

LISTA DE FIGURAS

<i>Figure 1 Ciclo de vida DevOps</i>	<i>22</i>
<i>Figure 2 Fases de DevOps,</i>	<i>23</i>
<i>Figure 3 DevOps en el ciclo de vida del desarrollo de software</i>	<i>24</i>
<i>Figure 4 Herramientas por fase</i>	<i>24</i>
<i>Figure 5 Tabla periódica de DevOps</i>	<i>25</i>
<i>Figure 6 Modelo CALM para DevOps</i>	<i>28</i>



INTRODUCCIÓN

El desarrollo de software es una disciplina que vive en un constante cambio y evolución apalancado por una avalancha de nuevos productos y diferentes formas de gestionar el ciclo de vida del desarrollo y entrega de productos de software que brinda la industria mundial. La adopción de nuevas tecnologías y metodologías en las compañías son actividades que no se deben tomar a la ligera y que deberían partir de una autoevaluación de cómo se están haciendo las cosas a nivel de procesos y metodologías internas para argumentar la decisión. Este tipo de análisis se debe realizar a partir de evaluar las necesidades y condiciones particulares de cada organización para así poder identificar la conveniencia o no de adoptar nuevos estándares de industria y no solo llevarse por modas o tendencias mundiales.

DevOps se ha posicionado en los últimos años como una práctica para la estandarización de actividades del desarrollo de software y su automatización a fin de lograr la entrega continua de productos. A la fecha, DevOps ha sido difundida y extendida en el mercado mundial por grandes corporaciones dedicadas al desarrollo de software como por ejemplo Amazon, Netflix, Facebook, HP, entre otras.

Esta investigación busca evaluar la conveniencia de adoptar un conjunto de buenas prácticas y cambios culturales, enmarcados en la cultura DevOps, en el área de sistemas de una empresa cuyo propósito principal no es producir software comercial a ritmos acelerados, sino ofrecer el soporte operativo de un centro de investigación en agricultura como lo es la Alianza Bioversity International y el Centro Internacional de Agricultura Tropical (de ahora en adelante CIAT).

El primer paso de esta propuesta es la definición de criterios claves para evaluar la conveniencia de adoptar DevOps como resultado de analizar y caracterizar las particularidades propias de CIAT, como ejemplo el tipo de empresa y su propósito versus las recomendaciones que ofrece DevOps para entrega de productos de software, y así determinar el nivel de madurez de este tipo de procesos en la compañía.

Una vez se obtengan los resultados del análisis se podrá determinar la conveniencia de adoptar DevOps basados en criterios de evaluación sólidos o precisar qué es lo que realmente necesita la compañía dada sus particularidades, por último, se definirá un plan de acción acorde a lo identificado para que la compañía adopte para mejorar los procesos de entrega de productos de software y así pueda para brindar soluciones rápidas y confiables a sus clientes internos.

1. DEFINICIÓN DEL PROBLEMA

1.1. PLANTEAMIENTO DEL PROBLEMA

El CIAT es un centro de investigación en agricultura con presencia mundial que se creó para enfrentar las crisis de cambio climático, pérdida de la biodiversidad, degradación ambiental y malnutrición, y para maximizar las oportunidades combinando nuestras fortalezas en puntos claves en el sistema alimentario. Es parte del CGIAR, el mayor consorcio mundial en investigación e innovación agrícola para un futuro sin hambre, dedicado a reducir la pobreza, contribuir a la seguridad alimentaria y nutricional y mejorar los recursos naturales. La alianza tiene presencia a nivel mundial representada en 4 sedes regionales donde la sede Regional para Las Américas – está ubicada en la ciudad de Palmira, Colombia.

Al ser un centro de investigación científica, el área informática no es el eje central del propósito misional corporativo de la organización, por lo que todos los productos de software desarrollados en la institución son para uso interno dentro de la sede regional y para las demás oficinas regionales alrededor del mundo. A nivel de stack tecnológico la empresa cuenta con un ERP que cubre procesos administrativos y financieros institucionales, complementado con un grupo de aplicaciones a la medida desarrolladas en casa denominadas aplicaciones periféricas. El departamento de informática está centrado en el aseguramiento de la operación y soporte institucional por lo que el factor monetario no es el eje central para inicio de un proyecto de software, trasladando la responsabilidad de decisión a la media y alta gerencia del departamento. Para los casos de proyectos cortos y medianos no mayores a 6 meses, normalmente son cubiertos con el equipo de desarrollo de software de planta y se priorizan de acuerdo a la disponibilidad del recurso humano disponible.

En los últimos años, el área de desarrollo se ha ido desalineando y perdiendo rigurosidad en el uso de directrices y metodologías definidas para gestionar el ciclo de vida de un producto de software particularmente en procesos de calidad y documentación de los proyectos, trasladando la toma de decisión a seguir el juicio las personas involucradas en las actividades. En cuanto a pruebas y calidad de software las tareas se hacen a criterio y experticia del desarrollador y se complementan con validaciones a nivel operativo realizadas por usuarios finales funcionales clave, normalmente sin criterios de evaluación bien definidos por lo que las evidencias no quedan documentadas de la mejor manera. A nivel de infraestructura se cuenta con servidores que cubren ambientes de desarrollo, pruebas y producción. El proceso de despliegue de soluciones de software a cada uno de los ambientes es completamente manual en todo su proceso, desde la creación del empaquetado de producto terminado (war para las aplicaciones java) hasta el

despliegue en dichos servidores, solo para el caso del ambiente de producción el área de redes y operaciones se encarga del despliegue.

Desde inicios del 2020 el área de desarrollo y soporte administrativo se reestructuró en una nueva área llamada integración de sistemas con una nueva administración y el propósito de darle orden a los procesos asociados al ciclo de vida del desarrollo de software, razón por la cual a finales del 2020 en conjunto con las áreas de redes y desarrollo de negocio, se materializó un primer acercamiento al concepto DevOps donde a través de un modelo de consultoría se instalaron un conjunto de herramientas propias de DevOps como un orquestador de tareas, un gestor de dependencias y un analizador de código estático pero el proceso solo se limitó al componente técnico en el uso de las herramientas pero no se acompañó de un plan de adopción ni mucho menos de reestructuración de los procedimientos internos . En la actualidad y 6 meses después, no se le está dando el debido uso extendido y cotidiano al grupo de herramientas instaladas, influidos por factores como falta de definición de procesos, librerías de código fuente de proyectos gestionados manualmente, etc.

Esta propuesta busca realizar un diagnóstico inicial usando un marco de evaluación definido que ayudará a determinar el nivel de madurez de CIAT versus lo que ofrece DevOps con respecto a las fases del ciclo de vida del desarrollo de software en las áreas de integración de sistemas y redes (operaciones). Una vez se obtengan los resultados del análisis se determinará la conveniencia de adoptar practicas DevOps o precisar qué es lo que realmente necesita la compañía dado su tipo, propósito y particularidades. Una vez identificadas las necesidades puntuales se definirá un plan de acción para que la compañía adopte las recomendaciones en los procesos del ciclo de vida de desarrollo de software enfocado en buenas prácticas de codificación y automatización para brindar soluciones rápidas y confiables a la organización.

1.1. FORMULACIÓN DEL PROBLEMA

A partir de las condiciones operaciones actuales y las oportunidades de mejora que puede brindar DevOps para mejor el proceso de entrega continua de soluciones de software en el área de integración de sistemas del CIAT.

¿Cómo evaluar si DevOps es lo que realmente requiere CIAT para iniciar un proceso de transformación de sus procesos de entrega continua de productos de software?

¿Cómo definir un marco de evaluación para determinar estado actual de los procesos de entrega continua de productos de software del CIAT frente a las recomendaciones DevOps?

¿Cómo diagnosticar los procesos de entrega de software versus las buenas prácticas propuestas por DevOps acorde al tipo de empresa que es CIAT?

¿Cómo planear la adopción progresiva de DevOps en los procesos de entrega continua de productos de software acorde a las necesidades puntuales de CIAT?

2. OBJETIVOS DEL PROYECTO

2.1 OBJETIVO GENERAL

Evaluar la conveniencia de adoptar DevOps como estrategia para la entrega continúa de productos de software de la Alianza Bioversity - Centro Internacional de Agricultura Tropical – CIAT – Sede Regional para las Américas - Palmira, Colombia

2.2 OBJETIVOS ESPECÍFICOS

1. Definir los factores a considerar para evaluar la conveniencia de adoptar DevOps como estrategia de entrega continua de productos en la organización.
2. Proponer un artefacto para evaluar la conveniencia de adoptar DevOps en el CIAT a partir de los factores identificados.
3. Diagnosticar el estado de los procesos de entrega de software en el CIAT con el instrumento propuesto para determinar si DevOps es una solución adecuada a las necesidades del CIAT.
4. Proponer un plan de acción a partir del diagnóstico obtenido.

2.3 RESULTADOS ESPERADOS

- Artefacto para diagnosticar el nivel de madurez de las prácticas actuales en la entrega de productos software en el CIAT versus las buenas prácticas que ofrece DevOps
- Resultado de la evaluación de las necesidades de CIAT versus DevOps
- Plan de acción a partir de los resultados del análisis de las necesidades particulares en la entrega de productos de software en el CIAT
- Una guía de recomendaciones para mejorar el proceso de entrega de productos de software en el CIAT en caso de que la conclusión del diagnóstico sea NO usar DevOps.

3. ALCANCE

Este trabajo se centrará en definir criterios de evaluación para construir un artefacto que permita realizar un diagnóstico del estado de madurez de los procesos de entrega de productos de software por parte del área de integración de sistemas de CIAT versus las recomendaciones y buenas prácticas que ofrece DevOps para la entrega continua de productos de software.

A partir del resultado del diagnóstico se podrá determinar si las recomendaciones y buenas prácticas DevOps son convenientes para atender las necesidades de CIAT dado sus particularidades y tipo de compañía. Por último, se espera generar un plan de acción a partir de los hallazgos encontrados para que la empresa pueda iniciar una adopción progresiva.

En el caso de que el resultado del análisis diagnóstico concluya que DevOps no es lo que realmente necesita la compañía, se generará una guía de recomendaciones para potenciar el proceso actual de entrega de productos de software.

4. JUSTIFICACIÓN

El CIAT es un centro de investigación científica en agricultura, por esta razón el rol informático no es el eje central de su funcionamiento. Sin embargo, como en toda organización la computación juega un papel clave para ayudar a la organización a cumplir sus objetivos misionales.

Este proyecto busca evaluar en qué medida DevOps, entendido como un conjunto de buenas prácticas para coordinar operación y desarrollo, es conveniente para una empresa cuyo propósito principal no es producir software comercial, sino usar el desarrollo de sistemas de información para dar soporte operacional y estratégico que garantice la operación administrativa.

El tipo de empresa y la particularidad de la compañía en no ser una casa de desarrollo de software y que la mayoría de los nuevos proyectos son cortos (no sobrepasan los 6 meses de desarrollo) hacen que la adopción de estrategias de entrega continua de productos de software mediante DevOps no pierda su foco de crear productos de valor a nuestros clientes internos, antes, por el contrario, potencia la calidad de los servicios ofrecidos incrementando la eficiencia de los recursos IT.

El principal reto de este documento es definir factores claves para determinar el nivel de conveniencia de adoptar DevOps como estrategia para la entrega continua de productos de software y como la empresa puede hacer un proceso de adopción progresivo de acuerdo a sus particularidades operacionales.

Por otro lado, es importante aprovechar y potenciar un grupo de herramientas DevOps que ya se tienen disponibles para su uso en el CIAT, a las cuales no se están aprovechando y en algunos casos no se usan.

5. MARCO TEÓRICO DE REFERENCIA Y ANTECEDENTES

5.1. MARCO TEÓRICO

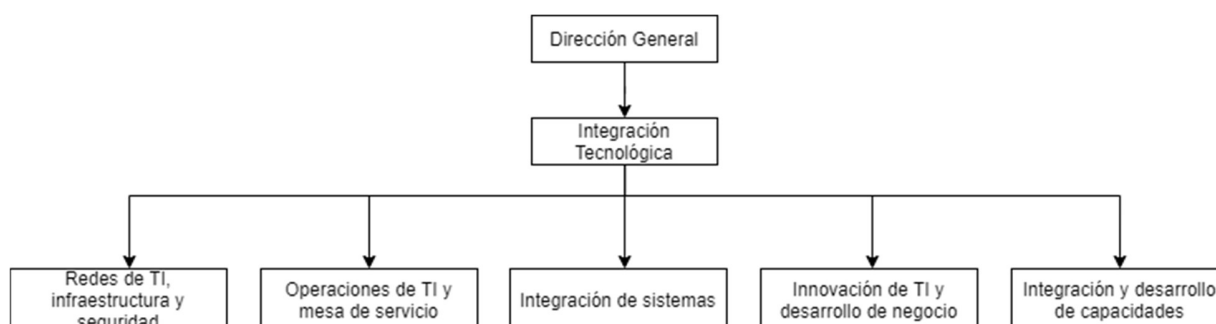
5.1.1. Alianza Bioveristy - CIAT

La Alianza entre Bioversity International y el Centro Internacional de Agricultura Tropical, en adelante CIAT, es unión de fuerzas entre dos centros de investigación que en conjunto suman más de 100 años de experiencia aplicando ciencia para abordar retos mundiales. Esta alianza se creó para enfrentar las crisis de cambio climático, pérdida de la biodiversidad, degradación ambiental y malnutrición, y para maximizar las oportunidades combinando nuestras fortalezas en puntos claves en el sistema alimentario. La alianza es parte del CGIAR, el mayor consorcio mundial en investigación e innovación agrícola para un futuro sin hambre, dedicado a reducir la pobreza, contribuir a la seguridad alimentaria y nutricional y mejorar los recursos naturales.

La alianza oficializó su operación conjunta en el 2020 con presencia a nivel mundial representada en 4 sedes distribuidas en 4 continentes:

- Oficina Principal- Roma, Italia
- Sede Regional para las Américas - Palmira, Colombia
- Sede Regional para África - Nairobi, Kenia
- Sede Regional para Asia - Hanoi, Vietnam

El área de integración tecnológica es la encargada de apoyar la operación de la organización a través de servicios tecnológicos a nivel de global y está dividida en 5 subáreas tal y como se muestra a continuación:



El departamento de integración tecnológica cubre todo el soporte operativo de la organización a nivel global, a nivel de desarrollo de software las áreas involucradas son:

- Integración de sistemas, responsable por el funcionamiento y operación de las soluciones y servicios relacionados con la operación de la compañía. El área está compuesta por 10 personas de los cuales 6 están directamente relacionados con desarrollo de software así: 3 desarrolladores, 1 administrador de bases de datos, 1 jefe de desarrollo y el gerente del área, el resto del área tienen dedicación exclusiva a temas del ERP.
- Redes de TI, infraestructura y seguridad, encargados de la gestión de la infraestructura física de red y operación de los servidores y servicios informáticos de la compañía. De esta hay 3 personas directamente involucrados con desarrollo de software, en tareas como administración de servidores, seguridad de red y servicios y la gerencia del área.
- Innovación de TI y desarrollo de negocio, es otra área donde se hace desarrollo de software, focalizados en soluciones de gestión para programas de investigación a nivel del consorcio CGIAR, el área está compuesta de 10 personas de los cuales 6 están dedicados a temas asociados a desarrollo de software.

A nivel de infraestructura la compañía cuenta con un centro de datos propio administrado por el área de redes de TI, allí están alojados mediante servidores virtualizados los diferentes ambientes de servidores y servicios utilizados en las diferentes etapas del desarrollo de software.

Explicar aquí el organigrama más en detalle sobre el CIAT. Explicar cuántas personas tiene el área de sistemas de integración y cuántos se encargan de la parte del desarrollo.

5.1.2. DevOps

(Carvalho, Moreira, Rosso, Ibiri 2017) en [1] resume la definición de DevOps y su historia:

El concepto DevOps empezó a fraguarse entre el 2007 y el 2008, “En este año, empezaron a utilizar el término Infraestructura ágil en algunos foros de debates con foco en el desarrollo ágil, y en la misma época durante el evento Agile 2008, surgieron charlas que abordaban el tema metodología ágil para la administración de infraestructura, inspirada en el modelo ágil de desarrollo”.

Historia:

El término DevOps surgió de la casualidad. Aparece por primera vez en 2009 cuando Patrick Debois organiza en la ciudad de Gante, Bélgica, un foro técnico donde se dieron cita profesionales del sector. Se dice que durante una de las charlas Patrick escribió en una pizarra DevOps como un

acrónimo de “Develop and Operations” (Desarrollo y operaciones). El término gustó y comenzó a ser usado de forma masiva desde entonces.

Consolidando las definiciones de Smith, Safari 2020 en [2] y de Martín M. C 2018 en [3] se genera la siguiente definición:

Definición:

Se entiende DevOps como el conjunto de prácticas que automatizan los procesos entre los equipos de desarrollo de software y operaciones TI; para que las organizaciones puedan compilar, probar y publicar software con mayor rapidez y fiabilidad, basándose en una cultura de colaboración entre equipos que, tradicionalmente, venían trabajando en grupos aislados.

DevOps pone un gran énfasis en las responsabilidades compartidas entre todos los equipos a lo largo del ciclo de vida del desarrollo de software. Los límites de las funciones de trabajo se suavizan, ya que los miembros del equipo de operaciones asumen tareas que tradicionalmente estaban más enfocadas a los desarrolladores, y los miembros del equipo de desarrollo hacen lo mismo.

A medida que se amplificaba la conversación sobre DevOps, iba saliendo a la luz la realidad sobre el modelo de desarrollo de software imperante hasta el momento: equipos o grupos de trabajo aislados y solo preocupados por su parte del proyecto, líneas de comunicación inexistentes, largas jornadas de trabajo, errores sin corregir, y clientes insatisfechos. Y así se empezó a expandir la palabra de la “cultura DevOps” entre profesionales y organizaciones conscientes de que en un entorno altamente competitivo como es el desarrollo de software era necesario unir desarrollo y operaciones.

DevOps nació como un nuevo paradigma en contraposición al modelo de desarrollo de software tradicional con el objetivo de mejorar la comunicación y la colaboración entre los diferentes equipos de trabajo y aumentar así su eficiencia. Se trata de una nueva forma de trabajo, un nuevo enfoque para crear software que hace hincapié en acortar distancias en la relación entre los equipos de desarrollo, sistemas/operaciones y Testing/QA. DevOps se basa por tanto en la idea ágil de que las personas y sus interacciones están por encima de herramientas y procesos.

Como metodología de desarrollo ágil, DevOps ofrece una serie de técnicas y prácticas de automatización y monitorización de procesos en cada fase del ciclo de vida de un producto digital: Planificación Programación (código) Construcción Pruebas Entrega Despliegue Operación Monitorización Estas fases no están estrictamente delimitadas como en un ciclo en cascada, sino que DevOps se basa en un flujo iterativo en el que pueden solaparse diferentes procesos.

Este ciclo de vida no es un proceso con un ciclo final, es algo iterativo que siempre debe empezarse desde el primer paso para mejorar y organizar la manera en llevar nuevos features a producción de la manera más rápida y eficiente posible tratando de evitar bugs o fallos de seguridad.

Ciclo de vida:

El ciclo de vida iterativo en DevOps consta de los siguientes etapas o fases (ver Imagen tomada de <https://tech.tribalyte.eu/blog-introduccion-devops>

Figure 1):

1. **Plan:** Sin importar la metodología que a utilizar como Waterfall o Agile, en esta primera etapa se definen las labores del equipo, los requerimientos necesarios a implementar en la plataforma o producto
2. **Code:** En esta fase se empieza a escribir el código necesario para resolver los problemas que plantearon en el paso de plan. Todo este código deber estar almacenado en un repositorio de código fuente centralizado
3. **Build:** En esta fase, el desarrollo de software se lleva a cabo constantemente. dividiendo, todo el proceso de desarrollo en pequeños ciclos. Esto beneficia al equipo de DevOps para acelerar el desarrollo de software y el proceso de entrega.
4. **Test:** El equipo de QA usa herramientas para identificar y corregir errores en el nuevo código de manera continua.
5. **Release:** En esta fase, la nueva funcionalidad se integra con el código existente y se llevan a cabo las pruebas. El desarrollo continuo sólo es posible debido a la integración y las pruebas continuas.
6. **Deploy:** En esta etapa, el proceso de implementación tiene lugar continuamente. Se realiza de tal manera que cualquier cambio realizado en cualquier momento en el código, no debe afectar el funcionamiento del sitio web de alto tráfico.
7. **Operate:** En esta fase, el equipo de operación se encargará del comportamiento inadecuado del sistema o de los errores que se encuentran en la producción trabajando siempre sobre un sistema sólido y estable.
8. **Monitor:** Esta última etapa de un proceso DevOps, es una fase permanente y que se aplica a todo el ciclo completo. Aquí el equipo de operación se encargará de definir las medidas para monitorizar y controlar el estado de salud de las aplicaciones y su infraestructura.



Imagen tomada de <https://tech.tribalyte.eu/blog-introduccion-devops>

Figure 1 Ciclo de vida DevOps

Fases DevOps

Las fases DevOps, son solo la clasificación de una o más etapas de ciclo de vida en bloques con un propósito definido. En las figuras Figure 2 y Figure 3 se muestran las diferentes fases que involucran el ciclo de vida DevOps.

- **Continuous Development:**

En esta fase, se planifica el ciclo de vida del desarrollo de software, se codifica la solución con buenas prácticas de código fuente y se almacena en un repositorio de control de versiones. El código se extrae para realizar una integración continua para crear una compilación y probarla en el servidor de pruebas y, finalmente, desplegarla en la producción.

- **CT (Continuos Testing):**

Esta fase se encarga de probar todos los cambios a través de pruebas automatizadas para evaluar la calidad del software en cada paso del proceso de entrega continua mediante pruebas tempranas y frecuentes.

- **CI (Continuous Integration):**

Es una práctica de desarrollo de software mediante la cual los desarrolladores combinan los cambios en el código en un repositorio central de forma periódica, tras lo cual se ejecutan versiones y pruebas automáticas. Los objetivos clave de la integración continua consisten en encontrar y arreglar errores con mayor rapidez, mejorar la calidad del software y reducir el

tiempo que se tarda en validar y publicar nuevas actualizaciones de software.

- **CD (Continuos Deployment):**

En esta fase se compilan, prueban y preparan automáticamente los cambios en el código y se entregan a la fase de producción. Amplía la integración continua al implementar todos los cambios en el código en un entorno de pruebas o de producción después de la fase de creación.

Cuando la entrega continua se implementa de manera adecuada, los desarrolladores dispondrán siempre de un artefacto listo para su implementación que se ha sometido a un proceso de pruebas estandarizado.

- **Continuos Monitoring:**

Esta fase de encarga de supervisar e identificar los problemas de cumplimiento y los riesgos de seguridad a lo largo de cada fase de los ciclos de vida de DevOps y de las operaciones de TI. El monitoreo continuo puede considerarse como el paso final del proceso de DevOps. Es uno de los pasos más cruciales del ciclo de vida de DevOps y ayudará a conseguir una verdadera eficiencia y escalabilidad.

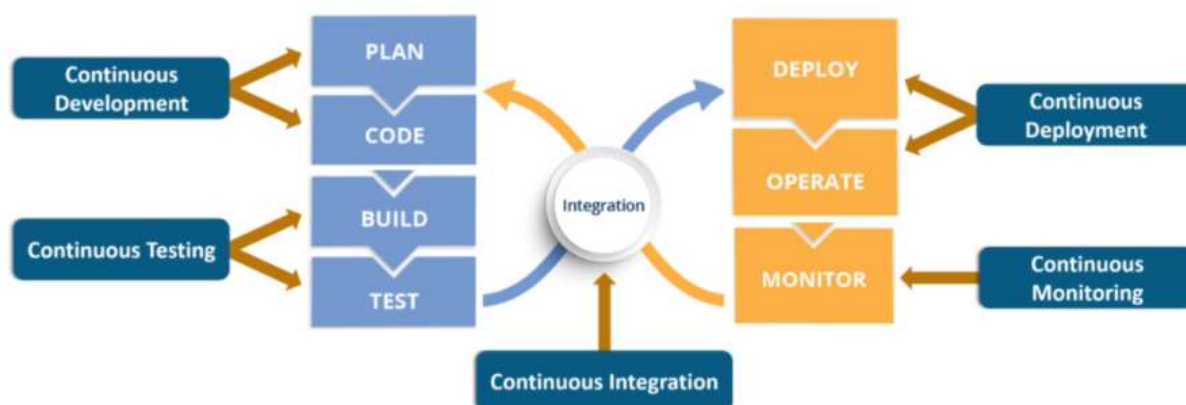


Imagen tomada de <https://www.edureka.co/blog/DevOps-tools>

Figure 2 Fases de DevOps,

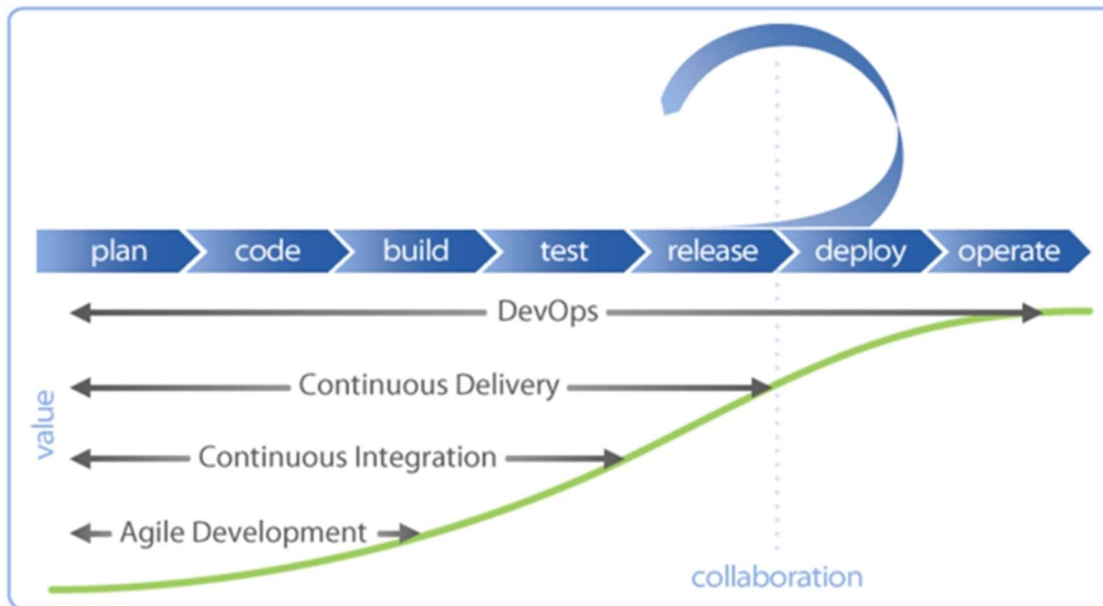


Imagen tomada de DevOps - Introducción y Fundamentos no técnicos (<https://www.udemy.com/course/devops-en-espanol>)

Figure 3 DevOps en el ciclo de vida del desarrollo de software

El gran reto de practicar DevOps radica en alinear los objetivos de los diversos equipos que trabajan en un mismo proyecto de manera que se genere el caldo de cultivo necesario para poder lanzar aplicaciones confiables de manera rápida y frecuente.

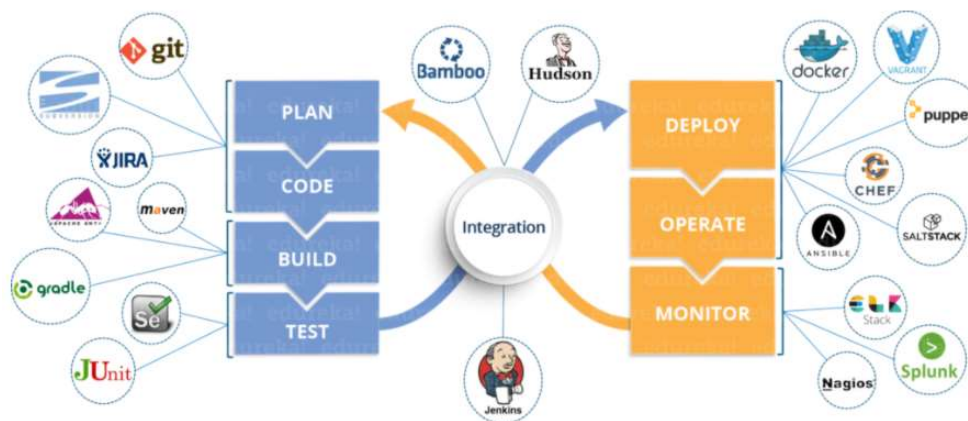


Imagen tomada de <https://www.edureka.co/blog/DevOps-tools>

Figure 4 Herramientas por fase

La empresa digital AI, en conjunto con de un amplio grupo de profesionales crearon un recurso de referencia para identificar las mejores herramientas en el ciclo de vida de la entrega de

software o ecosistema DevOps al que llamaron tabla periódica de DevOps (ver Figure 5).

Además, permite tener una visión general de cómo se integran todas estas herramientas en un entorno, desde el repositorio donde se aloja el código, pasando por la automatización en la provisión de infraestructura, orquestación de servicios y hasta la monitorización del entorno.

The Periodic Table of DevOps Tools (V4.2)

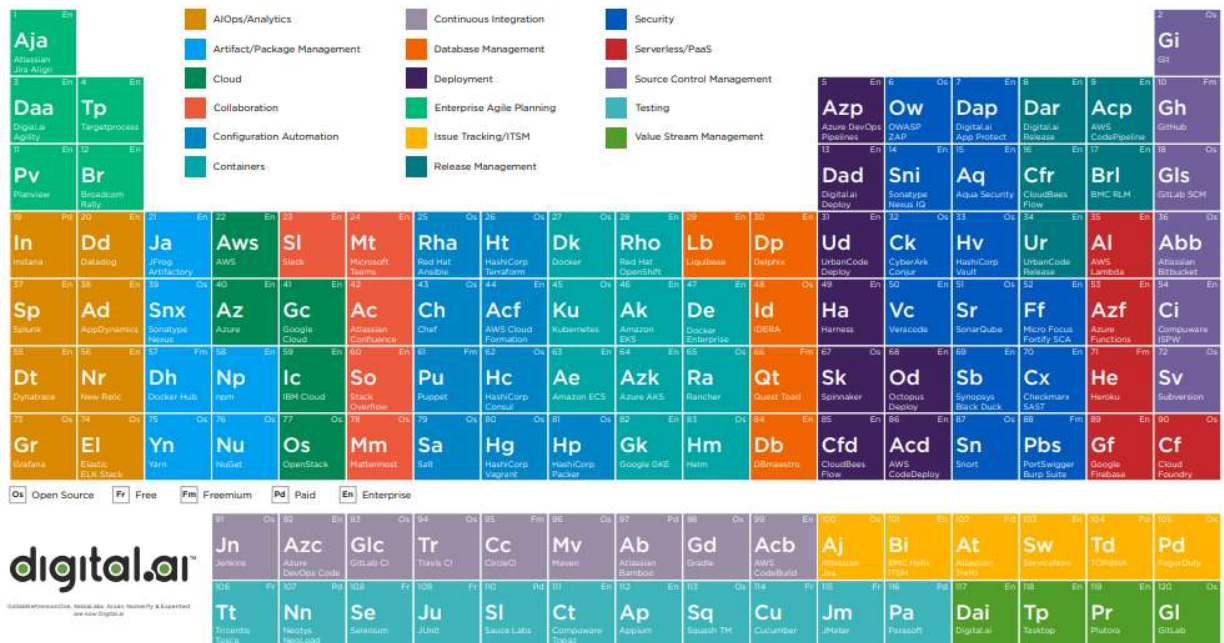


Figure 5 Tabla periódica de DevOps

A continuación, se presenta un resumen a alto nivel de Soto 2019 [4] y Smith 2020 [2], donde se dan definiciones para identificar que no es DevOps:

¿Qué no es DevOps?

No pocas personas tienen falsas creencias acerca de este término. Quizá motivadas porque, aunque su filosofía y bases son claras, no es tanto su contexto de aplicación.

Se tiende a pensar que es un término relacionado con empresas tecnológicas en exclusiva, como Netflix, Facebook, LinkedIn o Google. Si bien estas empresas fueron pioneras, los principios de DevOps son abiertos y quizá sin saberlo, tu empresa ya aplica alguna de sus premisas, como por ejemplo Agile o Lean.

Del mismo modo se tiende a pensar equivocadamente que todos los miembros de los equipos tengan que aprender de todo (desarrollo software, testing, operaciones, etc.). En realidad, esto es completamente falso. DevOps no dice que los equipos de operaciones tengan que saber programar ni viceversa, si bien es cierto que ambos hacen uso de scripts para administrar entornos

o realizan tareas con elementos software, en realidad lo que se pretende es que cada área tenga una noción de las funciones y tecnologías que usan el resto. De este modo, todos pueden entender y participar de manera activa en el desarrollo de un proyecto (control de versiones, rastreo de errores, etc.) y de esta manera contribuir al ciclo de vida del software sobre el que trabajan y aportar valor al negocio.

No existe un rol de DevOps 'per se'. Si se encuentra un currículum que diga que alguien es ingeniero en DevOps se debe desconfiar. DevOps es cultura, herramientas y buenas prácticas para hacer una organización, una organización altamente colaborativa entre equipos a distintos niveles para entregar productos de valor al cliente.

No existe una biblia o manual DevOps, que contenga las instrucciones necesarias para realizar dicha transformación. DevOps se basa en las experiencias positivas y negativas que van acumulando los profesionales relacionados con la transformación digital, y por tanto se encuentra en constante evolución tanto tecnológicamente, a través de sus herramientas, como en cuanto al resto de elementos que lo componen. Y debe adaptarse siempre a la situación inicial de la empresa donde quiera aplicarse.

Principales beneficios de una adopción DevOps:

- Time to market,
- Cerrar la brecha entre Dev y Ops,
- Calidad del producto y satisfacción mejorada del cliente.

Además, se identificaron desafíos, entre ellos:

- Resistencia al cambio
- Falta de comprensión del significado de DevOps
- Falta de colaboración entre desarrollo y operaciones
- Cambiar la cultura de la empresa

En la práctica, los mejores procesos y culturas de DevOps se extienden más allá del desarrollo y las operaciones para incorporar las aportaciones de todas las partes interesadas en la aplicación e incluyendo la ingeniería de la plataforma y la infraestructura, la seguridad, el cumplimiento, la gobernanza, la gestión de riesgos, la línea de negocio, los usuarios finales y los clientes- en el ciclo de vida del desarrollo de software.

Soto 2019 en [4], se definen principios a considerar para la adopción de DevOps

Principios y recomendaciones de implementación.

En la literatura no existe un método milagroso o definitivo para implementar DevOps, esto se da porque cada organización es diferente y particular, existe un término que se ha popularizado y se

denomina CALMS que define un conjunto de pilares fundamentales sobre los cuales se puede llevar a cabo una adopción DevOps.

CALMS, es un acrónimo que significa Cultura, Automatización, Lean, Medición y Sharing (Compartir). Una serie de puntos clave que resultan particularmente útiles para analizar la estructura de una organización. Simple pero efectivo, CALMS marca las directrices para todas las partes interesadas, marcando claramente qué hay que implementar, así como pautas y herramientas para articular e integrar las diferentes partes y procesos de cara a automatizarlos u optimizarlos al máximo para lograr los mejores resultados posibles y esto, de alguna manera, es lo que permite hablar plenamente de DevOps dentro de una organización.

Cultura: Este punto es crucial en una organización. El conocimiento de cómo funciona una organización y conseguir involucrar a sus actores para generar valor es especialmente importante para lograr que todos los actores se involucren y no solo los departamentos técnicos.

La cultura en una organización es el conjunto de saberes, creencias y pautas de conducta, incluyendo los medios materiales que usan sus miembros para comunicarse entre sí, y resolver sus necesidades.

Automatización: Liberar a los equipos de tareas repetitivas o redundantes, logra que los diferentes equipos puedan concentrarse en tareas más productivas.

En este sentido, DevOps propone la aplicación de herramientas que automaticen gran parte de los procesos existentes a lo largo del ciclo de vida como:

Automatización de las revisiones de código. Automatización de la construcción y del despliegue. Automatización de los controles de calidad. Automatización del aprovisionamiento de infraestructuras.

Lean: El desarrollo de software Lean tiene 7 principios, y el primero es “eliminar los desperdicios”, es decir, todo aquello redundante y que supone una pérdida de tiempo ya que no aporta valor. Estos son:

- Código o funcionalidad innecesaria en las aplicaciones.
- Sobrepasar la capacidad del equipo, o empezar más de lo que puede terminarse.
- Requisitos poco claros o con cambios constantes.
- Burocracia.
- Comunicación lenta o inefectiva
- Trabajo parcialmente terminado
- Defectos o problemas de calidad
- Cambio de tareas

Medición (Metrics) Lo que puede medirse puede gestionarse. Dicho de otra manera, lo que se mide se puede gestionar de forma eficaz, y DevOps también pone el foco en esta parte.

El uso de métricas da visibilidad sobre el estado real de un proyecto de forma que los responsables de los equipos, y sus miembros coordinen acciones coherentes con los objetivos corporativos, alineando la visión y la estrategia de la organización.

La medición es el combustible que alimenta el motor DevOps que se centra en la mejora continua, basada en el “empirismo”, y por tanto en la experiencia y los datos obtenidos.

Compartir (Share) La creación de valor surge de la colaboración, del aprovechamiento de experiencias y conocimientos multidisciplinares del equipo. Gracias a compartir conocimiento los integrantes del equipo, y de otras áreas, pueden tener acceso al mismo, de modo que formaran parte del modelo de referencia para un proceso iterativo de aprendizaje y mejora continua, optimizando iterativamente los procesos de la organización.

Compartir también implica invertir en políticas de desarrollo de talento e innovación, que faciliten o permitan cambios dentro de la cultura de la organización.

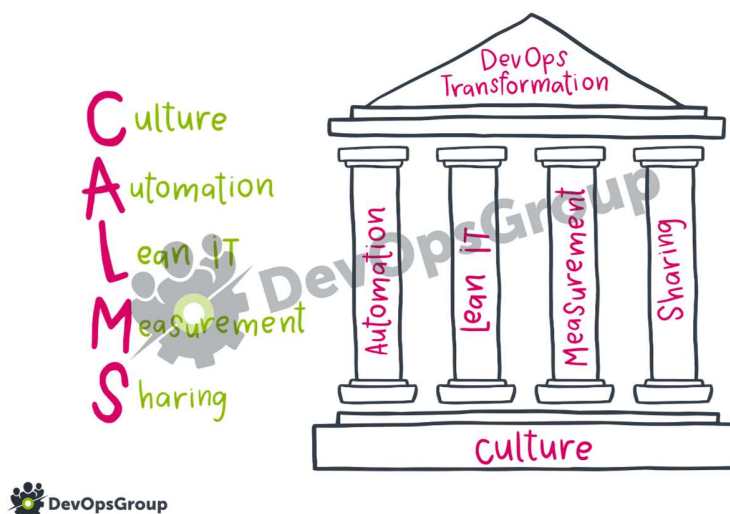


Figure 6 Modelo CALM para DevOps

Imagen tomada de <https://www.devopsgroup.com/insights/resources/diagrams/all/calms-model-of-devops>

DevOps como movimiento global

DevOps es un movimiento mundial que se basa en la experiencia de equipos y organizaciones que ya trabajan con este concepto. Un referente del mercado mundial es empresa Puppet que en los últimos 10 años anualmente realiza una encuesta entre más de 30000 empresas y profesionales

técnicos de todo el mundo en tendencias y uso de DevOps. Generando un reporte anual conocido como “State of DevOps Report” (que puede encontrar en [5]) con el resumen de las experiencias, tendencias de diferentes prácticas y adopción DevOps.

5.2. TRABAJOS RELACIONADOS

DevOps y calidad de software.

En la búsqueda de literatura sobre estos temas existen miles de artículos que enfocan DevOps y muchos más en calidad de software, lo que se busca con este análisis es identificar los documentos más relevantes que unan estos dos conceptos en pro de asegurar que se entreguen soluciones funcionales en periodos de tiempo moderados y que cumplan con buenas prácticas de calidad, documentación y mantenibilidad.

Ordóñez 2020 en [6] presentan un conjunto de lineamientos para la implementación del modelo CALMS para DevOps en miPymes de desarrollo de software en el contexto suramericano, en donde se describen los aspectos de cambios organizacional a tener en cuenta, como temas técnicos relacionados a las herramientas utilizadas para la creación de un pipeline de integración y despliegue continuo. Presentan los resultados de aplicar dicho modelo en una empresa mipyme obteniendo grandes resultados en términos de despliegues, pasando de un despliegue semanal a un despliegue diario. Como conclusión adicional dicen la que “la adopción de DevOps debe ser gradual, iterativa e incremental”

A. Mishra and Z. Otaiwi 2020 en [7] presentan un mapeo sistemático del impacto de DevOps en la calidad del software analiza las implicaciones de las características de DevOps en la calidad del software, proporcionando una mejor comprensión de la estrecha relación que se puede generar DevOps en la calidad del software, concluyendo que DevOps contribuye positivamente en el aseguramiento de la calidad de software.

Perera and Silva (2017) [8], realiza un estudio sobre el impacto de la práctica DevOps en la calidad del software basándose en explorar las prácticas actuales de DevOps en la industria y los factores clave que llevaron a esta conclusión. El autor nos indica que se fundamentó en un sondeo de la experiencia de expertos para determinar cómo mejorar la calidad de software a partir de la práctica de DevOps. Expresan una ecuación para la medición cualitativa de la calidad de software a partir en la correlación de 5 pilares de la cultura DevOps, Cultura, Automatización, Medición, Compartir.

V. L. Cruz and A. B. Albuquerque 2018 en [9], presentan una experiencia de despliegue de DevOps para aplicaciones Legacy o heredados. En este documento se presentaron las herramientas y las modificaciones necesarias para que un proyecto Legacy se integre en un flujo DevOps basándose en los siguientes subprocesos

- Integración continua.
- Automatización de la implementación
- Automatización de pruebas
- Automatización del monitoreo
- Automatización de la infraestructura

ACIS 2018 en [10] hace referencia a un estudio a 1.200 ejecutivos de TI sobre el uso de Agile y DevOps en la transformación digital en el que presenta las características de los “Maestros en Agilidad” (18% de los encuestados), de las cuales forman parte las organizaciones más cercanas a la adopción total y que realizan las principales, o casi todas, las acciones para hacer de Agile y DevOps una parte esencial de sus operaciones diarias. Estos “Maestros en Agilidad” también están más propensos a utilizar prácticas del enfoque Agile en otras funciones de la empresa, por lo que el aumento del 60% en los ingresos y ganancias y la probabilidad de expandir sus negocios es 2,4 veces mayor - con una tasa de aumento por encima del 20% -, no es sólo una coincidencia.

6. METODOLOGÍA

A continuación, se presenta las actividades que se realizarán para el cumplimiento de cada objetivo específico planteado en este documento

1. Definir los factores a considerar para evaluar la conveniencia de adoptar DevOps como estrategia de entrega continua de productos en una organización.
 - Investigar y referenciar marcos de evaluación de madurez DevOps para una organización
 - Investigar y referenciar autores referencia original y regional sobre la adopción de DevOps
 - Identificar y filtrar los criterios de evaluación ajustados a la particularidad de la compañía
2. Proponer un artefacto para evaluar la conveniencia de adoptar DevOps en el CIAT a partir de los factores identificados.
 - Definir los criterios necesarios para determinar el estado de madurez del ciclo de vida de desarrollo de una organización versus las recomendaciones DevOps
 - Caracterizar los elementos particulares del tipo de empresa frente al manejo del ciclo de vida de desarrollo
3. Diagnosticar el estado de los procesos de entrega de software en el CIAT con el instrumento propuesto para determinar si DevOps es una solución adecuada a las necesidades del CIAT.
 - Determinar el estado de madurez de los procesos de entrega de software en la compañía
 - Analizar las ventajas y desventajas que le traería a CIAT la adopción de procesos DevOps
 - Obtener el resultado del análisis de la conveniencia o no de adoptar DevOps en CIAT
4. Proponer un plan de acción a partir del diagnóstico obtenido.
 - Identificar actores y procesos de entrega de software a ser intervenidos acorde al resultado del diagnóstico.
 - Definir etapas de adopción de las recomendaciones independientemente si es o no DevOps la mejor opción para el CIAT.
 - Proponer la estrategia de cambio para que CIAT adopte progresivamente las recomendaciones de mejora de los procesos de entrega de software acorde a las buenas prácticas y recomendaciones que ofrece DevOps.

- Generar una guía de recomendaciones para mejorar el proceso de entrega de productos de software en el CIAT en caso de que la conclusión del diagnóstico no sea usar DevOps

7. RECURSOS A EMPLEAR

7.1. HUMANOS

7.1.1. DIRECTOR

Felipe Mejía Villegas

Ingeniero informático de la Universidad EAFIT, con MBA en administración y gestión de empresas de la Universidad Regiomontana

Resumen

Experiencia de más de 15 años en desarrollo de software en roles de facilitador, gerente de proyectos, coordinador, líder en agilidad, líder técnico y desarrollador de software. Experto en manejo de equipos de trabajo con orientación a entregas de producto con valor y adaptabilidad al cambio.

Experiencia de más de 3 años en docencia que refuerzan los conocimientos y prácticas llevadas a cabo en la industria.

Experiencia laboral

Cofundador y director de operaciones en Castor transformación digital

- Encargado de la coordinación, facilitación y liderazgo en procesos administrativos y operativos de desarrollo de software.
Introducción y liderazgo dentro de la empresa en Scrum y agilidad desde 2008.
Acompañamiento en transformación ágil y digital en múltiples empresas de diferentes sectores.
- Consultoría en agilidad y coordinación de desarrollo de productos en Uniban durante 2 años donde se transformó los procesos de TI en la empresa a trabajar por productos con marcos ágiles logrando:
 - Orquestar células ágiles
 - Implementar Scrum
 - Liberación de productos de software al menos una vez al mes de forma incremental (DevOps)
 - Entregar valor adicional en los productos propuestos, por encima de lo inicialmente pactado logrando entre otros coordinar el desarrollo e implantación de un producto móvil para uso de más de 50 productores aliados y la mejora de 3 productos web para gestionar la relación comercial de la empresa
- Capacitación en agilidad y arquitectura de software a Cesde actualizando a sus docentes de tecnología en nuevos marcos de trabajo

7.1.2. CO-DIRECTORA:

Luisa Fernanda Rincón

Directora Maestría/Especialización Ingeniería de Software

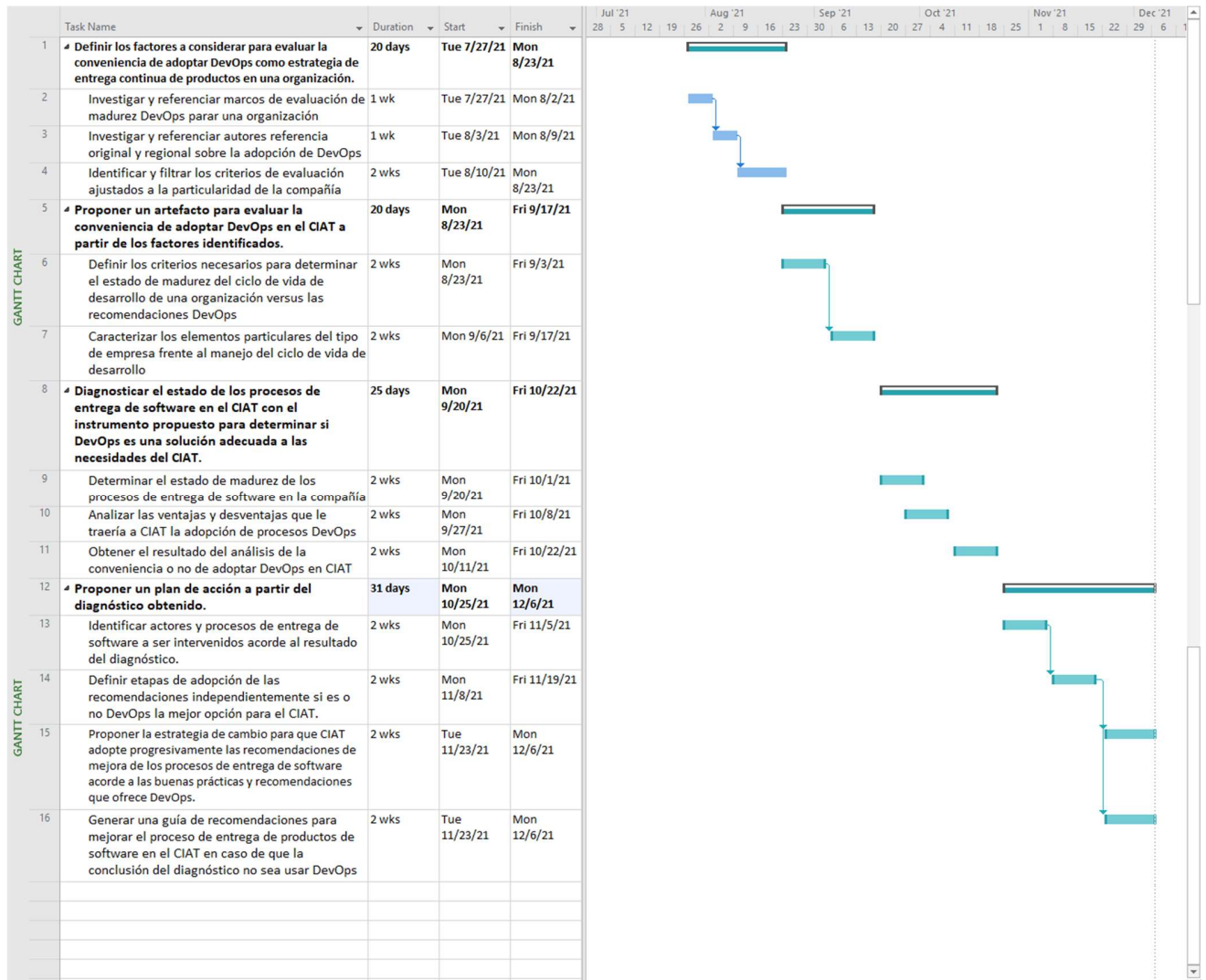
Facultad de Ingeniería y Ciencias

Profesor titular, Pontificia Universidad Javeriana Cali

7.1.3. EMPRESA QUE AVALA

Alianza Bioverisity International y el Centro Internacional de Agricultura tropical

8. CRONOGRAMA DE ACTIVIDADES



9. REFERENCIAS BIBLIOGRÁFICAS

- [1] C. I. Jose Augusto Carvalho, Luciano Moreira, Leonardo Rosso, “Introducción a DevOps y DevSecOps,” 2017.
- [2] J. Smith and an O. M. C. Safari, *Operations Anti-Patterns, DevOps Solutions*. 2020.
- [3] M. & C. M. en P. S. S. Martín, María José, “¿Qué es DevOps? Aclarando el significado,” 12/07/2018, 2018. <https://profile.es/blog/que-es-devops-harder-better-faster-stronger/>.
- [4] J. J. M. Soto, “DevOps: fundamentos, beneficios y cómo implementarlo (PARTE II),” 30 Julio, 2019 - 09:00, 2019. <https://ifgeekthen.everis.com/es/devops-fundamentos-beneficios-y-como-implementarlo>.
- [5] Puppetlabs, “State of DevOps Report.” <https://puppet.com/resources/?refinementList%5Btype%5D%5B0%5D=Report&page=1&configure%5BhitsPerPage%5D=18>.
- [6] H. A. Ordóñez and V. Buchelli, “Lineamientos para la implementación del modelo CALMS de DevOps en mipymes desarrolladoras de software en el contexto sur colombiano,” *Rev. Guillermo Ockham*, vol. 18, no. 1, pp. 81–91, 2020, doi: 10.21500/22563202.4270.
- [7] A. Mishra and Z. Otaiwi, “DevOps and software quality: A systematic mapping,” *Comput. Sci. Rev.*, vol. 38, p. 100308, 2020, doi: 10.1016/j.cosrev.2020.100308.
- [8] P. Perera, R. Silva, and I. Perera, “Improve software quality through practicing DevOps,” *17th Int. Conf. Adv. ICT Emerg. Reg. ICTer 2017 - Proc.*, vol. 2018-Janua, pp. 13–18, 2017, doi: 10.1109/ICTER.2017.8257807.
- [9] V. L. Cruz and A. B. Albuquerque, “A devops introduction process for legacy systems,” *Proc. - 2018 44th Lat. Am. Comput. Conf. CLEI 2018*, pp. 139–148, 2018, doi: 10.1109/CLEI.2018.00025.
- [10] ACIS – Asociación Colombiana de Ingenieros de Sistemas, “El uso de Agile y DevOps genera un aumento de hasta un 60% en las ganancias de las productoras de software,” *Abr 02, 2018*, 2018. <https://acis.org.co/portal/content/NoticiaDelSector/el-uso-de-agile-y-devops-genera-un-aumento-de-hasta-un-60-en-las-ganancias-de-las>.