto que o ícone do  
microfone aparece



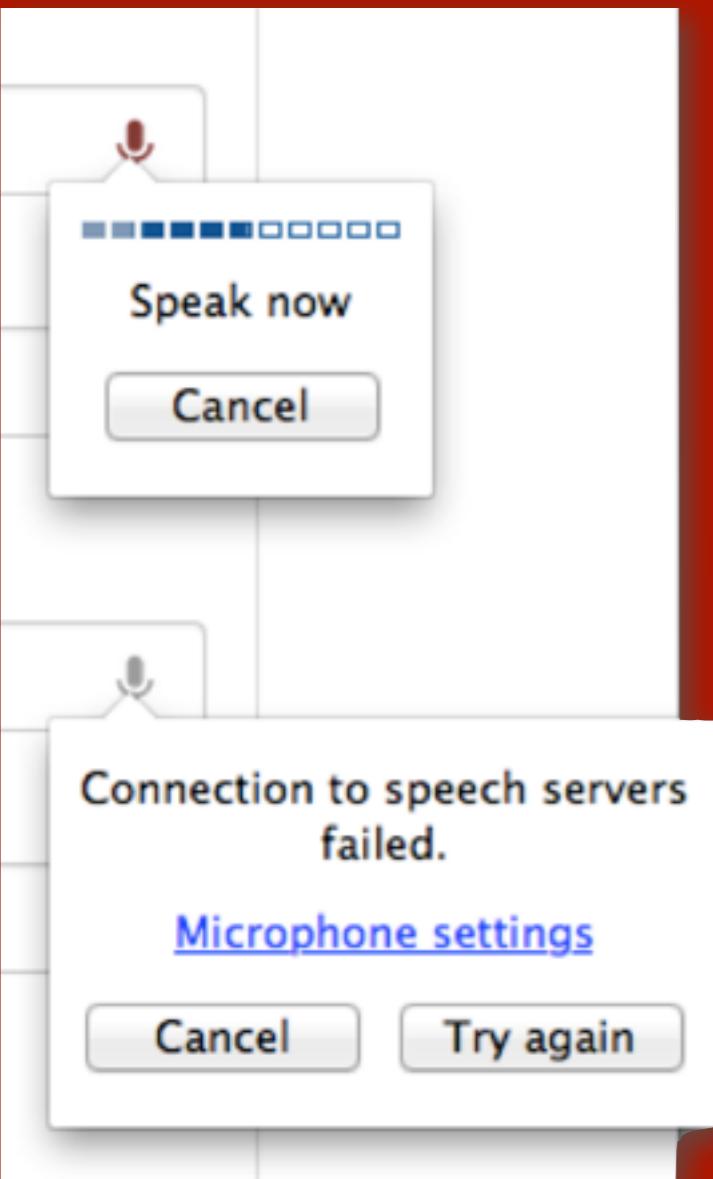
# funciona, mas...

- só em campos <input>
- não pode usar diretamente numa <textarea> ou em outras tags que possuem formatação: <div>, etc.



/luisleao

# interface fixa :'(



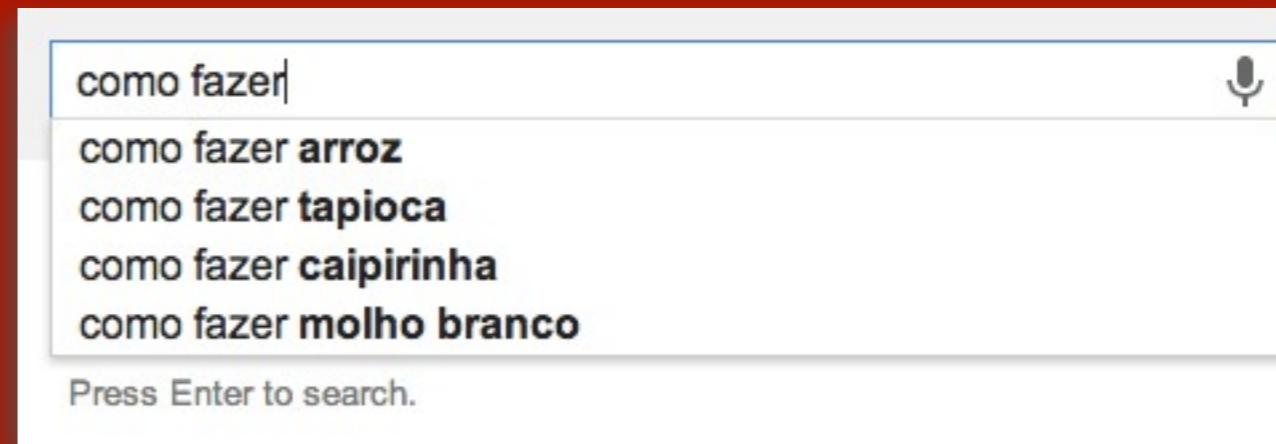
- é o navegador que implementa a interface de reconhecimento:
- tamanho do ícone, forma, sombra e localização são fixas na caixa de texto
- não dá para mudar o balão de fala e as suas funcionalidades



/luisleao

# flexibilidade, pra quê?

- Apenas insere o texto reconhecido no campo
- Qualquer outra coisa precisa ser criada em JavaScript:
  - Validar a entrada de texto
  - Fazer o post do formulário
  - Processar/exibir as melhores alternativas



/luisleao

# acabou? não!

- não dá para iniciar/parar o reconhecimento via script
- tratamento limitado de eventos
- sem streaming dos resultados



/luisleao



/luisleao

2011 + 2



BWS

24

Santa Cruz Beach Boardwalk



is a white sandy S



agora!

Web Speech API



/luisleao

# Web Speech API

- iniciar/parar reconhecimento via programação
- reconhecimento único (uma frase) ou contínuo (pode durar vários minutos)
- transmissão com resultados provisórios (exibe os resultados e adapta enquanto você fala)



/luisleao

# Web Speech API

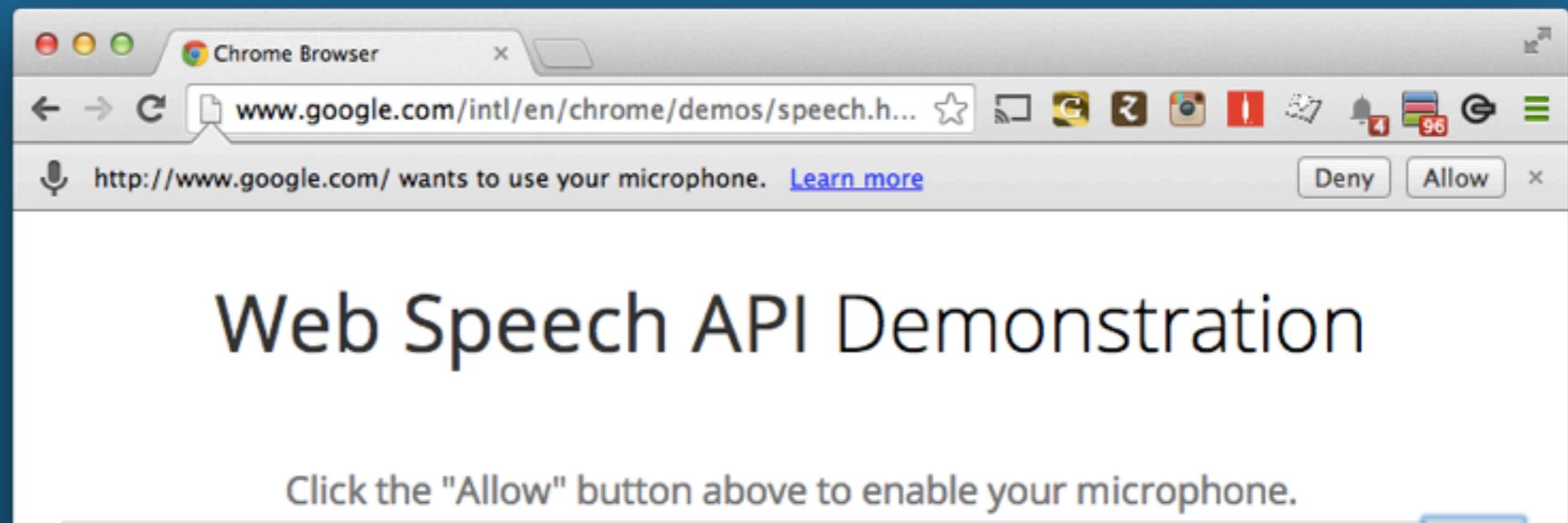
- o reconhecimento pode ser processado on-the-fly e exibido em qualquer lugar
- tratamento de eventos de áudio, som, fala e erro
- seleção de idioma RFC/BCP47 (61 na demo)
- retorno das N-melhores alternativas (parâmetro)



/luisleao

# Web Speech API

- interface flexível
- somente a parte de permissão é implementada pelo navegador



/luisleao

# show me the code!

```
// verificar suporte Web Speech API pelo navegador.  
var SpeechRecognition = window.SpeechRecognition || window.webkitSpeechRecognition;  
if (!SpeechRecognition) {  
    suggestUserUpgradesBrowser();  
} else {  
    // Criar objeto de reconhecimento e definir alguns atributos.  
    var recognition = new SpeechRecognition();  
    recognition.continuous = true;  
    recognition.interimResults = true;  
    recognition.lang = "pt-BR";  
    // Definir funções de tratamento dos eventos  
    recognition.onstart = function() { ... }  
    recognition.onresult = function(event) { ... }  
    recognition.onerror = function(event) { ... }  
    recognition.onend = function() { ... }  
    ...
```



/luisleao

# #naotafacio



# webspeech.js

## simples

- você pode adicionar o reconhecimento de voz no seu site com poucas linhas de javascript
- essa biblioteca simplifica os usos mais comuns da API

## vários recursos

- suporte a reconhecimento contínuo
- exibe resultados provisórios enquanto usuário está falando
- Layout flexível: permite <input> <textarea> <div> <span> e outras tags



/luisleao

# webspeech.js

você pode personalizar o comportamento da  
biblioteca

sobrescrevendo praticamente todos os atributos e funções desta  
biblioteca



/luisleao

# dicas

use em sites com SSL: o navegador grava a permissão do usuário

pontuação falada só em inglês =/



/luisleao

# usabilidade é importante!



# usabilidade

indicador quando estiver com o reconhecimento ativo

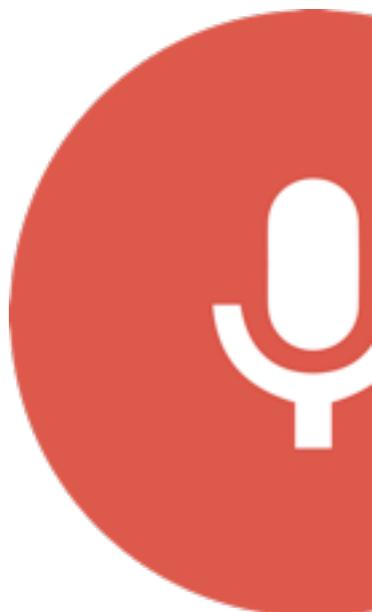
Fale agora...

mostrar progresso do reconhecimento

Texto final e texto em andamento

indicadores de erros

Não reconhecemos nenhuma fala. Talvez você precise ajustar as [configurações de microfone](#).



# usabilidade

deixe o usuário controlar quando o microfone está ativo!

- Botão para **INICIAR**
- Botão para **PARAR**
  - necessário no reconhecimento contínuo
  - o usuário pode não querer esperar pelo timeout, ou
  - ambientes com ruído podem não parar automaticamente quando o usuário terminar de falar



/luisleao

# tratamento de erros

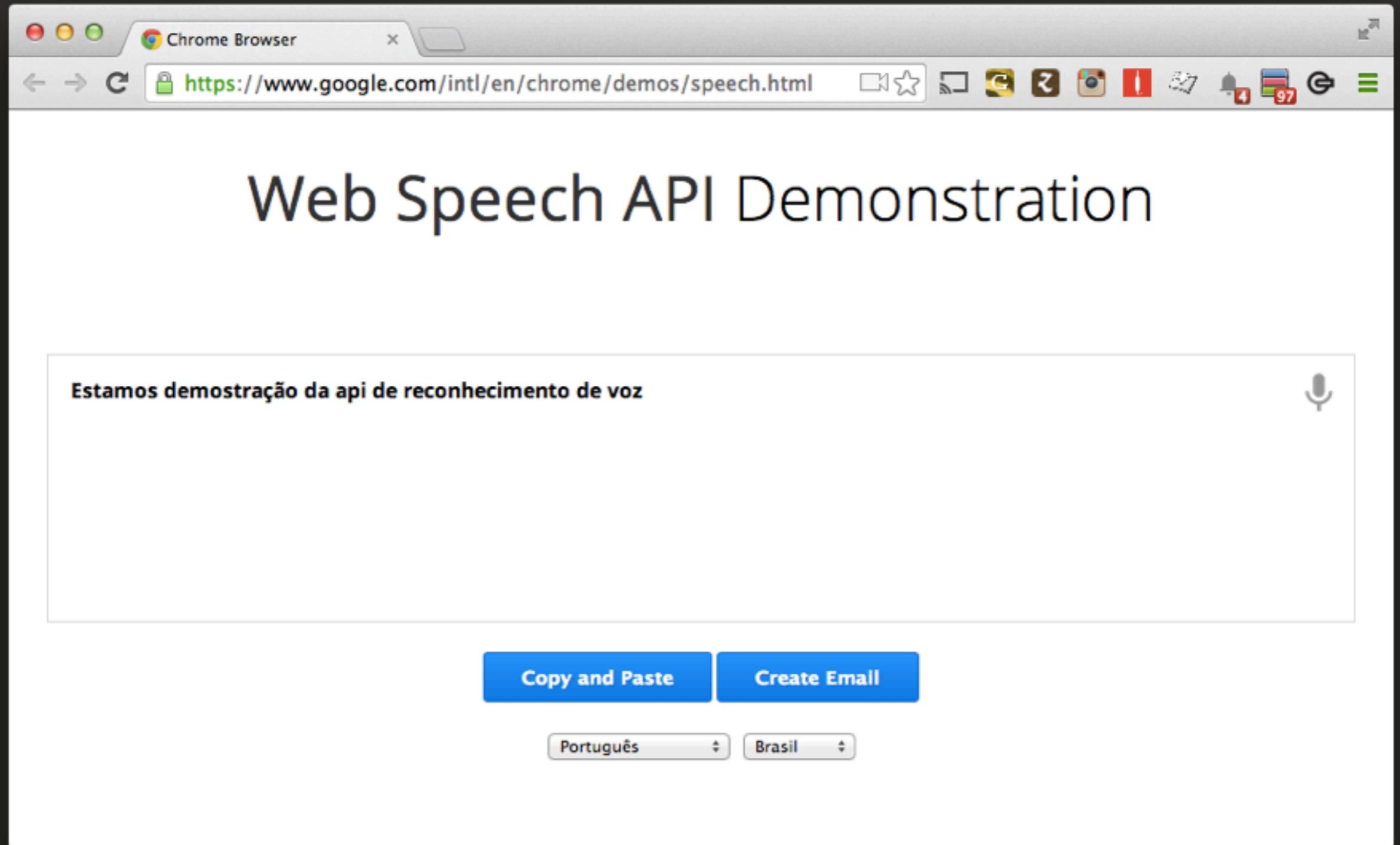
- Ações devem ser fáceis de desfazer e corrigir
- Torne fácil selecionar e corrigir textos
- Permita que o usuário cancele e tentar novamente



# EXEMPLOS & DEMOS



/luisleao



<https://www.google.com/intl/en/chrome/demos/speech.html>

The

**PEANUT**

*Gallery*

The  
**PEANUT**  
*Gallery*

apresenta:



SELECIONE UM CLIPE DE FILME PARA COMEÇAR

[www.peanutgalleryfilms.com](http://www.peanutgalleryfilms.com)

# 21:15

diga "oi câmara"

<http://github.com/luisleao/oicamara>

<https://github.com/luisleao/oicamara>

# Recursos

- W3C Community Group:  
[www.w3.org/community/speech-api/](http://www.w3.org/community/speech-api/)
- Especificação da WebSpeech API:  
<http://dvcs.w3.org/hg/speech-api/raw-file/tip/speechapi.html>
- Demos:  
[www.peanutgalleryfilms.com](http://www.peanutgalleryfilms.com)
- Tutorial e demo de escrita de e-mail:  
<http://updates.html5rocks.com/2013/01/Voice-Driven-Web-Apps-Introduction-to-the-Web-Speech-API>
- Biblioteca webspeech.js e Codelab:  
<http://goo.gl/vgsDn>



/luisleao

# OBRIGADO!



[/luisleao/webbr2013](https://github.com/luisleao/webbr2013)

[eu@luisleao.com.br](mailto:eu@luisleao.com.br)