



Documentación Proyecto Final NBot

Jose Julian Pira Naranjo

Sergio Nicolas Rey Mora

Javier Esteban Martinez Girón

Fabián Espitia Sotelo

Facultad de Ingeniería

Departamento de Ingeniería de Sistemas e Industrial

Ingeniería de Software I

Oscar Eduardo Alvarez Rodriguez

Contenido

Contenido.....	2
Índice de Figuras.....	3
Introducción.....	4
Levantamiento de Requerimientos.....	5
Contexto.....	5
Análisis de Requerimientos.....	6
Must Have.....	7
Should Have.....	7
Could Have (Opcional).....	8
Won't Have (No será implementado por ahora).....	8
Requerimiento no funcionales.....	8
Must Have.....	8
Should Have.....	9
Could Have (Opcional).....	9
Won't Have (No será implementado por ahora).....	9
Análisis de gestión de Software.....	11
Triada de gestión de proyectos.....	11
Tiempo.....	11
Costo.....	14
alcance.....	18
Diseño y arquitectura.....	19
Patrones de Diseño.....	22
Conclusión.....	23
Bibliografía.....	24

Índice de Figuras

Introducción

A lo largo de la historia, la humanidad ha presenciado diferentes elementos revolucionarios que han causado grandes transformaciones en el mundo. Un ejemplo destacado es la tecnología de la computación, la cual ha experimentado un crecimiento desmesurado en capacidades y alcance desde sus inicios. En la actualidad, el software se ha convertido en una herramienta indispensable para las organizaciones y la sociedad en general, permitiendo a las personas realizar actividades de manera más eficiente y rápida, lo cual mejora la calidad de vida y aumenta la competitividad en su desempeño.

Este elemento de competitividad cobra hoy en día mayor relevancia con la aparición de la inteligencia artificial (IA), una tecnología que provee capacidades inimaginables hasta hace unos años. La IA ha desencadenado una ola de implementaciones por parte de las organizaciones en aras de mejorar procesos y aspectos como la eficiencia, la calidad, la seguridad y la toma de decisiones. Específicamente, el ámbito de las importaciones contiene procesos y tareas que pueden tomar tiempo y ser considerablemente difíciles de desarrollar, lo cual puede afectar la eficiencia de una empresa en su actividad comercial. Por ello, el presente proyecto, NBot, tiene como objetivo crear un asistente virtual inteligente que pueda proporcionar información precisa y relevante sobre importaciones hacia México, como costos, requerimientos, normas, especificaciones e impuestos, permitiendo así aumentar la competitividad de aquellas empresas o usuarios que deseen realizar esta actividad comercial, y a su vez, aprovechar las oportunidades, virtudes y mejoras que la IA ofrece.

El presente documento tiene como objetivo presentar los aspectos técnicos y conceptuales que han sido llevados a cabo en el desarrollo de un producto mínimo viable (MVP) de NBot. Se consideran puntos como tecnologías, ideación, diagramación, requerimientos y otros elementos relevantes para alcanzar dicho objetivo.

Levantamiento de Requerimientos

Contexto

La **idea** de un asistente virtual para importaciones **emerge** a partir de la observación de las dificultades que presentaba un compañero de trabajo al momento de realizar la planificación de una nueva idea de negocio de importaciones hacia el país de México, donde, los productos que planteaban mover al país, eran, computadores de distintas gamas, marcas, tipos y valores. Se evidenció que los procesos que se requieren para llevar a cabo satisfactoriamente estas actividades son bastante complejos y tediosos, en los cuales, se requiere de información abundante y de alta precisión. Razón por la cual, se necesita de expertos en importación hacia este país, esto se convierte en una **problemática**, puesto que, encontrar a estos individuos supone una tarea de alta dificultad y de alto costo, y no muy escalable para responder a una alta demanda.

Por otro lado, este inconveniente ya mencionado que se puede presentar al momento de realizar importaciones, puede **ser abordado** usando software, puesto que, en la actualidad se cuenta con múltiples herramientas, tecnologías y estándares para crear soluciones a problemas de la vida real, como lo es, el uso de IA, la cual, se convierte en un elemento de software es esencial para el desarrollo de Nbot.

La **elección** de esta idea como proyecto se determinó considerando varios factores, como tiempos, recursos y tecnologías a utilizar. Inicialmente, estos elementos se analizaron de manera general y sin mucho detalle, ya que en ese momento no se habían definido los requerimientos del proyecto. Por lo tanto, se tomó en cuenta una visión simplificada de la viabilidad de construir el asistente virtual. Además, se evaluó el nivel de motivación e interés del equipo en desarrollar este Bot. Finalmente, se realizó un censo dentro del equipo para determinar la aceptación de esta idea como proyecto de la asignatura de Ingeniería de Software I.

Los **usuarios** esperan que el sistema les proporcione información detallada y precisa sobre los productos que desean importar a México, sin tener ninguna barrera por el tipo de producto. Entre las información clave que se espera se incluyan es:

1. **Códigos Arancelarios (Tariff Code)** : Los códigos tarifarios correspondientes a los productos que se desean importar.
2. **Cálculo de impuestos:** Información sobre los impuestos aplicables, como el Impuesto General de Importación (IGI) y el Impuesto al Valor Agregado (IVA), para que los usuarios puedan prever precisamente los costos asociados a la importación.
3. **Cálculo de otros costos y cargos:** Detalles sobre otros posibles gastos relacionados con el proceso de importación, como el Derecho de Trámite Aduanero (DTA), regulaciones y

requisitos específicos, incluyendo las Normas Oficiales Mexicanas (NOMs) que puedan aplicar.

4. **Resumen de costos por unidad:** Información clara y explícita sobre los costos unitarios de importación.
5. **Requisitos adicionales para la importación:** El sistema debe incluir información sobre requisitos adicionales que varían según el tipo de producto, como Certificados de Origen, etiquetado, y regulaciones específicas para productos con características especiales (por ejemplo, baterías integradas).

El **beneficio principal** que se espera de este sistema es poder brindar toda la información necesaria para el proceso de importación. Se espera que con solo una pregunta los usuarios puedan reducir las dudas, optimizar y prever los costos y cumplir todos los requisitos legales y aduaneros sin ningún contratiempo. Toda esta información brindada debe estar completamente actualizada y conforme a las normas vigentes.

En cuanto a los **beneficios** pero ya internos del equipo. Consideramos el desarrollo de Nbot una oportunidad significativa de aprendizaje y crecimiento profesional para cada uno de nosotros. Esperamos que este proyecto nos permita poner en práctica nuestras habilidades de trabajo en equipo, experiencia en el desarrollo de software, tratando de solucionar esta problemática del mundo real, la cual puede llegar a tener un impacto significativo.

Más allá de la implementación de nuestras habilidades técnicas, este proyecto nos da la oportunidad de fortalecer habilidades fundamentales en la realización de proyectos de software bien estructurados. Habilidades como lo son la documentación clara, una planeación estructurada o el levantamiento de requerimientos. En nuestro equipo aunque tenemos algo de experiencia en la creación de software, estas habilidades de documentación son débiles, por lo que es una oportunidad perfecta, practicar estas habilidades claves que ayudan a garantizar que el sistema cumpla con las necesidades de los usuarios y por supuesto la viabilidad técnica.

Análisis de Requerimientos

Posteriormente, y partiendo del levantamiento de requerimientos se pudieron identificar las diferentes funcionalidades o características necesarias para poder cumplir con el objetivo propuesto para el asistente virtual Nbot. Se usó el método Moscow, donde, se pudo clasificar la prioridad de cada elemento a desarrollar, arrojando los siguientes resultados a partir del análisis llevado a cabo.

Requerimientos Funcionales

Prioridad	Funcionalidad	Descripción
<i>Must Have</i>	Consulta de costos e impuestos.	Tariff Code (Código arancelario) Impuestos aplicables (IGI e IVA) Cálculo de costos adicionales (DTA, NOMs, etc.)
	Normativas y requisitos específicos de importación	Regulaciones aplicables por tipo de producto Certificados de origen y etiquetado
	Interfaz de chat conversacional, funcional	Procesamiento de lenguaje natural para interpretar preguntas Capacidad de responder en español e inglés
	Autenticación de usuarios	Inicio de sesión con Google Modo invitado con acceso limitado
<i>Should Have</i>	Bases de datos estructuradas	Almacenamiento de consultas y respuestas para usuarios autenticados
	Historial de consultas accesible para usuarios autenticados	Acceso a consultas pasadas para facilitar el seguimiento
	Generación automática de etiquetas de las NOMs en imágenes	Visualización de etiquetado correcto según normativas
	Interfaz gráfica más amigable y optimizada	Mejor diseño UI/UX para facilitar la navegación

	Mejora en la detección de contexto en las preguntas del usuario	Capacidad para recordar detalles dentro de una misma conversación.
<i>Could Have (Opcional)</i>	Soporte para más idiomas además de inglés y español	Traducción automática de respuestas a otros idiomas
	Modo experto para usuarios avanzados	Opciones adicionales para consultar información técnica específica
	Exportación de datos	Generación de archivos Excel con información de importaciones
<i>Won't Have (No será implementado por ahora)</i>	Automatización del proceso de importación	Integración con plataformas de logística para gestionar envíos
	Soporte en tiempo real con agentes humanos	Conexión con expertos para responder dudas complejas
	Asesoría personalizada basada en el historial de consultas	Análisis de datos para recomendar mejores estrategias de importación

Requerimiento no funcionales

Prioridad	Funcionalidad	Descripción
<i>Must Have</i>	Rendimiento	Respuesta del bot de 3 segundos en condiciones normales de carga.
	Seguridad	Autenticación con OAuth 2.0 para inicio de sesión con Google
	Mantenibilidad	Arquitectura basada en capas y uso de contenedores Docker para portabilidad y despliegue.

<i>Should Have</i>	Escalabilidad	Capacidad para soportar 10000 consultas simultáneas sin degradación.
	Usabilidad	Diseño UI/UX y responsive para facilitar la navegación sin importar el dispositivo.
<i>Could Have (Opcional)</i>	Exportación en otro formatos	Soporte para PDF además de excel.
	Almacenar distintas conversaciones	Un apartado en el cual puedes tener una nueva conversación. Permitiendo realizar diferentes consultas de distintos productos en distintos contexto.
	Modo offline	Posibilidad de acceder a las últimas consultas sin conexión.
<i>Won't Have (No será implementado por ahora)</i>	compatibilidad con otros proveedores de autenticación	Permitir la autenticación de otros proveedores como lo son linkedin o facebook.
	Tarifas en tiempo real	Integración con APIs de terceros para obtención de tarifas en tiempo real
	Ventana de actualizaciones en las normativas	Ventana con las últimas noticias y actualizaciones en las normativas de importación.

La estimación de esfuerzo con la secuencia de Fibonacci es una técnica utilizada en metodologías ágiles para asignar puntos a las tareas o funcionalidades de un proyecto. La secuencia de Fibonacci es una serie de números en la que cada número es la suma de los dos anteriores: 1, 2, 3, 5, 8, 13, etc. Esta técnica ayuda a los equipos a evaluar la complejidad y el esfuerzo relativo de cada tarea de manera más precisa.

Funcionalidad	Esfuerzo	Justificación
Consulta de		Es un proceso simple, ya que los costos e impuestos no varían

costos e impuestos	3	tanto durante el tiempo y en caso de presentar errores, se le indica al modelo como debe corregir esta información.
Normativas y requisitos específicos de la importación	8	Se coloca una calificación de 8, ya que obtener información fiable sobre las normativas y requisitos aplicables a cierto tipo de productos es bastante complejo y requiere el apoyo de un experto en importaciones, ya que la información que abunda en internet es propensa a errores.
Detector de idiomas	2	Este es un proceso bastante sencillo y bastará con pasarle la información a un modelo de OpenAI, indicarle el formato en el que queremos la respuesta y por último, utilizarla.
Interfaz de chat conversacional, funcional	5	Se coloca una calificación de 5, ya que no es una interfaz muy compleja y no se requiere de mucho conocimiento para realizar lo planeado.
Autenticación de los usuarios	3	Se dictamina una calificación de 3 ya que la autenticación se maneja con google y este nos provee la capa de seguridad y se encarga de la mayoría del proceso.
Bases de datos estructurada	3	Al no ser una base de datos compleja y con mucha información, realizar su estructura no es un proceso que tenga mucha dificultad.
Historial de consultas accesibles para usuarios autenticados	3	Este apartado tiene una calificación de 3, ya que al tener almacenadas cada mensaje de las conversaciones asociados al id de un usuario, bastará con hacer un endpoint que retorne todos los mensajes de la conversación y que el frontend le haga la petición con el id del usuario correspondiente.
Generación automática de etiquetas de las Noms en Imágenes	3	Se toma este valor, ya que para realizar este apartado debemos conseguir ejemplos de las etiquetas y adecuarlas gráficamente para que sean lo más claras y amigables con el usuario.
Interfaz gráfica más amigable y optimizada.	3	Ya que desde un comienzo se planeó realizar un excelente código y una plataforma optimizada, no es muy complejo realizar este paso.
Mejora en la detección de contexto en las preguntas de usuario	8	Parece un proceso simple, sin embargo, para que un modelo hecho en langchain tenga memoria, se debe tener conocimiento de langGraph y entender a fondo cómo funciona el manejo de memoria basado en esta extensión del framework indicado inicialmente.

Soporte para idiomas adicionales al inglés y español	3	Ya que se planea usar una metodología semejante a la que se usa en i18n, bastará con traducir el prompt inicial al idioma deseado y con el detector de idiomas verificar si es necesario que el bot tome este idioma dependiendo al lenguaje del mensaje del usuario.
Modo experto para usuarios avanzados	13	Ya que este apartado planea resolver preguntas muy específicas de los usuarios, el modelo deberá conectarse a otras fuentes (las cuales también hay que desarrollar) para obtener esta información.
Exportación de datos (Excel)	8	Se coloca esta calificación, ya que este proceso implica capturar la información con expresiones regulares, IA y con pandas organizar la información en el excel para que sea lo más ordenada posible.
Automatización del proceso de importación	13	Este proceso tiene un gran grado de dificultad, ya que debemos contactar y hacer acuerdos con agencias de logística mexicanas para que podamos implementar esta funcionalidad.
Asesoría personalizada basada en el historial de consultas	13	La complejidad de esta tarea radica en la combinación de procesamiento de datos históricos, análisis predictivo y personalización dinámica, lo que la hace desafiante y requiere una implementación robusta.

Análisis de gestión de Software

En un desarrollo de un nuevo producto o funcionalidad se debe llevar a un conjunto de actividades que permitan realizar una correcta gestión, planificación y ejecución de las distintas fases de se tengan dentro de un proyecto. Para NBot puesto que se plantea su desarrollo a modo de MVP, solo se consideraron aquellas funcionalidades o características mínimas para obtener el producto mínimo viable y que otorga una visión suficientemente completa de las pretensiones del proyecto como tal. Consecuentemente, se realizó el siguiente análisis en la tríada de la gestión de proyectos.

Triada de gestión de proyectos

La triada de gestión de proyectos será utilizada como modelo que apoya la descripción de las restricciones principales que existen en cualquier proyecto, en especial, aquellas que se consideran para Nbot, las cuales son las siguientes.

Tiempo

Para esta restricción su evaluación comienza con la determinación del tiempo que se tiene para el desarrollo del MVP, donde, se contaban con **10** semanas para su construcción, lo cual, requirió una delimitación específica en cada las etapas que se debían llevar a cabo, razones por las cual, se toma la decisión de segmentarlo de la siguiente manera.

- 2 Semanas para la planificación, levantamiento y análisis de requerimientos.
- 1 Semana para diseño del sistema.
- 6 Semanas para el desarrollo e implementación.
- 1 Semana de validación y documentación.

Cada uno de estos puntos, se consideraron a modo de etapas, las cuales en su totalidad constituyeron el cronograma del proyecto. En cada una de las etapas se llevaron a cabo elementos de entendimiento del equipo respecto a la nueva fase a realizar, lo que se esperaba en cada una y las responsabilidades de cada integrante.

Teniendo esto en cuenta se eligieron las funcionalidades que como equipo se podrían abordar en el tiempo limitado que se disponía, se eligieron las siguientes funcionalidades y su duración estimada de desarrollo:

- Funcionalidades de tipo **Must Have**:
 - Consulta de costos e impuestos 3 puntos
 - Normativas y requisitos específicos de la importación 8 puntos
 - Detector de idiomas 2 puntos
 - Interfaz de chat conversacional, funcional 5 puntos
 - Autenticación de los usuarios 3 puntos
 - Bases de datos estructurada 3 puntos
- Funcionalidades de tipo **Should Have**:
 - Historial de consultas accesibles para usuarios autenticados 3 puntos
 - Generación automática de etiquetas de las Noms en Imágenes 3 puntos
 - Interfaz gráfica más amigable y optimizada 3 puntos
 - Mejora en la detección de contexto en las preguntas de usuario 8 puntos
- Funcionalidades de tipo **Could Have**:
 - Soporte para idiomas adicionales al inglés y español 3 puntos
 - Exportación de datos (Excel) 8 puntos

Por otro lado, se descartaron las siguientes funcionalidades, pues, por el corto tiempo disponible se consideraron prescindibles, adicionalmente, se explica en detalle cada una de las razones.

- Funcionalidades de tipo **Could Have**:
 - Modo experto para usuarios avanzados 13 puntos

- Funcionalidades de tipo **Won't Have**:

- Automatización del proceso de importación 13 puntos
- Soporte en tiempo real con agentes humanos13 puntos
- Asesoría personalizada basada en el historial de consultas 13 puntos

Para llevar a cabo esta determinación de funcionalidades a considerar para el MVP primero, se tomó en cuenta la prioridad después del análisis de Moscow. Posteriormente, teniendo en cuenta que se contaba con 6 semanas para la realización del desarrollo y ejecución del proyecto, se hizo uso de Chat GPT como apoyo para poder calcular correctamente el tiempo necesario para cada tarea considerando los puntos asignados mediante Fibonacci. De esta manera, se recopiló la siguiente información:

- Se cuenta con un equipo de trabajo compuesto por 4 personas (desarrolladores), adicionalmente, se tienen 6 semanas para la ejecución del desarrollo, por lo cual, se pueden tener 3 sprints de 2 semanas, en los cuales se distribuyen las funcionalidades a realizar. Por otro lado, la sumatoria de los puntos es 54, lo cual implica que cada integrante debe desarrollar 13 puntos de historias de usuario, para así ser equitativos con el esfuerzo. Se organizan los sprints de la siguiente forma:

- **Sprint 1 (Semana 1 y 2) - 19 puntos:**

- Normativas y requisitos específicos de la importación (**8 puntos**).
 - Responsable: integrante 1.
- Consulta de costos e impuestos (**3 puntos**).
 - Responsable: integrante 2.
- Interfaz de chat conversacional (**5 puntos**).
 - Responsable: integrante 4.
- Autenticación de los usuarios (**3 puntos**).
 - Responsable: integrante 3.

- **Sprint 2 (Semana 3 y 4):**

- Historial de consultas accesibles (**3 puntos**).
 - Responsable: integrante 1.
- Detector de idiomas (**2 puntos**).
 - Responsable: integrante 3.
- Soporte para idiomas adicionales al inglés y español (**3 puntos**).
 - Responsable: integrante 2.
- Bases de datos estructurada (**3 puntos**).
 - Responsable: integrante 4.

- **Sprint 3 (Semana 5 y 6):**

- Generación automática de etiquetas de las Noms en Imágenes (**3 puntos**).
 - Responsable: integrante 1.

- Exportación de datos (**8 puntos**).
 - Responsable: integrante 2.
- Mejora en la detección de contexto (**8 puntos**).
 - Responsable: integrante 3.
- Interfaz gráfica más amigable y optimizada (**3 puntos**).
 - Responsable: integrante 4.

A partir de esta organización se construyó una tabla que permitiera identificar la cantidad de puntos que serían asignados a cada integrante, y de esta forma, poder mantener un valor equitativo de esfuerzo entre todo el equipo. También cabe destacar que algunos tienen que aportar algo más de trabajo que otros, pero esto se equilibra con los distintos niveles de conocimiento y experiencia. Siguiendo se muestra la tabla.

Resumen de puntos de sprints por persona				
Equipo	Sprint 1	Sprint 2	Sprint 3	Total
Integrante 1	8	3	3	14
Integrante 2	3	3	8	14
Integrante 3	3	2	8	13
Integrante 4	5	3	3	11

Costo

Para poder analizar correctamente este aspecto, debemos partir de la identificación de aquellos elementos o recursos que son necesarios para el desarrollo del proyecto y que tienen un costo asociado a su uso o disposición. En el desarrollo de software se pueden considerar los siguientes:

- **Recursos humanos:** son los costos relacionados a los integrantes del equipo y su contratación.
- **Licencias de Software:** gastos asociados a licencias de herramientas de software que se puedan requerir en el desarrollo proyecto.
- **Cuentas o usuarios de servicios de nube:** asignación de cuentas de nube para realizar las diferentes pruebas que requieran el uso de servicio cloud.
- **Infraestructura:** costos de servidores, almacenamiento en la nube, y otros recursos tecnológicos necesarios.

Adicionalmente, para el desarrollo del proyecto NBot ya se han definido el uso de algunas tecnologías específicas, por lo cual, también es un insumo en la estimación del costo de la realización del proyecto, siendo estas las siguientes:

- **FrontEnd:**

- **Next.js:** Framework de React para el desarrollo de aplicaciones web modernas.
- **Tailwind CSS:** Sistema de utilidades CSS para crear interfaces de usuario elegantes y responsivas.

- **BackEnd:**

- **FastAPI:** Framework rápido para construir APIs modernas y eficientes.
- **LangChain:** Herramienta para el desarrollo de aplicaciones basadas en modelos de lenguaje.
- **LangGraph:** Solución para modelar y gestionar flujos de trabajo complejos.

- **Base de datos:**

- **PostgreSQL:** Sistema de gestión de bases de datos relacional de alto rendimiento.

- **Infraestructura y automatización:**

- **Google Cloud Platform:** Como cloud provider a utilizar en el desarrollo.
- **Docker:** Como tecnología de contenerización o virtualización de componentes.
- **CI/CD:** Herramientas como GitHub Actions o Azure DevOps.

Considerando lo anteriormente expuesto y las demás necesidades del proyecto, se han creado las siguientes tablas a modo de síntesis de la información relacionada con el costo. Es importante destacar que estas estimaciones o análisis se realizan desde una perspectiva investigativa y no se utilizarán para el presupuesto del desarrollo actual del proyecto. Estos datos servirán para considerar desarrollos futuros, así como para ampliar y concientizar sobre el costo real asociado que puede acarrear el desarrollo de un proyecto.

Tabla de costos Recursos Humanos (Equipo)			
Recurso	Justificación	Responsabilidades	Costo
Desarrollador Full Stack	Necesitamos un desarrollador que pueda trabajar tanto en el frontend como en el backend. Esto asegura una integración fluida entre las diferentes capas del sistema y permite una mayor flexibilidad en el desarrollo.	Desarrollo de la interfaz de usuario (Next.JS y Tailwind CSS), implementación de la lógica del servidor (FastAPI y LangChain), y conexión con la base de datos (PostgreSQL).	Salario mensual promedio: <ul style="list-style-type: none"> ● Talent: \$4.500.000 ● Glassdoor: \$4.000.000

Tabla de costos Recursos Humanos (Equipo)			
Especialista en Inteligencia Artificial	La IA es el núcleo del asistente virtual, por lo que es crucial contar con un experto en esta área para desarrollar y optimizar los modelos de IA.	Desarrollo y entrenamiento de modelos de IA, implementación de procesamiento de lenguaje natural (NLP), y mejora continua de la precisión y relevancia de las respuestas.	Salario mensual promedio: <ul style="list-style-type: none"> • Talent: \$6.500.000 • Glassdoor: NA
Ingeniero de Datos	La precisión y relevancia de la información proporcionada por el asistente virtual dependen de la calidad de los datos. Un ingeniero de datos es esencial para gestionar y optimizar la base de datos.	Diseño y mantenimiento de la base de datos (PostgreSQL), aseguramiento de la calidad de los datos, y optimización de consultas y almacenamiento.	Salario mensual promedio: <ul style="list-style-type: none"> • Talent: \$4.273.000 • Glassdoor: \$4.000.000
Diseñador UX/UI	Una interfaz de usuario intuitiva y atractiva es fundamental para la aceptación y uso del asistente virtual por parte de los usuarios.	Diseño de la interfaz de usuario, mejora de la experiencia del usuario (UX), y creación de prototipos y maquetas.	Salario mensual promedio: <ul style="list-style-type: none"> • Talent: \$3.629.444 • Glassdoor: \$4.200.000
Especialista en Seguridad	La seguridad de los datos y la protección contra posibles amenazas son aspectos críticos en cualquier proyecto de software.	Implementación de medidas de seguridad, auditorías de seguridad, y gestión de autenticación y autorización de usuarios.	Salario mensual promedio: <ul style="list-style-type: none"> • Talent: \$3.000.000 • Glassdoor: \$8.000.000
Especialista en Importaciones	Un gerente de proyecto es esencial para coordinar las actividades del equipo, asegurar el cumplimiento de los plazos y gestionar los recursos de manera eficiente.	Planificación y seguimiento del proyecto, gestión de riesgos, y comunicación con las partes interesadas.	Salario mensual promedio: <ul style="list-style-type: none"> • Talent: \$2.750.000 • Glassdoor: NA
Gerente de Proyecto	Dado que el asistente virtual se centra en proporcionar información sobre	Asesoramiento sobre normativas y requisitos de importación,	Salario mensual promedio:

Tabla de costos Recursos Humanos (Equipo)			
	importaciones, es crucial contar con un experto en esta área para asegurar la precisión y relevancia de la información.	validación de la información proporcionada por el asistente, y actualización de datos según cambios en las regulaciones.	<ul style="list-style-type: none"> ● Talent: \$6.500.000 ● Glassdoor: \$7.000.000
Total (Para las 10 semanas)			\$84.503.055

Tabla de costos Licencias, Cuentas Nube e Infraestructura			
Recurso	Justificación	Especificación	Costo
Tecnologías BackEnd	Herramientas y Tecnologías para el desarrollo de componente BackEnd	FastAPI y Langhain - Son tecnologías de código abierto y por ende gratuitas. LangGraph - LangGraph es una biblioteca de código abierto y gratuita bajo la licencia MIT. Aunque, podría llegar a ser de pago en el caso de LangGraph Cloud es de pago. Editor de código Visual Studio Code - Gratuito	No habría asociados a esta categoría.
Tecnologías FrontEnd	Herramientas y Tecnologías para el desarrollo de componente FrontEnd	Next.js y Tailwind CSS - Son tecnologías de código abierto.	No habría costos relacionados en este aspecto.
Base de Datos PostgreSQL	Instancia de PostgreSQL para realización de pruebas	Instancia Cloud SQL con las siguientes características: <ul style="list-style-type: none"> ● 4 vCPUS, 8 GB RAM y 512GB HDD 	Costo mes: \$1.033.440
Licencia para Jira	Herramienta para gestión de proyectos y seguimientos de tareas	Versión premium que cuenta con un buen número de funcionalidades. <ul style="list-style-type: none"> ● Valor mensual por usuario: 70692 ● Son 7 usuarios 	Costo mes: \$494.844

Tabla de costos Licencias, Cuentas Nube e Infraestructura			
Licencia GitHub/GitHub Actions	Plataforma de control de versiones CI/CD	Versión Team que incluye las funcionalidades que se podrían requerir: <ul style="list-style-type: none"> • Valor mensual de \$16.633 • Son 7 usuarios 	Costo mes: \$116.431
Cuentas de Google Cloud y Infraestructura a contratar para el proyecto	Recursos y servicios necesarios para desarrollo, pruebas y despliegue	Servicios que posiblemente se utilizarían: <ul style="list-style-type: none"> • Compute Engine <ul style="list-style-type: none"> ◦ 2 Instancias de 4 vCPUs y 15 de RAM - \$200 USD/mes • Cloud Storage <ul style="list-style-type: none"> ◦ 1 TB de almacenamiento - \$20 USD/mes • Cloud Run <ul style="list-style-type: none"> ◦ 1 vCPU y 1GiB - \$52 USD/mes • BigQuery <ul style="list-style-type: none"> ◦ 1 TB \$5 USD/mes • Vercel <ul style="list-style-type: none"> ◦ Plan pro \$20 USD/mes 	Costo mes: \$1.235.038
Total (Para las 10 semanas)			\$7.199.382

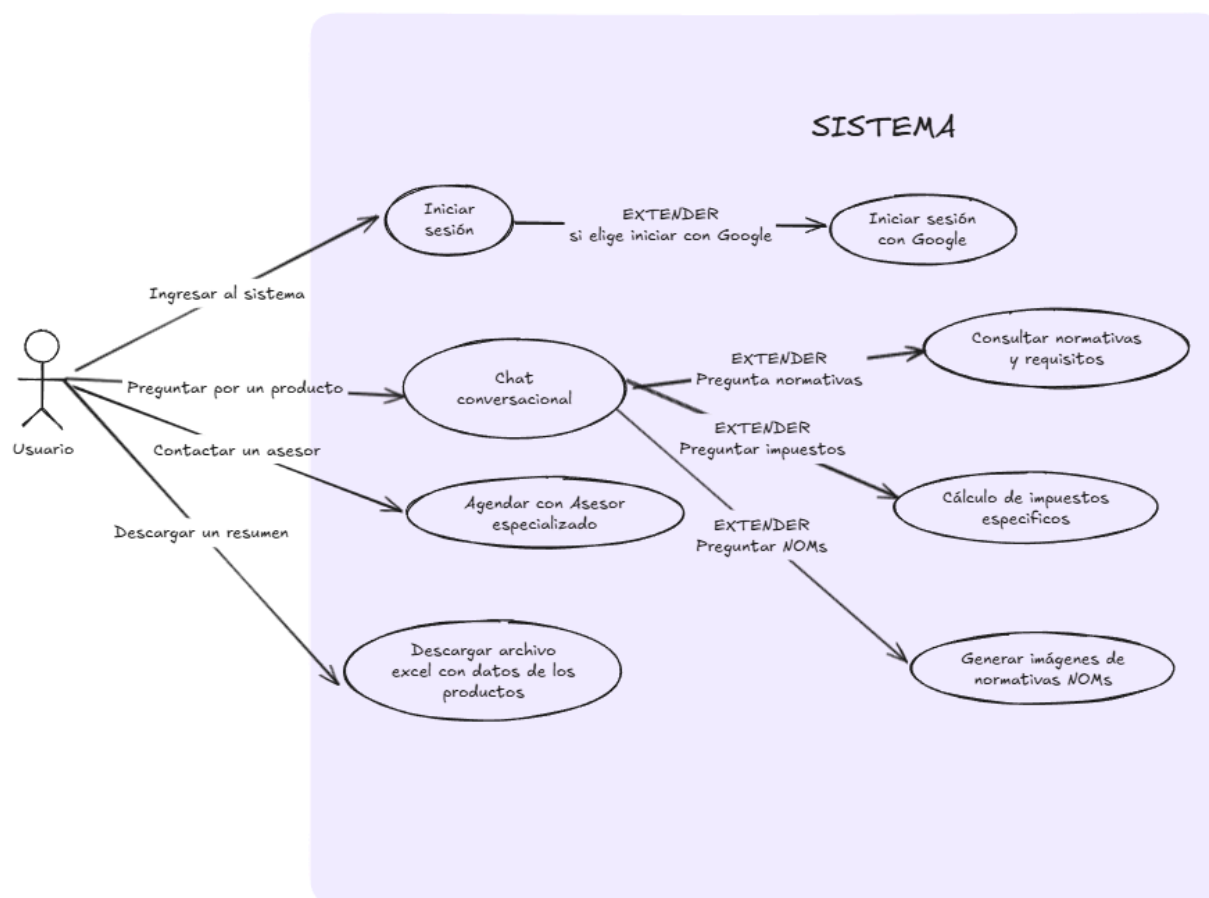
Posteriormente a la construcción de estas tablas, se ha realizado una estimación desde la perspectiva del equipo sobre cuánto podría llegar a costar la ejecución de un desarrollo como el planteado para el asistente de importaciones en NBot. Se obtuvo un costo estimado de \$91.702.437 para un período de 10 semanas. Es importante destacar que esta estimación se realizó de manera investigativa y no se utilizará para el presupuesto real del proyecto. La estimación incluye varios elementos que, si no se consideran, podrían llevar a un costo mucho más contenido. Estos datos servirán para considerar desarrollos futuros y para ampliar la conciencia sobre el costo real asociado al desarrollo de un proyecto de esta magnitud.

Para el desarrollo real del proyecto únicamente se dispondrá de los 4 integrantes del equipo, adicionalmente, se tratará en todo momento de buscar alternativas gratuitas en el uso de los recursos necesarios, tal como es: uso de licencias gratuitas temporales de cuentas de la nube si el caso, para poder mantener el proyecto sin costo alguno.

alcance

Como se ha mencionado anteriormente en la sección de tiempos de la triada ya se han definido las funcionalidades que se considerarán para el MVP, por lo cual este sería el alcance para el proyecto, donde se consideran 10 semanas, y se descartaron funcionalidades las cuales se contemplaban en la sección won't have.

Diagrama casos de uso



Diseño y arquitectura

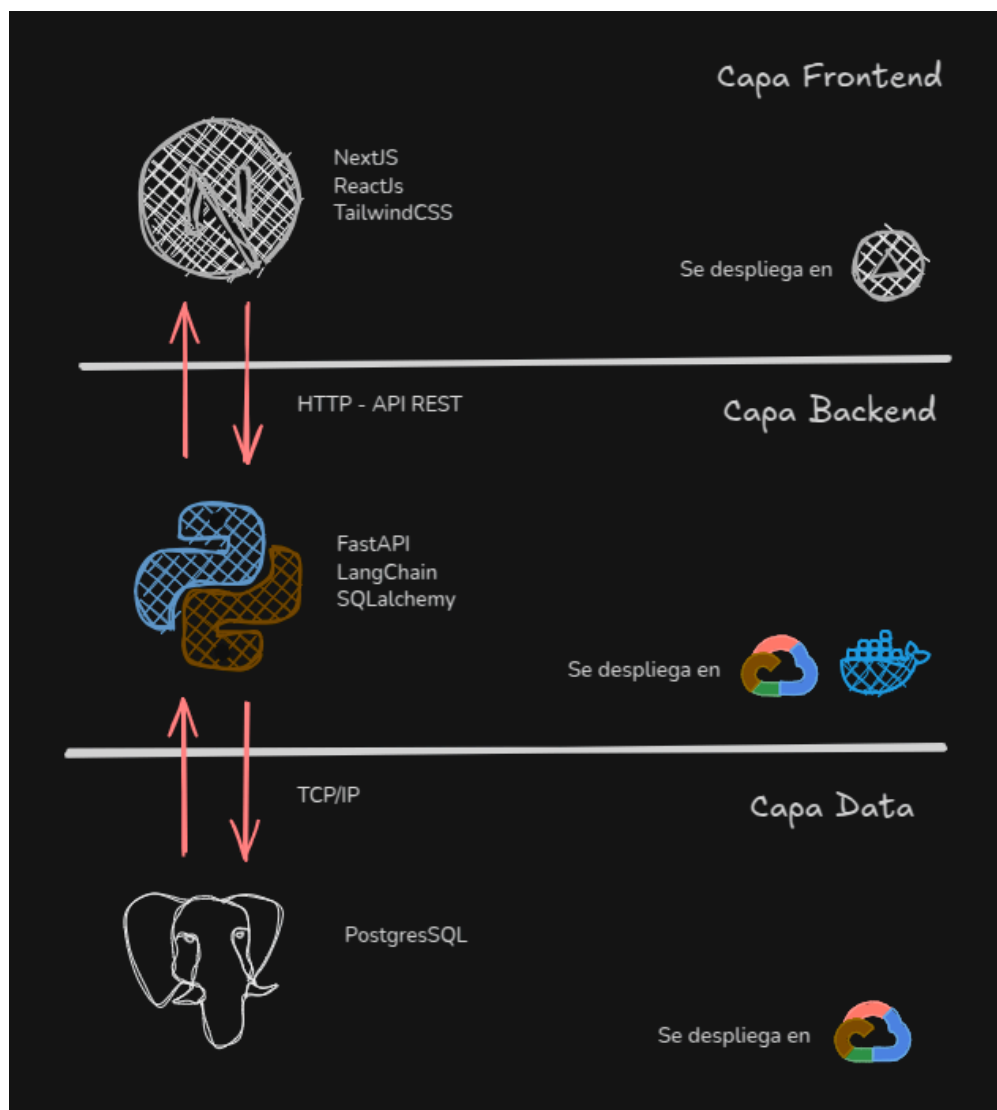
El proyecto **NBOT** está diseñado con una arquitectura cliente-servidor, más específicamente una arquitectura de 3 capas, ya que actualmente el sistema se compone de las siguientes 3 capas:

- Cliente - Frontend
- Servidor - Backend
- Data - Base de Datos

Además, cada capa se comunicada de manera jerárquica y únicamente, con las capas contiguas, por ende el *cliente* únicamente se logra comunicar con la capa del *servidor* o backend a través del protocolo HTTP/3, mediante una API REST, donde el *servidor* está en espera de solicitudes por parte del *cliente*, y esta solamente es consumida por esta capa ya que es la superior, lo mismo con la capa de *datos*. La base de datos únicamente es llamada por parte de la capa contigua, el *servidor*, ya que esta es la capa superior y contiene la lógica del negocio.

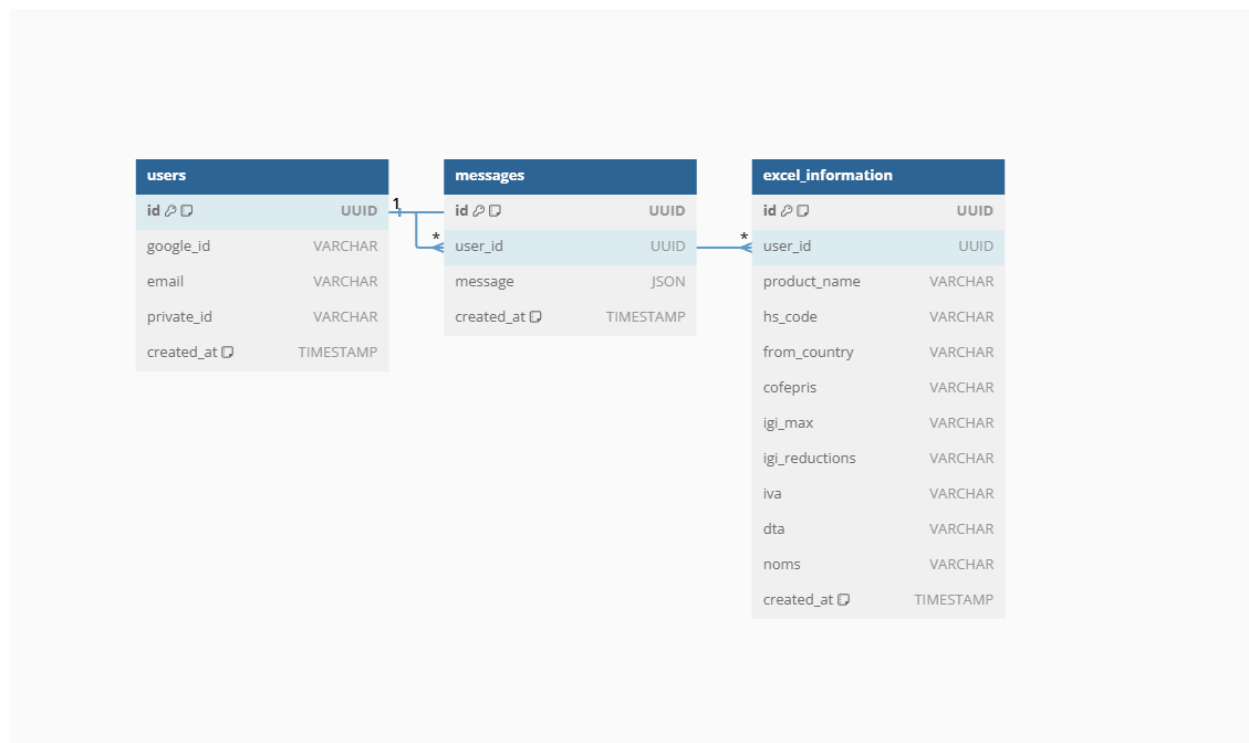
La capa de *datos* o la base de datos no consume ninguna otra capa ya que está, a nivel jerárquico es la de menor nivel, por lo tanto, no puede realizar ningún llamado o uso de otra capa.

La siguiente imagen ilustra la arquitectura por capas realizada en el desarrollo del proyecto:



La anterior arquitectura fue seleccionada ya que nos otorgaba la flexibilidad de poder utilizar tecnologías más afines a la experiencia de desarrollo del equipo encargado y a las necesidades mismas del proyecto, por lo tanto las personas encargadas del desarrollo el frontend les resultaba mucho más cómodo y sencillo la realización de un producto mucho más robusto que generará una experiencia de mayor calidad a los usuarios, ya que a través de esta arquitectura se pudo utilizar tecnologías enfocadas especialmente al desarrollo web, otorgando una gran comunidad y paquetes o librerías con este fin. Lo mismo para la capa del backend, se utilizaron tecnologías especializadas en el desarrollo backend y manejo de sistemas de AI, generando que si se encuentra algún error o complicación sea fácilmente poderlo solucionar.

Modelo entidad relación



El diseño del esquema sigue una estructura normalizada para evitar redundancia y mejorar la integridad de los datos. Se divide en tres tablas (users, messages y excel_information), donde users es la tabla principal y las otras dos contienen información relacionada referenciada por user_id. Esto permite una mejor organización y evita inconsistencias.

Las claves primarias y foráneas están definidas con UUID, asegurando identificadores únicos y evitando colisiones. user_id en messages y excel_information establece relaciones 1:N con users, garantizando la integridad referencial.

Se ha elegido una base de datos relacional (SQL) debido a la necesidad de integridad y consistencia en los datos. SQL nos permite definir relaciones claras con claves foráneas y facilita consultas estructuradas mediante JOINS. Además, ofrece transacciones ACID y una estructura predecible, lo que es ideal para manejar datos bien definidos como usuarios, mensajes y registros de productos.

Patrones de Diseño

PENDIENTE POR REVISAR

Conclusión

PENDIENTE POR REVISAR

Bibliografía

PENDIENTE POR REVISAR