Introducción a los bots Webex Teams

Febrero 2020

Agenda

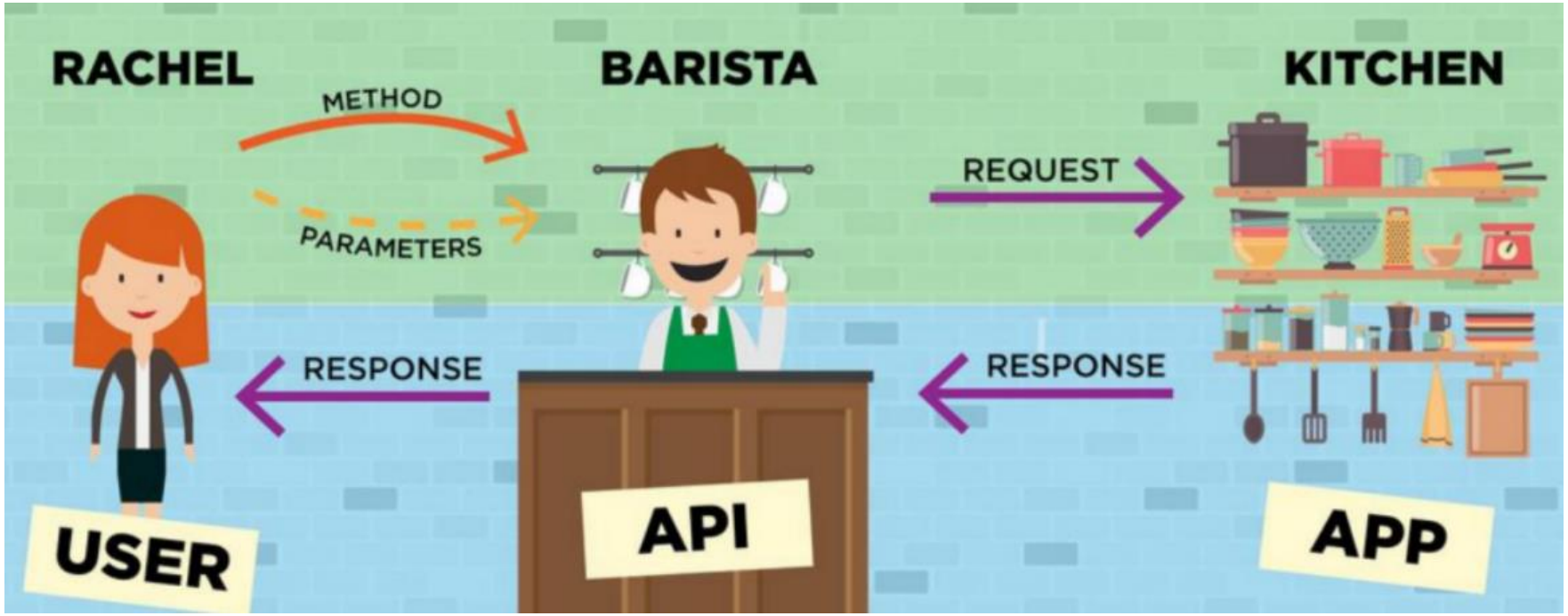
1. Introducción
2. ¿Qué es una API?
3. ¿Qué es un bot?
4. Uso de cards/buttons en los bots
5. Como se crea un bot como LatinCUG
6. Ejemplos de bots creados por Makenai

1. Introducción

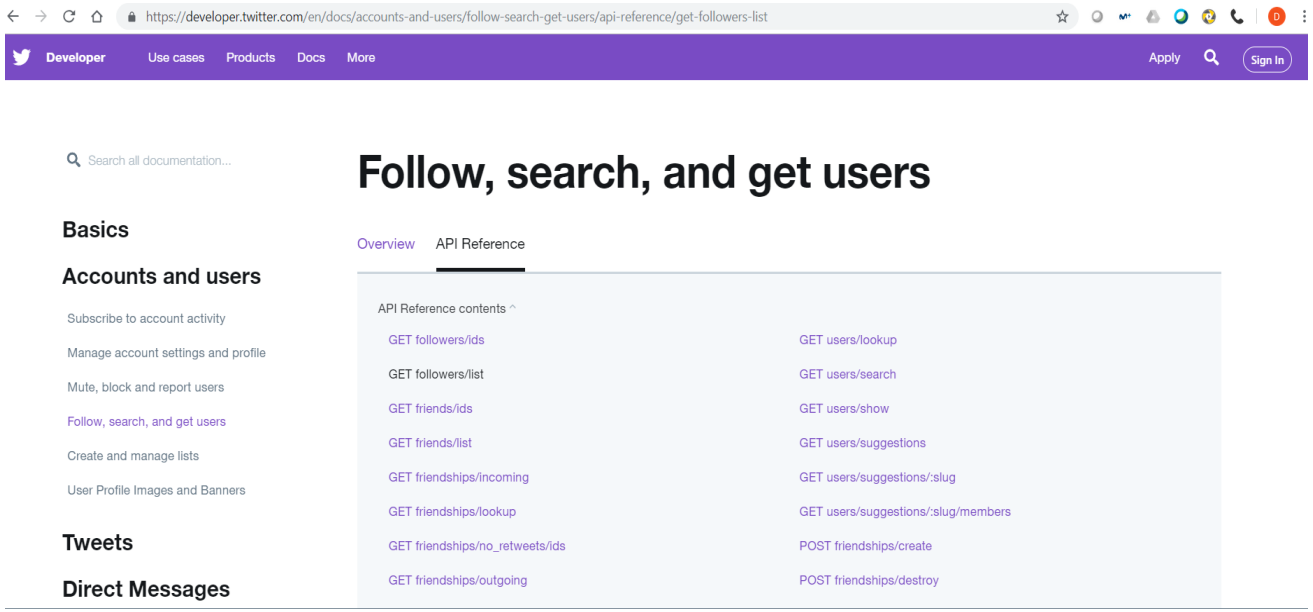
2. ¿Qué es una API?

Basics. APIs

Una **API** (Application Programming Interface) es básicamente una forma de hacer una solicitud desde una aplicación a otra aplicación.



Basics. APIs



https://developer.twitter.com

Ejemplo: si estamos desarrollando una aplicación y queremos mostrar el dato de las personas que nos siguen en Twitter, podríamos solicitar esta información a la **API** de Twitter:

"Dame una lista de todas las personas que me siguen".

No hay problema si nuestra aplicación está desarrollada en un lenguaje de programación y Twitter otro lenguaje porque la **API** transmite y recibe datos en un formato común (normalmente **JSON** o **XML**). Esto permite que dos aplicaciones totalmente separadas envíen y reciban datos.

Basics. APIs

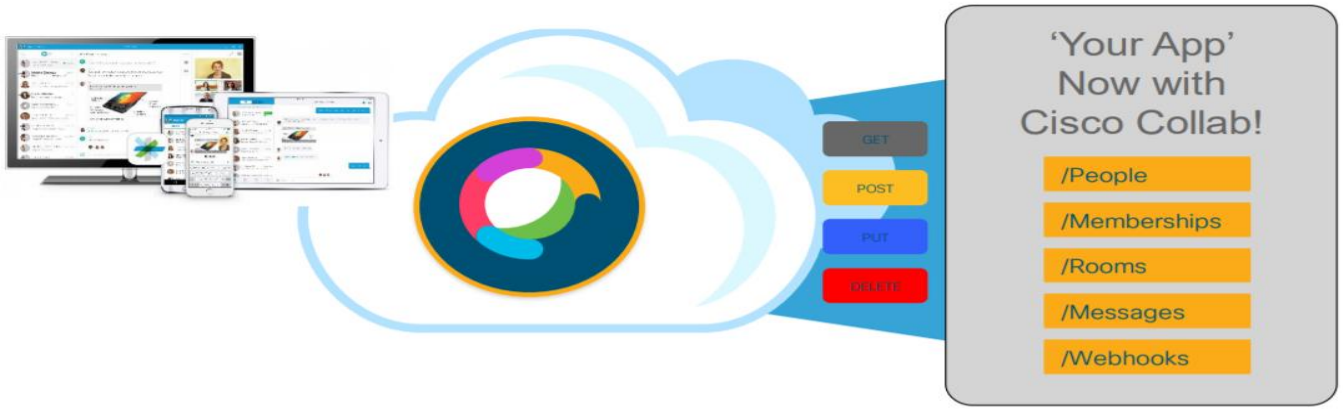
```
POST /v1/messages
{
    "recipient_type": "individual" | "group",
    "to": "whatsapp-id" | "whatsapp_group_id",
    "type": "audio" | "document" | "image",

    "audio": {
      "id": "your-media-id",
    }

    "document": {
       "id": "your-media-id",
       "caption": "your-document-caption"
    }

    "image": {
       "id": "your-media-id",
       "caption": "your-image-caption"
    }
}
```

https://www.whatsapp.com/business/api
https://developers.facebook.com/docs/whatsapp

Basics. Webex APIs

Extiende la Cloud de Colaboración de Cisco

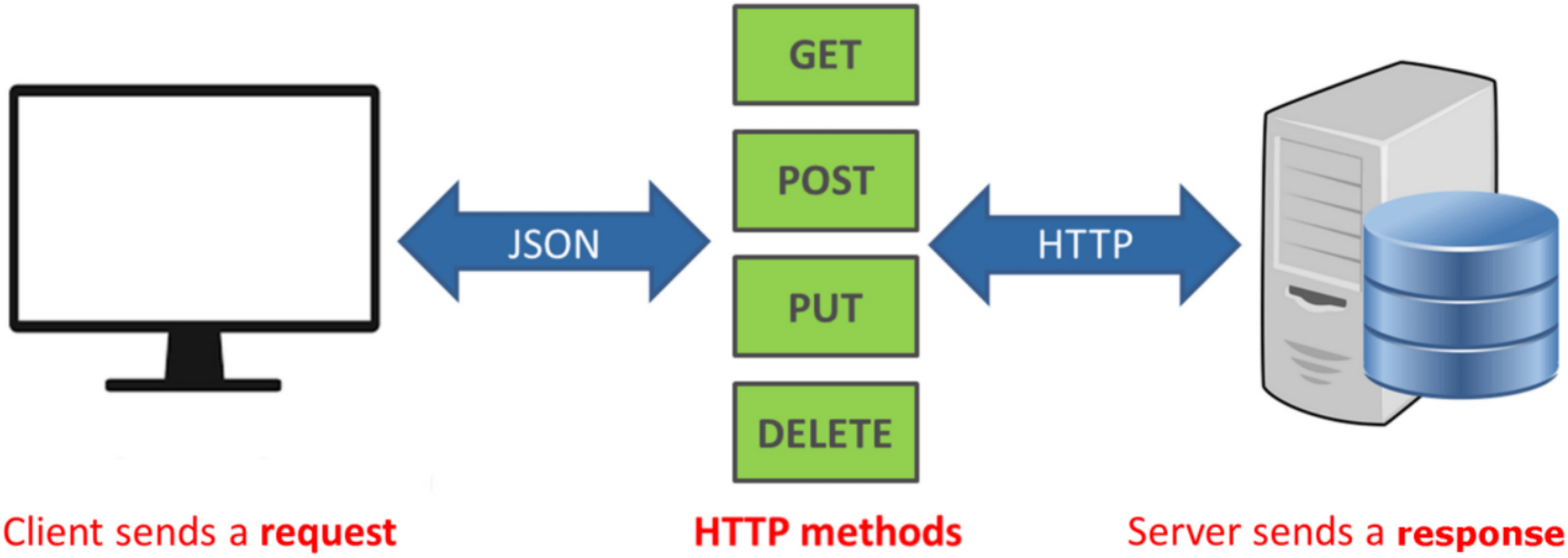
Integra las aplicaciones de los usuarios con Cisco Webex y proporciona a los desarrolladores herramientas para transformar las experiencias de colaboración.



/devices	/places	/xapi
GET List Devices	GET List Places	GET Query Status
POST Activation code	POST Create a Place	POST Execute Command
GET Get details	GET Get Place details	
	PUT Update a Place	
DELETE Delete a Device	DELETE Delete a Place	

https://developer.webex.com/docs/platform-introduction
https://developer.webex.com/docs/api/changelog

Basics. API rest



Basics. Peticiones HTTP - Métodos

- **GET** – solicitar información
- **PUT** – actualizar contenido
- **POST** – crear contenido
- **DELETE** – eliminar información

- W3C RFC 7230-7 para listado completo métodos HTTP
  - Head, Patch, Options….

Basics. Webex APIs. Respuestas

- **1xx – Information**

  100 continue
- **2xx – Sucess**

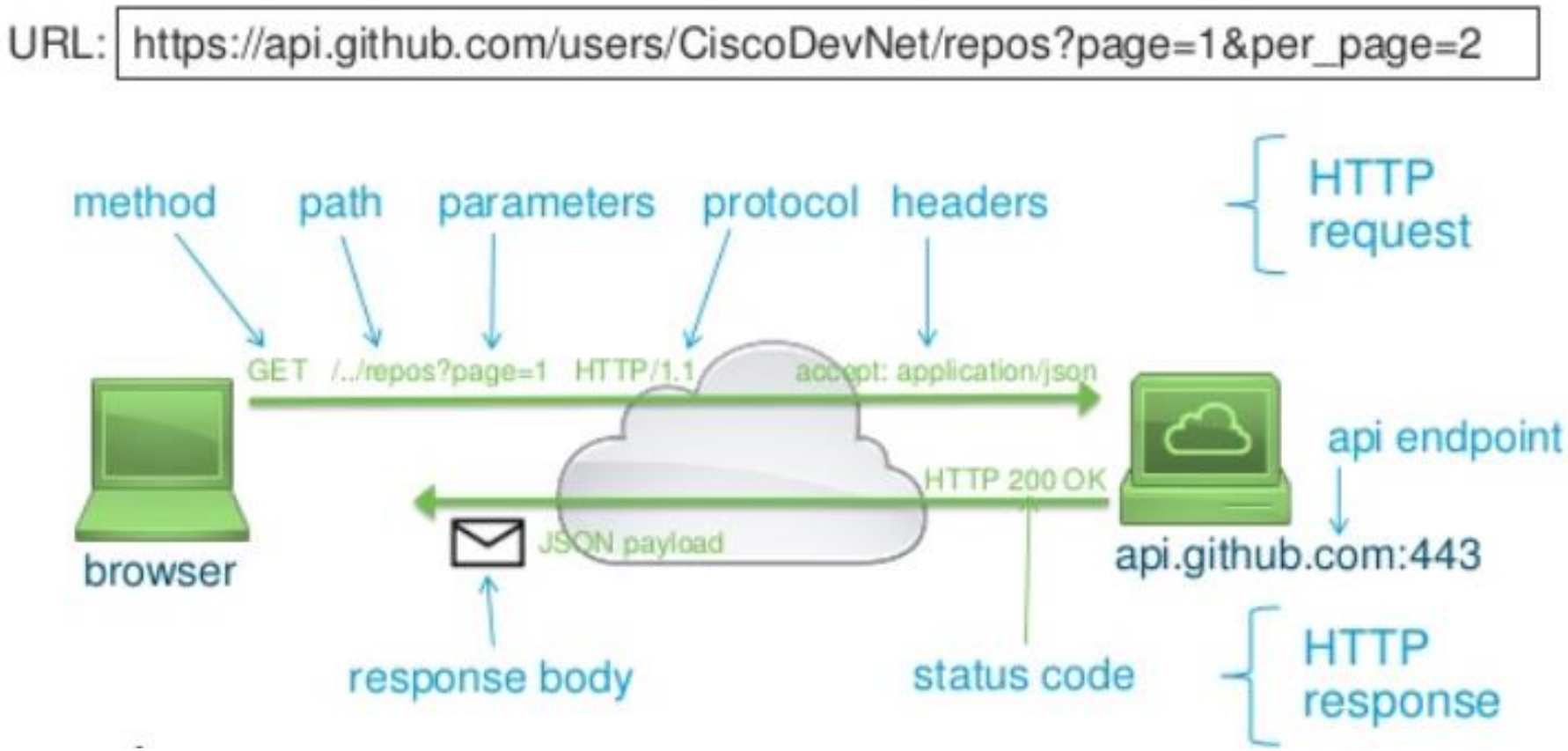
  200 OK, 201 Created, 204 No Content
- **3xx – Redirection**

  301 Moved permanently , 304 Not Modified
- **4xx – Client error**

  400 Bad Request, 401 unauthorized, 403 Forbidden, 404 Not Found
- **5xx – Server error**

  500 Internal Error, 501 Not Implemented, 503 Service Unavailable

Basics. Ejemplo petición API Rest



https://api.github.com/users/CiscoDevNet/repos?page=1&per_page=2

Basics. Webhooks

- Una API es una forma de comunicar datos entre aplicaciones separadas. Pero, ¿qué pasa si quieres que te notifiquen cambios en determinado evento o elemento en otra aplicación? En este caso, una API sería extremadamente ineficiente. En su lugar, necesitas utilizar un **Webhook**.

- Un **Webhook** es una manera de ser notificado cuando un evento ha ocurrido en tu aplicación o en la de un tercero. Es básicamente una solicitud POST que se envía a una URL específica. Esa URL está configurada para recibir el cuerpo de la solicitud POST y procesarla de alguna manera.

- Una API se utiliza para hacer preguntas directas y un **Webhook** se utiliza para notificar cuando se producen ciertos eventos. En lugar de preguntar constantemente si algo ha cambiado, un **Webhook** puede activarse y notificarnos automáticamente cuando se produzca el evento.

Basics. JSON

- **JSON** es el acrónimo para *JavaScript Object Notation*, y aunque su nombre lo diga, no es necesariamente parte de JavaScript, de hecho es un estándar basado en texto plano para el intercambio de información

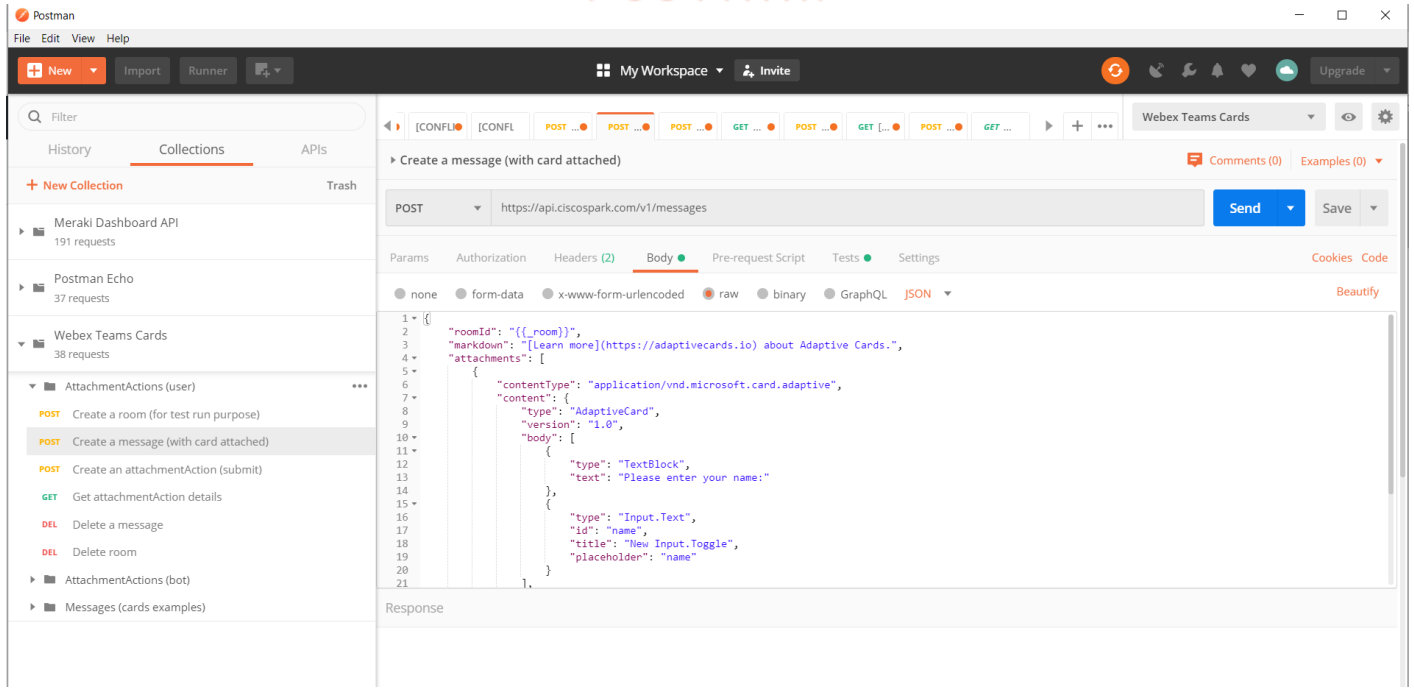
- La ventaja de **JSON** al ser un formato que es independiente de cualquier lenguaje de programación, es que los servicios que comparten información por éste método, no necesitan hablar el mismo idioma, es decir, el emisor puede ser Java y el receptor PHP, cada lenguaje tiene su propia librería para codificar y decodificar cadenas de **JSON**.

- Nace como alternativa a XML

```
{
  "users": [
      {user-object},
      {user-object},
      {user-object}
  ],
  "next_cursor": 1489467234237774933,
  "next_cursor_str": "1489467234237774933",
  "previous_cursor": 0,
  "previous_cursor_str": "0"
}
```

Basics. Entornos de prueba APIs

POSTMAN



https://www.postman.com/
https://github.com/CiscoDevNet/postman-webex

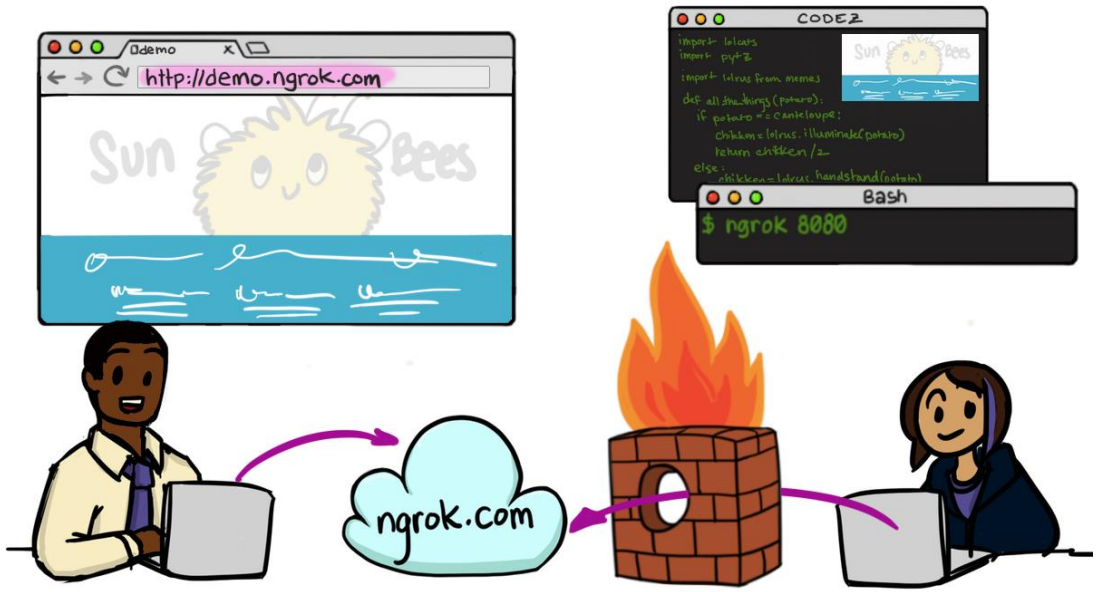
curl://

```
Aniqas-MacBook-Pro:decathlon-microservices aniqarehman$ curl --header "X-Cisco-Meraki-API-Key: ef00d1c5b0741230d05e45e533c05e87841fda85" https://n146.meraki.com/api/v0/organizations
[{"id":841342,"name":"Camara"},{"id":853785,"name":"Makenai"}]Aniqas-MacBook-Pro
```

https://github.com/curl/curl

Basics. ¿Cómo probar localmente nuestro bot?

ngrok

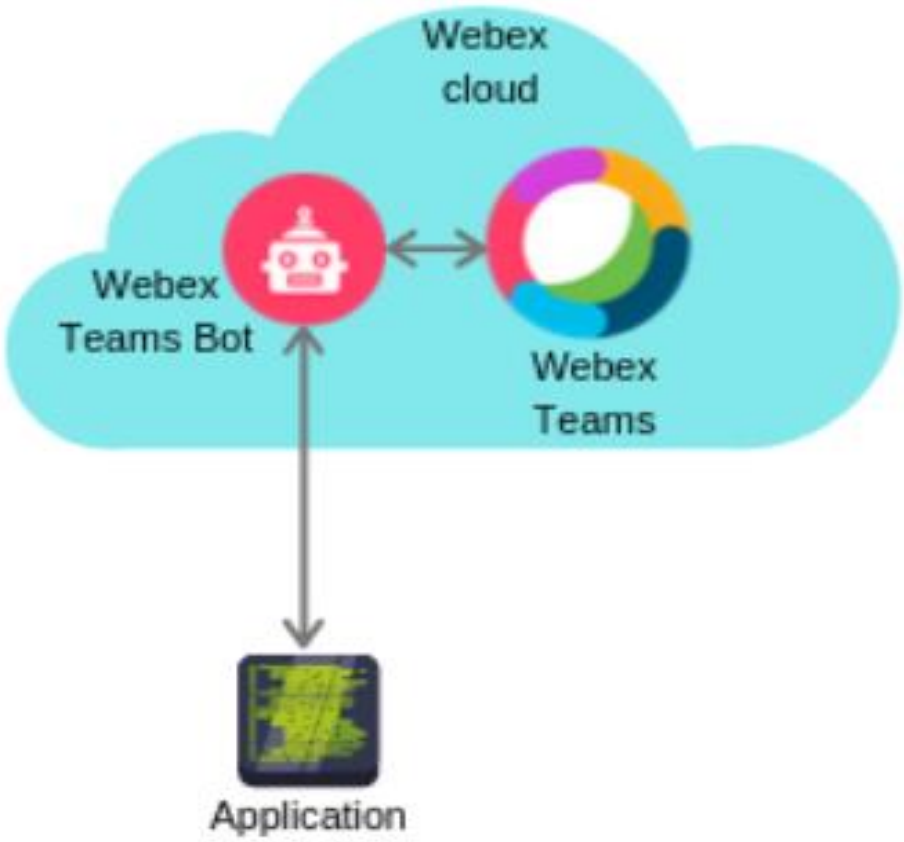


Con **NGROK** podemos abrir nuestro servidor local a internet

- **NGROK** es una herramienta que permite exponer nuestro servidor local a internet sin hacer ninguna configuración específica en el router o firewall
- Uso sencillo de esta funcionalidad (instalación de un pequeño software y ejecución de un comando)
- Muy útil para realización de demos o de pruebas de nuestro código o cuando trabajamos con un equipo de desarrolladores de forma remota

2. ¿Qué es un bot?

Conceptos Bots



Proporciona a los usuarios de Webex Teams acceso a servicios externos directamente desde sus espacios de Webex Teams. Los bots ayudan a los usuarios a automatizar tareas, incorporar contenido externo a los espacios y ganar eficiencia.

Conceptos Bots

- Los bots se comportan de forma similar a los usuarios normales de Webex Teams
- Pueden participar en espacios 1:1 o grupales y los usuarios pueden intercambiarse mensajes directamente con los bots o añadirlos a un espacio grupal. Una acreditación especial es añadida al avatar del bot para que lo usuarios sepan que están interactuando con un bot en lugar de con un humano.
- Un bot puede solo acceder a los mensajes enviados directamente a él. En los espacios de grupo los bots pueden ser @mencionados para acceder al mensaje. En espacios 1:1 el bot accede a todos los mensajes.
- Los bots no realizan acciones dentro de Webex Teams de parte de un usuario de Webex Teams. Si estás creando una aplicación que necesita participar en Webex Teams y realizar acciones con la cuenta de un usuario, la solución es realizar Integraciones.

Tipos de Bots

Los bots se pueden utilizar para multitud de funcionalidades, en diferentes formas y tamaños. Algunas ideas de aplicación:

- **Notifiers**: por ejemplo recepción de actualizaciones en CRMs o cambios en GitHub en cliente Webex Teams
- **Controllers**: por ejemplo actualizaciones desde cliente Webex Teams en JIRA
- **Assistants**: asistentes virtuales

https://cloud.google.com/natural-language/

Creando un bot en Cisco Webex

Pasos para crear un bot

1. Acceder a https://developer.webex.com/my-apps
2. Seleccionar "Create a New App" y después "Create a Bot" para iniciar el wizard.
3. Introducir información básica y descriptiva sobre el bot:
   - Nombre del bot
   - Username
   - Icono
4. Seleccionar “Añadir bot”. Si todo correcto dispondrás de un access token para el nuevo bot, que servirá para autenticar tu bot con el Webex REST API.
5. El access token solo aparece una vez, con posibilidad de regenerarlo siempre que sea necesario

Frameworks & Tools

Hay varios frameworks que simplifican el proceso de Desarrollo, a bajo nivel del bot.

- Botkit es un popular open source bot framework con soporte conversacional avanzado así como integraciones con Proveedores de mecanismos de procesamiento de lenguaje natural y almacenamiento.
- Cisco Devnet Botkit
- El programa Cisco Webex Teams Ambassador tiene links a otros open source bot starter kits.

https://github.com/CiscoDevNet/botkit-template?files=1

3. Cards/Buttons

Overview

- **Cards/Buttons** - Mejora interactividad con los mensajes de Webex Teams
- Nuevas opciones enriquecidas para que los usuarios puedan interactuar con sus aplicaciones sin salir del cliente Webex Teams
- Bots e Integrationes pueden añadir cards a espacios de Webex Teams incluyendo una card adjunta cuando se envía un mensaje.
- Uso de la especificación **Adaptive Cards** de Microsoft para definir el contenido de la tarjeta.
- **Soporte multiplataforma**. Los diferentes clientes de Webex Teams ofrecerán el mismo aspecto de las tarjetas en cada plataforma, permitiendo poner el foco en el contenido y en la interacción sin preocuparte de la presentación.

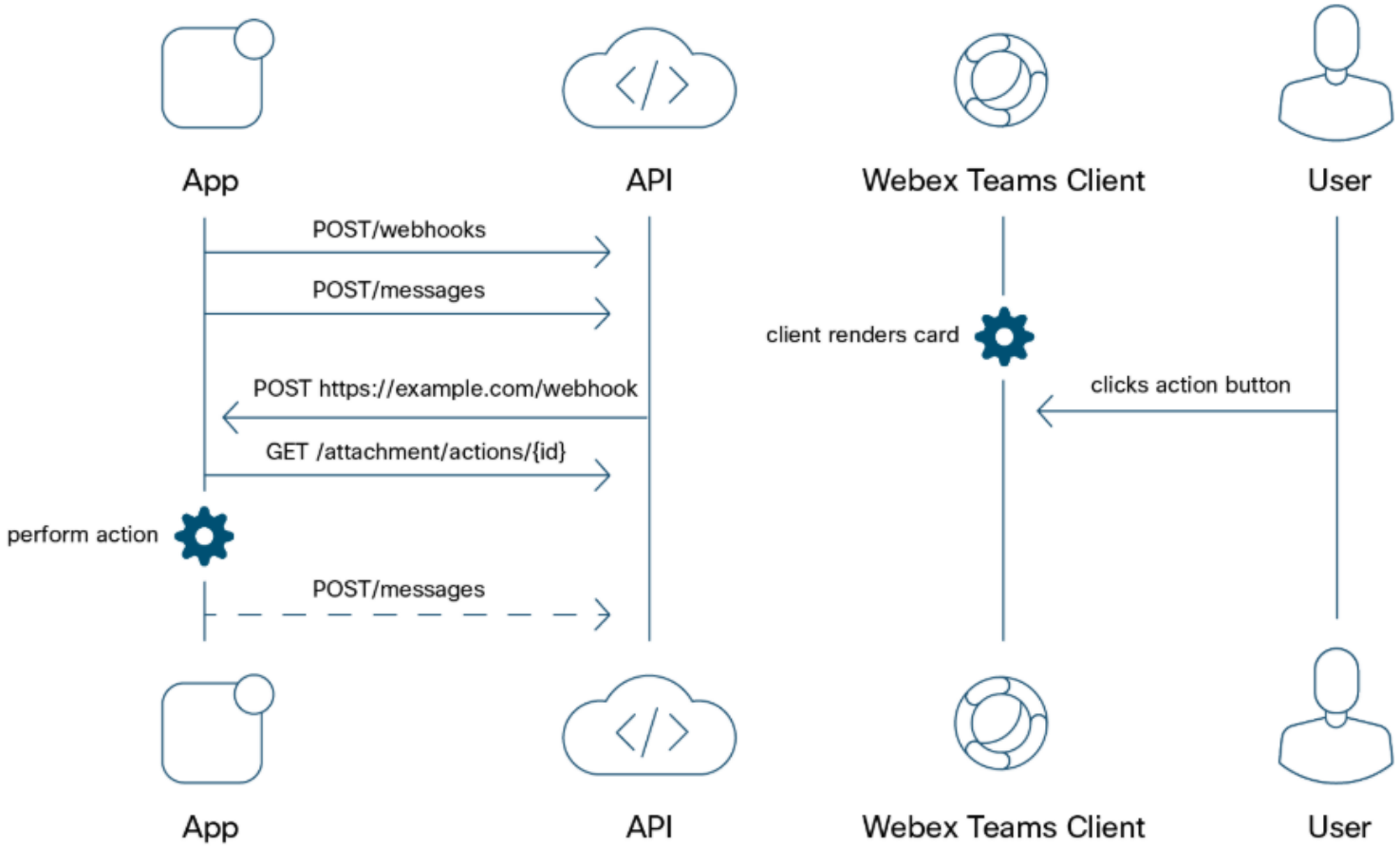
Descripción

- Las Cards pueden ser interactivas, solicitando una acción desde el usuario, como responder a una encuesta, o pueden ser puramente informativos, como mostrar las actuales condiciones atmosféricas.

- Para hacer que las cards sean interactivas, se incluyen botones. Los botones son acciones que los usuarios pueden usar para abrir una URL, mostrar otra tarjeta, o enviar datos. Cards pueden incluir hasta cinco botones (acciones). Si incluyes un formulario, los datos introducidos por el usuario serán encriptados y almacenados dentro de la plataforma Webex. Puedes usar webhooks para recibir notificaciones para generar nuevos formularios Al igual que otros recursos API que contienen datos encriptados, el cuerpo del webhook no incluye los datos de formulario enviados. Después de que recibes el webhook, necesitarás recuperar los datos de formulario mediante el endpoint **Attachment Actions API**.

Descripción

Cómo usar webhooks, mensajes y attachment actions para crear tarjetas interactivas



Descripción

- Cards pueden ser usadas en espacios 1:1 o en espacios grupales. Cuando son enviadas a un espacio grupal, cada usuario en el espacio puede interactuar con la card.

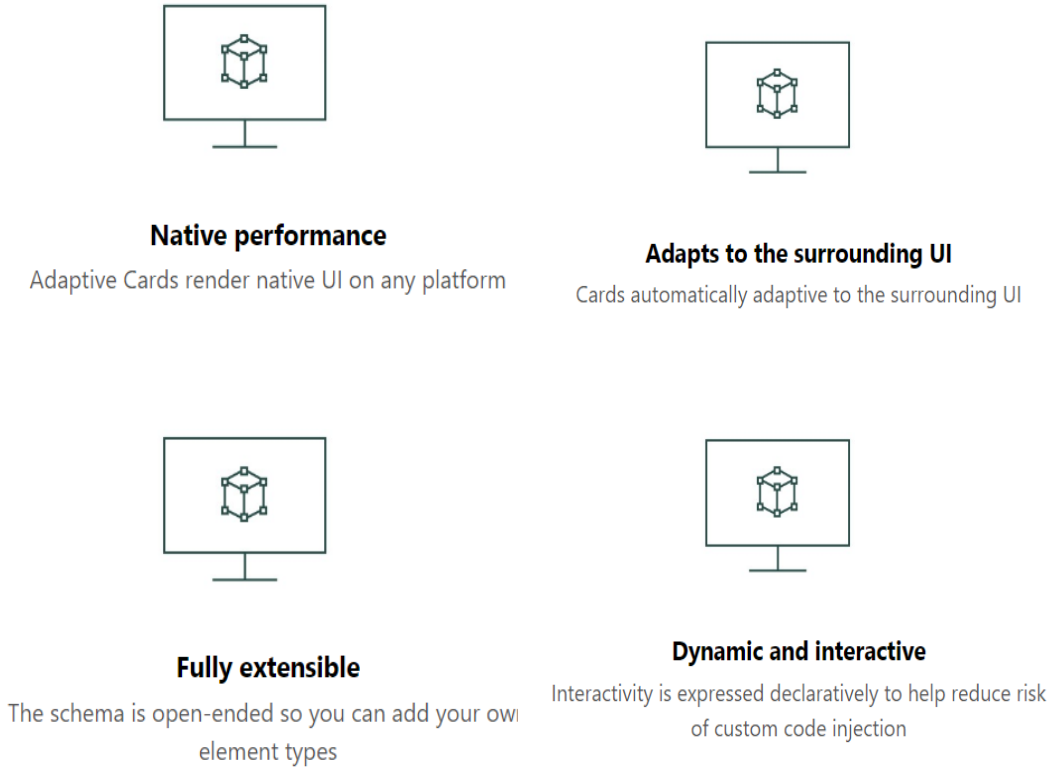
- El contenido de la tarjeta no puede ser cambiado después de ser creado. Si necesitas proporcionar feedback a un usuario después de la interacción con una tarjeta, tienes varias opciones:
  - Borra la tarjeta y enviar un mensaje con el resultado de la acción
  - Si la tarjeta está en un grupo donde varias personas interactua con ella, puedes enviar un mensaje 1:1 después de la interacción

- Para eliminar la tarjeta, simplemente se borra el mensaje que contiene la card adjunta. La tarjeta será eliminada del espacio.

Descripción

Las **tarjetas adaptables** son fragmentos de interfaz de usuario independientes de la plataforma, creados en JSON, que las aplicaciones y los servicios pueden intercambiar abiertamente. Cuando se entrega a una aplicación específica, el JSON se transforma en una interfaz de usuario nativa que se adapta automáticamente a su entorno. Ayuda a diseñar e integrar una interfaz de usuario ligera en las principales plataformas y frameworks

https://docs.microsoft.com/es-es/adaptive-cards/
https://adaptivecards.io/

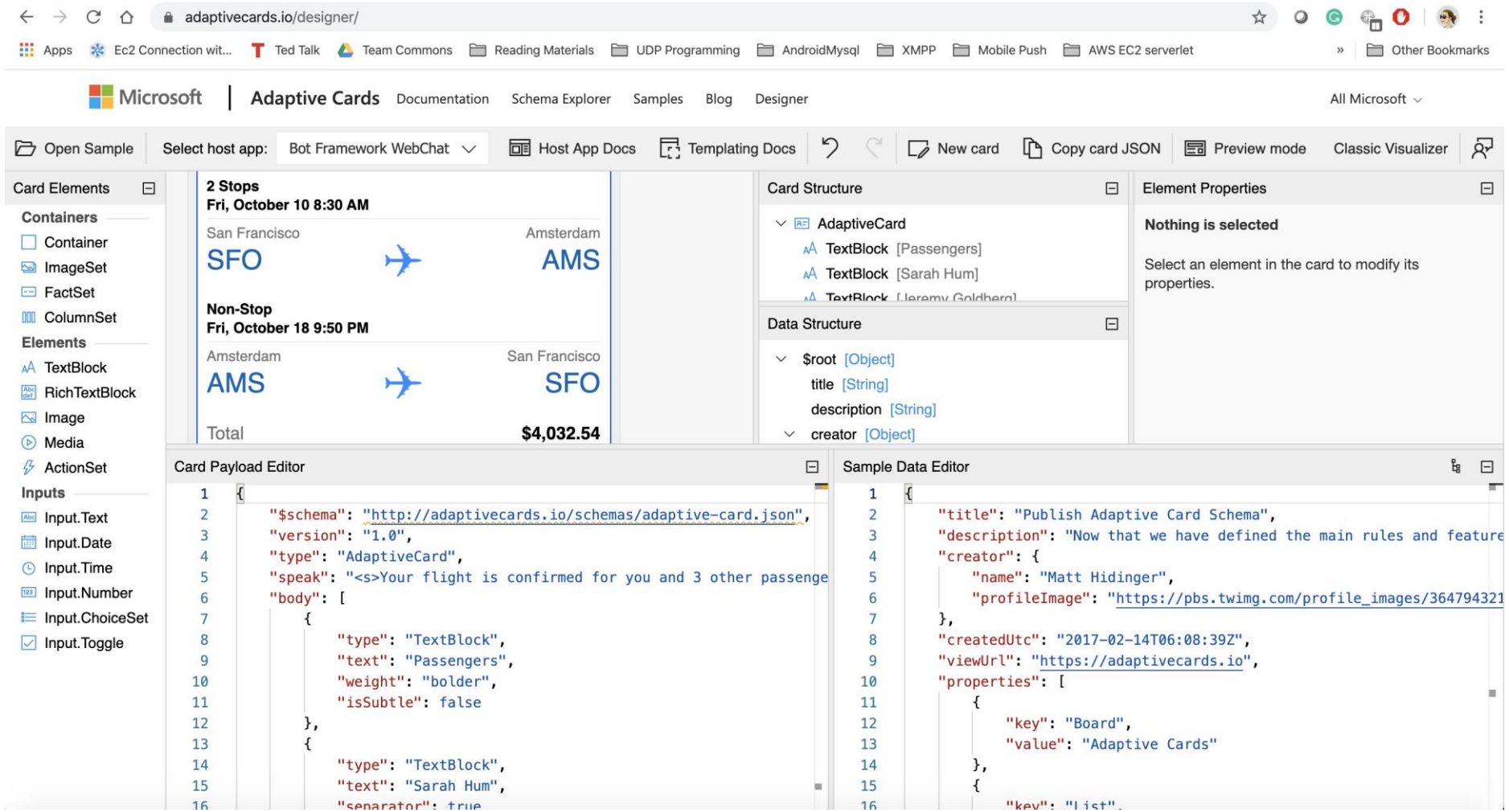


Microsoft Adaptative Cards

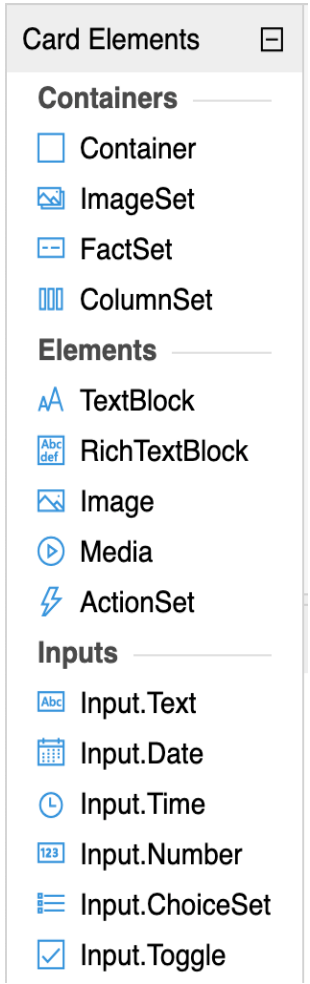
Herramientas de ayuda para el creación de cards:

- Schema explorer
  https://adaptivecards.io/explorer/
- Interactive card designer
  https://adaptivecards.io/designer/

Interactive card designer



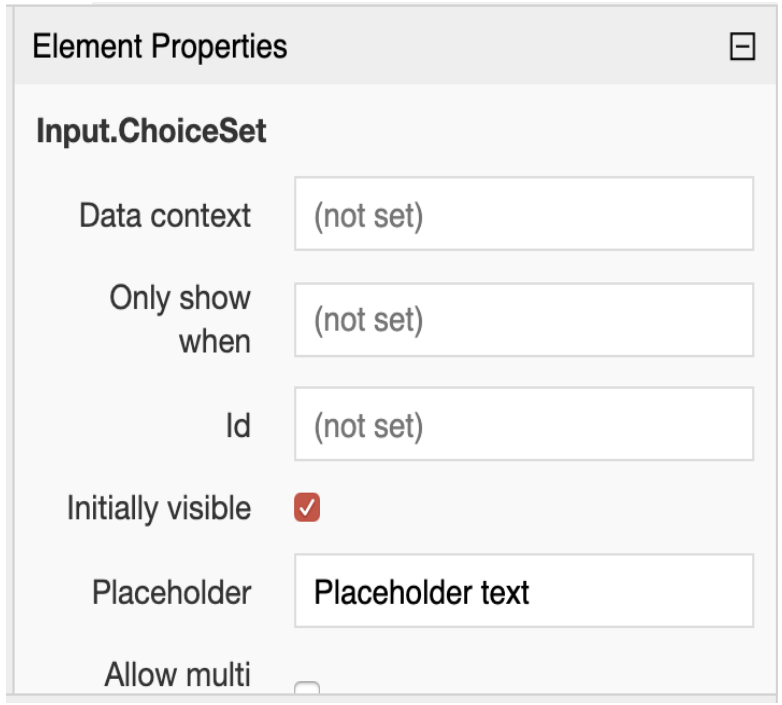
Interactive card designer



Cards elements

Estos son todos los elementos que estás disponibles en **Microsoft Adaptive card designer.** Media es el único que en estos momentos no está soportado en el entorno de Cisco Teams. **Inputs** son usados cuando es requerido recoger alguna información del usuario. **Containers** se usan para estructurar las tarjetas y el resto de elementos se incorporan al contenedor. **Elements block** contienen elementos para mostrar información con bloques de texto o imagen y conjuntos de acciones. (botones para acciones, para abrir otra tarjeta o enviar información)

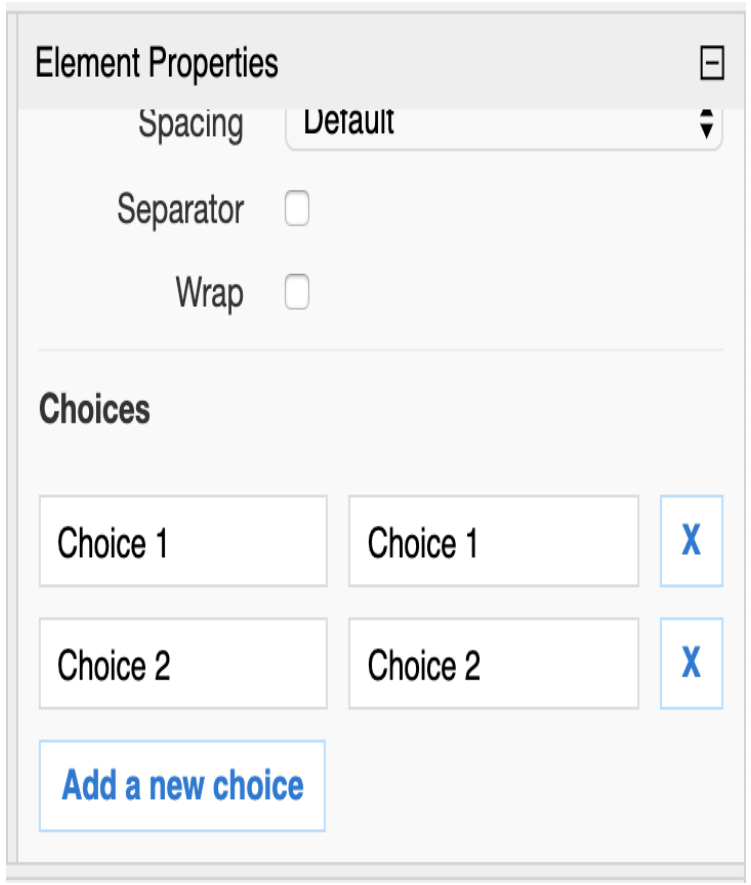
Interactive card designer



Element properties

El bloque más a la derecha contiene las propiedades del element específico seleccionado. En este ejemplo choice set es seleccionado donde podemos definer un ID el cual será usado por la acción para enviar información para ser procesada por el backend usando webhook.

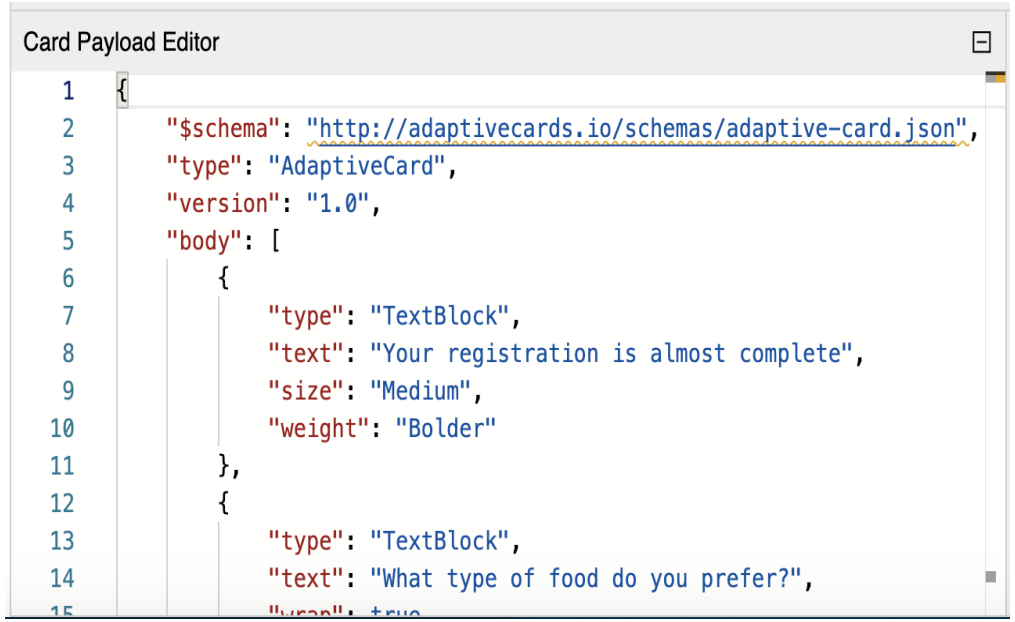
Interactive card designer



Element properties

Otras propiedades que tiene el elemento Choice Set son las elecciones reales con su nombre (title) y valor. La parte del valor es la parte que se almacena en backend y la parte nombre (title) es la que se muestra al usuario.

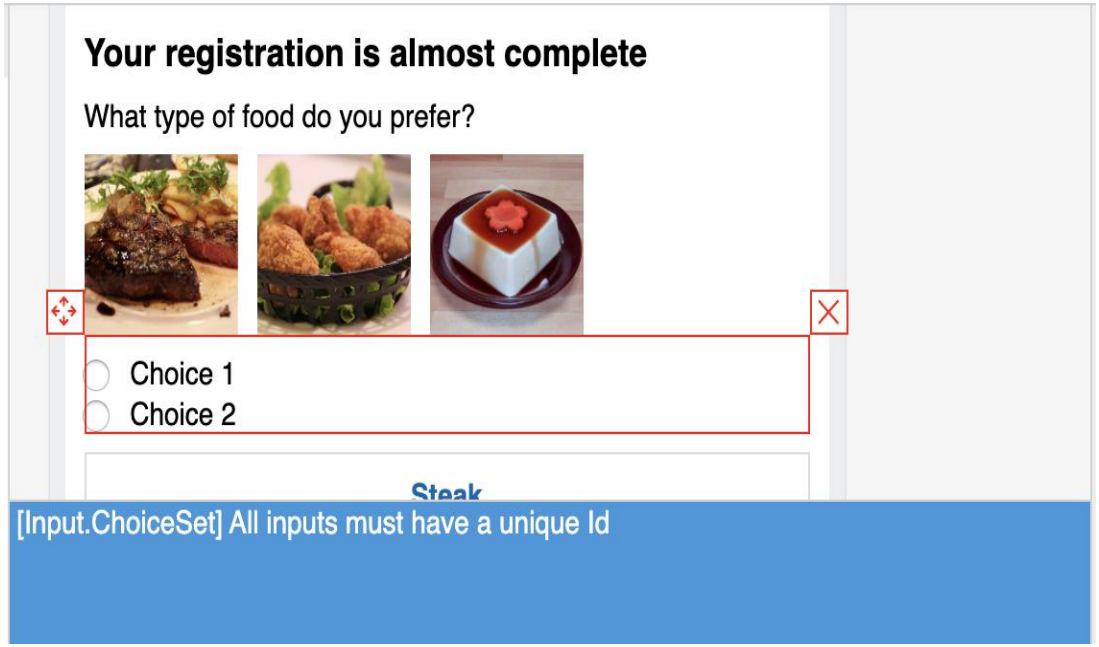
Interactive card designer



Payload editor

En la parte inferior izquierda encontramos el payload editor. Tenemos dos opciones para trabajar con la tarjeta. Una forma interactiva, donde se realiza drag and drop de los elementos y después completar las propiedades para cada uno de ellos. La segunda forma es este editor donde escribir el JSON adecuado y esto mostrará un preview de la tarjeta si no hay errores en el JSON. Este JSON es usado entonces para hacer un post Message con la API de Cisco Webex para enviar la tarjeta.

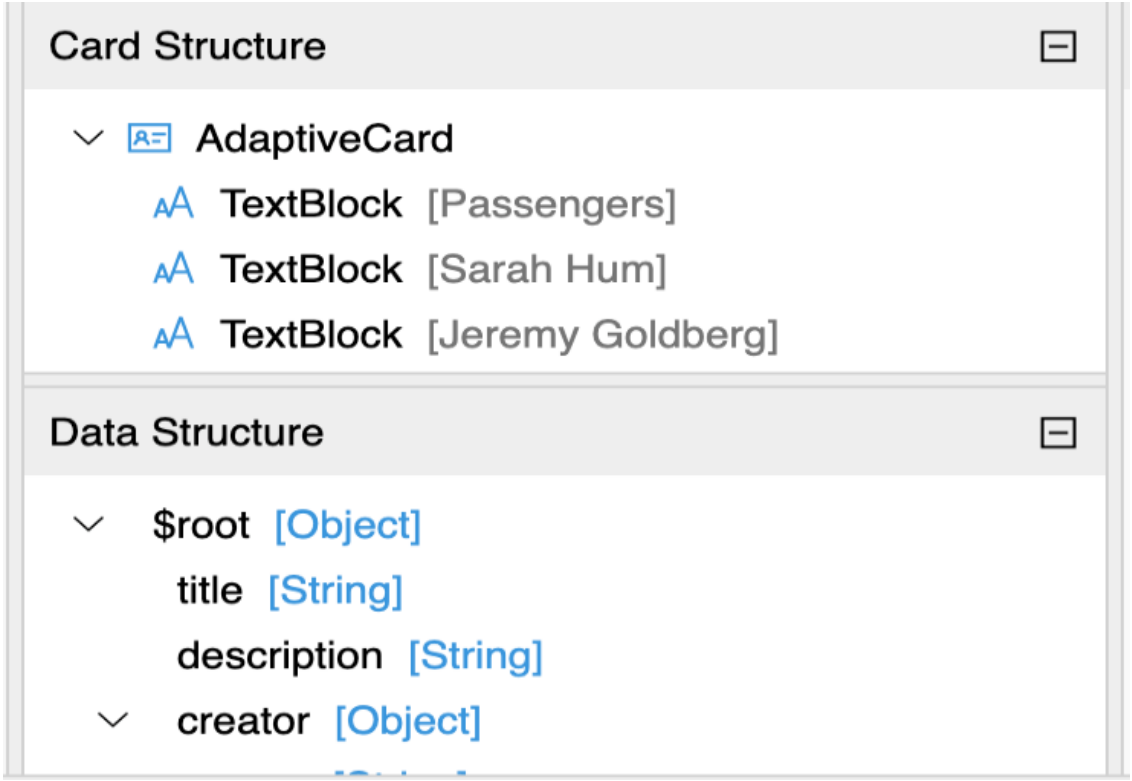
Interactive card designer



Card Preview

En la parte superior izquierda, despues de los elementos, Podemos ver el aspecto visual que tendrá la tarjeta cuando sea enviada. Ya que las tarjetas en Microsoft son un poco diferentes que en Cisco Webex, el aspecto puede que sea un poco diferente, aunque el JSON es el mismo.
También recibimos un mensaje si alguna propiedad es obligatoria para ciertos elementos.

Interactive card designer



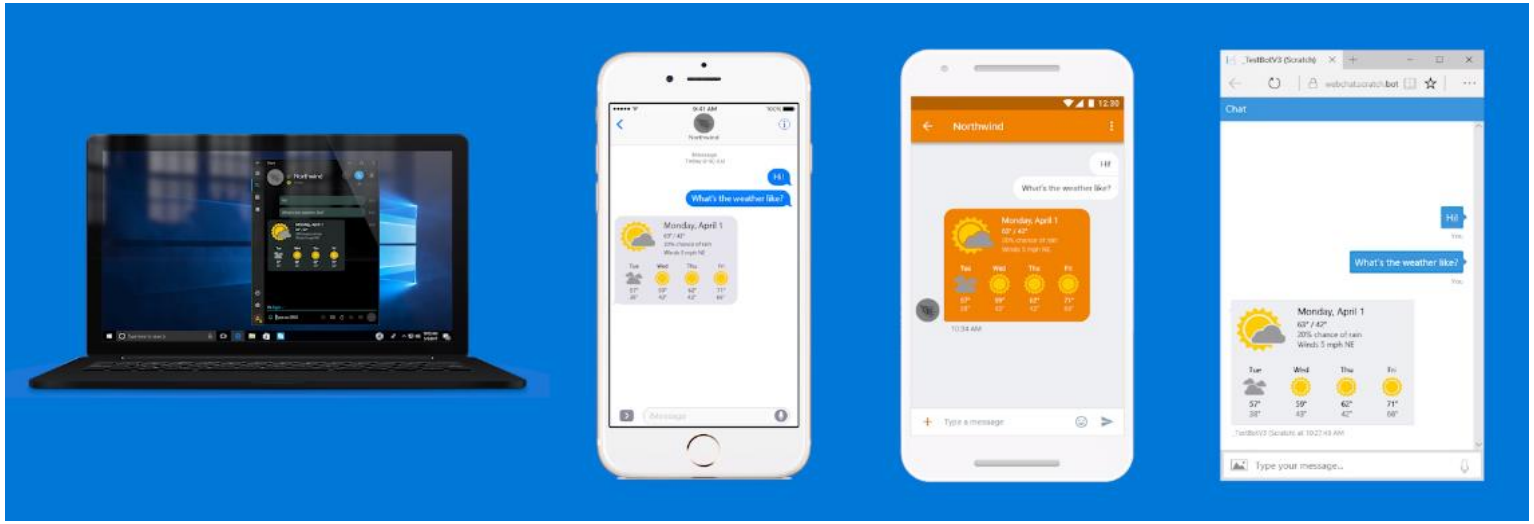
Card Structure

Además del aspecto visual de la tarjeta, podemos ver la estructura de la misma y podemos rellenar las propiedades seleccionando el elemento específico aquí.

Interactive card designer

Soporte en las siguientes plataformas:

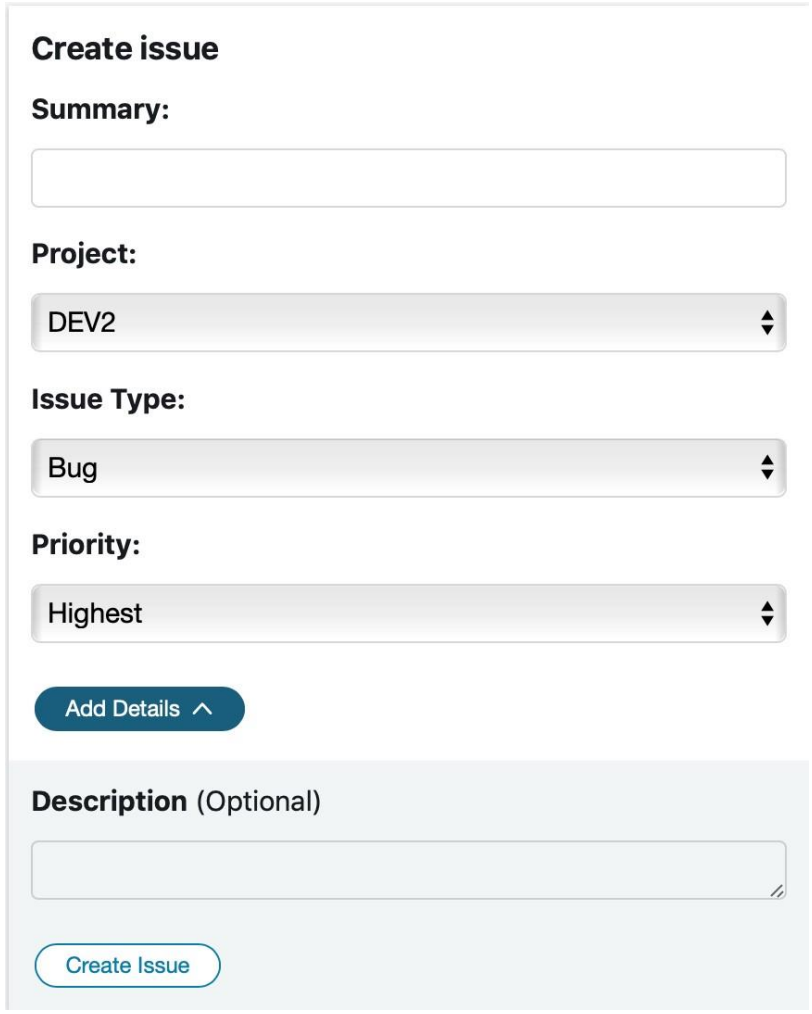
- Desktop Clients: Windows and Mac
- Mobile Clients: Android and iOS
- Web Client
- Browser SDK



Limitaciones actuales

- Webex Teams soporta Adaptive Cards 1.1.
- Todos los card elements están disponibles para uso salvo **Media**.
- Todas las card properties está disponibles para uso salvo **backgroundImage**, **iconUrl**, y **speak**.
- Cards pueden incluir hasta 5 acciones.
- Mensajes con cards pueden incluir hasta 10 imágenes.
- Solo una card puede ser enviada por petición.

Ejemplo de Bot en Depot - JIRA



https://help.webex.com/en-us/w49a6e/Jira-Cloud-Bot-for-Webex-Teams#id_122521

4. Cómo se crea un bot – LatinCUG

Bot LatinCUG. Pasos realizados

- Crear Bot en Developer Webex
- Guardar el Access Token
- Crear un proyecto desde una plantilla (https://github.com/CiscoDevNet/botkit-template). Uso de herramientas de edición de código como Visual Studio Code
- Probar el bot:
  - Arrancar ngrok en el puerto PORT 3000 (./ngrok http 3000)
  - Actualizar fichero .env de la plantilla con:
    - WEBEX_ACCESS_TOKEN = bot access token
    - PUBLIC_URL = https url from ngrok
  - Arrancar el bot (node bot.js)
- Realizar el despliegue en servidor de producción (en este caso en Google Cloud)
- Si se desea incluirlo en Webex App Hub, realizar trámites necesarios para recibir validación de código por parte de Cisco (https://apphub.webex.com/)

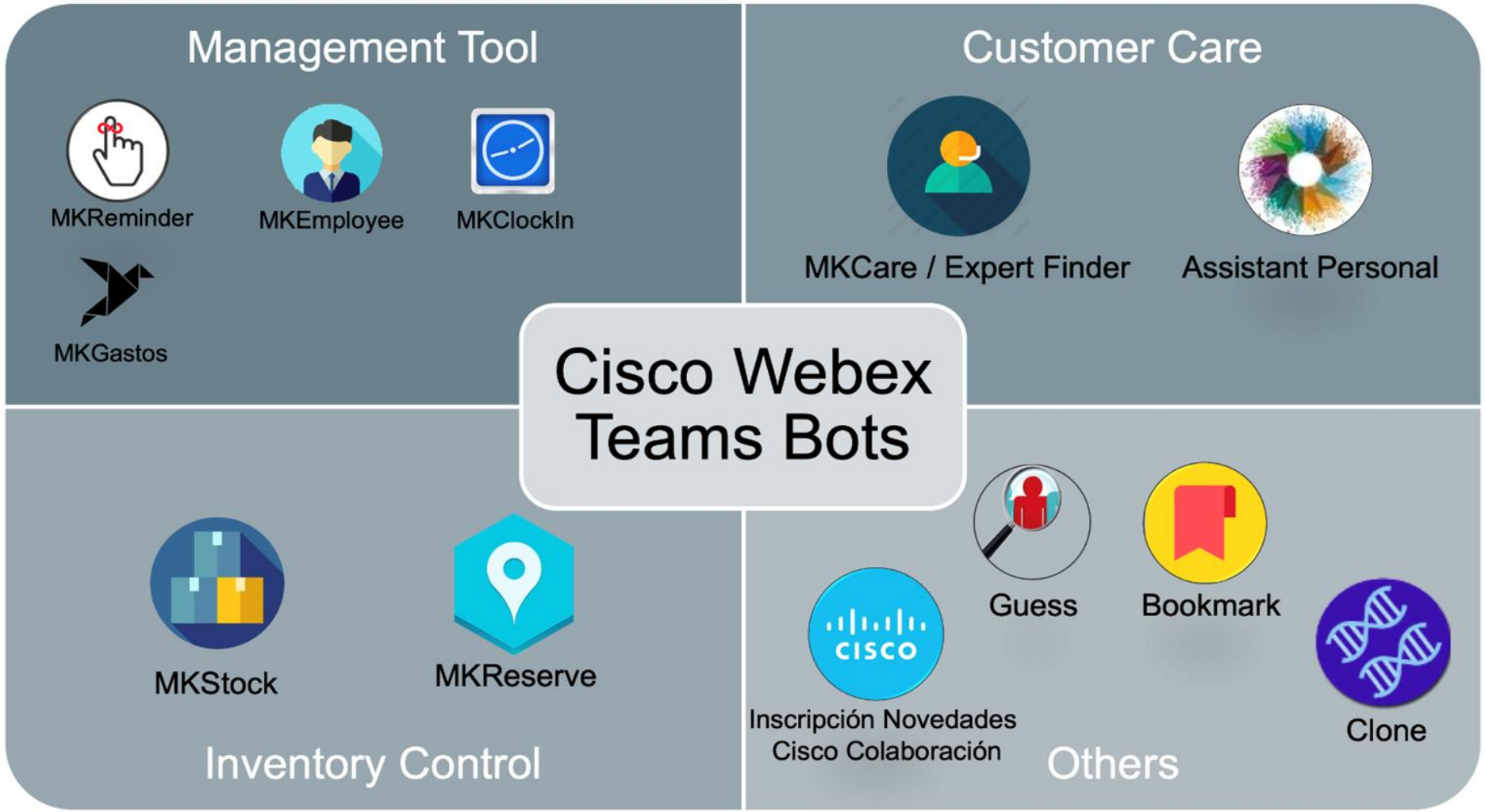
Bot LatinCUG. Skills

Opciones disponibles accesibles habitualmente con comando help o ayuda en el bot:

- **resultpollcountry** - para ver los estadísticas de la encuesta sobre país de origen
- **send** - para enviar mensaje a todos los miembros del grupo 'LatinCUG - Webex y Teams LATAM', desde espacio 1:1, disponible solo para los administradores
- **countdomain** para ver los estadísticas de domain con miembros mas de dos personas
- **comment** - para enviar un comentario a los administradores del grupo 'LatinCUG - Webex y Teams LATAM', desde espacio 1:1
- **listarcomentarios** - para listar los comentarios, disponible solo para los administradores

5. Ejemplos de bots

Casos de uso


Management Tool
MKReminder
MKEmployee
MKClockIn
MKGastos
Customer Care
MKCare / Expert Finder
Assistant Personal
Cisco Webex Teams Bots
MKStock
MKReserve
Inventory Control
Guess
Bookmark
Inscripción Novedades Cisco Colaboración
Clone
Others

Makenai

We are ready!

info@makenai.es

+34 91 193 95 21
+34 93 199 12 00

www.makenai.es

Febrero 2020