



UNIVERSIDAD CÉSAR VALLEJO

**FACULTAD DE INGENIERÍA Y ARQUITECTURA
ESCUELA PROFESIONAL DE INGENIERÍA DE SISTEMAS**

Chatbot para el aprendizaje de TypeScript

TESIS PARA OBTENER EL TÍTULO PROFESIONAL DE:
Ingeniero de Sistemas

AUTOR:

Cardenas Rios, Armando (orcid.org/0000-0002-7256-7909)

ASESOR:

Dr. Alfaro Paredes, Emigdio Antonio (orcid.org/0000-0002-0309-9195)

LÍNEA DE INVESTIGACIÓN:

Sistema de información y comunicaciones

LÍNEA DE RESPONSABILIDAD SOCIAL UNIVERSITARIA:

Desarrollo económico, empleo y emprendimiento

LIMA – PERÚ

2022

Dedicatoria

Esta tesis se la dedico en primer lugar a mi madre Dora Mirarda Rios Julca, quien me ha brindado su apoyo incondicional desde el principio y también a mi hermano Melquiades y a mis hermanas Dina y Karin, quienes han sido el motivo de seguir adelante.

Agradecimiento

Agradezco al Dr. Emigdio Antonio Alfaro Paredes por el apoyo y paciencia constante en la construcción de esta tesis. También a mi enamorada Jessica Soledad Mallco Rios, quien siempre estuvo apoyándome en el logro de este objetivo. Asimismo, agradezco a mis profesores, compañeros y amigos por su apoyo transmitiendo sus conocimientos y a quienes me dieron las fuerzas para sacar adelante esta investigación.

Índice de contenidos

I. INTRODUCCIÓN	1
II. MARCO TEÓRICO	9
III. METODOLOGÍA	17
3.1 Tipo y diseño de investigación	18
3.2 Variables y operacionalización	18
3.3 Población, muestra y muestreo	20
3.4 Técnicas e instrumentos de recolección de datos	20
3.5 Procedimientos	21
3.6 Método de análisis de datos	21
3.7 Aspectos éticos	22
IV. RESULTADOS	23
4.1 Prueba de la hipótesis específica 1	24
4.2 Prueba de la hipótesis específica 2	26
4.3 Prueba de la hipótesis específica 3	29
4.4 Prueba de la hipótesis específica 4	32
4.5 Prueba de la hipótesis específica 5	35
4.6 Prueba de la hipótesis general	37
4.7 Resumen	37
V. DISCUSIÓN	38
VI. CONCLUSIONES	42
VII. RECOMENDACIONES	45
REFERENCIAS	48

Índice de tablas

Tabla 1 Incremento de conocimiento hacia el aprendizaje de TypeScript	24
Tabla 2 Prueba de normalidad del incremento de conocimiento	25
Tabla 3 Prueba T sobre el incremento de conocimiento.....	26
Tabla 4 Prueba de muestras emparejadas.....	26
Tabla 5 Incremento de motivación hacia el aprendizaje TypeScript.....	27
Tabla 6 Prueba de normalidad el incremento de motivación hacia el aprendizaje de TypeScript.....	28
Tabla 7 Rangos de prueba de signos – el incremento de motivación hacia el aprendizaje de TypeScript.....	28
Tabla 8 Estadística de prueba Z – Incremento de motivación hacia el aprendizaje de TypeScript.....	29
Tabla 9 Incremento de la satisfacción con el aprendizaje TypeScript.....	30
Tabla 10 Pruebas de normalidad del incremento de la satisfacción con el aprendizaje.....	30
Tabla 11 Rangos de prueba de signos – Incremento de la satisfacción con el aprendizaje.....	31
Tabla 12 Estadística de prueba Z – Incremento de la satisfacción con el aprendizaje	31
Tabla 13 Medición de la asertividad por cada pregunta del usuario	32
Tabla 14 Puntaje total de la asertividad por cada usuario	33
Tabla 15 Tiempo de interacción por usuario en minutos	35
Tabla 16 Medición de la verificación de sintaxis por cada pregunta del usuario	36
Tabla 17: Matriz de operacionalización de variables	54
Tabla 18: Matriz de consistencia	55
Tabla 19: Instrumento de recolección de datos para el incremento de conocimiento	56
Tabla 20: Instrumento de recolección de datos para el incremento de la motivación hacia el aprendizaje	56
Tabla 21: Instrumento de recolección de datos para el incremento de la satisfacción con el aprendizaje	57
Tabla 22: Instrumento de recolección de datos para el incremento de la asertividad de la respuesta	57
Tabla 23: Instrumento de recolección de datos para la reducción del tiempo de aprendizaje....	58
Tabla 24: Instrumento de recolección de datos para la eficiencia en la verificación de sintaxis ..	58
Tabla 25: Pre-test para la satisfacción con el aprendizaje	59
Tabla 26: Pre-test para la motivación hacia el aprendizaje.....	60
Tabla 27: Post-test para la satisfacción con el aprendizaje	68
Tabla 28: Post-test para la motivación hacia el aprendizaje	68
Tabla 29: Scrum team.....	91
Tabla 30: Historias de usuarios	91
Tabla 31: Pila de producto o tareas iniciales	92
Tabla 32: Lista de Sprints.....	94
Tabla 33: Diccionario de la base de datos.....	98
Tabla 34: Descripción de los campos de la tabla tb_answer	98
Tabla 35: Descripción de los campos de la tabla tb_answer_bot	99
Tabla 36: Descripción de los campos de la tabla tb_chat_type	99
Tabla 37: Descripción de los campos de la tabla tb_counter	100
Tabla 38: Descripción de los campos de la tabla tb_dictionary.....	100
Tabla 39: Descripción de los campos de la tabla tb_question	100
Tabla 40: Descripción de los campos de la tabla tb_question_user	101
Tabla 41: Descripción de los campos de la tabla tb_typescript_type	101
Tabla 42: Descripción de los campos de la tabla tb_user	101
Tabla 43: Descripción de los campos de la tabla tb_user_key	102
Tabla 44: Descripción de los campos de la tabla tb_user_role	102

Índice de figuras

Figura 1: Arquitectura del chatbot	84
Figura 2: Flujograma	85
Figura 3: Algoritmo principal del chatbot	86
Figura 4: Algoritmo Tokenizer.....	87
Figura 5: Algoritmo TF-ID para el cálculo de las similitudes.....	88
Figura 6: Pseudocódigo del algoritmo Tokenizer.....	89
Figura 7: Pseudocódigo del algoritmo TF-IDF	89
Figura 8: Pseudocódigo Fydebot	90
Figura 9: Plan de proyecto	96
Figura 10: Modelo base de datos	97
Figura 11: Pantalla de inicio	103
Figura 12: Pantalla de registro de usuario.....	104
Figura 13: Pantalla de login.....	105
Figura 14: Pantalla de tutorial del chatbot	106
Figura 15: Módulo de ruta de aprendizaje	107
Figura 16: Modulo preguntas destacadas	108
Figura 17: Módulo preguntas y respuestas	109
Figura 18: Diseño del chat.....	110
Figura 19: Mensaje de inicio.....	111
Figura 20: Respuesta del chatbot en texto	111
Figura 21: Respuesta del chatbot en código	112
Figura 22: Respuesta no destacada	112
Figura 23: Respuesta destacada	113
Figura 23: Apartado de la verificación de sintaxis.....	113
Figura 24: Respuesta correcta de la sintaxis.....	114
Figura 25: Respuesta incorrecta de la sintaxis	114

Índice de anexos

Anexo 1: Matriz de operacionalización de variables	54
Anexo 2: Matriz de consistencia	55
Anexo 3: Instrumento de recolección de datos.....	56
Anexo 4: Cuestionario de satisfacción con el aprendizaje pre-test.....	59
Anexo 5: Cuestionario de motivación hacia el aprendizaje pre-test	60
Anexo 6: Test de conocimiento pre-test.....	61
Anexo 7: Cuestionario de satisfacción con el aprendizaje post-test	68
Anexo 8: Cuestionario de motivación hacia el aprendizaje pot-test.....	68
Anexo 9: Test de conocimiento post-test	69
Anexo 10: Arquitectura del Chatbot	84
Anexo 11: Flujograma y pseudocódigo del algoritmo del chatbot.....	85
Anexo 12: Algoritmo principal de Fydebot	86
Anexo 13: Algoritmo Tokenizer.....	87
Anexo 14: Algoritmo TF-IDF	88
Anexo 15: Pseudocódigo del chatbot.....	89
Anexo 16: Ejecución de la metodología Scrum	91
Anexo 17: Modelo relacional de la base de datos	97
Anexo 18: Diccionario de la base datos general.....	98
Anexo 19: Capturas de pantalla del Chatbot	103

Índice de abreviaturas

Sigla	Significado	Pág.
SQL	Structure Query Language [lenguaje estructurado a base de sentencias] (Balderas et al., 2020)	8
IA	Inteligencia Artificial (Whiting, 2020)	13
IDC	International Data Corporation o Corporación Internacional de Datos (Whiting, 2020)	13
TF-IDF	Term Frequency - Inverse Document Frequency (Frecuencia de Tiempo - Frecuencia de Documento Inversa) (Gelemet et al., 2022).	13

Resumen

Esta investigación tiene como problema general, ¿Cuál fue el efecto del uso del chatbot para el aprendizaje de TypeScript? El objetivo fue determinar el efecto del uso del chatbot para el aprendizaje de TypeScript, como una buena opción en el aprendizaje de este lenguaje de programación, enfocado en los programadores profesionales, estudiantes y/o egresados, con la finalidad de medir el incremento de conocimiento, el nivel de motivación, el nivel de satisfacción, la asertividad de las respuestas y la reducción del tiempo de aprendizaje. El diseño de esta investigación fue experimental, el tipo de diseño fue pre-experimental, el tipo de investigación fue aplicada y el enfoque cuantitativo.

El chatbot para el aprendizaje de TypeScript fue desarrollado con Java Spring Boot y servicios REST como back-end y Angular como front-end; además, se usó contenedores Docker para englobar todas las dependencias del software. La muestra tuvo 20 estudiantes y/o programadores con conocimientos básicos en programación. Se obtuvo los siguientes resultados: 70.29% en el incremento de conocimiento, 14.81% en el incremento de la motivación, 14.81% en el incremento de la satisfacción, 88.15% en la asertividad de la respuesta, el tiempo promedio de uso fue 104.35 minutos (lo que fue menor al tiempo promedio con otras aplicaciones encontradas) y la eficiencia de la verificación de sintaxis fue 100%.

El uso del chatbot logró incrementar el aprendizaje del lenguaje de programación TypeScript con la implantación de los módulos: preguntas destacadas, ruta de aprendizaje y la verificación de sintaxis. Además, para el aprendizaje de cualquier lenguaje de programación es mejor utilizar las tecnologías de gamificación y microlearning, web scraping y chatbot (en la web), añadiendo la verificación de sintaxis. Se recomendó implementar evaluaciones por cada módulo o tema de aprendizaje y chatbots de aprendizaje en un sistema web con una arquitectura de interacción continua y cómoda que evite distracciones.

Palabras clave: Chatbot, TypeScript, Inteligencia Artificial, Automatización, Implementación

Abstract

This research has as a general problem, what was the effect of using the chatbot for learning TypeScript? The objective was to determine the effect of using the chatbot for learning TypeScript, as a good option in learning this programming language, focused on professional programmers, students and/or graduates, in order to measure the increase in knowledge, the level of motivation, the level of satisfaction, the assertiveness of the responses and the reduction in learning time. The design of this research was experimental, the type of design was pre-experimental, the type of research was applied and the approach was quantitative.

The TypeScript learning chatbot was developed with Java Spring Boot and REST services as back-end and Angular as front-end; In addition, Docker containers were used to encompass all the dependencies of the software. The sample had 20 students and/or programmers with basic programming knowledge. The following results were obtained: 70.29% in the increase in knowledge, 14.81% in the increase in motivation, 14.81% in the increase in satisfaction, 88.15% in the assertiveness of the response, the average time of use was 104.35 minutes (which was less than the average time with other applications found) and the efficiency of the syntax check was 100%.

The use of the chatbot managed to increase the learning of the TypeScript programming language with the implementation of the modules: Featured questions, learning path and syntax verification. In addition, for learning the programming language, it is better to use gamification and microlearning technologies, web scraping and chatbot (on the web), adding syntax verification. It was recommended to implement evaluations for each module or learning topic and learning chatbots in a web system with a continuous and comfortable interaction architecture that avoids distractions.

Keywords: Chatbot, TypeScript, Artificial Antelligence, Automation, Implementation

I. INTRODUCCIÓN

En esta investigación se desarrolló un chatbot para el aprendizaje de TypeScript. A medida que los sitios web evolucionan, también lo hace el desarrollo web y es por esta razón que JavaScript se ha convertido en uno de los principales lenguajes de programación para el desarrollo web en la parte de front-end (Jiang et al., 2021). Sin embargo, este lenguaje tiene carencias de tipados en los tipos de datos, cayendo en errores continuos al momento de la ejecución y compilación del código (Kristensen y Møller, 2017). A partir de esto, TypeScript se ha convertido una buena opción para manejar estos errores (Golec y Plechawska-Wójcik, 2022). Asimismo, muchos estudiantes y/o programadores tienen la necesidad de aprender el lenguaje de programación TypeScript, permitiendo la prevención de errores relacionados con los tipos de datos en la programación (Golec y Plechawska-Wójcik, 2022).

Además, esta investigación se justificó de manera social, tecnológico y teórico. Por lo tanto, el problema planteado fue ¿Cuál es el efecto del uso del chatbot en el aprendizaje de TypeScript?; el objetivo fue determinar el efecto del uso del chatbot en el aprendizaje de TypeScript con el fin de evaluar los indicadores asociados a: conocimiento, motivación, satisfacción, asertividad de la respuesta, tiempo de aprendizaje, precisión en la verificación de sintaxis y reducción del tiempo de búsqueda de la respuesta. Vinculado a esto, se planteó como hipótesis que el uso del chatbot incrementa el aprendizaje de TypeScript.

Li (2019) mencionó que los sitios web al que accedemos frecuentemente son mucho más interactivos, dinámicos y atractivos, en donde también se puede interactuar con mapas, juegos, comunicaciones al instante, pagos en línea, etc. y todo ello gracias al rápido crecimiento del desarrollo web front-end. Por consecuencia, JavaScript es uno los lenguajes de programación más utilizados en el desarrollo de front-end por ser ligero, flexible y potente (Jiang et al., 2021). Por otra parte, Kristensen y Møller (2017) mencionaron que el lenguaje de programación JavaScript tiene un análisis estático muy poco sólido y, en consecuencia, a menudo pasan por alto errores e informan de advertencias engañosas que puede ser perjudicial al momento de la compilación del código (Kristensen y Møller, 2017).

Al respecto, Grzegorz (2021) mencionó que en el año 2020 la casa de software, la cual es un importante soporte de conocimientos de la satisfacción, la experiencia y seguimiento laboral diario de los programadores. Pues esta base de conocimientos hizo una encuesta para evaluar el porcentaje de uso y/o conocimiento del lenguaje de programación TypeScript, en donde el 77% de los encuestados mencionaron que usan y conocen esta tecnología, mientras que el 23% no conocen o usan este lenguaje; además el 94% de los desarrolladores encuestados que conocen y usan indicaron que les gusta TypeScript (Grzegorz, 2021).

En efecto el nuevo lenguaje de programación TypeScript de Microsoft se ha convertido en uno de los más populares lenguajes para el desarrollo web a lo largo de los años, clasificándose como el décimo idioma más utilizado en la encuesta de Stack Overflow del 2019 (Verbeek, 2020). Asimismo Yin et al. (2021) concluyeron que en los últimos años los chatbots han obtenido popularidad en la educación gracias a las estrategias de microaprendizaje de conocimientos. Además, los chatbots para aprendizaje pueden ayudar al incremento de conocimiento sin la necesidad de un instructor presencial (Yin et al., 2021).

Considerando los estudios previos no se encontraron estudios acerca del aprendizaje de TypeScript mediante los chatbots. También, se encontraron estudios de aprendizaje relacionados al lenguaje de desarrollo de software, pero no están vinculados con TypeScript ni con la verificación de sintaxis. En tal sentido, esta investigación tuvo como objetivo principal determinar el efecto del uso del chatbot para el aprendizaje de TypeScript.

Inbenta (2021) mencionó que la ventaja de los Bots o sistemas automáticos es mostrar siempre una información veraz, además, los Bots trabajan constantemente con un cimiento de información, ya que no tienen diversas fuentes de información, este cimiento va a extender la información indicada en todos los canales seleccionados, la cual puede ser actualizada constantemente. Por lo tanto, podemos decir que la información siempre será veraz y actualizada sea cual sea el escenario (Inbenta, 2021).

La ausencia de estudios sobre el aprendizaje de TypeScript no generaría los resultados deseados en el desarrollo de front-end y en su lugar provocaría malas prácticas al momento de desarrollo que acabaría finalmente en un producto de software no deseado (Golec y Plechawska-Wójcik, 2022). En este sentido Merkel y Bogner (2021) mencionaron que TypeScript muestra una mejor calidad, comprensión lectora de código y mejor seguridad en los tipos de datos, lo cual conduce a una mejor calidad de software que la de JavaScript

Seguidamente, se presentan las justificaciones tanto teórica, tecnológica y social. A continuación, se presenta la justificación teórica del aprendizaje del lenguaje de programación TypeScript y la importancia que tiene éste en el desarrollo web front-end. De esta manera Rodríguez Garcia (2021) mencionó que la implementación de un chatbot para el aprendizaje ha provocado el uso frecuente de otros lenguajes de programación, por ende, aporta mejoras para el dominio y conocimiento de cualquier lenguaje de programación.

Esta investigación asistió con la justificación tecnológica, ya que con el uso del Chatbot se transmitirá información acerca del uso del lenguaje de programación TypeScript. De esta manera, Ariste y Ramírez (2020) citaron a Huerta (2019), quien detalló que la importancia y la influencia de utilizar los chatbots en estos últimos años aumentó considerablemente, ya que las organizaciones los han empezado a utilizar como fuente principal de información hacia sus clientes para que sus productos y servicios puedan ser conocidos y acogidos mediante el contacto directo con sus clientes. Además, los chatbots pueden ser multiplataforma; por lo tanto, los usuarios solo necesitarán un dispositivo con internet (Ariste y Ramírez, 2020).

La justificación se demuestra de forma social hacia los usuarios desarrolladores de aplicaciones o sistemas web front-end que quieran aprender el lenguaje de programación TypeScript. De esta manera, De Luis (2019) mencionó que con los nuevos canales de comunicación las empresas cada vez más apuestan por el uso de la tecnología conversacional. A medida que los usuarios opten por el uso de los asistentes conversacionales más conocidos como chatbots, las empresas u organizaciones también tendrán que optar por los canales que demanden sus clientes (De Luis, 2019).

En el contexto de la realidad problemática se desarrolló y evaluó el problema general de la investigación: ¿Cuál fue el efecto del uso del Chatbot para el aprendizaje de TypeScript? Correspondientemente al problema general, surgieron los siguientes problemas específicos:

- **PE1:** ¿Cuál fue el efecto del uso del chatbot para el aprendizaje de TypeScript en el conocimiento?
- **PE2:** ¿Cuál fue el efecto del uso del chatbot para el aprendizaje de TypeScript en la motivación hacia el aprendizaje?
- **PE3:** ¿Cuál fue el efecto del uso del chatbot para el aprendizaje de TypeScript en la satisfacción con el aprendizaje?
- **PE4:** ¿Cuál fue el efecto del uso del chatbot para el aprendizaje de TypeScript en la asertividad de las respuestas?
- **PE5:** ¿Cuál fue el efecto del uso del chatbot para el aprendizaje de TypeScript en el tiempo de aprendizaje?
- **PE6:** ¿Cuál fue el efecto del uso del chatbot para el aprendizaje de TypeScript en la eficiencia de verificación de sintaxis?

El objetivo general fue determinar el efecto del uso del chatbot para el aprendizaje de TypeScript. Los objetivos específicos fueron los siguientes:

- **OE1:** Determinar el efecto del uso del chatbot para el aprendizaje de TypeScript en el conocimiento.
- **OE2:** Determinar el efecto del uso del chatbot para el aprendizaje de TypeScript en la motivación hacia el aprendizaje.
- **OE3:** Determinar el efecto del uso del chatbot para el aprendizaje de TypeScript en la satisfacción con el aprendizaje.
- **OE4:** Determinar el efecto del uso del chatbot para el aprendizaje de TypeScript en la asertividad de las respuestas.
- **OE5:** Determinar el efecto del uso del chatbot para el aprendizaje de TypeScript en el tiempo de aprendizaje.
- **OE6:** Determinar el efecto del uso del chatbot para el aprendizaje de TypeScript en la eficiencia de verificación de sintaxis.

La hipótesis general fue: “El uso del chatbot incrementó el aprendizaje de TypeScript” (Balderas et al., 2020; BendeZú y Canales, 2020). Las hipótesis específicas fueron las siguientes:

- **HE1:** El uso del chatbot incrementó el conocimiento del aprendizaje de TypeScript.

Ariste y Ramírez (2020) expresaron que después que los estudiantes hayan interactuado con el chatbot obtuvieron una calificación de aprobación de 85% sobre el aprendizaje de la fotosíntesis generando conocimientos a los estudiantes, de esta manera el uso del chatbot para el aprendizaje de la fotosíntesis lo determina como un factor muy importante para los estudiantes (p. 51). Asimismo, BendeZú y Canales (2020) sostuvieron que gracias a la gamificación y microlearning como tablas de posiciones, juegos de preguntas y recompensas logró un crecimiento de conocimiento del 96.84% del lenguaje de programación JavaScript (p. 45). Además, Meza y Yurivilca (2020) consiguieron gracias al desarrollo un chatbot en Telegram un incremento de conocimiento de 64.93% en el aprendizaje de seguridad electrónica.

- **HE2:** El uso del chatbot incrementó la motivación hacia el aprendizaje de TypeScript.

Ariste y Ramírez (2020) describieron que el uso del chatbot genera un impacto favorable a los estudiantes en cuanto a la usabilidad del bot, ya que el 82% de los estudiantes se sentían entre los niveles de motivado y muy motivado, por esta razón el chatbot tuvo un rol muy importante en cuanto a la motivación de los estudiantes hacia el aprendizaje de la fotosíntesis. También, Valdivia (2021) desarrolló un chatbot para el aprendizaje del cuidado de bebés, consiguiendo un incremento de la motivación de 94.93%. De igual modo BendeZú y Canales (2020) mencionaron que la aplicación con microlearning y gamificación motivan mediante estrategias de niveles, insignias entre otros, obteniendo un incremento del 14.59% con un total de 66.7% de

estudiantes motivados (p. 45).

- **HE3:** El uso del chatbot incrementó la satisfacción con el aprendizaje de TypeScript.

Ariste y Ramírez (2020) afirmaron que el uso del chatbot genera un impacto beneficioso para los estudiantes en cuanto a la usabilidad del bot, ya que consiguieron una satisfacción de 80% de los estudiantes del nivel secundario. Por otro lado, Bendezú y Canales (2020) implementaron conceptos básicos de microlearning y gamificación como videos cortos, evaluaciones y recompensas; de esta manera los estudiantes lograron un nivel de satisfacción del 13.47% luego del uso de la aplicación para el aprendizaje de JavaScript.

Asimismo, Huaccachi y Mejía (2021) alcanzaron un incremento de satisfacción de 32.26% con la aplicación móvil de aprendizaje de acciones ante la violencia a menores de edad, Larico y Reyes (2020) obtuvieron un incremento de satisfacción de 69.68% con el chatbot de aprendizaje de la limpieza y desinfección en el hogar y (Valdivia, 2021) alcanzó un incremento de la satisfacción de 96.67% con el chatbot de aprendizaje del cuidado de bebés.

- **HE4:** El uso del chatbot incrementó la asertividad de las respuestas del aprendizaje de TypeScript.

Ogosi (2021) enfatizó que los chatbot pueden proporcionar respuestas más exactas y de calidad actualizando las preguntas y respuestas, de esta manera entrenamos al bot para la eficiencia en el proceso de aprendizaje con el humano. En ese mismo contexto Valdivia (2021) llegó obtener una asertividad de 15.58% en el chatbot de aprendizaje enfocado al cuidado y bienestar de un recién nacido. Asimismo, Iquiño y Castillo (2021) lograron obtener un asertividad de 89.52% con el chatbot para el aprendizaje sobre los procesos de vacunación.

- **HE5:** El uso del chatbot redujo el tiempo de aprendizaje de TypeScript.

Daud et al., (2020) expresaron que el chatbot reduce el tiempo de

aprendizaje, obteniendo ayuda de la respuesta exacta para su problema en lugar de un ejemplo de código similar de la web, de esta manera el chatbot e-Java puede ayudar a los estudiantes a comprender el tema de programación que resulta difícil de entender. Asimismo, Balderas et al. (2020) mencionaron que el aprendizaje de lenguaje SQL presenta dificultades para los estudiantes de ingeniería informática, sin embargo, con el uso del chatbot los estudiantes aprendieron de manera eficiente completando los ejercicios propuestos. De esta forma, Huaccachi y Mejía (2021) obtuvo un tiempo de aprendizaje de promedio de 180.8167 minutos con la aplicación para el aprendizaje sobre las medidas que se deben tomar ante la violencia a mejores de edad.

II. MARCO TEÓRICO

En este capítulo se da a saber una síntesis de estudios en relación al trabajo de investigación, donde se investigaron y hallaron precedentes de nivel internacional y nacional, también se hallaron y describieron conceptos relacionados al tema y otras teorías. La información relacionada a la investigación se obtuvo de distintas bases de datos, repositorios, revistas indexadas entre otras.

En esta primera sección se encuentran los antecedentes de los trabajos previos realizados, donde se evidencia la inexistencia de un chatbot con la capacidad de aprendizaje de TypeScript. Con respecto a las teorías relacionadas, se precisó conceptos de TypeScript, agentes conversaciones, chatbot y también las tecnologías que se utilizaron para el desarrollo del bot.

Gonzales (2018) implementó una aplicación con chatbot para la Universidad tecnológica del Perú, en la cual los docentes de la modalidad semipresencial de pregrado han presentado problemas para realizar una actividad virtual en la plataforma Canvas, esto genera que los estudiantes demoren en presentar las actividades previamente programadas. El objetivo de la investigación fue aclarar los efectos en la motivación y logro de aprendizaje, el tipo de estudio fue experimental con el muestreo de 25 docentes de pregrado, los instrumentos para la recolección de datos fue la encuesta de pregunta con escala de Likert y una ficha de observación para entender el efecto y mejora del logro de aprendizaje (Gonzales, 2018).

Bonales et al. (2020) estudiaron el grado de alcance de las plataformas conversacionales que ofrecen a diversas empresas para la atención médica a distancia en España, para la revisión de las características se tomó como muestra a 5 plataformas conversacionales con mayor demanda; además, realizaron encuestas a usuarios que utilizan el chatbot y entrevistas a expertos que conocen la importancia y el impacto de estas plataformas. El resultado de esta investigación confirma que el uso de las plataformas conversaciones con fines solidarios para la COVID-19 ha alcanzado el 30% con el tope en los meses de abril y mayo del año 2020 (Bonales et al., 2020).

Ariste et al. (2020) desarrollaron un chatbot interactivo y dinámico con la tecnología de inteligencia artificial, con el fin de incrementar y motivar los conocimientos del proceso de la fotosíntesis para los estudiantes de nivel secundario. Además, con el crecimiento de tecnologías multiplataformas y mediante la mensajería instantánea como Telegram, los estudiantes podrán tener la oportunidad fortalecer y adaptarse al uso de los agentes expertos como el chatbot (Ariste et al., 2020). Además, el diseño de esta investigación fue pre experimental con instrumentos y técnicas de recolección de datos; esta investigación aportó conocimientos y motivaciones a los estudiantes de 13 a 15 años para conocer y comprender el proceso de la fotosíntesis (Ariste et al., 2020).

Balderas et al. (2020) aplicaron el desarrollo de un chatbot para el aprendizaje de SQL enfocado en los estudiantes universitarios, permitiendo que puedan hacer cualquier consulta o ejercicios sobre el lenguaje SQL. Además, la aplicación chatbot hace una retroalimentación a la base de datos con fin de tener actualizada la información para los estudiantes (Balderas et al., 2020). Para obtener los resultados sobre el uso del chatbot, este desarrollo fue probado por diversos estudiantes universitarios llegando a la conclusión que el chatbot para el aprendizaje de SQL no solo es útil para los estudiantes universitarios, sino también para profesionales con experiencia en mundo de desarrollo de software o gestores de SQL (Balderas et al., 2020).

Alaaeldin et al. (2021) elaboraron un chatbot para ayudar a los gerentes a usar y comprender el análisis de Big Data basado en los datos de su entorno organizacional; para el diseño de esta aplicación utilizaron una base de datos de conocimientos para la alimentación de información del chatbot. Además, Alaaeldin et al. (2021) indicaron que la base de datos tiene una lista de preguntas y respuestas que será consumida por la plataforma conversacional; las preguntas fueron realizadas por los tomadores de decisiones y las respuestas se derivan de los modelos analíticos, también la aplicación de chatbot ayudó a los tomadores de decisiones en el análisis de Big Data y para sus objetivos comerciales.

Bendezú y Canales (2020) desarrollaron una aplicación móvil para el aprendizaje del lenguaje JavaScript, esta aplicación llegó a incrementar el conocimiento de los estudiantes de la carrera de ingeniería de sistemas, la investigación fue de enfoque cuantitativo, tipo aplicada y de diseño pre-experimental; por consiguiente, esta aplicación obtuvo un crecimiento del conocimiento de 96.88% y un incremento en la motivación de 13.14%. Por otro lado, se recomendó utilizar esta aplicación a los estudiantes para incrementar su conocimiento del lenguaje de programación JavaScript (Bendezú y Canales, 2020).

Larico y Reyes (2020) implementaron un chatbot con el objetivo de aprender sobre la limpieza y desinfección en el hogar y de esta manera protegerse del COVID-19; en esta investigación se utilizó el enfoque cuantitativo, de tipo aplicada y el diseño pre-experimental. Además, se utilizó los cuestionarios para medir el aprendizaje del conocimiento mediante el pre-test y post-test y el lenguaje de programación Python para el desarrollo del chatbot (Larico y Reyes, 2020). En consecuencia, se obtuvo resultados satisfactorios en la limpieza y desinfección del hogar con un 93.83% del aprendizaje de los usuarios; además, recomiendan que deben mantener desinfectado y limpio el hogar para evitar enfermedades como el COVID-19 (Larico y Reyes, 2020).

Rodríguez y Pino (2019) construyeron un chatbot para agilizar las operaciones del servicio de la mesa de ayuda de la empresa HITTS Ecuador, el cual tiene la finalidad de responder y ayudar a los usuarios sobre inquietudes tecnológicos en un horario de alta disponibilidad. Esta aplicación de chatbot se ha desarrollado la parte de Front End con el Framework Angular 8 y configurado en Dialog Frow, por la parte de Back End el Framework Spring Boot, la metodología para el desarrollo fue en cascada y con el diseño tradicional (Rodríguez y Pino, 2019). Finalmente, con el uso de la aplicación de chatbot el índice de satisfacción de los usuarios se incrementaron al utilizar esta herramienta, permitiendo la comunicación directa entre los operadores del sistema (Rodríguez y Pino, 2019).

Kasinathan et al. (2021) sostuvieron como objetivo desarrollar un chatbot multilingüe que permita a las empresas y organizaciones personalizar y

desplegar su propio servicio de chatbot multilingüe con dos características ampliadas, que son: el chat en vivo y el sistema de tickets. El chatbot fue capaz de entender y responder en inglés, malayo y chino. Para lograr esto, el desarrollo utiliza TypeScript para la aplicación web *front-end* mientras que Go como el desarrollo *back-end*, el prototipo desarrollado se evaluó a través de un cuestionario de encuesta y los resultados sugieren que el sistema propuesto sería capaz de ayudar a las pequeñas y medianas empresas y organizaciones a desplegar su propio sistema de chatbot como una alternativa a los servicios de atención a los clientes (Kasinathan et al., 2021).

Iáñez (2018) desarrolló una interfaz conversacional para localizar servicios del municipio de Madrid con la ayuda de Dialogflow, lo cual es una potente intérprete de lenguaje natal; este chatbot ayueó en la integración de la ciudad inteligente, también se integró con las redes sociales como Facebook, Twitter, Instagram, entre otros. Por consecuencia, los usuarios que utilicen la interfaz conversacional tendrán la información sin importar la plataforma que la contenga, para ello el enfoque fue la población de la ciudad de Madrid y de tipo cuantitativo; en conclusión, el proyecto ayudó al usuario a tener cualquier tipo de información relacionado al municipio de la ciudad de Madrid (Iáñez, 2018).

Auz (2021) implementó un prototipo de chatbot que permita la ayuda virtual, el aprendizaje y enseñanza de la cátedra de ingeniería de software de la escuela de ciencias de la computación de la Universidad Politécnica Salesiana. El tipo de estudio fue cuantitativo con una población de 44 estudiantes, en donde se realizaron 1100 preguntas (Auz, 2021). La investigación sobre los agentes conversacionales concluyó que los chatbots se utilizan en cualquier ámbito, estas brindan la ayuda en cualquier instante sin la necesidad de un humano para las respuestas esperadas de las preguntas frecuentes; para la implementación de un aplicado web basado en Python, se recomienda utilizar el Framework Flash ya que es más sencillo y ligero de implementar (Auz, 2021).

Whiting (2020) concluyó que a medida que la inteligencia artificial y machine learning encuentra cada vez más su camino; desde los sistemas de diagnósticos medios de vanguardia hasta la electrónica de consumo, sistemas avanzados de computación cuántica y los asistentes inteligentes, la inteligencia

artificial y el aprendizaje automático han sido tema de moda en el año 2020, según el investigador de mercado IDC fue probable que los ingresos recaudados en los servicios de IA, el hardware y software alcance los 156.500 millones de dólares en todo el mundo del año 2020, un 12.3% más que el año anterior.

A continuación, se menciona las teorías concernientes a la investigación, en las que se detalla metodologías, instrumentos, beneficios, conceptos, efectos y teorías relacionadas a las definiciones: TypeScript (Richards et al., 2015; Rodriguez, 2021), chatbot (De la Rosa, 2020; Balderas et al., 2020), algoritmo (Gelemet et al., 2022) y Scrum (Schwaber y Sutherland, 2021).

Richards et al. (2015) explicaron que TypeScript es una extensión de JavaScript de Microsoft que introduce clases, tipado estructural, anotaciones de tipo en propiedades, argumentos y tipos de retornos. Sintácticamente la extensión que llamamos Strong Script, es mínima, consiste en un único constructor de tipos para los tipos concretos, semánticamente los cambios son más sutiles (Richards et al., 2015). Un sistema de tipos permite a los desarrolladores elegir entre escribir código no tipado (es decir, todas las variables son de cualquier tipo en JavaScript), código opcionalmente tipado que no afecta a la semántica de los programas dinámicos (es decir, no hay nuevos errores dinámicos) y código concretamente tipado que proporciona las garantías tradicionales de corrección pero afecta a la semántica del código dinámico (es decir, los tipos son retenidos por el compilador y utilizados para optimizar el programa); pero pueden aparecer nuevos errores dinámicos (Richards et al., 2015).

Rodriguez (2021) mencionó que TypeScript es un lenguaje de programación que se distribuye como licencia libre y fue creado por Microsoft, donde su función principal es darle la posibilidad a JavaScript de manejar tipados en la declaración del código. Además, TypeScript es compatible con JavaScript, esto significa que se puede ejecutar el código de JavaScript en TypeScript, también es posible indicar el tipo de las variables que se declara, es decisión de cada programador; por lo tanto, no podemos decir que es estáticamente tipado porque el tipado es opcional o a criterio del programador; no obstante, se

aconseja utilizar estos tipados para evitar algunos errores en la compilación del código (Rodríguez, 2021).

La tecnología de información es fundamental para el aprendizaje de los estudiantes y un aliado para el profesorado (Balderas et al. 2020, p.92). La educación a distancia o situaciones de confinamiento como la reciente llegada de la pandemia causada por el COVID-19, el aprendizaje puede ser más dependiente de la tecnología; por tal motivo, los agentes conversacionales, más conocidos como Chatbot, se han incorporado dentro de la plataforma de aprendizaje, estos Chatbot son capaces de responder preguntas e incluso puede transmitir emociones a través de las respuestas o diálogos (Balderas et al. 2020, p.92).

De la Rosa (2020) manifestó que los Chatbots toman seria importancia en las organizaciones. La consultora Forrester indica que el 87% del CEO está tratando de difundir su planilla de IA; además, la consultora Gartner estima que en el año 2022 el 70% de los trabajadores usarán a diario las plataformas conversacionales; si los resultados de estas consultoras se cumplen, las plataformas conversacionales y el talento humano sumarán esfuerzos de manera interactiva por lo tanto en solo 2 años las organizaciones estarán ante un contexto relacional completamente nuevo (De la Rosa, 2020).

Gelemet et al. (2022) explicaron que TF-IDF es un algoritmo utilizado para hacer énfasis en la palabra de un documento. El cálculo de este método TF-IDF se calcula por separado y multiplicándolos entre sí (Gelemet et al., 2022). TF se calcula por la relación entre el número de repeticiones de la palabra en el documento relevante y el número total en el documento (Gelemet et al., 2022). Mientras que el valor IDF se calcula como el valor logarítmico de la relación entre el número total de documentos en el conjunto de datos y el número total de documentos en los que aparece la palabra seleccionada. Como resultado, el valor de peso de la palabra seleccionada se calcula multiplicando los valores TF e IDF y valor que use en la ecuación representa una palabra en la oración y representa el documento (Gelemet et al., 2022).

Schwaber y Sutherland (2021) sostuvieron que Scrum es un marco de trabajo en mundo del desarrollo ágil conformado por un pequeño grupo llamado Scrum Team. Este equipo está formado por los desarrolladores y un Scrum Master, no existe jerarquías ni subgrupos (Schwaber y Sutherland, 2021). El equipo de Scrum es responsable de todas las actividades relacionadas con el producto, incluida la colaboración de las partes interesadas, las pruebas, el mantenimiento, las operaciones, las pruebas, la investigación y el desarrollo, y cualquier otra actividad; pueden trabajar a través de Sprints a un ritmo sostenible mejora el enfoque y la consistencia del equipo Scrum (Schwaber y Sutherland, 2021). Todo el equipo Scrum es responsable de crear una parte de incremento útil y valiosa después de cada Sprint. Scrum define tres roles específicos de responsabilidad dentro de un equipo Scrum: desarrolladores, propietario del producto y Scrum Master (Schwaber y Sutherland, 2021).

Por otra parte, Meyer et al. (2019) mencionaron que los desarrolladores pasan el mayor de su tiempo sin programar, debido a que su tiempo lo emplea en reuniones, planificaciones, pruebas de código y búsqueda de información técnica en la web. Además, los desarrolladores aumentan su motivación y el estado de ánimo al momento de aprender algo nuevo, también aumentan su comprensión en el proyecto y ayuda a los compañeros de trabajo a aprender algo nuevo o desbloquear a alguien de un problema con el que estaban atascados (Meyer et al., 2019). Asimismo, Wang et al. (2018) indicaron que los desarrolladores pasan el mayor de su tiempo en los sitios de preguntas y respuestas en la web, ya que constantemente enfrentan desafíos de desarrollo, lo cual requieren la ayuda de un experto del tema.

III. METODOLOGÍA

En el presente apartado se presentará el diseño de la investigación que será preexperimental, con enfoque cuantitativo y de tipo aplicada. Además, se indican las variables de incremento de conocimiento, de satisfacción, de motivación, de asertividad, reducción de tiempo de aprendizaje, de búsqueda información y eficiencia de la verificación de sintaxis. También se estableció la población con recientes estudios brindados por el SUNEDU (Superintendencia Nacional de Educación Superior Universitaria) y se estableció la muestra por conveniencia. Asimismo, se utilizarán los métodos de análisis como Shapiro-Wilk para la normalidad de los datos y Wilcoxon para la prueba de hipótesis. Finalmente, se da a conocer los ámbitos de la ética en la investigación.

3.1 Tipo y diseño de investigación

La presente investigación fue de enfoque cuantitativo. Al respecto, Cárdenas (2018) mencionó que una investigación de datos cuantitativos es cuando la información obtenida es numérica o la información recolectada se convierte en escalas numéricas. Además, en la investigación cuantitativa los datos de respuesta son numéricas, mientras que la investigación cualitativa son datos de recolección como imágenes o palabras (Cárdenas, 2018).

Asimismo, el diseño de esta investigación fue pre-experimental. En tal sentido Hernández y Mendoza (2018) mencionaron que se denomina pre-experimental o pre-experimento porque tiene un grado de control mínimo, y con el diseño de un grupo único; además, se aplica una preprueba al tratamiento experimental para luego aplicar la posprueba al estímulo.

3.2 Variables y operacionalización

El efecto del uso del chatbot para el aprendizaje de TypeScript es la variable de estudio. Asimismo, se plasma la matriz de operacionalización de variables en el **Anexo 3**. Consecutivamente se define los aspectos:

- A. Definición conceptual: Los bots informáticos se ejecutan de manera remota los cuales cumplen funciones concretas y adoptan el nombre de chatbot; estos interactúan con los humanos mediante el diálogo autóctono (Bonale et al., 2020). Además, (Richards et al., 2015)

mencionaron que, TypeScript extiende de JavaScript con anotaciones de tipado opcionales que son poco sólidas y que el compilador de TypeScript descarta cuando emite código.

B. Definición operacional: El efecto del chatbot para el aprendizaje de TypeScript es el beneficio para la resolución de dudas de programación de los programadores (Balderas et al., 2020; Bendezú y Canales, 2020).

C. Dimensiones:

- Conocimiento (Ariste y Ramírez, 2020; Bendezú y Canales, 2020; Meza y Yurivilca, 2020)
- Motivación (Ariste y Ramírez, 2020; Bendezú y Canales, 2020; Valdivia, 2021).
- Satisfacción (Ariste y Ramírez, 2020; Bendezú y Canales, 2020; Huaccachi y Mejía, 2021; Larico y Reyes, 2020; Valdivia Pérez, 2021).
- Asertividad (Ogosi, 2021; Valdivia, 2021; Iquiño y Castillo, 2021).
- Tiempo de aprendizaje (Daud et al., 2020; Balderas et al., 2020; Huaccachi y Mejía 2021).
- Eficiencia de la verificación de sintaxis (Balderas et al., 2020).

D. Indicadores:

- Incremento de conocimiento (Ariste y Ramírez, 2020; Bendezú y Canales, 2020; Meza y Yurivilca, 2020).
- Incremento de la motivación (Ariste y Ramírez, 2020; Bendezú y Canales, 2020; Valdivia, 2021).
- Incremento de la satisfacción (Ariste y Ramírez, 2020; Bendezú y Canales, 2020; Huaccachi y Mejía, 2021; Larico y Reyes, 2020; Valdivia Pérez, 2021).
- Incremento de la asertividad de las respuestas (Ogosi, 2021; Valdivia, 2021; Iquiño y Castillo, 2021).
- Reducción del tiempo de aprendizaje (Daud et al., 2020; Balderas et al., 2020; Huaccachi y Mejía 2021).
- Eficiencia de la verificación de sintaxis (Balderas et al., 2020).

3.3 Población, muestra y muestreo

Seguidamente, se da a conocer los conceptos relacionados de población, muestreo y unidad de análisis:

A. Población: Los especialistas de la Superintendencia Nacional de Educación Superior Universitaria (SUNEDU) en su informe estadístico de: Informe Bimestral Sobre La Realidad Universitaria En El Perú 2020, mencionaron que hubo 181,594 de egresados y 302,547 estudiantes de pregrado de Ingeniería y Tecnología en el 2018 (Superintendencia Nacional de Educación Superior Universitaria, 2020, p. 122). En efecto, la población objetiva alcanzó un total de 484,141 estudiantes y/o egresados de Ingeniería y Tecnología. Los criterios de inclusión y de exclusión fueron los siguientes:

- Criterios de inclusión: programadores profesionales, estudiantes y/o egresados con edades entre 18 a 35 años, que tengan conocimientos básicos de programación.
- Criterio de exclusión: programadores profesionales, estudiantes y/o egresados con edades entre 18 a 35 años, que no tengan algún conocimiento básico de programación.

B. Muestra y muestreo: Para la muestra se empleó el muestreo no probabilístico por conveniencia de 20 programadores profesionales, estudiantes y/o egresados que están contemplados dentro de la población del informe de SUNEDU.

C. Unidad de análisis: Es el grupo que forman parte de la investigación que son programadores profesionales, estudiantes y/o egresados entre los 18 y 35 años.

3.4 Técnicas e instrumentos de recolección de datos

La técnica de recolección de datos de la investigación fue la encuesta. Los instrumentos de recolección de datos fueron el cuestionario y la ficha de recolección de datos, los que se encuentran en los anexos 3, 4, 5, 6, 7, 8 y 9. Se consideró la validez de contenido para la variable, las dimensiones y los indicadores de la investigación, la que está sustentada con las fuentes de

información citadas en la matriz de operacionalización de la variable que se encuentra en el anexo 1. Además, se utilizó un nivel de confianza del 95% en las pruebas estadísticas utilizadas en la investigación.

3.5 Procedimientos

Al término del uso del chatbot se procedió a conseguir la información de la muestra de 20 programadores profesionales, estudiantes y/o egresados. De este modo, se hizo una prueba de conocimiento con el fin de medir el aprendizaje antes del uso del chatbot y posterior al uso del chatbot, donde se evaluó el dominio y conocimiento sobre el tema, la motivación hacia el aprendizaje, la satisfacción con el aprendizaje, la asertividad, el tiempo de aprendizaje y eficacia en la verificación de sintaxis.

El procedimiento para la recolección de datos fue el siguiente:

- A. Realizar la prueba piloto con una muestra inicial de 20 programadores.
- B. Aplicar los cuestionarios pre-test a la muestra completa. Los cuestionarios de motivación hacia el aprendizaje y satisfacción con el aprendizaje para el pre-test están en los anexos 6 y 7.
- C. Calcular la validez de constructo y la confiabilidad de la muestra completa. En el caso que no se obtuviera valores mayores o iguales a 0.7 se debe eliminar los casos atípicos.
- D. Usar el chatbot Fydebot.
- E. Aplicar los cuestionarios post-test a la muestra completa. Los cuestionarios de motivación hacia el aprendizaje y satisfacción con el aprendizaje para el post-test están en los anexos 8 y 17.
- F. Calcular los indicadores que servirán para la comprobación de las hipótesis.
- G. Realizar las pruebas de hipótesis.

3.6 Método de análisis de datos

En este apartado, se detallada los métodos de análisis de datos de la investigación. Se ejecutó la técnica de Shapiro-Wilk para la evaluación de la

normalidad, ya que la muestra alcanzada tuvo 20 programadores profesionales, estudiantes y/o egresados. Además, se utilizó la prueba T para muestras emparejadas cuando las distribuciones de las muestras se ajustaban a la distribución normal y el test de Wilcoxon para muestras relacionadas para las muestras que no se ajustaban a la distribución normal.

3.7 Aspectos éticos

La ética es fundamental para los estudiantes o profesionales que desean efectuar una investigación en la UCV conforme a la resolución de consejo universitario N° 0262 de la Universidad César Vallejo 2020 (Vicerrectorado de Investigación, 2020). En esta investigación se cumplió con la recolección de las fuentes de información de autores mediante el correcto uso del código de ética, en que la Universidad César Vallejo impulsa que todas las investigaciones sean originales indicando que el plagio es un delito, pues se hizo el citado de manera correcta y con el buen uso del formato deseado de acuerdo con la Universidad César Vallejo (Vicerrectorado de Investigación, 2020)

Por otra parte, se tomó en cuenta los valores que debe poseer con respecto a los principios de bioética, a lo que el Vicerrectorado de Investigación mencionó a la UNESCO en el cual propone que se comprometan y tomen principios éticos universales en relación al respeto y protección de sus derechos de la dignidad humana (Vicerrectorado de Investigación, 2020, p. 3).

En consecuencia, para la presente investigación se ha respetado la veracidad de la información de los saltados obtenidos, además como base de apoyo de la investigación se consideró la discreción y confidencialidad de la información. Los conceptos e ideas que se obtuvieron para avalar la presente investigación fueron citados debidamente con la ISO 690. Además, se hizo buen uso de las investigaciones citadas sin realizar plagio en el resultado de estudio.

IV. RESULTADOS

A continuación, se presentan los resultados alcanzados durante la investigación de los siguientes indicadores: incremento del conocimiento, incremento de la motivación, incremento de la satisfacción, incremento de la asertividad de las repuestas, reducción del tiempo de aprendizaje y eficiencia de la verificación de sintaxis. Además, fue evaluado el efecto de uso del chatbot para el aprendizaje de TypeScript.

4.1 Prueba de la hipótesis específica 1

HE10: El uso del chatbot no incrementó el conocimiento del aprendizaje de TypeScript.

HE11: El uso del chatbot incrementó el conocimiento del aprendizaje de TypeScript.

El resultado fue obtenido a través del grupo experimental de programadores profesionales, estudiantes y/o egresados interesados en aprender TypeScript. Los usuarios interactuaron con Fydebot mediante preguntas relacionados al desarrollo front-end. Al finalizar con el uso del chatbot se realizó la evaluación del incremento de conocimiento; la evaluación se detalló en los siguientes cuadros estadísticos.

Tabla 1 Incremento de conocimiento hacia el aprendizaje de TypeScript

		Estadístico	Error estándar
Pre-test	Media	8.75	0.680
Post-test	Media	14.90	0.717

Con respecto a la tabla 1 se da a conocer el incremento de conocimiento de los programadores profesionales, estudiantes y/o egresados que usaron el chatbot Fydebot para el aprendizaje de TypeScript. El desarrollo del pre-test alcanzó la media de 8.75 y en el caso del post-test se logró la media de 14.90; de tal modo se puede evidenciar la mejora en el post-test, con un incremento de conocimiento del 70.29%. Seguidamente se plasma la fórmula para el cálculo correspondiente.

IC = Incremento de conocimiento

PE = Prueba de entrada

PS = Prueba de Salida

$$IC = \frac{[PS - PE]}{PE} * 100\%$$

$$A = \frac{[14.90 - 8.75]}{8.75} * 100\% = 70.29\%$$

Prueba de normalidad

Según la prueba de normalidad se ejecutó la técnica de Shapiro-Wilk, porque la muestra alcanzada fue de 20 programadores profesionales, estudiantes y/o egresados. En la siguiente tabla se a conocer el pre-test y post-test.

Tabla 2 Prueba de normalidad del incremento de conocimiento

	Estadístico	GI	Sig.
Pre-test	0.957	20	0.480
Post-test	0.946	20	0.311

En el cual:

Prueba de Entrada

Después de aplicar la prueba de normalidad, que los resultados de la prueba de entrada (pre-test) reflejan un nivel de significancia mayor a 0.05, lo cual demuestra que hay una distribución normal.

Prueba de Salida

Después de aplicar la prueba de normalidad, que los resultados de la prueba de salida (post-test) reflejan un nivel de significancia mayor a 0.05, lo cual demuestra que hay una distribución normal.

Prueba T para muestras relacionadas

Tabla 3 Prueba T sobre el incremento de conocimiento.

	Media	N	Desviación estándar	Media de error estándar
Pre-test	8.75	20	3.041	0.680
Post-test	14.90	20	3.206	0.717

Por la ubicación de las variables de pre-test y post-test en la opción de la prueba T para muestras relacionadas, se obtuvo las siguientes hipótesis.

H0: La Nota antes = La Nota después

H1: La Nota antes \neq La Nota después

La media de la nota antes fue de 8.75 y su desviación estándar fue 3.041. Además, la media de la nota después fue 14.90 y su desviación estándar fue 3.206.

Tabla 4 Prueba de muestras emparejadas

	Media	Desv. Deviación	Desv. Error Promedio	95% intervalo de confianza de la diferencia		t	gl	Sig. (bilateral)
				Inferior	Superior			
Nota 1 – Nota 2	-6.1500	4.2491	0.9501	-8.1387	-4.1613	-6.473	19	0.000

Después de haber examinado los datos con el SPSS y considerando que la significancia de dos factores fue menor a 0.001, en consecuencia, se rechaza la HE1₀ y se acepta la HE1₁; es decir que el chatbot de aprendizaje de TypeScript tiene un incremento del conocimiento de 70.28%.

4.2 Prueba de la hipótesis específica 2

HE2₀: El uso del chatbot no incrementó la motivación hacia el aprendizaje de TypeScript.

HE2₁: El uso del chatbot incrementó la motivación hacia el aprendizaje de TypeScript.

Para el presente indicador se trabajó con 20 usuario (entre ellos programadores profesionales, estudiantes y/o egresados) que interactuaron con el chatbot (Fydebot) con preguntas relacionadas a TypeScript y una encuesta elaborada para medir el nivel de motivación del aprendizaje, la misma que se diseñó con una sola interrogante y valoradas con las puntuaciones del 1 a 5, donde 1 es nada motivado y 5 muy motivado. La evaluación se detalló en los siguientes cuadros estadísticos.

Tabla 5 Incremento de motivación hacia el aprendizaje TypeScript.

		Estadístico	Error estándar
Pre-test	Media	4.05	0.246
Post-test	Media	4.65	0.131

La tabla 5 refleja el nivel de incremento de la motivación hacia el aprendizaje obtenido de los programadores profesionales, estudiantes y/o egresados, teniendo una media el pre-test de 4.05 y en la media de post-test 4.65. Seguidamente se plasma la fórmula para el cálculo correspondiente.

IM = Incremento de motivación

CS = Cuestionario de salida

CE = Cuestionario de entrada

$$IM = \frac{[CS - CE]}{CE} * 100\%$$

$$IM = \frac{[4.65 - 4.05]}{4.05} * 100\% = 14.81\%$$

Prueba de normalidad

Según la prueba de normalidad se ejecutó la técnica de Shapiro-Wilk, porque la muestra alcanzada fue de 20 programadores profesionales, estudiantes y/o egresados. En la siguiente tabla se a conocer el pre-test y post-test.

Tabla 6 Prueba de normalidad el incremento de motivación hacia el aprendizaje de TypeScript.

	Estadística	gl	Sig.
Pre-test	0.789	20	0.001
Post-test	0.632	20	0.000

Prueba de entrada

Después de aplicar la prueba de normalidad, que los resultados de la prueba de entrada (pre-test) reflejan un nivel de significancia menor a 0.05, lo cual demuestra que no hay una distribución normal.

Prueba de salida

Después de aplicar la prueba de normalidad, que los resultados de la prueba de salida (post-test) reflejan un nivel de significancia menor a 0.05, lo cual demuestra que no hay una distribución normal.

Prueba de Wilcoxon

En la tabla 7 se observa la prueba de Wilcoxon con los siguientes detalles.

Tabla 7 Rangos de prueba de signos – el incremento de motivación hacia el aprendizaje de TypeScript.

		N	Rango promedio	Suma de rangos
Pre-test	Rangos negativos	1 ^a	2.50	2.50
Post-test		7 ^b	4.79	33.50
	Empates	12 ^c		
	Total	12		

a) Prueba de salida < prueba de entrada

b) Prueba de salida > prueba de entrada

c) Prueba de salida = prueba de entrada

Tabla 8 Estadística de prueba Z – Incremento de motivación hacia el aprendizaje de TypeScript.

Estadísticos de prueba ^a	
	Prueba de salida - prueba de entrada
z	-2.203 ^b
Siga sin. (bilateral)	0.028

a. Prueba de rangos con signo de Wilcoxon.

b. Se basa en rangos negativos.

Después de haber examinado los datos con el SPSS en la zona Z se obtuvo -2.203b; por lo tanto, se ubicó en la región de rechazo y se alcanzó un valor $p = 0.028 < 0.05$, por ende, se desestima la HE2⁰ y se reconoce la HE2¹; en otras expresiones, las medias entre el pre-test y post-test son de gran relevancia, por ello, se reconoció y aceptó que el uso del chatbot incrementa la motivación hacia el aprendizaje de TypeScript con un incremento de motivación de 14.81%.

4.3 Prueba de la hipótesis específica 3

HE3₀: El uso del chatbot no incrementó la satisfacción con el aprendizaje de TypeScript.

HE3₁: El uso del chatbot incrementó la satisfacción con el aprendizaje de TypeScript.

Datos estadísticos del Incremento de satisfacción

Para el presente indicador se trabajó con 20 usuario (entre ellos programadores profesionales, estudiantes y/o egresados) que interactuaron con el chatbot (Fydebot) con preguntas relacionadas a TypeScript y una encuesta elaborada para medir el nivel de satisfacción del aprendizaje, la misma que se diseñó con una sola interrogante y valoradas con las puntuaciones del 1 a 5, donde 1 es nada satisfecho y 5 muy satisfecho. La evaluación se detalló en los siguientes cuadros estadísticos.

Tabla 9 Incremento de la satisfacción con el aprendizaje TypeScript

		Estadístico	Error estándar
Pre-test	Media	4.05	0.235
Post-test	Media	4.65	0.131

La tabla 9 refleja el nivel de incremento de la satisfacción hacia el aprendizaje obtenido de los programadores profesionales, estudiantes y/o egresados, teniendo una media el pre-test de 4.05 y en la media de post-test 4.65. Seguidamente se plasma la fórmula para el cálculo correspondiente.

IS = Incremento de satisfacción

PS = Prueba de salida

PE = Prueba de entrada

$$IS = \frac{[PS - PE]}{PE} * 100\%$$

$$IS = \frac{[4.65 - 4.05]}{4.05} * 100\% = 14.81\%$$

Prueba de normalidad

Para la prueba de normalidad se utilizó el método de Shapiro-Wilk, ya que la muestra fue de 20 personas y por ser menor a 50 se utiliza el método en mención. A continuación, se presentan los resultados para ambas pruebas (entrada y salida).

Tabla 10 Pruebas de normalidad del incremento de la satisfacción con el aprendizaje.

	Estadístico	gl	Sig.
Pre-test	0.815	20	0.001
Post-test	0.632	20	0.000

Prueba de entrada

Después de aplicar la prueba de normalidad, que los resultados de la prueba de entrada (pre-test) reflejan un nivel de significancia menor a 0.05, lo cual demuestra que no hay una distribución normal

Prueba de salida

Después de aplicar la prueba de normalidad, que los resultados de la prueba de salida (post-test) reflejan un nivel de significancia menor a 0.05, lo cual demuestra que no hay una distribución normal

Prueba de Wilcoxon

Tabla 11 Rangos de prueba de signos – Incremento de la satisfacción con el aprendizaje.

		N	Rango promedio	Suma de rangos
Pre-test	Rangos negativos	2 ^a	4.00	8.00
Post-test				
	Rangos positivos	9 ^b	6.44	58.00
	Empates	9 ^c		
	Total	20		

a) Prueba de salida < prueba de entrada

b) Prueba de salida > prueba de entrada

c) Prueba de salida = prueba de entrada

Tabla 12 Estadística de prueba Z – Incremento de la satisfacción con el aprendizaje

	Prueba de salida - Prueba de entrada
z	-2.292 ^b
Sig. Asin. (bilateral)	0.022

a) Prueba de rangos con signo de Wilcoxon

b) Se basa en rangos negativos

Después de haber examinado los datos con el SPSS en la zona Z se obtuvo - 2.292b; por lo tanto, se ubicó en la región de rechazo y se alcanzó un valor $p = 0.022 < 0.05$, por ende, se desestima la HE3⁰ y se reconoce la HE3¹; en otras expresiones, las medias entre el pre-test y post-test son de gran relevancia, por ello, se reconoció y aceptó que el uso del chatbot incrementa la motivación hacia el aprendizaje de TypeScript con un incremento de motivación de 14.81%

4.4 Prueba de la hipótesis específica 4

HE4₀: El uso del chatbot no incrementó la asertividad de las respuestas del aprendizaje de TypeScript.

HE4₁: El uso del chatbot incrementó la asertividad de las respuestas del aprendizaje de TypeScript.

Tabla 13 Medición de la asertividad por cada pregunta del usuario

				Nada preci so	Algo preci so	Medianam ente preciso	Altame nte preciso	Totalmente preciso
Nº	Pregunta del usuario	Respuesta del chatbot	Respuesta Código del chatbot	1	2	3	4	5
1	como se declara un any	Ok, ahora un ejemplo de del tipo de dato any:	//la variable 'filtro' puede tomar cualquier valor let filtro: any = "Hello World!"; filtro = 23; filtro = true;					5
2	como funciona un for	Puedes usar un bucle de la siguiente manera:	//bucle for for (let i = 0; i < this.paises.lenght; i++) { console.log(this.paises[i].n ame); }				4	
3	Ejemplo de boolean	boolean representa los dos valores true y false				3		
4	como se declara un string	Aquí un ejemplo del tipo de dato string	//declaración normal let nombreCompleto: string; nombreCompleto = "tsbot web"; //declaración simple let nombreCompleto2 = "tsbot web";					5

				Nada preciso	Algo preciso	Medianam ente preciso	Altame nte preciso	Totalmente preciso
Nº	Pregunta del usuario	Respuesta del chatbot	Respuesta Código del chatbot	1	2	3	4	5
5	como se declara el dato boolean	Aquí un ejemplo de la declaración de tipo de dato boolean	// declaración normal let esActivo: boolean; esActivo = true; // declaración simple let esAdmin = true;					5
6	tipo de dato string	string representa valores de cadena como "Hola, mundo"						5
7	ejemplo de any	Ok, ahora un ejemplo de del tipo de dato any:	//la variable 'filtro' puede tomar cualquier valor let filtro: any = "Hello World!"; filtro = 23; filtro = true;					5
N	-	-	-					

En la tabla 13 se llega a visualizar una muestra de la recolección de preguntas del usuario y la respuesta del chatbot; en esta tabla se encuentran las preguntas realizadas por los programadores profesionales, estudiantes y/o egresados y las respuestas del chatbot en texto y/o código con la sintaxis correcta de TypeScript. Además, la medición de la precisión de la asertividad donde 1 es nada preciso y 5 es totalmente preciso, los puntajes asignados a cada respuesta fue evaluado y calificado por un experto en la materia.

Tabla 14 Puntaje total de la asertividad por cada usuario

Número de participantes	Puntaje por usuario
1	173
2	195
3	175
4	190
5	171
6	161
7	188
8	189
9	158
10	188
11	183
12	171

Número de participantes	Puntaje por usuario
13	171
14	190
15	168
16	178
17	171
18	155
19	171
20	170

En la tabla 14 se visualiza el puntaje total de la asertividad de las respuestas por usuario, para determinar el puntaje total por usuario, se hizo la suma del puntaje de la precisión por cada respuesta del chatbot. La muestra para la evaluación fue de 40 interacciones por usuario obteniendo un puntaje promedio de 173.3; para el cálculo de la asertividad se utilizó la siguiente formula.

AR = Asertividad de la respuesta

PP = Promedio del puntaje

MP = Máximo puntaje

$$AR = \frac{PP}{MP} * 100\%$$

$$AR = \frac{176.3}{200} * 100\% = 88.15\%$$

En tal sentido, se realizó la comparación del resultado obtenido con los estudios de Valdivia (2021). Se determinó que en el incremento de la asertividad que se obtuvo de 88.15% fue mayor a los resultados de Valdivia (2021), quien llegó a obtener una asertividad de 15.58%; por lo tanto, se desestima la HE4₀ y se reconoce la HE4₁ donde el uso del chatbot incrementó la eficiencia de verificación de sintaxis de TypeScript en los programadores profesionales, estudiantes y/o egresados.

4.5 Prueba de la hipótesis específica 5

HE5₀: El uso del chatbot no redujo el tiempo de aprendizaje de TypeScript.

HE5₁: El uso del chatbot redujo el tiempo de aprendizaje de TypeScript.

Tabla 15 Tiempo de interacción por usuario en minutos

Usuario	Tiempo de aprendizaje con el Chatbot en minutos
1	121
2	84
3	109
4	106
5	101
6	106
7	104
8	98
9	84
10	97
11	107
12	108
13	137
14	106
15	106
16	104
17	102
18	105
19	109
20	93

En la tabla 15 se visualiza el tiempo de interacción de los programadores profesionales, estudiantes y/o egresados con respecto al uso del chatbot; se obtuvo el tiempo de aprendizaje en minutos, donde 104.35 minutos fue el promedio de aprendizaje durante los 7 días de prueba que se realizaron con los usuarios. Por lo tanto, se llegó a comparar el tiempo de aprendizaje con la investigación de Huaccachi y (2021). Se concluyó que el tiempo de aprendizaje obtenido de 104.35 minutos fue menor a los resultados de Huaccachi y (2021), quien obtuvo un tipo de aprendizaje de 180.8167 minutos. En tal sentido, se desestima la HE5₀ y se reconoce la HE5₁ donde uso del chatbot redujo el tiempo de aprendizaje en los programadores profesionales, estudiantes y/o egresados.

Por otra parte, se presenta la medición de la eficiencia en la verificación de sintaxis. Hernández y Mendoza (2018) mencionaron que los datos descriptivos se limitan a medir y describir los aspectos, variables, componentes o dimensiones del problema o fenómeno a investigar; al ser un estudio de medición, recolección de datos y reporte de información no tienen una hipótesis. Entonces, se describe la medición de la eficiencia en la verificación de sintaxis del chatbot para la verificación de sintaxis.

Tabla 16 Medición de la verificación de sintaxis por cada pregunta del usuario

				Nada preciso	Algo preciso	Medianamente preciso	Altamente preciso	Totalmente preciso
Nº	Pregunta del usuario	Respuesta del chatbot	Respuesta code error	1	2	3	4	5
1	<code>arr:Type[]</code>	<code>arr:Type[]</code>	1. Cannot find name 'Type'. 2. An element access expression should take an argument.					5
2	<code>class vehiculo{ id: number modelo:string placa:string numeroMotor:numero mbre }</code>	<code>class vehiculo{ id: number modelo:string placa:string numeroMotor:numero mbre }</code>	1. Cannot find name 'numero'.					5
3	<code>class vehiculo{ id: number modelo:string placa:string numeroMotor:numero mber }</code>	<code>class vehiculo{ id: number modelo:string placa:string numeroMotor:numero mber }</code>						5
N	-	-	-					

En la tabla 16 se visualiza una parte de la muestra de la recolección de datos por usuario para la eficiencia en la verificación de sintaxis; en la columna 2 se encuentran los códigos de programación enviados por los usuarios, en la columna 3 se visualiza la respuesta del chatbot con la sintaxis que envió el usuario y en la 4 columna se muestran los errores de sintaxis que encontró el chatbot; si la sintaxis del código es correcta no se mostrará ningún error y responderá un 'Ok' de manera coloquial. Además, en la tabla se encuentran las mediciones de la precisión de sintaxis donde 1 es nada preciso y 5 es totalmente

preciso. Asimismo, la comprobación de la verificación de sintaxis lo realizó un experto en la materia.

Para la evaluación de la eficiencia en la verificación de sintaxis, se tomaron 12 interacciones de sintaxis de 20 programadores profesionales, estudiantes y/o egresados, el puntaje máximo era de 60 por usuario. En este sentido, el experto de TypeScript analizó y calificó cada una de las respuestas de sintaxis del chatbot de los 20 usuarios, llegando a obtener un promedio de puntaje de 60. Por consiguiente, la eficiencia de la verificación de sintaxis es 100% preciso.

4.6 Prueba de la hipótesis general

Debido a que las hipótesis específicas fueron aceptadas, también la hipótesis general: “El uso del chatbot incrementó el aprendizaje de TypeScript.” fue aceptada.

4.7 Resumen

Seguidamente se presenta el resumen de las hipótesis desestimadas y aceptadas en esta investigación;

Cód.	Hipótesis	Resultado
HE1	El uso del chatbot incrementó el conocimiento del aprendizaje de TypeScript.	Aceptada
HE2	El uso del chatbot incrementó la motivación hacia el aprendizaje de TypeScript.	Aceptada
HE3	El uso del chatbot incrementó la satisfacción con el aprendizaje de TypeScript.	Aceptada
HE4	El uso del chatbot incrementó la asertividad de las respuestas del aprendizaje de TypeScript.	Aceptada
HE5	El uso del chatbot redujo el tiempo de aprendizaje de TypeScript.	Aceptada
HG	El uso del chatbot incrementó el aprendizaje de TypeScript.	Aceptada

V. DISCUSIÓN

En efecto, el chatbot logró el objetivo de la mejora del aprendizaje de TypeScript, específicamente obteniendo mejora en los siguientes indicadores: incremento del conocimiento en 70.29%, incremento en la motivación en 14.81%, incremento de la satisfacción en 14.81%, incremento en la asertividad de la respuesta en 88.15%, reducción en el tiempo promedio de aprendizaje de 104.35 minutos y por último, la eficiencia de verificación de sintaxis de 100%. En líneas generales el chatbot para el aprendizaje de TypeScript es una buena opción para el aprendizaje de este lenguaje.

El incremento de conocimiento logrado con el chatbot fue 70.29%, resultado que fue mayor al resultado de Meza y Yurivilca (2020), quienes lograron un incremento de conocimiento 64.93% con la implementación del chatbot para el aprendizaje de sobre seguridad electrónica. Este último chatbot no contó con una ruta de aprendizaje exacta donde los usuarios puedan tener el panorama exacto de los temas que maneja el chatbot. Sin embargo, el incremento de conocimiento logrado con el chatbot de 70.29% fue menor al logrado en los estudios de Ariste y Ramírez (2020) y Bendezú y Canales (2020), en los que se logró 85% y 96.84%, respectivamente, tal como se detalla en el siguiente párrafo.

Ariste y Ramírez (2020) obtuvieron un incremento en el conocimiento de 85% con la investigación chatbot para el aprendizaje de la fotosíntesis, siendo mayor al resultado de esta investigación, debido a que implementaron su chatbot con la técnica de web scraping, extrayendo información de los sitios web, lo cual ayudó a mejorar la respuesta del chatbot con diversas fuentes de información. Además, Bendezú y Canales (2020) lograron un incremento en el conocimiento de 96.84% con la aplicación de aprendizaje del lenguaje de programación JavaScript, siendo mayor a lo obtenido en esta investigación debido a que la aplicación de aprendizaje de JavaScript contaba con evaluaciones de seis preguntas por cada módulo de aprendizaje, reforzando aún más lo aprendido en el módulo.

Por otra parte, el incremento de la motivación de aprendizaje de TypeScript fue 14.81%, siendo mayor que la investigación de Bendezú y Canales (2020) quienes lograron un incremento en la motivación de 14.49% de una

muestra de 33 estudiantes de los primeros ciclos de la carrera de sistemas con la aplicación móvil de aprendizaje de JavaScript llamada AprenderJS. A diferencia de la aplicación AprenderJS, esta investigación contó con el módulo de respuestas destacadas, dando la libertad a los usuarios a que puedan seleccionar cualquier respuesta a su conveniencia del chatbot y tenerlo todo en un solo módulo para el rápido acceso. Sin embargo, el incremento de la motivación logrado con el chatbot de 14.81% fue menor a lo conseguido en los estudios de Ariste y Ramírez (2020) y Valdivia (2021), en los que se logró 82% y 94.93%, respectivamente, tal como se detalla en el siguiente párrafo.

Ariste y Ramírez (2020) consiguieron un incremento en la motivación de 82% hacia el aprendizaje de la fotosíntesis en los estudiantes de nivel secundaria con el uso del chatbot. Asimismo, Valdivia (2021) desarrolló un chatbot para el aprendizaje del cuidado de bebés, consiguiendo un incremento de la motivación de 94.93%. Los chatbots mencionados obtuvieron un incremento en la motivación mayor que los resultados de esta investigación (14.81%), ya que la muestra de 20 programadores profesionales, estudiantes y/o egresados de esta investigación en su mayoría estaban interesados y motivados en aprender TypeScript, obteniendo una media alta en el pre-test, por lo que no hubo tanta diferencia con el post-test; en consecuencia, se tuvo un incremento de la motivación menor que en otros estudios.

El incremento de la satisfacción del chatbot fue 14.81%, siendo mayor al 13.47% logrado en la investigación de Bendezú y Canales (2020), ya que no implementaron la interacción de sintaxis del código, a diferencia de esta investigación que implementó la verificación de sintaxis en la que el usuario podía aprender el lenguaje de programación escribiendo código. Sin embargo, el incremento de la satisfacción logrado con el chatbot de esta investigación (14.81%) fue menor a lo obtenido en los estudios de Huaccachi y Mejía (2021), Larico y Reyes (2020), Ariste y Ramírez (2020) y Valdivia (2021) en los que se logró 32.26%, 69.68%, 80% y 96.97%, respectivamente, tal como se detalla en el siguiente párrafo.

Huaccachi y Mejía (2021) alcanzaron un incremento de satisfacción de 32.26% con la aplicación móvil para el aprendizaje de acciones ante la violencia

contra menores de edad. Además, Larico y Reyes (2020) obtuvieron un incremento de satisfacción de 69.68% con el chatbot de aprendizaje de la limpieza y desinfección en el hogar. Ariste y Ramírez (2020) lograron un incremento en la satisfacción del 80% con el uso del chatbot para el aprendizaje de la fotosíntesis. Adicionalmente, Valdivia (2021) logró un incremento de la satisfacción de 96.67% con el chatbot de aprendizaje del cuidado de bebés. En tal sentido, la satisfacción de esta investigación fue menor a las investigaciones mencionadas, ya que la muestra que se obtuvo de 20 programadores profesionales, estudiantes y/o egresados obtuvieron una satisfacción alta con los medios actuales de aprendizaje en el pre-test, por lo que no hubo tanta diferencia con el post-test.

La asertividad de las respuestas del chatbot de esta investigación fue 88.15%, siendo mayor que el obtenido por el chatbot de Valdivia (2021), quien llegó a obtener una asertividad de 15.58%. Asimismo, Iquiño y Castillo (2021) obtuvieron una asertividad de 89.52% con el chatbot de aprendizaje sobre los procesos de vacunación, lo que fue ligeramente mayor a la obtenida en esta investigación. A diferencia de los chatbots mencionados, esta investigación contó con más de 800 preguntas y respuestas como parte de la base de conocimientos y también tuvo fuentes directas de la página oficial de TypeScript. Además, la asertividad fue analizada y evaluada por un experto en la materia, mientras que los chatbots de Valdivia (2021) y Iquiño y Castillo (2021) fueron los usuarios quienes evaluaron la asertividad de las respuestas.

Por último, el tiempo de aprendizaje obtenido fue 104.35 minutos, siendo menor a los 180.8167 minutos de la investigación de Huaccachi y Mejía (2021) con la aplicación móvil de aprendizaje de acciones ante la violencia a menores de edad. El tiempo de aprendizaje de esta investigación fue menor porque las respuestas del chatbot en un 70% estuvieron compuestas por ejemplos con fragmento de códigos con la sintaxis correcta de TypeScript.

VI. CONCLUSIONES

Las conclusiones de la investigación fueron las siguientes:

1. Se consiguió un incremento del conocimiento de 70.29%, debido a la construcción de una arquitectura web sólida para el chatbot, la implementación de un editor de código en la web y la elaboración de una ruta de aprendizaje.
2. Se incrementó la motivación de los usuarios en 14.81%, ya que en la prueba inicial los usuarios en su mayoría estaban interesados y muy motivados en aprender TypeScript, de tal forma que no hubo tanta diferencia con el post-test. No obstante, se logró incrementar la motivación con la implementación del módulo repuestas destacadas, donde el usuario pudo tener un rápido acceso a las respuestas previamente seleccionadas.
3. Se obtuvo un incremento en la satisfacción de 14.81%, ya que en la prueba inicial los usuarios en su mayoría se sentían satisfechos con los medios de información actuales para el aprendizaje de TypeScript, de tal forma que no hubo tanta diferencia con el post-test. Sin embargo, se logró incrementar en la satisfacción con la implementación de la verificación de sintaxis, siendo beneficioso para el rápido aprendizaje de los usuarios.
4. Con respecto al nivel de asertividad de la respuesta se logró un 88.15%, debido a la diversidad de preguntas y respuestas que fueron ingresadas a la base de conocimientos con fuentes directas de la página oficial de TypeScript. Cabe resaltar que la asertividad de las respuestas del chatbot fue evaluada por un experto en la materia.
5. Se redujo el tiempo de aprendizaje en 76.47 minutos, debido a la base de conocimientos de Fydebot, lo cual tuvo un porcentaje mayor en las respuestas con ejemplos de la sintaxis correcta de TypeScript.
6. Se logró una eficiencia de la verificación de sintaxis del 100%, debido a la verificación se implementó con el mismo API de TypeScript, por tal motivo se logró la máxima puntuación en la verificación de sintaxis.

7. Para el aprendizaje relacionado con la programación es mejor implementar una aplicación con gamificación y microlearning o un chatbot en la web con la técnica web scraping en lugar de un chatbot en cualquier red social (Facebook, Telegram y WhatsApp), ya que así es posible interactuar con las librerías de código del lenguaje de programación correspondiente.

VII. RECOMENDACIONES

Las recomendaciones para futuras investigaciones son las siguientes:

1. Implementar los chatbots de aprendizaje en un sistema web con una arquitectura de interacción continua, de este modo los usuarios podrán aprender de manera cómoda utilizando todos los recursos de la web evitando distracciones de las redes sociales.
2. Utilizar una base de datos relacional para el mejor control de los datos, guardando la interacción de los usuarios con el chatbot y con un registro de tiempos que contenga milisegundos, para medir el tiempo de la respuesta del chatbot.
3. Mejorar el algoritmo de Fydebot con el algoritmo base de TF-IDF, de tal forma que en la comprobación de las similitudes se pueda aumentar la importancia de palabras clave de la bolsa de palabras, para lograr una mejor asertividad en las repuestas.
4. Trabajar con microservicios para separar el back-end del front-end, obteniendo un mejor control de la persistencia en la base de datos. Además, con la implementación del back-end el chatbot puede convertirse en multiplataforma, ya que toda la lógica de chatbot lo tendrá la parte del back-end y a través de la API se puede exponer los servicios e implementar de manera sencilla en Facebook, Telegram, web, etc.
5. Utilizar Docker en la implementación de la lógica del chatbot, ya que ayuda englobar todas las dependencias del aplicativo en un solo contenedor, consiguiendo que la aplicación o el chatbot funcione de manera autónoma en cualquier ambiente del proyecto (local, desarrollo, producción, etc.).
6. Emplear el marco de trabajo de DevOps, ya que ayuda en desplegar los nuevos cambios o mejoras de la aplicación con la interacción continua, además se puede manejar en diferentes ambientes de desarrollo.

7. Implementar evaluaciones por cada módulo o tema de aprendizaje para la mejora del aprendizaje a través del chatbot.

REFERENCIAS

- ALAAELDIN, R., ASFOURA, E. y ABDEL-HAQ, M.S., 2021. Developing Chatbot System To Support Decision Making Based on Big Data Analytics. *Journal of Management Information and Decision Science*, vol. 24, no. 2, pp. 1-15. ISSN 15325806.
- ARISTE MALAGA, J.S. y RAMÍREZ PAREJA, E.J., 2020. Chatbot para el aprendizaje de la fotosíntesis utilizando la técnica web scraping [en línea]. S.l.: Universidad Cesar Vallejo. Disponible en: http://repositorio.ucv.edu.pe/bitstream/handle/20.500.12692/47102/Gutierrez_RS-SD.pdf?sequence=1&isAllowed=y.
- AUZ ROSERO, Deisy Tatiana. Creación de un prototipo de Chatbot para el apoyo en el proceso de enseñanza-aprendizaje virtual de la Cátedra de Ingeniería de software de la Carrera de Ciencias de la Computación. 2021. Tesis de Licenciatura.
- BALDERAS, A., PERSON, T., MEJIAS-RAMIREZ, D., PALOMO-DUARTE, M. y RUIZ-RUBE, I., 2020. Desarrollo de un agente conversacional para el aprendizaje de SQL [en línea]. 2020. S.l.: s.n. Disponible en: <http://hdl.handle.net/10498/23892>.
- BEN HAIM, Yael; TOM TOV, Elad. A Streaming Parallel Decision Tree Algorithm. *Journal of Machine Learning Research*, 2010, vol. 11, no 2.
- BENDEZÚ TARQUI, J.M. y CANALES ALCALDE, A.Da., 2020. Aplicación móvil con gamificación y microlearning para el aprendizaje de programación de JavaScript [en línea]. S.l.: s.n. ISBN 0000000344128. Disponible en: http://repositorio.ucv.edu.pe/bitstream/handle/20.500.12692/47102/Gutierrez_RS-SD.pdf?sequence=1&isAllowed=y.
- BONALES, G., PRADILLA, N. y CITLALI, E., 2020. Chatbot como herramienta comunicativa durante la crisis sanitaria de la COVID-19 en España. *Revista ComHumanitas*, vol. 11, no. 3, pp. 1-22.
- CÁRDENAS, J., 2018. Investigación cuantitativa. Ministerio Federal de Cooperación Económica y Desarrollo [en línea], pp. 70. DOI 10.17169/refubium-216. Disponible en: https://www.programa-trandes.net/Ressources/Manuales/Manual_Cardenas_Investigacion_cuantitativa.pdf.
- Daud, S. H. M., Teo, N. H. I., & Zain, N. H. M. (2020). E-java chatbot for learning programming language: A post-pandemic alternative virtual tutor.

- International Journal of Emerging Trends in Engineering Research, 8(7), 3290–3298. <https://doi.org/10.30534/ijeter/2020/67872020>
- DE LA ROSA, M., 2020. Chatbots y RRHH , la puerta de entrada de la IA. Capital Humano [en línea], vol. 353, no. 1, pp. 20-25. Disponible en: <http://search.ebscohost.com/login.aspx?direct=true&db=fap&AN=143060985&lang=es&site=eds-live&scope=site>.
- DE LUIS MOLERO, T., 2019. Chatbots, una solución ventajosa para las empresas. Capital Humano [en línea], pp. 91-99. Disponible en: <http://search.ebscohost.com/login.aspx?direct=true&db=fap&AN=140208222&authtype=sso&custid=s9226804&lang=es&site=eds-live&scope=site>.
- GELEMET, Özlem; AYDIN, Hakan; ÇETİNKAYA, Ali. Netflix verileri üzerinde TF-IDF algoritması ve Kosinüs benzerliği ile bir İçerik Öneri Sistemi Uygulaması. AJIT-e: Bilişim Teknolojileri Online Dergisi, 2022, vol. 13, no 48, p. 31-51.
- GOLEC, Marcin; PLECHAWSKA-WÓJCIK, Małgorzata. Análisis comparativo de marcos que utilizan TypeScript para crear aplicaciones de servidor. Revista del Instituto de Ciencias de la Computación , 2022, vol. 23, pág. 128-134.
- GONZALES, A., 2018. Aplicación móvil con Chatbot para el aprendizaje en el uso de la plataforma Canvas en docentes de la UTP. Universidad Cesar Vallejo [en línea], pp. 0-93. Disponible en: <http://repositorio.ucv.edu.pe/handle/UCV/30232>.
- GRZEGORZ, K., 2021. Why use TypeScript? Could it be the best way to write frontend in 2021? [en línea]. Disponible en: <https://tsh.io/blog/why-use-typescript/>.
- HERNÁNDEZ SAMPIERI, R. and MENDOZA TORRES, C.P., 2018. Metodología de la investigación : las rutas cuantitativa, cualitativa y mixta. S.I.: s.n. ISBN 9781456260965.
- HUACCACHI LEÓN, P.T. and MEJÍA ALVARADO, A.M., 2021. Aplicación móvil para el aprendizaje de acciones ante violencia a menores de edad. . S.I.:
- IÁÑEZ GONZÁLEZ, S.F., 2018. Desarrollo de un Chatbot con Dialogflow en el Marco de las Ciudades Inteligentes. Universidad Carlos III de Madrid, pp. 1-91.
- IQUIÑO HUAMANI, R.C. and VALERIO CASTILLO, F.A., 2021. Chatbot para el aprendizaje sobre los procesos para la vacunación para prevenir la Covid-

19. . S.I.:

INBENTA, D.F., 2021. Los chatbots se convierten en el aliado perfecto para el contact center. [en línea]. Disponible en:

<https://tecnohotelnews.com/2021/01/04/chatbots-contact-center-aliado/>.

SUPERINTENDENCIA NACIONAL DE EDUCACIÓN SUPERIOR

UNIVERSITARIA, 2020. Perú: Informe Bimestral Sobre La Realidad

Universitaria En El Perú 2020. Disponible en:

<https://cdn.www.gob.pe/uploads/document/file/1230044/Informe%20Bienal.pdf>.

JIANG, Zijian; ZHONG, Hao; MENG, Na. Investigating and recommending co-changed entities for JavaScript programs. *Journal of Systems and Software*, 2021, vol. 180, p. 111027.

KASINATHAN, V., MUSTAPHA, A. y BIN, C.K., 2021. A customizable multilingual chatbot system for customer support. *Annals of Emerging Technologies in Computing*, vol. 5, no. Special issue 5, pp. 51-59. ISSN 2516029X. DOI 10.33166/AETiC.2021.05.006.

KRISTENSEN, Erik Krogh; MØLLER, Anders. Type test scripts for TypeScript testing. *Proceedings of the ACM on Programming Languages*, 2017, vol. 1, no OOPSLA, p. 1-25.

LARICO MAMANI, J.R. y REYES ESPINOZA, L.F., 2020. Chatbot para el aprendizaje de la limpieza y desinfección para protegerse de la COVID 19 en el hogar [en línea]. S.I.: s.n. ISBN 0000000344128. Disponible en: http://repositorio.ucv.edu.pe/bitstream/handle/20.500.12692/47102/Gutierrez_RS-SD.pdf?sequence=1&isAllowed=y.

LI, Mujin; XU, Honghui; DENG, Yong. Evidential decision tree based on belief entropy. *Entropy*, 2019, vol. 21, no 9, p. 897.

LI, Y., 2019. Front-end testing – an important part of quality assurance in front-end of quality assurance in front-end.

MERKEL, M., WAGNER, S. and BOGNER, J., 2021. Do TypeScript Applications Show Better Software Quality than JavaScript Applications? A Repository Mining Study on GitHub. . S.I.:

MEZA MANCO, Rodny José Fabricio; YURIVILCA ROMÁN, Marco Antonio. Chatbot para aprendizaje sobre seguridad electrónica. 2020.

MEYER, André N., et al. Today was a good day: The daily life of software

- developers. IEEE Transactions on Software Engineering, 2019, vol. 47, no 5, p. 863-880.
- OGOSI, José Antonio. Chatbot del proceso de aprendizaje universitario: Una revisión sistemática. Alpha Centauri, 2021, vol. 2, no 2, p. 29-43.
- RODRÍGUEZ PANCHANA, Carlos Steven; PINO CEDEÑO, Roberto Carlos. Chatbot con habilidades sociales para interactuar con Usuarios del Servicio de mesa de ayuda que brinda la empresa "Hitss Ecuador". 2019. Tesis de Licenciatura. Universidad de Guayaquil. Facultad de Ciencias Matemáticas y Físicas. Carrera de Ingeniería en Sistemas Computacionales.
- RICHARDS, G., NARDELLI, F.Z. y VITEK, J., 2015. Concrete types for TypeScript. Leibniz International Proceedings in Informatics, LIPIcs, vol. 37, pp. 76-100. ISSN 18688969. DOI 10.4230/LIPIcs.ECOOP.2015.76.
- RODRIGUEZ GARCIA, J., 2021. El lenguaje de programación Typescript. [en línea], pp. 1-11. Disponible en: <https://www.campusmvp.es/recursos/post/typescript-contra-javascript-cual-deberias-utilizar.aspx>.
- SCHWABER, Ken; SUTHERLAND, Jeff. The Scrum Guide. 2020. Accessed April, 2021.
- VALDIVIA PÉREZ, C.A., 2021. Chatbot para el aprendizaje del cuidado de bebés. . S.l.:
- VERBEEK, Hilde. An overview of Microsoft TypeScript. 2020.
- VICERRECTORADO DE INVESTIGACIÓN DE LA UNIVERSIDAD CÉSAR VALLEJO. Resolución Rectoral N° 0442-2019/UCV. Universidad Cesar Vallejo, 2020
- WANG, Shaowei; CHEN, Tse-Hsun; HASSAN, Ahmed E. Understanding the factors for fast answers in technical Q&A websites. Empirical Software Engineering, 2018, vol. 23, no 3, p. 1552-1593.
- WHITING, R., 2020. 5 tendencias emergentes de IA y aprendizaje automático para ver en 2021. [en línea]. Disponible en: <https://www.crn.com/news/cloud/5-emerging-ai-and-machine-learning-trends-to-watch-in-2021>.
- YIN, J., GOH, T.T., YANG, B. and XIAOBIN, Y., 2021. Conversation Technology With Micro-Learning: The Impact of Chatbot-Based Learning

on Students' Learning Motivation and Performance. *Journal of Educational Computing Research*, vol. 59, no. 1, pp. 154–177. ISSN 15414140. DOI 10.1177/0735633120952067.

Anexo 1: Matriz de operacionalización de variables

Tabla 17: Matriz de operacionalización de variables

Variable	Definición Conceptual	Definición Operacional	Dimensión	Indicador	Instrumento	Unidad de Medida
Efecto del uso del Chatbot para el aprendizaje de TypeScript (Balderas et al., 2020; Bendezú y Canales, 2020)	Los bots informáticos se ejecuta de manera remota lo cual cumplen funciones concretas y adoptan el nombre de Chatbot; estos interactúan con los humanos mediante el diálogo autóctono (Bonale et al., 2020). Además, (Richards et al., 2015) mencionaron que, TypeScript extiende de JavaScript con anotaciones de tipado opcionales que son poco sólidas y que el compilador de TypeScript descarta cuando emite código.	El efecto del chatbot para el aprendizaje de TypeScript es el beneficio para la resolución de dudas de programación de los programadores (Balderas et al., 2020; Bendezú y Canales, 2020)	Conocimiento (Ariste y Ramírez, 2020; Bendezú y Canales, 2020; Meza y Yurivilca, 2020).	Incremento de conocimiento (Ariste y Ramírez, 2020; Bendezú y Canales, 2020; Meza y Yurivilca, 2020).	Test	Razón
			Motivación (Ariste y Ramírez, 2020; Bendezú y Canales, 2020; Valdivia, 2021).	Incremento de la motivación (Ariste y Ramírez, 2020; Bendezú y Canales, 2020; Valdivia, 2021).	Cuestionario	Ordinal
			Satisfacción (Ariste y Ramírez, 2020; Bendezú y Canales, 2020; Huaccachi y Mejía, 2021; Larico y Reyes, 2020; Valdivia Pérez, 2021).	Incremento de la satisfacción (Ariste y Ramírez, 2020; Bendezú y Canales, 2020; Huaccachi y Mejía, 2021; Larico y Reyes, 2020; Valdivia Pérez, 2021).	Cuestionario	Ordinal
			Asertividad (Ogosi, 2021; Valdivia, 2021; Iquiño y Castillo, 2021).	Incremento de la asertividad de las respuestas (Ogosi, 2021; Valdivia, 2021; Iquiño y Castillo, 2021).	Hoja de tabulación de datos	Escala de valor
			Tiempo de aprendizaje (Daud et al., 2020; Balderas et al., 2020; Huaccachi y Mejía 2021).	Reducción del tiempo de aprendizaje (Daud et al., 2020; Balderas et al., 2020; Huaccachi y Mejía 2021).	Hoja de tabulación de datos	Escala de valor
			Eficiencia en la verificación de sintaxis (Balderas et al., 2020).	Eficiencia de la verificación de sintaxis (Balderas et al., 2020).	Hoja de tabulación de datos	Escala de valor

Anexo 2: Matriz de consistencia

Tabla 18: Matriz de consistencia

PROBLEMA GENERAL	OBJETIVO GENERAL	HIPÓTESIS GENERAL	VARIABLE	DIMENSIONES	INDICADORES
¿Cuál fue el efecto del uso del Chatbot para el aprendizaje de TypeScript?	Determinar el efecto del uso del chatbot para el aprendizaje de TypeScript	El uso del chatbot incrementó el aprendizaje de TypeScript (Balderas et al., 2020; BendeZú y Canales, 2020).			
PROBLEMAS ESPECIFICOS	OBJETIVO ESPECIFICOS	HIPÓTESIS ESPECIFICOS	Efecto del uso del Chatbot para el aprendizaje de TypeScript (Balderas et al., 2020; BendeZú y Canales, 2020)	DIMENSIONES	INDICADORES
¿Cuál fue el efecto del uso del chatbot para el aprendizaje de TypeScript en el conocimiento?	Determinar el efecto del uso del chatbot para el aprendizaje de TypeScript en el conocimiento	El uso del chatbot incrementó el conocimiento del aprendizaje de TypeScript (Ariste y Ramírez, 2020; BendeZú y Canales, 2020; Meza y Yurivilca, 2020).		Conocimiento (Ariste y Ramírez, 2020; BendeZú y Canales, 2020; Meza y Yurivilca, 2020).	Incremento de conocimiento (Ariste y Ramírez, 2020; BendeZú y Canales, 2020; Meza y Yurivilca, 2020).
¿Cuál fue el efecto del uso del chatbot para el aprendizaje de TypeScript en la motivación hacia el aprendizaje?	Determinar el efecto del uso del chatbot para el aprendizaje de TypeScript en la motivación hacia el aprendizaje	El uso del chatbot incrementó la motivación hacia el aprendizaje de TypeScript (Ariste y Ramírez, 2020; BendeZú y Canales, 2020; Valdivia, 2021).		Motivación (Ariste y Ramírez, 2020; BendeZú y Canales, 2020; Valdivia, 2021).	Incremento de la motivación (Ariste y Ramírez, 2020; BendeZú y Canales, 2020; Valdivia, 2021).
¿Cuál fue el efecto del uso del chatbot para el aprendizaje de TypeScript en la satisfacción con el aprendizaje?	Determinar el efecto del uso del chatbot para el aprendizaje de TypeScript en la satisfacción con el aprendizaje	El uso del chatbot incrementó la satisfacción con el aprendizaje de TypeScript (Ariste y Ramírez, 2020; BendeZú y Canales, 2020; Huaccachi y Mejía, 2021; Larico y Reyes, 2020; Valdivia Pérez, 2021)		Satisfacción (Ariste y Ramírez, 2020; BendeZú y Canales, 2020; Huaccachi y Mejía, 2021; Larico y Reyes, 2020; Valdivia Pérez, 2021).	Incremento de la satisfacción (Ariste y Ramírez, 2020; BendeZú y Canales, 2020; Huaccachi y Mejía, 2021; Larico y Reyes, 2020; Valdivia Pérez, 2021).
¿Cuál fue el efecto del uso del chatbot para el aprendizaje de TypeScript en la asertividad de las respuestas?	Determinar el efecto del uso del chatbot para el aprendizaje de TypeScript en la asertividad de las respuestas	El uso del chatbot incrementó la asertividad de las respuestas del aprendizaje de TypeScript (Ogosi, 2021; Valdivia, 2021; Iquiño y Castillo, 2021).		Asertividad (Ogosi, 2021; Valdivia, 2021; Iquiño y Castillo, 2021).	Incremento de la asertividad de las respuestas (Ogosi, 2021; Valdivia, 2021; Iquiño y Castillo, 2021).
¿Cuál fue el efecto del uso del chatbot para el aprendizaje de TypeScript en el tiempo de aprendizaje?	Determinar el efecto del uso del chatbot para el aprendizaje de TypeScript en el tiempo de aprendizaje	El uso del chatbot redujo el tiempo de aprendizaje de TypeScript (Daud et al., 2020; Balderas et al., 2020; Huaccachi y Mejía 2021).		Tiempo de aprendizaje (Daud et al., 2020; Balderas et al., 2020; Huaccachi y Mejía 2021).	Reducción del tiempo de aprendizaje (Daud et al., 2020; Balderas et al., 2020; Huaccachi y Mejía 2021).
¿Cuál fue el efecto del uso del chatbot para el aprendizaje de TypeScript en la eficiencia de verificación de sintaxis?	Determinar el efecto del uso del chatbot para el aprendizaje de TypeScript en la eficiencia de verificación de sintaxis			Eficiencia de la verificación de sintaxis (Balderas et al., 2020).	Eficiencia de la verificación de sintaxis (Balderas et al., 2020).

Anexo 3: Instrumento de recolección de datos

En la tabla 19 se encuentra el instrumento de recolección de datos para el sd incremento de conocimiento.

Ficha de recolección de datos	
Título de la investigación	Chatbot para el aprendizaje de TypeScript
Investigador:	Cardenas Rios Armando
Fecha de recolección de datos:	06/07/2022
Indicador:	Incremento de conocimiento

Nº	Nota Examen Antes	Nota Examen Después	Incremento de conocimiento
1			
2			
3			
4			
...			
N			

Tabla 19: Instrumento de recolección de datos para el incremento de conocimiento

En la tabla 20 se encuentra el instrumento de recolección de datos para el incremento de la motivación hacia el aprendizaje.

Ficha de recolección de datos	
Título de la investigación	Chatbot para el aprendizaje de TypeScript
Investigador:	Cardenas Rios Armando
Fecha de recolección de datos:	06/07/2022
Indicador:	Incremento de motivación hacia el aprendizaje

Nº	Nivel de motivación hacia el aprendizaje Antes	Nivel de motivación hacia el aprendizaje Después	Incremento de la motivación hacia el aprendizaje
1			
2			
3			
4			
...			
N			

Tabla 20: Instrumento de recolección de datos para el incremento de la motivación hacia el aprendizaje

En la tabla 21 se encuentra el instrumento de recolección de datos para el incremento de la satisfacción hacia el aprendizaje.

Ficha de recolección de datos	
Título de la investigación	Chatbot para el aprendizaje de TypeScript
Investigador:	Cardenas Rios Armando
Fecha de recolección de datos:	06/07/2022
Indicador:	Incremento de satisfacción hacia el aprendizaje

Nº	Nivel de satisfacción con el aprendizaje Antes	Nivel de satisfacción con el aprendizaje Después	Incremento de la satisfacción con el aprendizaje
1			
2			
3			
4			
...			
N			

Tabla 21: Instrumento de recolección de datos para el incremento de la satisfacción con el aprendizaje

En la tabla 22 se encuentra el instrumento de recolección de datos para el incremento de la asertividad de la respuesta hacia el aprendizaje.

Ficha de recolección de datos	
Título de la investigación	Chatbot para el aprendizaje de TypeScript
Investigador:	Cardenas Rios Armando
Fecha de recolección de datos:	06/07/2021
Indicador:	Incremento de la asertividad de la respuesta hacia el aprendizaje

Nº	Puntaje	% de la asertividad del chatbot	#¡DIV/0!
1		% de la asertividad con el mejor chatbot anterior	
2			
3		Reducción del tiempo de aprendizaje	#¡DIV/0!
4			
...			
N			
Promedio			
% asertividad			

Tabla 22: Instrumento de recolección de datos para el incremento de la asertividad de la respuesta

En la tabla 23 se encuentra el instrumento de recolección de datos para la reducción del tiempo de aprendizaje hacia el aprendizaje.

Ficha de recolección de datos	
Título de la investigación	Chatbot para el aprendizaje de TypeScript
Investigador:	Cardenas Rios Armando
Fecha de recolección de datos:	6/07/2021
Indicador:	Reducción del tiempo de aprendizaje

Nº	Tiempo de aprendizaje con el chatbot	Tiempo promedio de aprendizaje con el chatbot	#¡DIV/0!
1		Tiempo promedio de aprendizaje con el mejor chatbot anterior	
2			
3		Reducción del tiempo de aprendizaje	#¡DIV/0!
4			
...			
N			

Tabla 23: Instrumento de recolección de datos para la reducción del tiempo de aprendizaje

En la tabla 24 se encuentra el instrumento de recolección de datos para el incremento de la eficiencia en la verificación de sintaxis hacia el aprendizaje.

Ficha de recolección de datos	
Título de la investigación	Chatbot para el aprendizaje de TypeScript
Investigador:	Cardenas Rios Armando
Fecha de recolección de datos:	06/07/2021
Indicador:	eficiencia en la verificación de sintaxis hacia el aprendizaje

Nº	Puntaje	% verificación de sintaxis	#¡DIV/0!
1		% verificación de sintaxis con el mejor chatbot anterior	
2			
3		Reducción del tiempo de aprendizaje	#¡DIV/0!
4			
...			
N			
Promedio			
% verificación de sintaxis			

Tabla 24: Instrumento de recolección de datos para la eficiencia en la verificación de sintaxis

Anexo 4: Cuestionario de satisfacción con el aprendizaje pre-test

Tabla 25: Pre-test para la satisfacción con el aprendizaje

El encuestado deberá marcar según su criterio, la escala de medición se asigna de la siguiente manera: 1 = Nada satisfecho. 2 = poco satisfecho. 3 = Satisfacción normal. 4 = Muy satisfecho. 5 = Totalmente satisfecho.					
Pregunta.	1	2	3	4	5
¿Qué tanto satisfecho te sientes hacia el aprendizaje de TypeScript con los medios de información actuales (videos, libros, páginas web, aplicaciones, etc.)?					

Anexo 5: Cuestionario de motivación hacia el aprendizaje pre-test

Tabla 26: Pre-test para la motivación hacia el aprendizaje

El encuestado deberá marcar según su criterio, la escala de medición se asigna de la siguiente manera: 1 = Nada motivador. 2 = poco motivador. 3 = Motivación normal. 4 = Muy motivador. 5 = Totalmente motivador.					
Pregunta.	1	2	3	4	5
¿Qué tanto motivado te sientes hacia el aprendizaje de TypeScript con los medios de información actuales (videos, libro, páginas web, aplicaciones etc.)?					

Anexo 6: Test de conocimiento pre-test

1. ¿Qué es TypeScript?
 - a) Es una librería de JavaScript.
 - b) Es un lenguaje de programación.
 - c) Es un tipo de dato para JavaScript.

2. ¿Cuál es diferencia entre TypeScript y JavaScript?
 - a) JavaScript es un lenguaje de programación, mientras que TypeScript es una librería de tipos.
 - b) TypeScript es tipo de dato, mientras que JavaScript es un lenguaje de programación.
 - c) TypeScript maneja un fuerte tipado estático, mientras que JavaScript maneja tipos genéricos.

3. ¿Cuál es beneficio de TypeScript?
 - a) Controlar el flujo de una lista en JavaScript.
 - b) Controlar el tipo dinámico ya que posee un poderoso tipado estático.
 - c) Optimizar la lectura de una lista genérica en JavaScript

4. ¿Cuáles son los principales tipos de datos en TypeScript?
 - a) String
 - b) String
 - c) Integer
 - d) Number
 - e) Boolean
 - f) Double
 - g) Any

5. ¿Cuál de las siguientes definiciones pertenece al tipo de dato string?
 - a) Representa un objeto de texto como {texto1, texto2}.
 - b) Es un conjunta de caracteres y se representa como [a] [b] [c].

c) Representa valores de texto como "Hola, mundo".

6. ¿Cómo se declara un tipo de dato string?

a) Código 1

```
1 let nombre:string;  
2 nombre = "Jhoel"
```

b) Código 2

```
1 let nombre;  
2 nombre = {'Jhoel'}
```

c) Código 4

```
1 let nombre:String  
2 nombre = [Jhoel]
```

7. ¿Cuál es valor por defecto de un string?

- a) Null
- b> ""
- c) Undefined

8. ¿Qué es 'number' en TypeScript?

- a) Es una decoración para el mapeo de números enteros.
- b) Es un tipo de dato numérico.
- c) Es tipo de dato que representa número decimales.

9. ¿Con que tipo de dato se declara un número entero?

- a) int
- b) double
- c) decimal

d) number

10. ¿Qué significa la palabra 'boolean' en TypeScript?

- a) Es una anotación para utilizar en un if.
- b) Representa los valores (verdadero) o false (falso) de una lista.
- c) Es tipo de dato que representa true (verdadero) o false (falso).

11. ¿Cómo se declara un tipo de dato boolean?

a) Código 1

```
1 let isUser:[] = true;
```

b) Código 2

```
1 let isUser:boolean;
```

c) Código 3

```
1 let isUser:true;  
2 isUser = boolean;
```

12. ¿Cuál es valor por defecto de un boolean?

- a) null
- b) undefined
- c) false
- d) true

13. ¿Qué es el tipo dato any?

- a) Es un tipo dinámico parecida a JavaScript, o como solución alternativa para las limitaciones del sistema de tipado.
- b) Es una palabra reservada para el tipo de dato string.

- c) Es una anotación que representa a las variables que no tienen tipos definidos.

14. ¿Cómo se declara un any?

a) Código 1

```
1 let usuario:any;
```

b) Código 2

```
1 let usuario:@any;
```

c) Código 3

```
1 let usuario: undefined;
```

15 ¿Cómo se define el tipo a una variable?, escoge 2 opciones.

a) Código 1

```
1 let nombre: string;
```

b) Código 2

```
1 nombre = "Jhoel";
```

c) Código 3

```
1 def nombre: string;
```

d) Código 4

```
1 $ nombre = "Jhoel";
```

16. ¿Cuál es la forma correcta de declarar una función?

a) Código 1

```
1 function printToConsole(s: string)
  {
2   console.log(s);
3 }
```

b) Código 2

```
1 let printToConsole(s: string) {
2   console.log(s);
3 }
```

c) Código 3

```
1 var printToConsole(s: string) {
2   console.log(s);
3 }
```

17. ¿Cuál es la diferencia entre la función y un método?

- a) La función es una variable con argumentos, mientras que el método se declara sin argumento.
- b) La función se define fuera de la clase. El método se define dentro de las clases y es parte de ella.
- c) El método solo se usa dentro del componente, la función se puede llamar desde cualquier otro componente.

18. ¿Cómo se declara una lista?

a) Código 1

```
1 let usuarios: [];
```

b) Código 2

```
1 let usuarios: list<>;
```

c) Opción 3

```
1 let usuarios: List;
```

19. ¿Cuál de los siguientes códigos es una forma correcta de recorrer una lista?

a) Código 1

```
1 for (const s of str) {  
2   console.log(s);  
3 }
```

b) Código 2

```
1 for (const s: str) {  
2   console.log(s);  
3 }
```

c) Código 3



```
1 for (const s > str) {  
2   console.log(s);  
3 }
```

20. ¿Cuáles de las siguientes variables no es de tipo array?

a) Código 1



```
1 let libros = new Array();
```

b) Código 2



```
1 let libros: [];
```

c) Código 3



```
1 let libros: list<>;
```

d) Código 4



```
1 let libros: Usuarios[];
```

Anexo 7: Cuestionario de satisfacción con el aprendizaje post-test

Tabla 27: Post-test para la satisfacción con el aprendizaje

El encuestado deberá marcar según su criterio, la escala de medición se asigna de la siguiente manera:					
1 = Nada satisfecho.					
2 = poco satisfecho.					
3 = Satisfacción normal.					
4 = Muy satisfecho.					
5 = Totalmente satisfecho.					
Pregunta.	1	2	3	4	5
¿Qué tanto satisfecho te sientes hacia el aprendizaje de TypeScript con los medios de información actuales (videos, libros, páginas web, aplicaciones, etc.)?					

Anexo 8: Cuestionario de motivación hacia el aprendizaje pot-test

Tabla 28: Post-test para la motivación hacia el aprendizaje

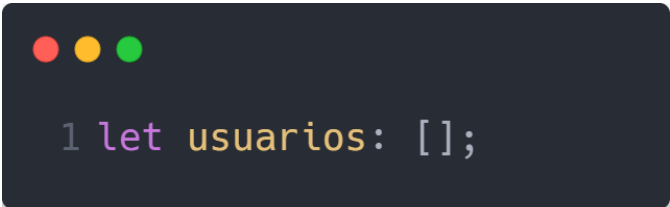
El encuestado deberá marcar según su criterio, la escala de medición se asigna de la siguiente manera:					
1 = Nada motivador.					
2 = poco motivador.					
3 = Motivación normal.					
4 = Muy motivador.					
5 = Totalmente motivador.					
Pregunta.	1	2	3	4	5
¿Qué tanto motivado te sientes hacia el aprendizaje de TypeScript con los medios de información actuales (videos, libro, páginas web, aplicaciones etc.)?					

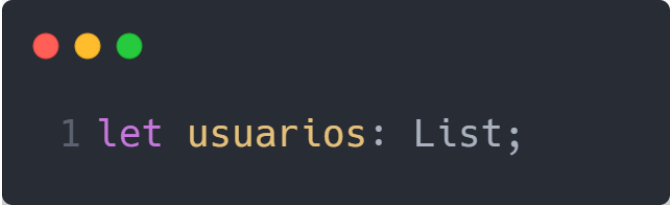
Anexo 9: Test de conocimiento post-test

1. ¿Con que tipo de dato se declara un número entero?
 - e) double
 - f) decimal
 - g) int
 - h) number

2. ¿Cuál de las siguientes definiciones pertenece al tipo de dato string?
 - a) Representa un objeto de texto como {texto1, texto2}.
 - b) Representa valores de texto como "Hola, mundo".
 - c) Es un conjunto de caracteres y se representa como [a],[b] [c].

3. ¿Qué es 'number' en TypeScript?
 - a) Es una decoración para el mapeo de números enteros.
 - b) Es tipo de dato que representa número decimales.
 - c) Es un tipo de dato numérico.

4. ¿Cómo se declara una lista?
 - a) Código 1


```
1 let usuarios: [];
```
 - b) Opción 2


```
1 let usuarios: List;
```
 - c) Código 3

```
1 let usuarios: list<>;
```

5. ¿Cuál es diferencia entre TypeScript y JavaScript?

- a) JavaScript es un lenguaje de programación, mientras que TypeScript es una librería de tipos.
- b) TypeScript maneja un fuerte tipado estático, mientras que JavaScript maneja tipos genéricos.
- c) TypeScript es tipo de dato, mientras que JavaScript es un lenguaje de programación.

6. ¿Cómo se declara un tipo de dato string?

a) Código 1

```
1 let nombre:string;  
2 nombre = "Jhoel"
```

b) Opción 2

```
1 let nombre:String  
2 nombre = [Jhoel]
```

c) Código 3

```
1 let nombre;  
2 nombre = {'Jhoel'}
```

7. ¿Cuál es la diferencia entre la función y un método?

- a) La función es una variable con argumentos, mientras que el método se declara sin argumento.
- b) El método solo se usa dentro del componente, la función se puede llamar desde cualquier otro componente.
- c) La función se define fuera de la clase. El método se define dentro de las clases y es parte de ella.

8. ¿Cuál es beneficio de TypeScript?

- a) Controlar el flujo de una lista en JavaScript.
- b) Optimizar la lectura de una lista genérica en JavaScript.
- c) Controlar el tipo dinámico ya que posee un poderoso tipado estático.

9. ¿Qué significa la palabra 'boolean' en TypeScript?

- a) Es una anotación para utilizar en un if.
- b) Es tipo de dato que representa true (verdadero) o false (falso).
- c) Representa los valores (verdadero) o false (falso) de una lista.

10. ¿Cómo se declara un tipo de dato boolean?

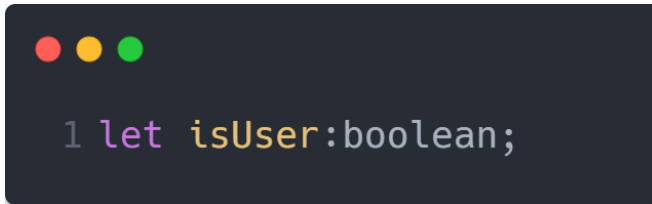
a) Código 1

```
1 let isUser:[] = true;
```

b) Opción 2

```
1 let isUser:true;  
2 isUser = boolean;
```

c) Código 3



```
1 let isUser:boolean;
```

11. ¿Qué es TypeScript?

- a) Es una librería de JavaScript.
- b) Es un tipo de dato para JavaScript.
- c) Es un lenguaje de programación.

12. ¿Cuál es valor por defecto de un boolean?

- a) null
- b) undefined
- c) true
- d) false

13. ¿Qué es el tipo dato any?

- a) Es un tipo dinámico parecida a JavaScript, o como solución alternativa para las limitaciones del sistema de tipado.
- b) Es una anotación que representa a las variables que no tienen tipos definidos.
- c) Es una palabra reservada para el tipo de dato string.

14. ¿Cuáles son los principales tipos de datos en TypeScript?

- a) String
- b) string
- c) Integer
- d) double
- e) number
- f) boolean
- g) any

15. ¿Cómo se declara un any?

a) Código 1

```
1 let usuario: undefined;
```

b) Código 2

```
1 let usuario:any;
```

c) Código 3

```
1 let usuario:@any;
```

16. ¿Cuáles de las siguientes variables no es de tipo array?

a) Código 1

```
1 let libros = new Array();
```

b) Código 2

```
1 let libros: list<>;
```

c) Código 3

```
1 let libros: [];
```

d) Código 4

```
1 let libros: Usuarios[];
```

17 ¿Cómo se define el tipo a una variable?, escoge 2 opciones.

a) Código 1

```
1 let nombre: string;
```

b) Opción 2

```
1 def nombre: string;
```

c) Código 3

```
1 nombre = "Jhoel";
```

d) Código 4

```
1 $ nombre = "Jhoel";
```

18. ¿Cuál de los siguientes códigos es una forma correcta de recorrer una lista?

a) Código 1

```
1 for (const s of str) {  
2   console.log(s);  
3 }
```

b) Opción 2

```
1 for (const s > str) {  
2   console.log(s);  
3 }
```

c) Código 3

```
1 for (const s: str) {  
2   console.log(s);  
3 }
```

19 ¿Cuál es valor por defecto de un string?

- a) null
- b) undefined
- c> ""

20. ¿Cuál es la forma correcta de declarar una función?

a) Opción 1

```
1 let printToConsole(s: string) {  
2   console.log(s);  
3 }
```

b) Código 2

```
1 function printToConsole(s: string)  
  {  
2   console.log(s);  
3 }
```

c) Código 3

```
1 var printToConsole(s: string) {
2   console.log(s);
3 }
```

21. Es una ventaja de TypeScript.

- a) Manejar los tipos de java.
- b) Es más rápido de compilar y en muchas ocasiones un poco más veloz que JavaScript.
- c) Presenta errores en el momento de la disposición mientras que JavaScript en el tiempo de ejecución.

22. ¿Qué es un if?

- a) Es una condición que solo maneja números.
- b) Es una estructura condicional que se utiliza cuando es necesario realizar una acción dependiendo si la condición se cumple o no.
- c) Es un bucle condicional que se utiliza cuando es necesario realizar una acción dependiendo si la condición se cumple o no

23. ¿Qué es una clase?

- a) Es una plantilla para crear objetos. Encapsulan datos con código para trabajar con esos datos.
- b) Es una variable que acepta todo tipo de datos
- c) Es una estructura condicional que se utiliza cuando es necesario realizar una acción.

24. ¿Cuál de los siguientes códigos no es una variable en TypeScript?

- a) Código 1

```
def usuario: string;
usuario = "Juanito"
```

b) Código 2

```
let usuario: string;
usuario = "Juanito";
```

c) Código 3

```
usuario = "Juanito";
```

25. ¿Qué es un switch?

- a) Son una plantilla para crear objetos. Encapsulan datos con código para trabajar con esos datos.
- b) Es un bloque de código donde se repite un bucle según la condición.
- c) Es una estructura de control, se utiliza para agilizar la toma de decisiones múltiples

26. ¿Cuál de los siguientes códigos representa la sintaxis de un bucle?

a) Código 1

```
if(esUsuario){
  console.log("Activo")
}
```

b) Código 2

```
while (index < 10) {  
  console.log(index);  
  index++;  
}
```

c) Código 3

```
bucle (index < 10) {  
  console.log(index);  
  index++;  
}
```

27. ¿Qué es un for?

- a) Es un decorador para implementar bucles indeterminados.
- b) Es una variable para declarar un ciclo de interacción.
- c) Es un bucle repetitivo donde podemos marcar el número de interacciones.

28. ¿Cómo se crea una clase en TypeScript?

a) Código 1

```
class Usuario {  
  id: number;  
  nombre: string;  
}
```

b) Opción 2

```
class = Usuario {  
  id: number;  
  nombre: string;  
}
```


c) Código 3

```
class Usuario([]) {  
  id: numero;  
  nombre: string;  
}
```

29. ¿Qué es un constructor?

- a) Es lista de caracteres que son inicializados dentro de una clase.
- b) Es un método especial de una clase para crear e inicializar una instancia de un objeto de esa clase.
- c) Es una función que inicializa los atributos fuera de una clase y son utilizados para crear bucles indeterminados.

30. ¿Qué es un método?

- a) Es una plantilla para crear objetos y siempre recibe parámetros del mismo tipo de la clase.
- b) Es un fragmento de código que cumple una acción dentro de una clase, también pueden tener parámetros de entrada y retornar algún valor.
- c) Son objetos especiales que interactuaran directamente con la instancia de la clase.

31. ¿Cuál de los siguientes códigos no es la sintaxis de una clase?

a) Código 1

```
class Point {}
```

b) Código 2

```
class Point {  
  x: number;  
  y: number;  
}
```

c) Código 3

```
const pt = new Point();  
pt.x = 0;  
pt.y = 0;
```

32. ¿Cómo se declara un constructor?

a) Código 1

```
constructor(nombre = "u1") {  
  this.nombre = nombre;  
}
```

b) Código 2

```
constructor();  
constructor = {"u1"}
```

c) Opción 3

```
constructor = {  
  nombre: "u1";  
}
```

33. ¿Qué es un objeto?

- a) Es una anotación que representa a las variables que no tienen tipos definidos.
- b) Es una plantilla para crear objetos y siempre recibe parámetros del mismo tipo de la clase.
- c) Son entidades que combinan estado y comportamiento.

34. ¿Cuál es la sintaxis de los parámetros opcionales de una función?

a) Código 1

```
function f(x?: number) {  
  // code  
}
```

b) Código 2

```
function f(x: number?) {  
  // code  
}
```

c) Código 3

```
function f(x: number) {  
  // code  
}
```

35. ¿Cuál es la sintaxis correcta de una for?

a) Código 1

```
for (let i) {  
  console.log(i);  
}
```

b) Código 2

```
for (let i = 0; i < 10; i++) {  
  console.log(i);  
}
```

c) Código 3

```
let for (let i; i++) {
  console.log(i);
}
```

36. ¿Cuál de las siguientes opciones no es un bucle while?

a) Código 1

```
while (index < 10) {
  console.log(index);
  index++;
}
```

b) Código 2

```
do(index < 10) {
  console.log(index);
  index++;
} while;
```

c) Código 3

```
while(i < 10) {
  i++;
}
```

37. ¿Los bucles predeterminados son for y for-each?

- a) Verdadero
- b) Falso

38. ¿Cuál es la sintaxis de un objeto?

a) Código 1

```
def persona: Persona;  
persona.name = "Jhoel";  
persona.inactivar();
```

b) Código 2

```
class persona: Persona;  
persona.name = "Jhoel";  
persona.inactivar();
```

c) Código 3

```
let persona: Persona;  
persona.name = "Jhoel";  
persona.inactivar();
```

39. ¿Los métodos se pueden declarar fuera de una clase?

- a) Verdadero
- b) Falso

40. ¿Los bucles indeterminados son while y do-while?

- a) Verdadero
- b) Falso

Anexo 10: Arquitectura del Chatbot

En la figura 1 se muestra la arquitectura del chatbot que está compuesta por back-end y front-end, divididos en 2 ambiente: 1 desarrollo y 2 como producción. En la parte de desarrollo back-end tenemos 3 capas del proyecto para darle mayor seguridad al servicio web. La capa DAO se encarga de hacer la persistencia de la base de datos Mysql, la capa LB o lógica de negocio se encarga de la lógica que permitirá la implementación del algoritmo. En la parte de desarrollo front-end tenemos las distintas tecnologías para consumir el servicio web y posteriormente mostrar al usuario a través de la web o teléfono celular. En el ambiente de producción tenemos la tecnología gitlab donde tendrá comunicación directa con Jenkins para la interacción continua del chatbot.

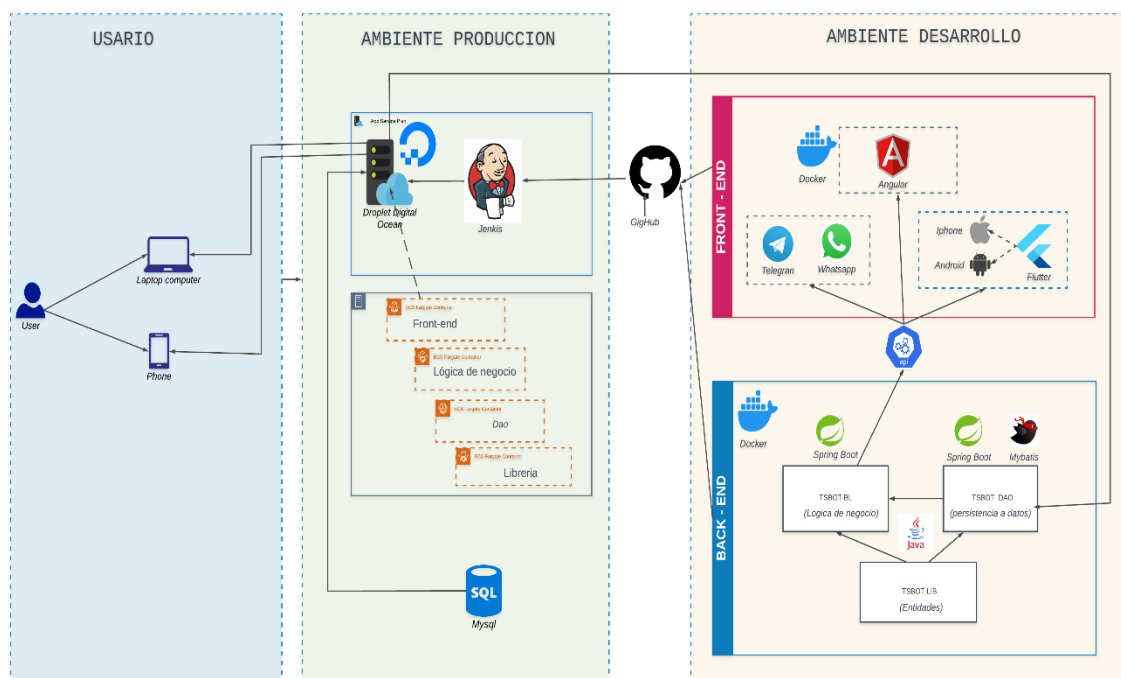
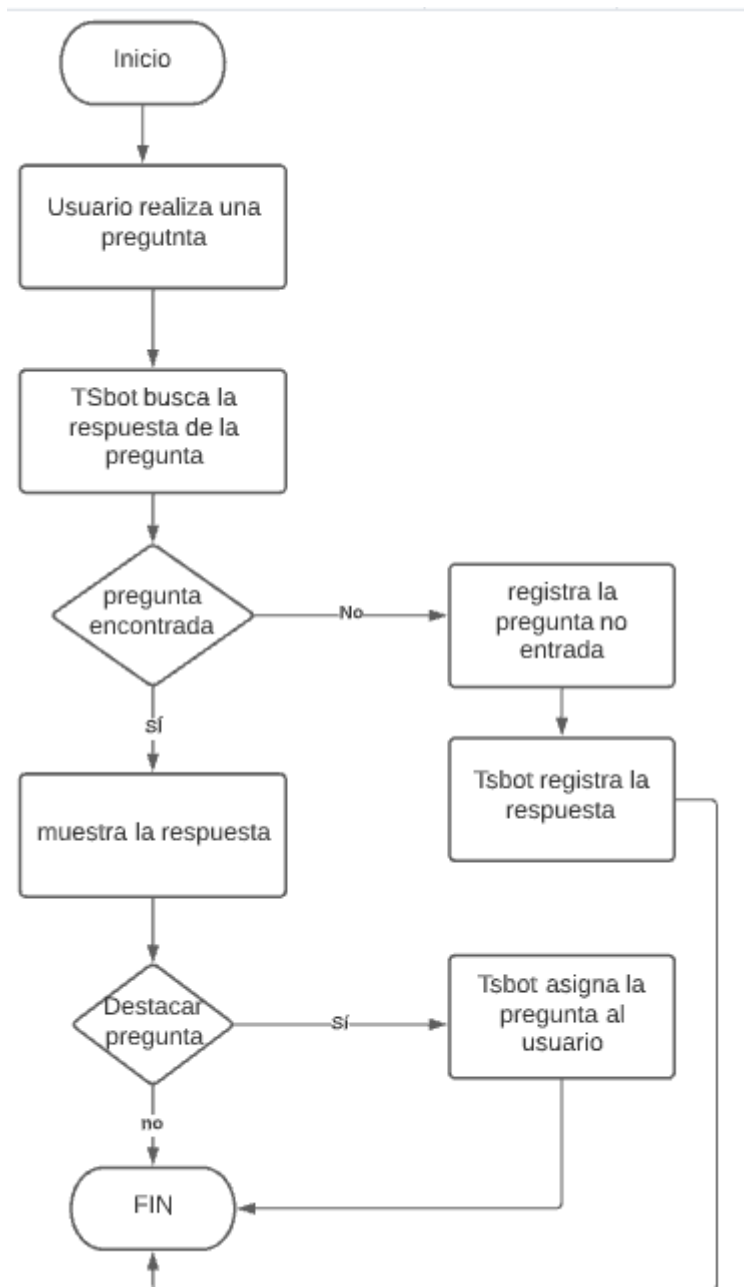


Figura 1: Arquitectura del chatbot

Anexo 11: Flujograma y pseudocódigo del algoritmo del chatbot

Figura 2: Flujograma



Anexo 12: Algoritmo principal de Fydebot

En la figura 3 se muestra el algoritmo principal del chatbot, combinado con la función tokenizer para limpiar y transformar el texto, también con la función TF-IDF para obtener la similitud más cercana.

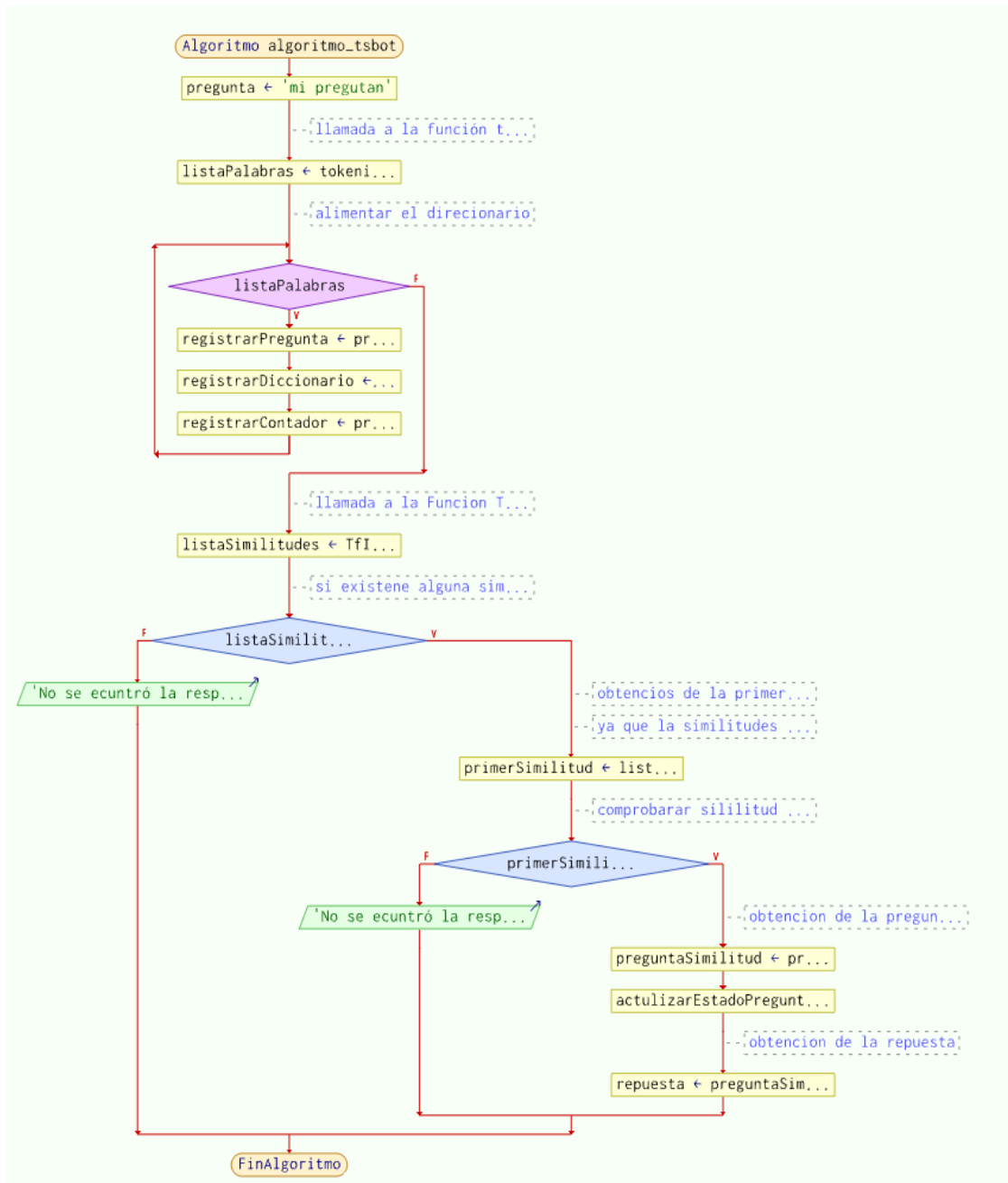


Figura 3: Algoritmo principal del chatbot

Anexo 13: Algoritmo Tokenizer

En la figura 4 se muestra la función Tokenizer, que sirve para preparar las preguntas en forma de texto.

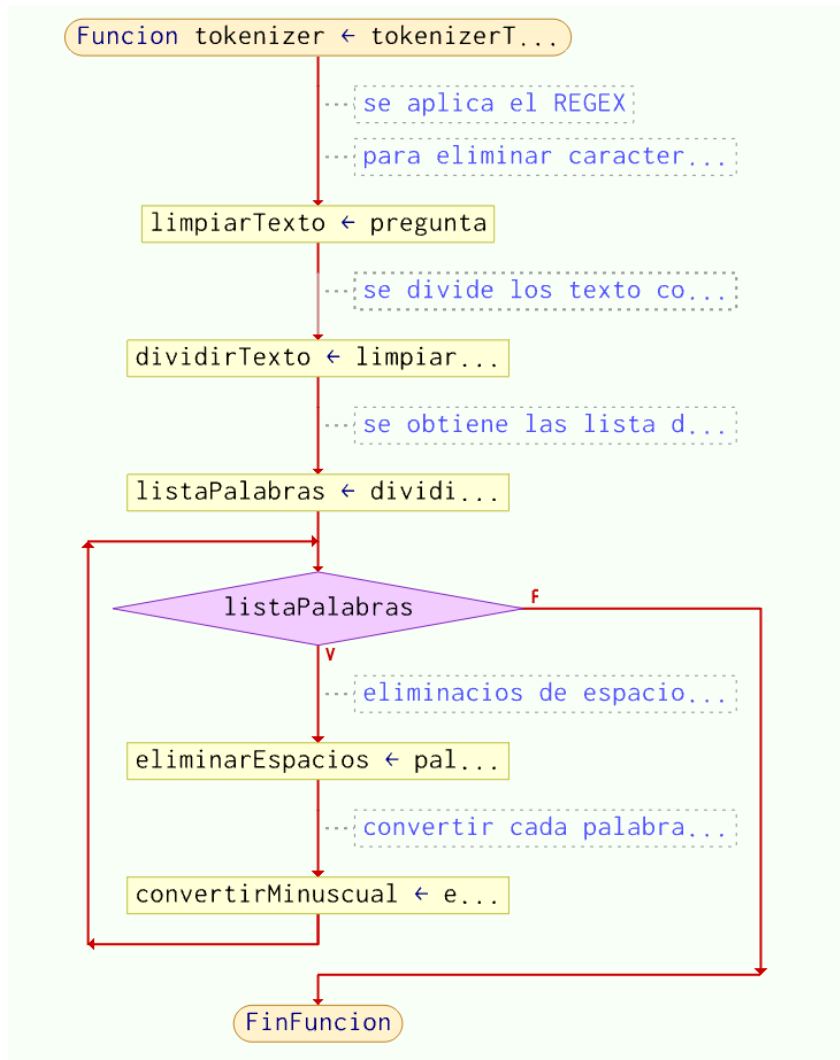


Figura 4: Algoritmo Tokenizer

Anexo 14: Algoritmo TF-IDF

En la figura 5 se muestra la función o algoritmo TF-IDF, lo cual se utiliza para encontrar la similitud más cercana a la pregunta del usuario.

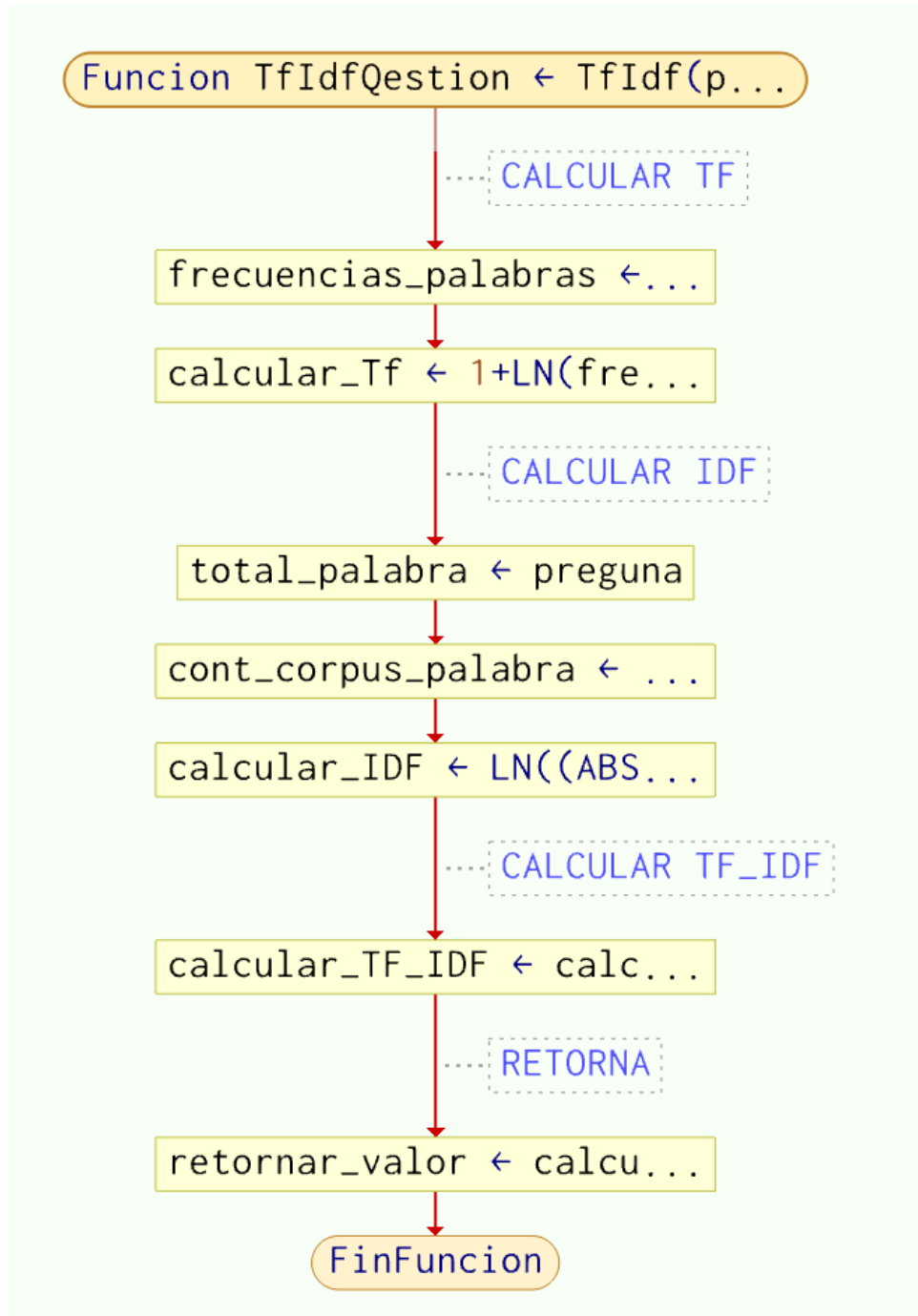


Figura 5: Algoritmo TF-ID para el cálculo de las similitudes

Anexo 15: Pseudocódigo del chatbot

El la figura 6,7 y 8 se visualiza el pseudocódigo del Chatbot la cual contiene los algoritmos bases y el algoritmo principal.

```
15 Funcion tokenizer ← tokenizerText ( pregunta )
16     //se aplica el REGEX
17     //para eliminar caracteres especiales
18     limpiarTexto ← pregunta
19     //se divide los texto con espacios en blaco
20     dividirTexto ← limpiarTexto
21     //se obtiene las lista de palabras
22     listaPalabras ← dividirTexto
23     Mientras listaPalabras Hacer
24         //eliminacios de espacios en blancos
25         eliminarEspacios ← palabra
26         // convertir cada palabra en minuscula
27         convertirMinuscual ← eliminarEspacios
28     Fin Mientras
29 Fin Funcion
30
```

Figura 6: Pseudocódigo del algoritmo Tokenizer

```
1 Funcion TfIdfQestion ← TfIdf(pregunta)
2     //CALCULAR TF
3     frecuencias_palabras ← pregunta
4     calcular_Tf ← 1+LN(frecuencias_palabras)
5     //CALCULAR IDF
6     total_palabra ← pregunta
7     cont_corpus_palabra ← pregunta
8     calcular_IDF ← LN((ABS(cantidad_palabras)/cont_corpus_palabra))
9     //CALCULAR TF_IDF
10    calcular_TF_IDF ← calcular_Tf*calcular_IDF
11    //RETORNA
12    retornar_valor ← calcular_TF_IDF
13 Fin Funcion
14
```

Figura 7: Pseudocódigo del algoritmo TF-IDF

```

31 Algoritmo algoritmo_tsbob
32     pregunta ← 'mi preguntan'
33     //llamada a la función tokenizer
34     listaPalabras ← tokenizerText(pregunta)
35     //alimentar el directorio
36     Mientras listaPalabras Hacer
37         registrarPregunta ← pregunta
38         registrarDiccionario ← palabra
39         registrarContador ← pregunta + palabra
40     Fin Mientras
41     //llamada a la Funcion TfIdf
42     listaSimilitudes ← TfIdf(pregunta)
43     //si existene alguna similitud
44     Si listaSimilitudes Entonces
45         //obtencios de la primera similitud
46         //ya que la similitudes estan oredenado de mán a menos
47         primerSimilitud ← listaSimilitudesTotal
48         //comprobarar sililitud mínima
49         Si primerSimilitud > 0.4 Entonces
50             //obtencion de la pregunta con la mayor similitud
51             preguntaSimilitud ← primerSimilitud
52             actulizarEstadoPregunta ← pregunta
53             //obtencion de la repuesta
54             repuesta ← preguntaSimilitud
55         SiNo
56             Escribir 'No se ecunró la respuesta a su pregunta'
57         Fin Si
58     SiNo
59         Escribir 'No se ecunró la respuesta a su pregunta'
60     Fin Si
61 FinAlgoritmo

```

Figura 8: Pseudocódigo Fydebot

Anexo 16: Ejecución de la metodología Scrum

Scrum team

En la siguiente tabla se visualiza la lista de roles y cargo involucrados dentro del desarrollo Fydebot.

Tabla 29: Scrum team

Equipos Scrum		
Rol	Cargo	Persona
Product Owner	Analista funcional	Cardenas Rios Armando
Scrum Master	Líder del proyecto	Cardenas Rios Armando
Desarrollador	Desarrollador	Cardenas Rios Armando

Historia de usuario

En la tabla 30 se muestra la lista de historias de usuarios ordenas prioritariamente.

Tipo de Incidencia	Clave	Resumen	Responsable	Prioridad	Estado
Historia	FID-18	Inicio sesión front-end	Armando Cardenas Rios	Medium	Finalizada
Historia	FYD-12	Inicio y configuración back-end front-end	Armando Cardenas Rios	Medium	Finalizada
Historia	FYD-13	Arquitectura	Armando Cardenas Rios	Medium	Finalizada
Historia	FYD-20	Creación de login	Armando Cardenas Rios	Medium	Finalizada
Historia	FYD-21	Registrar Usuario	Armando Cardenas Rios	Medium	Finalizada
Historia	FYD-23	Creación de algoritmo Fydebot	Armando Cardenas Rios	Medium	Finalizada
Historia	FYD-24	Creación de modulo preguntas destacadas	Armando Cardenas Rios	Medium	Finalizada
Historia	FYD-25	Creación de módulo ruta de aprendizaje	Armando Cardenas Rios	Medium	Finalizada
Historia	FYD-26	Creación del chat	Armando Cardenas Rios	Medium	Finalizada
Historia	FYD-27	Modulo preguntas y respuestas	Armando Cardenas Rios	Medium	Finalizada
Historia	FYD-28	Pruebas	Armando Cardenas Rios	Medium	Finalizada

Tabla 30: Historias de usuarios

Pila de producto

En la siguiente tabla se da a conocer la lista de tareas que contiene cada historia de usuario.

Tabla 31: Pila de producto o tareas iniciales

Tipo de Incidencia	Clave	Resumen	Responsable	Prioridad	Historias de Usuario
Tarea	FYD-14	Creación arquitectura desarrollo	Armando Cardenas Rios	Medium	Arquitectura
Tarea	FYD-15	Creación arquitectura producción	Armando Cardenas Rios	Medium	Arquitectura
Tarea	FYD-16	Creación de base de datos	Armando Cardenas Rios	Medium	Inicio y configuración back-end front-end
Tarea	FYD-17	Configuración back-end	Armando Cardenas Rios	Medium	Inicio y configuración back-end front-end
Tarea	FYD-18	Configuración front-end	Armando Cardenas Rios	Medium	Inicio y configuración back-end front-end
Tarea	FYD-22	Asignación de roles	Armando Cardenas Rios	Medium	Creación de login
Tarea	FYD-30	Creación del servicio login	Armando Cardenas Rios	Medium	Creación de login
Tarea	FYD-31	Generación de token	Armando Cardenas Rios	Medium	Creación de login
Tarea	FYD-32	Diseño de login	Armando Cardenas Rios	Medium	Creación de login
Tarea	FYD-33	Consumo del servicio login	Armando Cardenas Rios	Medium	Creación de login
Tarea	FYD-34	Gestionar del localStorage (usuario y token)	Armando Cardenas Rios	Medium	Creación de login
Tarea	FYD-35	Creación servicio registrar usuario	Armando Cardenas Rios	Medium	Registrar Usuario
Tarea	FYD-36	Diseño registrar usuario	Armando Cardenas Rios	Medium	Registrar Usuario
Tarea	FYD-37	Consumo del servicio registrar usuario	Armando Cardenas Rios	Medium	Registrar Usuario

Tipo de Incidencia	Clave	Resumen	Responsable	Prioridad	Historias de Usuario
Tarea	FYD-38	Creación Pseudocódigo algoritmo Tokenizer	Armando Cardenas Rios	Medium	Creación de algoritmo Fydebot
Tarea	FYD-39	Creación Pseudocódigo algoritmo TF-IDF	Armando Cardenas Rios	Medium	Creación de algoritmo Fydebot
Tarea	FYD-40	Creación Pseudocódigo algoritmo Fydebot	Armando Cardenas Rios	Medium	Creación de algoritmo Fydebot
Tarea	FYD-41	Implementación del algoritmo Fydebot	Armando Cardenas Rios	Medium	Creación de algoritmo Fydebot
Tarea	FYD-42	Creación de servicio de preguntas destacadas	Armando Cardenas Rios	Medium	Creación de modulo preguntas destacadas
Tarea	FYD-43	Diseño preguntas destacadas	Armando Cardenas Rios	Medium	Creación de modulo preguntas destacadas
Tarea	FYD-44	Consumo del servicio preguntas destacadas	Armando Cardenas Rios	Medium	Creación de modulo preguntas destacadas
Tarea	FYD-45	Creación del servicio ruta de aprendizaje	Armando Cardenas Rios	Medium	Creación de módulo ruta de aprendizaje
Tarea	FYD-46	Diseño ruta de aprendizaje	Armando Cardenas Rios	Medium	Creación de módulo ruta de aprendizaje
Tarea	FYD-47	Consumo de la ruta de aprendizaje	Armando Cardenas Rios	Medium	Creación de módulo ruta de aprendizaje
Tarea	FYD-48	Creación del servicio chat	Armando Cardenas Rios	Medium	Creación del chat
Tarea	FYD-49	Creación del diseño chat	Armando Cardenas Rios	Medium	Creación del chat
Tarea	FYD-50	Implementación del editor código TypeScript	Armando Cardenas Rios	Medium	Creación del chat
Tarea	FYD-51	Implementación de la verificación de sintaxis	Armando Cardenas Rios	Medium	Creación del chat
Tarea	FYD-52	Listar preguntas y respuestas	Armando Cardenas Rios	Medium	Modulo preguntas y respuestas
Tarea	FYD-53	Registrar pregunta y respuesta	Armando Cardenas Rios	Medium	Modulo preguntas y respuestas
Tarea	FYD-54	Obtener pregunta y respuesta	Armando Cardenas Rios	Medium	Modulo preguntas y respuestas

Tipo de Incidencia	Clave	Resumen	Responsable	Prioridad	Historias de Usuario
Tarea	FYD-55	Editar pregunta y respuesta	Armando Cardenas Rios	Medium	Modulo preguntas y respuestas
Tarea	FYD-56	Prueba del login	Armando Cardenas Rios	Medium	Pruebas
Tarea	FYD-57	Prueba de registrar usuario	Armando Cardenas Rios	Medium	Pruebas
Tarea	FYD-58	Prueba de ruta de aprendizaje	Armando Cardenas Rios	Medium	Pruebas
Tarea	FYD-59	Prueba del chatbot	Armando Cardenas Rios	Medium	Pruebas
Tarea	FYD-60	Prueba de la verificación de sintaxis	Armando Cardenas Rios	Medium	Pruebas
Tarea	FYD-61	Configuración del servicio	Armando Cardenas Rios	Medium	Despliegue
Tarea	FYD-62	Configuración del droplet	Armando Cardenas Rios	Medium	Despliegue
Tarea	FYD-63	Configuración del docker	Armando Cardenas Rios	Medium	Despliegue
Tarea	FYD-64	configuración del jenkins	Armando Cardenas Rios	Medium	Despliegue
Tarea	FYD-65	configuración del dominio	Armando Cardenas Rios	Medium	Despliegue
Tarea	FYD-66	configuración de nginx	Armando Cardenas Rios	Medium	Despliegue

Asignación de historia de usuario al sprint

A continuación, se presenta la lista de sprint; cada sprint tiene un tiempo de duración entre 1 y 3 semanas, dependiendo de la dificultad de las historias de usuarios.

Tabla 32: Lista de Sprints

Sprint	Duración	Tipo de Incidencia	Clave	Resumen	Responsable
Sprint 1	3 semanas	Historia	FID-18	Inicio sesión front-end	Armando Cardenas Rios

Sprint	Duración	Tipo de Incidencia	Clave	Resumen	Responsable
		Historia	FYD-12	Inicio y configuración back-end front-end	Armando Cardenas Rios
		Historia	FYD-13	Arquitectura	Armando Cardenas Rios
Sprint 2	1 semana	Historia	FYD-20	Creación de login	Armando Cardenas Rios
		Historia	FYD-21	Registrar Usuario	Armando Cardenas Rios
Sprint 3	3 semanas	Historia	FYD-23	Creación de algoritmo Fydebot	Armando Cardenas Rios
		Historia	FYD-24	Creación de modulo preguntas destacadas	Armando Cardenas Rios
Sprint 4	3 semanas	Historia	FYD-25	Creación de módulo ruta de aprendizaje	Armando Cardenas Rios
		Historia	FYD-26	Creación del chat	Armando Cardenas Rios
Sprint 5	2 semanas	Historia	FYD-27	Modulo preguntas y respuestas	Armando Cardenas Rios
		Historia	FYD-28	Pruebas	Armando Cardenas Rios
Sprint 6	1 semana	Historia	FYD-29	Despliegue	Armando Cardenas Rios

Plan de proyecto

En la siguiente imagen se visualiza la hoja de ruta de las historias de usuario que se llevó cabo. La duración del desarrollo duró 3 meses de del 04/04/2022 as 26/06/2022.

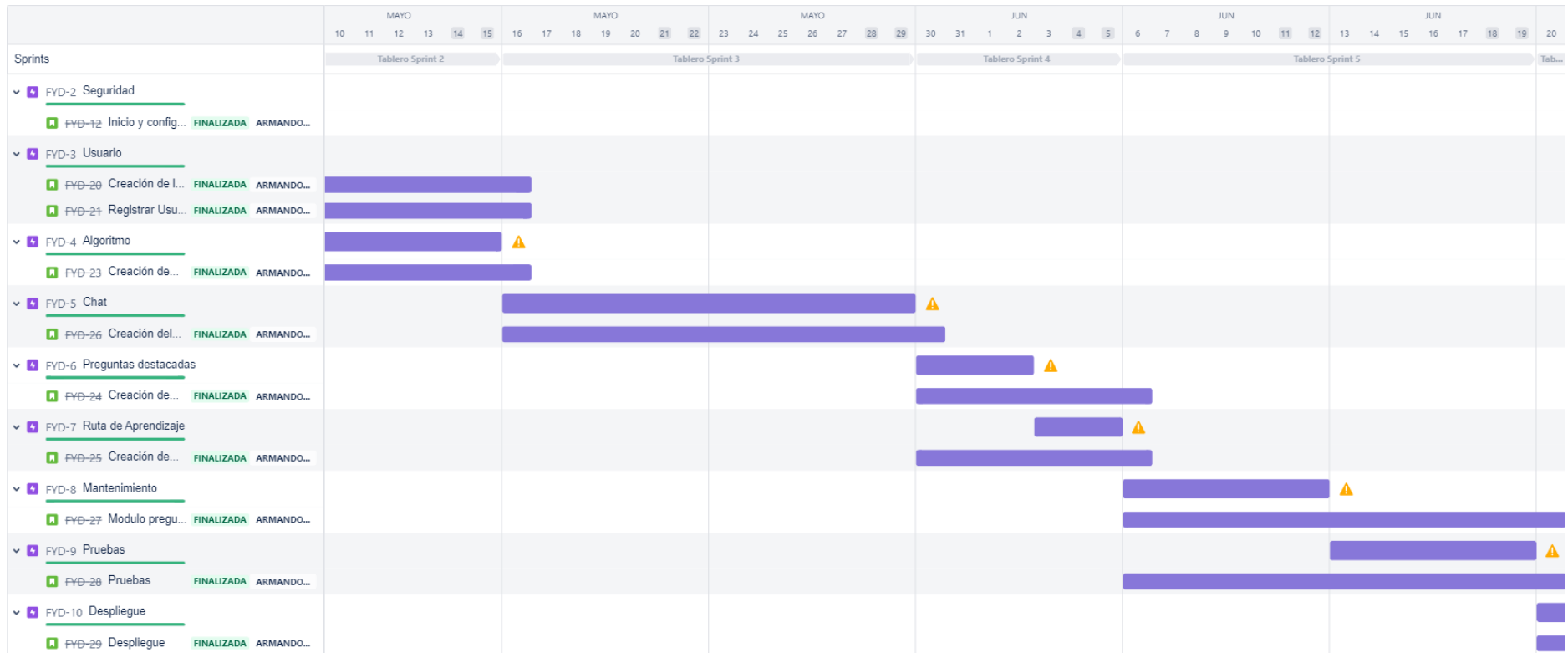
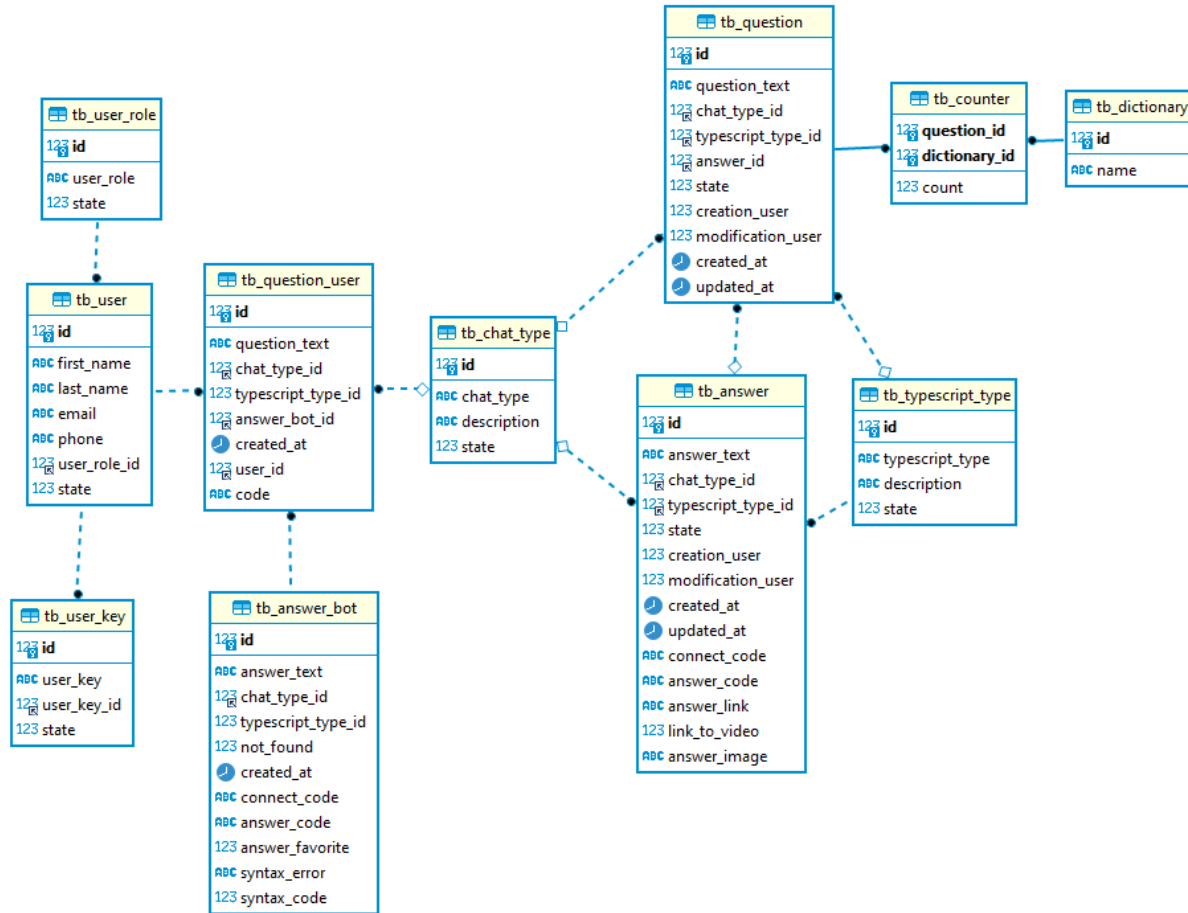


Figura 9: Plan de proyecto

Anexo 17: Modelo relacional de la base de datos

Figura 10: Modelo base de datos



Anexo 18: Diccionario de la base datos general

En la tabla 33, se define las tablas para la base de datos.

Tabla 33: Diccionario de la base de datos

Nombre tabla	Descripción
tb_answer	Respuestas de la base de conocimiento
tb_answer_bot	Respuestas del chatbot hacia el usuario
tb_chat_type	Tipo de chat (texto, imagen, video o código)
tb_counter	Conteo de palabras repetidas por preguntas para el Algoritmo.
tb_dictionary	Bolsa de palabras para el algoritmo
tb_question	Preguntas de la base de conocimiento
tb_question_user	Preguntas de los usuarios
tb_typescript_type	Tipo o temas de TypeScript
tb_user	Usuarios registrados
tb_user_key	Contraseña de los usuarios
tb_user_role	Rol del usuario

En la tabla 34 se visualiza el detalle y los campos de la tabla tb_answer, donde se encuentra las respuestas de la base de conocimiento.

Tabla 34: Descripción de los campos de la tabla tb_answer

tb_answer					
Field	Type	Null	Key	Default	Extra
id	int	NO	PRI		auto_increment
answer_text	varchar(500)	YES			
chat_type_id	int	YES	MUL		
typescript_type_id	int	NO	MUL		
state	bit(1)	NO		b'1'	
creation_user	int	NO			
modification_user	int	YES			
created_at	timestamp	NO		current_timestamp	default_generated
updated_at	timestamp	YES			
connect_code	varchar(100)	YES			
answer_code	varchar(500)	YES			

tb_answer					
Field	Type	Null	Key	Default	Extra
answer_link	varchar(200)	YES			
link_to_video	bit(1)	YES		b'0'	
answer_image	varchar(500)	YES			

En la tabla 35 se visualiza el detalle y los campos de la tabla tb_answer_bot, donde se encuentra las respuestas de chatbot.

Tabla 35: Descripción de los campos de la tabla tb_answer_bot

tb_answer_bot					
Field	Type	Null	Key	Default	Extra
id	int	NO	PRI		auto_increment
answer_text	varchar(500)	YES			
chat_type_id	int	YES	MUL		
typescript_type_id	int	YES			
not_found	bit(1)	NO		b'0'	
created_at	timestamp	NO		current_timestamp	default_generated
connect_code	varchar(100)	YES			
answer_code	varchar(500)	YES			
answer_favorite	bit(1)	YES		b'0'	
syntax_error	varchar(500)	YES			
syntax_code	bit(1)	YES		b'0'	

En la tabla 36 se visualiza el detalle y los campos de la tabla tb_chat_type, donde se encuentran el tipo de chat (texto, imagen, video o código).

Tabla 36: Descripción de los campos de la tabla tb_chat_type

tb_chat_type					
Field	Type	Null	Key	Default	Extra
id	int	NO	PRI		auto_increment
chat_type	varchar(10)	NO			
description	varchar(50)	YES			
state	bit(1)	NO		b'1'	

En la tabla 37 se visualiza el detalle y los campos de la tabla tb_counter, donde se encuentra el conteo de palabras repetidas por preguntas para el Algoritmo.

Tabla 37: Descripción de los campos de la tabla tb_counter

tb_counter					
Field	Type	Null	Key	Default	Extra
question_id	int	NO	PRI		
dictionary_id	int	NO	PRI		
count	int	NO			

En la tabla 38 se visualiza el detalle y los campos de la tabla tb_dictionary, donde se encuentra la bolsa de palabras para el algoritmo.

Tabla 38: Descripción de los campos de la tabla tb_dictionary

tb_dictionary					
Field	Type	Null	Key	Default	Extra
id	int	NO	PRI		auto_increment
name	varchar(250)	NO			

En la tabla 39 se visualiza el detalle y los campos de la tabla tb_question, donde se encuentran las preguntas de la base de conocimiento.

Tabla 39: Descripción de los campos de la tabla tb_question

tb_question					
Field	Type	Null	Key	Default	Extra
id	int	NO	PRI		auto_increment
question_text	varchar(500)	NO			
chat_type_id	int	YES	MUL		
typescript_type_id	int	YES	MUL		
answer_id	int	YES	MUL		
state	bit(1)	NO		b'1'	
creation_user	int	NO			
modification_user	int	YES			
created_at	timestamp	NO		current_timestamp	default_generated
updated_at	timestamp	YES			

En la tabla 40 se visualiza el detalle y los campos de la tabla tb_question_user, donde se encuentran las preguntas de los usuarios.

Tabla 40: Descripción de los campos de la tabla tb_question_user

tb_question_user					
Field	Type	Null	Key	Default	Extra
id	int	NO	PRI		auto_increment
question_text	varchar(500)	YES			
chat_type_id	int	YES	MUL		
typescript_type_id	int	YES			
answer_bot_id	int	NO	MUL		
created_at	timestamp	NO		current_timestamp	default_generated
user_id	int	NO	MUL		
code	varchar(500)	YES			

En la tabla 41 se visualiza el detalle y los campos de la tabla tb_typescript_type, donde se encuentran los tipos o temas de TypeScript.

Tabla 41: Descripción de los campos de la tabla tb_typescript_type

tb_typescript_type					
Field	Type	Null	Key	Default	Extra
id	int	NO	PRI		auto_increment
typescript_type	varchar(20)	NO			
description	varchar(50)	YES			
state	bit(1)	NO		b'1'	

En la tabla 42 se visualiza el detalle y los campos de la tabla tb_user, donde se encuentran los usuarios registrados de Fydebot.

Tabla 42: Descripción de los campos de la tabla tb_user

tb_user					
Field	Type	Null	Key	Default	Extra
id	int	NO	PRI		auto_increment
first_name	varchar(50)	NO			
last_name	varchar(50)	YES			
email	varchar(50)	NO			
phone	varchar(10)	YES			
user_role_id	int	NO	MUL		
state	bit(1)	NO		b'1'	

En la tabla 43 se visualiza el detalle y los campos de la tabla tb_user_key, donde se encuentran las contraseñas de los usuarios.

Tabla 43: Descripción de los campos de la tabla tb_user_key

tb_user_key					
Field	Type	Null	Key	Default	Extra
id	int	NO	PRI		auto_increment
user_key	varchar(100)	NO			
user_key_id	int	NO	MUL	1	
state	bit(1)	NO		b'1'	

En la tabla 44 se visualiza el detalle y los campos de la tabla tb_user_role, donde se encuentran los roles del usuario.

Tabla 44: Descripción de los campos de la tabla tb_user_role

tb_user_role					
Field	Type	Null	Key	Default	Extra
id	int	NO	PRI		auto_increment
user_role	varchar(50)	NO			
state	bit(1)	NO		b'1'	

Anexo 19: Capturas de pantalla del Chatbot

A continuación, se mostrará las capturas de pantalla: pantalla de inicio, login, registrarse, módulo de ruta de aprendizaje, módulo de preguntas destacadas, módulo del mantenimiento de preguntas y respuestas y el chatbot.

En la figura 11 se muestra la pantalla de inicio de Fydebot. En este apartado se visualiza los botones para iniciar sesión, registrarse, información básica del chatbot.

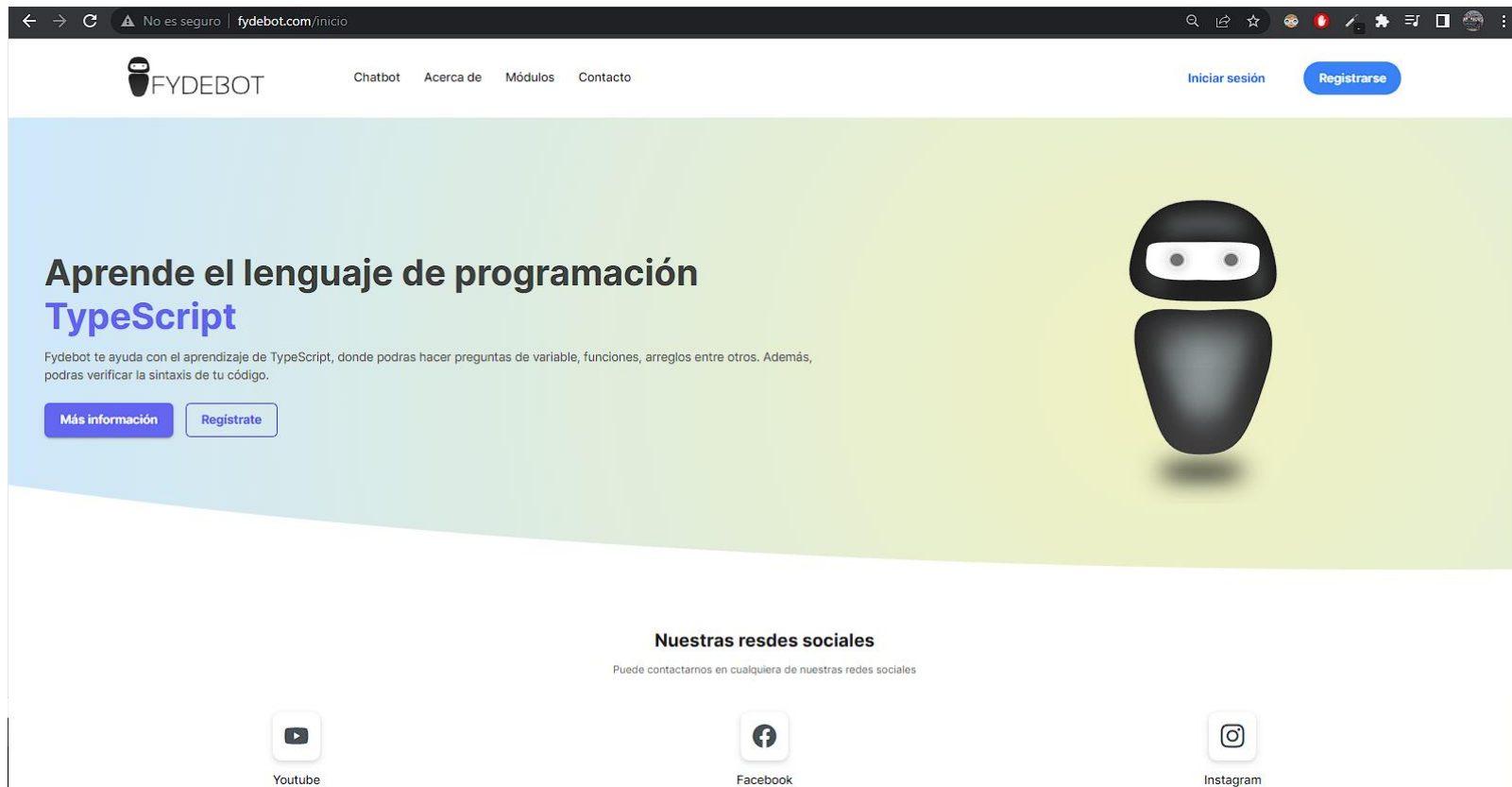


Figura 11: Pantalla de inicio

En la figura 12 se muestra la pantalla de registro de usuario, con los campos necesario y obligatorios.



FYDEBOT

CREACIÓN DE CUENTA

Regístrate para continuar

Nombres

⊗ El nombre es requerido

Apellidos

⊗ El apellido es requerido

Celular

Correo electrónico

⊗ El correo es requerido

Contraseña

⊗ La contraseña es requerida

Confirmar contraseña

⊗ Debe confirmar su contraseña

Continuar

Iniciar sesión

Figura 12: Pantalla de registro de usuario

En la figura 13 se visualiza la pantalla de login, con la opción de recordar la cuenta y el boto de registro si aún no está registrado.

FYDEBOT

Bienvenido a TSBOT!

Inicia sesión para continuar

Correo electrónico

Contraseña

Recordar [¿Olvidaste la contraseña?](#)

Continuar

Regístrate

Figura 13: Pantalla de login

En la figura 14 se observa la pantalla de tutorial cuando el usuario es nuevo en Fydebot

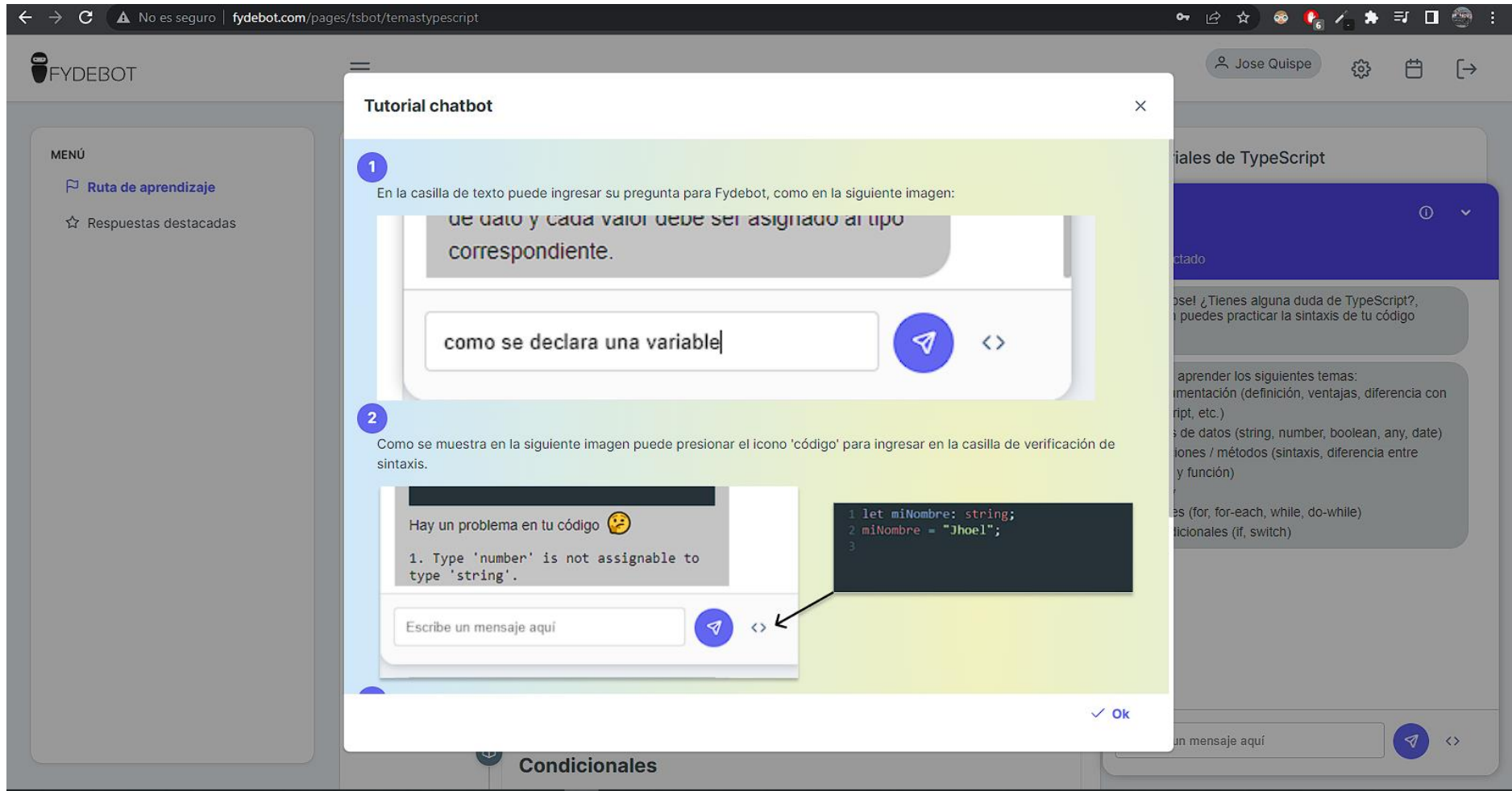


Figura 14: Pantalla de tutorial del chatbot

En la figura 15 se visualiza el módulo de **ruta de aprendizaje**, en este módulo se encuentra todos los temas que actualmente maneja el chatbot, además con videos tutoriales de TypeScript.

The screenshot shows a web browser at the URL `fydebot.com/pages/tsbot/temastypescript`. The page features a navigation menu on the left with the following items:

- MENÚ
- [Ruta de aprendizaje](#) (highlighted with a red arrow)
- Respuestas destacadas

The main content area is titled "Ruta de aprendizaje" and contains the following sections:

- Conoce TypeScript**: En esta sección podrá aprender sobre la definición, ventajas y comparación con JavaScript. Los siguientes temas lo podrá consultar al chatbot:
 - Definición
 - Ventajas o beneficios
 - Comparación con JavaScript
- Tipos de datos**: En esta sección podrá aprender los principales tipos de datos. Los siguientes temas lo podrá consultar al chatbot:
 - string
 - number
 - boolean
 - any
 - date
- Condicionales**

On the right side, there is a "Tutoriales de TypeScript" section with two video thumbnails:

- Tipo de dato number en TypeScript
- Tipo de dato Any en TypeScript

A chatbot icon is visible in the bottom right corner of the page.

Figura 15: Módulo de ruta de aprendizaje

En la figura 16 se muestra el módulo de **respuestas destacadas**, en este módulo se encuentra todas las preguntas favoritas o destacadas del usuario previamente seleccionado desde el chatbot.

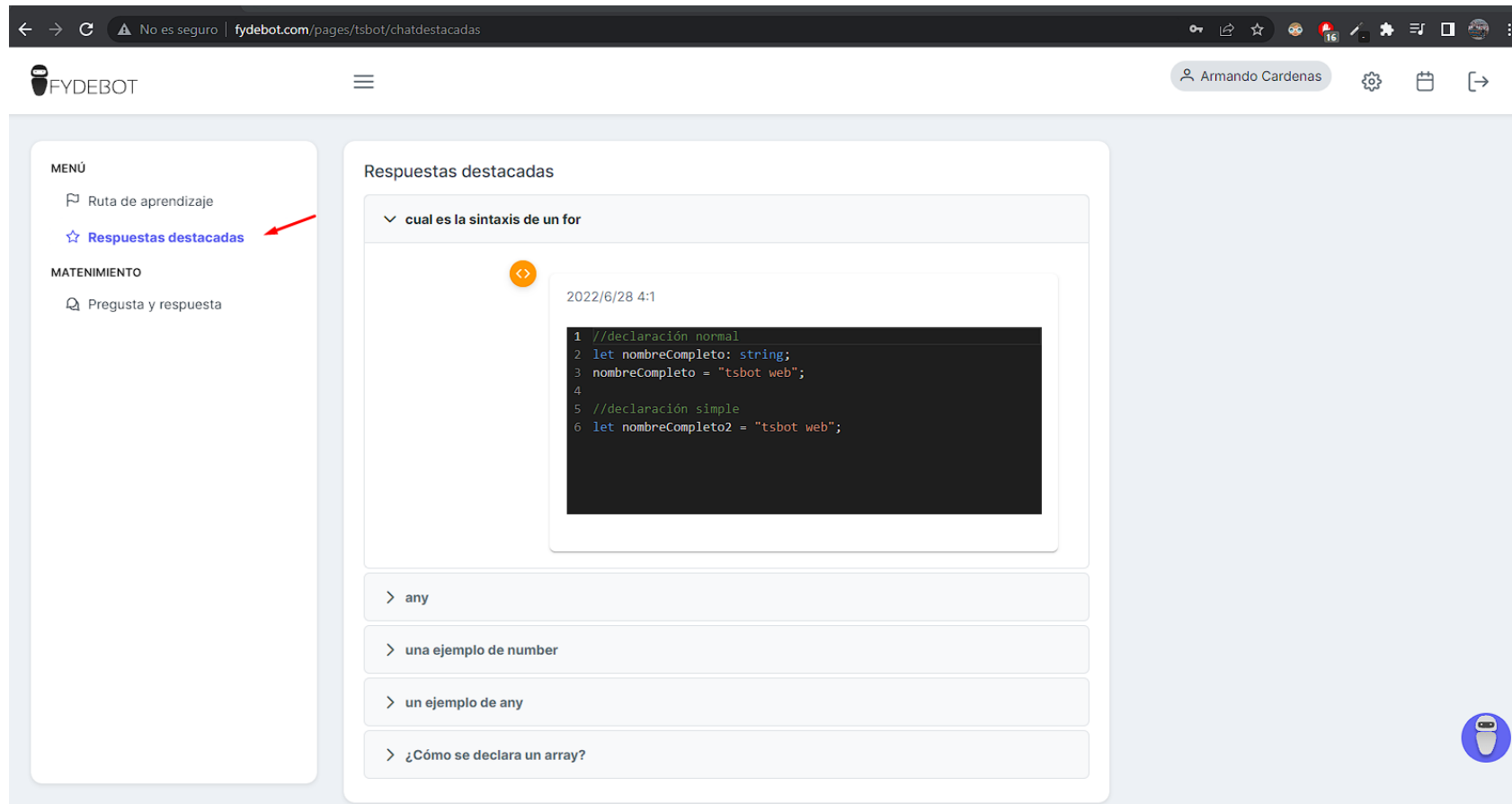


Figura 16: Modulo preguntas destacadas

En la figura 17 se muestra el módulo de **Preguntas y repuestas**, en este módulo el mantenimiento de todas las preguntas y respuestas de la base de conocimiento, además un pequeño dashboard que nos indica en total de preguntas y respuestas por estado.

The screenshot shows the 'Preguntas y respuestas' module in the FYDEBOT system. The interface includes a sidebar menu with 'Preguntas y respuesta' highlighted, a dashboard with three summary cards, and a main table listing individual questions.

Dashboard Summary:

- Preguntas activas: 575 (hoy 27 nuevas pregunta agregada)
- Preguntas por revisar: 323 (hoy 1 nueva pregunta por revisar)
- Preguntas sin respuestas: 102 (hoy 0 nueva pregunta sin respuesta)

Table of Questions:

Pregunta	Estado	Categoría	Fecha	Administrador	Respuesta
que es un función	POR REVISAR	Documentación	20/07/2022	Armando Cardenas	Respuesta
ejemplo clase	POR REVISAR	Documentación	15/07/2022	Armando Cardenas	Respuesta
declarar una clase	POR REVISAR	Documentación	15/07/2022	Armando Cardenas	Respuesta
diferencia JavaScript y TypeScript	POR REVISAR	Documentación	15/07/2022	Armando Cardenas	Respuesta
Cómo se crea una clase en TypeScript	POR REVISAR	Documentación	08/07/2022	Kevin Navarro Alfaro	Respuesta

Figura 17: Módulo preguntas y respuestas

Seguidamente la interacción directa con el chatbot:

En la figura 18 se visualiza el diseño del chat, con el icono de minimizar, el icono de información, la casilla de texto, el botón de envío y el botón código (para ingresar a la opción de la verificación de sintaxis).

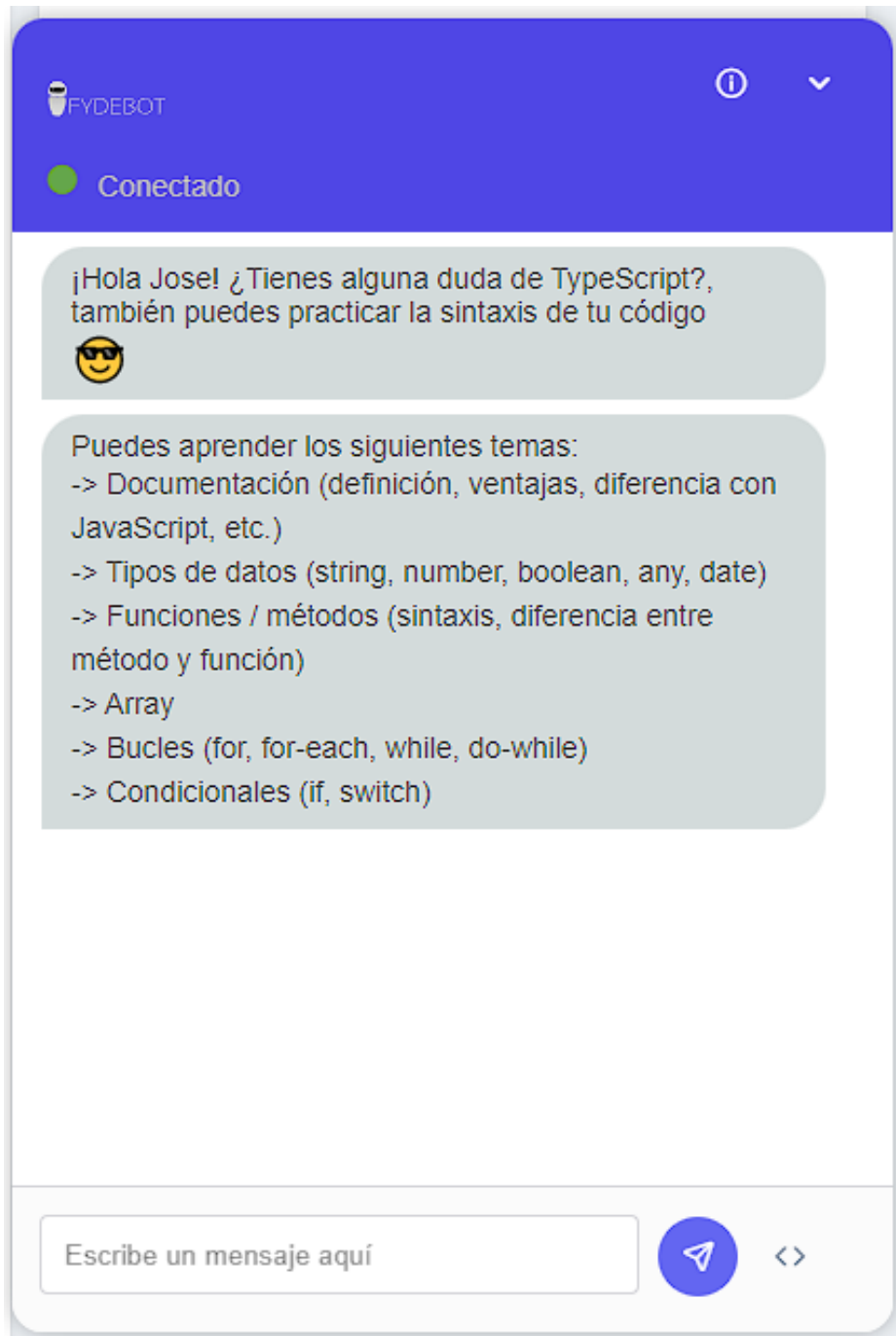


Figura 18: Diseño del chat.

En la figura 19 se muestra el mensaje de inicio, el mensaje solo se mostrará cuando el usuario es nuevo en Fydebot.

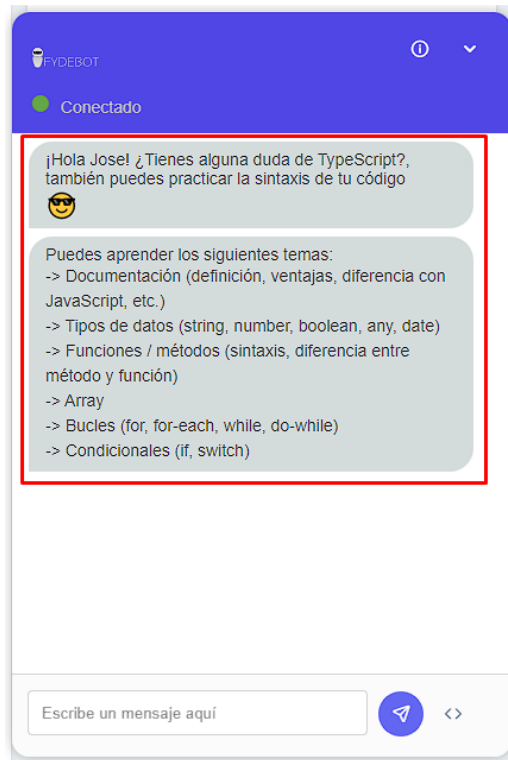


Figura 19: Mensaje de inicio

En la figura 20 se visualiza la repuesta de chatbot en texto y con link la página oficial de TypeScript.

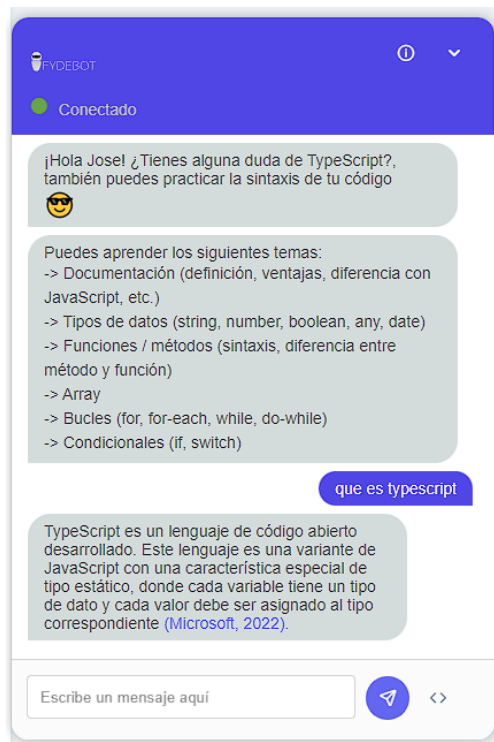


Figura 20: Respuesta del chatbot en texto

En la figura 21 se visualiza la respuesta de chatbot en código

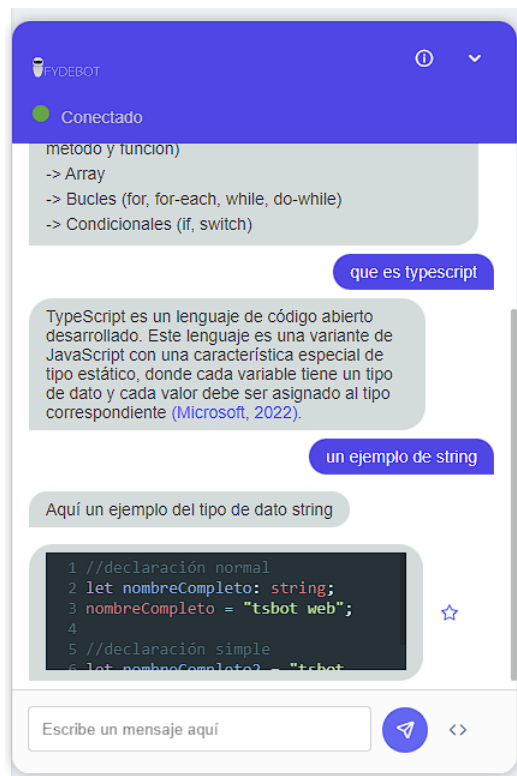


Figura 21: Respuesta del chatbot en código

En la figura 22 se observa a la respuesta no destacada del chatbot, al pasar el curso por el mensaje de respuesta aparecerá un icono de estrella para destacar la respuesta.

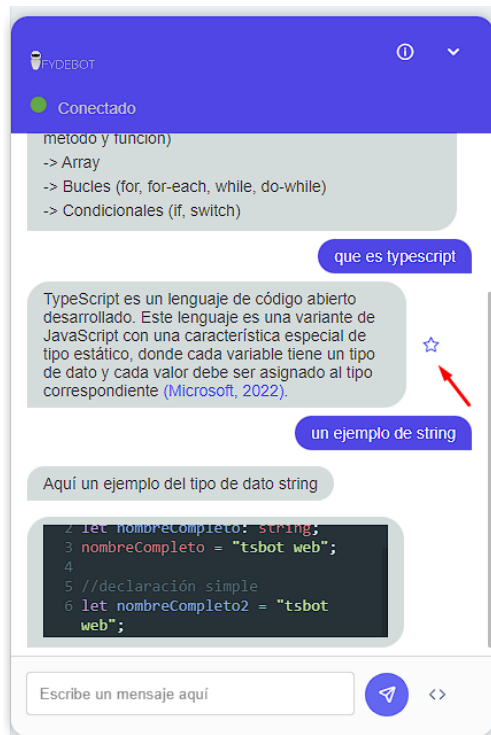


Figura 22: Respuesta no destacada

En la figura 23 se observa a la respuesta destacada del chatbot.

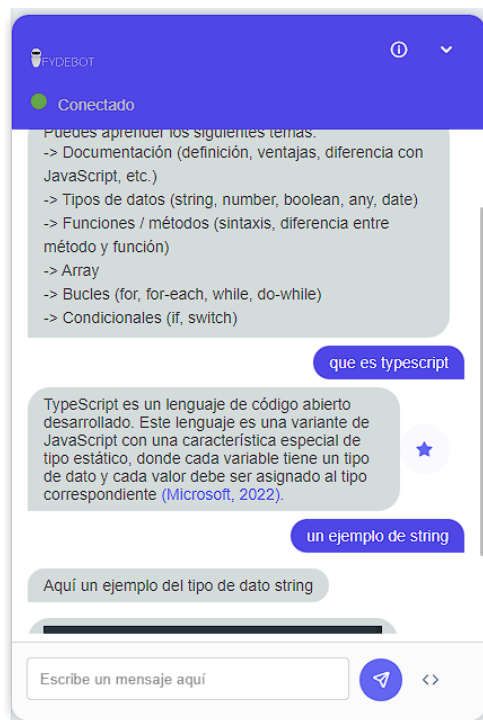


Figura 23: Respuesta destacada

En la figura 24 se visualiza el apartado de la verificación de sintaxis, puede ingresar a la opción presionando el icono de código (<>) en la parte inferior de chatbot.

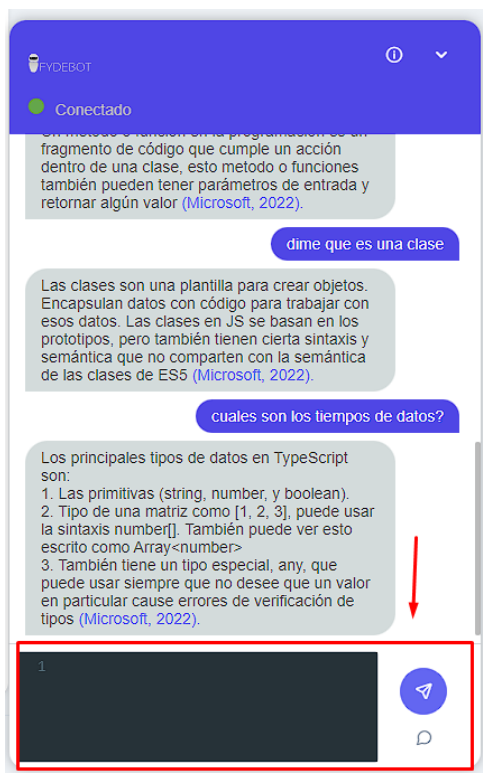


Figura 23: Apartado de la verificación de sintaxis

En la figura 24 se muestra la respuesta correcta de la sintaxis

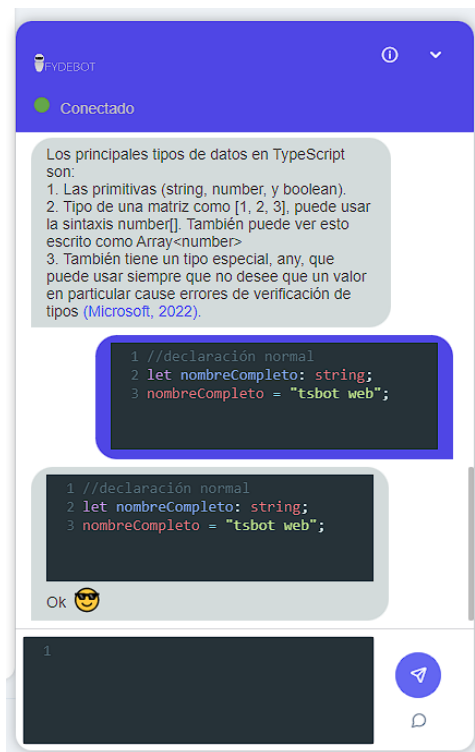


Figura 24: Respuesta correcta de la sintaxis

En la figura 25 se observa la respuesta incorrecta de la sintaxis, con los errores cometidos.

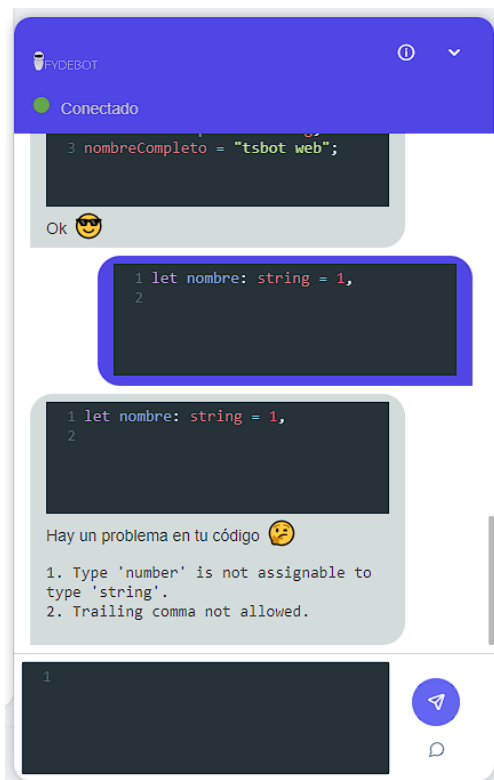


Figura 25: Respuesta incorrecta de la sintaxis