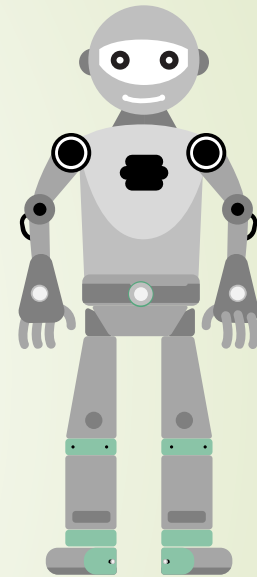


# Creando un bot con Bot Framework y LUIS



Luis Beltrán

Tomás Bata University in Zlín

# Luis Beltrán

- Estudiante de Doctorado en la Universidad Tomás Bata (Zlín, República Checa)
- Docente en el Instituto Tecnológico de Celaya (México)
- MVP en Visual Studio y Tecnologías de Desarrollo
- Xamarin Certified Mobile Developer



## CONTACTO

**Email:** [luis@luisbeltran.mx](mailto:luis@luisbeltran.mx)

**Twitter:** [darkicebeam](https://twitter.com/darkicebeam)

**LinkedIn:** [/in/luisantonibeltran](https://in.luisantonibeltran)

**Blog:** <https://luisbeltran.mx>

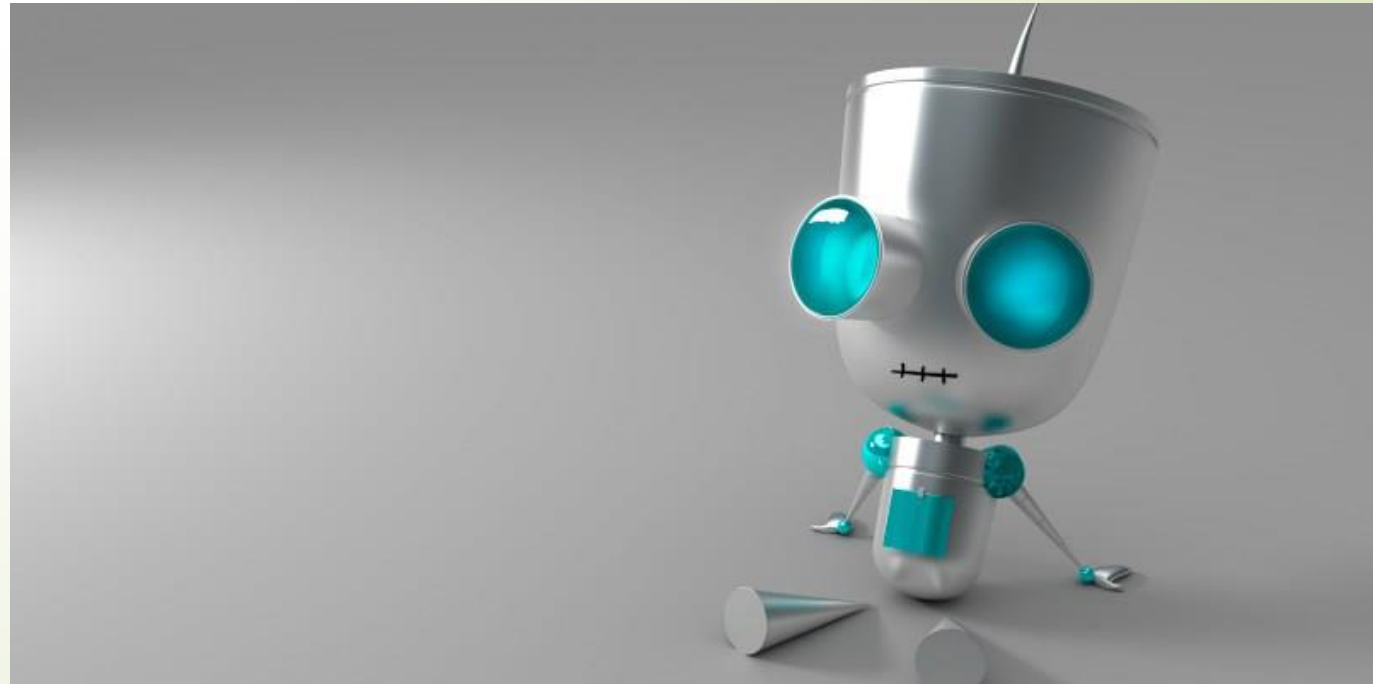


# Agenda

- Hablemos de Bots
- Conociendo Bot Framework
- LUIS y cómo añadir inteligencia y lenguaje natural a un servicio
- Demo time!
- Q&A

# ¿Qué es un bot?

- Un bot es un sistema que ha sido diseñado específicamente para realizar una tarea (simple o compleja).
- Por ejemplo, puedes utilizar un bot para reservar un vuelo, una orden de compra o preguntar sobre el clima.
- Un bot no es únicamente conversacional; también puede tener un botón, una interfaz o alguna otra interfaz para interactuar con el usuario.





# Características de un bot

- Un bot no solo devuelve respuestas a consultas
- Actúa como un agente para el usuario
- Detecta la intención del usuario
- Realiza acciones
- Combina (miles/millones de) datos de diferentes fuentes
- Mantiene un contexto del diálogo
- Puede presentar personalidad







# Algunas predicciones

65% de los usuarios de teléfonos inteligentes utilizan asistentes de voz en sus teléfonos

80% de los ejecutivos creen que la inteligencia artificial mejora el desempeño de los empleados y crean trabajos

Para 2018, al menos 50% de las nuevas versiones de software empresarial incluirán algunas capacidades basadas en CaaP.

Para 2020, al menos 80% de las nuevas apps empresariales tendrán un considerable uso de chat bots.



# Momentum



800 M

Usuarios mensuales de WeChat activos, 200M con tarjetas de crédito enlazadas a su cuenta



67,000

Desarrolladores en el Microsoft Bot Framework



55,000

Desarrolladores de bots en Facebook

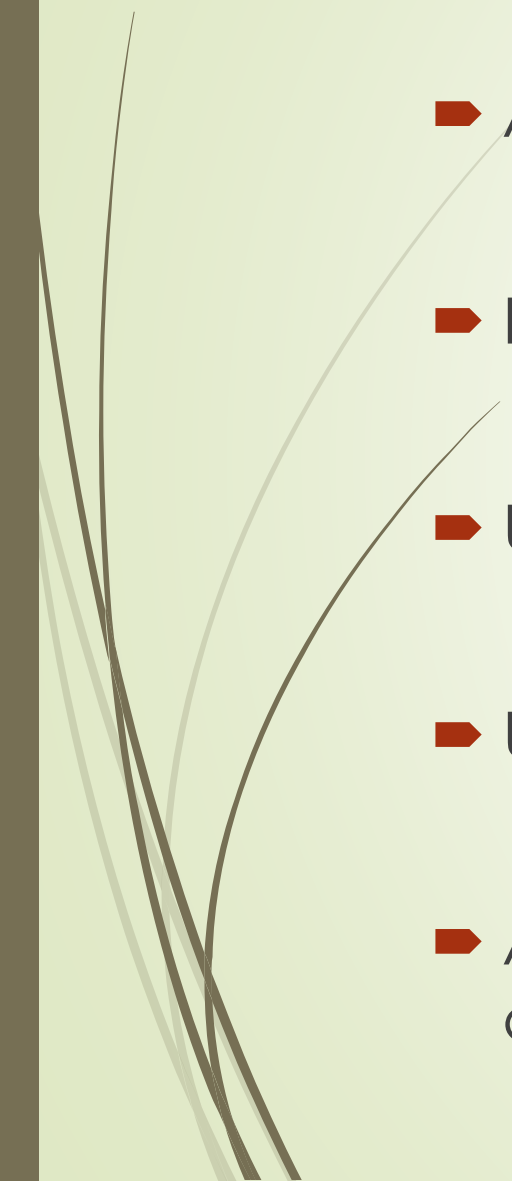


15,000

Skills disponibles en la Amazon Alexa Store



# Potenciales beneficios

- Ajustar el negocio a la “generación del chat”
  - Bajo costo de implementación para proporcionar servicios.
  - Un mejor enfoque de recursos a otras tareas más críticas
  - Una rápida conexión de los clientes con información y servicios
  - Actúa como un navegador simple para manejo de datos complicados
- 

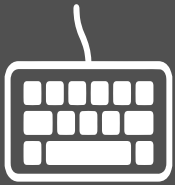


# Diseñando un bot

- Un bot que procesa datos
  - En su forma más simple, un bot acepta una entrada de datos, la procesa y produce una salida. Este es el modelo clásico de una computadora.
  - Una buena forma de describir un bot es como un mago que guía al usuario a través de un proceso. Cuando estás reservando un corte de cabello o cambiando una contraseña, no necesitas una conversación, simplemente esperas un resultado. El contexto es importante al diseñar un bot.
- Entender el lenguaje natural no es necesario
  - Puedes utilizar expresiones regulares o analizar texto
- El contexto es clave
  - Una forma crucial de mejorar un bot es que mantenga el context durante la conversación, de manera que cuando el usuario diga “eso”, “esto”, “aquello”, etc. haciendo referencia a “el libro que estoy comprando” el bot pueda entender y responder de acuerdo a lo esperado.

# ¿Por qué un bot?

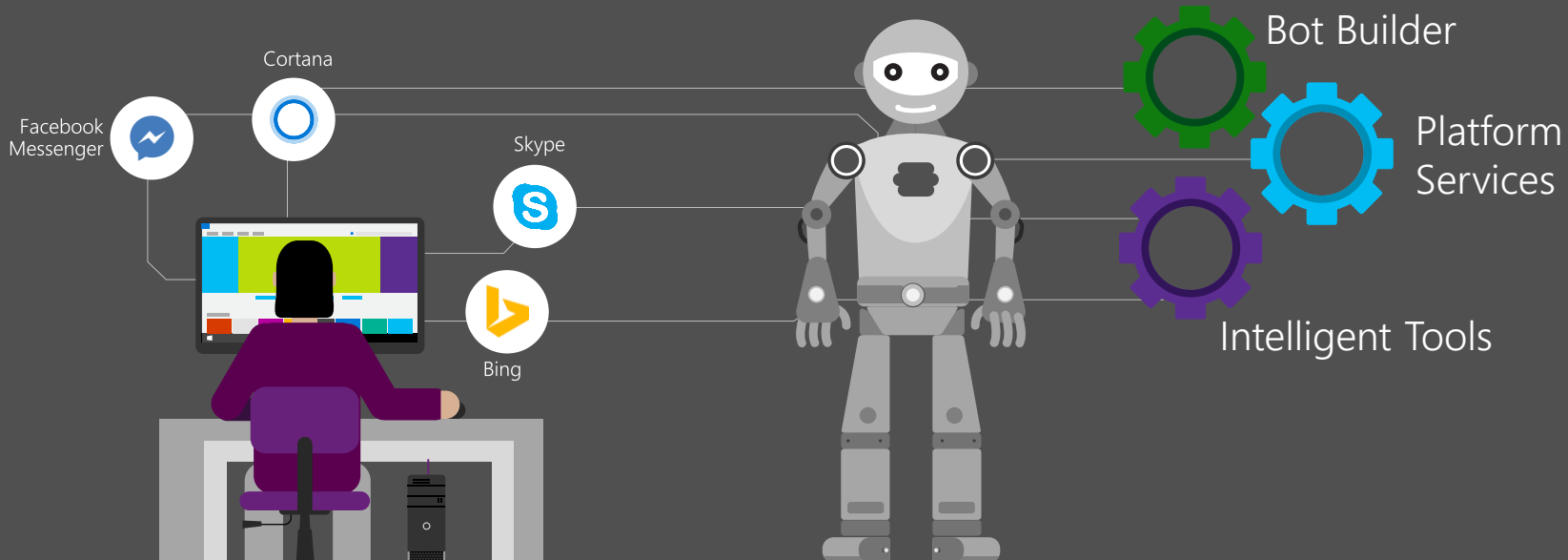
ESCRIBE



HABLA



CLICK



# ¿Qué es el Bot Framework?

## ¿Que?

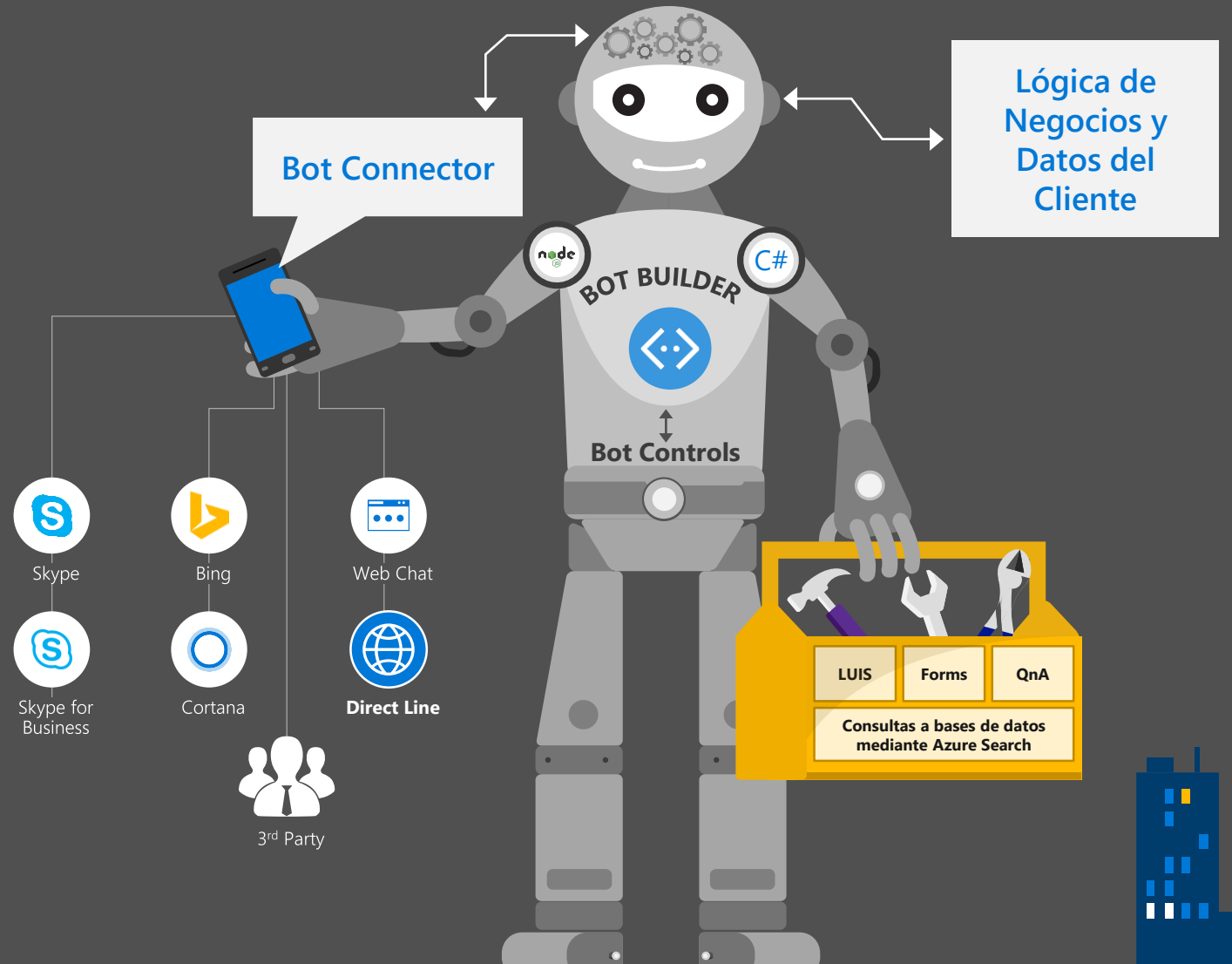
- Herramientas para construir sitios web REST
- Servicios para enriquecer la experiencia
- Mecanismos para recibir eventos
- Datos para depurar y analizar

## ¿Por qué?

- Implementa protocolos estándar
- Modelar conversaciones es difícil. ¡Las herramientas ayudan!
- El entendimiento del lenguaje es difícil
- Patrones comunes y bien entendidos

## Objetivos

- Comience simple. Agrega Complejidad.
- El Bot se adapta al usuario, según el contexto
- Controles inteligentes para gestionar la complejidad





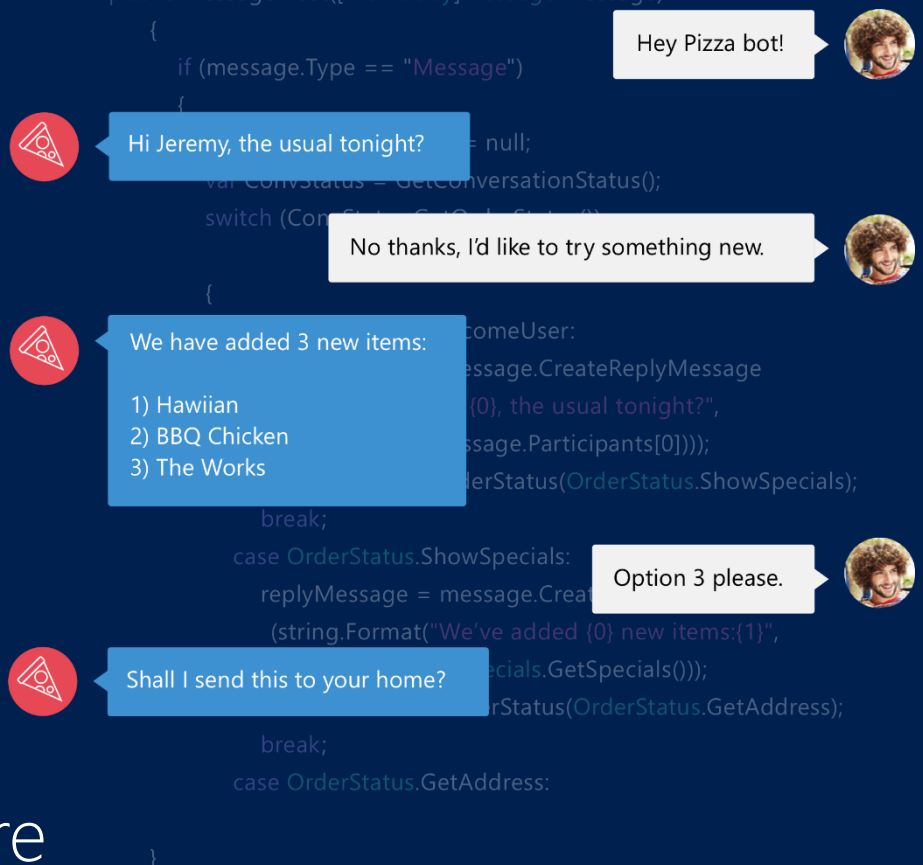
# Bot Framework

Bot Framework es un servicio y un SDK operados por Microsoft.

Bot Framework es una de varias herramientas que Microsoft ofrece para construir un Bot completo.

Otras herramientas: LUIS, APIs de Speech, Azure Bot Service y más

```
public Message Post([FromBody]Message message)
{
    if (message.Type == "Message")
    {
        if (message.From == null)
        {
            var ConvStatus = GetConversationStatus();
            switch (ConvStatus)
            {
                case ConversationStatus.Started:
                    // Welcome user
                    message.CreateReplyMessage(
                        "Hey Pizza bot!",
                        message.Participants[0]));
                    break;
                case ConversationStatus.Continued:
                    // User has returned
                    message.CreateReplyMessage(
                        "Hi Jeremy, the usual tonight?",
                        message.Participants[0]));
                    break;
                case ConversationStatus.ShowSpecials:
                    // User wants to see specials
                    message.CreateReplyMessage(
                        "We have added 3 new items:",
                        message.Participants[0]));
                    break;
                case ConversationStatus.GetAddress:
                    // User wants to know address
                    message.CreateReplyMessage(
                        "Shall I send this to your home?",
                        message.Participants[0]));
                    break;
            }
        }
    }
}
```



Hey Pizza bot!

Hi Jeremy, the usual tonight?

No thanks, I'd like to try something new.

We have added 3 new items:

- 1) Hawaiian
- 2) BBQ Chicken
- 3) The Works

Option 3 please.

Shall I send this to your home?



Your bots — wherever your users are talking.

Build and connect intelligent bots to interact with your users naturally wherever they are, from text/sms to Skype, Slack, Office 365 mail and other popular services.

Hey Pizza bo

Hi Jeremy, the usual tonight?

No thanks, I'd like to try something new.

We have added 3 new items:

- 1) Hawaiian
- 2) BBQ Chicken
- 3) The Works

<http://docs.botframework.com>

Bot Framework Overview

The Microsoft Bot Framework provides just what you need to build and connect intelligent bots that interact naturally wherever your users are talking, from text/sms to Skype, Slack, Office 365 mail and other popular services.

Your Bot Framework Bot		
<b>Bot Connector</b> Connect your bot(s) to text/sms, Office 365 mail, Skype, Slack, Facebook Messenger and more...	<b>Bot Builder SDKs</b> Build great dialogs within your Node.js- or C#-based bot	<b>Bot Directory</b> <i>COMING SOON</i> - Try, use, and add published bots to the world's top conversation experiences
<ul style="list-style-type: none"><li>• Register, connect, publish and manage your bot through the bot dashboard</li><li>• Message routing</li><li>• Automatic translation to 30+ languages</li><li>• User and state management</li><li>• Embeddable web chat control</li><li>• Debugging tools</li></ul>	<ul style="list-style-type: none"><li>• Open source SDK on Github</li><li>• From simple built-in prompts and command dialogs to simple to use yet sophisticated 'FormFlow' dialogs</li><li>• Libraries, samples and tools to make a great conversationalist</li><li>• Chat emulator</li><li>• Leverage related services</li></ul>	<ul style="list-style-type: none"><li>• Public directory of bots registered and approved with Bot Framework</li><li>• Users can try your bot from the directory via the web chat control</li><li>• Users can discover and add your bot to the channels on which it is configured</li></ul>

## Options:

- Connector
- Builder
- Directory



# Bot Connector

Getting started | Docum × +

docs.botframework.com/connector/getstarted

The Microsoft Bot Framework Connector is a communication service that helps you connect your Bot with many different communication channels (GroupMe, SMS, email, and others). If you write a conversational Bot or agent and expose a Microsoft Bot Framework-compatible API on the internet, the Connector will forward messages from your Bot to a user, and will send user messages back to your Bot.

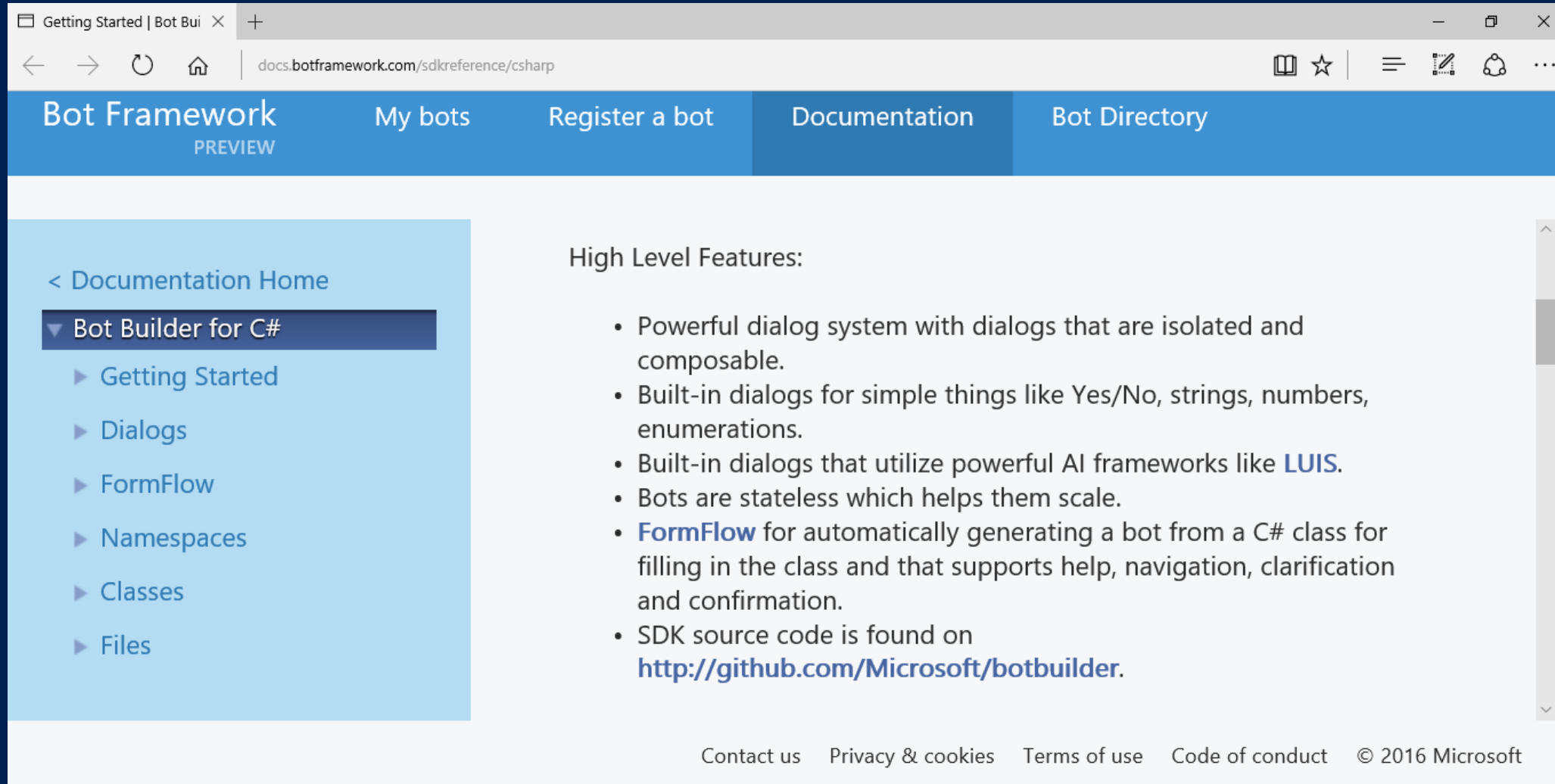
The diagram illustrates the Bot Connector architecture. On the left, a box labeled 'Bots' contains 'Your bot's web service'. In the center, a box labeled 'Bot Connector' lists its functions: 'Routes messages', 'Manages state', 'Bot registration, directory', 'Session tracking', 'Services (translation...)', and 'Per-user, per-bot storage SDK, APIs'. On the right, a box labeled 'Channels' lists various communication methods: 'Email', 'GroupMe', 'Skype', 'Slack', 'SMS', 'Telegram', 'Web (chat control)', and 'additional channels'. Double-headed arrows connect the 'Bots' box to the 'Bot Connector' box, and the 'Bot Connector' box to the 'Channels' box, indicating bidirectional communication.

## Channels:

- Email
- GroupMe
- Skype
- Slack
- SMS
- Telegram
- Web
- etc...

<http://docs.botframework.com/connector/getstarted>

# Bot Builder SDK for C#



The screenshot shows a web browser window with the URL [docs.botframework.com/sdkreference/csharp](https://docs.botframework.com/sdkreference/csharp). The page has a blue header with navigation links: Bot Framework PREVIEW, My bots, Register a bot, Documentation (active), and Bot Directory. On the left, a sidebar shows the navigation structure: < Documentation Home, Bot Builder for C# (expanded), and sub-items: Getting Started, Dialogs, FormFlow, Namespaces, Classes, and Files. The main content area is titled 'High Level Features:' and lists several bullet points. At the bottom, there is a footer with links for Contact us, Privacy & cookies, Terms of use, Code of conduct, and a copyright notice for © 2016 Microsoft.

Getting Started | Bot Bui X +

docs.botframework.com/sdkreference/csharp

Bot Framework PREVIEW My bots Register a bot Documentation Bot Directory

< Documentation Home

▼ Bot Builder for C#

- ▶ Getting Started
- ▶ Dialogs
- ▶ FormFlow
- ▶ Namespaces
- ▶ Classes
- ▶ Files

High Level Features:

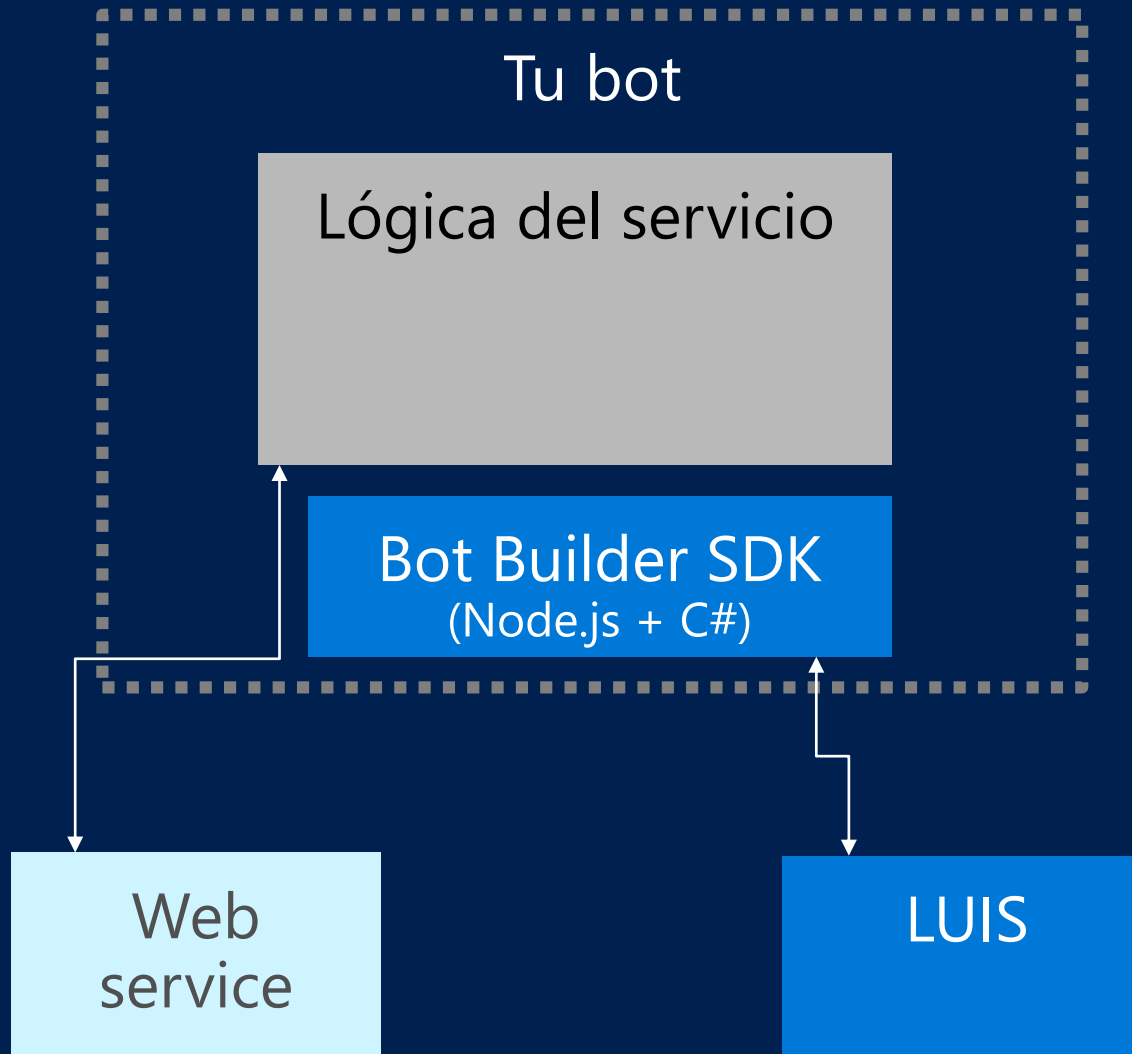
- Powerful dialog system with dialogs that are isolated and composable.
- Built-in dialogs for simple things like Yes/No, strings, numbers, enumerations.
- Built-in dialogs that utilize powerful AI frameworks like **LUIS**.
- Bots are stateless which helps them scale.
- **FormFlow** for automatically generating a bot from a C# class for filling in the class and that supports help, navigation, clarification and confirmation.
- SDK source code is found on <http://github.com/Microsoft/botbuilder>.

Contact us Privacy & cookies Terms of use Code of conduct © 2016 Microsoft

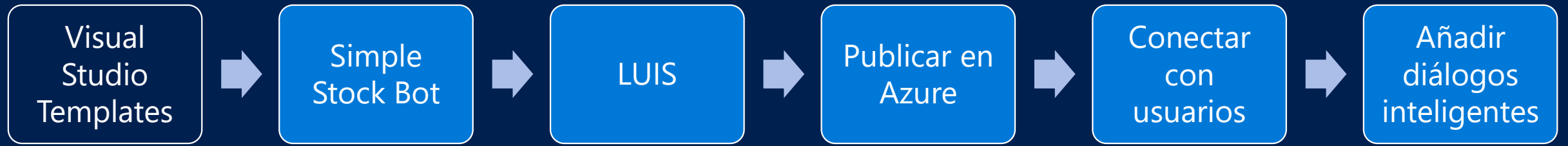
## Features:

- Dialogs
- AI
- LUIS
- Stateless
- Scalable
- FormFlow
- GitHub!

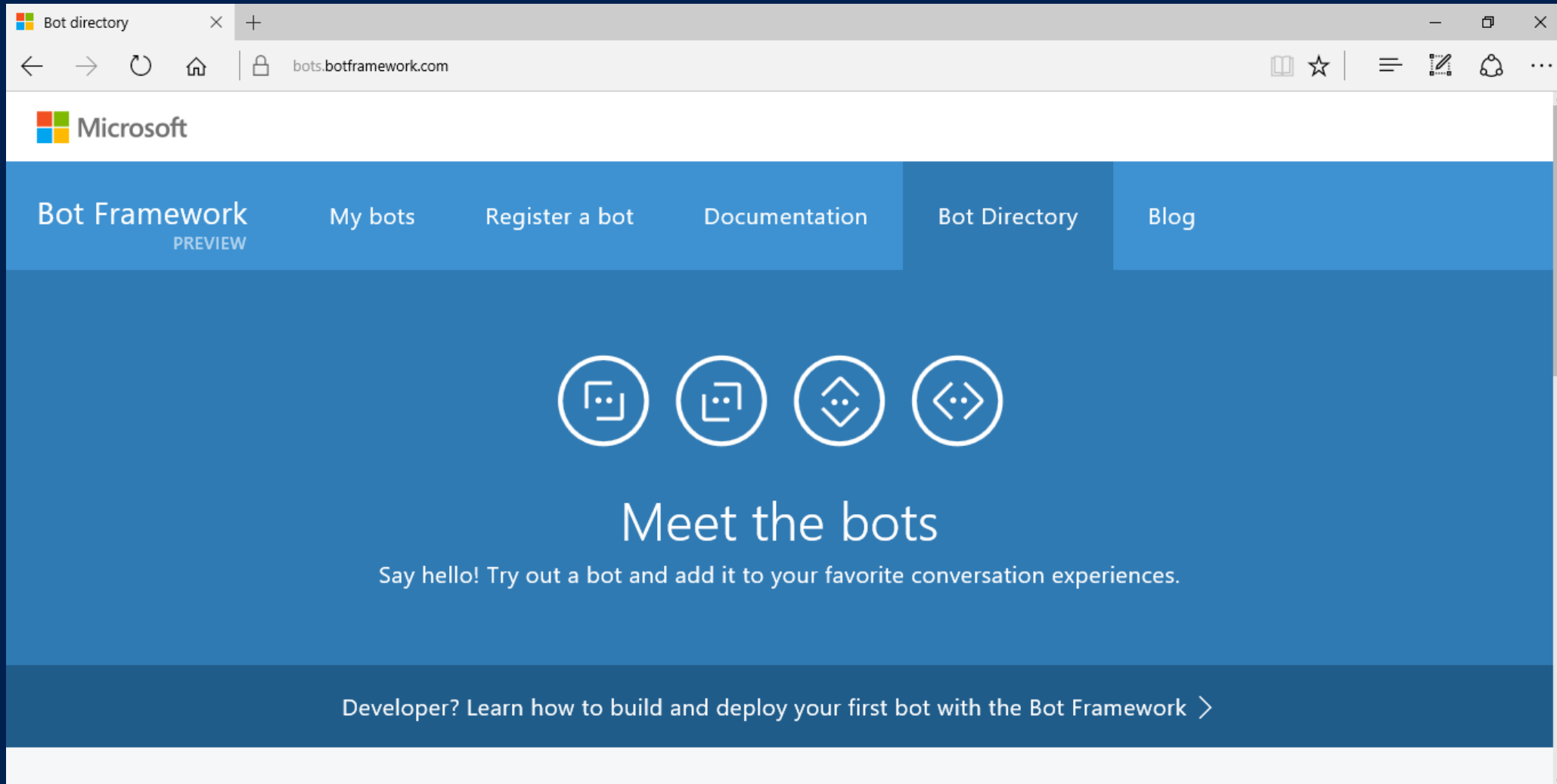
<http://docs.botframework.com/sdkreference/csharp>



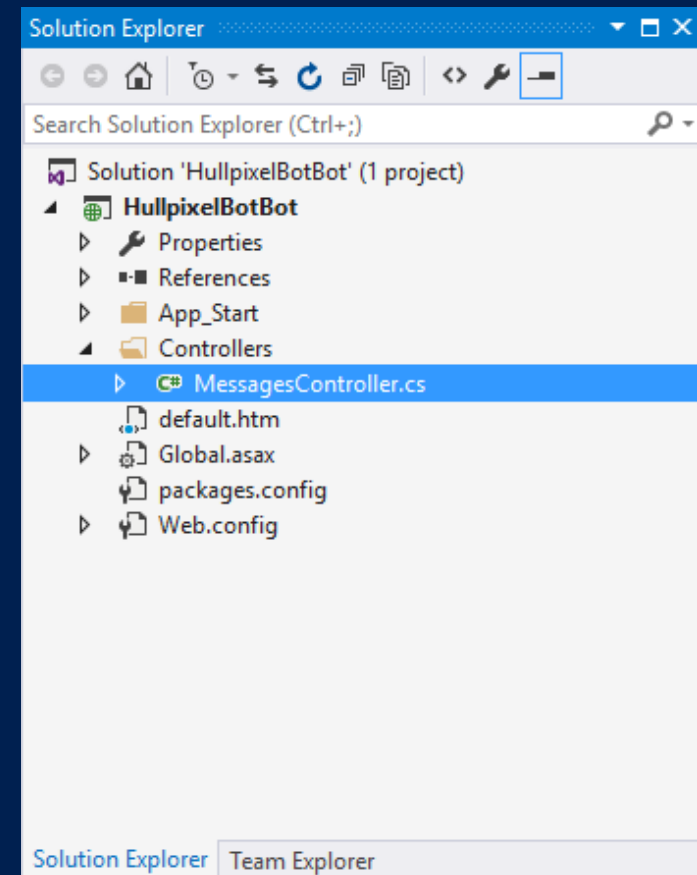
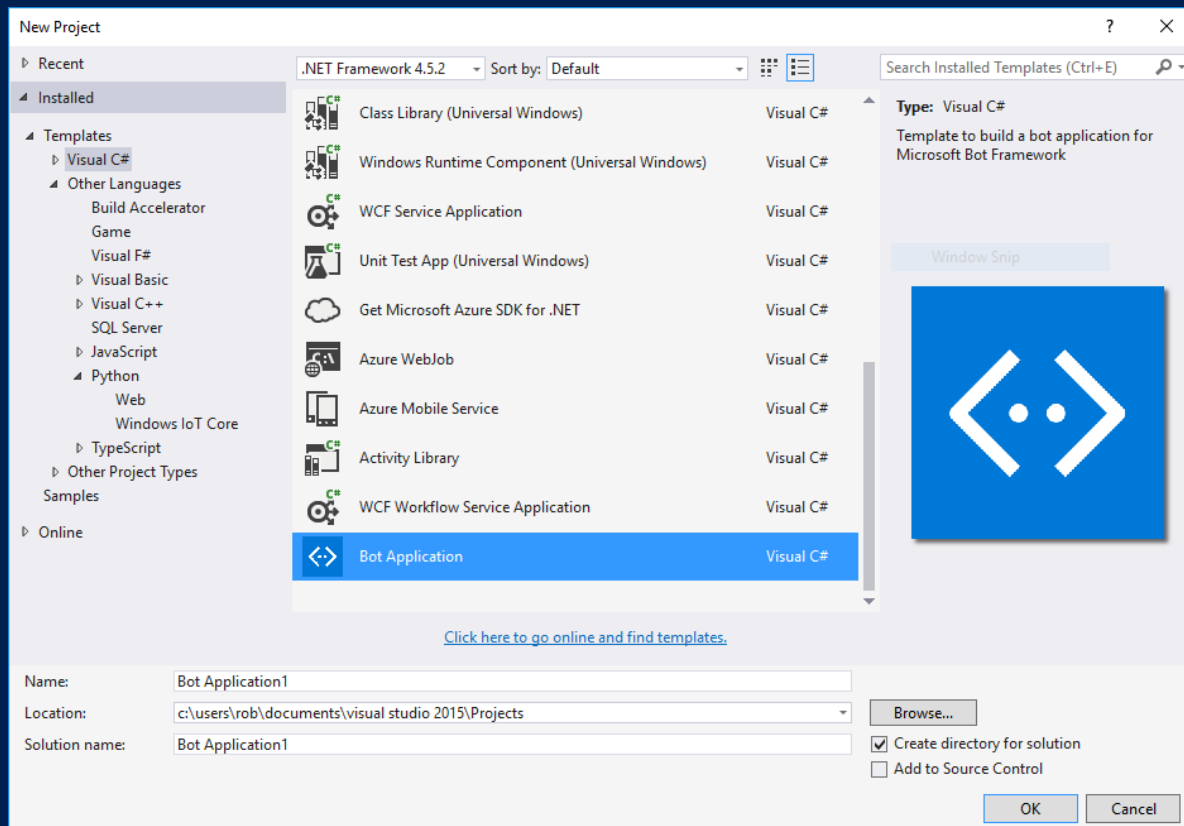




# Bot Directory



<https://bots.botframework.com/>





## 1990s GUI e Internet

- Búsquedas
- Usuario visita sitios web

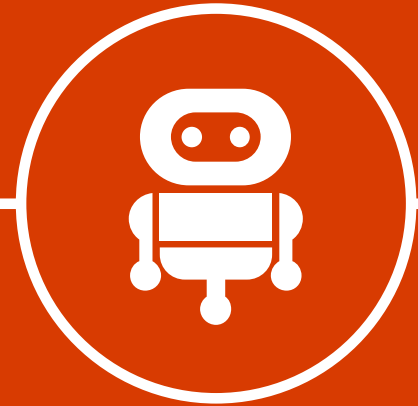
## Un nuevo canvas: Conversations

- Lenguaje natural entre las personas y la tecnología
- Contextual
- Bots y agentes

## 1980s: PC

- Command Line Interface

## 2000s NUI y Mobile

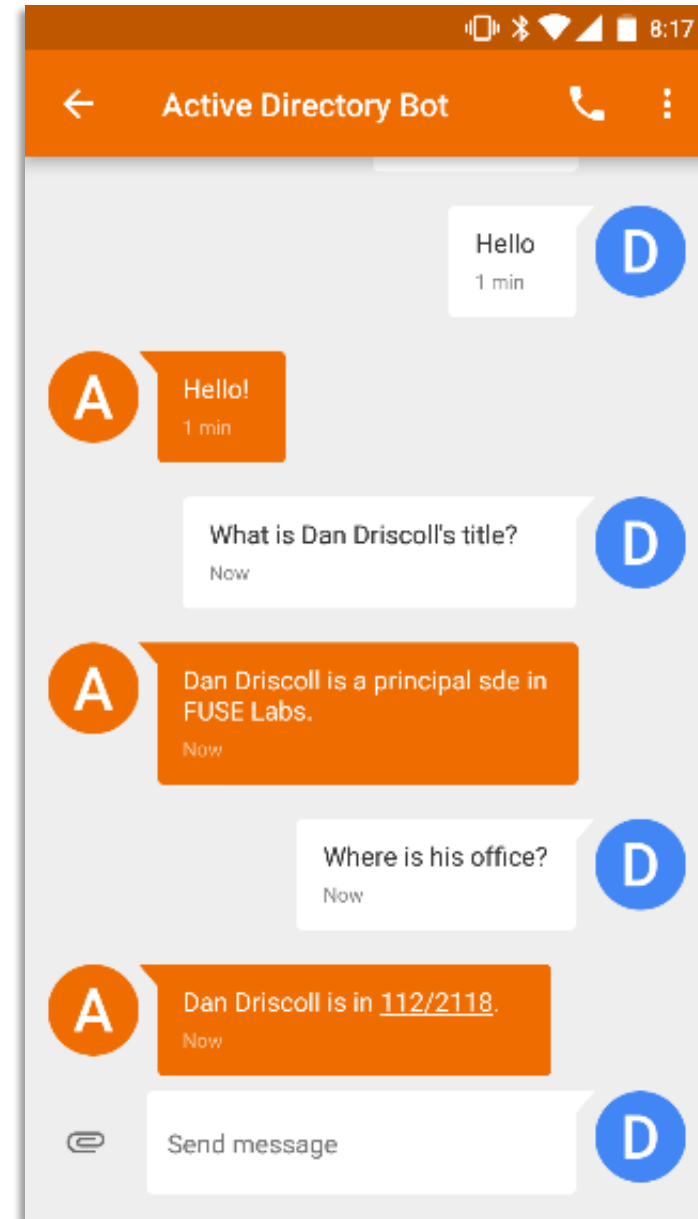


# Language Understanding

¿Cuándo llega el vuelo AF32?

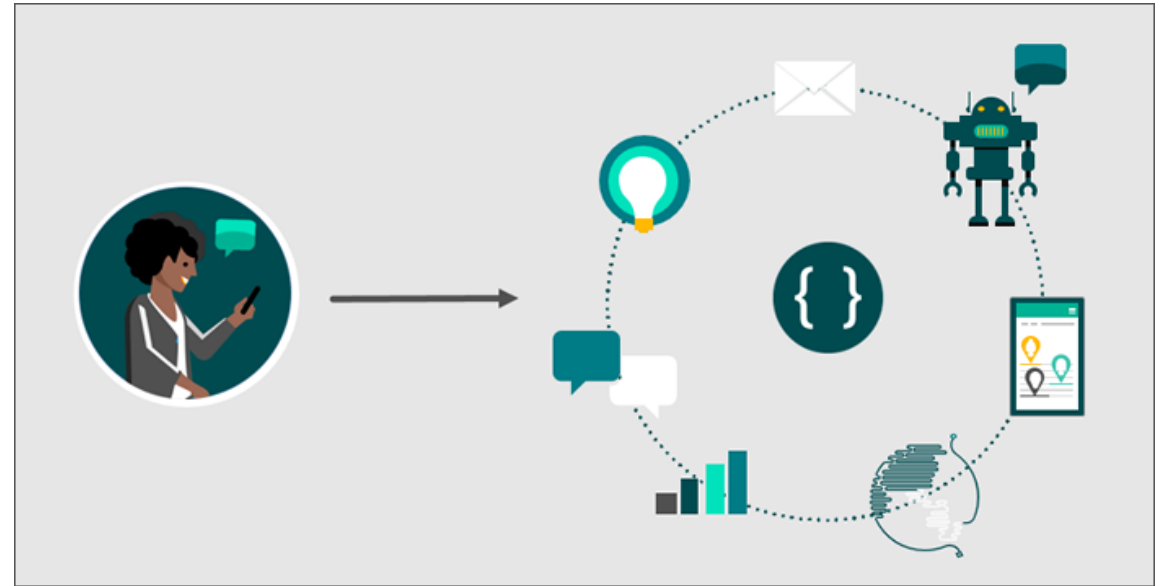
¿Dónde está el baño más cercano?

Por favor reserva un vuelo a Orlando.



# Microsoft Language Understanding Intelligent Service (LUIS)

Language Understanding Intelligent Service (LUIS) es una tecnología interactiva de aprendizaje automático y comprensión de idiomas y conjunto de herramientas diseñado para agregar fácilmente la comprensión del lenguaje y la interpretación semántica a las aplicaciones.





# Language understanding intelligent service (LUIS)

Reduce el esfuerzo de etiquetado con funciones interactivas

Use visualizaciones para medir el rendimiento y las mejoras

Aproveche el reconocimiento de voz con una integración perfecta

Despliegue usando solo algunos ejemplos con aprendizaje activo

Define  
conceptos



Proporciona  
ejemplos



Despliega

Aprendizaje activo





# See Language Understanding in action

What the user says (utterances)

Book me a flight to Cairo

Order me 2 pizzas

Remind me to call my dad tomorrow

Where is the nearest club?



What LUIS returns

```
{
  "query": "Book me a flight to Cairo",
  "topScoringIntent": {
    "intent": "BookFlight",
    "score": 0.9887482
  },
  "intents": [
    {
      "intent": "BookFlight",
      "score": 0.9887482
    }
  ]
}
```



# Language Understanding Concepts

LUIS es parte de la oferta de Servicios Cognitivos de Microsoft que se puede usar para cualquier dispositivo, en cualquier plataforma y en cualquier escenario de aplicación.

- Es, en esencia, lo que Cortana usa en el backend para el lenguaje y las interpretaciones semánticas.
- Proporciona una lógica "incorporada" que se puede aprovechar "out of the box"
- Inteligencia natural, adaptable y conversacional
- Basado en modelos a través de intents y entidades

# Terminología de LUIS

- Entities
- Intents
- Labels
- Utterances
- Phrase lists
- Training
- Publishing

The screenshot displays the LUIS portal interface. On the left is a navigation sidebar for the application 'Newsy' (Version: 0.1), containing links for Settings, Dashboard, Intents, Entities, Prebuilt domains (marked as PREVIEW), Features, Train & Test, and Publish App. The main content area is titled 'Entities' and includes the instruction: 'Manage a list of entities in your application and track and control their instances with'. Below this is an 'Entities list' table with a yellow 'Add custom entity' button. The table has columns for 'Entity Name' and 'Prebuilt Entity'. It lists several entities: 'NewsCategory', 'NewsTopic', and 'encyclopedia'. A modal window titled 'Add prebuilt entities' is open in the foreground, showing a list of prebuilt entities with checkboxes to enable them. The entities listed in the modal are: 'age' (Age of a person or thing), 'datetimeV2 (NEW)' (Date time extractor and resolver), 'dimension' (Spatial dimensions, including length, distance, area, and volume), 'email' (Email addresses), and 'geography'. The modal has 'Save' and 'Cancel' buttons at the bottom.

Entity Name	Prebuilt Entity
NewsCategory	
NewsTopic	
encyclopedia	

Enable	Prebuilt Entity
<input type="checkbox"/>	<b>age</b> Age of a person or thing 10-month-old, 19 years old, 58 year-old
<input type="checkbox"/>	<b>datetimeV2 (NEW)</b> Date time extractor and resolver Date time extractor and resolver
<input type="checkbox"/>	<b>dimension</b> Spatial dimensions, including length, distance, area, and volume 2 miles, 650 square kilometres, 9,350 feet
<input type="checkbox"/>	<b>email</b> Email addresses user@site.net, user_name@mysite.com.eg, user.Name12@website.net
<input type="checkbox"/>	<b>geography</b>

# Flujo general



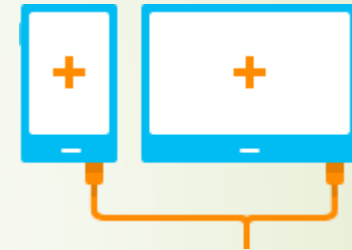
Crear modelos de comprensión del lenguaje.



Use modelos preconstruidos de Bing y Cortana.



Implemente sus modelos en un endpoint HTTP.



Acceda a los modelos en cualquier dispositivo.

# Entidades preconstruidas

LUIS incluye un conjunto de entidades preconstruidas. Cuando se incluye una entidad preconstruida en su aplicación, sus predicciones se incluirán en su aplicación publicada.

Entity	Examples	JSON
<b>builtin.number</b>	<i>ten, 3.1415</i>	{ "type": "builtin.number", "entity": "ten" }
<b>builtin.ordinal</b>	<i>first, 10<sup>th</sup></i>	{ "type": "builtin.ordinal", "entity": "10th" }
<b>builtin.temperature</b>	<i>10 degrees celsius, 78 F</i>	{ "type": "builtin.temperature", "entity": "78 f" }
<b>builtin.dimension</b>	<i>2 miles, 650 square kilometers</i>	{ "type": "builtin.dimension", "entity": "2 miles" }
<b>builtin.money</b>	<i>1000.00 US dollars, \$ 67.5 B</i>	{ "type": "builtin.money", "entity": "1000.00 us dollars" }
<b>builtin.age</b>	<i>100 year old, 10 years old</i>	{ "type": "builtin.age", "entity": "100 year old" }
<b>builtin.geography</b>	<i>seattle, amazon river</i>	{ "type": "builtin.geography.city", "entity": "seattle" }
<b>builtin.encyclopedia</b>	<i>geddy lee, diet coke, super bowl xxxix</i>	{ "type": "builtin.encyclopedia.music.musician", "entity": "geddy lee" }
<b>builtin.datetime</b>	<i>tomorrow, the meeting ends at 5:30 PM</i>	{ "type": "builtin.datetime.date", "entity": "tomorrow", "resolution": { "date": "2015-08-15" } }



# Aprendizaje Activo

Una característica crítica de LUIS es el concepto de aprendizaje "activo".

En un escenario de aprendizaje activo, LUIS revisa todos los enunciados vistos y hace sugerencias sobre las etiquetas recomendadas.

The diagram illustrates the LUIS Active Learning process. It shows two panels. The left panel contains a list of utterances with checkboxes for selection:

- ☐ Utterance text
- ☐ find some great
- ☐ find [golden sta
- ☐ find [game d
- ☐ get some great
- ☐ find [microsoft]

A green arrow points from the utterance "find [game d" to the right panel. The right panel shows a list of suggested intents with checkboxes for selection:

- ☐ Utterance text
- ☐ find some great news about [\$encyclopedia]
- ☐ find [\$encyclopedia ] news
- ☐ find [\$encyclopedia ] news
- ☐ get some great news about [\$encyclopedia]
- ☐ find [\$encyclopedia] news

Both panels have a header with the following options: Save, Discard, Delete, and Reassign Intent (with a dropdown arrow).

# Entrenando un modelo

El entrenamiento de un modelo es el término que se utiliza cuando LUIS "generaliza" los ejemplos que ha etiquetado, y desarrolla un código para reconocer los intentos y las entidades aplicables en el futuro.



# Publicación de modelos

Antes de que un modelo LUIS pueda estar disponible para una aplicación, debe publicarse.

Las pruebas y publicaciones se pueden realizar a través de la misma interfaz.

Publish settings

Endpoint slot

Production

Slot info

Published version Id: 0.1

Published date: May 25, 2017 9:26:58 PM

Endpoint url

<https://westus.api.cognitive.microsoft.com/luis/v2.0/apps/be2aa692-8ff3-44fe-9027-cdd6451877822f99592d88&timezoneOffset=0&verbose=true&q=>

☒ Add verbose flag

☐ Enable Bing spell checker

Train

Publish

# Integrando LUIS

LUIS ofrece un conjunto de APIs que los desarrolladores pueden usar para automatizar el proceso de creación de aplicaciones.

Estas API también le permiten crear y publicar su aplicación en un punto final seguro para tener acceso a las experiencias de su

## Query

set an alarm for 3:00 pm

## URL

<https://api.projectoxford.ai/luis/v2/?q=31f271e5477398c77428cc4455>

```
{
  "query": "set an alarm for 3:00 pm",
  "intents": [
    {
      "intent": "builtin.intent.alarm.set_alarm"
    }
  ],
  "entities": [
    {
      "entity": "3:00 pm",
      "type": "builtin.alarm.start_time",
      "resolution": {
        "resolution_type": "builtin.datetime.time",
        "time": "T15:00"
      }
    }
  ]
}
```



# Pasos para implementar un bot

1. Crea un Proyecto de Visual Studio
2. Registra el bot con Microsoft Bot Framework
3. Construye el bot utilizando el Microsoft Bot Connector API
4. Pruébalo por medio del Bot Framework Emulator
5. Desplégalo a un servicio en la nube
6. Crea un app package
7. Realiza un sideload del app package o publícalo en la tienda





# Demo



# Q&A



# Enlaces de interés

Sitio web de LUIS

<https://luis.ai>

Microsoft Bot Framework

<https://dev.botframework.com/>

Comunidad Xamarin en Español

<https://www.facebook.com/groups/xamarindiplomadoitc/>

Aprendiendo Azure

<https://www.facebook.com/groups/AprendiendoAzure/>



# ¡Gracias por tu atención!

Luis Beltrán

Tomás Bata University in Zlín



## CONTACTO

**Email:** [luis@luisbeltran.mx](mailto:luis@luisbeltran.mx)

**Twitter:** [darkicebeam](https://twitter.com/darkicebeam)

**LinkedIn:** [/in/luisantonibeltran](https://in.luisantonibeltran)

**Blog:** <https://luisbeltran.mx>

