

Guía ágil para el desarrollo de software en entidades públicas en Colombia empleando CMMI 2.0 y SCRUM

Nombre de los estudiantes

Jhon Mario Calderón Jaraba
Breyner Alberto Lascarro Altamiranda
Cesar Orlando Mattos Badillo
Victor Manuel Mercado Cervantes

Trabajo de Investigación o Tesis Doctoral como requisito para optar el título de

Especialista de ingeniería de software

Tutores

Adriana María Iglesias Solano

RESUMEN

El presente trabajo de investigación propone una guía en marcada dentro de los principios y valores del agilismo que oriente los procesos de desarrollo de soluciones de software en entidades públicas en Colombia, buscando la armonización entre CMMI 2.0 y el framework de Scrum. Durante el desarrollo de la investigación se logró realizar en primera instancia la aplicación de un instrumento diagnóstico en un grupo de entidades de naturaleza pública, posteriormente, se realizó una caracterización que permitió la identificación de componentes, roles, artefactos y actividades de la guía a proponer. Finalmente, se describe la guía denominada GIDSEO.

Antecedentes:**1. Introducción:**

Las entidades de carácter público en Colombia tienen la misión de ofrecer servicios a sus ciudadanos basado en la gestión de administrar los recursos públicos, por lo que estas entidades requieren implementar soluciones tecnológicas que permitan operar de forma eficiente y además de prestar servicios a los ciudadanos. Por lo anterior, el Ministerio de Tecnologías Información y Comunicaciones – MINTIC expide en el año 2017 el decreto 1499, el cuál describe la reglamentación del Sistema Integrado de Planeación y Gestión denominado Modelo Integrado de Planeación y Gestión – MIPG que establece un marco de referencia que permite dirigir, planear, ejecutar, hacer seguimiento, evaluar y controlar la gestión de las entidades públicas.

De acuerdo con lo anteriormente expuesto, es importante resaltar la evolución de estos programas del gobierno de Colombia en la búsqueda de transparencia, promoción, uso y aprovechamiento de las tecnologías de la información y las comunicaciones - TIC en provecho del estado y sobre todo de los ciudadanos. Siendo así, como luego de varios años de implementación de la estrategia de Gobierno en línea en Colombia, esta evolucionó a lo que se conoce como Gobierno Digital, siendo este último un modelo que debe ser adoptado por los organismos y entidades de los órdenes nacional y territorial.

Esto ha requerido que estas entidades establezcan mecanismos para poder dar cumplimiento a estas políticas de Gobierno Digital, una de ellas se enfoca en el desarrollo de soluciones de software o sistemas de información que apoyan la gestión y el día a día de cada entidad, y que atiendan las necesidades tanto internas como hacia sus ciudadanos, esto ha impulsado la creación de equipos de desarrollo de software a medida al interior de las diferentes oficinas y dependencias.

Como estas entidades no son fábricas de software, hace que quizás no se apliquen buenas prácticas y metodologías concernientes a la ingeniería del software que permitan reducir la tasa de fallos y que permita mantener atributos de calidad, el uso inadecuado de estas metodologías de desarrollo de software ha traído como consecuencia productos de software con fallas o deficiencias que redundan en reprocesos o proyectos cancelados, a esto se le agrega que cada equipo de desarrollo de software suele aplicar sus propios criterios o modelos de desarrollo de aplicaciones al no existir una guía base que les de las orientaciones y buenas prácticas que le permitan a la entidad darle continuidad a estos proyectos sin importar si hay rotación de personal o cambio de administraciones.

Teniendo en cuenta esta situación y el impacto que este genera sobre la eficiencia y eficacia de los proyectos de software es importante brindar apoyo a todos los equipos de desarrollo de aplicaciones para que la entidad pueda contar con soluciones de software de calidad que puedan suplir sus necesidades, sin llegar a definir una metodología formal pero que le permita cumplir con los requerimientos de los entes de control en cuanto a llevar de forma ordenada y documentada cada uno de los procesos. Debido a esto se propone una guía que establezca prácticas en el desarrollo de software para las entidades públicas conservando elementos de buenas prácticas y agilidad.

2. Planteamiento del problema:

Actualmente los proyectos de software ejecutados por los equipos de desarrollo de las entidades públicas en Colombia presentan una serie de dificultades que han incidido en el éxito o fracaso de los proyectos, en el listado de estas problemáticas tenemos: el software no cumple con las expectativas, falta de procedimientos claros y estandarizados para gestionar la calidad del software que se produce y proyectos de software que no aportan valor a la entidad. Ahora, si se aborda un poco más al detalle las problemáticas mencionadas, podríamos evidenciar más claramente estas falencias:

- El software no cumple con las expectativas
- Falta de procedimientos claros y estandarizados para gestionar la calidad del software que se produce
- Proyectos que no aportan valor a la entidad

3. Formulación del problema:

De acuerdo con la definición del problema descrito se plantea la siguiente pregunta:

¿Existe una guía para desarrollar software aplicando modelo de calidad y gestión de proyectos ágiles para entidades públicas en Colombia?

Objetivos:

1. Objetivo general

Proponer una guía ágil para el desarrollo de software en entidades públicas en Colombia empleando CMMI 2.0 y SCRUM.

2. Objetivos específicos

- Analizar a través de un instrumento diagnóstico el estado actual en cuanto al desarrollo de software en entidades públicas en Colombia.
- Relacionar cada una de las actividades y prácticas de CMMI 2.0, Scrum y los lineamientos de gobierno digital, estableciendo una correspondencia entre cada una de ellas
- Identificar los componentes de calidad de software y la gestión del proceso de desarrollo en soluciones de software, que conformará la guía.

Materiales y Métodos:

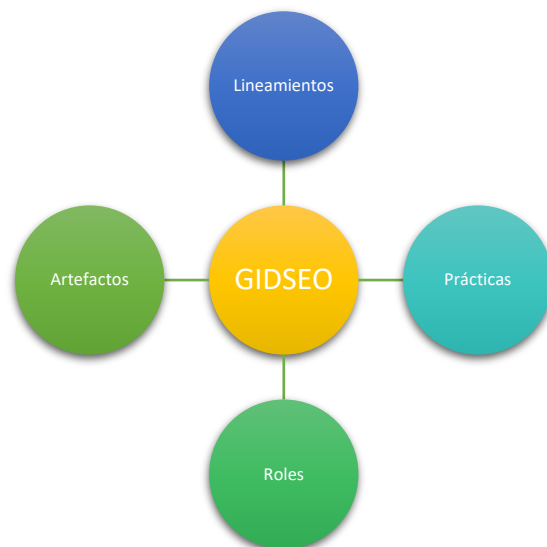
Este trabajo de investigación se centrará básicamente en un tipo de investigación descriptiva, ya que a partir de los hallazgos encontrados en la revisión documental y del diagnóstico se planteará una guía para el desarrollo de software en entidades públicas en Colombia, para lograrlo se abarcarán las siguientes tres (3) fases metodológicas:

- **Fase de análisis:** en esta fase se realizará un diagnóstico a partir de la aplicación de un instrumento que permita determinar la forma en cómo son gestionados los proyectos de desarrollo de software en las entidades de naturaleza pública en Colombia.
- **Fase de correspondencia:** en esta fase se caracteriza el modelo de calidad escogido, que en este caso será CMMI 2.0, en específico para desarrollo, y el framework de Scrum en su versión 2020. Posterior a este proceso, se pasa a definir cada uno de los componentes y relaciones entre estos, indicando roles, artefactos, actividades, etapas, prácticas que formaran parte de la guía a proponer.
- **Fase de definición:** en esta fase se busca establecer un modelo de madurez que de lineamientos a la implementación de la guía a proponer

Resultados:

Guía práctica para desarrollos de software en entidades oficiales "GIDSEO".

GIDSEO es una guía para el desarrollo de software que armoniza tres componentes los cuáles son: calidad de software, gestión de proyectos por medio de políticas Nacionales y framework de desarrollo de software enmarcado en el agilismo. Sin embargo, GIDSEO se orienta a las entidades públicas nacionales o territoriales en Colombia. Adentrándonos en la guía tenemos que esta se encuentra compuesta por cuatro (4) elementos los cuáles son:



A continuación, se describe cada uno de los elementos de GIDSEO señalando las prácticas, lineamientos, artefactos y roles correspondientes.

GIDSEO			
Lineamiento	Practica	Artefactos	Roles
Asegure la Generación de Valor para la Entidad y los Usuarios	Desarrollo y gestión de requisitos	<ul style="list-style-type: none"> Historias de usuario Estudios previos Objetivo del Sprint Sprint Planning 	<ul style="list-style-type: none"> Product owner Stakeholders Arquitecto de solución QA
	Aseguramiento de la calidad del proceso	<ul style="list-style-type: none"> Definición de Terminado Sprint Retrospective 	<ul style="list-style-type: none"> Scrum master Team dev
	Capacitación organizacional	<ul style="list-style-type: none"> Matriz de habilidades Plan Táctico de Capacitación Organizacional 	<ul style="list-style-type: none"> Líder talento humano
Gestione el Proyecto de Principio a Fin	Gestión de riesgos y oportunidades	<ul style="list-style-type: none"> Plan de Administración de Riesgos. Análisis de Riesgos (Fuentes de Riesgos, Categorías de Riesgos, Identificación de Riesgos, Evaluación del Impacto, Evaluación de la Probabilidad, Mitigación de Riesgos). PIP (Propuesta de Mejora de procesos) 	<ul style="list-style-type: none"> Project manager Scrum master
	Verificación y validación	<ul style="list-style-type: none"> Incremento Sprint Review. Daily Scrum. Sprint Retrospective Product backlog. Inspecciones. 	<ul style="list-style-type: none"> Product owner Team dev Testers
	Revisiones por pares	<ul style="list-style-type: none"> Mejores Prácticas y Lecciones Aprendidas. Herramientas para pruebas de Integración, Ambientes de Pruebas. 	<ul style="list-style-type: none"> Team dev Testers
	Estimaciones	<ul style="list-style-type: none"> Objetivo del Producto. Objetivo del Sprint. Burn-up-downs 	<ul style="list-style-type: none"> Product owner Project manager Scrum master

	Monitoreo y control	<ul style="list-style-type: none"> • Estimación Inicial • Reporte de Monitoreo del Proyecto. • Herramienta de control de versiones: Repositorio Líneas Base. • Product backlog • Cumulative flows • Documentación de Pruebas (Estrategia, Criterio de Aprobación / Rechazo, Criterios de Suspensión y Reanudación de Pruebas, Tareas de Pruebas) • Registro de Administración de la Configuración, Backups. 	<ul style="list-style-type: none"> • Product owner • Team dev • Testers
	Planificación	<ul style="list-style-type: none"> • Acta de inicio de proyecto • Sprint Planning • Planificación del product backlog • Propuesta Técnica 	<ul style="list-style-type: none"> • Product owner • Project manager • Stakeholders • Team dev
	Gestión de la configuración	<ul style="list-style-type: none"> • Plan de Administración de la Configuración • Resumen de Acciones Correctivas • Solicitud de Cambios a Línea Base • Registro de Administración de la Configuración, Backups • Solicitud de Cambios a Línea Base 	<ul style="list-style-type: none"> • Product owner • Arquitecto de solución QA
Diseño integral del proyecto	Solución técnica	<ul style="list-style-type: none"> • Diseño de Alto Nivel • Diseño Detallado de Producto • Estrategia de Integración • Plan de Pruebas de 	<ul style="list-style-type: none"> • Product owner • Project manager • Arquitecto de solución • Stakeholders

		<p>Integración</p> <ul style="list-style-type: none"> • Selección de Alternativas 	
	Integración del producto	<ul style="list-style-type: none"> • Código Fuente, Estándares de codificación • Manual de Usuario • Manual de instalación • Producto Integrado • Sprint review • Sprint • Incremento 	<ul style="list-style-type: none"> • Scrum master • Product owner • Project manager • Arquitecto de solución • Team dev
	Análisis de decisiones y resolución	<ul style="list-style-type: none"> • Resultados de las pruebas • Documento de inspección a estrategia de integración, Defectos reportados en revisiones e inspecciones • Análisis de Decisiones y Resolución (Identificación de Decisiones que Requieren Análisis Formal: Criterios) • Plan de Análisis Formal de Decisiones • Product backlog • Sprint backlog • Sprint • Objetivo del Sprint 	<ul style="list-style-type: none"> • Scrum master • Product owner • Project manager • Arquitecto de solución • Team dev • QA • Testers

Conclusiones:

Los proyectos de desarrollo de software siempre han sido un reto para las organizaciones y para el caso puntual de este trabajo de investigación con las entidades públicas, es más evidente toda vez que muchas de ellas no utilizan o tienen adoptado una metodología o guía estándar al momento de desarrollar un software, esto se puede deber entre otros factores a que no es su naturaleza el desarrollar software, situación que conlleva a que no se asignen recursos para fortalecer estas áreas de TI encargadas de asumir aquellos proyectos pequeños de su operación diaria y solo aquellas que cuenten con más recursos se decantaran por la tercerización de sus proyectos, en especial aquellos de mayor complejidad.

En el instrumento aplicado a 11 entidades de carácter público o mixto se pudo evidenciar que, aunque indican que llevan sus proyectos de forma ordenada aplicando de forma empírica los principios de desarrollo de software como se muestra en la respuesta donde se observa que más del 50% de los encuestados no utilizan algún framework y las del 80% no utiliza alguna metodología para sus proyectos; aunque se puede pensar que es porque algunas de estas entidades tercerizan sus proyectos, si no se hace seguimiento al proveedor o tercero se puede perder el control del mismo, lo cual podría llevar a que el proyecto no alcance las expectativas y su desarrollo puede inclusive ser caótico.

Finalmente, una vez realizada la labor de correspondencia se pudo evidenciar que scrum deja unos vacíos respecto a las necesidades o lineamientos en el desarrollo de software de acuerdo con las normativas de orden nacional, también, se pudo observar que en correspondencia con el modelo de CMMI en su versión 2.0, scrum no tiene productos de trabajo que abarquen ciertas prácticas dentro de CMMI en nivel 3.

Como trabajos futuros se recomienda realizar un proceso de validación de la guía, estableciendo parámetros y criterios que permitan determinar el grado de cumplimiento en proyectos de desarrollo de software y de mantenimiento de acuerdo con la naturaleza de las entidades públicas en Colombia.

Palabras clave: entidades públicas, Colombia, agilismo, procesos de desarrollo de software, Scrum, CMMI, calidad de software.

SUMMARY

This research paper proposes a marked guide within the principles and values of agile that guides the processes of development of software solutions in public entities in Colombia, seeking harmonization between CMMI 2.0 and the Scrum framework. During the development of the research, it was possible to carry out in the first instance the application of a diagnostic instrument in a group of entities of a public nature, later, a characterization was carried out that allowed the identification of components, roles, artifacts and activities of the guide to be proposed. Finally, the guide called GIDSEO is described.

Background:**1. Introduction:**

Public entities in Colombia have the mission of offering services to their citizens based on the management of administering public resources, so these entities need to implement technological solutions that allow them to operate efficiently and in addition to providing services to citizens. Therefore, the Ministry of Information and Communications Technologies – MINTIC issued in 2017 Decree 1499, which describes the regulation of the Integrated Planning and Management System called the Integrated Planning and Management Model – MIPG that establishes a reference framework that allows directing, planning, executing, monitoring, evaluating and controlling the management of public entities.

In accordance with the above, it is important to highlight the evolution of these programs of the Government of Colombia in the search for transparency, promotion, use and exploitation of information and communication technologies - ICT for the benefit of the state and especially of citizens. Thus, after several years of implementation of the online government strategy in Colombia, it evolved into what is known as Digital Government, the latter being a model that must be adopted by the agencies and entities of the national and territorial orders.

This has required these entities to establish mechanisms to comply with these Digital Government policies, one of them focuses on the development of software solutions or information systems that support the management and day to day of each entity, and that meet the needs both internally and towards its citizens, this has led to the creation of custom software development teams within the different offices and dependencies.

As these entities are not software factories, it means that good practices and methodologies concerning software engineering may not be applied to reduce the failure rate and to maintain quality attributes, the inappropriate use of these software development methodologies has resulted in software products with failures or deficiencies that result in reprocesses or canceled projects, to this is added that each software development team usually applies its own criteria or models of application development as there is no base guide that gives them the guidelines and good practices that allow the entity to give continuity to these projects regardless of whether there is staff rotation or change of administrations.

Taking into account this situation and the impact it generates on the efficiency and effectiveness of software projects, it is important to provide support to all application development teams so that the entity can count on quality software solutions that can meet their needs, without defining a formal methodology but that allows them to comply with the requirements of the control entities in terms

of carrying out orderly and documented form each of the processes. Due to this, a guide is proposed that establishes practices in the development of software for public entities while preserving elements of good practices and agility.

2. Problem statement:

Currently the software projects executed by the development teams of public entities in Colombia present a series of difficulties that have influenced the success or failure of the projects, in the list of these problems we have: the software does not meet expectations, lack of clear and standardized procedures to manage the quality of the software that is produced and software projects that do not add value to the entity. Now, if the aforementioned problems are addressed in a little more detail, we could more clearly show these shortcomings:

- Software does not meet expectations
- Lack of clear and standardized procedures to manage the quality of the software that is produced
- Projects that do not add value to the entity

3. Problem formulation:

According to the definition of the problem described, the following question arises:

Is there a guide to develop software applying quality model and agile project management for public entities in Colombia?

Objectives:

1. General objective

Propose an agile guide for software development in public entities in Colombia using CMMI 2.0 and SCRUM.

2. Specific objectives

- Analyze through a diagnostic instrument the current state of software development in public entities in Colombia.
- Relate each of the activities and practices of CMMI 2.0, Scrum and digital government guidelines, establishing a correspondence between each of them
- Identify the components of software quality and the management of the development process in software solutions, which will make up the guide.

Materials and Methods:

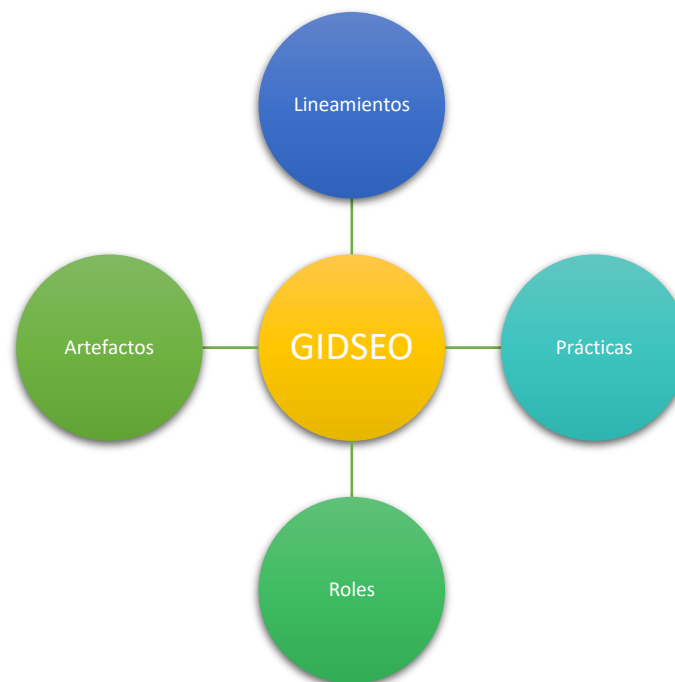
This research work will basically focus on a type of descriptive research, since from the findings found in the documentary review and diagnosis, a guide for the development of software in public entities in Colombia will be proposed, to achieve this the following three (3) methodological phases will be covered:

- **Analysis phase:** in this phase a diagnosis will be made from the application of an instrument that allows to determine the way in which software development projects are managed in entities of a public nature in Colombia.
- **Correspondence phase:** in this phase the chosen quality model is characterized, which in this case will be CMMI 2.0, specifically for development, and the Scrum framework in its 2020 version. After this process, we go on to define each of the components and relationships between them, indicating roles, artifacts, activities, stages, practices that will be part of the guide to be proposed.
- **Definition phase:** in this phase we seek to establish a maturity model that gives guidelines to the implementation of the guide to be proposed

Results:

Practical guide for software development in official entities "GIDSEO".

GIDSEO is a guide for software development that harmonizes three components which are: software quality, project management through National policies and software development framework framed in agile. However, GIDSEO is aimed at national or territorial public entities in Colombia. Going into the guide we have that it is composed of four (4) elements which are:



Each of the elements of GIDSEO is described below, pointing out the corresponding practices, guidelines, artifacts and roles.

GIDSEO			
Guideline	Practice	Artifacts	Roles
Ensure the Generation of Value for the Entity and Users	Requirements development and management	<ul style="list-style-type: none"> • User Stories • Previous studies • Objective of the Sprint • Sprint Planning 	<ul style="list-style-type: none"> • Product owner • Stakeholders • QA Solution Architect
	Process quality assurance	<ul style="list-style-type: none"> • Definition of Finished • Sprint Retrospective 	<ul style="list-style-type: none"> • Scrum master • Team dev
	Organizational training	<ul style="list-style-type: none"> • Skills Matrix • Tactical Organizational Training Plan 	<ul style="list-style-type: none"> • Human talent leader
Manage the Project from Start to Finish	Risk and opportunity management	<ul style="list-style-type: none"> • Risk Management Plan. • Risk Analysis (Risk Sources, Risk Categories, Risk Identification, Impact Assessment, Probability Assessment, Risk Mitigation). • PIP (Process Improvement Proposal) 	<ul style="list-style-type: none"> • Project manager • Scrum master
	Verification and validation	<ul style="list-style-type: none"> • Increment • Sprint Review. • Daily Scrum. • Sprint Retrospective • Product backlog. • Inspections. 	<ul style="list-style-type: none"> • Product owner • Team dev • Testers
	Peer reviews	<ul style="list-style-type: none"> • Best Practices and Lessons Learned. • Tools for Integration Testing, Test Environments. 	<ul style="list-style-type: none"> • Team dev • Testers
	Estimates	<ul style="list-style-type: none"> • Product Objective. • Objective of the Sprint. • Burn-up-downs • Initial Estimate 	<ul style="list-style-type: none"> • Product owner • Project manager • Scrum master
	Monitoring and control	<ul style="list-style-type: none"> • Project Monitoring Report. 	<ul style="list-style-type: none"> • Product owner • Team dev

		<ul style="list-style-type: none"> • Version control tool: Baselines Repository. • Product backlog • Cumulative flows • Test Documentation (Strategy, Approval/Rejection Criteria, Test Suspension and Resumption Criteria, Testing Tasks) • Configuration Management Log, Backups. 	<ul style="list-style-type: none"> • Testers
	Planning	<ul style="list-style-type: none"> • Project initiation act • Sprint Planning • Product backlog planning • Technical Proposal 	<ul style="list-style-type: none"> • Product owner • Project manager • Stakeholders • Team dev
	Configuration management	<ul style="list-style-type: none"> • Configuration Management Plan • Summary of Corrective Actions • Requesting Changes to Baseline • Configuration Management Log, Backups • Requesting Changes to Baseline 	<ul style="list-style-type: none"> • Product owner • QA Solution Architect
Comprehensive project design	Technical solution	<ul style="list-style-type: none"> • High Level Design • Detailed Product Design • Integration Strategy • Integration Test Plan • Selection of Alternatives 	<ul style="list-style-type: none"> • Product owner • Project manager • Solution Architect • Stakeholders
	Product Integration	<ul style="list-style-type: none"> • Source Code, Coding Standards • User Manual Installation Manual • Integrated Product • Sprint review • Sprint • Increment 	<ul style="list-style-type: none"> • Scrum master • Product owner • Project manager • Solution Architect • Team dev
	Decision analysis	<ul style="list-style-type: none"> • Test results 	<ul style="list-style-type: none"> • Scrum master

	and resolution	<ul style="list-style-type: none"> • Inspection document to integration strategy, Defects reported in reviews and inspections • Decision Analysis and Resolution (Identification of Decisions That Require Formal Analysis: Criteria) • Formal Decision Analysis Plan • Product backlog Sprint backlog Sprint Sprint Objective 	<ul style="list-style-type: none"> • Product owner • Project manager • Solution Architect • Team dev • QA • Testers
--	----------------	--	---

Conclusions:

Software development projects have always been a challenge for organizations and for the specific case of this research work with public entities, it is more evident since many of them do not use or have adopted a standard methodology or guide when developing software, this may be due among other factors to the fact that it is not their nature to develop software, situation that leads to not allocating resources to strengthen these areas of IT responsible for assuming those small projects of their daily operation and only those that have more resources will opt for the outsourcing of their projects, especially those of greater complexity.

In the instrument applied to 11 entities of a public or mixed nature, it was possible to show that, although they indicate that they carry out their projects in an orderly manner by empirically applying the principles of software development as shown in the response where it is observed that more than 50% of the respondents do not use any framework and those of the 80% do not use any methodology for their projects; although it can be thought that it is because some of these entities outsource their projects, if the supplier or third party is not monitored, control of it can be lost, which could lead to the project not reaching expectations and its development can even be chaotic.

Finally, once the correspondence work was carried out, it was possible to show that scrum leaves gaps regarding the needs or guidelines in the development of software in accordance with national regulations, also, it could be observed that in correspondence with the CMMI model in its version 2.0, scrum does not have work products that cover certain practices within CMMI at level 3.

As future work, it is recommended to carry out a validation process of the guide, establishing parameters and criteria that allow determining the degree of compliance in software development and maintenance projects according to the nature of public entities in Colombia.

Keywords: public entities, Colombia, agile, software development processes, Scrum, CMMI, software quality.

REFERENCIAS

1. Ciclo de vida del software: todo lo que necesitas saber. (28 de Noviembre de 2020). Obtenido de <https://intelequia.com/blog/post/2083/ciclo-de-vida-del-software-todo-lo-que-necesitas-saber>
2. Decreto 1083,
<http://www.funcionpublica.gov.co/sisjur/home/Norma1.jsp?i=62866> (2015:).
3. DIGITAL.AI. (2021). The 15th State of Agile Report is copyrighted by Digital.ai. All rights reserved.
4. Jurado, C. L. (29 de 01 de 2021). CCM. Obtenido de <https://es.ccm.net/contents/223-ciclo-de-vida-del-software>
5. Ken Schwaber, J. S. (2020). La Guía de Scrum. En J. S. Ken Schwaber, La Guía de Scrum (pág. 4).
6. MIPG. (2022). Función Pública. Obtenido de <https://www.funcionpublica.gov.co/web/mipg>
7. OECD. (21 de Mayo de 2018). <https://www.oecd-ilibrary.org>. Obtenido de <https://www.oecd-ilibrary.org>: <http://dx.doi.org/10.1787/9789264292147-es>
8. SEI CMMI - Niveles de Madurez. (s.f.). Obtenido de [tutorialspoint.com](https://www.tutorialspoint.com/es/cmmi/cmmi_maturity_levels.htm): https://www.tutorialspoint.com/es/cmmi/cmmi_maturity_levels.htm
9. TIC., M. (s.f.). Guía de buenas practicas en la adquisición de software y servicios asociados – Colombia Compra Eficiente. Obtenido de https://colombiacompra.gov.co/sites/cce_public/files/cce_documents/guia_de_buenas_practicas_software_ver1_logo01.pdf
10. Valores Scrum: cómo ponerlos en práctica. (s.f.). Obtenido de <https://www.scrumio.com>: <https://www.scrumio.com/blog/valores-scrum-como-ponerlos-en-practica/>