



UT.5 Comprensión del lenguaje natural.

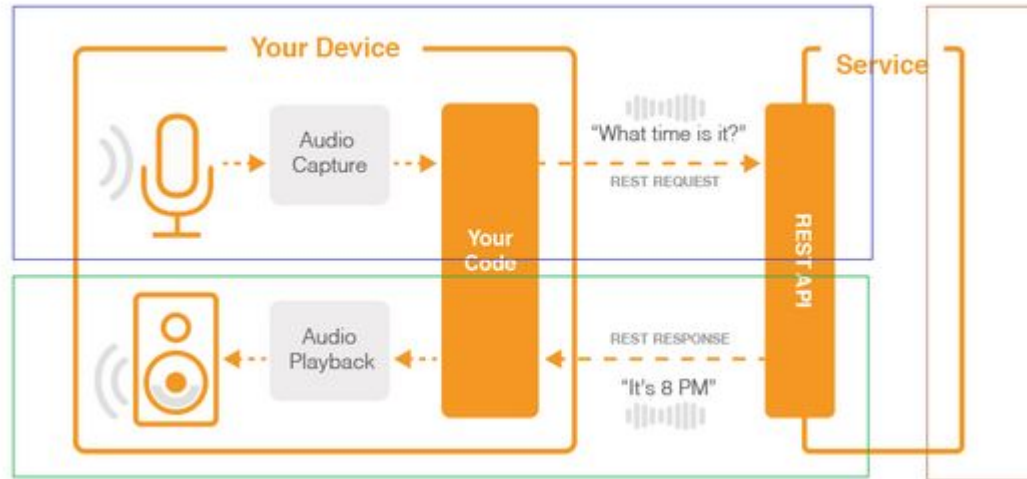
Bloques de la UT5:



1. Comprensión del lenguaje natural.
2. Introducción a AWS Lex v2.

1. Comprensión del lenguaje natural.

Recordatorio de arquitectura



1. Comprensión del lenguaje natural.



La comprensión del lenguaje natural es una rama de la inteligencia artificial que utiliza software para comprender la entrada en forma de oraciones usando texto o habla.

NLU (Natural Language Understanding) permite la interacción persona-computadora. Es la comprensión del lenguaje humano como el inglés, el español y el francés, por ejemplo, lo que permite a los móviles u ordenadores comprender comandos sin sintaxis de lenguajes de programación.

1. Comprensión del lenguaje natural.



En este tema y el siguiente veremos cómo realizar una comprensión de textos y a partir de esta poder realizar una respuesta óptima.

1. Comprensión del lenguaje natural.



Gestión de retardo: Añadimos una nueva capa de computación, por lo que tenemos una nueva variable de retardo:

- Internet usuario.
- Ubicación usuario.
- Computación TTS.
- Computación ASR.
- Computación NLU.

Recordar que uno de los mayores retos es que tiene que ser en **tiempo real** la comunicación con el humano.

1. Comprensión del lenguaje natural.



NLU analiza los datos para determinar su significado utilizando algoritmos para reducir el habla humana a una ontología estructurada — a un modelo de datos que consiste en semántica y definiciones pragmáticas. Dos conceptos fundamentales de NLU son la **intención** y el **reconocimiento de entidades**.

1. Comprensión del lenguaje natural.



El reconocimiento de **intenciones** es el proceso de identificar el sentimiento del usuario en el texto de entrada y determinar su objetivo. Es la primera y más importante parte de NLU porque establece el significado del texto.

El reconocimiento de **entidades** es un tipo específico de NLU que se enfoca en identificar las entidades en un mensaje y luego extraer la información más importante sobre esas entidades. Hay dos tipos de entidades: entidades nombradas y entidades numéricas. Las entidades nombradas se agrupan en categorías, como personas, empresas y ubicaciones. Las entidades numéricas se reconocen como números, monedas y porcentajes.

1. Comprensión del lenguaje natural.



Aplicaciones IVR y enrutamiento de mensajes:

- Respuesta de Voz Interactiva (IVR) se utiliza para autoservicio y enrutamiento de llamadas.
- Las primeras iteraciones fueron basadas en sistemas expertos.
- Hoy en día predomina los modelos cognitivos. El sistema procesa la voz del usuario, convierte las palabras en texto y luego analiza la estructura gramatical de la oración para determinar la probable intención de la persona que llama.

1. Comprensión del lenguaje natural.



Aplicaciones de atención al cliente y servicio a través de asistentes personales inteligentes:

- NLU es la tecnología detrás de los chatbots.
- Tiene distintas capas de diferentes procesos en la tecnología NLU, como extracción de características y clasificación, vinculación de entidades y conocimiento administrativo.

1. Comprensión del lenguaje natural.



En temas anteriores hemos analizado cómo funcionan los IVR telefónicos y los asistentes virtuales.

En ambos casos se obtienen grandes ventajas sobre modelos tradicionales:

- Utilización 7x24 con tiempos de respuesta inmediatos.
- Tiempos de resolución muy bajos en casos cotidianos.
- Gestión de RRHH.
- Automatización de procesos.

2. Introducción a AWS Lex v2.



Amazon Lex V2 es un servicio de AWS para crear interfaces de conversación para aplicaciones que usen voz y texto.

Proporciona la amplia funcionalidad y flexibilidad de la comprensión del lenguaje natural (NLU) y el reconocimiento automático de voz (ASR).

La comprensión del lenguaje natural y el reconocimiento de voz son algunos de los problemas principales que deben resolver las ciencias computacionales, lo que requiere entrenar sofisticados algoritmos de aprendizaje profundo con enormes cantidades de datos e infraestructura.

2. Introducción a AWS Lex v2. Bot.



Un **bot** realiza tareas automatizadas, como pedir una pizza, reservar un hotel, encargar flores, etc. Un bot Amazon Lex V2 funciona con capacidades de reconocimiento automático de voz (ASR) y comprensión del lenguaje natural (NLU).

Los bots de Amazon Lex V2 pueden comprender la entrada del usuario proporcionada con voz o texto y conversar lenguaje natural.

2. Introducción a AWS Lex v2. Idioma.

Un **bot** de Amazon Lex V2 puede conversar en uno o más idiomas. Cada **idioma** es independiente de los demás, puede configurar Amazon Lex V2 para conversar con un usuario utilizando palabras y frases nativas.

es_419	Español (Latinoamérica)
es_es	Español (España)
es_US	Español (EE. UU.)

Ca_es	Catalán (España)
-------	------------------

2. Introducción a AWS Lex v2. Intención.



Una **intención** representa una **acción** que el usuario desea realizar. Puede crear un **bot** que admita una o más intenciones relacionadas.

Existe un máximo de 100 intenciones por bot.

Por ejemplo, puede crear una intención que pida pizzas y bebidas. Para cada intención, debe proporcionar la siguiente información obligatoria:

2. Introducción a AWS Lex v2. Intención.



- **Nombre de la intención:** nombre descriptivo de la intención. Por ejemplo, OrderPizza.
- **Enunciados de muestra**— Cómo podría comunicar la intención un usuario. Por ejemplo, un usuario podría decir «¿Puedo pedir una pizza?» o «Deseo pedir una pizza».
- **Cómo cumplir la intención**— Cómo desea llevar a cabo la intención después de que el usuario proporcione la información necesaria. Recomendamos que cree una función Lambda para el cumplimiento de la intención.

2. Introducción a AWS Lex v2. Slots.



Una intención puede requerir cero o más **slots** o parámetros. Puede añadir parámetros como parte de la configuración. Durante el tiempo de ejecución, Amazon Lex V2 solicita al usuario determinados valores de slot.

El usuario debe proporcionar valores para todos los slots necesarios antes de que Amazon Lex V2 lleve a cabo la intención.

2. Introducción a AWS Lex v2. Slots.



Por ejemplo, la OrderPizza La intención requiere slots como el tamaño, el tipo de masa y el número de pizzas. Para cada slot, hay que proporcionar el tipo de slot y uno o varios mensajes que Amazon Lex V2 envía al cliente para obtener valores del usuario. Un usuario puede responder con un valor de slot que contenga palabras adicionales, como «una pizza grande, por favor» o «prefiero el tamaño pequeño». Amazon Lex V2 sigue entendiendo el valor de la ranura.

2. Introducción a AWS Lex v2. Slots.



Tipo de slot— Cada slot tiene un tipo. Puede crear su propio tipo de slot o utilizar tipos de slot integrados. Por ejemplo, puede crear y utilizar los siguientes tipos de slot para la intención OrderPizza:

- Tamaño: con los valores de enumeración Small, Medium y Large.
- Masa: con los valores de enumeración Thick y Thin.

2. Introducción a AWS Lex v2. Versión.



Una **versión** es una instantánea numerada de su trabajo que puede publicar para su uso en diferentes partes de su flujo de trabajo, como, por ejemplo, el desarrollo, la implementación beta y la producción. Después de crear una versión, puede utilizar un bot tal y como estaba cuando se creó la versión. Después de crear una versión, esta no cambia mientras continúa trabajando en su aplicación.

2. Introducción a AWS Lex v2. Nuestro primer bot.

Seleccionamos
módulos.

[Página de inicio](#)

[Contenidos](#)

[Foros de discusión](#)

AWS Academy Learner Lab [65336]



El Laboratorio para el alumnado de AWS Academy proporciona un entorno de pruebas de larga duración para la exploración ad hoc de los servicios de AWS. Dentro de esta clase, los alumnos tendrán acceso a **un conjunto restringido de servicios**. No todos los tutoriales de documentación o los laboratorios de muestra de AWS que funcionan en una cuenta de producción de AWS funcionarán en el entorno del laboratorio para el alumnado. Mantendrás el acceso a los recursos de AWS configurados en este entorno mientras dure este curso. Limitamos tu presupuesto (100 USD), por lo que debes tener cuidado para evitar cargos que lo consuman demasiado rápido. Si superas el presupuesto, perderás el acceso al entorno de trabajo.

Cada sesión dura cuatro horas por defecto, aunque puedes prolongar una sesión para que dure más tiempo pulsando el botón de inicio para reiniciar el temporizador de la sesión. Al final de cada sesión, los recursos que hayas creado se mueren, cerramos automáticamente las instancias EC2. Otros recursos, como las instancias RDS, siguen funcionando. Ten en cuenta que no detenemos algunas funciones de AWS, por lo que pueden seguir incurriendo en cargos entre sesiones, como un Elastic Load Balancer o un NAT. Es posible que desees eliminar esos tipos de recursos y volver a crearlos cuando sea necesario para probar tu trabajo durante una sesión. Tendrás acceso a este entorno mientras dure la clase en la que estás matriculado. Cuando termine la clase, también terminará tu acceso al laboratorio.

Solo para instructores/profesores

Si eres un instructor que utiliza un laboratorio para el alumnado en tu curso, consulta el área de **Recursos** de la página de inicio del Portal de AWS Academy para ver la lista de servicios compatibles con cada clase del laboratorio para el alumnado. El entorno de pruebas se utiliza para el trabajo de proyectos, ejercicios de laboratorio o prácticas diseñadas por el instructor que se crean y prueban dentro del laboratorio.

Comenzar



Selecciona **Módulos** para
iniciar el curso.



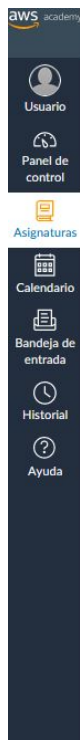
Usa **Debates** para conectar
con compañeros.



Consulta **Soporte del curso**
para obtener ayuda.

2. Introducción a AWS Lex v2. Nuestro primer bot.

Lanzamiento
del Laboratorio
para el
alumnado de
AWS Academy



[Página de inicio](#)

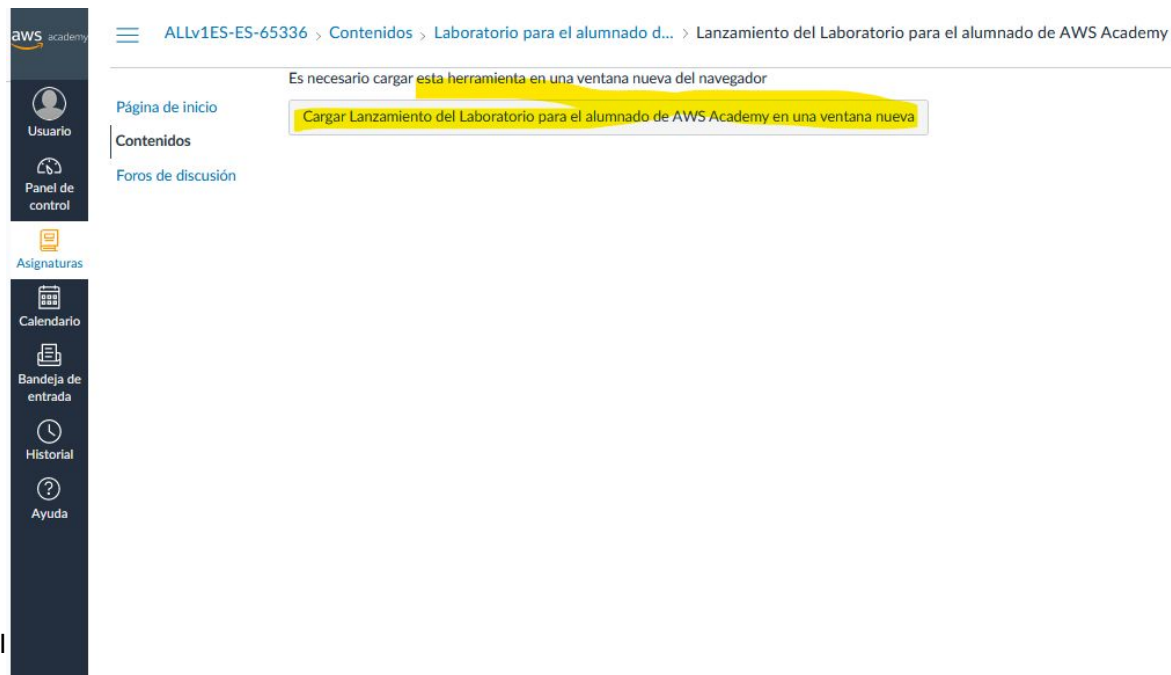
[Contenidos](#)

[Foros de discusión](#)

Menú de contenidos general para el curso	
🔗	Encuesta previa al curso [Pre-Course Survey ES-ES]
🔗	Guía del alumno del Laboratorio para el alumnado de AWS Academy
▼ Conformidad y seguridad del Laboratorio para el alumnado de AWS Academy	
🔗	Aprende a utilizar eficazmente el Laboratorio para el alumnado de Academy
📄	Módulo Prueba de conocimientos 100 puntos Puntuación mínima 70.0
▼ Laboratorio para el alumnado de AWS Academy	
🔗	Lanzamiento del Laboratorio para el alumnado de AWS Academy
▼ Recursos de los Laboratorios para el alumnado de AWS Academy	
🔗	Demo - Cómo acceder al Laboratorio para el alumnado
🔗	Demo - Consejos generales para solucionar problemas
🔗	Demo - Cómo lanzar servicios a través de la consola de AWS

2. Introducción a AWS Lex v2. Nuestro primer bot.

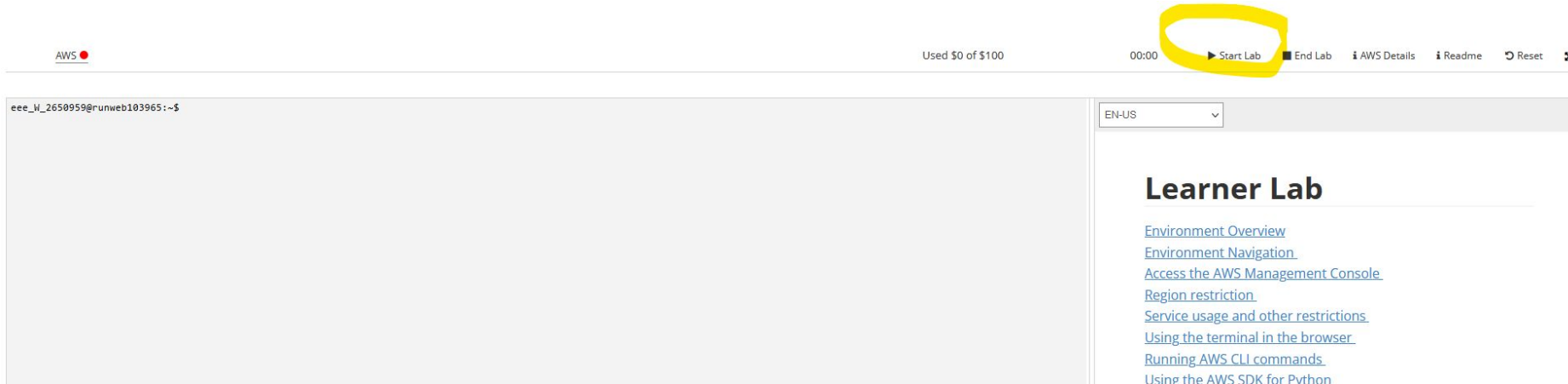
Lanzamiento en una ventana nueva



The screenshot shows the AWS Academy interface. On the left is a dark blue sidebar with the AWS Academy logo at the top. Below the logo are icons and labels for: 'Usuario' (User), 'Panel de control' (Control Panel), 'Asignaturas' (Courses), 'Calendario' (Calendar), 'Bandeja de entrada' (Inbox), 'Historial' (History), and 'Ayuda' (Help). The main content area has a breadcrumb trail: 'ALLv1ES-ES-65336 > Contenidos > Laboratorio para el alumnado d... > Lanzamiento del Laboratorio para el alumnado de AWS Academy'. Below the breadcrumb is a message: 'Es necesario cargar esta herramienta en una ventana nueva del navegador'. A yellow box highlights the text 'Cargar Lanzamiento del Laboratorio para el alumnado de AWS Academy en una ventana nueva'.

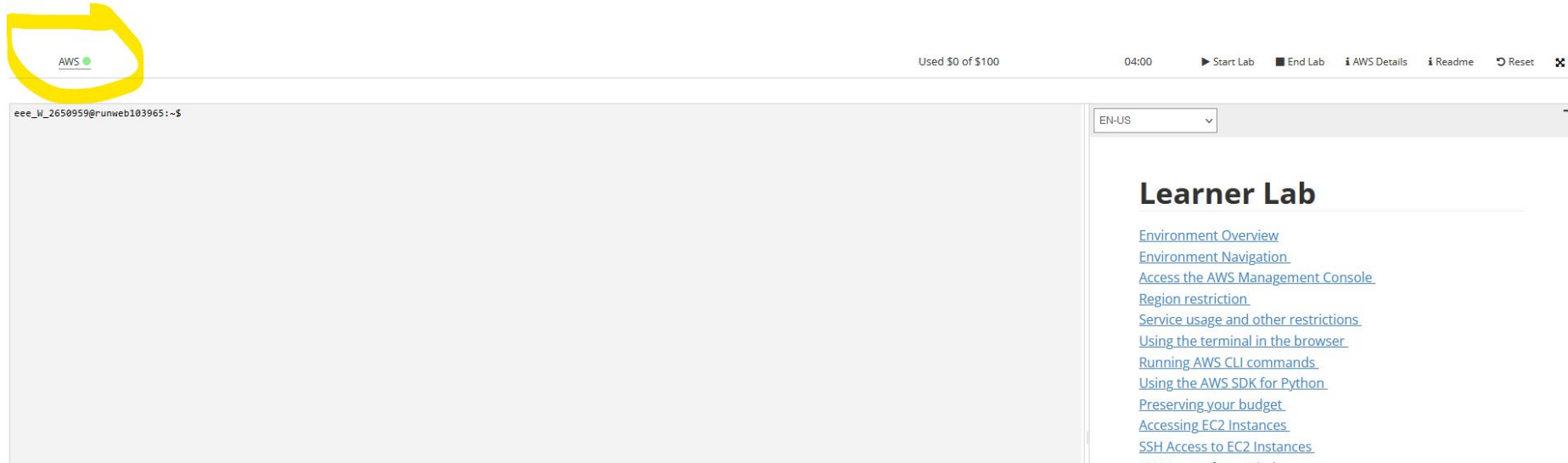
2. Introducción a AWS Lex v2. Nuestro primer bot.

Hacemos click en StartLab y esperamos luz verde en AWS. La primera vez tarda unos minutos.



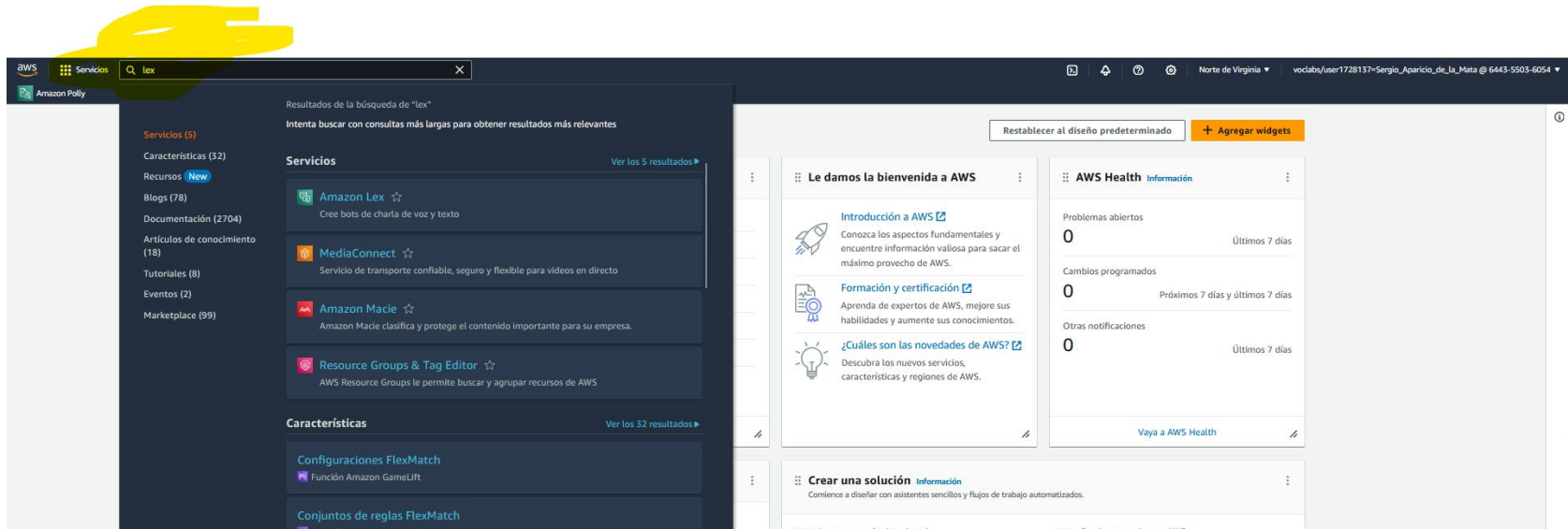
2. Introducción a AWS Lex v2. Nuestro primer bot.

Hacemos click en AWS.



2. Introducción a AWS Lex v2. Nuestro primer bot.

En el buscador de servicios insertamos Lex y lo cargamos.



2. Introducción a AWS Lex v2. Nuestro primer bot.

Esta es la pantalla principal de AWS, damos al botón de la derecha Creat Bot.

The screenshot displays the Amazon Lex console interface. On the left, a sidebar lists navigation options: Bots, Bot templates, Networks of bots, Test workbench, and Recursos relacionados. The main panel is titled 'Lex > Bots' and features a 'Bots (1)' section with a search bar and a table of existing bots. The table contains one entry, 'prueba', which is in an 'Available' state. Below the bots section is a 'Historial de importación o exportación (0)' section, which is currently empty.

Nombre	Descripción	Estado	Última versión	Last updated
prueba	-	Available	-	2 días hace

Tipo	Bot	Estado	Errores	Última actualización	Archivo	Versión
No se han encontrado registros de importación y exportación						

2. Introducción a AWS Lex v2. Nuestro primer bot.

Comience con un
ejemplo y “BookTrip”

Método de creación

- ☐ Crear un bot en blanco
Cree un bot básico sin lenguajes, intenciones ni tipos de ranuras preconfigurados.
- ☒ **Comience con un ejemplo**
Un bot de ejemplo tiene idiomas, intenciones y tipos de ranura preconfigurados. Usted puede cambiar esta configuración.
- ☐ Comience con las transcripciones
Genere automáticamente las intenciones a partir de las transcripciones de las conversaciones que cargue. Solo está disponible el idioma inglés (EE. UU.) cuando se empieza con una transcripción.

Ejemplos de bots

Un bot de ejemplo tiene lenguajes, intenciones y tipos de ranura preconfigurados. Podrá cambiar estos ajustes después de crear el bot.

Ejemplo de bot

BookTrip
Bot to make reservations necessary for a visit to a city

Idiomas admitidos: Alemán (AT), Alemán (DE), Catalán (ES), Coreano (KR), Español (ES), Español (LATAM), Español (US), Francés (CA), Francés (FR), Inglés (AU), Inglés (GB), Inglés (IN), Inglés (US), Inglés (ZA), Italiano (IT), Japonés (JP), Mandarín (RPC), Portugués (BR), Portugués (PT)

2. Introducción a AWS Lex v2. Nuestro primer bot.

Configuración de bots

Nombre del bot

miprimerbot

Máximo 100 caracteres. Caracteres válidos: A-Z, a-z, 0-9, -, _

Descripción - *opcional*

Esta descripción aparece en la página de lista del bot. Puede ayudarle a identificar el propósito de su bot.

Mi primer bot de la asignatura de modelos de IA

Máximo 200 caracteres.

Permisos de IAM [Info](#)

Los permisos de IAM se utilizan para acceder a otros servicios en su nombre.

Rol de tiempo de ejecución

Elija un rol que defina los permisos para su bot. Para crear un rol personalizado, utilice la consola de IAM.

☐ Crear un rol con permisos básicos de Amazon Lex.

☒ Usar un rol existente.

Rol existente

Elija un rol existente para utilizarlo con este bot de Lex. El rol debe tener permisos para cargar registros en Amazon CloudWatch logs (los registros de Amazon CloudWatch).

AWSServiceRoleForLexV2Bots_G0XQZ0G64V7

Configuración de bots

Nombre del bot

miprimerbot

Máximo 100 caracteres. Caracteres válidos: A-Z, a-z, 0-9, -, _

Descripción - *opcional*

Esta descripción aparece en la página de lista del bot. Puede ayudarle a identificar el propósito de su bot.

Mi primer bot de la asignatura de modelos de IA

Máximo 200 caracteres.

Permisos de IAM [Info](#)


Los permisos de IAM se utilizan para acceder a otros servicios en su nombre.

Rol de tiempo de ejecución

Elija un rol que defina los permisos para su bot. Para crear un rol personalizado, utilice la consola de IAM.

☒ Crear un rol con permisos básicos de Amazon Lex.

☐ Usar un rol existente.

 La creación de un rol lleva unos minutos. No elimine el rol ni modifique las políticas de confianza o permisos de este rol hasta que hayamos terminado de crearlo.

Nuevo rol

Amazon Lex crea un rol de tiempo de ejecución con permiso para cargar en Amazon CloudWatch Logs.

CI AWSServiceRoleForLexV2Bots_NL9ZBFRWV8M

a/ Modelos de Inteligencia Artificial

2. Introducción a AWS Lex v2. Nuestro primer bot.

Agregar idioma al bot [Info](#)

▼ Idioma: Español (ES)

Seleccionar idioma

Español (ES) ▼

Descripción - *opcional*

Máximo 200 caracteres.

Interacción por voz

La voz de texto a voz que su bot utiliza para interactuar con los usuarios.

Ninguno. Esta es sólo una aplicación basada en texto ▼

⚠ User: arn:aws:sts::644355036054:assumed-role/voclabs/user1728137=Sergio_Aparicio_de_la_Mata is not authorized to perform: polly:DescribeVoices because no identity-based policy allows the polly:DescribeVoices action

Umbral de confianza de clasificación de intenciones

0.40

Min:0.00, máx:1.00.

Cancelar [Agregar otro idioma](#) Terminado

2. Introducción a AWS Lex v2. Nuestro primer bot.

Ahora debemos de dar a crear y ya tendríamos nuestro primer bot.

The screenshot displays the AWS Lex console interface. At the top, a green banner indicates 'Bot: miprimerbot creado correctamente'. Below this, a blue header shows the path 'Crear idioma Español (ES) en bot: miprimerbot'. The main content area is titled 'Intención: BookHotel' and includes a description: 'Una intención representa una acción que satisface la solicitud de un usuario. Las intenciones pueden tener argumentos denominados ranuras que representan información variable.' It features a 'Flujo de conversación' section and a 'Detalles de intención' section with fields for 'Nombre de la intención' (BookHotel) and 'Descripción - opcional' (Intención de reservar un hotel en StayBooker). The 'Ejemplos de enunciados' section shows three example phrases. The bottom of the console has a dark blue bar with a 'Versión borrador' dropdown, the language 'Español (ES)' with a 'Creando' status, and buttons for 'Guardar intención', 'Crear', and 'Prueba'.

2. Introducción a AWS Lex v2. Nuestro primer bot.



¿ Qué ha configurado el asistente de ejemplo?

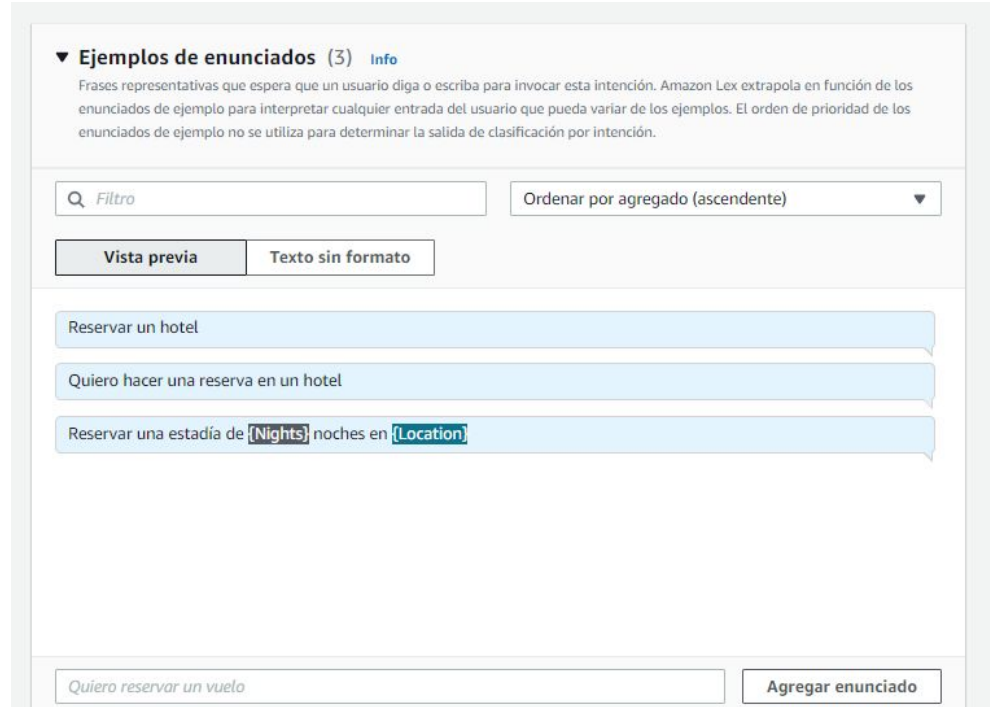
3 intenciones:

- BookCar.
- BookHotel.
- FallbackIntent.

2. Introducción a AWS Lex v2. Nuestro primer bot.

Entramos en BookHotel

Vemos que hay 3 enunciados de ejemplo. En el último intenta describir las variables fechas y localización.



▼ Ejemplos de enunciados (3) Info

Frases representativas que espera que un usuario diga o escriba para invocar esta intención. Amazon Lex extrapola en función de los enunciados de ejemplo para interpretar cualquier entrada del usuario que pueda variar de los ejemplos. El orden de prioridad de los enunciados de ejemplo no se utiliza para determinar la salida de clasificación por intención.

🔍 Filtro

Ordenar por agregado (ascendente) ▼

Vista previa Texto sin formato

Reservar un hotel

Quiero hacer una reserva en un hotel

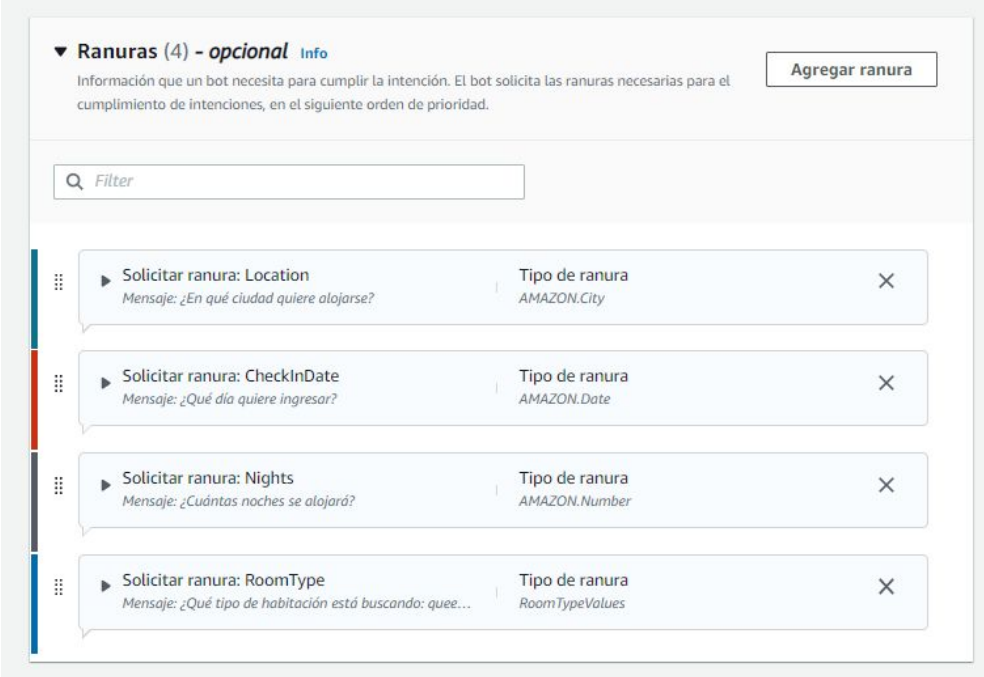
Reservar una estadia de [Nights] noches en [Location]

Quiero reservar un vuelo

Agregar enunciado

2. Introducción a AWS Lex v2. Nuestro primer bot.

Tenemos configurado 4 slot, información que necesita nuestro CRM para procesar la reserva.



The screenshot displays the AWS Lex console interface for configuring a bot. At the top, there is a section titled "▼ Ranuras (4) - opcional Info" with an "Agregar ranura" button. Below this, a search bar labeled "Filter" is present. The main content area lists four configured slots, each with a vertical ellipsis icon on the left, a title, a message, a type, and a delete button (X).

Slot Title	Message	Type
Solicitar ranura: Location	Mensaje: ¿En qué ciudad quiere alojarse?	AMAZON.City
Solicitar ranura: CheckInDate	Mensaje: ¿Qué día quiere ingresar?	AMAZON.Date
Solicitar ranura: Nights	Mensaje: ¿Cuántas noches se alojará?	AMAZON.Number
Solicitar ranura: RoomType	Mensaje: ¿Qué tipo de habitación está buscando: quee...	RoomTypeValues

2. Introducción a AWS Lex v2. Nuestro primer bot.

Al ser una reserva es muy importante facilitar esta información al cliente.

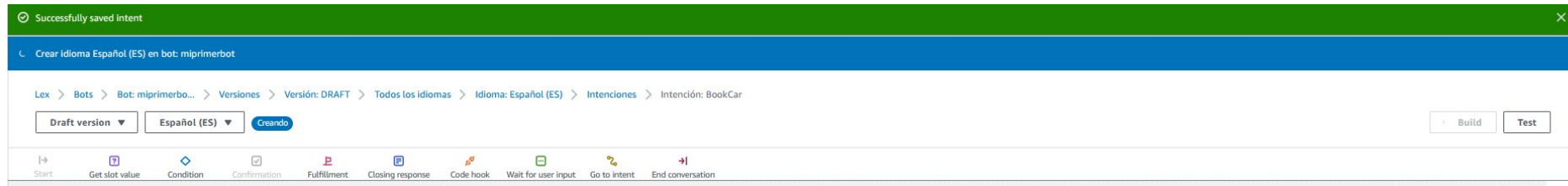


The screenshot displays the AWS Lex v2 Confirmation interface. At the top, the word "Confirmation" is followed by a blue "Info" icon. To the right, there is a toggle switch labeled "Activo". Below this, a subtitle reads: "Las solicitudes ayudan a aclarar si el usuario desea cumplir la intención o cancelarla." The main content area is divided into two columns. The left column is titled "► Solicitudes para confirmar la intención" and shows a message: "Mensaje: Bien, lo tengo anotado para una estadía de {...}". The right column is titled "Respuestas enviadas cuando el usuario rechaza la intención" and shows a message: "Mensaje: Bien, he cancelado su reserva en progreso."

2. Introducción a AWS Lex v2. Nuestro primer bot.

Vamos a probar a hacer la reserva.

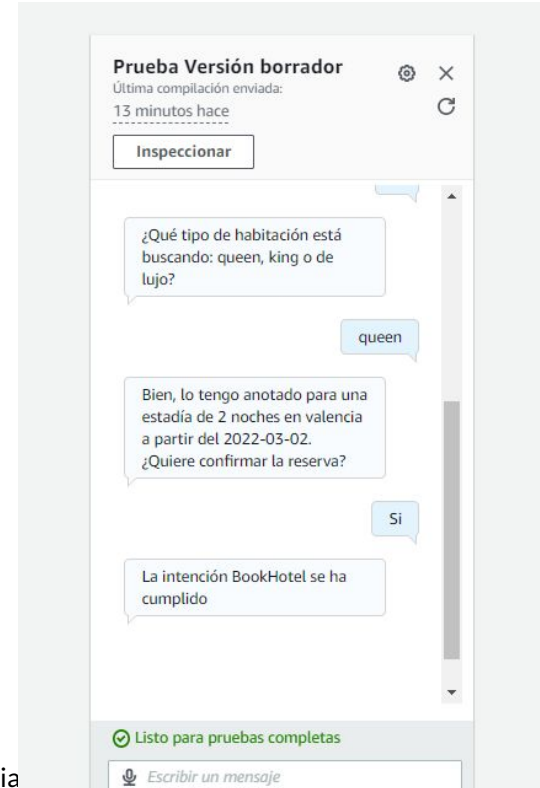
Damos al botón de build y después test.



2. Introducción a AWS Lex v2. Nuestro primer bot.

Hacemos todo el recorrido de una reserva de habitación.

¿ Qué primeras impresiones teneis ?



2. Introducción a AWS Lex v2. Nuestro primer bot.



Ahora intenta alquilar un coche. ¿ Qué pasa ?

2. Introducción a AWS Lex v2. Nuestro primer bot.



Vamos primero al solventar el problema del enunciado.

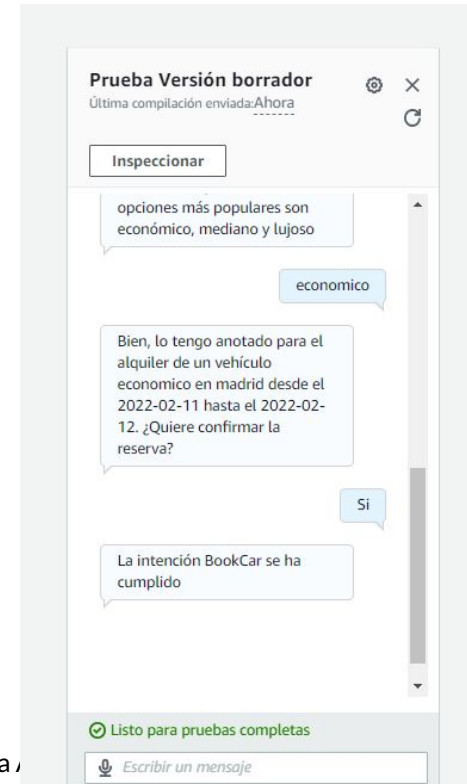
Nos vamos a BookCar

- Quiero alquilar un coche.
- Quiero reservar un auto.

Pulsamos en “Guardar intención”, crear y después probar.

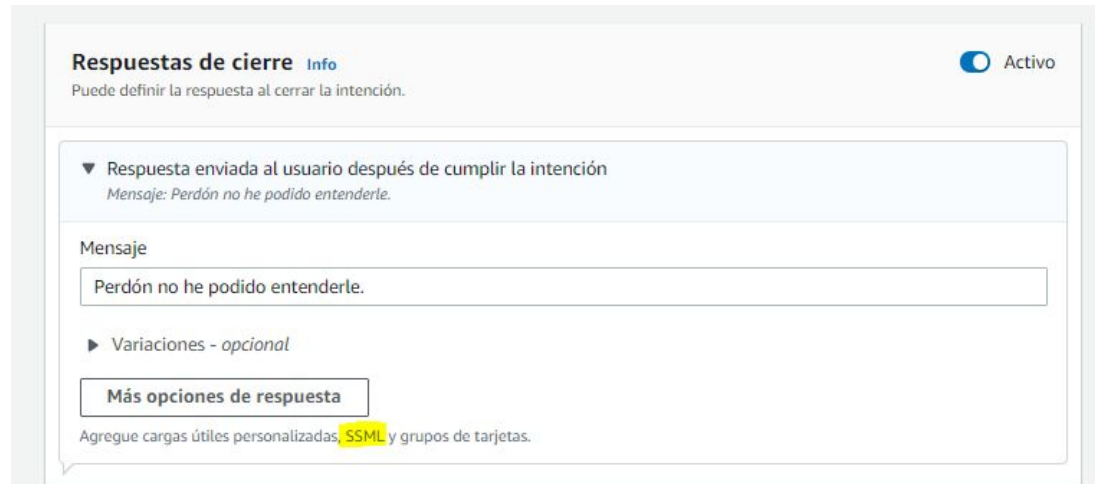
2. Introducción a AWS Lex v2. Nuestro primer bot.

Con la inclusión del verbo alquilar y del sustantivo. Hemos podido cumplir una intención básica de reserva de coches.



2. Introducción a AWS Lex v2. Nuestro primer bot.

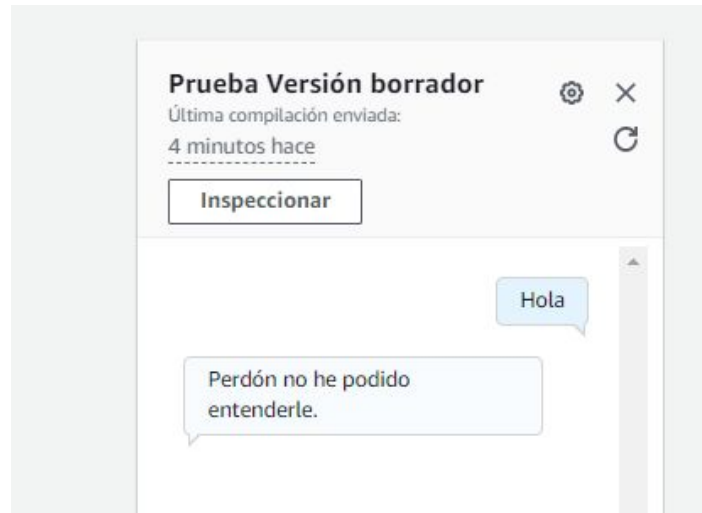
El otro problema es que no tenemos configurado la intención FallbackIntent. Hacemos click sobre la intención y configuramos la siguiente respuesta de cierre.



The screenshot shows the AWS Lex console interface for configuring a bot's response. At the top, it says "Respuestas de cierre" with an "Info" link and a toggle switch labeled "Activo". Below this, a subtitle reads "Puede definir la respuesta al cerrar la intención." The main section is titled "Respuesta enviada al usuario después de cumplir la intención" with a subtitle "Mensaje: Perdón no he podido entenderle." There is a text input field containing "Perdón no he podido entenderle." Below the input field, there is a section for "Variaciones - opcional" with a button labeled "Más opciones de respuesta". At the bottom, there is a note: "Agregue cargas útiles personalizadas, SSML y grupos de tarjetas."

2. Introducción a AWS Lex v2. Nuestro primer bot.

Damos a crear y probar. Si ponemos “Hola” como no hay ninguna intención compatible nos saltara la intención FallbackIntent .

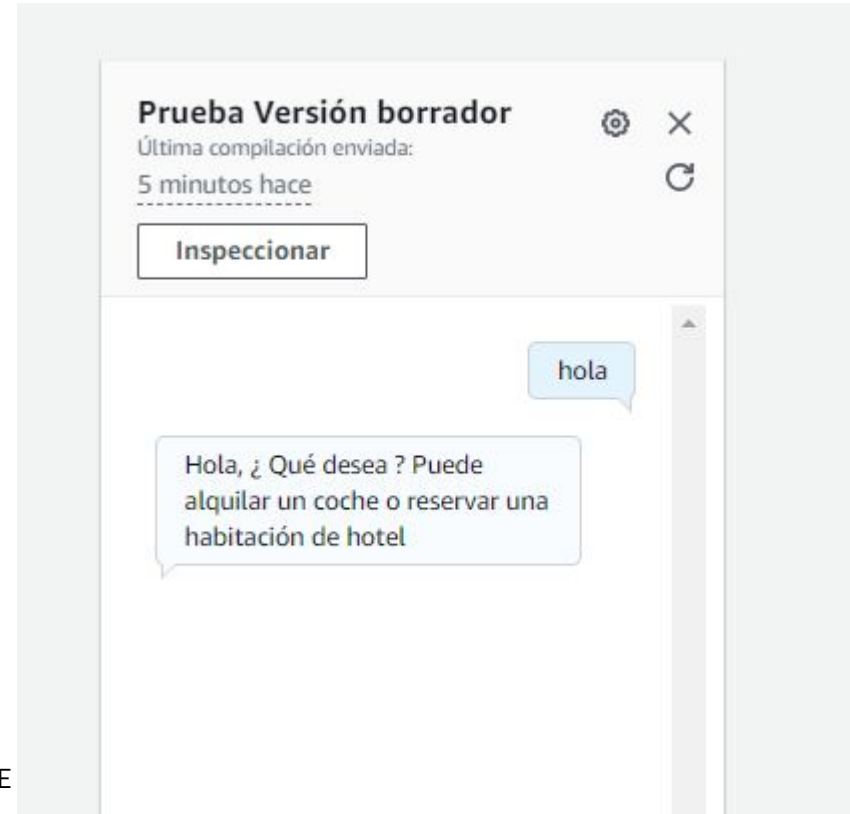


2. Introducción a AWS Lex v2. Nuestro primer bot.

Ejercicio: Crear una intención ante un saludo inicial del usuario.

“Hola, ¿ Qué desea ? Puede alquilar un coche o reservar una habitación de hotel.”

El resultado tiene que ser como el pantallazo que se muestra a continuación.



2. Introducción a AWS Lex v2. Alexa

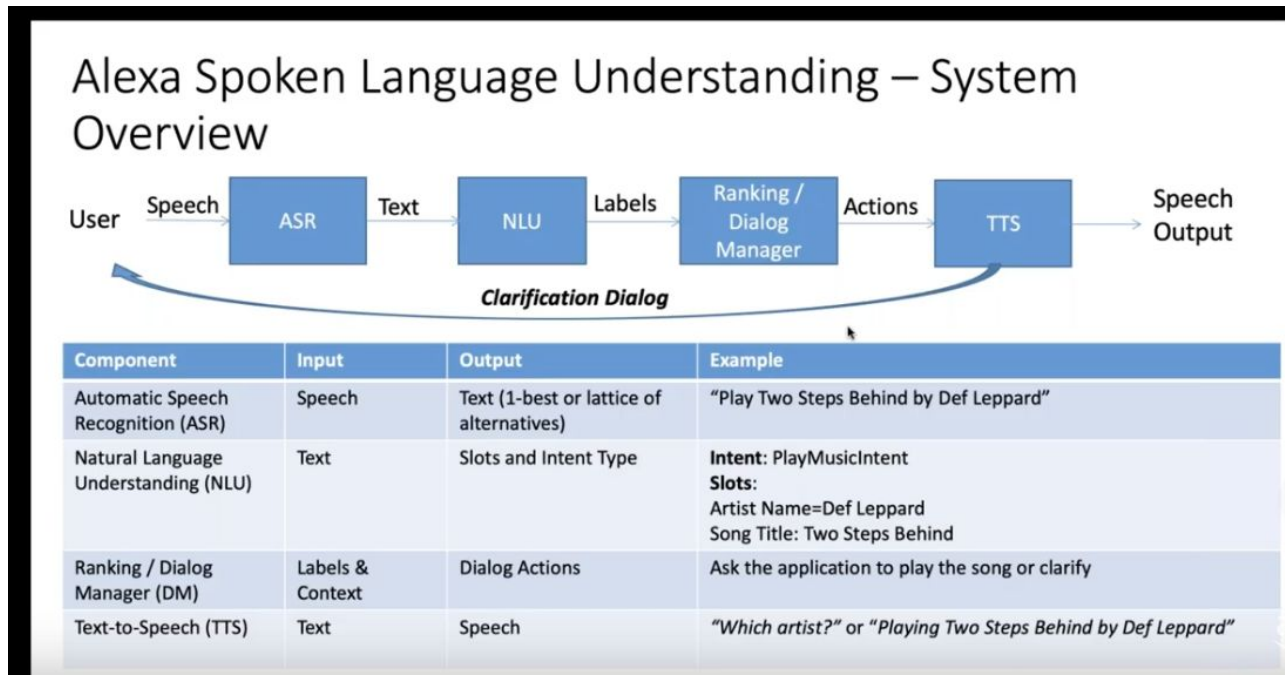


En la UT5 vimos como funcionaba Alexa a nivel de infraestructura y funcional.

Con lo aprendido hasta el momento ya podemos analizar los elementos TTS, ASR y NLU, los flujos entre estos y variables de entrada / salida.

Vamos a ver un ejemplo en la siguiente diapositiva.

2. Introducción a AWS Lex v2. Alexa



2. Introducción a AWS Lex v2. Alexa

EJ1: Indicar el siguiente flujo en Alexa ¿ Qué tiempo hace en Valencia ?

Componente	Entrada	Salida	Ejemplo

2. Introducción a AWS Lex v2. Alexa

EJ2: Indicar el siguiente flujo en Alexa ¿ Cuáles son las noticias del día ?

Componente	Entrada	Salida	Ejemplo

2. Introducción a AWS Lex v2. Alexa

EJ3: Indicar el siguiente flujo en Alexa ¿ Pon una alarma para las 7 de la mañana ?

Componente	Entrada	Salida	Ejemplo
IES Abastos		CE Inteligencia Artificial y Big Data/ Modelos de Inteligencia Artificial	