

Travel Assistant

Curso: Capstone (PTY4614_007V)

Profesor: Arturo Guerra Castro

Integrantes:

- Cristóbal Cabezas Espinoza
- José Tobar Estay
- Edgard León

1. Índice

1. Índice.....	2
2. Descripción del proyecto.....	4
2.1 Contexto.....	4
2.2 Problemas a solucionar.....	4
2.3 Propuesta de solución.....	4
3. Objetivo del Proyecto.....	6
3.1 Objetivo general.....	6
3.2. Objetivos secundarios.....	6
3.3 Equipo y roles.....	6
3.3.1 Del equipo de desarrollo.....	6
3.3.2 De los usuarios.....	7
4. Diseño del proyecto.....	8
4.1 Mapa de Usuarios.....	8
4.2 Historias de usuarios.....	9
4.3 Diseño del sistema (Figma).....	10
4.4 Alcances del proyecto.....	10
4.5 Limitaciones del proyecto.....	10
5. Competencias de carrera.....	11
6. Metodología de trabajo para el desarrollo del proyecto.....	12
6.1 Selección de metodología.....	12
6.1.1 Uso de la Metodología Ágil (Scrum).....	12
6.1.3 Beneficios de Utilizar Scrum.....	12
6.2 Cronograma.....	13
6.3 Plan del Proyecto: Desarrollo de Asistente Virtual.....	13
6.4 Uso de Jira.....	14
7. Solución técnica.....	15
7.1 Arquitectura.....	15
7.1.1 Arquitectura del proyecto "CTS Travel Assistant".....	15
7.1.2 Flujo de trabajo general:.....	16
7.2 Herramientas de desarrollo.....	17
7.2.1 Gestión de Proyecto y Colaboración.....	17
7.2.2 Desarrollo y Control de Versiones.....	17
7.2.3 Desarrollo Backend.....	17
7.3 Entorno (Cloud).....	18
7.4 Repositorio de código.....	18
8. Prompt Engineering.....	19
8.1 Contexto.....	19
8.2 Explicación del flujo.....	19
9. Resultados obtenidos.....	21

9.1 Casos de uso.....	21
9.1.1. Caso de Uso: Reserva de Hotelería.....	21
9.1.2. Caso de Uso: Reserva de Excursión o Traslado.....	21
9.1.3. Caso de Uso: Modificación de Reserva de Hotel.....	22
9.1.4. Caso de Uso: Anulación de Reserva.....	22
9.2 Pruebas funcionales.....	23
10. Obstáculos presentados durante el desarrollo.....	23
11. Conclusiones.....	25

2. Descripción del proyecto

2.1 Contexto

CTS Turismo es una empresa de turismo enfocado al mercado receptivo (extranjero). Con más de 40 años de existencia, se ha posicionado como un tour operador líder en Chile en la venta de servicios turísticos hacia el extranjero, tanto en hotelería, traslados y excursiones, entre otros.

Como parte de un constante esfuerzo por mejorar la calidad de los servicios que CTS presta a sus clientes, es que desde el último tiempo ha apostado por el desarrollo y mejoramiento de sus servicios digitales, entre los que destaca portal denominado Booking CTS, un sitio web dedicado a la venta de servicios turísticos, que incluye hospedajes en hoteles, traslados y excursiones, todos disponibles a través de convenios establecidos con diferentes hoteles y proveedores. El sistema permite a los ejecutivos de CTS realizar reservas, gestionar preferencias y utilizar funcionalidades como un carro de compras y un sistema de pago en línea, facilitando la administración de los servicios disponibles.

2.2 Problemas a solucionar

- a) **Resistencia al cambio:** Los usuarios -principalmente ejecutivos de la empresa y de sus clientes- muestran reticencia a modificar sus hábitos existentes, priorizando la toma de reservas mediante los canales tradicionales por sobre las soluciones digitales que CTS Turismo tiene a su disposición.
- b) **Dificultad en el uso de plataformas digitales:** Muchos usuarios encuentran complicado integrar en su flujo de trabajo las herramientas digitales actualmente existentes. Lo anterior demuestra la necesidad de buscar soluciones que permitan una mayor adopción de los canales digitales.

2.3 Propuesta de solución

Desarrollar un Asistente Virtual (AV), que pueda facilitar la toma de reservas de hotelería y servicios turísticos ofrecidos en la web Booking CTS a ejecutivos y clientes de la empresa.

- **Interfaz de Usuario:** El AV se presentará como un ícono en la página principal de CTS Turismo. Al hacer clic en este ícono, se abrirá una ventana de chat donde los usuarios podrán iniciar una conversación.
- **Conversación Natural:** El AV no se limitará a ofrecer opciones predefinidas, sino que podrá mantener conversaciones fluidas, similares a las que se tendrían con una persona real. Esto se logrará mediante el uso de técnicas de inteligencia artificial y procesamiento del lenguaje natural (LLM), a través de la implementación del framework conocido como Langchain, muy popular en la actualidad para el desarrollo de aplicaciones que aprovechen los modelos LLM actualmente en funcionamiento, tales como ChatGPT, Llama, etc.
- **Filtros según preferencias:** Durante la conversación, el AV podrá filtrar los resultados de búsqueda a partir de las preferencias que el usuario le exprese, como

destinos deseados, rangos de precios, tipos de alojamiento, actividades de interés, entre otros. Con esta información, el AV podrá listar opciones más adecuadas para las preferencias del usuario.

- **Interacción con el Sistema:** Una vez que el usuario haya elegido uno o más servicios recomendados, el AV procederá a la toma de la reserva. Previamente, el AV consultará al usuario si desea continuar o no con su reserva, dejando al usuario tomar la decisión definitiva. Dicho de otra manera, el AV no se limitará a responder preguntas, sino que se espera que tome decisiones que devengarán en acciones (como crear una reserva, modificarla o anularla).
- **Facilitación del Proceso de Compra:** El AV guiará al usuario a lo largo de todo el proceso de selección y compra, desde la consulta de los servicios hasta la creación de la reserva, asegurando que incluso aquellos con menos experiencia tecnológica puedan completar su transacción sin mayores dificultades.

3. Objetivo del Proyecto

3.1 Objetivo general

El objetivo general del proyecto es incrementar la adopción de la plataforma digital Booking CTS mediante la implementación de nuevas tecnologías que mejoren la experiencia de usuario y complementen las funcionalidades existentes.

3.2. Objetivos secundarios

- a) Incrementar la adopción de la plataforma web Booking CTS entre ejecutivos y clientes de la empresa (usuarios finales).
- b) Constituir a la plataforma web Booking CTS como un estándar para la toma de reservas de hotelería y servicios entre los usuarios de la empresa y clientes.
- c) Simplificar el proceso de reserva: Crear un flujo continuo y eficiente desde la creación de la reserva hasta su confirmación final.
- d) Crear nuevas experiencias de usuario aprovechando los avances tecnológicos de los últimos años en modelos LLM.
- e) Mejorar la usabilidad: Diseñar una interfaz intuitiva que aumente la eficiencia de los ejecutivos en sus tareas diarias.

3.3 Equipo y roles

3.3.1 Del equipo de desarrollo

Para el correcto desempeño de cada uno de los miembros del equipo que desarrollará y llevará adelante este proyecto, se definieron los siguientes roles a cada uno de los miembros:

Nombre	Rol	Descripción
Cristóbal Cabezas Espinoza	Product Owner / Desarrollador (full stack)	Comprende las necesidades y lógica del negocio, y desarrolla la solución usando las herramientas seleccionadas para el caso.
José Tobar Estay	Diseñador UX y UI / Scrum master	Se encarga de diseñar interfaces intuitivas y liderar al equipo en la metodología Agile, asegurando una experiencia de usuario óptima y un desarrollo

		eficiente.
Edgard León	Administrador de Infraestructura / Project manager	Es el responsable de gestionar la infraestructura tecnológica y coordinar proyectos, asegurando eficiencia, seguridad, cumplir los plazos y resolución de problemas y comunicación entre equipos técnicos y stakeholders.

3.3.2 De los usuarios

El sistema contará con un único tipo de usuario general para garantizar una experiencia simple y accesible en el Proyecto de Asistente Virtual de Agencia de Viajes CTS.

Usuario General

- **Descripción:** Usuario único con acceso completo a las funcionalidades necesarias para interactuar con el sistema.
- **Permisos:**
 - Realizar compras.
 - Revisar ofertas disponibles en el sistema, incluidas promociones y descuentos especiales.
 - Consultar y gestionar reservas (editar, modificar, anular o cancelar).

4. Diseño del proyecto

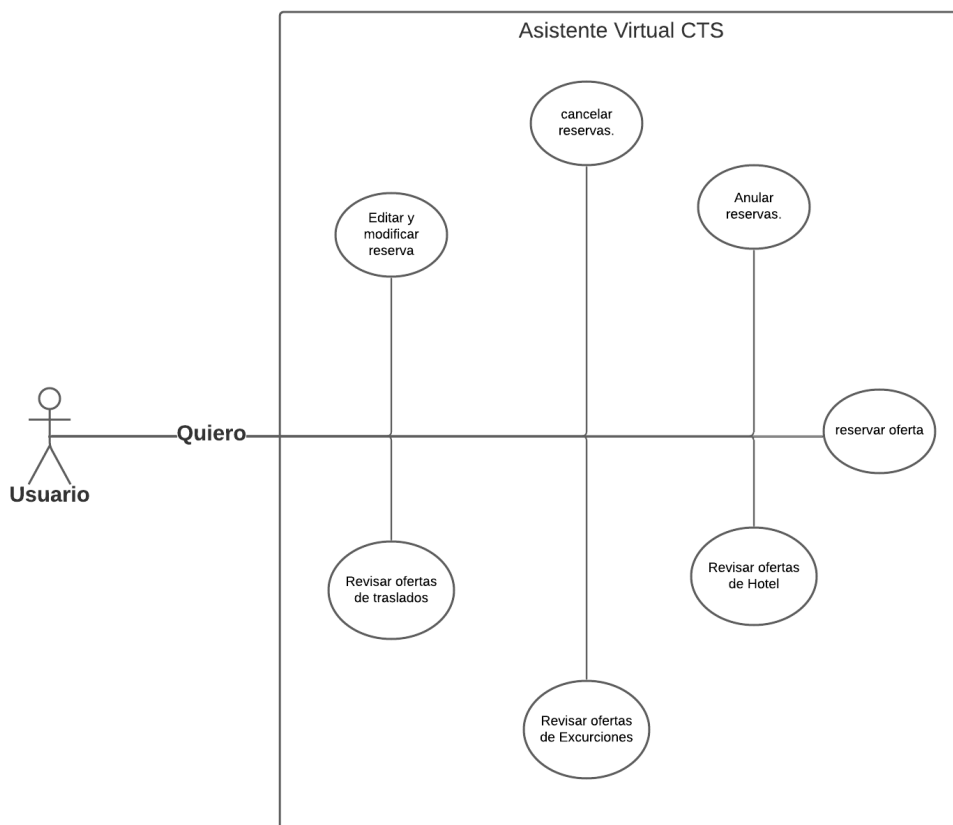
4.1 Mapa de Usuarios

El mapa de usuarios del sistema se simplifica para reflejar un único rol que integra todas las responsabilidades y permisos necesarios para operar el sistema.

1. Usuario General

- **Rol:** Usuario único con acceso a todas las funcionalidades principales del sistema.
- **Permisos:**
 - Realizar compras.
 - Revisar ofertas y promociones disponibles.
 - Consultar y gestionar reservas (editar, modificar, anular o cancelar).
 - Acceso a campañas y descuentos especiales.

“Al contar con un usuario se optimiza la gestión del sistema al consolidar las funcionalidades en un único rol, facilitando la interacción de los usuarios y eliminando la complejidad asociada a la administración de múltiples perfiles. Esto garantiza una experiencia más fluida, intuitiva y eficiente.”



4.2 Historias de usuarios

- **HU1.** Como usuario, quiero poder decirle al asistente el servicio que necesito, o bien que me pregunte el que necesito.
- **HU2.** Como usuario, quiero poder decirle al asistente mi destino o que me pregunte por él.
- **HU3.** Como usuario, quiero poder decirle al asistente la fecha de reserva que necesito o que me pregunte por ella.
- **HU4.** Como usuario, quiero poder decirle al asistente la cantidad de pasajeros que necesito para la reserva o que me pregunte por aquellas.
- **HU5.** En caso de servicio de hotelería, como usuario, quiero poder decirle al Asistente Virtual la cantidad de habitaciones que necesito, o bien que me lo pregunte.
- **HU6.** Como usuario, quiero que el Asistente Virtual me recomiende servicios de acuerdo a mis necesidades.
- **HU7.** Como usuario con necesidades especiales, quiero que el asistente virtual me ofrezca opciones accesibles como habitaciones adaptadas para que mi viaje sea cómodo y seguro.
- **HU8.** Como usuario, quiero que el Asistente Virtual me de la posibilidad de consultar el detalle de los servicios recomendados, utilizando las herramientas de visualización ya existentes en el frontend.
- **HU9.** Como usuario, quiero que el Asistente Virtual pueda realizar un reserva a partir de las elecciones realizadas.
- **HU10.** Como usuario, quiero que me pregunte si deseo continuar con la reserva, antes de confirmarla.
- **HU11.** Como usuario, quiero que el asistente virtual pueda ser interrumpido (detenido) o pausado según mis preferencias.
- **HU12.** Como usuario, quiero que el Asistente Virtual, una vez concluida mi reserva, me genere el correspondiente hipervínculo que me redirija al detalle la misma.

4.3 Diseño del sistema (Figma)

Para el diseño de sistema, se ha generado un mockup con las vistas de usuario a través de las distintas historias de usuario antes señaladas. El señalado mockup ha sido generado con la herramienta Figma y su resultado es posible encontrarlo en **Anexo 1** de este informe.

También se encuentra disponible en el siguiente link:

<https://www.figma.com/design/hMYhUKvnPDAmFvVpFEPLTU/CTS-Travel-Assistant?node-id=0-1&t=AUXRq7x4i70ccYLk-1>

4.4 Alcances del proyecto

- a) **Desarrollo de Asistente Virtual:** Crear un asistente virtual que permita dar cumplimiento a los objetivos trazados.
- b) **Integración de Funcionalidades:** Proyecto debe estar integrado con las funcionalidades que el sitio web ya ofrece, como la posibilidad de elegir moneda e idioma, o es uso de las vistas para consultar información de hoteles y/o servicios disponibles.
- c) **Diseñar una interfaz fácil de usar:** que optimice la navegación y uso de la herramienta. Debe ser responsivo y habilitado para su uso en entornos móviles (celulares y tablets).
- d) **Despliegue en Producción:** Implementar el asistente en un entorno real de trabajo.

4.5 Limitaciones del proyecto

- a) **Tiempo y recursos limitados:** El proyecto debe completarse en 15 semanas, limitando el alcance de las funcionalidades.
- b) **Desafíos técnicos en integración de sistemas:** Desafíos técnicos al integrar el asistente con los demás sistemas de la empresa.
- c) **Flujo interrumpido por alucinaciones o “callejones sin salida”:** Comprender que el LLM tiene sus propias limitaciones que pueden provocar interrupción del flujo e incluso impedir conclusión.

5. Competencias de carrera

De acuerdo a las competencias de especialidad establecidas para la carrera de Ingeniería en Informática en Duoc UC, el presente proyecto desarrolla las siguientes competencias:

- a) **Desarrollo de solución de Software:** Construir programas y rutinas de variada complejidad para dar solución a requerimientos de la organización, acordes a tecnologías de mercado y utilizando buenas prácticas de codificación.
- b) **Gestión de Proyectos TI:** Gestionar proyectos informáticos, ofreciendo alternativas para la toma de decisiones de acuerdo a los requerimientos de la organización
- c) **Construcción de soluciones informáticas:** Ofrecer propuestas de solución informática analizando de forma integral los procesos de acuerdo a los requerimientos de la organización.
- d) **Seguridad Informática:** Resolver las vulnerabilidades sistémicas para asegurar que el software construido cumple las normas de seguridad exigidas por la industria.
- e) **Aseguramiento de la calidad (QA):** realizar pruebas de certificación tanto de los productos como de los procesos utilizando buenas practicas.

6. Metodología de trabajo para el desarrollo del proyecto

6.1 Selección de metodología

6.1.1 Uso de la Metodología Ágil (Scrum)

Para el desarrollo del Proyecto de Asistente Virtual de la Agencia de Viajes, hemos decidido implementar la metodología ágil, específicamente el marco de trabajo **Scrum**. Esta elección se basa en la necesidad de adaptabilidad, colaboración y entrega continua de valor que caracteriza a nuestro proyecto. La metodología Scrum es ideal para proyectos donde los requisitos pueden evolucionar con el tiempo y donde la interacción constante con los stakeholders es esencial para el éxito del producto.

6.1.3 Beneficios de Utilizar Scrum

1. **Flexibilidad y Adaptabilidad:** Scrum permite adaptarse rápidamente a cambios en los requisitos o al descubrimiento de nuevas necesidades del cliente. Esto es especialmente valioso en nuestro proyecto, donde los requerimientos pueden ser refinados o redefinidos a lo largo del desarrollo.
2. **Entrega Continua de Valor:** A través de sprints cortos, el equipo puede entregar incrementos funcionales del producto de manera frecuente. Esto asegura que se pueda obtener feedback temprano y frecuente, permitiendo ajustes y mejoras continuas.
3. **Transparencia y Visibilidad:** Las reuniones diarias (Daily Stand-ups), junto con las revisiones de sprint y las retrospectivas, promueven una comunicación abierta y una alta visibilidad del progreso del proyecto. Esto facilita la identificación temprana de obstáculos y permite tomar acciones correctivas oportunas.
4. **Mejora Continua:** Scrum fomenta la revisión continua del proceso de desarrollo, permitiendo que el equipo aprenda y mejore su rendimiento con cada sprint. Las retrospectivas al final de cada sprint son clave para identificar áreas de mejora y aplicar cambios.
5. **Participación Activa del Cliente:** Scrum facilita la colaboración constante con los stakeholders, asegurando que el producto final se alinee con las expectativas y necesidades del cliente. Esto es esencial en nuestro proyecto para garantizar que el asistente virtual cumpla con los requerimientos de la agencia de viajes.

La elección de Scrum también se fundamenta en la claridad y la naturaleza evolutiva de los requerimientos del proyecto. Aunque contamos con una definición inicial de los usuarios y sus permisos, reconocemos que los detalles y funcionalidades adicionales pueden surgir durante el desarrollo. Scrum nos ofrece la estructura necesaria para manejar estas

situaciones de manera efectiva, permitiendo iterar sobre los requerimientos y refinar el producto final en cada sprint.

En resumen, Scrum fue seleccionado debido a su capacidad para gestionar la incertidumbre y su enfoque en la entrega continua de valor, alineándose perfectamente con las necesidades y objetivos de nuestro proyecto.

6.2 Cronograma

Se presenta el plan de desarrollo de un asistente virtual para CTS Turismo, utilizando la metodología ágil con el marco de trabajo Scrum. El proyecto tiene una duración de 15 semanas, durante las cuales se realizarán actividades de planificación, desarrollo, integración, pruebas y despliegue.

El uso de Scrum nos permite trabajar de manera iterativa e incremental, entregando valor al cliente de forma continua y adaptándonos a posibles cambios en los requisitos a lo largo del proyecto. Cada sprint tiene un objetivo claro y definido, asegurando que el equipo mantenga el enfoque en la entrega de funcionalidades clave y en la mejora continua del producto.

6.3 Plan del Proyecto: Desarrollo de Asistente Virtual

Duración Total del Proyecto: 15 semanas

Sprint	Fechas	Objetivos	Historias de Usuario (HU)
Sprint 0: Planificación y Configuración Inicial	24/08/24 - 06/09/24	Configuración del entorno y planificación inicial	N/A
Sprint 1: Desarrollo del flujo de consultas	07/09/24 - 20/09/24	Implementar selección de servicio y destino	HU1, HU2, HU3, HU4, HU5, HU6, HU7, HU8
Sprint 2: Desarrollo de flujo de reservas	21/09/24 - 04/10/24	Implementar selección de fecha y pasajeros	HU9, HU10, HU11, HU12
Sprint 3: Desarrollo de la interfaz conversacional	05/10/24 - 18/10/24	Gestión de habitaciones y resultados limitados	H11
Sprint 4: Testing	19/10/24 - 04/11/24	Realización de pruebas funcionales y corrección de errores	N/A
Sprint 5: Implementación	05/11/24 - 22/11/24	Implementación de proyecto en Google Cloud Run	N/A

Características Ágiles de la Planificación:

1. **Sprints Cortos y Enfocados:** Cada sprint tiene una duración de aproximadamente 2 semanas (ajustado por días hábiles).
2. **Entregas Funcionales:** Cada sprint entrega una parte funcional del asistente virtual, lista para ser probada y refinada.
3. **Flexibilidad:** Los sprints no dependen estrictamente de otros, lo que permite flexibilidad y adaptación según el progreso y feedback.
4. **Iteración Continua:** En cada sprint se realiza una revisión y adaptación de las funcionalidades anteriores si es necesario.
5. **Prioridad en Valor al Usuario:** Las historias de usuario están organizadas para ofrecer valor al usuario desde los primeros sprints.

6.4 Uso de Jira

Para dar seguimiento al plan de trabajo antes señalado, se ha optado por el uso de la herramienta denominada Jira, de la compañía Atlassian, para gestionar y administrar las distintas fases del proyecto.

El cronograma y realización de las diferentes tareas y actividades han quedado plasmadas en el **Anexo 2** del presente informe.

7. Solución técnica

7.1 Arquitectura

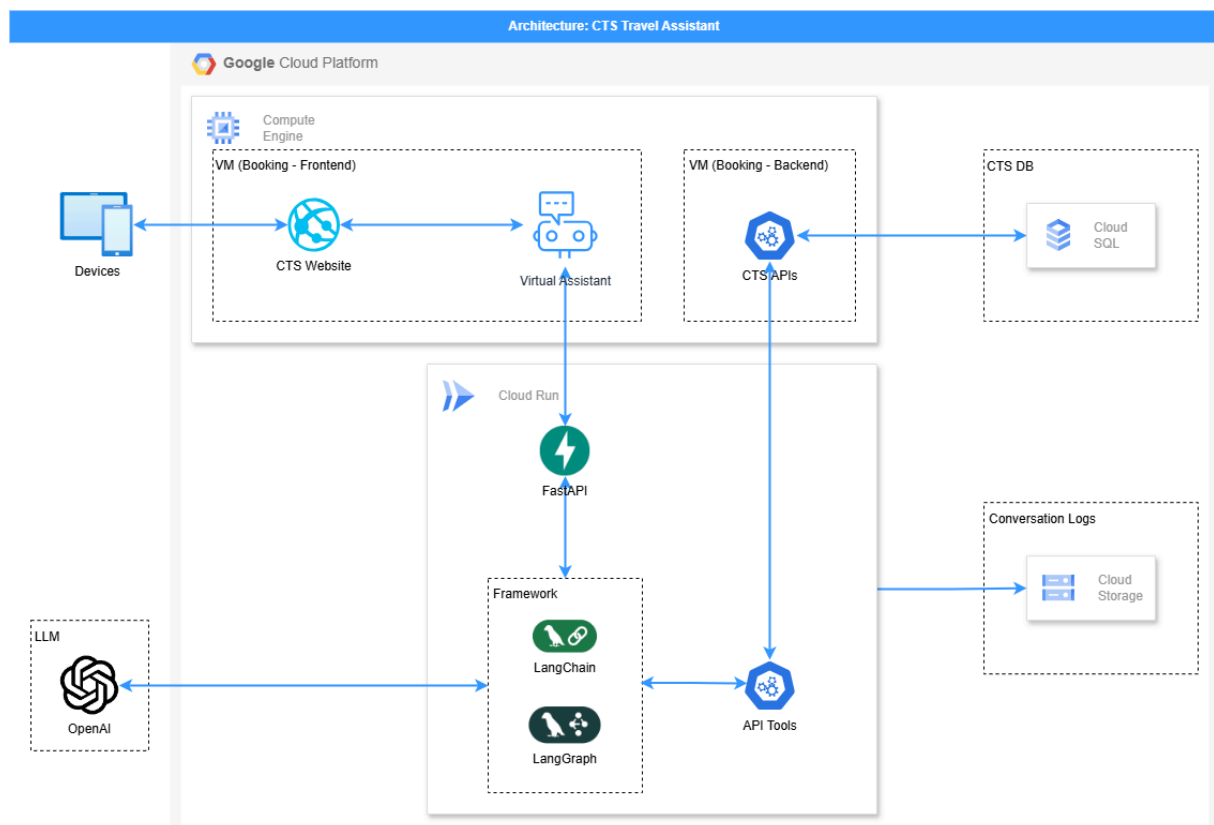
7.1.1 Arquitectura del proyecto "CTS Travel Assistant"

1. **Dispositivos (Devices):** Representa a los usuarios finales que acceden al sistema a través de diversos dispositivos como computadoras, tablets o smartphones. Estos usuarios interactúan con el sitio web de CTS para realizar consultas y reservas.
2. **CTS Website (Frontend):** Este es el sitio web principal donde los usuarios pueden interactuar con el asistente virtual (chatbot). Está desplegado en una máquina virtual (VM) dentro de Google Compute Engine y ofrece una interfaz amigable para realizar acciones como explorar servicios turísticos y completar reservas.
3. **Chatbot:** Es una funcionalidad integrada en el sitio web que utiliza LangChain para generar respuestas personalizadas y guiar a los usuarios en su experiencia. Este chatbot actúa como intermediario entre los usuarios y el sistema backend.
4. **CTS APIs (Backend):** Este módulo se encuentra en otra máquina virtual dentro de Google Compute Engine. Contiene las APIs que manejan la lógica del negocio, como el procesamiento de reservas, gestión de servicios turísticos y conexión con la base de datos.
5. **Cloud SQL:** Es la base de datos donde se almacena toda la información relevante del sistema, incluyendo los detalles de los usuarios, preferencias, reservas y los servicios turísticos ofrecidos.
6. **LangChain:** Es el framework que permite integrar el modelo de lenguaje de OpenAI para enriquecer la interacción del chatbot con los usuarios. Este módulo procesa las consultas y genera respuestas más contextuales y detalladas.
7. **API en Cloud Run:** Una API específica está desplegada en Google Cloud Run, un servicio serverless que escala automáticamente. Esta API conecta las consultas del chatbot y otras funcionalidades del sistema con el modelo de OpenAI.
8. **OpenAI:** Es el servicio externo utilizado para procesar el lenguaje natural, mejorar la interacción con los usuarios y generar respuestas inteligentes a través del chatbot.

7.1.2 Flujo de trabajo general:

1. Los Ejecutivos acceden al sitio web de CTS desde sus dispositivos.
2. Interactúan con el chatbot, que recibe las consultas, y el servicio de LangChain las procesa.
3. LangChain envía hacia OpenAI la consulta realizada para que este último genere respuestas dinámicas y/o personalizadas.
4. La IA selecciona, a partir de las API's Tools existentes, la herramienta que mejor se adecue a la consulta realizada, y hace un llamado al Backend de Booking CTS, el cual se encuentra en una instancia VM de Compute Engine.
5. El backend, a través de sus propias API's, consulta a la base de datos, el cual se encuentra en una instancia de Cloud SQL, la disponibilidad habida para el servicio requerido de acuerdo a los parámetros ingresados por el usuario.
6. Una vez obtenida la información, ésta es devuelta a la API Tool, el cual procesa el resultado y la devuelve a la IA en forma que pueda ser leída y utilizada por esta.
7. La IA procesa la información y la despliega al usuario en un formato de conversación.
8. La instancia en Google Cloud Run, a su vez, asegura la comunicación eficiente y escalable entre las diferentes partes del sistema.

Esta arquitectura combina soluciones escalables (como Cloud Run) con procesamiento avanzado de lenguaje natural (LangChain + OpenAI), proporcionando una experiencia robusta y optimizada para el usuario.



7.2 Herramientas de desarrollo

7.2.1 Gestión de Proyecto y Colaboración

- **Jira:** Para la gestión de tareas, seguimiento del progreso del proyecto y manejo del backlog en Scrum.

7.2.2 Desarrollo y Control de Versiones

- **GitHub:** Plataformas para el alojamiento del código, gestión de ramas, pull requests, y code review.
- **Visual Studio Code** o **PyCharm:** Editores de código con soporte para diferentes lenguajes, integración de Git, y herramientas de desarrollo adicionales.

7.2.3 Desarrollo Backend

- **Python:** Lenguaje de programación principal para el desarrollo del backend del asistente virtual.
- **Poetry:** Para la administración de las dependencias del proyecto.
- **Docker:** Para manejar la aplicación mediante contenedores que podamos luego implementar en una instancia de Google Cloud Run.
- **LangChain:** Framework para la construcción de aplicaciones impulsadas por modelos de lenguaje, como asistentes virtuales, permitiendo la integración y orquestación de diferentes modelos de IA.
- **LangGraph:** Biblioteca del ecosistema Langchain que se utiliza para trabajar con múltiples agentes mediante flujos estructurados, el cual nos permitirá crear un asistente virtual completamente funcional capaz de crear una reserva.

7.2.4 Desarrollo Frontend

- Al igual que en el caso del backend, en el caso del front se trabajará con el framework base del proyecto original, el cual es Nuxt 3.

7.2.5 Integración y Automatización

- Para el desarrollo del Asistente Virtual este se nutrirá con la información que las API's del backend de Booking CTS proporciona.
- Lo anterior significa que el presente proyecto no contempla el desarrollo de un nuevo backend, sino el aprovechamiento del existente.

7.2.6 Pruebas y QA

- Por la naturaleza del proyecto y de las tecnologías a utilizar, las pruebas de calidad están muy centradas en la realización de prompt engineering, de manera que nuestro LLM devuelva los resultados esperados.

7.3 Entorno (Cloud)

El proyecto será implementado en un entorno de Google Cloud Platform, junto al proyecto original de Booking CTS. Este nuevo proyecto será implementado al interior de una máquina virtual (VM) de Compute Engine y estará habilitado y encendido las 24 horas del día, junto a los demás servicios online de CTS Turismo.

7.4 Repositorio de código

El código se ha almacenado en un repositorio en Github, el cual es posible consultarlo en el siguiente link:

<https://github.com/CristobalCabezas/travel-assistant>

8. Prompt Engineering

8.1 Contexto

Es fundamental para configurar un asistente virtual eficaz, ya que define su comportamiento, tono y alcance. Este proceso consiste en diseñar instrucciones claras que permitan respuestas precisas, útiles y alineadas con las necesidades de los usuarios.

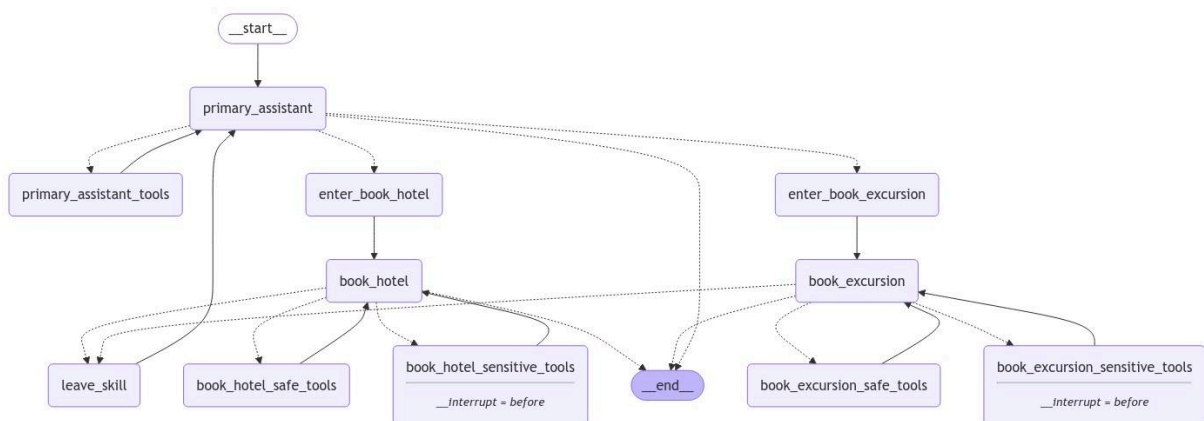
En el contexto del turismo, un buen diseño de prompts debe considerar:

- **Objetivo:** Responder preguntas, recomendar paquetes turísticos o gestionar reservas.
- **Tono:** Cercano, profesional y orientado al cliente.
- **Contexto:** Enfocado en información turística relevante, respetando los límites del conocimiento del asistente.

El diseño óptimo de prompts no solo mejora la experiencia del usuario, sino que también optimiza el flujo de trabajo de los ejecutivos.

8.2 Explicación del flujo

El flujo de nuestro chatbot se puede graficar de la siguiente manera:



El señalado diagrama de flujo contiene la descripción de la navegación del AV en sus diversos estados y transiciones entre aquellos, el cual se desglosa de la siguiente manera:

I. Estado principal

1. **__start__**: Consiste en el punto de entrada del flujo, mediante el cual se da inicio a la interacción del usuario con el AV.
2. **primary_assistant**: Tal como lo indica su nombre, consiste en el nodo principal de nuestro AV, el cual representa una fase primaria del chatbot que recopila la información básica del usuario y ofrece asistencia general. Desde este es posible transitar hacia tres estados diferentes:
 - a. **primary_assistant_tools**, el cual maneja las herramientas (tools) y funcionalidades comunes del asistente principal;
 - b. **enter_book_hotel**, el cual transiciona hacia el asistente de reservas hoteleras; o
 - c. **enter_book_excursion**, el cual transiciona hacia el asistente de reservas de excursiones o traslados.

II. Estado intermedio: Reserva de Hotel o Excursión/Traslado

Una vez que el asistente primario haya recopilado la información necesaria de las necesidades del usuario, delegará y remitirá al usuario a los asistentes secundarios para continuar con el resto del flujo, que en el caso del presente proyecto se han establecido dos: uno para la realización de las reservas hoteleras y otro para la realización de reservas de excursiones o traslados.

1. **book_hotel**: Consiste en el estado principal para la acción de reservar un hotel, el cual recopila información del usuario hasta la creación de una reserva.
2. **book_excursion**: Consiste en el estado principal para la acción de reservar una excursión o traslado, el cual recopila información del usuario hasta la creación de una reserva.

III. Safe Tools y Sensitive Tools

Cada uno de estos subasistentes cuentan con un set de herramientas (tools) para dar cumplimiento a los requerimientos del usuario:

1. **safe_tools**: Contienen herramientas que no requieren de confirmación previa del usuario para ejecutarse. Generalmente corresponden a herramientas de consulta, tales como consultar disponibilidad, habitaciones disponibles, traer información del hotel, etc.
2. **sensitive_tools**: Contienen herramientas que sí requieren de confirmación previa del usuario para ejecutarse. Generalmente corresponden a herramientas de acción, tales como crear, modificar o anular una reserva.
 - a. Estas herramientas, por tanto, se encuentran sujetas al **__interrupt = before**, mediante el cual se interrumpe la acción hasta que el usuario la haya consentido expresamente.

IV. Salida y finalización del flujo

Una vez que se haya ejecutado una o más tools del set establecido para el respectivo asistente, existen dos vías posibles: Escalar o finalizar:

1. **leave_skill**: Nodo utilizado para salir del contexto específico en el que se encuentra el usuario. Por ejemplo, para volver al asistente primario o para cambiar de subasistente.
2. **end**: Nodo que finaliza el proceso. Ocurre cuando el usuario cierra la conversación desde la interfaz visual.

9. Resultados obtenidos

9.1 Casos de uso

9.1.1. Caso de Uso: Reserva de Hotelería

Interacción inicial: El usuario accede al sitio web de Booking CTS desde su dispositivo. Una vez en el sitio, el usuario habilita una conversación con el asistente virtual, al cual le consulta opciones de hospedaje en Puerto Varas para 2 personas, con fechas del 15 al 17 de diciembre de 2024.

1. El asistente **despliega la disponibilidad** solicitada.
2. El usuario **selecciona un hotel**.
3. El asistente despliega las **habitaciones disponibles**.
4. El usuario **elige una de las habitaciones** disponibles.
5. El asistente solicita al usuario **datos del pasajero** para completar la reserva.
6. El usuario le **indica al asistente** los mencionados datos.
7. El asistente, previo a realizar la reserva, le consulta al usuario **si está seguro que desea continuar**.
8. Si el usuario dice que sí, el asistente **creará sin más la reserva**. Si dice que no, continuará asistiendo al usuario.
9. Finalmente, el asistente indicará al usuario el **número de su reserva** y le generará un **link para visualizar el detalle** de la misma.

9.1.2. Caso de Uso: Reserva de Excursión o Traslado

Interacción inicial: El usuario accede al sitio web de Booking CTS desde su dispositivo. Una vez en el sitio, el usuario habilita una conversación con el asistente virtual, al cual le consulta opciones de servicios de excursión/traslado en Santiago para 1 persona, para el 15 de enero de 2025.

1. El asistente **despliega la disponibilidad** solicitada.
2. El usuario **selecciona un servicio de excursión/traslado**.
3. El asistente despliega las **opciones disponibles** de servicio (por ejemplo, si es privado o compartido, en español o inglés, si es con guía o no, etc.).

4. El usuario **elige una de las opciones** disponibles.
5. El asistente solicita al usuario **datos del pasajero** para completar la reserva.
6. El usuario le **indica al asistente** los mencionados datos.
7. El asistente, previo a realizar la reserva, le consulta al usuario **si está seguro que desea continuar**.
8. Si el usuario dice que sí, el asistente **creará sin más la reserva**. Si dice que no, continuará asistiendo al usuario.
9. Finalmente, el asistente indicará al usuario el **número de su reserva**.

9.1.3. Caso de Uso: Modificación de Reserva de Hotel

Interacción inicial: El usuario accede al sitio web de Booking CTS desde su dispositivo. Una vez en el sitio, el usuario habilita una conversación con el asistente virtual, al cual le solicita modificar una reserva.

1. El asistente le solicita al usuario indicar **qué tipo de servicio** desea modificar.
2. El usuario indica el **tipo de servicio** (solo hotelería).
3. El asistente **solicita número de reserva** y qué desea cambiar.
4. El usuario **indica el número de reserva** al asistente y lo que desea modificar (por ejemplo, agregar número de vuelo, o agregar solicitudes especiales al proveedor del servicio, etc.).
5. El asistente consulta al usuario **si está seguro** de modificar.
6. Si dice que sí, el asistente **modificará la reserva ingresada**; si dice que no, continuará asistiendo al usuario.
7. El asistente, finalmente, señalará al usuario que la reserva **ha sido anulada correctamente**.

9.1.4. Caso de Uso: Anulación de Reserva

Interacción inicial: El usuario accede al sitio web de Booking CTS desde su dispositivo. Una vez en el sitio, el usuario habilita una conversación con el asistente virtual, al cual le solicita anular una reserva.

8. El asistente le solicita al usuario indicar **qué tipo de servicio** desea anular.
9. El usuario indica el **tipo de servicio** (hotelería/excursión/traslado).
10. El asistente **solicita número de reserva** a anular.
11. El usuario **indica el número de reserva** al asistente.
12. El asistente consulta a usuario **si está seguro** de anular.
13. Si dice que sí, el asistente **anulará la reserva ingresada**; si dice que no, continuará asistiendo al usuario.
14. El asistente, finalmente, señalará al usuario que la reserva **ha sido anulada correctamente**.

9.2 Pruebas funcionales

En el marco del desarrollo y evaluación de nuestro proyecto, hemos llevado a cabo una serie de pruebas funcionales para asegurar que todas las características y funcionalidades del sistema operen de acuerdo con los requisitos establecidos. Estas pruebas nos permiten verificar la correcta implementación de cada componente y su interacción dentro del sistema.

Recomendamos revisar el anexo para obtener una comprensión completa de los resultados y las conclusiones derivadas de las pruebas.

Para la realización de pruebas funcionales se han considerado las siguientes categorías:

- 1) **Búsqueda y consulta:** Pruebas realizadas con el objeto de evaluar el comportamiento del sistema en la elección de los servicios de hotelería, excursiones o traslados.
- 2) **Generación de reservas:** Pruebas realizadas con el objeto de evaluar el comportamiento del sistema en la creación de reservas.
- 3) **Modificación de reservas:** Pruebas realizadas con el objeto de evaluar el comportamiento del sistema en la modificación de reservas.
- 4) **Anulación de reservas:** Pruebas realizadas con el objeto de evaluar el comportamiento del sistema en la anulación de reservas.
- 5) **Respuestas y URL's generadas:** Pruebas realizadas con el objeto de evaluar el comportamiento del sistema en relación a la generación de las vistas.
- 6) **Interacción en lenguaje natural:** Pruebas realizadas con el objeto de evaluar el comportamiento del sistema en relación a la conversación.

El detalle de los resultados de las pruebas funcionales realizadas se encuentran adjuntas a este informe en el denominado **Anexo 3**.

10. Obstáculos presentados durante el desarrollo

Durante el desarrollo de nuestro proyecto de título enfrentamos diversos desafíos que nos permitieron aprender y crecer como equipo. Uno de los mayores retos fue la **falta de experiencia con la metodología ágil**: aunque seguimos Scrum, al inicio no teníamos claro cómo organizar los sprints ni dividir correctamente las tareas, lo que ocasionó atrasos en el cronograma.

Además, no todos contábamos con un dominio avanzado de las tecnologías necesarias, como **Python y LangChain**, lo que nos obligó a dedicar tiempo extra a investigar y aprender

sobre la marcha. Esto impactó directamente en el desarrollo técnico, haciéndolo más lento de lo esperado.

Otro obstáculo significativo fue la **integración del backend con el frontend y las APIs externas**, un proceso más complicado de lo que imaginamos. La **falta de experiencia en integración de sistemas** nos llevó a enfrentar múltiples problemas técnicos no anticipados, consumiendo tiempo adicional en su resolución.

Por otro lado, la **falta de sincronización en el equipo** también fue un desafío. Al combinar nuestras responsabilidades como estudiantes, trabajadores y en muchos casos padres, no siempre logramos mantener un ritmo constante ni una disponibilidad uniforme, lo que afectó el avance grupal. Finalmente, los **problemas en la gestión del tiempo** hicieron que en ciertos momentos las tareas se acumularan, aumentando la presión sobre el equipo.

A pesar de todo, estos retos nos ayudaron a mejorar nuestras habilidades, fortalecer nuestro trabajo en equipo y adaptarnos a los imprevistos, demostrando nuestra capacidad de superación.

11. Conclusiones

La implementación del Travel Assistant para CTS Turismo ha sido una experiencia significativa que combina innovación tecnológica con la búsqueda de una solución efectiva para el sector del turismo. A lo largo del desarrollo del proyecto, se han cumplido los objetivos principales: integrar un asistente virtual con capacidades avanzadas de procesamiento de lenguaje natural, optimizar el proceso de reservas y mejorar la interacción usuario-sistema. A través de LangChain y modelos de lenguaje de última generación, logramos desarrollar un flujo de conversación natural y flexible, lo que permitió simplificar tareas previamente complejas para los usuarios.

Los resultados obtenidos reflejan el éxito técnico del proyecto, incluyendo la correcta integración de herramientas de backend, generación dinámica de respuestas y una experiencia de usuario accesible e intuitiva. No obstante, enfrentamos desafíos significativos, como el manejo inicial de la metodología Scrum, la curva de aprendizaje en tecnologías clave y la integración de sistemas complejos. Estos obstáculos fueron superados gracias al compromiso del equipo y a una mejora continua en habilidades técnicas y de gestión.

El Travel Assistant no solo responde a las necesidades actuales de CTS Turismo, sino que también sienta las bases para futuras mejoras y ampliaciones, permitiendo una experiencia más personalizada y eficaz. Este proyecto demuestra que la tecnología, combinada con una metodología ágil y un equipo comprometido, puede transformar procesos tradicionales en soluciones digitales que benefician tanto a la empresa como a sus clientes.