

20 DE MAYO DE 2024



HABILIDAD PARA EL IES ÁNGEL CORELLA EN ALEXA

MONTORO PALLERO, VÍCTOR
IES ÁNGEL CORELLA
DESARROLLO DE APLICACIONES WEB

Resumen:

El presente trabajo de investigación se centra en el desarrollo de una habilidad para Alexa destinada al IES Ángel Corella en Colmenar Viejo. Esta habilidad tiene como objetivo principal proporcionar información sobre la oferta educativa del centro, incluyendo programas académicos, proceso de admisión y noticias relevantes. La necesidad de esta habilidad surge de la creciente demanda de información accesible y actualizada por parte de estudiantes, padres y personal educativo.

El diseño básico del estudio se fundamenta en el uso del Alexa Skills Kit SDK para Python, aprovechando su capacidad para manejar intenciones y proporcionar respuestas dinámicas a través de archivos de texto externos. La estructura modular del código facilita la incorporación de nuevas funcionalidades y la gestión eficiente de diferentes aspectos de la habilidad.

Durante el desarrollo de la habilidad, se realizaron diversos descubrimientos relacionados con la implementación de intenciones específicas y el manejo de respuestas personalizadas. Así mismo, se identificaron tendencias en las consultas de los usuarios, lo que permitió ajustar y mejorar continuamente la experiencia de usuario ofrecida por la habilidad.

En cuanto a interpretaciones y conclusiones, se destaca la importancia de la adaptabilidad y la mejora continua en el desarrollo de habilidades para asistentes de voz. La interacción con usuarios reales proporcionó información valiosa para futuras iteraciones y mejoras en la habilidad, destacando la relevancia de la retroalimentación constante en el proceso de desarrollo.

En resumen, la habilidad desarrollada para el IES Ángel Corella representa un avance significativo en la accesibilidad y la comunicación dentro de la comunidad educativa, demostrando el potencial de las tecnologías de voz para mejorar la experiencia de usuario en entornos educativos.

Abstract:

The present research work focuses on the development of an Alexa skill for IES Ángel Corella in Colmenar Viejo. The main objective of this skill is to provide information about the educational offerings of the center, including academic programs, the admission process, and relevant news. The need for this skill arises from the growing demand for accessible and up-to-date information from students, parents, and educational staff.

The basic design of the study is based on the use of the Alexa Skills Kit SDK for Python, leveraging its capability to handle intents and provide dynamic responses through external text files. The modular structure of the code facilitates the incorporation of new functionalities and the efficient management of different aspects of the skill.

During the development of the skill, various discoveries were made related to the implementation of specific intents and the handling of customized responses. Additionally, user query trends were identified, allowing for continuous adjustment and improvement of the user experience offered by the skill.

In terms of interpretations and conclusions, the importance of adaptability and continuous improvement in the development of skills for voice assistants is highlighted. Interaction with real users provided valuable information for future iterations and enhancements to the skill, underscoring the relevance of constant feedback in the development process.

In summary, the skill developed for IES Ángel Corella represents a significant advancement in accessibility and communication within the educational community, demonstrating the potential of voice technologies to enhance the user experience in educational settings.

Índice

1. Introducción	4
2. Justificación del tema elegido	5
3. Objetivos	6
4. Fundamentación teórica	7
5. Metodología de desarrollo	8
6. Cronografía	9
Requisitos:	9
Requisitos funcionales:	9
Requisitos no funcionales:	10
Análisis:	10
Identificación de entidades y acciones:	11
Diagrama de flujo de la conversación:	11
Programación:	13
Implementación de la lógica de negocio:	13
Manejo de Intents y Slots:	13
Pruebas:	13
Pruebas Unitarias:	13
Pruebas de Integración:	13
Pruebas de aceptación de usuario:	14
7. ¿Qué ocurre cuando se habla con Alexa?	14
8. Desarrollo	16
Conceptos fundamentales y herramientas utilizadas:	16
Creación de la habilidad:	18
Gestión de intenciones y clases en la implementación de la habilidad	25
Estructura de las clases de manejadores de intenciones:	28
Lectura dinámica de archivos .txt	29
9. Diagrama UML:	31
10. Guía de usuario	32
12. Adaptación para Otro Centro	33
11. Mejoras	34
12. Conclusiones	35
13. Bibliografía:	37

1. Introducción

En la era digital actual, la tecnología está transformando rápidamente la forma en que accedemos y consumimos información en diversos ámbitos de la vida cotidiana, y la educación no es una excepción. Con el crecimiento de las tecnologías de la información y la comunicación, las instituciones educativas enfrentan el desafío de adaptarse a un entorno en constante cambio, donde la accesibilidad y la relevancia de la información educativa son aspectos cruciales para estudiantes, padres y personal educativo.

Este trabajo de fin de grado tiene como objetivo principal desarrollar una habilidad de voz en Alexa para el centro educativo IES Ángel Corella, una institución educativa comprometida con la excelencia académica y la formación integral de sus estudiantes. La creación de esta habilidad no solo busca mejorar la visibilidad del centro educativo, sino también proporcionar una herramienta innovadora y accesible para que los usuarios obtengan información detallada sobre los cursos disponibles, los requisitos de admisión y otras características relevantes de la institución.

En este contexto, la introducción proporcionará una visión general del contenido y las contribuciones más relevantes de este trabajo. Se discutirán los antecedentes y la justificación detrás del desarrollo de la habilidad de voz, los objetivos que guían esta investigación, la metodología utilizada para su diseño e implementación, así como el alcance y las posibles implicaciones de este proyecto.

2. Justificación del tema elegido

La elección de este trabajo de fin de grado se fundamenta en la necesidad de abordar el creciente impacto de la tecnología en el ámbito educativo, específicamente en el contexto del IES Ángel Corella. En la era digital actual, la tecnología desempeña un papel fundamental en la forma en que accedemos, procesamos y compartimos información. Con la aparición de las tecnologías de la información y la comunicación, se presenta una oportunidad sin precedentes para mejorar la accesibilidad y la eficacia de la educación.

La justificación de este trabajo de fin de grado radica en varios aspectos fundamentales:

1. **Relevancia y pertinencia:** En un mundo donde la tecnología avanza rápidamente, es crucial que las instituciones educativas se adapten para satisfacer las necesidades cambiantes de los estudiantes, padres y personal educativo. Desarrollar una habilidad de voz para el centro anteriormente indicado responde a la necesidad de utilizar herramientas innovadoras para mejorar la visibilidad y accesibilidad de los programas ofrecidos.
2. **Mejora de la experiencia del usuario:** Desarrollar una habilidad de voz ofrece una experiencia interactiva y accesible para los usuarios. Les permite interactuar mediante comandos de voz, proporcionándoles de manera eficiente información relevante sobre los cursos disponibles, requisitos de admisión y otras características importantes de la institución educativa.
3. **Innovación tecnológica:** El desarrollo de habilidades para plataformas de voz como Alexa representa una oportunidad única para explorar nuevas formas de interactuar con la tecnología y mejorar la experiencia del usuario. Al aprovechar las capacidades de la tecnología de voz, este proyecto busca impulsar la innovación en el ámbito educativo y promover prácticas pedagógicas modernas.
4. **Avance en el campo educativo:** Este proyecto no solo tiene como objetivo mejorar la promoción de la oferta educativa del IES Ángel Corella, sino que también busca establecer las bases para futuras investigaciones y desarrollos en el ámbito educativo. Al explorar el potencial de la tecnología de voz en este contexto, este trabajo contribuye al avance del conocimiento y la práctica educativa

3. Objetivos

Los objetivos del trabajo de fin de grado se centran en proporcionar una guía clara y detallada sobre las metas específicas que se persiguen con el desarrollo de habilidades para Alexa en el contexto del IES Ángel Corella. Estos objetivos abarcan tanto aspectos técnicos como funcionales y de experiencia de usuario, con el fin de garantizar un diseño integral y efectivo de la habilidad de voz. A continuación, se detallan los objetivos específicos:

1. **Desarrollo técnico:** El primer objetivo consiste en llevar a cabo el desarrollo técnico de la habilidad de voz para Alexa, lo que incluye la programación y configuración de la interfaz de usuario, la integración de intenciones y la implementación de funcionalidades específicas según los requisitos del proyecto.
2. **Mejora de la experiencia del usuario:** Nuestro objetivo es optimizar la experiencia del usuario al diseñar una interacción fluida y comprensible con la habilidad de voz. Esto implica realizar pruebas de usabilidad y recopilar comentarios para identificar áreas de mejora y ajustar la comprensión y respuesta de la habilidad según sea necesario.
3. **Personalización y adaptabilidad:** Otro objetivo es integrar características de personalización y adaptabilidad que permitan a los usuarios acceder a información relevante y específica según sus necesidades e intereses individuales a través de comandos de voz. Esto podría incluir la capacidad de solicitar resultados específicos, configurar preferencias de usuario y recibir recomendaciones personalizadas adaptadas a sus interacciones con la habilidad en Alexa.
4. **Integración de contenido educativo:** Se pretende integrar contenido educativo relevante y actualizado en la habilidad de voz, proporcionando información detallada sobre los programas educativos ofrecidos por el IES Ángel Corella.
5. **Evaluación y mejora continua:** Por último, se establece como objetivo realizar una evaluación exhaustiva de la habilidad de voz una vez desarrollada, con el fin de identificar áreas de mejora y realizar ajustes para garantizar su efectividad y utilidad a largo plazo. Esto podría implicar la realización de pruebas beta con usuarios reales, análisis de datos de uso y retroalimentación del personal educativo y administrativo del IES Ángel Corella.

En resumen, los objetivos del TFG buscan garantizar un desarrollo integral y efectivo de la habilidad de voz para Alexa, centrándose en aspectos técnicos, funcionales y de experiencia de usuario para ofrecer una herramienta innovadora y útil para la promoción de la oferta educativa del IES Ángel Corella.

4. Fundamentación teórica

La fundamentación teórica de este trabajo proporciona un respaldo sólido a través de la exploración de la literatura existente sobre el desarrollo de habilidades para Alexa, así como de los principios y mejores prácticas en este campo emergente de la tecnología de voz. En este apartado, se abordarán diversos aspectos relacionados con el desarrollo de habilidades para asistentes de voz, centrándose especialmente en Amazon Alexa.

Para comenzar, se analizarán los fundamentos teóricos sobre la interacción humano-computadora (HCI) y la interfaz de usuario conversacional. Se explorarán conceptos clave como la usabilidad, la accesibilidad y la naturalidad en la interacción con asistentes de voz, así como las diferencias en el diseño de interfaces de usuario para dispositivos de voz en comparación con interfaces gráficas tradicionales.

A continuación, se revisará la arquitectura y el funcionamiento básico de las habilidades de Alexa, incluyendo el papel de los servicios en la nube de Amazon AWS, los modelos de voz y lenguaje natural (NLU), y la estructura de las solicitudes y respuestas de voz.

Posteriormente, se profundizará en las mejores prácticas y estrategias de diseño de habilidades para Alexa, abordando aspectos como la estructura de la conversación, el diseño de diálogos, la personalización de respuestas y la optimización del flujo de interacción para mejorar la experiencia del usuario.

Además, se examinarán estudios y casos de éxito previos en el desarrollo de habilidades para Alexa en diferentes contextos, desde aplicaciones educativas hasta comerciales y de entretenimiento, con el objetivo de identificar patrones comunes y lecciones aprendidas que puedan aplicarse al desarrollo de la habilidad de voz para promover la oferta educativa del IES Ángel Corella.

En resumen, la fundamentación teórica de este trabajo proporcionará una base sólida para el diseño y desarrollo de la habilidad de voz, al explorar la literatura existente y las mejores prácticas en el campo del desarrollo de habilidades para Alexa, con el fin de garantizar un enfoque informado y efectivo en todas las etapas del proyecto.

5. Metodología de desarrollo

La metodología de desarrollo utilizada para este proyecto de habilidades para Alexa se basa en un enfoque iterativo e incremental, que permite adaptarse a los requisitos cambiantes y optimizar el proceso de diseño y desarrollo a lo largo del tiempo. A continuación, se describen las principales etapas y elementos de esta metodología:

1. **Análisis de requisitos:** En esta etapa inicial, se identifican y analizan los requisitos del proyecto, incluyendo las necesidades de los usuarios, los objetivos del negocio y las funcionalidades deseadas de la habilidad de voz. Se recopila información sobre el centro IES Ángel Corella y se define el alcance y los objetivos del proyecto.
2. **Diseño de la conversación:** Se desarrolla un diseño detallado de la conversación que guiará la interacción entre el usuario y la habilidad de voz. Se definen los diferentes pasos y opciones disponibles para el usuario, así como las respuestas y acciones correspondientes de la habilidad. Se presta especial atención a la estructura de la conversación, la claridad de las instrucciones y la coherencia en el tono y el estilo de la respuesta.
3. **Desarrollo del modelo de interacción:** En esta fase, se lleva a cabo la implementación del modelo de interacción diseñado en la etapa anterior. Esto implica configurar la lógica de la habilidad, definir las intenciones del usuario y sus correspondientes respuestas, así como integrar las diversas intenciones. Para ello, se emplean herramientas y tecnologías específicas diseñadas para el desarrollo de habilidades para Alexa, como el kit de habilidades de Alexa (ASK) y el lenguaje de programación Python, elegido por su capacidad para adaptarse a nuestras necesidades.
4. **Pruebas y refinamiento:** Se llevan a cabo pruebas exhaustivas para validar el funcionamiento y la usabilidad de la habilidad de voz. Esto incluye pruebas de funcionalidad para asegurar que la habilidad responde correctamente a las solicitudes del usuario, así como pruebas de usabilidad para evaluar la facilidad de uso y la experiencia del usuario. Se recopilan comentarios de los usuarios y se realizan ajustes según sea necesario para mejorar la calidad y la efectividad de la habilidad.
5. **Implementación y despliegue:** Una vez completadas las pruebas y realizado cualquier ajuste necesario, la habilidad de voz se implementa y despliega en el entorno de producción de Alexa. Se realizan las configuraciones finales y se publica la habilidad en la tienda de habilidades de Alexa, donde estará disponible para su uso por parte de los usuarios.

En resumen, la metodología de desarrollo utilizada combina un enfoque centrado en el usuario con prácticas ágiles de desarrollo de software, permitiendo la creación de una habilidad de voz para promover la oferta educativa del IES Ángel Corella que sea efectiva, intuitiva y fácil de usar para los usuarios finales.

6. Cronografía

Requisitos:

Requisitos funcionales:

En el desarrollo de una habilidad de voz para Alexa, los requisitos funcionales son los elementos esenciales que definen las capacidades y funcionalidades que la aplicación debe ofrecer a los usuarios. Estos requisitos establecen las acciones específicas que la habilidad puede realizar en respuesta a las solicitudes de los usuarios a través de comandos de voz.

Se explorarán en detalle los requisitos funcionales necesarios para desarrollar una habilidad de voz efectiva y útil para los usuarios del IES Ángel Corella. Cada requisito funcional identificado jugará un papel fundamental en la creación de una experiencia de usuario enriquecedora y satisfactoria, al tiempo que cumple con los objetivos y expectativas del proyecto. La habilidad presenta los siguientes requisitos funcionales:

Información sobre la oferta educativa:

La habilidad debe ser capaz de proporcionar información detallada sobre centro. Esto incluye descripciones de los programas académicos, cursos disponibles, así como posibles especializaciones.

Proceso de admisión:

La habilidad deberá informar a los usuarios sobre el proceso de admisión al IES Ángel Corella. Esto incluye requisitos de admisión, fechas importantes, procedimientos de solicitud y cualquier documentación necesaria.

Entrega de noticias relevantes:

La habilidad deberá mantener a los usuarios actualizados sobre noticias relevantes relacionadas con el IES Ángel Corella. Estas noticias pueden incluir eventos escolares, logros de estudiantes, actualizaciones, etc.

Información sobre el centro:

La habilidad debe proporcionar a los usuarios detalles relevantes sobre el IES Ángel Corella. Esto puede incluir información sobre la misión del centro, instalaciones disponibles, personal docente, horarios de atención, ubicación y cualquier otro dato relevante que los usuarios puedan necesitar conocer sobre el centro educativo.

Requisitos no funcionales:

Al diseñar y desarrollar una habilidad de voz para Alexa destinada al IES Ángel Corella, es fundamental considerar una serie de requisitos no funcionales que complementan los aspectos técnicos y funcionales de la aplicación. Estos requisitos abordan aspectos cruciales relacionados con la experiencia del usuario, la escalabilidad y el rendimiento del sistema.

Usabilidad intuitiva:

Se espera que la habilidad sea fácil de usar para los usuarios, con comandos de voz claros y una estructura de conversación natural.

Velocidad de respuesta rápida:

La habilidad debe tener una velocidad de respuesta rápida para garantizar una experiencia de usuario fluida y sin interrupciones. Esto implica que las respuestas a las consultas de los usuarios deben ser instantáneas o con una mínima demora perceptible.

Escalabilidad:

La implementación de la habilidad debe ser escalable para manejar un aumento en el número de usuarios sin comprometer el rendimiento del sistema. Se debe garantizar que la habilidad pueda adaptarse a un crecimiento potencial de la base de usuarios sin afectar negativamente a la experiencia del usuario.

Estos requisitos proporcionan una guía integral para el diseño, desarrollo y evaluación de la habilidad de Alexa relacionada con el IES Ángel Corella. Se espera que estos criterios sean considerados y cumplidos durante todas las etapas del proyecto.

Análisis:

La etapa de análisis en el desarrollo de una habilidad de voz para Alexa es fundamental para comprender a fondo las necesidades de los usuarios y establecer los requisitos específicos que guiarán el diseño y la implementación del proyecto. Durante esta fase, se lleva a cabo una evaluación exhaustiva de las funcionalidades requeridas, las interacciones esperadas y los flujos de conversación entre el usuario y la habilidad.

El análisis se centra en dos aspectos principales: la identificación de las entidades y acciones relevantes para la habilidad, y la creación de diagramas de flujo de conversación que representen visualmente cómo debería interactuar el usuario con la aplicación en diferentes escenarios.

A través de esta sección, exploraremos en detalle las entidades clave relacionadas con el Instituto de Educación Secundaria (IES) Ángel Corella, así como las acciones que los usuarios esperan realizar con cada una de ellas. Además, se desarrollarán diagramas de flujo de conversación para ilustrar de

manera clara y concisa cómo deberían transcurrir las interacciones entre el usuario y la habilidad en diferentes situaciones.

El análisis proporcionará una base sólida para el diseño y la implementación efectiva de la habilidad, asegurando que se satisfagan las necesidades de los usuarios de manera intuitiva y eficiente.

Identificación de entidades y acciones:

1. Oferta Educativa:

- Obtener información sobre programas académicos.
- Consultar detalles de cursos disponibles.
- Solicitar información sobre especializaciones.

2. Admisión:

- Obtener detalles sobre el proceso de admisión.
- Consultar fechas importantes y procedimientos.
- Solicitar requisitos de solicitud y documentación necesaria.

3. Noticias:

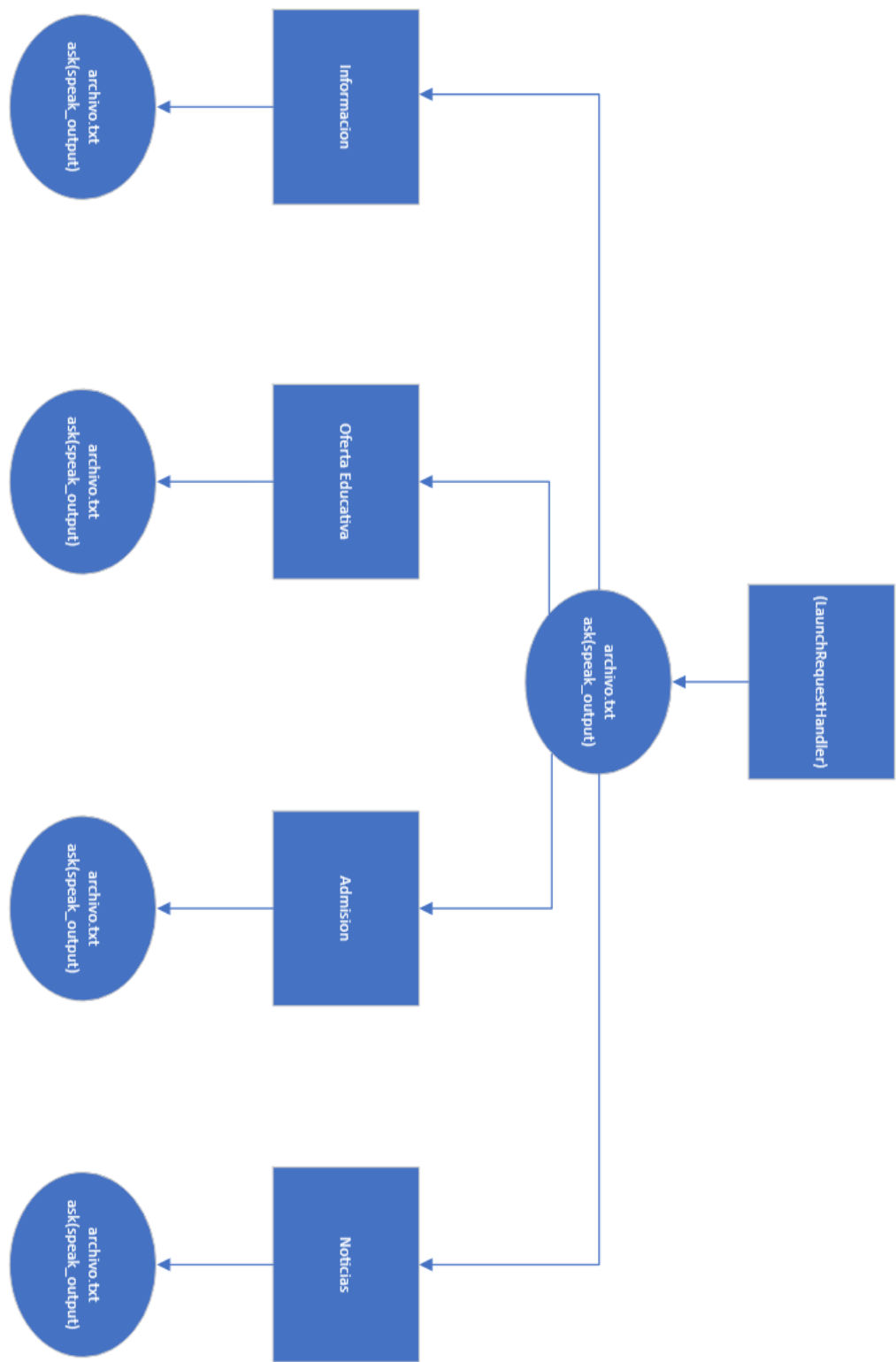
- Obtener actualizaciones sobre eventos escolares.
- Consultar logros y actividades estudiantiles.
- Solicitar noticias institucionales y anuncios.

4. Información:

- Obtener información sobre el centro.
- Consultar los programas del centro.

Diagrama de flujo de la conversación:

Este diagrama proporciona una representación visual de cómo podría desarrollarse una conversación entre el usuario y la habilidad para obtener información sobre el IES Ángel Corella.



Programación:

Implementación de la lógica de negocio:

Para la implementación de la lógica de negocio de la habilidad, utilizaremos el código previamente desarrollado como base. Este código proporciona las estructuras y funcionalidades necesarias para manejar las solicitudes del usuario y generar respuestas adecuadas. En esta etapa, nos enfocaremos en asegurar que la lógica de negocio esté correctamente integrada y operativa.

Manejo de Intents y Slots:

El manejo adecuado de los intents y slots definidos en el modelo de interacción es esencial para garantizar una experiencia fluida y precisa para el usuario. Cada intent representa una acción que el usuario desea realizar, mientras que los slots proporcionan información adicional necesaria para completar esa acción. Es crucial que la habilidad pueda identificar y procesar correctamente estos intents y slots para ofrecer respuestas relevantes y útiles.

Durante la programación, nos aseguraremos de manejar cada intent definido en el modelo de interacción, así como cualquier slot asociado que pueda ser requerido para cumplir con la solicitud del usuario. Esto incluirá la lectura de archivos de texto, la generación de respuestas dinámicas basadas en la entrada del usuario y la gestión de posibles escenarios de error o falta de información.

La correcta implementación de la lógica de negocio y el manejo de intents y slots son aspectos críticos para el funcionamiento efectivo de la habilidad, y nos aseguraremos de abordarlos de manera meticulosa y exhaustiva durante esta fase de programación.

Pruebas:

Pruebas Unitarias:

Las pruebas unitarias son esenciales para garantizar que cada componente de la habilidad funcione correctamente de forma individual. Durante esta fase, se llevarán a cabo pruebas exhaustivas para cada función y método implementado en la habilidad. Esto incluye la verificación de la lógica de negocio, la manipulación adecuada de datos y archivos, así como la respuesta esperada a diferentes escenarios de entrada. Cada prueba unitaria se enfocará en un aspecto específico de la funcionalidad de la habilidad, asegurando su correctitud y robustez.

Pruebas de Integración:

Las pruebas de integración son cruciales para validar que todos los componentes de la habilidad funcionen de manera cohesiva y sin problemas cuando se combinan. Durante esta fase, se simularán interacciones completas entre la habilidad y Alexa para asegurarse de que todas las partes trabajen juntas de

manera efectiva. Se verificará la comunicación adecuada entre los distintos handlers y la correcta transición entre las diferentes etapas de la interacción. Las pruebas de integración son clave para identificar posibles puntos de fallo en el flujo de conversación y garantizar una experiencia de usuario consistente.

Pruebas de aceptación de usuario:

Las pruebas de aceptación del usuario son fundamentales para evaluar la experiencia de uso de la habilidad desde la perspectiva del usuario final. Durante esta fase, se solicitará a usuarios reales que prueben la habilidad y proporcionen retroalimentación sobre su facilidad de uso, utilidad y cualquier problema encontrado. Esta retroalimentación será invaluable para identificar áreas de mejora y realizar ajustes finales antes del lanzamiento oficial de la habilidad. Las pruebas de aceptación del usuario garantizarán que la habilidad cumpla con las expectativas y necesidades de su público objetivo, brindando una experiencia satisfactoria y efectiva.

7. ¿Qué ocurre cuando se habla con Alexa?

Cuando un usuario interactúa con Alexa, se desencadena una serie de procesos técnicos que permiten que la asistente virtual comprenda y responda adecuadamente a las solicitudes. Este flujo de interacción se puede dividir en varias etapas clave:

1. Activación de Alexa

La interacción con Alexa comienza con la detección de la palabra de activación o "wake word", que por defecto es "Alexa". Esta palabra sirve como señal para que el dispositivo Alexa pase de un estado de escucha pasiva a un estado activo de procesamiento de comandos.

2. Reconocimiento automático del habla (ASR)

Una vez activada, Alexa graba la voz del usuario y envía la grabación a los servidores de Amazon. Aquí, el sistema de Reconocimiento Automático del Habla (ASR, por sus siglas en inglés) convierte la señal de audio en texto. Este proceso implica el uso de modelos avanzados de machine learning que pueden interpretar y transcribir el habla humana con alta precisión.

3. Comprensión del lenguaje natural (NLU)

El texto generado por el ASR se pasa al sistema de Comprensión del Lenguaje Natural (NLU, por sus siglas en inglés). El NLU analiza el texto para extraer la intención del usuario (intent) y cualquier información relevante (slots). Por ejemplo, si el usuario dice "Alexa, ¿cuál es la oferta educativa del IES Ángel Corella?", el NLU identificará que el intent es **OfertaEducativaIntent** y no hay slots adicionales en este caso.

4. Enrutamiento de la solicitud

Una vez que la intención y los slots se han identificado, Alexa enruta la solicitud al backend de la habilidad correspondiente. En el caso de las habilidades personalizadas, esta solicitud se envía a una función de AWS Lambda, que ejecuta la lógica de negocio definida para manejar la solicitud.

5. Procesamiento en AWS Lambda

La función Lambda recibe la solicitud con el intent y los slots. Aquí, se ejecuta el código necesario para generar una respuesta adecuada. Por ejemplo, si el intent es **OfertaEducativaIntent**, la función Lambda podría buscar en una base de datos la información sobre los cursos y programas ofrecidos por el IES Ángel Corella y formar una respuesta.

6. Generación de la respuesta

La función Lambda genera una respuesta en formato JSON que Alexa puede interpretar. Esta respuesta incluye el texto que Alexa dirá al usuario y puede incluir otras instrucciones, como mostrar contenido en un dispositivo Echo Show o reproducir un audio específico.

7. Síntesis de voz

El JSON de respuesta se envía de vuelta a los servidores de Alexa, donde el sistema de Síntesis de Voz (TTS, por sus siglas en inglés) convierte el texto de respuesta en habla. Alexa utiliza modelos de síntesis de voz para producir un discurso natural y fluido.

8. Respuesta al usuario

Finalmente, el dispositivo Alexa reproduce la respuesta sintetizada, brindando al usuario la información o realizando la acción solicitada. Este proceso ocurre en cuestión de segundos, proporcionando una experiencia interactiva y en tiempo real.

Ejemplo de Interacción

Para ilustrar este proceso, consideremos el siguiente ejemplo:

1. El usuario dice: "Alexa, ¿qué cursos ofrece el IES Ángel Corella?"
2. Alexa detecta la palabra de activación "Alexa" y comienza a grabar la solicitud.
3. El ASR convierte el audio en el texto: "¿qué cursos ofrece el IES Ángel Corella?"
4. El NLU identifica el **OfertaEducativaIntent**.
5. La solicitud se envía a la función Lambda correspondiente.
6. La función Lambda procesa la solicitud y devuelve una respuesta JSON con la información sobre los cursos.

7. Alexa convierte el texto de respuesta en habla.
8. Alexa responde al usuario: "El IES Ángel Corella ofrece cursos de Educación Secundaria Obligatoria, Bachillerato y Formación Profesional".

8. Desarrollo

Conceptos fundamentales y herramientas utilizadas:

Antes de sumergirnos en el desarrollo, es fundamental entender algunos conceptos clave:

1. **Intents o Intenciones:** Estas representan las acciones que los usuarios desean realizar con la habilidad. Cada intent corresponde a una tarea específica que la habilidad puede llevar a cabo.
2. **Utterances o Enunciados:** Son las frases que los usuarios pronuncian para interactuar con la habilidad y ejecutar una determinada intención.
3. **Slots o Variables:** Son los elementos variables dentro de un enunciado, como nombres, fechas o números, que la habilidad debe reconocer y procesar para brindar una respuesta precisa. Los slots son importantes para capturar información específica proporcionada por el usuario.

Las habilidades se basan en un robusto servicio de voz de Alexa en la nube, que utiliza dos importantes etapas de machine learning durante la interacción del usuario:

1. **Reconocimiento Automatizado del Habla (ASR):** En esta etapa, lo que el usuario dice se convierte en texto para su procesamiento.
2. **Comprendimiento del Lenguaje Natural (NLU):** Aquí, se asigna un significado a la frase convertida en texto.

Cuando se interactúa con Alexa, es crucial comenzar con la palabra de activación o "wake word", que normalmente es "Alexa", aunque puede ser configurable. Luego viene la palabra de lanzamiento, donde se indica la acción deseada, como "Abre".

El Invocation Name es cómo los usuarios identificarán y llamarán a nuestra habilidad, por lo que es recomendable decidirlo con cuidado. Después, el usuario puede proporcionar más información a través de "Utterances" o frases específicas.

Alexa, abre ies ángel corella

wake word launch Invocation name

Al crear una habilidad, debemos definir un conjunto de "Utterances" que se asignen a un intent específico. Es importante tener en cuenta que cada vez que definimos un conjunto de "Utterances" para un intent, estamos proporcionando material de entrenamiento, por lo que no es necesario contemplar todas las posibles variaciones que un usuario podría decir con respecto a ese intent.

Además, al trabajar con slots, es esencial considerar qué tipos de información variable necesitamos capturar de los usuarios. Por ejemplo, si nuestra habilidad tiene la intención de reservar una cita, necesitaremos slots para capturar la fecha, la hora y el motivo de la cita.

Para el desarrollo de la habilidad, se emplearon diversas herramientas y tecnologías:

- **SDK de Alexa Skills Kit (ASK):** Se utilizó el SDK de ASK para Python, proporcionado por Amazon, que facilita la creación de habilidades para dispositivos Alexa. Este SDK simplifica la interacción con la API de Alexa y ofrece funcionalidades predefinidas para manejar solicitudes y respuestas.
- **Lenguaje de Programación Python:** Python fue el lenguaje de programación principal utilizado en el desarrollo de la habilidad. Python es conocido por su simplicidad, legibilidad y versatilidad, lo que lo hace ideal para proyectos de desarrollo rápido como este. Además, la amplia disponibilidad de bibliotecas y frameworks en Python proporciona una variedad de herramientas para implementar diferentes aspectos de la habilidad de forma eficiente.
- **AWS Lambda:** Para manejar la lógica del backend de la habilidad, se utilizó AWS Lambda. Lambda permite ejecutar código sin necesidad de gestionar servidores, proporcionando un entorno escalable y de alta disponibilidad. Al emplear Lambda, se pueden desarrollar funciones que respondan a eventos específicos de la habilidad, como el procesamiento de intents y la gestión de slots. Además, la integración con otros servicios de AWS facilita la implementación de funcionalidades adicionales, como el almacenamiento de datos y la autenticación de usuarios.

Al emplear el SDK de ASK junto con Python y AWS Lambda, se pudo desarrollar una habilidad interactiva y fácil de usar para dispositivos Alexa, aprovechando las funcionalidades avanzadas de la plataforma y brindando una experiencia de usuario fluida y eficiente para los usuarios del IES Ángel Corella.

Con estos conceptos claros, estamos listos para iniciar el desarrollo de nuestra habilidad en Alexa Developer Console.

El desarrollo del código se llevó a cabo utilizando la herramienta web Alexa Developer de Amazon (disponible en <https://developer.amazon.com/>). Para

acceder a esta plataforma, se requiere autenticación mediante una cuenta de Amazon, la cual puede ser creada en caso de no poseer una previamente.

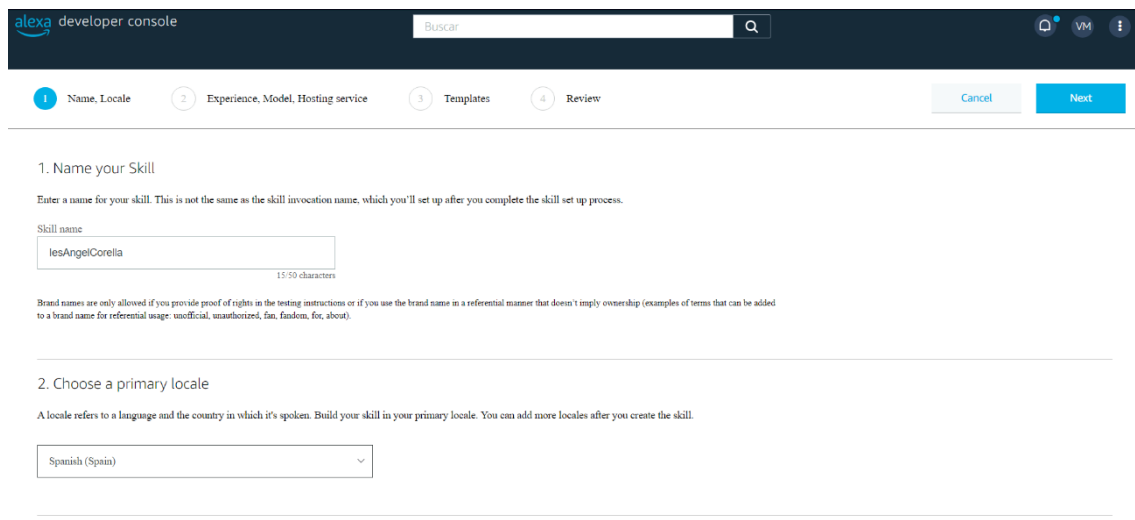
Una vez dentro de la plataforma, se observa un menú que contiene diversas opciones. Se recomienda dirigirse a la sección denominada "Alexa Skill Kits", la cual corresponde al entorno de desarrollo de habilidades para el asistente virtual Alexa.

Para iniciar el proceso de creación de la habilidad, se debe seleccionar la opción "Create Skill" o su equivalente en el menú correspondiente. Esto permite acceder a la interfaz de desarrollo donde se configuran y programan las funcionalidades de la habilidad deseada.

Creación de la habilidad:

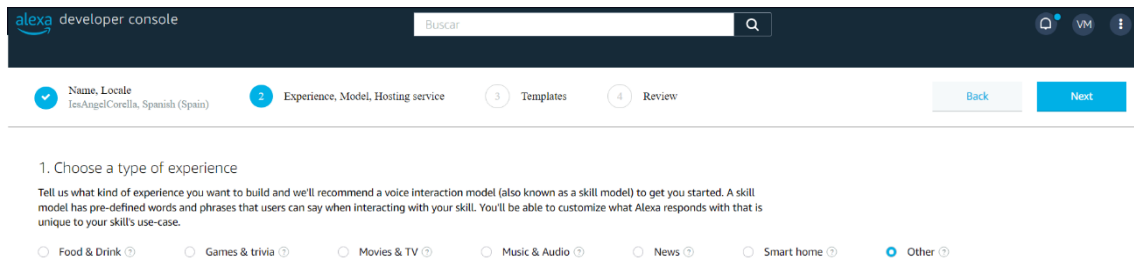
Para crear la habilidad, primero debemos definir una serie de opciones en el menú proporcionado por la herramienta.

1. **Nombre y lenguaje:** El primer paso es definir el nombre de la habilidad y el idioma en el que operará. El nombre es crucial, ya que es como los usuarios identificarán la habilidad. El lenguaje puede ser editado posteriormente si es necesario.



The screenshot shows the 'Name your Skill' step in the Alexa Developer Console. The interface includes a progress bar at the top with four steps: 1. Name, Locale (active), 2. Experience, Model, Hosting service, 3. Templates, and 4. Review. Below the progress bar, the text reads: '1. Name your Skill. Enter a name for your skill. This is not the same as the skill invocation name, which you'll set up after you complete the skill set up process.' A text input field for 'Skill name' contains 'lesAngelCorella' and shows a character count of '15/50 characters'. Below this, a note states: 'Brand names are only allowed if you provide proof of rights in the testing instructions or if you use the brand name in a referential manner that doesn't imply ownership (examples of terms that can be added to a brand name for referential usage: unofficial, unauthorized, fan, fandom, for, about)'. The next step is '2. Choose a primary locale', with a note: 'A locale refers to a language and the country in which it's spoken. Build your skill in your primary locale. You can add more locales after you create the skill.' A dropdown menu for 'Primary locale' is set to 'Spanish (Spain)'.

- 2. Tipo de experiencia:** A continuación, se debe seleccionar el tipo de experiencia que la habilidad ofrecerá. Para este proyecto en particular, seleccionamos "Other", ya que no se ajusta a ninguna otra categoría predefinida.



alex developer console

Buscar

1 Name, Locale
IesAngelCorella, Spanish (Spain)

2 Experience, Model, Hosting service

3 Templates

4 Review

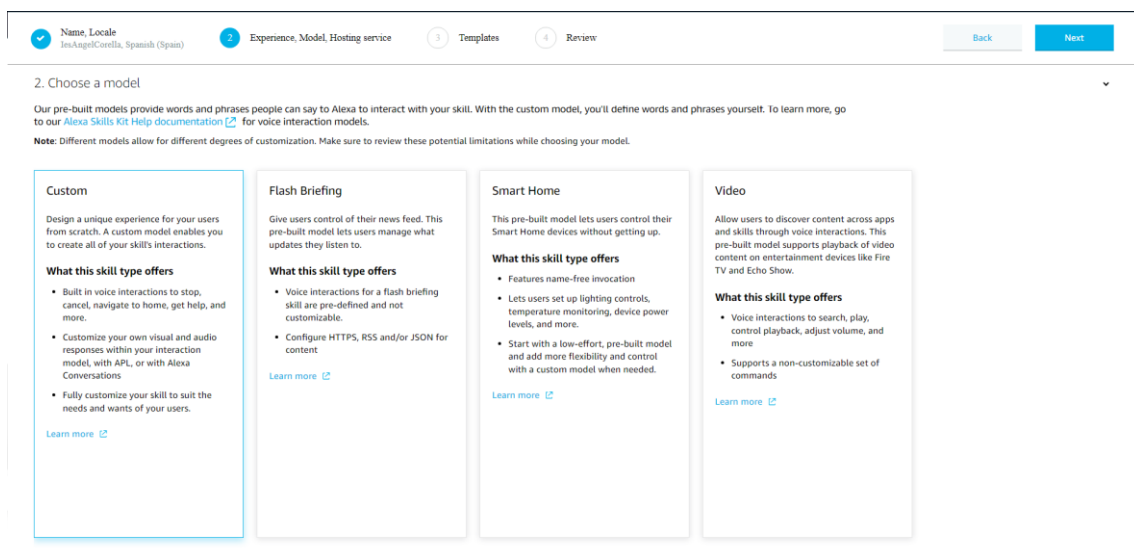
Back Next

1. Choose a type of experience

Tell us what kind of experience you want to build and we'll recommend a voice interaction model (also known as a skill model) to get you started. A skill model has pre-defined words and phrases that users can say when interacting with your skill. You'll be able to customize what Alexa responds with that is unique to your skill's use-case.

☐ Food & Drink ☐ Games & trivia ☐ Movies & TV ☐ Music & Audio ☐ News ☐ Smart home ☒ Other

- 3. Modelo de habilidad:** Luego, elegimos el modelo de habilidad a utilizar. En este caso, optamos por "Custom", ya que somos nosotros quienes definiremos las palabras y frases que la habilidad entenderá.



1 Name, Locale
IesAngelCorella, Spanish (Spain)

2 Experience, Model, Hosting service

3 Templates

4 Review

Back Next

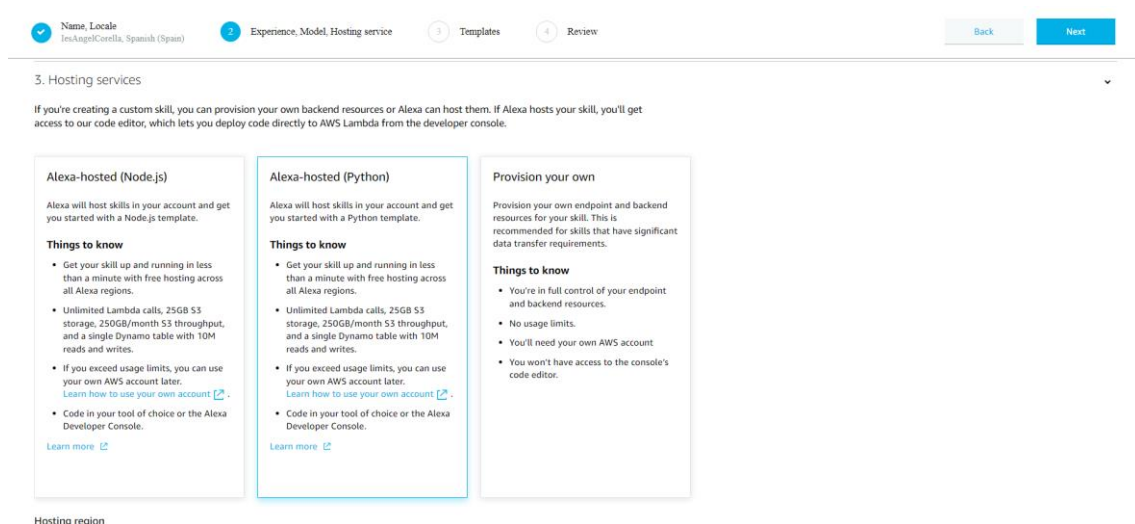
2. Choose a model

Our pre-built models provide words and phrases people can say to Alexa to interact with your skill. With the custom model, you'll define words and phrases yourself. To learn more, go to our [Alexa Skills Kit Help documentation](#) for voice interaction models.

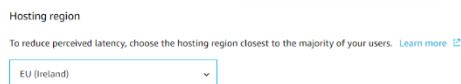
Note: Different models allow for different degrees of customization. Make sure to review these potential limitations while choosing your model.

Custom	Flash Briefing	Smart Home	Video
<p>Design a unique experience for your users from scratch. A custom model enables you to create all of your skill's interactions.</p> <p>What this skill type offers</p> <ul style="list-style-type: none">Built in voice interactions to stop, cancel, navigate to home, get help, and more.Customize your own visual and audio responses within your interaction model, with APL or with Alexa ConversationsFully customize your skill to suit the needs and wants of your users. <p>Learn more</p>	<p>Give users control of their news feed. This pre-built model lets users manage what updates they listen to.</p> <p>What this skill type offers</p> <ul style="list-style-type: none">Voice interactions for a flash briefing skill are pre-defined and not customizable.Configure HTTPS, RSS and/or JSON for content <p>Learn more</p>	<p>This pre-built model lets users control their Smart Home devices without getting up.</p> <p>What this skill type offers</p> <ul style="list-style-type: none">Features name-free invocationLets users set up lighting controls, temperature monitoring, device power levels, and more.Start with a low-effort, pre-built model and add more flexibility and control with a custom model when needed. <p>Learn more</p>	<p>Allow users to discover content across apps and skills through voice interactions. This pre-built model supports playback of video content on entertainment devices like Fire TV and Echo Show.</p> <p>What this skill type offers</p> <ul style="list-style-type: none">Voice interactions to search, play, control playback, adjust volume, and moreSupports a non-customizable set of commands <p>Learn more</p>

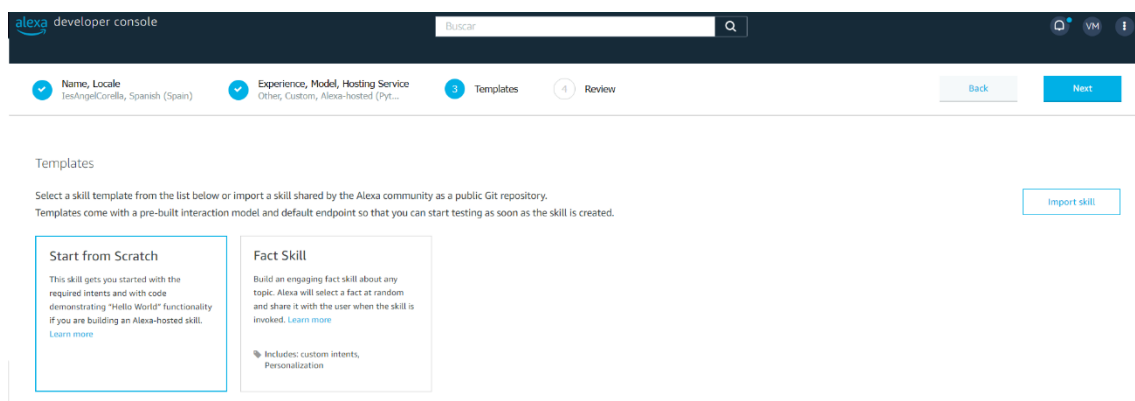
4. **Lenguaje de programación y hosting:** Es importante seleccionar el lenguaje de programación que se utilizará para desarrollar la habilidad. Además, se puede optar por alojar la habilidad en Amazon Web Services (AWS). Para este proyecto, se seleccionó Python como lenguaje de programación.



5. **Región de hosteo:** Se debe elegir la región en la que la habilidad será alojada. En este caso, seleccionamos "EU Irlanda" para estar más cerca del servidor y reducir los tiempos de respuesta.



6. **Plantilla de inicio:** A continuación, seleccionamos la plantilla desde la cual comenzar. En este caso, elegimos "Start From Scratch" para tener un lienzo en blanco y poder desarrollar la habilidad según nuestras necesidades.



7. **Revisión y creación:** Por último, se presenta una visión general de las opciones seleccionadas. Es importante revisar esta información para asegurarse de que todo esté correcto. Una vez confirmado, se hace clic en "Create Skill". Este proceso puede tardar unos minutos en completarse.

The screenshot shows the 'Review' step in the Alexa Developer Console. At the top, there are four progress indicators: 'Name, Locale' (checked), 'Experience, Model, Hosting Service' (checked), 'Templates' (checked), and 'Review' (active). Below the indicators, there are two buttons: 'Back' and 'Create Skill'. The main content area is titled 'Review your selections' and contains three sections: 1. Name, Locale: Skill name 'IesAngelCorella', Primary locale 'Spanish (Spain)'. 2. Experience, Model, Hosting Service: A warning message 'Make sure you have chosen the right model and hosting region. They can't be changed after you have created the skill.' Below this, a table shows 'Type of experience' as 'Other', 'Model' as 'Custom', 'Sync locales' as 'Disabled', 'Hosting service' as 'Alexa-hosted (Python)', and 'Hosting region' as 'EU (Ireland)'. 3. Templates: 'Start from Scratch'.

Review your selections		
1. Name, Locale Edit		
Skill name	Primary locale	
IesAngelCorella	Spanish (Spain)	
2. Experience, Model, Hosting Service Edit		
⚠ Make sure you have chosen the right model and hosting region. They can't be changed after you have created the skill.		
Type of experience	Model	Sync locales
Other	Custom	Disabled
Hosting service	Hosting region	
Alexa-hosted (Python)	EU (Ireland)	
3. Templates Edit		
Templates		
Start from Scratch		

Siguiendo estos pasos, se inicia el proceso de creación de la habilidad en la plataforma de Alexa Developer.

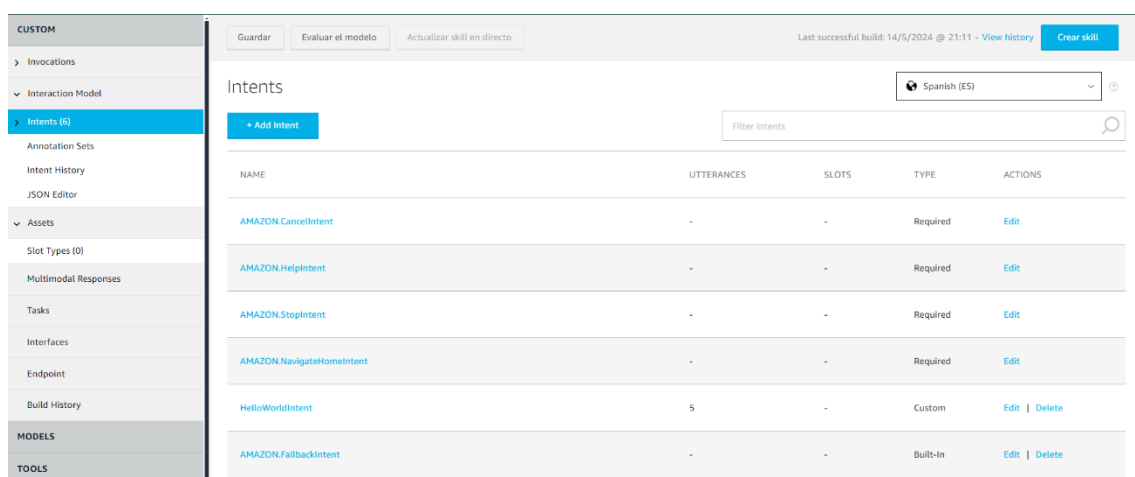
Dentro de Alexa Developer, nos encontramos con varias pestañas. Las principales son las siguientes:

1. **Build:** Esta es la interfaz gráfica donde configuramos nuestra habilidad. Aquí podemos definir los intents, slots y configurar la lógica de la habilidad.
2. **Code:** En esta pestaña, tenemos acceso al código que reside en AWS Lambda. Aquí implementamos la lógica de la habilidad, como manejar las solicitudes del usuario y generar respuestas.
3. **Test:** En esta pestaña podremos verificar el funcionamiento de nuestra habilidad. Se puede interactuar con la habilidad como lo haría un usuario real, a través del asistente de voz o mediante texto, dependiendo de la configuración de la pestaña. Se puede identificar cualquier error o fallo en la lógica de la habilidad antes de publicarla para su uso general.

Dentro del Alexa Developer Console, encontraremos un menú en la parte izquierda que contiene diversas opciones de esta herramienta. Con la creación de la habilidad, se han creado unos intents predeterminados que podemos editar a nuestro gusto añadiendo las frases clave para usar cada uno de ellos. Los intents son:

- **AMAZON.CancelIntent:** Se activa cuando el usuario desea cancelar la acción actual.
- **AMAZON.HelpIntent:** Se activa cuando el usuario solicita ayuda.
- **AMAZON.StopIntent:** Se activa cuando el usuario desea detener la acción actual y salir de la habilidad.
- **AMAZON.NavigateHomeIntent:** Se activa cuando el usuario desea volver al inicio de la habilidad.

También se proporciona un intent por defecto con algunos ejemplos para seguir. Podemos modificar el nombre y las frases asociadas para que nuestra habilidad responda adecuadamente a las solicitudes de los usuarios.

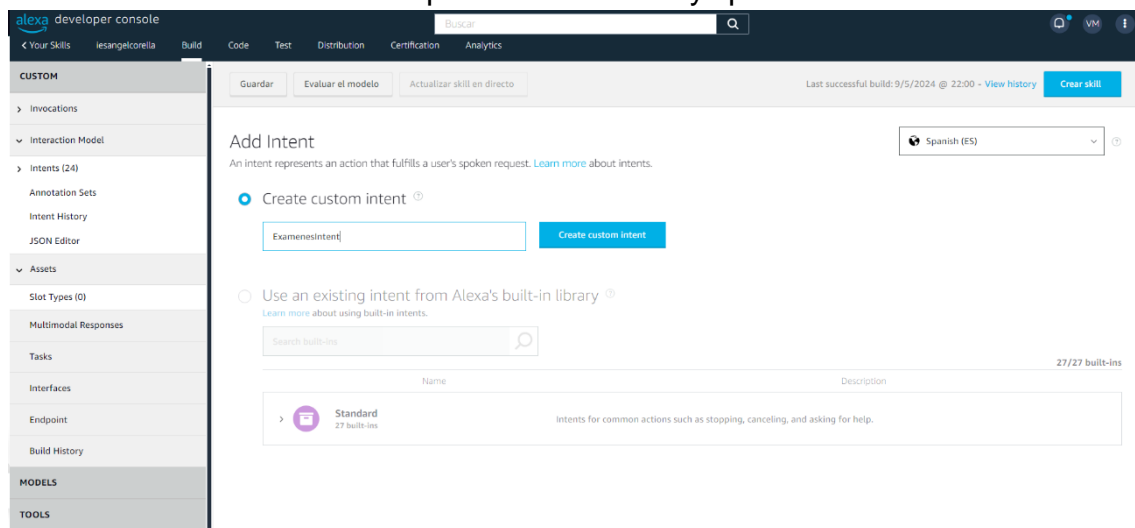


The screenshot shows the 'Intents' section of the Alexa Developer Console. On the left, a sidebar lists navigation options: CUSTOM, Invocations, Interaction Model, Intents (6), Annotation Sets, Intent History, JSON Editor, Assets, Slot Types (0), Multimodal Responses, Tasks, Interfaces, Endpoint, Build History, MODELS, and TOOLS. The 'Intents (6)' section is selected. The main area displays a table of intents for the 'Spanish (ES)' locale. The table has columns for NAME, LUTTERANCES, SLOTS, TYPE, and ACTIONS. The intents listed are AMAZON.CancelIntent, AMAZON.HelpIntent, AMAZON.StopIntent, AMAZON.NavigateHomeIntent, HelloWorldIntent, and AMAZON.FallbackIntent. Each intent has an 'Edit' or 'Delete' link next to it.

NAME	LUTTERANCES	SLOTS	TYPE	ACTIONS
AMAZON.CancelIntent	-	-	Required	Edit
AMAZON.HelpIntent	-	-	Required	Edit
AMAZON.StopIntent	-	-	Required	Edit
AMAZON.NavigateHomeIntent	-	-	Required	Edit
HelloWorldIntent	5	-	Custom	Edit Delete
AMAZON.FallbackIntent	-	-	Built-in	Edit Delete

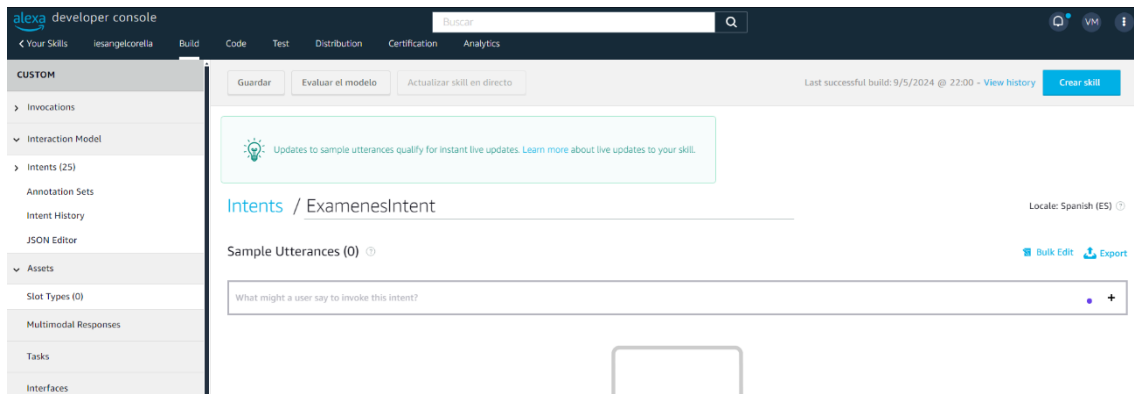
Para la creación de intents, contamos con dos opciones: la interfaz gráfica y el JSON Editor.

En la interfaz gráfica, podemos definir la estructura de los intents y agregar las frases clave que los activarán. Para hacerlo, accedemos al menú de la izquierda al apartado "Interaction Model", donde seleccionamos "Intents". Aquí, encontraremos los intents predeterminados y podemos crear uno nuevo



The screenshot shows the 'Add Intent' form in the Alexa Developer Console. The sidebar is the same as in the previous screenshot, but 'Intents (24)' is selected. The main area is titled 'Add Intent' and includes a description: 'An intent represents an action that fulfills a user's spoken request. Learn more about intents.' There are two radio buttons: 'Create custom intent' (selected) and 'Use an existing intent from Alexa's built-in library'. The 'Create custom intent' option has a text input field containing 'ExamenIntent' and a 'Create custom intent' button. The 'Use an existing intent from Alexa's built-in library' option has a search input field. Below the search field, there is a list of built-in intents. The first one is 'Standard' with 27 built-ins, described as 'Intents for common actions such as stopping, canceling, and asking for help.'

haciendo clic en “+Add Intent”. Es importante asignarle un nombre sin espacios, idealmente terminado con la palabra "Intent" para facilitar su referencia en el código. Una vez creado, definimos las Utterances del Intent.



Por otro lado, en el JSON Editor, también ubicado en el apartado “Interaction Model”, podremos editar directamente el archivo JSON que contiene los intents y las muestras de frases (utterances). Aquí, debemos seguir la estructura definida y tenemos la opción de importar un JSON creado en otra herramienta como Visual Studio Code. Además, en el campo "invocationName", debemos definir el nombre con el cual los usuarios invocarán nuestra habilidad.

```
1 {
2   "interactionModel": {
3     "languageModel": {
4       "invocationName": "change me",
5       "intents": [
6         {
7           "name": "AMAZON.CancelIntent",
8           "samples": []
9         },
10        {
11          "name": "AMAZON.HelpIntent",
12          "samples": []
13        },
14        {
15          "name": "AMAZON.StopIntent",
16          "samples": []
17        },
18        {
19          "name": "AMAZON.NavigateHomeIntent",
20          "samples": []
21        },
22        {
23          "name": "HelloWorldIntent",
24          "slots": [],
25          "samples": [
26            "hola",
27            "como estás",
28            "di hola mundo",
29            "di hola",
30            "hola mundo"
31          ]
32        },
33        {
34          "name": "AMAZON.FallbackIntent",
35          "samples": []
36        }
37      ],
38      "types": []
39    }
40  }
```

En la pestaña "Code" del Alexa Developer Console, podemos visualizar el código predefinido junto con ejemplos que nos guían en el desarrollo de nuestra

habilidad. Este código está escrito en Python y utiliza el Alexa Skills Kit SDK para Python.

En el proceso de desarrollo de una habilidad de Alexa en Python, nos encontramos con varios componentes esenciales que nos ayudan a construir una experiencia de usuario fluida y eficiente. El archivo **lambda_function.py** el código comienza importando las bibliotecas necesarias y configurando un registro para mensajes de información. Luego, define varias clases que manejan diferentes tipos de solicitudes que la habilidad puede recibir, como el lanzamiento de la habilidad, la intención de saludar, la solicitud de ayuda y la solicitud de cancelación o finalización.

Cada clase implementa dos métodos principales: **can_handle()** y **handle()**. El método **can_handle()** determina si la clase puede manejar la solicitud entrante, mientras que el método **handle()** contiene la lógica para manejar la solicitud y generar una respuesta adecuada.

Además, se incluyen clases para manejar solicitudes de fin de sesión, solicitudes de reflexión de intención y excepciones genéricas. Estas clases garantizan un manejo adecuado de diferentes situaciones y errores que pueden surgir durante la interacción con la habilidad.

Finalmente, el objeto **SkillBuilder** se utiliza para agregar todos los controladores de solicitud y excepción, y se crea un controlador lambda que actúa como punto de entrada para la habilidad.

Por otro lado, en el archivo **utils.py**, se proporciona una función llamada **create_presigned_url** que genera una URL firmada para compartir un objeto en un bucket de Amazon S3. Esta función utiliza la biblioteca **boto3** para interactuar con AWS S3 y generar la URL firmada.

Finalmente, en el archivo **requirements.txt**, se especifican las dependencias del proyecto, incluyendo las versiones de las bibliotecas **boto3** y **ask-sdk-core** que se requieren para ejecutar la habilidad de Alexa.

En conjunto, estos archivos proporcionan una base sólida para el desarrollo de una habilidad de Alexa en Python, ofreciendo flexibilidad para personalizar y ampliar la funcionalidad según las necesidades específicas del proyecto.

Para comenzar el desarrollo del proyecto en la herramienta Alexa Developer, primero exploramos las funciones predeterminadas que ofrece la plataforma. Luego, nos sumergimos en el proceso de creación de intents, que son las acciones o intenciones que nuestra habilidad de Alexa puede reconocer y responder.

Gestión de intenciones y clases en la implementación de la habilidad

En nuestro proyecto, hemos definido un total de 24 intenciones, cada uno con sus correspondientes utterances, que son las frases o expresiones que los usuarios utilizarán para interactuar con la habilidad. Estas intenciones se dividen en dos categorías:

1. **Intenciones por defecto:** Estos son provistos automáticamente por la plataforma y están diseñados para manejar acciones comunes, como cancelar, pedir ayuda, detener la interacción y navegar de regreso al inicio. Los intents por defecto incluyen:
 - AMAZON.CancelIntent
 - AMAZON.HelpIntent
 - AMAZON.StopIntent
 - AMAZON.NavigateHomeIntent
2. **Intenciones creados:** Además de las intenciones predeterminados, hemos creado una serie de intenciones personalizadas para satisfacer las necesidades específicas de nuestra habilidad. Estas intenciones están diseñados para capturar las distintas acciones o consultas que los usuarios pueden realizar. Las intenciones creadas en nuestro proyecto son:
 - lesAngelCorella
 - InformacionIntent
 - OfertaEducativaIntent
 - EducacionSecundariaObligatoriaIntent
 - BachilleratoIntent
 - FormacionProfesionalIntent
 - FormacionProfesionalBasicaIntent
 - FormacionProfesionalMediaIntent
 - FormacionProfesionalSuperiorIntent
 - AdmisionIntent
 - NoticiasIntent
 - FPSEnseAnimSociodeportivaIntent
 - FPSAdminFinanzasIntent
 - FPSDesarrolloWebIntent

- FPSDesarrolloMultilIntent
- FPMSMicroRedesIntent
- FPMAAdminGestionIntent
- FPMInstalacionesElectricasIntent
- FPBElectricidadIntent
- FPBAdminGestionIntent

Estas intenciones abarcan una variedad de temas y acciones relevantes para nuestra habilidad, permitiendo una interacción completa y satisfactoria con los usuarios. Cada intención se desarrolla cuidadosamente para garantizar una comprensión precisa de las solicitudes de los usuarios y proporcionar respuestas relevantes y útiles.

Para la implementación del código, se han desarrollado 26 clases individuales, cada una correspondiente a un intent específico previamente definido. Estas clases están diseñadas para manejar las solicitudes asociadas a cada intención de manera independiente. Cuando una de estas clases es invocada por su respectivo intent, muestra el contenido de un archivo de texto correspondiente, lo que facilita enormemente la edición de los textos que serán pronunciados por Alexa a través de voz.

Las clases creadas son las siguientes:

- **LaunchRequestHandler:** Maneja las solicitudes de inicio de la habilidad.
- **InformacionIntentHandler:** Maneja las solicitudes relacionadas con la información acerca del centro.
- **AdmisionIntentHandler:** Maneja las solicitudes relacionadas con la admisión.
- **NoticiasIntentHandler:** Maneja las solicitudes de noticias.
- **OfertaEducativaIntentHandler:** Maneja las solicitudes relacionadas con la oferta educativa.
- **EducacionSecundariaObligatoriaIntentHandler:** Maneja las solicitudes relacionadas con la educación secundaria obligatoria.
- **BachilleratoIntentHandler:** Maneja las solicitudes relacionadas con el bachillerato.
- **FormacionProfesionalIntentHandler:** Maneja las solicitudes relacionadas con la formación profesional.
- **FormacionProfesionalBasicaIntentHandler:** Maneja las solicitudes relacionadas con la formación profesional básica.

- **FPBElectricidadIntentHandler**: Maneja las solicitudes relacionadas con la formación profesional en electricidad.
- **FPBAdminGestionIntentHandler**: Maneja las solicitudes relacionadas con la formación profesional en administración y gestión.
- **FormacionProfesionalMediaIntentHandler**: Maneja las solicitudes relacionadas con la formación profesional media.
- **FPMMicroRedesIntentHandler**: Maneja las solicitudes relacionadas con la formación profesional en sistemas microinformáticos y redes.
- **FPMAdminGestionIntentHandler**: Maneja las solicitudes relacionadas con la formación profesional en administración y gestión.
- **FPMInstalacionesElectricasIntentHandler**: Maneja las solicitudes relacionadas con la formación profesional en instalaciones eléctricas.
- **FormacionProfesionalSuperiorIntentHandler**: Maneja las solicitudes relacionadas con la formación profesional superior.
- **FPSEnseAnimSociodeportivaIntentHandler**: Maneja las solicitudes relacionadas con la formación profesional en enseñanza y animación sociodeportiva.
- **FPSAdminFinanzasIntentHandler**: Maneja las solicitudes relacionadas con la formación profesional en administración y finanzas.
- **FPSDesarrolloWebIntentHandler**: Maneja las solicitudes relacionadas con la formación profesional en desarrollo de aplicaciones web.
- **FPSDesarrolloMultiIntentHandler**: Maneja las solicitudes relacionadas con la formación profesional en desarrollo de aplicaciones multiplataforma.

Además, se han editado las siguientes clases por defecto proporcionadas por el SDK de Alexa Skills Kit para Python:

- **HelpIntentHandler**: Maneja las solicitudes de ayuda.
- **CancelOrStopIntentHandler**: Maneja las solicitudes de cancelación o finalización.
- **FallbackIntentHandler**: Maneja las solicitudes cuando Alexa no puede entender la entrada del usuario.
- **SessionEndedRequestHandler**: Maneja las solicitudes de finalización de sesión.
- **IntentReflectorHandler**: Refleja el intent activado para pruebas y depuración del modelo de interacción.

- **CatchAllExceptionHandler:** Maneja excepciones genéricas para capturar cualquier error sintáctico o de enrutamiento.

Este enfoque modular proporciona una estructura organizada y fácilmente mantenible para el desarrollo de la habilidad de Alexa, asegurando que cada solicitud sea manejada de manera efectiva y que las respuestas proporcionadas sean relevantes y útiles para los usuarios.

Estructura de las clases de manejadores de intenciones:

En el contexto del desarrollo de habilidades para Alexa, la implementación de los manejadores de intents es esencial para definir cómo la habilidad responderá a las solicitudes de los usuarios. A continuación, se presenta una explicación detallada de la estructura y funcionalidad de las clases de manejadores de intenciones en el código de la habilidad.

Todas las clases definidas en el código siguen una estructura similar a la clase **OfertaEducativaIntentHandler**, proporcionando una organización coherente y modular del código para manejar diferentes tipos de solicitudes de los usuarios.

```
class OfertaEducativaIntentHandler(AbstractRequestHandler):  
    """Handler for Oferta Educativa Intent."""
```

Cada clase de manejador de intents contiene un método **can_handle()** que determina si el manejador puede procesar la solicitud actual del usuario. Este método utiliza la función **ask_utils.is_intent_name("NombreDelIntent")** para verificar si la solicitud del usuario coincide con el intent correspondiente. Si la solicitud se relaciona con el intent específico, el método devuelve **True**, lo que indica que el manejador es capaz de manejar la solicitud.

```
def can_handle(self, handler_input):  
    return ask_utils.is_intent_name("OfertaEducativaIntent")(handler_input)
```

El método **handle()** es el punto central de procesamiento de solicitudes dentro de cada clase de manejador de intents. Este método se encarga de procesar la solicitud del usuario asociada con el intent correspondiente. En el código proporcionado, este método generalmente intenta abrir y leer un archivo que contiene la información necesaria para responder a la solicitud del usuario. Si el archivo se encuentra y se puede leer correctamente, su contenido se asigna a una variable, que luego se utiliza para generar la respuesta de la habilidad. En caso de que el archivo no se encuentre, se genera un mensaje de error indicando que la información solicitada no está disponible.

```
def handle(self, handler_input):  
    try:  
        with open("ofertaeducativa.txt", "r", encoding="utf-8") as file:  
            speak_output = file.read()  
    except FileNotFoundError:  
        speak_output = "Lo siento, no pude encontrar información sobre eso. Por favor pruebe de nuevo."
```

La respuesta de la habilidad se construye utilizando el objeto **response_builder**, que se encarga de configurar la salida de voz y cualquier interacción adicional con el usuario. En cada clase de manejador de intents, este objeto se configura para proporcionar una respuesta adecuada basada en la solicitud del usuario y la información disponible.

```
return (  
    handler_input.response_builder  
        .speak(speak_output)  
        .ask(speak_output)  
        .response  
)
```

La implementación uniforme y coherente de todas las clases de manejadores de intents garantiza que la habilidad de voz proporcione respuestas precisas y útiles a una variedad de solicitudes de los usuarios, manteniendo al mismo tiempo un código limpio y fácil de mantener.

Lectura dinámica de archivos .txt

La habilidad "IES Ángel Corella" incorpora una funcionalidad crucial que le permite ofrecer contenido dinámico y actualizado a los usuarios, como noticias sobre la institución educativa o información sobre los diferentes programas académicos disponibles. Esta funcionalidad se logra mediante la lectura de archivos de texto que contienen estos datos, lo que permite una actualización sencilla y ágil del contenido sin necesidad de modificar el código fuente de la habilidad.

Implementación en el código:

El proceso de lectura de archivos de texto se gestiona de manera eficiente y estructurada en el código fuente de la habilidad. Primero, se define la estructura de los archivos de texto que contienen la información deseada, asegurando que estén formateados de manera coherente y sean fácilmente interpretables por la habilidad.

Luego, en el código, se implementa una función dedicada para abrir y leer estos archivos de texto cuando se solicita información específica por parte del usuario. Por ejemplo, cuando un usuario pregunta sobre la oferta educativa del centro, la habilidad accede al archivo correspondiente que contiene esta información y extrae el contenido relevante para proporcionar una respuesta precisa y detallada.

Manejo de la lectura de archivos:

Para garantizar un manejo adecuado de la lectura de archivos, se implementan mecanismos de manejo de errores y validación de datos. La habilidad verifica la

existencia y accesibilidad de los archivos antes de intentar leer su contenido, lo que ayuda a prevenir posibles fallos en tiempo de ejecución.

Además, se utilizan técnicas de procesamiento de texto para filtrar y formatear la información extraída de los archivos de manera que sea legible y comprensible para el usuario final. Esto puede incluir la eliminación de caracteres especiales, la división del texto en párrafos o secciones, y la inclusión de etiquetas de formato para mejorar la presentación en la respuesta vocal de la habilidad.

Beneficios y flexibilidad:

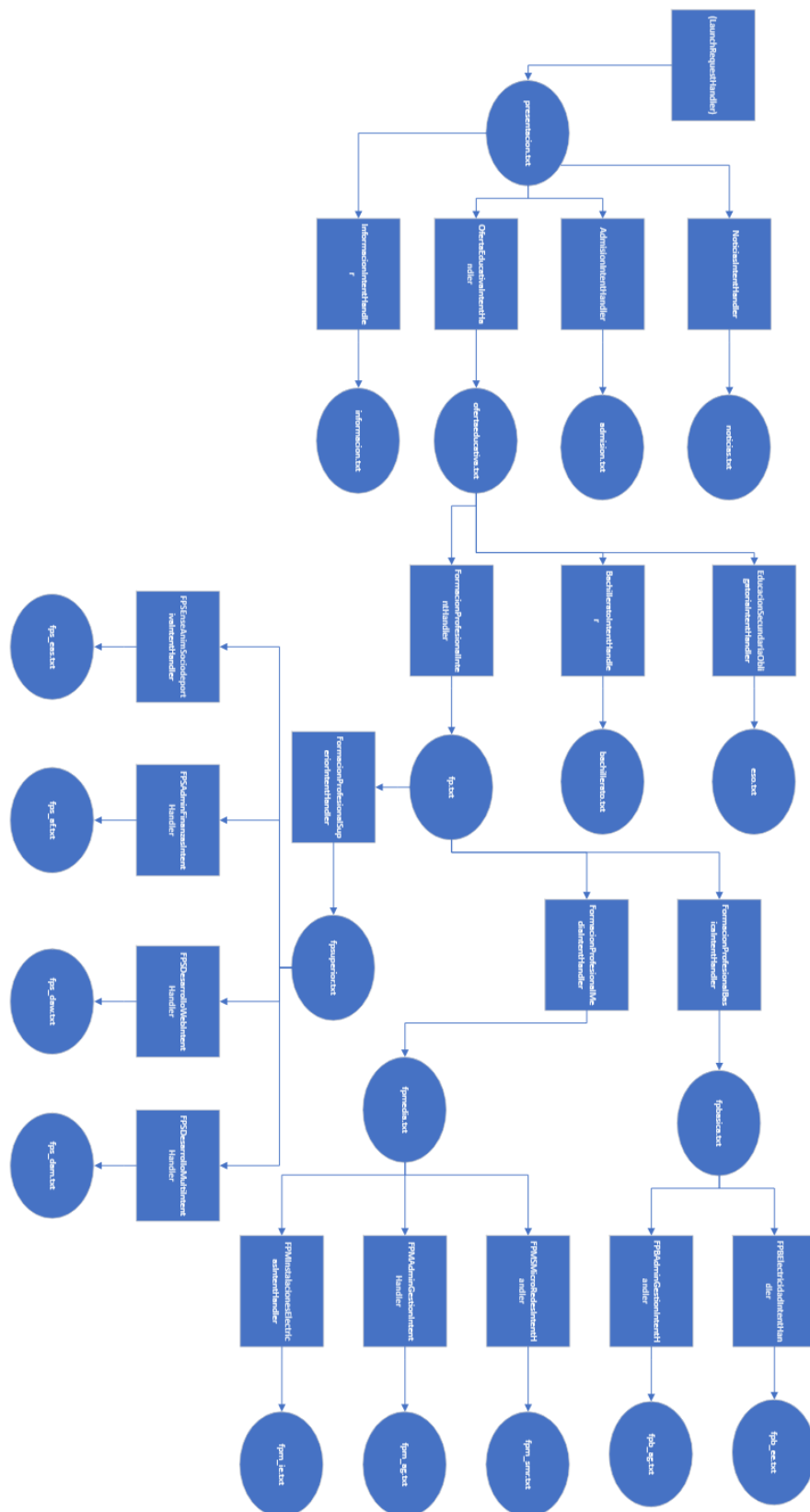
La lectura de archivos de texto ofrece varios beneficios significativos para la habilidad "IES Ángel Corella". En primer lugar, proporciona una manera eficiente de mantener y actualizar el contenido de la habilidad sin necesidad de modificar el código fuente, lo que permite una adaptación ágil a cambios en la información o eventos relevantes.

Además, esta funcionalidad ofrece una gran flexibilidad en cuanto al tipo y formato de contenido que se puede incorporar a la habilidad. Desde noticias y actualizaciones institucionales hasta descripciones detalladas de programas educativos, la lectura de archivos de texto permite una amplia variedad de información que puede ser ofrecida a los usuarios de manera dinámica y personalizada.

En resumen, la habilidad "IES Ángel Corella" utiliza la lectura de archivos de texto como una herramienta fundamental para ofrecer contenido actualizado y relevante a los usuarios, asegurando una experiencia de usuario enriquecedora y adaptada a sus necesidades educativas e informativas.

Una vez que hemos desarrollado la habilidad, podemos acceder a la pestaña "Distribution", completar cada uno de los apartados y publicar nuestra habilidad en Alexa para que todos los usuarios puedan acceder a ella.

9. Diagrama UML:



10. Guía de usuario

Este manual proporciona una guía detallada sobre cómo utilizar la habilidad IES Ángel Corella, una aplicación de voz diseñada para dispositivos Alexa que ofrece información sobre el IES Ángel Corella. La habilidad IES Ángel Corella ha sido desarrollada para proporcionar a los usuarios acceso rápido y sencillo a una variedad de detalles relacionados con el centro educativo.

Acceso a la habilidad

Para comenzar a utilizar la habilidad IES Ángel Corella, simplemente sigue estos pasos:

1. Enciende tu dispositivo Alexa y di "Abre IES Ángel Corella."
2. Alternativamente, puedes decir "Inicia IES Ángel Corella" o "Ábreme IES Ángel Corella" para iniciar la aplicación.

Funcionalidades Principales

1. Explorar Oferta Educativa

Los usuarios pueden explorar la amplia gama de programas educativos ofrecidos por el IES Ángel Corella. La habilidad IES Ángel Corella proporciona detalles sobre los cursos disponibles y las opciones académicas.

Ejemplos de comandos:

- "¿Qué cursos ofrece la escuela?"
- "Cuál es la oferta educativa."

2. Proceso de Admisión

La habilidad IES Ángel Corella ofrece orientación sobre el proceso de admisión al centro educativo. Los usuarios pueden obtener información sobre los requisitos de admisión y cómo presentar una solicitud.

Ejemplos de comandos:

- "Proceso de admisión."
- "Cuál es el proceso de admisión."

3. Noticias

Los usuarios pueden obtener información sobre las noticias y actualizaciones más recientes relacionadas con el IES Ángel Corella.

Ejemplos de comandos:

- "Dime qué noticias tenemos."
- "Noticias."

4. Información:

Los usuarios pueden solicitar información general sobre el IES Ángel Corella, incluyendo detalles sobre la historia del centro, su misión y visión.

Ejemplos de comandos:

- "Información."
- "Información acerca del centro."

Consejos para una mejor experiencia

Para una experiencia óptima con la habilidad IES Ángel Corella, se recomienda lo siguiente:

- Pronuncia claramente tus preguntas para que Alexa pueda entenderlas con precisión.
- Utiliza los ejemplos de comandos proporcionados para obtener información de manera eficiente.
- Para obtener ayuda adicional, puedes decir "Ayuda" en cualquier momento durante la interacción con la habilidad.

12. Adaptación para Otro Centro

Si deseamos adaptar la habilidad para otro centro educativo que pueda tener una oferta educativa diferente, es necesario realizar ciertas modificaciones para reflejar adecuadamente las características específicas de ese centro. Aquí hay algunas consideraciones clave a tener en cuenta:

1. Actualización del contenido:

- Será necesario revisar y actualizar todos los archivos de texto que contienen información sobre la oferta educativa, admisiones, noticias, etc., para reflejar la realidad del nuevo centro educativo.
- Se deben crear nuevos archivos de texto o modificar los existentes para incluir detalles sobre los programas académicos, procesos de admisión y cualquier otra información relevante específica del nuevo centro.

2. Modificación de Intents y respuestas:

- Los intents relacionados con la oferta educativa, admisiones y otras funcionalidades deben ser adaptados para reflejar las necesidades y características únicas del nuevo centro.
- Las respuestas generadas por la habilidad deben ser actualizadas para proporcionar información precisa y relevante sobre las diferentes áreas del nuevo centro educativo.

3. Ajustes en la lógica de negocio:

- La lógica de negocio de la habilidad debe ser ajustada para manejar adecuadamente las solicitudes específicas del nuevo centro, como la estructura de los programas académicos, los requisitos de admisión, etc.
- Se deben realizar cambios en la forma en que la habilidad procesa la información y genera respuestas para garantizar que sea coherente con las políticas y procedimientos del nuevo centro.

4. Personalización de mensajes de bienvenida y despedida:

- Los mensajes de bienvenida y despedida de la habilidad deben ser personalizados para incluir el nombre y la identidad del nuevo centro educativo, brindando así una experiencia más auténtica y relevante para los usuarios.

5. Pruebas adicionales:

- Después de realizar todas las modificaciones necesarias, se deben realizar pruebas exhaustivas para asegurar que la habilidad funcione correctamente y proporcione información precisa y útil sobre el nuevo centro educativo.
- Las pruebas deben incluir la verificación de la precisión de la información proporcionada, así como la coherencia y fluidez del flujo de conversación.

Al realizar estas modificaciones y ajustes, la habilidad podrá adaptarse eficazmente para representar y promover las características y programas educativos del nuevo centro con precisión y efectividad.

11. Mejoras

Para mejorar la calidad y la experiencia del usuario de la habilidad de Alexa desarrollada para el centro educativo, se pueden implementar una serie de mejoras que agreguen valor y funcionalidad adicional. Aquí se presentan algunas sugerencias para posibles mejoras:

Integración de datos dinámicos mediante Webscraping:

Incorporar la capacidad de actualizar dinámicamente la información sobre la oferta educativa, noticias y otros contenidos relevantes a través de una fuente de datos en línea. Esto garantizará que la información proporcionada a los usuarios esté siempre actualizada y precisa.

Personalización de la experiencia del usuario:

Permitir a los usuarios personalizar su experiencia mediante la selección de preferencias, como áreas de interés académico o noticias específicas que

deseen recibir. Esto puede lograrse mediante la implementación de perfiles de usuario.

Soporte multilingüe:

Ampliar la disponibilidad de la habilidad mediante la incorporación de soporte para múltiples idiomas. Esto permitirá que un mayor número de usuarios acceda a la información y los servicios proporcionados por la habilidad, lo que aumentará su utilidad y alcance.

Funcionalidad de búsqueda avanzada:

Implementar una funcionalidad de búsqueda avanzada que permita a los usuarios buscar información específica dentro de la habilidad. Esto podría incluir la capacidad de realizar búsquedas por palabras clave, categorías o fechas, facilitando así el acceso a la información deseada de manera rápida y eficiente.

Integración con asistentes virtuales externos:

Explorar la integración con otros asistentes virtuales populares, como Google Assistant o Siri, para garantizar una mayor accesibilidad y compatibilidad con una variedad de dispositivos y plataformas.

Mejoras en la interfaz de voz:

Refinar y mejorar la interfaz de voz para hacerla más natural, conversacional y fácil de entender para los usuarios. Esto podría implicar la incorporación de tecnologías de procesamiento de lenguaje natural más avanzadas y la optimización de los diálogos de conversación.

Feedback y mejora continua:

Implementar un sistema de retroalimentación de los usuarios para recopilar comentarios y sugerencias que ayuden a identificar áreas de mejora y priorizar futuras actualizaciones y características.

Al implementar estas mejoras, se puede crear una habilidad de Alexa más completa, útil y atractiva que satisfaga las necesidades y expectativas de los usuarios de manera más efectiva.

12. Conclusiones

El desarrollo de la habilidad "IES Ángel Corella" ha sido un proceso enriquecedor que ha permitido explorar y aplicar diversos conceptos y técnicas relacionadas con el diseño, desarrollo y despliegue de aplicaciones de voz utilizando la plataforma Alexa de Amazon. A través de este proyecto, se han alcanzado varios objetivos y se han adquirido importantes aprendizajes, que se resumen a continuación:

1. **Dominio de las herramientas de desarrollo:** El proceso de desarrollo de la habilidad ha requerido el dominio de las herramientas y tecnologías asociadas, incluyendo el uso del kit de desarrollo de Alexa (ASK) y el lenguaje de programación Python. Inicialmente, se comenzó el desarrollo

utilizando Node.js, lo cual fue una experiencia valiosa. Sin embargo, se optó por migrar a Python debido a su simplicidad, legibilidad y la amplia disponibilidad de bibliotecas y frameworks, que facilitaron la implementación de diversas funcionalidades. Este proyecto ha permitido fortalecer las habilidades en programación y comprensión de conceptos relacionados con el desarrollo de aplicaciones de voz.

2. **Diseño centrado en el usuario:** Durante la concepción y desarrollo de la habilidad, se ha puesto un énfasis particular en el diseño centrado en el usuario. Se han considerado las necesidades y expectativas de los potenciales usuarios de la habilidad, así como las mejores prácticas de diseño de interfaces de voz, para ofrecer una experiencia de usuario intuitiva, fluida y satisfactoria.
3. **Implementación de funcionalidades avanzadas:** La habilidad "IES Ángel Corella" incluye una variedad de funcionalidades avanzadas, como la lectura dinámica de archivos de texto para proporcionar contenido actualizado, el manejo de múltiples intenciones para responder de manera precisa a las solicitudes del usuario, y el uso de excepciones para gestionar posibles errores durante la ejecución de la habilidad.
4. **Adquisición de conocimientos educativos:** A través del contenido incorporado en la habilidad, relacionado con la oferta educativa del centro y los programas académicos disponibles, se ha profundizado en el conocimiento sobre el funcionamiento y estructura de la institución educativa, así como en las distintas opciones educativas disponibles para los estudiantes.
5. **Desafíos y oportunidades futuras:** Si bien el desarrollo de la habilidad ha sido un éxito, también se han enfrentado diversos desafíos durante el proceso, como la gestión de la complejidad del código y la optimización del rendimiento de la habilidad. Para futuras iteraciones, se identifican oportunidades de mejora en áreas como la incorporación de funcionalidades de interacción más complejas y la expansión del contenido disponible para los usuarios.

En resumen, el desarrollo de la habilidad "IES Ángel Corella" ha sido un proyecto apasionante que ha permitido aplicar y consolidar conocimientos en el campo del desarrollo de aplicaciones de voz, así como adquirir nuevas habilidades y perspectivas sobre el diseño y la implementación de soluciones tecnológicas centradas en el usuario. Este proyecto representa un punto de partida sólido para futuras investigaciones y desarrollos en el ámbito de la voz y la inteligencia artificial aplicada a la educación y otros campos.

13. Bibliografía:

Canal de YouTube: Alexa Developers. *YouTube*. Recuperado de https://www.youtube.com/channel/UCbx0SPpWT6yB7_yY_ik7pmg

Repositorio de GitHub: Alexa Samples. *GitHub*. Recuperado de <https://github.com/alexa-samples>

Curso de Udemy: Alexa Skills Desde Cero. *Udemy*. Recuperado de <https://www.udemy.com/course/alexa-skills/?couponCode=LEADERSALE24A>