

**UNIVERSIDAD AUTONOMA GABRIEL RENE MORENO
FACULTAD DE INGENIERIA EN CIENCIAS DE LA COMPUTACION Y
TELECOMUNICACIONES**



2DO EXAMEN PARCIAL

ESTUDIANTE: TORREZ CONDARCO CARLOS EDUARDO

MATERIA: INGENIERIA DE SOFTWARE I INF422-SC

DOCENTE: MARTINEZ CANEDO ROLANDO ANTONI

1. Introduccion.

Hoy en día Internet está muy masificado, resulta difícil destacar entre tanta información. Las empresas buscan los canales digitales para llamar la atención y tener contacto con los usuarios. Los consumidores aceptan este hecho y prefieren tener el contacto con las empresas a través de mensajes.

Adiós a los medios digitales de antes, los medios tradicionales como los call center, por ejemplo, ya están obsoletos. La solución que se ha planteado para hacer feliz tanto a las empresas como a los consumidores son: los chatbots.

Los chatbot dan la oportunidad de expandir el canal digital de las empresas y están impactando positivamente la experiencia de usuario. La tecnología de chatbot está aquí y tiene un rol importante en el futuro de las interacciones entre empresas y clientes. Preparémonos empezando por desarrollar una hoja de ruta para incorporar chatbots en la experiencia digital de nuestros clientes.

De manera sencilla y comprensible podemos definir un chatbot como un asistente que se comunica con los usuarios a través de mensajes de texto. En muchas otras ocasiones, toma forma convirtiéndose en un compañero virtual que se integra en sitios web, aplicaciones... conversando y ayudando a los usuarios.

Se trata de una tecnología que permite al usuario mantener una conversación a través de un software que se integra en un determinado sistema de mensajería, como, por ejemplo: Facebook, Twitter, Telegram, Whatsapp, etc.

El sistema está programado para que interactúe con el cliente y le resuelva dudas, pero sin que haya una persona física contestando. Tienen la ventaja de que están disponibles siempre para resolver las dudas de los usuarios que quieran contactar contigo a cualquier hora del día.

Los algoritmos desarrollados por la inteligencia artificial y de aprendizaje automático permiten que los chatbots sean capaces de aprender. Pueden llegar a intuir los hábitos y entender los gustos y preferencias de los usuarios.

2. Antecedentes.

2.1. Antecedentes generales.

El desarrollo de software se puede realizar de muchas maneras, hay muchos procesos de desarrollo de los proyectos. Desde la modelo cascada, hasta los procesos basados en objetivos, pasando por los procesos iterativos e incrementales. Hasta las metodologías ágiles.

Durante mucho tiempo el desarrollo de software siguió un proceso en cascada, en este modelo analizamos los requisitos, diseñamos y desarrollamos, probamos y por último desplegamos el sistema, una de las principales limitaciones del modelo en cascada es que no permite una retroalimentación en cada una de sus etapas.

Por todas las limitaciones del modelo en cascada surge el concepto de desarrollo iterativo e incremental, un proyecto puede verse como una serie de pequeñas cascadas, realizando entregas incrementales a los usuarios finales. De los cuales recibimos retroalimentación para la siguiente iteración. Esto es bueno en sistemas que sus requisitos no cambian mucho. Pero es un verdadero problema en sistemas que están en constantes cambios.

Por esta razón surgen las metodologías ágiles. Como SCRUM, Programación extrema (XP), Kanban, etc. Cuya visión se centra en el manifiesto ágil. Cuya visión se centra en el manifiesto ágil. Estas metodologías se enfocan en el trabajo en equipo. Donde se considera al cliente como parte del equipo donde se realizan entregables e iteraciones cortas, para que el cliente o el usuario constantemente realice retroalimentaciones, para construir un producto de calidad de acuerdo a sus especificaciones y necesidades.

La comunicación entre los clientes, usuarios y desarrolladores es muy importante para construir el sistema, los clientes exponen sus necesidades comerciales y los desarrolladores los plasman en requisitos funcionales y estos requisitos son transformados en código fuente. A esto se lo conoce como modelado.

2.2. Antecedentes Específicos.

Todo software cambia, los requisitos cambian, el diseño cambia, el negocio cambia, la tecnología cambia, el equipo cambia, los miembros cambian, el problema no es el cambio en sí mismo, puesto que sabemos que el cambio va a suceder. El problema es la incapacidad de adaptarnos a dicho cambio cuando este tiene lugar.

SCRUM es un marco de trabajo ágil que está basado en un pequeño conjunto de valores, principios y prácticas. Scrum ha ganado popularidad en estos últimos años ya que se adapta bien en proyectos, cuyos requisitos cambian constantemente, proyectos no predecibles y proyectos complejos y con gran incertidumbre.

Con el fin de aplicar estos conceptos en un ejemplo práctico, esta documentación tiene como objetivo desarrollar una aplicación chatbot que permite la comunicación entre el usuario y la aplicación de manera rápida y fluida, aplicando la notación del modelado visual UML y la metodología ágil SCRUM.

Para lograr alcanzar estos objetivos se llevará a cabo las siguientes tareas:

- Elaborar un product vision board para entender mejor la aplicación que se va a desarrollar y determinar la visión del proyecto.
- Capturar los requerimientos funcionales, para determinar el ámbito de la aplicación través de historias de usuarios escritas por los mismos clientes o por el product owner.
- Planificar las iteraciones necesarias para realizar la entrega de incrementos de calidad que proporcionen valor al cliente.
- Estimar el esfuerzo y el tiempo de las historias de usuario con el método puntos historia y planning póker.
- Identificar los diagramas UML necesarios para modelar de manera visual los artefactos de la aplicación.
- Utilizar herramienta de control de versiones para llevar un control adecuado de los cambios a lo largo del desarrollo de la aplicación.

3. Planteamiento del problema

muchas veces se malinterpreta lo que se manifestó de las metodologías ágiles referente al valor “software que funciona sobre documentación exhaustiva”. Esto no significa que se descarte toda documentación, sino que se logre obtener la documentación adecuada y necesaria que sirva de ayuda tanto para los miembros del equipo scrum como para los stakeholder.

4. Descripción del problema.

hoy en día el mundo está afrontando una crisis mundial sanitaria que ha afectado a millones y millones de personas, lo cual ha hecho más frecuente y masiva la interacción entre las personas y los centros de salud, las clínicas, las postas sanitarias, etc.

En esta ocasión nos enfocaremos en las clínicas, la problemática que ha surgido por el incremento de solicitudes de atención a los clientes, las clínicas han colapsado y muchas veces el personal no abastece para atender las solicitudes, lo cual hace que se pierdan clientes.

A continuación, voy a describir los problemas básicos que tienen las clínicas para gestionar las solicitudes de los clientes:

- Cada personal encargado de la atención al cliente podrá atender una solicitud a la vez.
- Cuando existen muchas solicitudes que sobrepasan al personal encargado, se pierden clientes.
- Los personales encargados de las solicitudes de los clientes se tardan mucho en atender a un cliente.
- Muchas veces las solicitudes de los clientes, son preguntas básicas, que no necesita mucha relevancia la respuesta, se pierde de atender a otros clientes que precisen atención sobre su salud.

5. Objetivos del problema.

5.1.Objetivo General.

Desarrollar una aplicación móvil para un Hospital, La aplicación tiene que tener un Chabot, para cumplir con las tareas habituales de un Call Center

5.2.Objetivos Especifico.

- Realizar la captura de los requerimientos funcionales de la aplicación móvil.
- Analizar los requerimientos obtenidos para cubrir las necesidades de la clínica.
- Utilizar el marco de trabajo SCRUM para el desarrollo del software.
- Utilizar Flutter para el desarrollo de la aplicación móvil
- Utilizar la herramienta de DialogFlow para crear una IA y entrenarla para cubrir las necesidades del call center.
- Diseñar e implementar una base de datos no SQL en tiempo real para la aplicación
- Diseñar una UI amigable para el usuario

6. Metodología de Investigación.

se utilizan los siguientes métodos de investigación, para la elaboración del presente trabajo.

6.1.Método bibliográfico.

Para el proceso de investigación bibliográfica se debe contar con material informativo como libros, revistas de divulgación o de investigación científica, sitios Web y demás información necesaria para iniciar la búsqueda.

7. Justificación.

uno de los valores del manifiesto ágil indica que “software que funciona sobre documentación exhaustiva”, este manifiesto no indica que se debe prescindir de toda la documentación más bien se debe elaborar una documentación mínima y necesaria para el desarrollo y mantenimiento de la aplicación.

SCRUM se basa en el desarrollo del proyecto y sus iteraciones en sus historias de usuario que son narraciones textuales que deben tener las características de la aplicación. Las historias de usuarios al ser pequeñas e independiente, a veces se vuelven complejas de comprender cuando se trata de ver el funcionamiento o el objetivo global de la iteración o del sistema y como estas características trabajan entre sí. En este contexto UML al ser una notación grafica de modelado mostraría de manera clara el funcionamiento global de la aplicación y ayudaría a comprender mejor los objetivos de la iteración.

8. Alcance y limitaciones.

Se desarrollará una aplicación móvil para una clínica, que integre un chatbot para la atención a las solicitudes de los clientes de manera fluida y rápida.

Se utilizará SCRUM como marco de trabajo de desarrollo ágil y UML como notación de modelado visual.

Se propone una aplicación móvil multiplataforma que se desarrollara bajo el patrón de arquitectura modelo vista proveedor (MVP). Se utilizará el SDK FLUTTER como entorno de desarrollo, Firebase como backend y una realtime database como gestor de base de datos.

A) Alcances

- Registro de usuarios.
- Inicio de sesión de usuarios.
- Asistente de chat inteligente para el call center de usuarios.

B) Limitaciones.

Para mantener un proyecto pequeño y enfocándose únicamente en la atención de las solicitudes de los usuarios:

- No tendrá videollamadas para las teleconsultas
- No tendrá una forma de método de pago online
- No administrara el historial clínico del paciente
- Solo cumplirá el rol de una simple secretaria, como agendar citas, derivar llamadas, dar información básica sobre el hospital

9. Marco teórico.

9.1. Manifestó Ágil.

Como consecuencia de una reunión donde se acuño el termino de metodología ágiles, se establecieron los valores de estas metodologías, agrupándolos en 4 postulados. Quedando esta agrupación denominada como manifestó ágil. A continuación, se menciona los cuatros postulados de este manifestó:

- Valorar más a los individuos y su interacción a los procesos y las herramientas.
- Valorar más el software que funciona que la documentación exhaustiva.
- Valorar más la colaboración con el cliente que la negociación contractual.
- Valorar la respuesta al cambio que el seguimiento de un plan.

9.2. SCRUM

es un marco de trabajo para la gestión y desarrollo de productos complejos, en un proceso iterativo e incremental utilizado comúnmente en entornos donde existe gran incertidumbre. Scrum está basado en un pequeño conjunto de valores, principios y prácticas que se basan en los principios de inspección y adaptación, enfocado más en el aspecto humano y el trabajo en equipo donde se involucra totalmente al cliente en todo el proceso.

9.3. Historias de Usuario.

es una narración que describe una funcionalidad del software que tiene valor para un usuario, las historias de usuario deben ser independientes, negociables, evaluables, estimables, pequeñas y se las deben probar.

9.4. Lenguaje de Modelado Unificado.

UML es un lenguaje de propósito general para visualizar, especificar, construir y documentar los artefactos de un sistema que involucre una gran cantidad de software.

UML es como una caja de herramientas, no es necesario utilizar todos los diagramas, mas solo los necesarios que te ayuden a resolver el problema.

9.5.Product visión board.

El product visión board (PVB) es una herramienta visual plasmada en un canvas, creada por roman pitcher que permite presentar la visión de forma que se entendimiento sea más fluido y además respondiendo preguntas claves que le servirán al equipo para mantener y utilizar la visión.

9.6.Puntos de Historia.

El método de los puntos de historia es la forma de realizar estimaciones de esfuerzo y tiempo a las historias de usuario. También nos sirve para medir la velocidad del sprint en la reunión de retrospectiva.

Se escoge una historia de usuario sencilla una que todo el mundo entiende para emplearla como referencia. Esa sería la definición de 1 punto de historia del proyecto para estimar las otras historias de usuario, bastara con compararlo con la historia de referencia, se emplea la actividad de planning póker para determinar el valor de los puntos y el tiempo de las historias de usuario según el siguiente cuadro.

Size	points	Time [days]
Xtra Small (XS)	1	½
Small (S)	2	1
Medium (M)	3	2
Large (L)	5	3
Xtra Large (XL)	8	5

10. Marco Practico.

10.1. Modelo de negocio.

La aplicación propuesta según los alcances y limitaciones descritos está orientado para cualquier clínica que necesite o precise de un asistente (chatbot) para la atención en línea de las solicitudes de sus usuarios clientes.

Hacemos uso del product visión board para comprender mejor la visión del producto que se va a desarrollar.

10.1.1. Producto visión board.

Agente (chatbot) para la atención en línea de solicitudes de usuarios clientes de manera rápida y confiable y fluida.			
Usuarios	Necesidades	Producto	valor
<ul style="list-style-type: none">• Clínicas• Hospital• Centro de salud.	<ul style="list-style-type: none">• Atender las solicitudes de los usuarios clientes de manera rápida y fluida.• Atención de consultas generales• derivación de consultas	<ul style="list-style-type: none">• Aplicación móvil.• Registro de usuarios• Inicio de sesión de los usuarios.• Agente (chatbot) que atienda solicitudes	<ul style="list-style-type: none">• Respuestas en línea a los usuarios.• Conversaciones y diálogos rápidos y fluidos.

	con especialistas.	de los usuarios clientes.	
--	-----------------------	---------------------------------	--

10.2. Roles y stakeholders.

Product owner: Carlos Eduardo Torrez Condarco.

Scrum master: Carlos Eduardo Torrez Condarco.

Development Team: Carlos Eduardo Torrez Condarco.

Client: Clínica, Hospital, centro de salud.

10.3. Historias de Usuario.

Para la determinación de requerimientos y/o características de la aplicación se recolectan historias de usuarios.

HU1: Ingresar a la aplicación como usuario cliente	
Como:	Usuario cliente
Quiero:	Ingresar a la aplicación con un email y password.
Para:	Para tener acceso a todas las funcionalidades de la aplicación

HU2: Registrarse en la aplicación como usuario cliente.	
Como:	Usuario cliente
Quiero:	registrarme en la aplicación con un email y contraseña
Para:	Para poder ingresar a la aplicación como usuario cliente.

HU3: atención al cliente	
Como:	Usuario cliente
Quiero:	Iniciar un chat con la clínica.
Para:	Para que puedan atender y responder mis solicitudes.

10.4. Product Backlog.

Historia de usuario	Descripción	Prioridad
HU1	Ingresar a la aplicación mediante inicio de sesión.	1
HU2	Registrarse en la aplicación.	1
HU3	atención al cliente.	3

10.5. Estimación del producto backlog.

Para estimar el tiempo y el esfuerzo de las historias de usuario se emplea el método de los puntos de historia y planning poker para asignar valores a cada historia de acuerdo a la tabla.

	HU	Descripción	Size	Puntos de historia	Tiempo [Días]
	H1	Ingresar a la aplicación como usuario mediante inicio de sesión.	Xtra Small (XS)	1	1/2

Primer sprint	H2	Registrarse en la aplicación.	Xtra Small (XS)	1	1/2
	H3	Atención al cliente	Xtra Large (XL)	8	5
Puntos de historia / tiempo estimado (time boxing)				10	6

De acuerdo al tamaño y a la cantidad de historias de usuario, el tiempo estimado para el desarrollo del sistema es de 6 días la cual hace ver que es un proyecto pequeño lo cual solo lo desarrollaremos en un solo sprint.

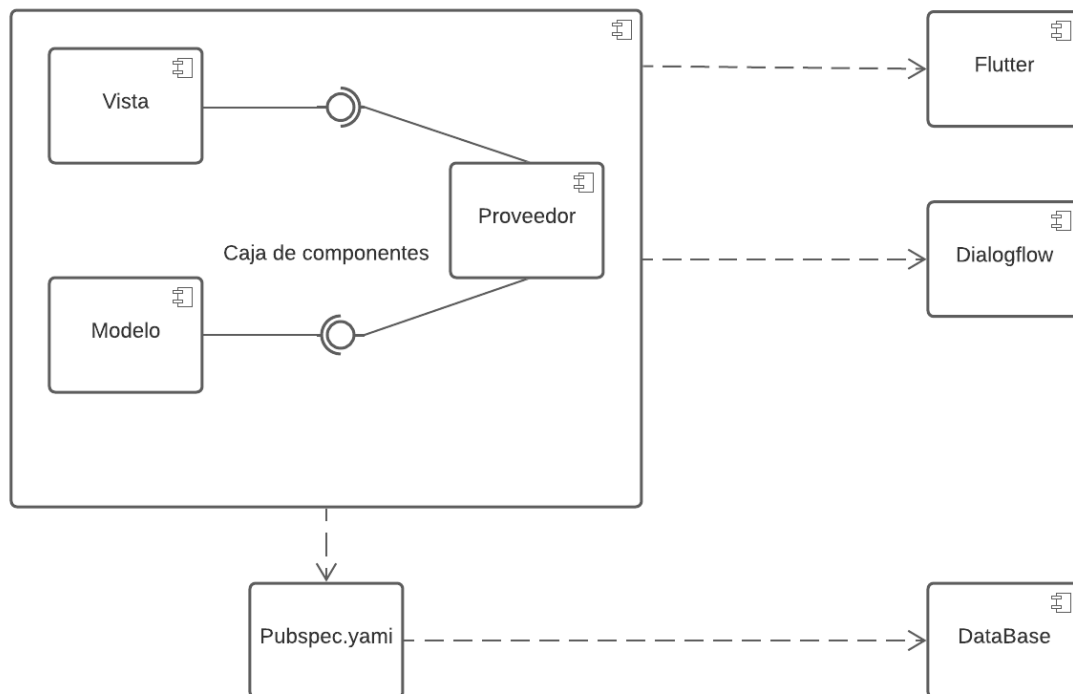
10.6. Definición de hecho.

Se establece la definición de hecho cuando todas las tareas definidas en el sprint backlog de una iteración se hayan completado se han verificado los criterios de aceptación y se ha desplegado el product increment y se hayan desplegado en un servidor.

10.7. Definición de la Arquitectura.

para modelar y analizar la arquitectura de la aplicación se utiliza el diagrama de componentes y el diagrama de despliegue del lenguaje modelado unificado UML.

10.7.1. Diagrama de componentes.



10.8. Primer Sprint.

10.8.1. Sprint planning.

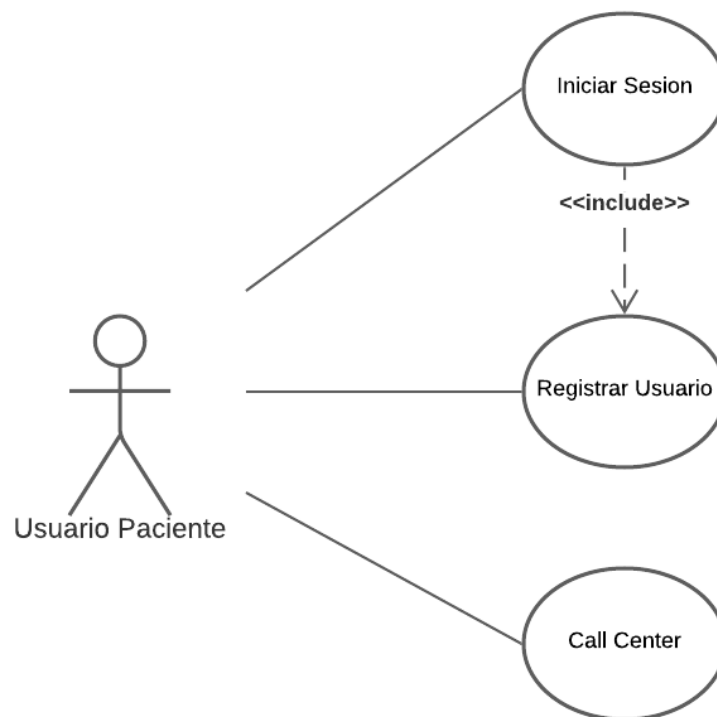
El sprint planning muestra lo que tiene que desarrollarse en esta iteración y los tiempos comprometidos para finalizar a cada historia de usuario.

HU	DESCRIPCION	PUNTOS DE HISTORIA	TIEMPO[DIAS]	INICIO	FINALIZACION
HU1	Ingresar a la aplicación mediante inicio de sesión.	1	½	18/07/2021	18/07/2021
HU2	Registrarse en la aplicación.	1	½	18/07/2021	18/07/2021
HU3	Atención al cliente.	8	5	20/07/2021	28/07/2021
Puntos de historia/tiempo estimado (time boxing)		10	6		

Para completar este sprint se estima un esfuerzo de 10 puntos de historia y tiempo de desarrollo de 6 días.

Para comprender mejor la funcionalidad del sprint se modela un diagrama de casos de uso.

10.8.2. Diagrama de caso de uso.



10.8.3. detalles de caso de uso.

Para los detalles de caso de uso, utilizaremos las historias de usuario detallados.

Caso de uso: Iniciar sesion.

Caso de uso	Iniciar sesion.
--------------------	-----------------

Historia de usuario.	HU1.
Propósito.	El usuario cliente podrá iniciar sesión en la aplicación móvil mediante un email y un password
Resumen.	Ingresa a la aplicación con un email y password.
Actor.	Usuario cliente.
Depende de.	HU2.
Flujo principal	<ul style="list-style-type: none"> • El usuario cliente ingresa su email y password y presiona el botón iniciar sesión. • La aplicación busca los datos introducidos por el usuario cliente en la base de datos. • Una vez verificado los datos, la aplicación muestra la actividad del chat en la cual el usuario podrá interactuar con la IA.
Excepción.	En el caso que el cliente no pueda ingresar a la aplicación, significa que no sus datos no son válidos y que tiene que registrarse.
Elaborado por	Carlos Eduardo Torrez Condarco.
Responsable	Carlos Eduardo Torrez Condarco.

Prototipo: Caso de Uso Iniciar sesión

12:22 AM

Bienvenido a
Med +

Correo

Password

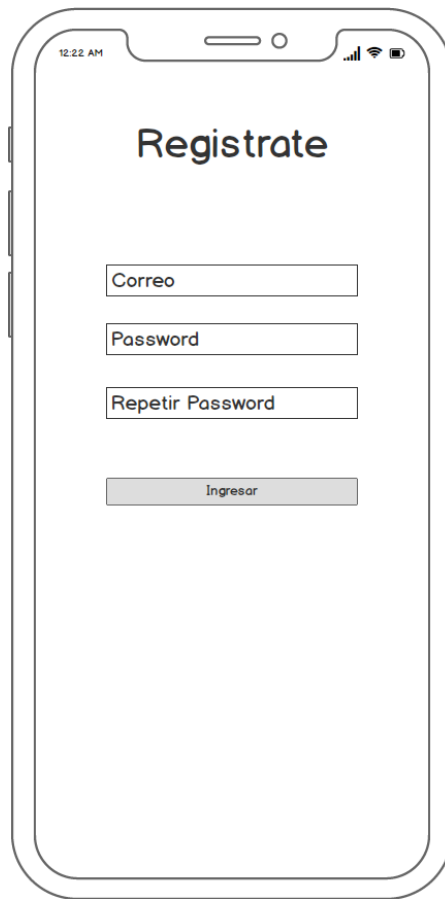
Ingresar

Registrar

Caso de uso: Registrar Usuario.

Caso de uso	Registrar Usuario.
Historia de usuario.	HU2.
Propósito.	El usuario cliente podrá registrarse en la aplicación ingresando un email y password, para así poder iniciar sesión y pueda ingresar a la aplicación.
Resumen.	Registrarse en la aplicación con un email y password.
Actor.	Usuario cliente.
Depende de.	Ninguno.
Flujo principal	<ul style="list-style-type: none"> • El usuario cliente ingresa su email y password y presiona el botón registrar. • El email debe ser válido, de lo contrario no podrá registrarse. • La aplicación hace las validaciones correspondientes antes de guardar los datos. • Una vez validados los datos, se guardan en la base de datos. • La aplicación muestra un mensaje indicando que el registro fue exitoso.
Excepción.	En el caso que el registro no pueda realizarse, significa que sus datos no son correctos. Y no cumplen con las validaciones.
Elaborado por	Carlos Eduardo Torrez Condarco.
Responsable	Carlos Eduardo Torrez Condarco.

Prototipo: Caso de Uso Registrar Usuario

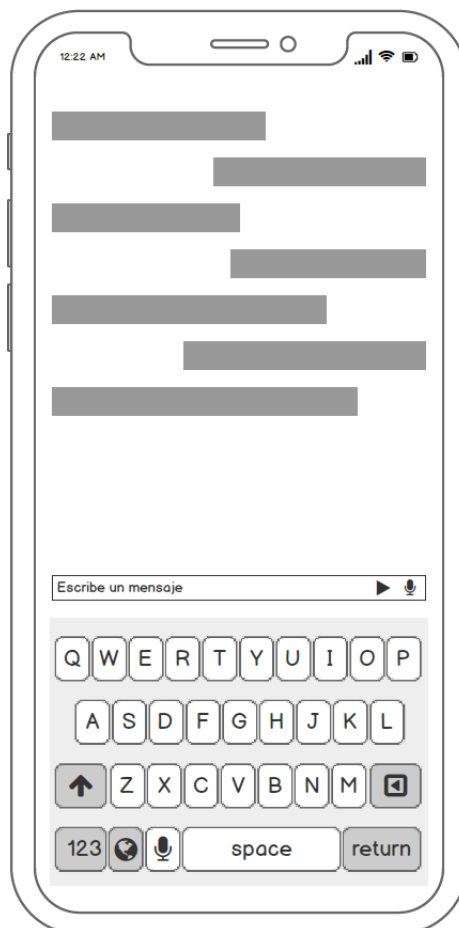


Caso de Uso: Call Center Atencion Paciente.

Caso de uso	Call Center Atencion Paciente.
Historia de usuario.	HU3.
Propósito.	El usuario cliente podrá consultar con la IA cualquier información básica que una secretaria puede proporcionar.
Resumen.	Atención al cliente mediante un chat dentro de la aplicación.
Actor.	Usuario cliente.
Depende de.	Ninguno.
Flujo principal	<ul style="list-style-type: none">• El usuario cliente inicia sesión en la aplicación, y se mostrara la pantalla del chat• Se mostrará un chat en el cual el usuario cliente podrá escribir mensajes sobre cualquier consulta que tenga, también podrá hacer mediante comando de voz• Una vez su mensaje esté correcto, presionara el botón enviar.• El chatbot contestara de manera inmediata la consulta, que le envíe el usuario cliente.
Excepción.	En caso que el chatbot no pueda comprender el mensaje que le envía el usuario cliente, el chatbot, tendrá que enviarle un mensaje diciendo que especifique su pregunta.

Elaborado por	Carlos Eduardo Torrez Condarco.
Responsable	Carlos Eduardo Torrez Condarco.

Prototipo: Caso de Uso atención al cliente.



10.8.4. Sprint Back log

Para determinar cómo se va a desarrollar el sprint, se identifican una o varias tareas por historia de usuario, las cuales son agrupadas en sprint back log y las cuales son asignadas a los miembros del equipo de desarrollo.

HU.	Descripción.	Tareas.
HU1	Ingresa a la aplicación mediante inicio de sesión.	T1. Diseñar e implementar las interfaces de login de usuario. T2. agregar las respectivas conexiones y validaciones.
HU2	Registrarse en la aplicación.	T3. Diseñar e implementar las interfaces de registro de usuario. T4. agregar las respectivas conexiones y validaciones.

HU3	Call Center.	T5. Diseñar las interfaces de atención al cliente. T6. Agregar y configurar la herramienta de GOOGLE DIALOGFLOW en nuestra aplicación.
------------	--------------	---

10.8.5. Sprint execution

Para controlar la ejecución del sprint y el avance, se utilizó Google calendar.

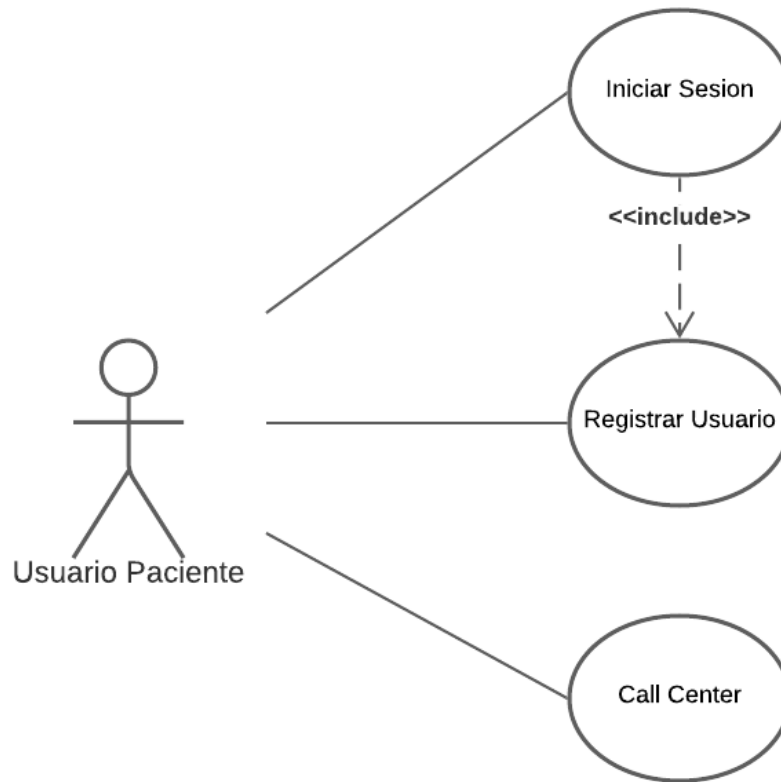
10.8.6. Modelo no relacional de la base de datos que se ha implementado en base a los requerimientos del sprint.



10.8.7. Sprint review.

el product increment representa los entregables realizados al product owner y al cliente, estos entregables reflejan el cumplimiento del primer sprint.

11. Diagrama general de caso de uso.



12. Conclusiones y recomendaciones.

Después de haber concluido las etapas de desarrollo de la documentación, puedo dar las siguientes recomendaciones:

12.1. Conclusiones.

Se puede trabajar con el marco de trabajo ágil SCRUM combinado con UML, sin dañar el concepto de manifiesto ágil. Solamente utilizando los diagramas necesarios que ayuden a comprender mejor la funcionalidad, las interacciones y la lógica de la aplicación.

UML al ser una notación de modelado visual mejora la comprensión del problema y el objetivo de las iteraciones, además de mejorar la comunicación con los miembros de tu equipo de desarrollo.

Se cumplieron a cabalidad las tareas que fueron marcadas al principio del proyecto.

- Con la ayuda del product vision board se logró comprender la visión del producto que fue desarrollado.
- Se descompuso la aplicación en historias de usuario que fueron determinadas por el product owner.
- Se cumplió con lo que fue planteado en el product back log siguiendo los plazos estimados.
- Se han modelados diagramas de caso de uso para tener una visión global de la funcionalidad dentro de cada sprint.
- Se hizo uso de los diagramas de clases para representar la lógica de la base de datos.
- También se usó los diagramas de componentes y despliegue para representar la arquitectura física y lógica de la aplicación.

12.2. Recomendaciones.

Si se va a utilizar el marco de trabajo ágil SCRUM combinado con una notación de modelado visual como UML, se recomienda utilizar solo los diagramas necesarios para comprender el proyecto, para así no dañar el concepto del manifiesto ágil.

13. Bibliografía.

Scrum. (2018, 15 febrero). Wikipedia. [https://es.wikipedia.org/wiki/Scrum_\(desarrollo_de_software\)](https://es.wikipedia.org/wiki/Scrum_(desarrollo_de_software))

Crear agente en Dialogflow: tutorial de primeros pasos en la GCP. (2021, 26 enero). Dialogflow experts. <https://dialogflowexperts.com/crear-agente-en-dialogflow/>

Guía de inicio rápido: Compila un agente / Dialogflow ES. (s. f.). Google Cloud. Recuperado 27 de julio de 2021, de <https://cloud.google.com/dialogflow/es/docs/quick/build-agent?hl=es-419>