

ASIGNATURA DE GESTIÓN DEL PROCESO DE DESARROLLO DE SOFTWARE

1. Competencias	Construir soluciones de software y sistemas inteligentes mediante la gestión de proyectos, integración de metodologías, modelos y herramientas de desarrollo bajo la normatividad aplicable para la optimización de proyectos de investigación, innovación, desarrollo tecnológico y de emprendimiento.
2. Cuatrimestre	Décimo
3. Horas Teóricas	20
4. Horas Prácticas	40
5. Horas Totales	60
6. Horas Totales por Semana Cuatrimestre	4
7. Objetivo de aprendizaje	El estudiante gestionará el proceso de desarrollo de software mediante herramientas de automatización para el despliegue continuo de aplicaciones confiables que se adapten a las necesidades cambiantes de las organizaciones.

Unidades de Aprendizaje	Horas		
	Teóricas	Prácticas	Totales
I. Planeación del proceso de desarrollo de software	4	8	12
II. Desarrollo e integración continua	8	16	24
III. Proceso de liberación y monitoreo	8	16	24
Totales	20	40	60

ELABORÓ:	Comité de Directores de la Carrera de Ingeniería en Desarrollo y Gestión de Software	REVISÓ:	Dirección Académica	
APROBÓ:	C. G. U. T. y P.	FECHA DE ENTRADA EN VIGOR:	Septiembre de 2020	

GESTIÓN DEL PROCESO DE DESARROLLO DE SOFTWARE

UNIDADES DE APRENDIZAJE

1. Unidad de aprendizaje	I. Planeación del proceso de desarrollo de software
2. Horas Teóricas	4
3. Horas Prácticas	8
4. Horas Totales	12
5. Objetivo de la Unidad de Aprendizaje	El alumno realizará la planeación del proceso de desarrollo de software con enfoque DevOps para la integración y entrega continua de software.

Temas	Saber	Saber hacer	Ser
Introducción a DevOps.	<p>Identificar las características y las fases del ciclo de vida de DevOps.</p> <p>Identificar normas y estándares aplicables a DevOps.</p> <p>Describir los elementos de DevOps:</p> <ul style="list-style-type: none"> - Infrastructure as code (IaC). - Contenedores y virtualización. - Servicios en la nube y su aprovisionamiento. - Administración de la configuración. - Integración y Entregas continuas (CI/CD). - Monitoreo: registros, métricas y trazas. 		<p>Analítico</p> <p>Sistemático</p> <p>Proactivo</p> <p>Sentido de planeación</p>
Planeación del proceso de desarrollo de software.	<p>Reconocer el proceso de planeación de las metodologías ágiles.</p> <p>Reconocer las características, el funcionamiento, las ventajas y desventajas de las herramientas de comunicación</p>	<p>Realizar la planeación del desarrollo con enfoque DevOps.</p> <p>Determinar la herramienta de comunicación a utilizar.</p>	<p>Analítico</p> <p>Sistemático</p> <p>Proactivo</p> <p>Trabajo en equipo</p> <p>Sentido de planeación</p> <p>Comunicación eficiente</p>

ELABORÓ:	Comité de Directores de la Carrera de Ingeniería en Desarrollo y Gestión de Software	REVISÓ:	Dirección Académica	
APROBÓ:	C. G. U. T. y P.	FECHA DE ENTRADA EN VIGOR:	Septiembre de 2020	

	<p>para proyectos de TI.</p> <p>Describir la adaptación de métodos DevOps a la planeación del proceso de desarrollo de software.</p> <p>Reconocer el proceso de pruebas de software.</p>		<p>Liderazgo</p> <p>Empatía</p>
<p>Preparación del entorno para desarrollo e integración continua.</p>	<p>Reconocer los flujos de trabajo y las herramientas para el control de versiones de software.</p> <p>Identificar las estrategias de despliegue de software:</p> <ul style="list-style-type: none"> - Despliegue con corte en la producción. - Actualización rodante o continua. - Lanzamiento azul/verde. - Lanzamiento canario. - Feature Flags. - Pruebas A/B. 	<p>Establecer el flujo de trabajo para el versionamiento.</p> <p>Determinar las plataformas y herramientas de versionamiento a utilizar.</p> <p>Realizar la configuración de las plataformas y herramientas de versionamiento.</p> <p>Establecer la estrategia de despliegue.</p>	<p>Analítico</p> <p>Sistemático</p> <p>Proactivo</p> <p>Trabajo en equipo</p> <p>Sentido de planeación</p> <p>Trabajo bajo presión</p> <p>Liderazgo</p> <p>Empatía</p>

ELABORÓ:	Comité de Directores de la Carrera de Ingeniería en Desarrollo y Gestión de Software	REVISÓ:	Dirección Académica	
APROBÓ:	C. G. U. T. y P.	FECHA DE ENTRADA EN VIGOR:	Septiembre de 2020	

GESTIÓN DEL PROCESO DE DESARROLLO DE SOFTWARE

PROCESO DE EVALUACIÓN

Resultado de aprendizaje	Secuencia de aprendizaje	Instrumentos y tipos de reactivos
<p>Genera un documento a partir de un caso de estudio el cual incluya:</p> <ul style="list-style-type: none">- Parámetros de configuración de las herramientas utilizadas.- Plan de pruebas.- Casos de pruebas.- Flujo de trabajo para el control de versiones.- Estrategia de despliegue. <p>Entrega un repositorio configurado para recibir el código fuente.</p>	<ol style="list-style-type: none">1. Explicar los elementos de DevOps.2. Identificar normas y estándares aplicables a DevOps.3. Analizar el proceso de desarrollo de software con enfoque DevOps.4. Explicar las estrategias de liberación y versionamiento.5. Relacionar la integración y entrega continua con el proceso de planeación.	<ul style="list-style-type: none">- Estudio de casos.- Lista de cotejo.

ELABORÓ:	Comité de Directores de la Carrera de Ingeniería en Desarrollo y Gestión de Software	REVISÓ:	Dirección Académica	
APROBÓ:	C. G. U. T. y P.	FECHA DE ENTRADA EN VIGOR:	Septiembre de 2020	

GESTIÓN DEL PROCESO DE DESARROLLO DE SOFTWARE

PROCESO ENSEÑANZA APRENDIZAJE

Métodos y técnicas de enseñanza	Medios y materiales didácticos
<ul style="list-style-type: none">- Análisis de casos.- Aprendizaje basado en proyectos.- Equipos colaborativos.	<ul style="list-style-type: none">- Equipos de cómputo.- Proyector.- Internet.- Pizarrón y marcadores.- Plataformas virtuales.- Ejercicios prácticos.- Herramientas de control de versiones.- Herramientas de comunicación.- Acceso a servicios en la nube.

ESPACIO FORMATIVO

Aula	Laboratorio / Taller	Empresa
X	X	

ELABORÓ:	Comité de Directores de la Carrera de Ingeniería en Desarrollo y Gestión de Software	REVISÓ:	Dirección Académica	
APROBÓ:	C. G. U. T. y P.	FECHA DE ENTRADA EN VIGOR:	Septiembre de 2020	

GESTIÓN DEL PROCESO DE DESARROLLO DE SOFTWARE

UNIDADES DE APRENDIZAJE

1. Unidad de aprendizaje	II. Desarrollo e integración continua
2. Horas Teóricas	8
3. Horas Prácticas	16
4. Horas Totales	24
5. Objetivo de la Unidad de Aprendizaje	El alumno determinará las herramientas de automatización de compilación y pruebas para la integración continua de software.

Temas	Saber	Saber hacer	Ser
Técnicas y herramientas para el seguimiento de pruebas.	Identificar las pruebas comunes en DevOps: Unitarias, análisis de código estático, aceptación, integración, regresión, rendimiento, esfuerzo. Identificar las herramientas para automatización de pruebas.	Determinar las herramientas de automatización de pruebas aplicables al tipo de desarrollo.	Analítico Sistemático Proactivo Trabajo en equipo Sentido de planeación Empatía
Herramientas para el proceso de desarrollo e integración continua.	Describir el proceso de automatización de los elementos de la Integración Continua (CI pipeline): - proceso de compilación (Build). - proceso de pruebas. - comunicación de resultados de las pruebas. Identificar las características de las herramientas para la automatización de la Integración Continua aplicables al tipo de desarrollo: - Herramientas de automatización de la	Establecer flujos de trabajo (pipeline) de la integración continua aplicables al tipo de desarrollo. Realizar la instalación y configuración de la herramienta de Integración Continua. Realizar la configuración e integración de las herramientas de automatización: - Compilación (build). - Pruebas.	Analítico Sistemático Proactivo Trabajo en equipo Sentido de planeación Trabajo bajo presión Liderazgo Empatía

ELABORÓ:	Comité de Directores de la Carrera de Ingeniería en Desarrollo y Gestión de Software	REVISÓ:	Dirección Académica	
APROBÓ:	C. G. U. T. y P.	FECHA DE ENTRADA EN VIGOR:	Septiembre de 2020	

	compilación (build). - Herramientas de automatización de pruebas.		
--	--	--	--

ELABORÓ:	Comité de Directores de la Carrera de Ingeniería en Desarrollo y Gestión de Software	REVISÓ:	Dirección Académica	
APROBÓ:	C. G. U. T. y P.	FECHA DE ENTRADA EN VIGOR:	Septiembre de 2020	

GESTIÓN DEL PROCESO DE DESARROLLO DE SOFTWARE

PROCESO DE EVALUACIÓN

Resultado de aprendizaje	Secuencia de aprendizaje	Instrumentos y tipos de reactivos
<p>Genera un documento a partir de un caso de estudio el cual incluya:</p> <ul style="list-style-type: none"> - Justificación del flujo de trabajo (pipeline) para la liberación y despliegue continuo. - Entorno requerido para la liberación y despliegue continuo. - Niveles de servicio acordados. - Métricas para el monitoreo de la aplicación. - Parámetros de configuración de las herramientas utilizadas. <p>Herramienta de liberación continua configurada y vinculada el entorno de despliegue.</p> <p>Entrega en un repositorio configurado:</p> <ul style="list-style-type: none"> - Scripts del flujo de trabajo (pipeline). - Scripts para la generación del entorno de liberación. - Scripts de pruebas en el entorno de liberación. - Scripts para la generación del despliegue. 	<ol style="list-style-type: none"> 1. Identificar el proceso y las herramientas para la liberación y despliegue de software 2. Identificar los elementos y herramientas para el monitoreo de aplicaciones 3. Comprender las métricas para la evaluación del rendimiento de aplicaciones 4. Relacionar planes de mejora con los resultados de la evaluación del rendimiento 	<ul style="list-style-type: none"> - Estudio de casos. - Lista de cotejo.

ELABORÓ:	Comité de Directores de la Carrera de Ingeniería en Desarrollo y Gestión de Software	REVISÓ:	Dirección Académica	
APROBÓ:	C. G. U. T. y P.	FECHA DE ENTRADA EN VIGOR:	Septiembre de 2020	

GESTIÓN DEL PROCESO DE DESARROLLO DE SOFTWARE

PROCESO ENSEÑANZA APRENDIZAJE

Métodos y técnicas de enseñanza	Medios y materiales didácticos
<ul style="list-style-type: none"> - Análisis de casos. - Aprendizaje basado en proyectos. - Prácticas en laboratorio - Equipos colaborativos. 	<ul style="list-style-type: none"> - Equipos de cómputo. - Proyector. - Internet. - Pizarrón y marcadores. - Plataformas virtuales. - Ejercicios prácticos. - Herramientas de control de versiones. - Herramientas de comunicación. - Acceso a servicios en la nube - Entorno de desarrollo - Herramienta de Integración y entrega continua - Herramientas de aprovisionamiento de infraestructura - Herramienta de automatización de pruebas. - Herramientas de monitoreo

ESPACIO FORMATIVO

Aula	Laboratorio / Taller	Empresa
X	X	

ELABORÓ:	Comité de Directores de la Carrera de Ingeniería en Desarrollo y Gestión de Software	REVISÓ:	Dirección Académica	
APROBÓ:	C. G. U. T. y P.	FECHA DE ENTRADA EN VIGOR:	Septiembre de 2020	

GESTIÓN DEL PROCESO DE DESARROLLO DE SOFTWARE

UNIDADES DE APRENDIZAJE

1. Unidad de aprendizaje	III. Proceso de liberación y monitoreo
2. Horas Teóricas	8
3. Horas Prácticas	16
4. Horas Totales	24
5. Objetivo de la Unidad de Aprendizaje	El alumno determinará las herramientas de automatización para la liberación, despliegue, monitoreo y evaluación del rendimiento continuo de software.

Temas	Saber	Saber hacer	Ser
Liberación y despliegue continuo de software (CD).	Identificar las características del proceso de liberación y despliegue continuo (CD): - Entrega continua de artefactos a preproducción. - Entrega continua de artefactos en ambiente de pruebas. - Despliegue automatizado. Identificar las herramientas para liberación y despliegue continuo (CD).	Realizar la configuración de herramientas para liberación y despliegue continuo.	Analítico Sistemático Proactivo Trabajo en equipo Sentido de planeación Empatía
Monitoreo de software.	Describir los elementos de monitoreo: - Métricas. - Registros (logs). - Trazas o rastros (Tracers). Identificar las herramientas para el monitoreo de aplicaciones.	Realizar la instalación y configuración de las herramientas de monitoreo.	Analítico Sistemático Proactivo Trabajo en equipo Sentido de planeación Empatía
Proceso de evaluación de rendimiento.	Identificar las características de los diferentes niveles de servicio con el cliente: - Acuerdo de nivel de servicio (SLA). - Objetivo de nivel de servicio (SLO). - Indicador de nivel de	Determinar las métricas pertinentes de la evaluación del rendimiento. Establecer planes de mejora según resultados de la evaluación del rendimiento.	Analítico Sistemático Proactivo Trabajo en equipo Sentido de planeación Empatía

ELABORÓ:	Comité de Directores de la Carrera de Ingeniería en Desarrollo y Gestión de Software	REVISÓ:	Dirección Académica	
APROBÓ:	C. G. U. T. y P.	FECHA DE ENTRADA EN VIGOR:	Septiembre de 2020	

	<p>servicio (SLI). - Presupuesto de errores.</p> <p>Identificar las métricas de desempeño, confiabilidad y disponibilidad de las aplicaciones.</p> <p>Relacionar los niveles de servicios con las métricas pertinentes.</p>		
--	---	--	--

ELABORÓ:	Comité de Directores de la Carrera de Ingeniería en Desarrollo y Gestión de Software	REVISÓ:	Dirección Académica	
APROBÓ:	C. G. U. T. y P.	FECHA DE ENTRADA EN VIGOR:	Septiembre de 2020	

GESTIÓN DEL PROCESO DE DESARROLLO DE SOFTWARE

PROCESO DE EVALUACIÓN

Resultado de aprendizaje	Secuencia de aprendizaje	Instrumentos y tipos de reactivos
<p>Genera un documento a partir de un caso de estudio el cual incluya:</p> <ul style="list-style-type: none"> - Justificación del flujo de trabajo (pipeline) para la liberación y despliegue continuo. - Entorno requerido para la liberación y despliegue continuo. - Niveles de servicio acordados. - Métricas para el monitoreo de la aplicación. - Parámetros de configuración de las herramientas utilizadas. <p>Herramienta de liberación continua configurada y vinculada el entorno de despliegue.</p> <p>Entrega en un repositorio configurado:</p> <ul style="list-style-type: none"> - Scripts del flujo de trabajo (pipeline). - Scripts para la generación del entorno de liberación. - Scripts de pruebas en el entorno de liberación. - Scripts para la generación del despliegue. 	<ol style="list-style-type: none"> 1. Identificar el proceso y las herramientas para la liberación y despliegue de software. 2. Identificar los elementos y herramientas para el monitoreo de aplicaciones. 3. Comprender las métricas para la evaluación del rendimiento de aplicaciones. 4. Relacionar planes de mejora con los resultados de la evaluación del rendimiento. 	<ul style="list-style-type: none"> - Estudio de casos. - Lista de cotejo.

ELABORÓ:	Comité de Directores de la Carrera de Ingeniería en Desarrollo y Gestión de Software	REVISÓ:	Dirección Académica	
APROBÓ:	C. G. U. T. y P.	FECHA DE ENTRADA EN VIGOR:	Septiembre de 2020	

GESTIÓN DEL PROCESO DE DESARROLLO DE SOFTWARE

PROCESO ENSEÑANZA APRENDIZAJE

Métodos y técnicas de enseñanza	Medios y materiales didácticos
<ul style="list-style-type: none"> - Análisis de casos. - Aprendizaje basado en proyectos. - Prácticas en laboratorio - Equipos colaborativos. 	<ul style="list-style-type: none"> - Equipos de cómputo. - Proyector. - Internet. - Pizarrón y marcadores. - Plataformas virtuales. - Ejercicios prácticos. - Herramientas de control de versiones. - Herramientas de comunicación. - Acceso a servicios en la nube - Entorno de desarrollo - Herramienta de Integración y entrega continua - Herramientas de aprovisionamiento de infraestructura - Herramienta de automatización de pruebas. - Herramientas de monitoreo

ESPACIO FORMATIVO

Aula	Laboratorio / Taller	Empresa
X	X	

ELABORÓ:	Comité de Directores de la Carrera de Ingeniería en Desarrollo y Gestión de Software	REVISÓ:	Dirección Académica	
APROBÓ:	C. G. U. T. y P.	FECHA DE ENTRADA EN VIGOR:	Septiembre de 2020	

GESTIÓN DEL PROCESO DE DESARROLLO DE SOFTWARE

CAPACIDADES DERIVADAS DE LAS COMPETENCIAS PROFESIONALES A LAS QUE CONTRIBUYE LA ASIGNATURA

Capacidad	Criterios de Desempeño
Determinar el tipo de proyecto de software mediante el análisis de un problema para determinar la forma de desarrollo.	Entrega un documento de definición del proyecto que incluya: <ul style="list-style-type: none"> - Descripción de la problemática. - Objetivos establecidos. - Metas y alcances. - Requerimientos. - Justificación del tipo de proyecto.
Gestionar el desarrollo de software mediante el seguimiento de la codificación y ejecución de pruebas para la integración continua del producto.	<ol style="list-style-type: none"> 1. Entrega los archivos de configuración del entorno de producción. 2. Entrega un documento que incluya: <ul style="list-style-type: none"> - Bitácoras del desarrollo de software. - Reportes de seguimiento del desarrollo de software. - Evidencias de control de cambios y versiones del código. - Resultado de pruebas.
Realizar la liberación de unidades estables de software mediante un plan de entregas continuas para contribuir al cumplimiento de los objetivos del proyecto.	<ol style="list-style-type: none"> 1. Entrega los resultados de las pruebas de sistema en entorno de operación. 2. Entrega un documento que incluya: <ul style="list-style-type: none"> - Evidencias del seguimiento de la ejecución del plan de entregas continuas. - Evidencia del acuerdo de aceptación. 3. Entrega archivos que incluya: <ul style="list-style-type: none"> - Unidades estables del software a producción.
Implementar soluciones de software mediante la integración de componentes de acuerdo a la arquitectura definida para su liberación en un ambiente de producción.	<ol style="list-style-type: none"> 1. Entrega manuales del software establecidos en el plan de desarrollo. 2. Entrega la versión estable del software integrando los componentes de acuerdo al plan de entregas continuas. 3. Entrega evidencia de almacenamiento de archivos de código fuente y actualizaciones de la documentación del proyecto en la plataforma de versionamiento establecida.

ELABORÓ:	Comité de Directores de la Carrera de Ingeniería en Desarrollo y Gestión de Software	REVISÓ:	Dirección Académica	
APROBÓ:	C. G. U. T. y P.	FECHA DE ENTRADA EN VIGOR:	Septiembre de 2020	

<p>Ejecutar pruebas a componentes de software a través del uso de técnicas y herramientas de testing para asegurar su correcto funcionamiento.</p>	<p>1. Entrega un documento que incluya:</p> <ul style="list-style-type: none"> - Justificación de las técnicas y herramientas utilizadas. - Descripción de las pruebas. - Informe de resultados del plan de pruebas. <p>2. Entrega archivos de código fuente y configuración del componente.</p>
<p>Establecer metodologías y herramientas de gestión con base en el tipo y características del proyecto identificando las normas, estándares, leyes y regulaciones aplicables para el cumplimiento de los requerimientos establecidos.</p>	<p>Entrega un documento que incluya:</p> <ul style="list-style-type: none"> - Justificación de la metodología seleccionada. - Justificación de las herramientas de gestión. - Listado de las normas, estándares, leyes y regulaciones aplicables.
<p>Supervisar el mantenimiento de la infraestructura de datos de acuerdo al plan de mantenimiento, los parámetros de especificaciones técnicas, herramientas de software de monitoreo, revisiones periódicas, para asegurar el acceso confiable y rápido.</p>	<p>Elabora reporte de supervisión del mantenimiento de la infraestructura, que incluya:</p> <ul style="list-style-type: none"> a) Registro de acciones de mantenimiento realizado <ul style="list-style-type: none"> - Fecha y responsable - Control de versiones: - Procedimientos. - Actualizaciones. - Acciones de contingencia. - Comentarios y observaciones. b) Resultados de las acciones de mantenimiento c) Análisis e interpretación de los resultados d) Conclusiones y sugerencias de mejoras

ELABORÓ:	Comité de Directores de la Carrera de Ingeniería en Desarrollo y Gestión de Software	REVISÓ:	Dirección Académica	
APROBÓ:	C. G. U. T. y P.	FECHA DE ENTRADA EN VIGOR:	Septiembre de 2020	

GESTIÓN DEL PROCESO DE DESARROLLO DE SOFTWARE

FUENTES BIBLIOGRÁFICAS

Autor	Año	Título del Documento	Ciudad	País	Editorial
Chacon, Scott; Straub, Benn	2021 ISBN: 978-1484200773	<i>Pro Git: Todo lo que necesitas saber acerca de Git (Segunda Edición)</i>		Estados Unidos	Apress
Henry Martin Jones	2019 ISBN-10:1704848849 ISBN-13:978-1704848846	<i>Manual de DevOps: Una Guía Para Implementar DevOps .</i>		EEUU	Independently published
Mikael Krief	2019 ISBN-10:1838642730 ISBN-13:978-1838642730	<i>Learning DevOps: The complete guide to accelerate collaboration with Jenkins, Kubernetes, Terraform and Azure DevOps</i>		Reino Unido	Packt Publishing
Thorpe, Ethan	2019 ISBN-10 : 1696089077 ISBN-13 : 978-1696089074	<i>Devops: GUÍA COMPLETA PARA PRINCIPIANTES APRENDE DEVOPS PASO A PASO</i>		EEUU	Independently published
Kinsbruner, Eran	2018 ISBN-10 : 1727132173 ISBN-13 : 978-1727132175	<i>Continuous Testing for DevOps Professionals: A Practical Guide From Industry Experts</i>		EEUU	Createspace Independent Publishing Platform
Braunton, Alex	2018 ISBN-10 : 1789138051 ISBN-13 : 978-1789138054	<i>Hands-On DevOps with Vagrant</i>		Reino Unido	Packt Publishing
Geerling, Jeff	2020 ISBN 978-0-9863934-3-3	<i>Ansible for DevOps: Server and configuration management for humans</i>		EEUU	Midwestern Mac, LLC
Sesto, Vincent, Sarathchandra , Sathsara	2020 ISBN-10 : 1838983449 ISBN-13 : 978-1838983444	<i>The Docker Workshop: Learn how to use Docker containers effectively to speed up</i>		Reino Unido	Packt Publishing

ELABORÓ:	Comité de Directores de la Carrera de Ingeniería en Desarrollo y Gestión de Software	REVISÓ:	Dirección Académica	
APROBÓ:	C. G. U. T. y P.	FECHA DE ENTRADA EN VIGOR:	Septiembre de 2020	

		<i>the development process</i>			
Patil, Ankita, Soni, Mitesh	2021 ISBN de origen : 9389898609	<i>Hands-on Pipeline as Code with Jenkins: CI/CD Implementation for Mobile, Web, and Hybrid Applications Using Declarative Pipeline in Jenkins</i>		India	BPB Publications
Arundel, John, Domingus, Justin	2019 ISBN-10 : 1492040762 ISBN-13 : 978- 1492040767	<i>Cloud Native Devops with Kubernetes: Building, Deploying, and Scaling Modern Applications in the Cloud</i>	California	EEUU	O'Reilly Media
Been, Henry, Gaag, Maik Van Der	2020 ISBN-10 : 1789619696 ISBN-13 : 978- 1789619690	<i>Implementing Azure DevOps Solutions: Learn about Azure DevOps Services to successfully apply DevOps strategies</i>		Reino Unido	Packt Publishing
Kumar, Sharath, Shukla, Pranav	2019 ISBN-10 : 1789954398 ISBN-13 : 978- 1789954395	<i>Learning Elastic Stack 7.0: Distributed search, analytics, and visualization using Elasticsearch, Logstash, Beats, and Kibana</i>		Reino Unido	Packt Publishing
Brazil, Brian	2018 ISBN-10 : 9352137531 ISBN-13 : 978- 9352137534	<i>Prometheus: Up & Running: Infrastructure and Application Performance Monitoring</i>	California	EEUU	O'Reilly Media
Butcher, Zack, Calcote, Lee	2019 ISBN-10 : 1492043788 ISBN-13 : 978- 1492043782	<i>Istio: Up and Running: Using a Service Mesh to Connect, Secure, Control, and Observe</i>	California	EEUU	O'Reilly Media

ELABORÓ:	Comité de Directores de la Carrera de Ingeniería en Desarrollo y Gestión de Software	REVISÓ:	Dirección Académica	
APROBÓ:	C. G. U. T. y P.	FECHA DE ENTRADA EN VIGOR:	Septiembre de 2020	