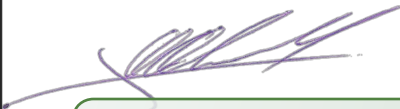


PROTOCOLO DEL MSL

Modalidad: Monografía



Héctor Xavier Limón Riaño
14/12/21



APROBADO

Por Jorge Octavio Ocharán Hernández fecha 11:02 , 15/12/2021

Ricardo Moguel Sánchez y Cesar Sergio Martínez Palacios

Proyecto Guiado | Lic. en Ingeniería de Software - FEI

Contenido

1. INTRODUCCIÓN	1
1.1. TRABAJO PRELIMINAR	1
1.2. TRABAJOS RELACIONADOS.....	1
2. PREGUNTAS DE INVESTIGACIÓN	5
3. ESTRATEGIA DE BÚSQUEDA	6
3.1 CONCEPTOS DE BÚSQUEDA.....	10
3.2 TÉRMINOS DE BÚSQUEDA.....	11
3.3 CADENAS DE BÚSQUEDA.....	12
3.4 SELECCIÓN DE FUENTES.....	15
4. SELECCIÓN DE LOS ESTUDIOS PRIMARIOS	15
4.1 CRITERIOS DE SELECCIÓN DE ESTUDIOS PRIMARIOS.....	15
4.2 PROCEDIMIENTO DE SELECCIÓN DE ESTUDIOS PRIMARIOS.....	16
5. EVALUACIÓN DE CALIDAD.....	17
6. EXTRACCIÓN DE LOS DATOS	20
7. ESTRATEGIA PARA LA SÍNTESIS DE DATOS	21
8. LIMITACIONES.	22
9. INFORME.....	22
10. GESTIÓN DE LA REVISIÓN	24
11. REFERENCIAS	1

I. Introducción

Distintos tipos de Devbots han sido generados con la intención de apoyar a un desarrollador de software y debido a esto se han logrado posicionar en el ojo de muchos desarrolladores, pero a pesar de que el término Bot se encuentra en plena culminación no existen muchos estudios y literatura sobre sus aplicaciones a diferentes campos de trabajo. Hay una escasa cantidad de trabajos de investigación y de compilación de fuentes académicas que citen el uso práctico de bots en la ingeniería de software.

El objetivo general de este mapeo sistemático de la literatura es analizar la investigación actual en el uso de bots para el desarrollo de software con el fin de contar con una descripción y organización del estado del arte y además se desea identificar diferentes DevBots que tengan algún tipo de aplicación dentro de la ingeniería de software.

En este primer punto se especifica el trabajo preliminar realizado por el equipo de investigación y además se mencionan trabajos relacionados para dar inicio al mapeo sistemático.

I.1. Trabajo Preliminar

Antes de realizar el protocolo de investigación y el protocolo del Mapeo Sistemático de Literatura (MSL) se condujo una investigación preliminar sobre la historia de bots y la clasificación de bots en donde se definen a los bots de software como un “paradigma de interfaces utilizado para conectar un usuario a un servicio de software”.(C. R. Lebeuf, 2018) Basándose en esta definición, en este trabajo cuando se refiere a los bots de desarrollo se utiliza esta definición. Adicionalmente se atendió al taller en línea *Third International Workshop on Bots in Software Engineering (BotSE)* el 4 de julio del 2021 en donde se presentaron los estudios más recientes hasta el momento en el área de bots. Por medio del trabajo preliminar, aparte de familiarizarse con los conceptos relacionados al tema de bots y sinónimos para su uso en el proceso de búsqueda.

I.2. Trabajos Relacionados

Para realizar una búsqueda exhaustiva y completa de trabajos relacionados se elaboraron cadenas de búsquedas con los conceptos importantes de bots en desarrollo y sus sinónimos. La cadena de búsqueda fue adaptada para su uso en Web Of Science y Scopus, indexadores de alta credibilidad que agrupan trabajos de investigación revisados por pares. Para identificar trabajos relacionados se introdujo la siguiente cadena de búsqueda en Scopus:

```
((("bot" OR "DevBot" OR "bots" OR "devbots" ) AND ("systematic literature review" OR "systematic review" OR "secondary study" OR "mapping study" OR "systematic mapping study" OR "literature review"))
```

Para Web Of Science se introdujo una cadena similar ajustada:

```
((("bot" OR "delbot" OR "bots" OR "devots") AND ("systematic literature review" OR "systematic review" OR "secondary study" OR "mapping study" OR "systematic mapping study" OR "literature review"))
```

Por medio de estas cadenas se identificaron varios trabajos relacionados, pero ninguno similar en términos de alcance y área de estudio. Para nuestro trabajo nos enfocaremos en las aplicaciones de bots en el desarrollo de software. De la búsqueda de trabajos, un total de once trabajos se enfocan en la detección o descripción de bots sociales o maliciosos. A continuación se muestran los 11 trabajos mencionados:

Numero:	Referencia
1	Kotkov, D., Pandey, G., & Semenov, A. (2018, December). Gaming bot detection: a systematic literature review. In <i>International Conference on Computational Social Networks</i> (pp. 247-258).
2	Stieglitz, S., Brachten, F., Ross, B., & Jung, A. K. (2017). Do social bots dream of electric sheep? A categorisation of social media bot accounts.
3	Orabi, M., Mouheb, D., Al Aghbari, Z., & Kamel, I. (2020). Detection of bots in social media: A systematic review.
4	de Morais, D. M. G., & Digiampietri, L. A. (2021, June). Methods and Challenges in Social Bots Detection: A Systematic Review.
5	Alothali, E., Zaki, N., Mohamed, E. A., & Alashwal, H. (2018, November). Detecting social bots on Twitter: A literature review.
6	Wiesenberg, M., & Tench, R. (2020). Deep strategic mediatization: Organizational leaders' knowledge and usage of social bots in an era of disinformation.
7	Collins, B., Hoang, D. T., Dang, D. T., & Hwang, D. (2020, November). Method of Detecting Bots on Social Media. A Literature Review.
8	Providel, E., & Mendoza, M. (2021). Misleading information in Spanish: a survey. <i>Social Network Analysis and Mining</i> .
9	Paul, H., & Nikolaev, A. (2021). Fake review detection on online E-commerce platforms: a systematic literature review.
10	Wang, P., Wu, L., Aslam, B., & Zou, C. C. (2009, August). A systematic study on peer-to-peer botnets.
11	Ben Sassi, I., & Ben Yahia, S. (2021). Malicious accounts detection from online social networks: a systematic review of literature.

Trabajos sobre la aplicación de bots, aunque no en el área de ingeniería de software encontrados incluyen seis estudios. Estos seis estudios investigan los chatbots en particular. De estos, dos hablan de chatbots en general, uno se enfoca en la educación, uno en reuniones empresariales y uno en el dominio de medicina. A continuación de muestran los 6 trabajos mencionados:

Numero:	Enfoque	Referencia
1	Educación	Pérez, J. Q., Daradoumis, T., & Puig, J. M. M. (2020). Rediscovering the use of chatbots in education: A systematic literature review.
2	Chatbots	Jung, H., Lee, J., & Park, C. (2020). Deriving Design Principles for Educational Chatbots from Empirical Studies on Human–Chatbot Interaction.
3	Medicina	Bhirud, N., Tataale, S., Randive, S., & Nahar, S. (2019). A Literature Review On Chatbots In Healthcare Domain.
4	Empresarial	Syvänen, S., & Valentini, C. (2020). Conversational agents in online organization–stakeholder interactions: a state-of-the-art analysis and implications for further research.
5	Chatbot	Borsci, S., Malizia, A., Schmettow, M., Van Der Velde, F., Tariverdiyeva, G., Balaji, D., & Chamberlain, A. (2021). The Chatbot Usability Scale: the Design and Pilot of a Usability Scale for Interaction with AI-Based Conversational Agents.

Aparte de estudios enfocándose a bots, existen estudios secundarios sobre la Inteligencia Artificial (IA) – en donde se mencionan bots aplicados a la ingeniería de software. Entre estos se incluyen cinco estudios sobre usos de IA fuera del área de Ingeniería de Software (IS). Finalmente, se identificaron dos estudios sobre la aplicación de bots para la automatización de procesos de negocio en empresas. A continuación, se muestran los 5 trabajos mencionados

Numero:	Referencia
1	Kaartemo, V., & Helkkula, A. (2018). A systematic review of artificial intelligence and robots in value co-creation: current status and future research avenues.
2	Ciupe, A., Meza, S., & Orza, B. (2020, September). Systematic Assessment of Interactive Instructional Technologies in Higher Engineering Education.
3	Orgeolet, L., Foulquier, N., Misery, L., Redou, P., Pers, J. O., Devauchelle-Pensec, V., & Saraux, A. (2020). Can artificial intelligence replace manual search for systematic literature? Review on cutaneous manifestations in primary Sjögren's syndrome.
4	Skinner, G., & Walmsley, T. (2019, February). Artificial intelligence and deep learning in video games a brief review.
5	Blicht, J. G. (1996). Artificial intelligence technologies for robot assisted urban search and rescue.

Revisiones o Mapeos Sistemáticos específicamente sobre bots en el desarrollo de software, de lo que investigamos, no existen. Aun así, existen investigaciones del uso de bots en otras disciplinas fuera de la IS y existen revisiones sistemáticas del uso de inteligencia artificial en el desarrollo de software. Las revisiones posteriores son importantes debido a que los bots son una especie de inteligencia artificial como menciona Lebeuf et al. (2019) “El término bot varía desde describir scripts simples que automatizan una tarea en segundo plano, hasta aplicaciones complejas que interactúan con uno o más humanos y se adaptan de manera autónoma a las actividades que realizan las personas y otros sistemas”.

Algunas revisiones sistemáticas de literatura (RSLs) con algunas similitudes se encontraron. Los trabajos con mayor semejanza son mostrados a continuación. El RSL “Systematic literature review: Teaching novices programming using robots” en donde se investigan bots utilizados para asistir a programadores novatos. (Major et al., 2012) Aunque es parecido, el enfoque de este trabajo es muy preciso y en este trabajo se busca ampliar el conocimiento en el uso de bots en diferentes tareas de desarrollo y no limitarnos a un caso en particular. Un trabajo de RSL con un área de cobertura más cercano al nuestro es el trabajo “Rediscovering the use of chatbots in education: A systematic literature review” en donde se estudia diversos usos de chatbots en el área de educación. (Perez et al., 2020) Cabe mencionar que chatbots cae dentro de los bots de desarrollo utilizados en la industria de informática. Afortunadamente, un estudio secundario con un alcance parecido al área de desarrollo “Ways of Applying Artificial Intelligence in Software Engineering” en la forma de un RSL que analiza literatura detallando usos de IA en la ingeniería de software. (Feldt et al., 2018)

A pesar de ser similar, este trabajo es diferente en que se está especificando a un solo aspecto de la ingeniería de software, el desarrollo, y una sola rama de la IA, las tecnologías bot. Por ende, de acuerdo con lo investigado, podemos concluir que no existe ningún trabajo de investigación dedicado a la recopilación de información sobre el uso práctico de bots en actividades de desarrollo de software.

Por esta razón se ve necesario realizar un estudio secundario que investigue los usos de bots en actividades del desarrollo de software. Para satisfacer esta necesidad, se realizará un mapeo sistemático de la literatura para identificar que tecnologías bot se aplican en una etapa o en general en el proceso de desarrollo de software.

2. Preguntas de investigación

En este punto se listan las preguntas de investigación a responder por medio del mapeo sistemático de la literatura junto con unas breves justificaciones.

PI-1 ¿Cuáles son las actividades de desarrollo en donde se usan bots?

Argumenta Erlendov et. al. (2019) que el futuro de DevBots está en la integración de IA a los Bot para crear asistentes artificiales en vez de herramientas de desarrollo. Para evaluar cuanto progreso a esta idealidad hay, consideramos necesario primero identificar en cuales actividades dentro del desarrollo de software los bots están siendo usados.

A, ¿Cuál es el objetivo del bot en la actividad de desarrollo usada?

Adicionalmente, se debe considerar porque se está utilizando en una etapa y como es de ayuda para un desarrollador.

B. ¿Cuáles son las actividades de desarrollo en donde más se reporta el uso de bots?

También, se piensa agrupar a los bots por el tipo de actividad en el que se utilizan para saber si existen actividades de desarrollo en donde se aprovecha más de los bots. Al responder esta pregunta se podrá identificar áreas del desarrollo en donde no se utilizan bots para trabajos futuros.

PI-2 ¿Cuáles son los beneficios reportados del bot aplicado a una actividad de desarrollo?

Aparte de saber en qué actividades se usan bots, también consideramos importante identificar beneficios reportados del uso de bots para poder categorizar los bots por beneficios posibles para guiar a los desarrolladores que desean utilizar bots.

PI-3 ¿Que problemas se presentan en las actividades del desarrollo que el *bot* contribuye a su solución?

Se sabe que nada es perfecto, y que la inclusión de alguna nueva herramienta al equipo de desarrollo incluye nuevos retos. Se requiere saber cuáles son los nuevos problemas o retos que implica el uso del bot en una etapa del desarrollo de software.

PI-4 ¿Cuál es el nivel de inteligencia del bot aplicado?

Debido a que el futuro de DevBots está en la integración de Inteligencia Artificial a los Bot para crear asistentes artificiales en vez de herramientas de desarrollo en donde bots son asistentes inteligentes se debe categorizar el nivel de inteligencia de un DevBot (Erlenhov et al., 2019). De esta forma se logra identificar el nivel de apoyo que brinda el DevBot dentro de la actividad de la Ingeniería de Software y su nivel de utilidad para la actividad especificada, las características de inteligencia se identifican con tres aspectos. Estos son Autonomía, Razonamiento y Adaptación (C. Lebeuf et al., 2018).

- Adaptación. Algunos bots son sensibles al contexto y pueden usar ese contexto para cambiar la forma en que interactúan con los usuarios.
- Razonamiento. Algunos bots siguen reglas lógicas simples; otros usan IA más avanzada para impulsar su comportamiento.
- Autonomía. Algunos bots son completamente autónomos, algunos dependen de la participación humana antes de actuar y otros utilizan un enfoque mixto.

3. Estrategia de búsqueda

En este punto se detallará toda la estrategia utilizada para la búsqueda de literatura relevante para la investigación en concreto detallando términos y conceptos encontrados, además de las cadenas de búsqueda obtenidas y sus respectivos rendimientos. La estrategia de búsqueda tendrá como objetivo lograr un nivel aceptable de completitud dentro de las limitaciones de tiempo y recursos humanos de la revisión. Se usarán dos criterios clave para evaluar la completitud de la búsqueda automatizada los cuales son el retorno (Dieste, Grimán & Juristo 2009) y la precisión (Zhang, Babar & Tell 2011).

El retorno de una búsqueda es la proporción (o porcentaje) de todos los estudios relevantes que se encuentran en la búsqueda y la precisión de una búsqueda es la proporción (o porcentaje) de los estudios encontrados que son relevantes para las preguntas de investigación que aborda una revisión. Se resume en las siguientes formulas:

Retorno = Número de estudios relevantes encontrados/ Total de estudios relevantes encontrados

Precisión= Número de estudios relevantes encontrados/ Total de estudios encontrados.

La estrategia seleccionada para el Mapeo Sistemático de la Literatura contará con 2 Fases muy importantes, la primera será aplicar la búsqueda automatizada siguiendo el Proceso Sistemático de Búsqueda propuesto por H. Zhang et al. (2009), y después vendrá una segunda fase de Snowballing basado en Wohlin, C. (2014)

Basándonos en el procedimiento de búsqueda sistemática de H. Zhang et al. (2009), mencionado anteriormente, se realizará un procedimiento de construcción de un Quasi-gold Standard se determinará realizando una búsqueda manual en un conjunto limitado de revistas de temas específicos, el conjunto de artículos relevantes encontrados se utilizará luego para evaluar la integridad de una búsqueda automatizada, las actividades a realizar durante la búsqueda de estudios primarios se pueden ver en la siguiente Figura:

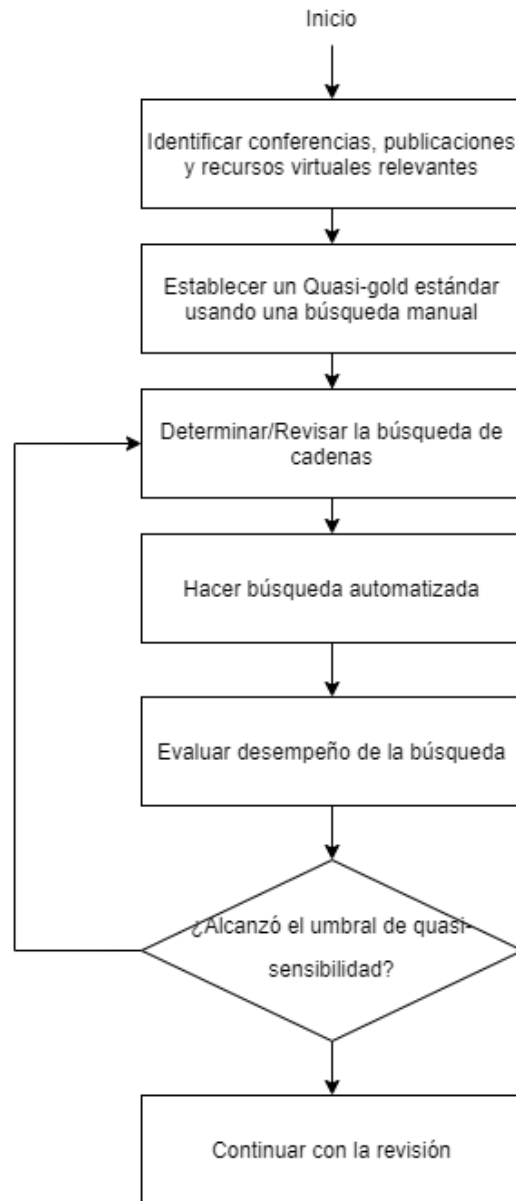


Figura 1 Proceso de obtención de Quasi-gold Standard. Adaptado de Zhang, H., Babar, M. A., & Tell, P. (2011)

Se ejecutarán 2 tipos de búsquedas, una búsqueda automática con una Cadena de Búsqueda establecida y una búsqueda manual. Conforme a la Etapa 1 del Proceso Sistemático de Búsqueda propuesto por H. Zhang mencionado, se tomarán en cuenta los motores de búsqueda IEEE Xplore, ACM Digital Library, Springer Link y Science Direct. Estos motores fueron seleccionados debido a que son reconocidos como fuentes oficiales de investigaciones revisadas por pares en el área técnica, la cual incluye la Informática y en específico la Ingeniería de Software. Aparte de estas bibliotecas digitales, se utilizará el indexador de estudios EBSCO host y la conferencia ICSE para expandir el universo de literatura de calidad a obtener.

La búsqueda se concentrará en trabajos del idioma inglés debido a que en este idioma es en el que se han realizado las investigaciones referentes a Bots y además se redujo a una búsqueda de trabajos realizados entre el año 2011 y el año 2020 para poder tener los trabajos más actualizados y modernos. Adicionalmente, se realizará una investigación aplicando Forward Snowballing en el futuro utilizando la literatura seleccionada resultante de la ejecución del MSL para incrementar el rigor de profundizaríamos demasiado las diferentes definiciones encontradas y no se busca esto.

El trabajo se realizará entre 2 personas y cada una leerá la mitad de los artículos encontrados por búsqueda automatizada y manual. Además de contar con diferentes criterios para la búsqueda de información, se realizarán sesiones informales entre los miembros, en caso de requerirlo, para la discusión sobre material encontrado sobre si es apto o no y resolver diferentes desacuerdos encontrados en el camino

En la segunda Fase, como ya se mencionó antes, se usará un proceso de Snowballing, lo primero que se hace es identificar un conjunto inicial de artículos para usar en el procedimiento de bola de nieve los cuales ya estarán definidos hasta este punto.

Primero se realizará un proceso de Snowballing hacia atrás para extraer la mayor cantidad de información posible del documento, aplicando los criterios de exclusión e inclusión, que se está examinando y no ir al nuevo documento hasta que no haya más información disponible en el documento que se está examinando mientras nos hacemos las siguientes preguntas:

Título: ¿Es provisionalmente un artículo para incluir?

Lugar de publicación: ¿Se publica en un lugar donde sea relevante?

Autores: ¿Los autores han publicado artículos relevantes en el área estudiada anteriormente?

Después a el documento se le aplicara Snowballing hacia delante, identificación de nuevos artículos basados en aquellos artículos que citan el artículo examinando. Se examinará el abstracto y en caso de que sea insuficiente se examina el lugar que cita al documento, y en un último caso que esto no brinde información suficiente se termina analizando el artículo completo. Todo esto aplicando los mismos criterios que en el Snowballing hacia atrás. En la siguiente figura se ve más a detalle el proceso de snowballing.

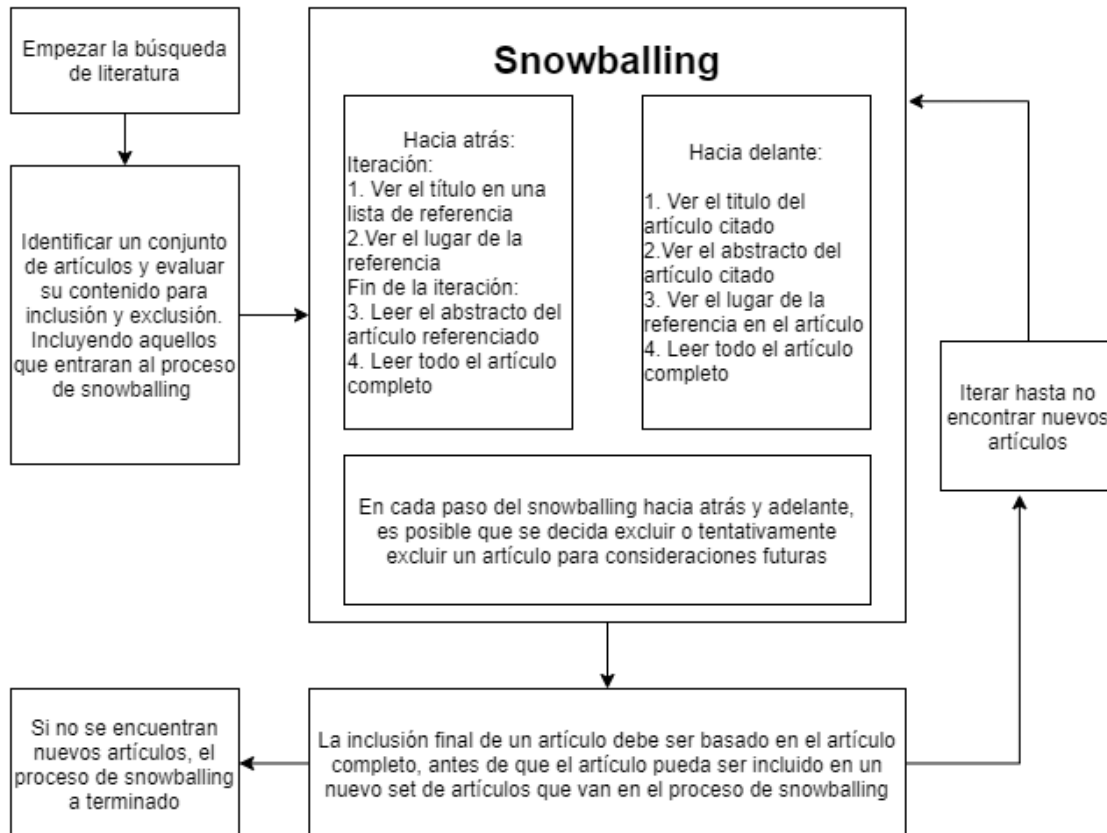


Figura 2 Proceso de snowballing. Adaptado de Wohlin, C. (2014).

3.1 Conceptos de búsqueda

Dentro de este punto se analizan los diferentes conceptos encontrados, la siguiente tabla se detallan conceptos clave del dominio de bots.

Concepto	Definición
Software Development	En español conocido como “desarrollo de software” se refiere a un conjunto de actividades de la ciencia computacional dedicadas al proceso de crear, diseñar, desplegar y mantener software de acuerdo con (IBM, 2014).
Bot	También conocidos como chatbots o software bots, en el ámbito de la ingeniería de software, se refiere a un grupo de tecnologías que conectan a un usuario con un servicio de software. (Lebeuf, 2018). Sinónimos incluyen asistentes y agentes de software
DevBot	Un Bot de desarrollo es definido como Bot que brinda una interfaz que conecta al desarrollador con servicios de software y adicionalmente aporta servicios adicionales (Lebeuf, 2018)
Automation	Aplicación de tecnologías para producir y desplegar un servicio o producto con mínima intervención humana. (Techopedia, 2021)
Software Engineering Tool	Una herramienta de software también conocidos como herramienta de programación, se refiere a un producto de software que proporciona soporte automático o semiautomático para un desarrollador. (Kuhn, 1989)
Software Development Life Cycle	El modelo “Software Development Life Cycle” (SDLC) describe de manera genérica el desarrollo de software independiente de la metodología específica al desarrollo. De acuerdo con este modelo, el desarrollo de software está dividido entre siete etapas. Estas son: Planeación, Análisis de requisitos, Diseño, Programación, Prueba, Despliegue, Operación y Mantenimiento. (Jevtic, 2019)
Software Development Methodology	La metodología de desarrollo de software especifica las etapas y actividades a realizar para un proyecto de software dependiendo de las necesidades y restricciones específicas de una empresa de desarrollo y el producto de software a elaborar. Metodologías reconocidas en la industria incluyen Cascada, Desarrollo orientado a funcionalidad, Metodología Ágil, SCRUM y Programación Extrema. (Majewski, 2019)
Development Activities	Las actividades de desarrollo son tareas realizadas por un equipo de desarrollo durante el ciclo de vida de desarrollo de software. Cada etapa incluye actividades específicas como la creación de documentación, manuales de uso, diseño arquitectural, ejecución de pruebas unitarias, etc. Las actividades de desarrollo a realizar dependen de la metodología de desarrollo seleccionada a seguir durante el desarrollo.

3.2 Términos de búsqueda

Dentro de este punto se verá los términos derivados de los conceptos de bots identificados en la Tabla de conceptos previa. A continuación, se muestra la tabla de los términos:

Terminos Relacionados	Concepto	Definición
Software Development Software Development Life Cycle	Software Development	En español conocido como “desarrollo de software” se refiere a un conjunto de actividades de la ciencia computacional dedicadas al proceso de crear, diseñar, desplegar y mantener software de acuerdo con (IBM, 2014).
Bot DevBot	Bot	También conocidos como chatbots o software bots, en el ámbito de la ingeniería de software, se refiere a un grupo de tecnologías que conectan a un usuario con un servicio de software. (Lebeuf, 2018)
DevBot Software Engineering Tool	DevBot	Un Bot de desarrollo es definido como Bot que brinda una interfaz que conecta al desarrollador con servicios de software y adicionalmente aporta servicios adicionales (Lebeuf, 2018)
Bot DevBot	Assitant	Sinónimo de bot. También conocidos como “Smart Assitant” asistente inteligente o “Smart Agent”
Automation DevBot	Automation	Aplicación de tecnologías para producir y desplegar un servicio o producto con mínima intervención humana. (Techopedia, 2021)
Software Engineering Tool Development Activities	Software Analysis	Actividad de desarrollo donde se examinan artefactos de software para evaluar la calidad en base a criterios como: robustez, seguridad, rendimiento entre otros. (Freescale Semiconductor, 2005)
Development Activities Software Development Life Cycle	Software Coding	Actividad de desarrollo en donde se transforma el diseño del sistema a código fuente en un lenguaje de programación. Se debe tomar énfasis en que codificación no es programación sino una actividad dentro de Programación. (Kingsley, 2021)
Development Activities Software Development Life Cycle	Software Onboarding	Proceso de integración de un nuevo miembro a un equipo de desarrollo para adaptarlo a las actividades, proyectos y flujos de trabajo establecidos por equipo. (Digital Adoption, 2019)

3.3 Cadenas de búsqueda

En este punto se verá detalladamente el proceso de obtención de una cadena de búsqueda, siguiendo el Proceso Sistemático de Búsqueda de H. Zhang, se ejecutando una búsqueda manual en donde se seleccionaron estudios relevantes dentro de ICSE en los motores de búsqueda IEEE Xplore y ACM Digital Library. En la siguiente tabla se resaltan los treinta artículos relevantes identificados.

Enumeración	Referencia
1	Balachandran, V. (2013, May). Reducing human effort and improving quality in peer code reviews using automatic static analysis and reviewer recommendation.
2	Kim, D., Nam, J., Song, J., & Kim, S. (2013, May). Automatic patch generation learned from human-written patches.
3	Thongtanunam, P., Kula, R. G., Cruz, A. E. C., Yoshida, N., & Iida, H. (2014, June). Improving code review effectiveness through reviewer recommendations
4	Beschastnikh, I., Lungu, M. F., & Zhuang, Y. (2017, May). Accelerating software engineering research adoption with analysis bots.
5	Guzman, E., Ibrahim, M., & Glinz, M. (2017, May). Mining twitter messages for software evolution.
6	Avellis, G., Harty, J., & Yu, Y. (2017, May). Towards mobile twin peaks for app development. In <i>2017 IEEE/ACM 4th International Conference on Mobile Software Engineering and Systems (MOBILESoft)</i> (pp. 189-193). IEEE.
7	Halder, R., Proença, J., Macedo, N., & Santos, A. (2017, May). Formal verification of ROS-based robotic applications using timed-automata.
8	Urli, S., Yu, Z., Seinturier, L., & Monperrus, M. (2018, May). How to design a program repair bot? insights from the repairator project.
9	Paikari, E., & Van Der Hoek, A. (2018, May). A framework for understanding chatbots and their future.
10	Bradley, N., Fritz, T., & Holmes, R. (2018, May). Context-aware conversational developer assistants.
11	Brown, C. (2019, May). Digital nudges for encouraging developer actions.
12	Moreno, D., Duenas, S., Cosentino, V., Fernandez, M. A., Zerouali, A., Robles, G., & Gonzalez-Barahona, J. M. (2019, May). SortingHat: wizardry on software project members.
13	Kavaler, D., Trockman, A., Vasilescu, B., & Filkov, V. (2019, May). Tool choice matters: JavaScript quality assurance tools and usage outcomes in GitHub projects.
14	van Tonder, R., & Le Goues, C. (2019, May). Towards s/engineer/bot: Principles for program repair bots.

15	Wyrich, M., & Bogner, J. (2019, May). Towards an autonomous bot for automatic source code refactoring.
16	Erlenhov, L., de Oliveira Neto, F. G., Scandariato, R., & Leitner, P. (2019, May). Current and future bots in software development.
17	Matthies, C., Dobrigkeit, F., & Hesse, G. (2019, May). An additional set of (automated) eyes: Chatbots for agile retrospectives.
18	Subramanian, V., Ramachandra, N., & Dubash, N. (2019, May). TutorBot: contextual learning guide for software engineers.
19	Kumar, R., Bansal, C., Maddila, C., Sharma, N., Martelock, S., & Bhargava, R. (2019, May). Building sankie: An ai platform for devops.
20	Sharma, V. S., Mehra, R., Kaulgud, V., & Podder, S. (2019, May). A smart advisor for software delivery-a bot framework for awareness, alerts and advice.
21	Marginean, A., Bader, J., Chandra, S., Harman, M., Jia, Y., Mao, K., ... & Scott, A. (2019, May). SapFix: Automated end-to-end repair at scale.
22	Pinheiro, A. M., Rabello, C. S., Furtado, L. B., Pinto, G., & de Souza, C. R. (2019, May). Expecting the unexpected: distilling bot development, challenges, and motivations.
23	Paikari, E., Choi, J., Kim, S., Baek, S., Kim, M., Lee, S., ... & Cheong, C. (2019, May). A chatbot for conflict detection and resolution.
24	Paikari, E., Choi, J., Kim, S., Baek, S., Kim, M., Lee, S., ... & Cheong, C. (2019, May). A chatbot for conflict detection and resolution.
25	Monperrus, M. (2019, May). Explainable software bot contributions: Case study of automated bug fixes.
26	Lin, C. T., Ma, S. P., & Huang, Y. W. (2020, June). MSABot: A chatbot framework for assisting in the development and operation of microservice-based systems.
27	Dominic, J., Houser, J., Steinmacher, I., Ritter, C., & Rodeghero, P. (2020, June). Conversational bot for newcomers onboarding to open source projects.
28	Dominic, J., Ritter, C., & Rodeghero, P. (2020, June). Onboarding Bot for Newcomers to Software Engineering.
29	Abdellatif, A., Costa, D., Badran, K., Abdalkareem, R., & Shihab, E. (2020, June). Challenges in chatbot development: A study of stack overflow posts.
30	Dey, T., Mousavi, S., Ponce, E., Fry, T., Vasilescu, B., Filippova, A., & Mockus, A. (2020, June). Detecting and characterizing bots that commit code.

De acuerdo con la Etapa 3 del Proceso Sistemático de Búsqueda de Zhang, se elaboraron cadenas de búsqueda en base a los términos de búsqueda previamente definidos. Las cadenas de búsqueda fueron ingresadas a los motores de búsqueda IEEE Xplore y ACM Digital Library para evaluar su rendimiento conforme al Estándar Quasi-Gold de Zhang. La evaluación del rendimiento se determinó en base a los criterios de Retorno y Precisión previamente definidos conforme a los treinta estudios relevantes resultantes de la búsqueda manual. El Estándar Quasi-Gold de Zhang dicta que una cadena de búsqueda es ideal para un Mapeo Sistemático de la Literatura con una precisión de 0.8 o mayor. La cadena con mejor rendimiento fue la cadena con Id-6 con un retorno de 80% y una precisión de 4.44% debido a que se encontraron veinticuatro de los treinta estudios relevantes entre 530 estudios identificados por la cadena.

En la siguiente tabla se muestran en detalle los resultados de la evaluación de las mejores cadenas de búsqueda elaboradas conforme el Estándar Quasi-Gold de Zhang ordenadas por Retorno incremental.

Id	Cadena	Estudios encontrados	Estudios relevantes encontrados	Retorno	Precisión
1	(("software development" OR "software project") AND (bot OR agent OR assistant))	262	9	0.3913	0.0344
2	("software development" AND (bot OR devbot))	107	10	0.4348	0.0935
3	"Software development" AND (*bot* OR agent OR assistant) AND (analiz* OR cod* OR automati* OR use)	9606	14	0.6087	0.0015
4	(Analysis bot OR Chatbot OR Conversational bot OR Conversational developer assistant OR Developer assistant OR Program repair bot OR Repair bot OR Review bot OR Software bot)	1602	23	0.7667	0.0144
5	((Analysis bot OR Chatbot OR Conversational bot OR Conversational developer assistant OR Developer assistant OR Repair bot OR automated repair OR Review bot OR Software bot))	2071	24	0.8	0.0116
6	(Analysis bot OR chatbot OR "conversational bot" OR "conversational developer assistant" OR "developer assistant" OR Repair bot OR automated repair OR review bot OR software bot) AND ("software development" OR "software project" OR "open source project" OR "software engineering" OR "repair")	540	24	0.80	0.04

3.4 Selección de fuentes

Para la búsqueda manual se revisarán las actas de congreso de International Conference on Software Engineering(ICSE) y sus subsidiarios incluyendo desde 2011 a 2020 relacionadas a Bots.

Para la búsqueda automatizada se implementará la cadena de búsqueda en las bibliotecas digitales ACM, IEEE Xplore, Springer Link, Science Direct y EBSCO Host.

Se seleccionaron los motores de búsqueda principalmente porque son las que tenemos acceso completo, además de que la ICSE declara en su sitio web que sus acuerdos de conferencias se encontrarán por completo en ACM e IEEE Xplore en particular, lo cual brinda una mayor confianza y estatus de que se encontrarán artículos necesarios para la investigación.

4. Selección de los estudios primarios

En este punto se detalla el procedimiento de selección de literatura relevante para la investigación junto con los criterios de inclusión y exclusión encontrados. En algunos casos, a pesar de haber sido leídos por completo, podría haber incertidumbre sobre la inclusión o no de un trabajo, si es así, seguramente habrá otra etapa en el proceso donde se traten este tipo de problemáticas y cómo se resolverán.

4.1 Criterios de selección de estudios primarios

Los siguientes criterios se usarán para determinar que estudios se incluyen o se desechan de la MSL. Ordenar por aplicación Los criterios de inclusión son los siguientes:

- CI-1 El acceso a la publicación es mediante CONRICyT aportado por la Universidad Veracruzana
- CI-2 La fecha de publicación es de 2011 a 2020.
- CI-3: La publicación debe ser artículo de investigación sobre software (Artículo de *Journal*, conferencia, o revista).
- CI-4: La publicación debe hacer referencia a al menos dos términos de búsqueda.

- CI-5 La publicación debe hacer referencia a bots en el desarrollo de software
- CI-6 La publicación debe responder al menos una pregunta de investigación.

Los criterios de exclusión son los siguientes:

- CE-1 La publicación no está en idioma inglés
- CE-2 La publicación es un estudio secundario o terciario
- CE-3 La publicación es una duplicación exacta de un estudio obtenido en otro motor de búsqueda.

4.2 Procedimiento de selección de estudios primarios

El número de participantes en la selección son dos. Estos dos son los mismos autores del MSL. De acuerdo con la metodología de Kitchenham et. al. (2015) se descarta literatura en etapas. Estas etapas son:

- 1) Eliminar literatura por idioma y año
Primero que nada, los investigadores aplicaran los filtros básicos basados en los criterios CE-1 y CI-2 los cuales hablan del idioma de la literatura deseada y el año de publicación
- 2) Eliminar literatura que no sea artículo de investigación sobre software
Basándose en el CI-3 se eliminará literatura que resulte irrelevante para la investigación
- 3) Eliminar literatura secundaria y terciaria
Después si la literatura resulta hacer referencia a el desarrollo de software, basándose en el CE-2, se analizará si la literatura no corresponde a literatura secundaria o terciaria con ayuda de los motores de búsqueda donde se tienen filtros por estudios primarios y secundarios
- 4) Eliminar estudios por título con base en términos de búsqueda
En esta etapa se excluirá de la literatura aquellos que no incluyan por lo menos dos términos de búsqueda haciendo referencia al criterio CI-4.
- 5) Eliminar estudios por título que no hablan de bots en el desarrollo de software
En esta etapa se excluirá de la literatura aquellos que no sean relevantes para la investigación hablando de temas ajenos a los bots en el desarrollo de software con base a su propio título.
- 6) Eliminar estudios por abstract que no hablan de bots en el desarrollo de software.
En esta etapa se excluirá de la literatura aquellos que no sean relevantes para la investigación hablando de temas ajenos a los bots en el desarrollo de software, con base a la lectura completa de su abstracto.
- 7) Examinar estudios relevantes por medio de revisión crítica
Ahora se leerán completamente desde introducción hasta conclusión la literatura restante para decidir si son de suficiente calidad con respecto a sus habilidades de responder una o varias de las preguntas de investigación con base al criterio CI-5. además de que en caso de dudas los investigadores tendrán reuniones de discusión para aclarar y examinar el artículo. En caso de no llegar a un consenso se consultará con los directores para un veredicto
- 8) Evaluar la calidad de los estudios

Finalmente, el estudio pasa por un proceso de evaluación de calidad por medio de un proceso específico detallado en el capítulo 5 del documento y tomando en cuenta el criterio CI-5, los investigadores deberán consultar con los directores en casos de dudas o inconvenientes. Además de que estos mismos seleccionaran aleatoriamente estudios y evaluará el análisis realizado por los investigadores.

5. Evaluación de Calidad

Aquí se mostrarán el proceso para evaluar la calidad de la literatura obtenida, esto para descubrir el nivel de importancia que tienen la literatura. Para la evaluación de calidad de estudios primarios se seguirán los pasos del método de Kitchenham et. al. (2015). Las etapas para esta fase del trabajo incluyen:

- Seleccionar lista de cotejo
- Asignar calificaciones de calidad
- Validar calificaciones
- Evaluar el impacto de las calificaciones

Se realiza este proceso de evaluación de calidad para asignar calificaciones a los trabajos primarios seleccionados para medir la importancia de los resultados de extracción de datos de los estudios individuales. La checklist a usar en este proceso fue propuesta por Dybå, T. y Dingsøy, T. (2008a) con su lista de verificación de calidad que se puede utilizar en varios tipos de estudios.

Adicionalmente se le agrego un sistema de puntos para calificar la calidad en base a la checklist el cual se basará en una puntuación de 10 puntos divididos entre las 11 preguntas:

1. ¿El documento se basa en una investigación (o es simplemente un informe de "lecciones aprendidas" basado en la opinión de expertos)? (1 pt.)
2. ¿Existe una declaración clara de los objetivos de la investigación? (1 pt.)
3. ¿Existe una descripción adecuada del contexto en el que se realizó la investigación? (1 pt.)
4. ¿El diseño de la investigación fue apropiado para abordar los objetivos de la investigación? (1 pt.)
5. ¿La estrategia de contratación fue adecuada a los objetivos de la investigación? (0.5 pt.)
6. ¿Había un grupo de control con el que comparar los tratamientos? (0.5 pt.)
7. ¿Se recopilaron los datos de una manera que abordará el tema de la investigación? (1 pt.)
8. ¿Fue el análisis de datos suficientemente riguroso? (1 pt.)
9. ¿Se ha considerado adecuadamente la relación entre el investigador y los participantes? (0.5 pt.)
10. ¿Existe una declaración clara de los hallazgos? (1 pt.)
11. ¿Tiene valor el estudio para la investigación o la práctica? (1.5 pt.)

Conociendo las puntuaciones podemos hacer la siguiente clasificación de puntaje de checklist:

Nivel de calidad	Rango
Alto	10-8.5 pts.
Medio	8-6 pts.
Bajo	5-1 pts.

Que los documentos cumplan con la calidad del trabajo es parte importante para el desarrollo y para verificar que la checklist es funcional se hizo una pequeña prueba con 5 estudios relevantes, obtenidos anteriormente, para verificar su calidad y la funcionalidad de la checklist los trabajos son los siguientes:

- Trabajo 1: Balachandran, V. (2013, May). Reducing human effort and improving quality in peer code reviews using automatic static analysis and reviewer recommendation
- Trabajo 2: Erlenhov, L., de Oliveira Neto, F. G., Scandariato, R., & Leitner, P. (2019, May). Current and future bots in software development.
- Trabajo 3: Lin, C. T., Ma, S. P., & Huang, Y. W. (2020, June). MSABot: A chatbot framework for assisting in the development and operation of microservice-based systems.
- Trabajo 4: Paikari, E., & Van Der Hoek, A. (2018, May). A framework for understanding chatbots and their future.
- Trabajo 5: Wyrich, M., & Bogner, J. (2019, May). Towards an autonomous bot for automatic source code refactoring.

Pregunta	Trabajo 1	Trabajo 2	Trabajo 3	Trabajo 4	Trabajo 5
¿El documento se basa en una investigación (o es simplemente un informe de "lecciones aprendidas" basado en la opinión de expertos)? (1 pt.)	Se basa en una investigación	Se basa en una investigación	Se basa en una investigación	Se basa en una investigación	Se basa en una investigación
¿Existe una declaración clara de los objetivos de la investigación? (1 pt.)	Sí	Sí	Sí	Sí	Sí
¿Existe una descripción adecuada del contexto en el que se realizó la investigación? (1 pt.)	Sí	Sí	Sí	Sí	Sí

¿El diseño de la investigación fue apropiado para abordar los objetivos de la investigación? (1 pt.)	Sí	Sí	Sí	Sí	Sí
¿La estrategia fue adecuada a los objetivos de la investigación? (0.5 pt.)	Sí	Sí	Sí	No	Sí
¿Había un grupo de control con el que comparar los tratamientos? (0.5 pt.)	Sí	No	Sí	No	No
¿Se recopilaron los datos de una manera que abordará el tema de la investigación? (1 pt.)	Sí	Sí	Sí	Sí	Sí
¿Fue el análisis de datos suficientemente riguroso? (1 pt.)	Sí	No	Sí	Sí	Sí
¿Se ha considerado adecuadamente la relación entre el investigador y los participantes? (0.5 pt.)	Sí	No	Sí	No	Sí
¿Existe una declaración clara de los hallazgos? (1 pt.)	Si	Si	No	Si	Sí
¿Tiene valor el estudio para la investigación o la práctica? (1.5 pt.)	Sí	Sí	Sí	No	Sí
Total	10 pts.	8 pts.	9 pts.	6.5 pts.	9.5 pts.

6. Extracción de los datos

En este punto se especificarán los datos que se van a obtener la literatura obtenida anteriormente, además de mostrar el proceso en caso de desacuerdos en el equipo de investigación

Siguiendo los pasos establecidos por el método a seguir de Kitchenham et. al. (2015)” Siempre que sea posible, la extracción de datos debe ser realizada de forma independiente por dos o más investigadores. Los datos de los investigadores deben compararse y los desacuerdos se deben resolver por consenso entre los investigadores o mediante el arbitraje de un investigador independiente adicional.”

Para la extracción de datos se realizará un proceso independiente con un equipo conformado por:

Director: Dr. Jorge Octavio Ocharán Hernández

Co director: Dr. Xavier Limón Riaño

Investigadores: César Sergio Martínez Palacios y Ricardo Moguel Sánchez

Después de completar su proceso de extracción se comparará y contrastará resultados en una discusión posterior. Desacuerdos no resueltos serán debatidos hasta llegar a un consenso. En el caso de no llegar a un consenso se consultará a los directores para resolver el desacuerdo. Al realizar la discusión se disminuyera los riesgos asociados con la calidad de la literatura seleccionada. Estos riesgos incluyen el sesgo de confirmación que al tener dos investigadores es reducido, Adicionalmente se reducen los riesgos de baja precisión y validez al extraer datos.

Para extraer los datos de la búsqueda automatizada se usarán hojas de cálculo de Microsoft Excel en formato CSV descargadas de los motores de búsquedas.

En la siguiente tabla se muestra el formato en el que se extraerá la información necesaria de cada estudio primario durante el procedimiento de búsqueda:

DATOS DE LA PUBLICACIÓN	
Título	
Autores	
Año	
Publicador	
Tipo de publicación (memorias, <i>journal</i> , etc.)	[memorias de congreso, journal científico, workshop]
DOI	
Palabras clave	
<i>Abstract</i> o resumen	
Objetivos	
Pregunta/s de investigación relacionada/s	
Base de datos fuente	
Tema	
Tecnologías	
Instrumentos de investigacion	

DATOS PARA LA EXTRACCIÓN	
Tipo de Bot	
Nivel de inteligencia del Bot (PI4)	
Objetivo del Bot (PI-2)	
Etapas de desarrollo (PI-1)	
Problemas enfrentados con el uso del bot (PI-3)	

7. Estrategia para la síntesis de datos

En este punto se hablará detalladamente del proceso para la síntesis de datos, especificando cada paso e identificando datos. Para la síntesis de datos se elaborará una Síntesis Temática de Creswell adaptada para la IS (Cruzes & Dybå, 2011) y se obtendrán los siguientes datos específicos:

Tipo de dato	Dato
Publicación	Autores, Año, Título, Fuente, Abstracto, Tema, Objetivos
Contexto	Temas, Tecnologías, Configuraciones, Instrumentos, Tipo de estudio
Descubrimiento	Datos literales, Origen, Fuerza de la evidencia

De acuerdo con esta estrategia los pasos a seguir después de la extracción de datos son los siguientes:

- 1) Identificar segmentos específicos
Una vez concluida la extracción de datos, se identificarán los segmentos de texto en donde se encuentran los datos para la extracción. Estos datos estarán concentrados dentro del formato de extracción de datos producido de cada estudio restante después de la extracción de datos. Entonces los segmentos consistirán en los siguientes datos: Tipo de Bot, Nivel de inteligencia del Bot, Objetivo del Bot, Etapas de desarrollo y Beneficios reportados del uso del Bot.
- 2) Etiquetar segmentos
Una vez identificados los segmentos, se asignarán nombres únicos bajo un esquema dependiendo del nombre de cada tipo de dato. Cada segmento tendrá una etiqueta única que consiste en dos partes, el tipo de dato y un número asignado de manera secuencial. Por ejemplo, las etiquetas seguirán el patrón: Tipo-01, Tipo-02, ... Tipo-N.
- 3) Traducir etiquetados a temas
Después de la etiquetación, se agruparán los segmentos bajo temas principales bajo los cuales los segmentos tienen áreas en común. Estas áreas en común son conocidas como patrones o temas.

4) Categorizar temas

Al tener un conjunto de temas, se realizará un análisis y consecuente categorización en donde los temas serán agrupados en los temas de mayor importancia y relevancia con nuestras preguntas de investigación. En este paso se incrementa el nivel de abstracción de los temas.

5) Elaborar síntesis temática

Al obtener temas de un alto nivel de abstracción, se pueden establecer relaciones entre temas dentro de un modelo. En nuestro caso el modelo es un mapa utilizado para responder las preguntas de investigación.

8. Limitaciones.

Este trabajo está limitado principalmente por el alcance. Debido a que el alcance solo abarca actividades dentro del desarrollo, actividades fuera del ciclo de vida de desarrollo no serán tomadas en cuenta. De manera similar, tecnologías de la IA, o herramientas que automatizan tareas de desarrollo fuera de la definición de DevBots establecida en Términos.

Además de que en base a los accesos de de los investigadores, no se buscara en otras bases de datos fuera de la CONRYCIT brindado por la Universidad Veracruzana, e igualmente algunos motores de búsqueda de las bases de datos a utilizar pueden limitar nuestro tipo de filtración debido a los diferentes soportes que estos brindan. También entra como limitante una desventaja de la estrategia automatizada, que es que se sabe específicamente cuantos trabajos son falsos positivos desde el inicio y en contraste en una búsqueda manual al revisar trabajos se descartan trabajos irrelevantes al momento que son identificados

9. Informe

Debido al uso de revisiones sistemáticas y una metodología rigurosa cualquier informe resultante de este mapeo debe demostrar claramente que ha utilizado un proceso de revisión sistemático adecuado, ya sea mostrando una buena trazabilidad para brindarle al lector un claro vínculo con las preguntas de investigación hasta la información necesitada para la respuesta de estas.

Se debe lograr llegar a la audiencia objetiva, las cuales son principalmente investigadores e interesados en bots dentro de la ingeniería de software, publicando en la International Conference in Software Engineering Research and Innovation (CONISOFT) con la ayuda y guía de nuestro director.

También demostrar una repetibilidad asegurándonos de que la metodología se definió y detallo claramente a diferencia de otros estudios. Siguiendo la metodología de Kitchenham et. al. (2015) se realizará un reporte de estudios con la siguiente estructura:

Punto del informe	Descripción
Abstracto	Ayudará a la selección, proporcionando al lector (o motor de búsqueda) suficiente información para sugerir relevancia.
Introducción	Establecerá un contexto y deja en claro por qué el estudio es una contribución útil a el campo de investigación y por qué una revisión sistemática es apropiada.
Trabajo Relacionado	Ampliará la descripción proporcionada en la introducción y proporcionará información sobre estudios anteriores y sus contribuciones.
Método	Incluirá los elementos centrales del protocolo de investigación justificando la elección del tipo de revisión y el plan.
Resultados	Aquí se describirán los resultados de la búsqueda y la inclusión / exclusión, así como de la extracción de datos.
Discusión	Interpretará y considerará todos los resultados del análisis.
Validez del estudio y limitante	Demostrara y evidenciara la calidad del estudio realizado y sus limites
Conclusión	Finalmente abordara qué tan bien se han respondido las preguntas de investigación y cuáles son las respuestas que se obtuvieron.
Referencias	Evidencia de los trabajos utilizados y referenciados en todo el informe
Apéndices	En caso de anexar un documento externo en el informe se anexara dentro de esta zona de apéndices

Con este informe a realizar se busca:

- Responder las preguntas de investigación de una manera clara y completa
- Informar correctamente la metodología de la investigación
- Brindar una trazabilidad completa desde las preguntas hasta los datos obtenidos

10. Gestión de la revisión

Esta sección habla sobre las decisiones de gestión del protocolo de MSL como herramientas de soporte, que no se han registrado en otras partes del protocolo. Para gestionar el material encontrado para el MSL se usarán las siguientes herramientas de soporte:

- Mendeley
- Microsoft Excel

Ambas herramientas nos brindaran un soporte a la hora de manejar la información de los artículos encontrados y sus diferentes referencias. Además de que se pueden manejar de manera que 2 personas puedan manipular una misma colección para evitar problemas de integración entre diferentes colecciones.

El cronograma para seguir se ilustra en la siguiente página:

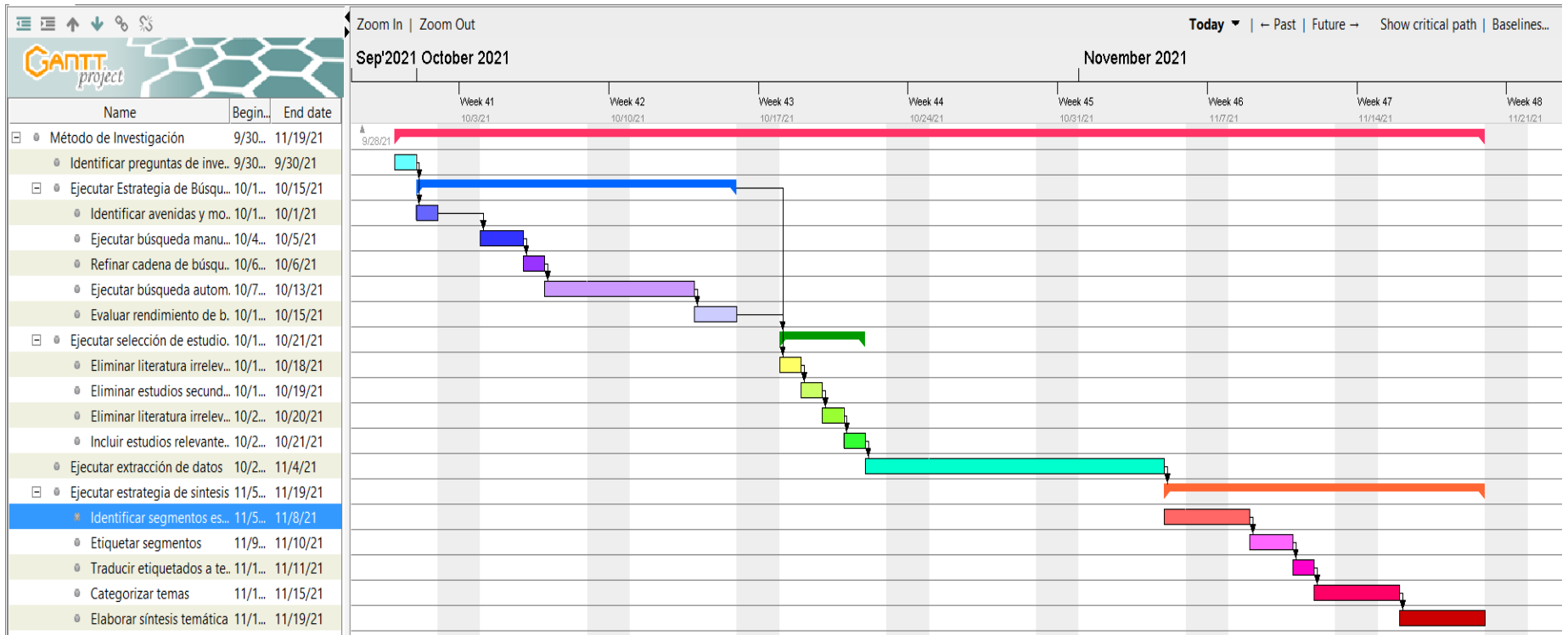


Figura 3 Cronograma de seguimiento del protocolo

11. Referencias

1. Babar, M. A., & Zhang, H. (2009). *Systematic literature reviews in software engineering: Preliminary results from interviews with researchers*. IEEE.
2. IBM. (1 de Julio de 2014). *Software Development*. Obtenido de IBM Research: https://researcher.watson.ibm.com/researcher/view_group.php?id=5227
3. Kitchenham, B. A., Budgen, D., & Brereton, P. (2015). *Evidence-based software engineering and systematic reviews*. CRC Press.
4. Lebeuf C., A. Z. (2019). Defining and Classifying Software Bots:A Faceted Taxonomy. *1st International Workshop on Bots in Software Engineering (BotSE)*.
5. Lebeuf, C. (2018). A Taxonomy of Software Bots: Towards a Deeper Understanding of Software Bot Characteristics. *University of Victoria, Greater Victoria, BC, Canada*, 16-18.
6. Wohlin, C. (2014). *Guidelines for Snowballing in Systematic Literature*. New York.
7. Zhang, H., Babar, M. A., & Tell, P. (2011). Identifying relevant studies in software engineering. *Information and Software Technology*, 53(6), 625-637.
8. Kotkov, D., Pandey, G., & Semenov, A. (2018, December). Gaming bot detection: a systematic literature review. In *International Conference on Computational Social Networks* (pp. 247-258).
9. Stieglitz, S., Brachten, F., Ross, B., & Jung, A. K. (2017). Do social bots dream of electric sheep? A categorisation of social media bot accounts.
10. Orabi, M., Mouheb, D., Al Aghbari, Z., & Kamel, I. (2020). Detection of bots in social media: A systematic review.
11. de Morais, D. M. G., & Digiampietri, L. A. (2021, June). Methods and Challenges in Social Bots Detection: A Systematic Review.
12. Alothali, E., Zaki, N., Mohamed, E. A., & Alashwal, H. (2018, November). Detecting social bots on Twitter: A literature review.
13. Wiesenber, M., & Tench, R. (2020). Deep strategic mediatization: Organizational leaders' knowledge and usage of social bots in an era of disinformation.
14. Collins, B., Hoang, D. T., Dang, D. T., & Hwang, D. (2020, November). Method of Detecting Bots on Social Media. A Literature Review.
15. Providel, E., & Mendoza, M. (2021). Misleading information in Spanish: a survey. *Social Network Analysis and Mining*.
16. Paul, H., & Nikolaev, A. (2021). Fake review detection on online E-commerce platforms: a systematic literature review.
17. Wang, P., Wu, L., Aslam, B., & Zou, C. C. (2009, August). A systematic study on peer-to-peer botnets.
18. Ben Sassi, I., & Ben Yahia, S. (2021). Malicious accounts detection from online social networks: a systematic review of literature.
19. Pérez, J. Q., Daradoumis, T., & Puig, J. M. M. (2020). Rediscovering the use of chatbots in education: A systematic literature review.
20. Jung, H., Lee, J., & Park, C. (2020). Deriving Design Principles for Educational Chatbots from Empirical Studies on Human–Chatbot Interaction.

21. Bhirud, N., Tataale, S., Randive, S., & Nahar, S. (2019). A Literature Review On Chatbots In Healthcare Domain.
22. Syvänen, S., & Valentini, C. (2020). Conversational agents in online organization– stakeholder interactions: a state-of-the-art analysis and implications for further research.
23. Borsci, S., Malizia, A., Schmettow, M., Van Der Velde, F., Tariverdiyeva, G., Balaji, D., & Chamberlain, A. (2021). The Chatbot Usability Scale: the Design and Pilot of a Usability Scale for Interaction with AI-Based Conversational Agents.
24. Kaartemo, V., & Helkkula, A. (2018). A systematic review of artificial intelligence and robots in value co-creation: current status and future research avenues.
25. Ciupe, A., Meza, S., & Orza, B. (2020, September). Systematic Assessment of Interactive Instructional Technologies in Higher Engineering Education.
26. Orgeolet, L., Foulquier, N., Misery, L., Redou, P., Pers, J. O., Devauchelle-Pensec, V., & Saraux, A. (2020). Can artificial intelligence replace manual search for systematic literature? Review on cutaneous manifestations in primary Sjögren's syndrome.
27. Skinner, G., & Walmsley, T. (2019, February). Artificial intelligence and deep learning in video games a brief review.
28. Blitch, J. G. (1996). Artificial intelligence technologies for robot assisted urban search and rescue.
29. Balachandran, V. (2013, May). Reducing human effort and improving quality in peer code reviews using automatic static analysis and reviewer recommendation.
30. Kim, D., Nam, J., Song, J., & Kim, S. (2013, May). Automatic patch generation learned from human-written patches.
31. Thongtanunam, P., Kula, R. G., Cruz, A. E. C., Yoshida, N., & Iida, H. (2014, June). Improving code review effectiveness through reviewer recommendations
32. Beschastnikh, I., Lungu, M. F., & Zhuang, Y. (2017, May). Accelerating software engineering research adoption with analysis bots.
33. Guzman, E., Ibrahim, M., & Glinz, M. (2017, May). Mining twitter messages for software evolution.
34. Avellis, G., Harty, J., & Yu, Y. (2017, May). Towards mobile twin peaks for app development. In 2017 IEEE/ACM 4th International Conference on Mobile Software Engineering and Systems (MOBILESoft) (pp. 189-193). IEEE.
35. Halder, R., Proença, J., Macedo, N., & Santos, A. (2017, May). Formal verification of ROS-based robotic applications using timed-automata.
36. Urli, S., Yu, Z., Seinturier, L., & Monperrus, M. (2018, May). How to design a program repair bot? insights from the repairnator project.
37. Paikari, E., & Van Der Hoek, A. (2018, May). A framework for understanding chatbots and their future.
38. Bradley, N., Fritz, T., & Holmes, R. (2018, May). Context-aware conversational developer assistants.
39. Brown, C. (2019, May). Digital nudges for encouraging developer actions.
40. Moreno, D., Duenas, S., Cosentino, V., Fernandez, M. A., Zerouali, A., Robles, G., & Gonzalez-Barahona, J. M. (2019, May). SortingHat: wizardry on software project members.
41. Kavalier, D., Trockman, A., Vasilescu, B., & Filkov, V. (2019, May). Tool choice matters: JavaScript quality assurance tools and usage outcomes in GitHub projects.
42. van Tonder, R., & Le Goues, C. (2019, May). Towards s/engineer/bot: Principles for program repair bots.
43. Wyrich, M., & Bogner, J. (2019, May). Towards an autonomous bot for automatic source code refactoring.
44. Erlenhov, L., de Oliveira Neto, F. G., Scandariato, R., & Leitner, P. (2019, May). Current and future bots in software development.
45. Matthies, C., Dobrigkeit, F., & Hesse, G. (2019, May). An additional set of (automated) eyes: Chatbots for agile retrospectives.
46. Subramanian, V., Ramachandra, N., & Dubash, N. (2019, May). TutorBot: contextual learning guide for software engineers.

47. Kumar, R., Bansal, C., Maddila, C., Sharma, N., Martelock, S., & Bhargava, R. (2019, May). Building sankie: An ai platform for devops.
48. Sharma, V. S., Mehra, R., Kaulgud, V., & Podder, S. (2019, May). A smart advisor for software delivery-a bot framework for awareness, alerts and advice.
49. Marginean, A., Bader, J., Chandra, S., Harman, M., Jia, Y., Mao, K., ... & Scott, A. (2019, May). SapFix: Automated end-to-end repair at scale.
50. Pinheiro, A. M., Rabello, C. S., Furtado, L. B., Pinto, G., & de Souza, C. R. (2019, May). Expecting the unexpected: distilling bot development, challenges, and motivations.
51. Paikari, E., Choi, J., Kim, S., Baek, S., Kim, M., Lee, S., ... & Cheong, C. (2019, May). A chatbot for conflict detection and resolution.
52. Monperrus, M. (2019, May). Explainable software bot contributions: Case study of automated bug fixes.
53. Lin, C. T., Ma, S. P., & Huang, Y. W. (2020, June). MSABot: A chatbot framework for assisting in the development and operation of microservice-based systems.
54. Dominic, J., Houser, J., Steinmacher, I., Ritter, C., & Rodeghero, P. (2020, June). Conversational bot for newcomers onboarding to open source projects.
55. Dominic, J., Ritter, C., & Rodeghero, P. (2020, June). Onboarding Bot for Newcomers to Software Engineering.
56. Abdellatif, A., Costa, D., Badran, K., Abdalkareem, R., & Shihab, E. (2020, June). Challenges in chatbot development: A study of stack overflow posts.
57. Dey, T., Mousavi, S., Ponce, E., Fry, T., Vasilescu, B., Filippova, A., & Mockus, A. (2020, June). Detecting and characterizing bots that commit code.