

Razones de Fracaso de Proyectos de Software: un Mapeo Sistemático

Barboza Héctor Emanuel¹, Balas José¹, Antonelli Leandro², Thomas Pablo³

¹ Facultad de Informática, UNLP.

² Lifia, Facultad de Informática, UNLP. CAETI - Facultad de Tecnología Informática - Universidad Abierta Interamericana

³ Instituto de Investigación en Informática (III-LIDI). Facultad de Informática, UNLP Centro Asociado CIC.

La Plata, Buenos Aires, Argentina.

{leandro.antonelli, pthomas}@lidi.info.unlp.edu.ar

Abstract. Existen varios motivos para que un proyecto de desarrollo de software se considere fracasado, y durante años se buscaron esos motivos. Como tal, se ha convertido en un tema de interés por varios investigadores y empresas que se dedican al desarrollo de software. En esta investigación se ha planteado realizar un mapeo sistemático o systematic mapping (SM, por sus siglas en inglés), tendiente a exponer diferentes causas de fracaso, que tienen los proyectos de software. Se encontró que los factores detallados en los casos de estudio tienen relación con factores de varias revisiones literarias, lo cual significa que estos factores son recurrentes y/o comunes.

Keywords: fracasos de proyectos de software, proyectos de gestión de software, éxito de proyectos de software, causas de fracaso de proyectos, factores críticos de fracaso.

1 Introduction

Las altas tasas de fracaso de los proyectos de desarrollo de software [8, 13] continúan preocupando a las organizaciones. Esta situación ha sido motivo para la disciplina durante décadas, lo que ha motivado a muchos investigadores a estudiar exhaustivamente los fracasos de los proyectos. Debido a la inmaterialidad del software, existe un problema de visualización que es fuente de muchos posibles fallos. La creciente complejidad e incertidumbre de los proyectos y los procesos de desarrollo de sistemas contemporáneos son un desafío para el fracaso de los proyectos de software [8]. La incertidumbre se genera a causa de una situación mal definida y falta de restricciones [11], es desconocido y sólo se puede reducir [33]. Es necesario comprender mejor los proyectos de software como iniciativas cada vez más complejas y complementar la investigación existente con enfoques que atiendan y analicen estos procesos complejos.

Muchos factores contribuyen a que un proyecto sea considerado como exitoso o fracasado [5]. Los motivos por los cuales los proyectos fracasan, se los conoce como Factores Críticos de Fracaso (FCF) [1]. Estos motivos son importantes conocerlos porque permite concentrar los esfuerzos del equipo para evitarlos. También, es habitual encontrar que un mismo proyecto sea considerado un éxito por unos y un fracaso por otros, ya sea el propietario, project manager, desarrolladores, contratistas, usuarios, etc. [11] Durante años, estas investigaciones fueron un tema de interés para prevenir futuros problemas y para encontrar la fórmula y las formas para garantizar el éxito de un proyecto [4]. Tanto el éxito como el fracaso se estudian intensamente en la literatura de gestión de proyectos informáticos, de sistemas, de software, entre otras.

Basado en la metodología de mapeo sistemática, se identificaron, analizaron e incluyeron 37 artículos y trabajos relevantes de estos últimos años que resumen cuáles son esos factores para que un proyecto de software fracase. Este documento está estructurado de la siguiente manera. En la sección 2 se resume la metodología de mapeo sistemático para detectar los factores críticos de fracaso; los resultados y la discusión se desarrollan en las secciones 3 y 4. Finalmente, la sección 5 presenta las conclusiones y trabajos futuros.

2 Metodología de Investigación

Para alcanzar el objetivo propuesto, el presente estudio utilizó la metodología de mapeo sistemático propuesta en [17], el cual es un método útil para construir clasificaciones y obtener información sobre el conocimiento existente en una temática específica.

2.1 Pregunta de Investigación

La idea fue recopilar la mayor cantidad posible de literatura relevante y tratar de presentar una evaluación justa de un tema de investigación utilizando una metodología que ofrezca una visión panorámica y general, sin entrar en detalles profundos, del estado en un área amplia. La principal pregunta de investigación es:

RQ1: *¿Por qué fracasan los proyectos de software en la actualidad?*

Así el objetivo es encontrar una respuesta para conocer cuáles son aquellos motivos.

2.2 Proceso de Búsqueda

Primero, se identificaron los principales motores de búsqueda donde se suben artículos, conferencias, libros, revistas científicas, etc. En base a nuestro criterio consideramos los más relevantes y de alta calidad como ResearchGate, Scielo, Google Scholar, ScienceDirect, entre otras. Se seleccionaron y ordenaron los artículos por

relevancia y años. En la búsqueda se usaron alguna de las siguientes palabras claves o cadena de búsqueda, tanto en inglés como en español: “failures”, “critical factors of failure”, “factors of failure”, “causes of failure”, “software fails”, “software project failure”, entre otras. En algunos casos, sumado a siglas y sinónimos como “IS” (Information System), ERP (Enterprise Resource Planning), “IT” (Information Technology), o “ISD” (Information System Development).

2.3 Criterios de inclusión y exclusión

En la Tabla 1, se detallan los criterios de inclusión y de exclusión, considerados en este trabajo.

Table 1. Tabla de Criterios de inclusión y exclusión .

Criterios de Inclusión	Criterios de Exclusión
<ul style="list-style-type: none"> • Artículos publicados en las plataforma • Se incluyen revisiones sistemáticas de literaturas previas • Los artículos deben contener términos que coincidan con los definidos en la búsqueda • Los trabajos incluyen el término fracaso ya sea en el título, en el resumen o en la conclusión. • Artículos en idiomas inglés o español. • Trabajos relacionados a: IT, IS y ISD, ERP 	<ul style="list-style-type: none"> • Se excluyen libros o ensayos • Se excluyen trabajos finales o tesis de grados o postgrados • Artículos duplicados • Se excluyen artículos publicados previamente al año 2020 • Se excluyen fracasos de proyectos fuera del desarrollo de software.

3 Resultados

La búsqueda dió como resultado aproximadamente 1540 artículos potencialmente relevantes. De estas, 37 publicaciones fueron consideradas primarias porque cumplieron con nuestros criterios. Los resultados que se detallan en Tabla 2, arrojaron artículos relacionados únicamente con el fracaso, éxito, o ambos, en proyectos de desarrollo de software de diversos tipos. Esto se debe a que algunos autores estudian lo que se debe hacer y otros lo que no se debe hacer para llegar al éxito de un proyecto.

Table 2. Enfoque de tema

Enfoque del tema	Nº de Estudios	Porcentaje	Ref. Artículo
Factores de éxito	6	16.22%	23,29,32,34,39,40
Factores de fracaso	11	29.73%	2,8,16,18,19,20,21,22,30,33,41
Ambos factores	14	37.83%	1,4,5,6,7,11,13,14,24,25,26,27,31,35
Otros temas que involucran el fracaso	6	16.22%	9,15,28,36,37,38
Total	37	100 %	

Se analizó el título y el resumen (abstract) para saber si se considera dentro de este trabajo. Ambas miradas, dependiendo del abstract o conclusiones que se sacan, se pusieron en consideración para hacer una contribución más enriquecedora y centrada en el fracaso. Se centró en artículos recientes y actuales, considerando un rango de fechas de publicaciones del año 2020 en adelante, es decir en estos últimos 4 años. A continuación en la Tabla 3 se pueden ver los trabajos encontrados con información sobre el enfoque metodológico, si es cualitativo o cuantitativo y a qué sectores o tipos proyectos considera.

Table 3. Detalle de artículos incluidos

Ref. Art.	Enfoque Metodológico	Tipo de investigación	Tipo de Proyectos
1	Revisión literaria y estudio teórico	Cualitativa	En general IT
2	Análisis teórico y estudio empírico	Cualitativa	En general IT
4	Estudio empírico	Cualitativa	En general IT
5	Estudio empírico	-	En general IT
6	Estudio empírico	Mixto	ERP, CRM y IT
7	Estudio empírico y revisión de datos	Cuantitativo	IT, ERP, IS, ISD, CR, Ágiles y otras metodologías
8	Revisión literaria	Cualitativo	En general ágiles y ISD
9	Análisis teórico y crítico	-	En general IT
11	Revisión literaria	Cualitativo	En general IT
13	Estudios empíricos	Mixto	Todo tipos
14	Revisión literaria / compilación y síntesis de estudios previos	Cualitativo	Todo tipos
15	Estudio teórico	Cualitativo	Empresarial en general
16	Estudio de caso	Cualitativo	En particular ERP, CRM y aplicaciones empresariales
18	Estudio teórico	Cualitativo	IT, IS, ISD
19	Revisión literaria	Cualitativo	Todos los tipos
20	Análisis empírico y revisión literaria	Mixto	Ágiles y en general IT
21	Estudio de caso	Cualitativo	Software en particular
22	Revisión literaria	Mixto	Específicamente ERP
23	Estudio empírico	Cuantitativa	Agilidad aplicable en sectores como IT y IS
24	Estudio empírico	-	Todos
25	Estudio empírico	Cuantitativo	En general IT
26	Revisión literaria	Cualitativo	En general IT y Agilidad
27	Revisión literaria	Cualitativo	Todos tipos
28	Revisión literaria	Cualitativo	Agilidad
29	Estudio empírico	Cuantitativo	Agilidad
30	Estudio empírico	Cualitativo	Todo tipo
31	Revisión literaria	Cualitativo	En general IT
32	Estudio empírico	Cuantitativo	Global Software Development
33	Estudio teórico	-	En general IT
34	Revisión literaria	Cualitativo	En general IT en Industria 4.0

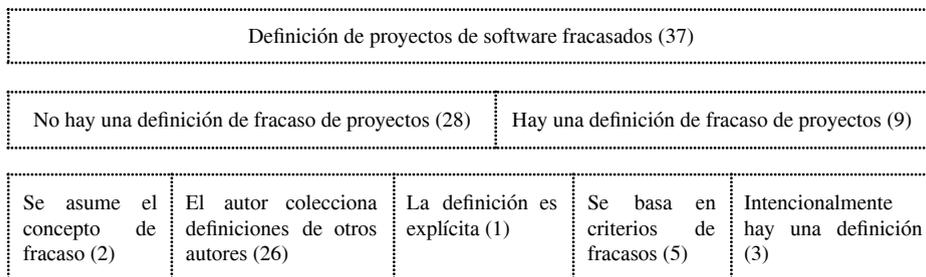
35	Estudio empirico	Cuantitativo	Agilidad
36	Estudio empirico	Cuantitativo	En general IT y Requisitos
37	Revision literaria	Cualitativa	Todo tipos
38	Revision literaria	Cualitativa	En general IT
39	Estudio empirico	Cuantitativo	Todo tipos
40	Revision literaria	Cualitativa	Todo tipos
41	Estudio empirico	Mixto	Todo tipos

Como se puede observar hay muchas revisiones literarias que analizan y sintetizan el conocimiento existente de estudios previos; y estudios empíricos que se basan en recopilación de datos de la vida real, ya sea por encuestas, experimentos o análisis de datos de proyectos. En definitiva se encontró que muchas de las conclusiones de estos artículos no son las mismas, por la variedad de perspectivas, pero son útiles para sacar nuestras conclusiones.

4 Discusión

Los tipos de definiciones de fracasos en proyectos de software se muestran en Fig. 1 donde se observa que hasta el día de hoy, no hay una definición de fracaso explícita y donde los autores coleccionan definiciones de otros autores a través de citas bibliográficas de estudios previos.

Fig. 1. Tipos de definiciones de proyectos de software fracasados



Cuando se realizó la búsqueda bibliográfica, una de las fuentes más citadas en muchos artículos es el famoso Chaos Report de Standish Group que a lo largo de los años, este informe ha sido una referencia clave para entender las tendencias y los factores críticos que influyen en los resultados de los proyectos de software. En el último informe publicado en 2020 [7], quedó demostrado que un alto porcentaje de proyectos de software fueron descartados y cuestionados. El 31,1% de los proyectos están clasificados como fracasados, lo que significa que fueron abandonados o cancelados y el 52,7% se completan por encima del costo, con el tiempo y/o carecen de la funcionalidad prometida. Antes, la referencia que se usaba era el triángulo de hierro o triple restricción [12] para establecer el éxito o no de un proyecto. Pero con el auge de las metodologías ágiles, las partes interesadas tuvieron que fijarse en otros factores que complementen como la satisfacción del cliente, calidad del producto, habilidades

técnicas, etc. Así los objetivos e intereses de las partes interesadas se extienden más allá de esta triple restricción [27].

Es importante destacar que no existe un consenso claro sobre la definición de fracaso (o éxito) de un proyecto [8, 23]. Determinar las razones subyacentes del fracaso aún sigue siendo un tema polémico y que todavía hay mucho que aprender al estudiar el fracaso de un proyecto [41]. La definición de fracaso de un proyecto de software también es importante para elegir los estudios de caso adecuados; sin embargo, es impreciso y variado [13]. Etiquetar un proyecto de exitoso o fracasado puede resultar dificultoso y problemático, porque la definición depende de la persona, del proyecto, del contexto, momento, etc. [1, 11,13], es decir, un fracaso parece ser algo que ha sucedido “a los ojos de quien lo mira” [23].

Se encontraron aproximadamente 23 factores de fracaso en esta investigación, algunas similares pero expresadas con otras palabras, donde las siete primeras son consideradas como las más críticas. A continuación, en Tabla 4 se detallan estos resultados obtenidos de la búsqueda.

Table 4. Factores críticos de fracasos detectados en proyectos de software

N°	Factor de fracaso	Ref. Artículo	Cant. estudios	Detalle
F1	Falta de comunicación entre los miembros del equipo	1,2,4,5,8,9,11,1,3,14,16,19,21,22,23,24,25,27,28,30,31,32,34,35,36,37,38,39,40	28	A través del análisis detallado, los autores de [6] sacan la conclusión de que todos los proyectos exitosos se beneficiaron de una excelente comunicación entre los miembros del equipo. Es por ello que la comunicación debe ser efectiva [1, 4, 5, 7]. La hipótesis de [12] es que la confianza de los miembros en el equipo modera la relación de cohesión percibida y éxito del proyecto.
F2	Requisitos incompletos, mal definidos, inadecuada captura, gestión deficiente de requisitos	2,6,7,9,11,13,19,18,21,23,26,32,33,37,40,41	16	Las tareas clave de análisis de requisitos deben practicarse de manera correcta y confiable sin importar qué metodología de desarrollo de sistemas se utilice [41]. Es necesario dar lugar a la implementación de prácticas sólidas para capturar, documentar y gestionar los requisitos del proyecto de manera continua [7]. Se requiere una planificación continua para hacer frente a requisitos nuevos y cambiantes, y las actividades de ingeniería de requisitos se realizan con conceptos compartidos [9].

F3	Cronograma poco realista, mala planificación, falta de planificación adecuada	1,4,13,16,19,23,24,27,28,30,33,35,36,37,40,41	16	“Se dice popularmente que no planificar es planificar el fracaso” [28]. La fase de planificación en un proceso de desarrollo está muy relacionado con los problemas de requisitos, otro factor de fracaso, donde un ingeniero de software reconoce que los requisitos disponibles no son consistentes ni estables [35]. En estudios de [39] la hipótesis plantea una pregunta de investigación sobre cuánto riesgo se evalúa en la planificación del proyecto y cómo se relaciona la planificación con el éxito del proyecto o no. Los resultados muestran que el efecto estimado de la planificación y gestión de riesgos para el éxito es insignificante.
F4	Falta de conocimientos y habilidades necesarias del equipo de trabajo	1,2,6,7,11,16,23,26,34,39,40,41	12	Gestionar las dificultades técnicas es mucho más sencillo que monitorear a personas sin conocimientos técnicos que tienen sus propias creencias, fortalezas y deficiencias [4,23]. El descuido del papel de los humanos está asociado a este factor, un tema sobre el que hay escasez de investigaciones, a pesar de que tiene una importancia estratégica en los proyectos.
F5	Mala gestión de proyectos en general	2,5,8,14,19,22,24,30,31,32,34	11	Es clave la mejora y formación de gestión de proyectos. Las encuestas realizadas en [4] remarcan bastante esto donde la gestión de proyectos es un área en la que se necesita un buen nivel de habilidades donde es clave la capacitación. Las empresas deberían invertir en la capacitación del equipo en prácticas modernas de gestión de proyectos y habilidades blandas tanto a líderes como miembros de equipo para resolver por ejemplo conflictos [5].
F6	Gestión inadecuada del cambio o resistencia al cambio	2,14,16,20,22,23,25,26,28,31,32	11	El cambio es un problema que puede tener varios efectos en el proyecto de software. No existe una causa fundamental para los cambios, lo que hace que la gestión de cambios sea una tarea desafiante [37]. Otra mal interpretación es que la recuperación de un fracaso o de un fracaso potencial no significa que el proyecto finalmente tendrá éxito según lo previsto en el plan original, sino que el proyecto podrá ofrecer una capacidad funcional dentro de un cronograma y presupuesto revisados que es un subconjunto [41].
F7	Falta de apoyo o compromiso de la alta gerencia.	1,2,7,14,16,20,22,23,24,27,41	11	No se puede dejar de enfatizar la importancia de la toma de decisiones como la variable de éxito con la puntuación más alta [27]. El apoyo de la gerencia es importante, pero no es tan decisivo en el éxito de un proyecto, pero la carencia de ello, es totalmente decisivo para que un proyecto fracase [1]. Sobre todo, el liderazgo y comunicación del project manager son absolutamente importantes.

A pesar de que las investigaciones han definido al fracaso de manera muy diferente, o hayan abordado la misma desde perspectivas muy diferentes, hay muchas causas o factores que son comunes. Para que los proyectos sean exitosos, los miembros del equipo deben tomar precauciones para mitigar los riesgos de fracaso del proyecto [6]. La idea es abrir la "caja negra" del fracaso y permitir a los individuos dar a conocer y compartir abiertamente sus experiencias. Las experiencias de fracaso proporcionan

una fuente mejorada para la transferencia de aprendizaje. Debido a su componente negativo, en comparación con las historias sobre éxitos, provocan en quienes las escuchan una respuesta motivacional más pronunciada, de modo que las personas elaboran el contenido más activamente como oportunidades de aprendizaje[15].

5 Conclusión y Trabajos Futuros

El presente trabajo realizó un mapeo sistemático de la literatura sobre las razones por las cuales fracasan los proyectos de software en la actualidad. Se consolidaron los factores de fracaso basándose en estudios recientes, se ordenaron estos factores por su importancia y se presentó un ranking junto con recomendaciones para adherirse a las mejores prácticas de desarrollo, gestión de proyectos y calidad. Los desafíos y sugerencias formulados a partir de este estudio pueden utilizarse para mejorar las habilidades y capacidades de las empresas de desarrollo de software.

El enfoque del fracaso de un proyecto de software como un estado final de angustia y situaciones problemáticas es común en varias bibliografías, pero se espera un enfoque más temprano para abordar problemas y generar valor relevante para la práctica. Una propuesta valiosa podría ser idear una metodología de desarrollo cuyos valores y prácticas apunten a los factores identificados en este trabajo.

Como trabajo futuro se pretende estudiar casos específicos, como fracasos en diferentes tipos de desarrollo de software (ágiles o tradicionales), tamaños de proyectos (pequeños, medianos y grandes), estructuras de equipo (jerárquicas, circulares, basadas en procesos, etc.), tipos de equipo (multidisciplinarios, formales o tradicionales), entre otros.

Referencias bibliográficas

1. Lam, ZhingFong & Peña, Marcos. (2023). Los Factores de Fracaso y Éxito en un proyecto de desarrollo de Software. *Revista peruana de computación y sistemas*. 5. 3-12. 10.15381/rpcs.v5i1.25799.
2. Lauesen, S. (2020). IT project failures, causes and cures. *IEEE access*, 8, 72059-72067.
3. Peláez Valencia, L. E., Cohuo Ávila, M. A., Toro Lazo, A., Arias Vargas, J. L., Rodríguez Franco, D. E., Gómez Osorno, Y., & Pérez Correa, D. A. (2020). Diseño de un sistema de información que implementa el modelo automatizado para asegurar la calidad de los requerimientos en proyectos de software.
4. Kotowaroo, M. Y., & Sungkur, R. K. (2022). Success and Failure Factors Affecting Software Development Projects from IT Professionals' Perspective. In *Soft Computing for Security Applications: Proceedings of ICSCS 2022* (pp. 757-772). Singapore: Springer Nature Singapore.
5. Basirati, Mohammad R. & Otasevic, Marko & Rajavi, Koushyar & Böhm, Markus & Krcmar, Helmut. (2020). Understanding the Relationship of Conflict and Success in Software Development Projects. *Information and Software Technology*. 126. 106331. 10.1016/j.infsof.2020.106331.

6. Ramírez Mora, Sandra & Oktaba, Hanna & Gomez Adorno, Helena. (2020). Descriptions of issues and comments for predicting issue success in software projects.
7. Standish Group, CHAOS Report (2020). Available at http://www.standishgroup.com/sample_research/chaos_2020.php [Accessed on 15th March 2022].
8. Baghizadeh, Z., Cecez-Kecmanovic, D., & Schlagwein, D. (2020). Review and critique of the information systems development project failure literature: An argument for exploring information systems development project distress. *Journal of Information Technology*, 35(2), 123-142.
9. Rasheed, A., Zafar, B., Shehryar, T., Aslam, N. A., Sajid, M., Ali, N., ... & Khalid, S. (2021). Requirement engineering challenges in agile software development. *Mathematical Problems in Engineering*, 2021(1), 6696695.
10. Khalid, A., Butt, S. A., Jamal, T., & Gochhait, S. (2020). Agile scrum issues at large-scale distributed projects: scrum project development at large. *International Journal of Software Innovation (IJSI)*, 8(2), 85-94.
11. Iriarte, C., & Bayona, S. (2020). IT projects success factors: a literature review. *International Journal of Information Systems and Project Management*, 8(2), 49-78.
12. Imam, H. y Zaheer, MK (2021). Liderazgo compartido y éxito del proyecto: los roles del intercambio de conocimientos, la cohesión y la confianza en el equipo. *Revista internacional de gestión de proyectos*, 39 (5), 463-473.
13. Nizam, Ali. (2022). Software Project Failure Process Definition. *IEEE Access*. 10.101109/ACCESS.2022.3162878.
14. Dempsey, M., Geitner, L., Brennan, A., & McAvoy, J. (2021). A review of the success and failure factors for change management. *IEEE Engineering Management Review*, 50(1), 85-93.
15. Dupleix, María Dolores. (2021). La teoría efectual y el fracaso empresarial. *Innovar*, 31(81), 139-153. Epub August 03, 2021. <https://doi.org/10.15446/innovar.v31n81.95581>.
16. Salazar-Fierro, F. A., Guamán-Cupacán, B. R., León-Fernández, C. V., Pineda-Manosalvas, C. A., Reascos-Paredes, I. M., & Herrera-Quispe, J. A. (2023). Fracazos en la implantación de aplicaciones informáticas empresariales: Casos de estudio y sus principales causas. *Revista Colombiana De Computación*, 24(2), 1–11. Recuperado a partir de <https://revistas.unab.edu.co/index.php/rcc/article/view/4663>.
17. Petersen, Kai & Vakkalanka, Sairam & Kuzniarz, Ludwik. (2015). Guidelines for conducting systematic mapping studies in software engineering: An update. *Information and Software Technology*. 64. 10.1016/j.infsof.2015.03.007.
18. Christoph A. Thieme, Ali Mosleh, Ingrid B. Utne, Jeevith Hegde, Incorporating software failure in risk analysis – Part 1: Software functional failure mode classification, *Reliability Engineering & System Safety*, Volume 197, 2020, 106803, ISSN 0951-8320.
19. Gazem, Adeeb & Thawaba, Abdulaziz. (2024). Review on Characteristics of Software Development Frameworks to Reduce Critical Systems Failures. 7. 10.54582/TSJ.2.2.94.
20. (2023). Failure Factors in Agile Software Projects in SME Organizations. *Al Bayan Scientific Journal*, (7), 225–208. Retrieved from <https://journal.su.edu.ly/index.php/bayan/article/view/2360>.
21. Ebad, Shouki. (2020). Healthcare software design and implementation-A project failure case. *Software: Practice and Experience*. 50. 10.1002/spe.2807.
22. Evren Coşkun, Bahar Gezici, Murat Aydos, Ayça Kolukısa Tarhan, Vahid Garousi, ERP failure: A systematic mapping of the literature, *Data & Knowledge Engineering*, Volume 142, 2022, 102090, ISSN 0169-023X, <https://doi.org/10.1016/j.datak.2022.102090>.
23. Carlos Tam, Eduardo Jôia da Costa Moura, Tiago Oliveira, João Varajão, The factors influencing the success of on-going agile software development projects, *International*

- Journal of Project Management, Volume 38, Issue 3, 2020, Pages 165-176, ISSN 0263-7863, <https://doi.org/10.1016/j.ijproman.2020.02.001>.
24. Ali, Israr & Hussaan, Aarij & Adil, Syed. (2020). A Study of The Role of Software Project Manager in the Outcome of the Project.
 25. Binboga, Burcu & Altin Gumussoy, Cigdem. (2024). Factors Affecting Agile Software Project Success. IEEE Access. PP. 1-1. 10.1109/ACCESS.2024.3384410.
 26. Ortega-Ordóñez, Wilson Alfredo, Pardo-Calvache, César Jesús, & Pino-Correa, Francisco José. (2019). Mapeo sistemático sobre la evaluación de la agilidad en organizaciones de desarrollo de software. Iteckne, 16(1), 64-76. <https://doi.org/10.15332/iteckne.v16i1.2162>.
 27. Robert Hans, Senyeki Marebane, Are software projects evaluated using software teams' success criteria? A systematic literature review, Procedia Computer Science, Volume 219, 2023, Pages 1599-1608, ISSN 1877-0509, <https://doi.org/10.1016/j.procs.2023.01.452>. (<https://www.sciencedirect.com/science/article/pii/S1877050923004635>).
 28. Lukman, Yusuf & Balkaran, Sanjay & Makiwane, N & Sentiwe, M. (2022). Success and failure of projects: A stakeholder's outlook in the wake of crises. 1-9.
 29. Leonor Barros, Carlos Tam, João Varajão, Agile software development projects–Unveiling the human-related critical success factors, Information and Software Technology, Volume 170, 2024, 107432, ISSN 0950-5849, <https://doi.org/10.1016/j.infsof.2024.107432>. (<https://www.sciencedirect.com/science/article/pii/S0950584924000375>).
 30. Coccia, Mario. (2023). Causes of failures in project management: Analysis and inductive evidence. Qeios. 10.32388/8SGX71.
 31. Pejic Bach, Mirjana & Omazić, Mislav & Miloloža, Ivan. (2021). Success determinants of projects of business software implementation: Research framework. Croatian Regional Development Journal. 2. 1-16. 10.2478/crdj-2021-0010.
 32. Alqahtani, Jarallah & Siddique, Ansar & Aseere, Ali & Alasiry, Areej & Quadri, Noorulhasan Naveed. (2024). Evaluating Success Factors of Software Project Management in Global Software Development. IEEE Access. PP. 1-1. 10.1109/ACCESS.2024.3360415.
 33. Kaisler, Stephen & Money, William & Cohen, Stephen. (2022). Applying Forensic Analysis Factors to Construct a Systems Dynamics Model for Failed Software Projects. 10.24251/HICSS.2022.145.
 34. Vrchota, J.; Řehoř, P.; Maříková, M.; Pech, M. Critical Success Factors of the Project Management in Relation to Industry 4.0 for Sustainability of Projects. Sustainability 2021, 13, 281. <https://doi.org/10.3390/su13010281>.
 35. Iqbal, Salwa & Raffat, Sheikh Kashif & Sarim, Muhammad. (2023). Software Projects Crest and Trough in Pakistan: A Management Spectrum. Sukkur IBA Journal of Computing and Mathematical Sciences. 7. 12-29. 10.30537/sjcms.v7i1.1249.
 36. Adnan Afaq, Syed & Faisal, Mohammad. (2020). Influence of Requirement Change on Software Failure.
 37. Carballo Muñoz, Lenna, & Barrientos Núñez, Ivette. (2020). Las causas del cambio en los requerimientos de software. Revista Cubana de Ciencias Informáticas, 14(2), 131-144. Epub 01 de junio de 2020.
 38. Morrison-Smith, S., & Ruiz, J. (2020). Desafíos y barreras en equipos virtuales: una revisión de la literatura. SN Applied Sciences , 2 (6), 1-33.
 39. Iqbal, Salwa & Raza, Akhter & Raffat, Sheikh Kashif. (2024). A Quantitative Study on Success Factors of Software Project Development in Pakistan.
 40. Ibraigheeth, Mohammad & Fadzli, Syed. (2019). Core Factors for Software Projects Success. JOIV : International Journal on Informatics Visualization. 3. 10.30630/joiv.3.1.217.
 41. Money, William & Kaisler, Stephen & Cohen, Stephen. (2021). Understanding Failed Software Projects through Forensic Analysis. Journal of Computer Information Systems. 62. 1-14. 10.1080/08874417.2021.1950076.