

# Diseño Curricular Jurisdiccional 2024

## TECNICATURA SUPERIOR EN DevOps



## **AUTORIDADES**

### **Gobernador**

**Alberto WERETILNECK**

### **Ministra de Educación**

**Patricia CAMPOS**

### **Secretaria de Educación**

**Ana Laura GIOVANINI**

### **Directora General de Educación**

**Romina FACCIO**

### **Directora de Educación Superior**

**Gabriela Laura YOCCO**

#### **Equipo Jurisdiccional Curricular**

**Eliana URQUIZA**

**Marianela BOGADO**

#### **Equipo Institucional**

##### **Instituto Técnico Superior Cipolletti**

**Director: Javier Alejandro GÓMEZ BIANCHI**

**Ref. TIC: Gabriela Larissa CAPIZZANO**

**Docente: Developer Sergio Alejandro ARRIAGADA**

**Docente: SysAdmin Mauro Alejandro BENEGA**

**Docente: SysAdmin y Desarrollador Carlos Sebastián CONTRERAS**

**Docente: Profesora Daniela Paola OBREQUE**

**Docente: Desarrollador Matías ORELLANA**

**Docente: Licenciado Sebastián Exequiel PACHECO VELIZ**

**Docente: SysAdmin Julián Sebastián ROSAS**

**Docente: Licenciado Exequiel Walter Fernando WIEDERMANN**

<b>CAPÍTULO I_MARCO DE LA POLÍTICA EDUCATIVA PROVINCIAL Y NACIONAL PARA LA EDUCACIÓN TÉCNICO PROFESIONAL</b>	<b>5</b>
1.1 La Educación Técnico Profesional en la provincia de Río Negro. Antecedentes y nuevos contextos.	5
1.2 Marco Normativo para la Educación Técnico Profesional	7
<b>CAPÍTULO II_LA PROPUESTA DE FORMACIÓN</b>	<b>9</b>
2.1. Finalidades de la Tecnicatura Superior en DevOps: aproximaciones a su campo de estudio, conocimiento.	9
2.2 Descripción de la Carrera	10
2.2.1 Objetivos de la Carrera	12
2.3. Ficha Técnica	14
2.4. Perfil Profesional	14
2.4.1 Alcance del perfil profesional	15
2.4.2. Área ocupacional	15
2.4.3. Capacidades Profesionales	15
2.5. Modalidad de Cursado	15
2.6. Duración de la Carrera	16
2.7. Carga horaria total de la carrera	16
2.8. Condiciones de ingreso	16
<b>CAPÍTULO III_FUNDAMENTOS PEDAGÓGICOS DE LA PROPUESTA CURRICULAR</b>	<b>18</b>
3.1 Acerca del Currículum, el Conocimiento, la Enseñanza y el Aprendizaje	18
3.2. Acerca de la evaluación en la ETP	20
3.3 Acerca de las prácticas profesionalizantes	21
<b>CAPÍTULO IV_ORGANIZACIÓN CURRICULAR</b>	<b>22</b>
4.1 Definición y caracterización de los Campos de la Formación y sus relaciones	22
4.2 Carga horaria por Campo de Formación	22
4.3 Definición de los Formatos Curriculares que integran la propuesta	23
<b>CAPÍTULO V_ESTRUCTURA CURRICULAR</b>	<b>25</b>
5.1 Mapa Curricular	25

**CAPÍTULO VI\_UNIDADES CURRICULARES 26**

**6.1 Presentación de las Unidades Curriculares. Componentes básicos. 26**

**6.1.1 Formación General 26**

6.1.1.1 Unidad curricular: Inglés I 26

6.1.1.2 Unidad curricular: Inglés II 27

6.1.1.3 Unidad curricular: Inglés III 28

6.1.1.4 Unidad curricular: Ética y Legislación 29

**6.1.2 Formación de Fundamentos 29**

6.1.2.1 Unidad curricular: Cultura DevOps y Adopción 29

6.1.2.2 Unidad curricular: Metodologías Ágiles 31

6.1.2.3 Unidad curricular: Control de Versiones 32

6.1.2.4 Unidad curricular: Bases de Datos como Servicio 33

6.1.2.5 Unidad curricular: Refactoring y Testing 34

6.1.2.6 Unidad curricular: Observabilidad 35

6.1.2.7 Unidad curricular: Seguridad Informática 36

**6.1.3 Formación Específica 37**

6.1.3.1 Unidad curricular: Aplicaciones Cloud Nativas 37

6.1.3.2 Unidad curricular: Automatización y Scripting 38

6.1.3.3 Unidad curricular: Sistemas Operativos 39

6.1.3.4 Unidad curricular: Laboratorio I 40

6.1.3.5 Unidad curricular: Calidad, Pruebas y Automatización 42

6.1.3.6 Unidad curricular: Redes Virtuales 43

6.1.3.7 Unidad curricular: Integración y Entrega Continua 44

6.1.3.8 Unidad curricular: Laboratorio II 45

6.1.3.9 Unidad curricular: Operaciones y Escalamiento 46

6.1.3.10 Unidad curricular: Laboratorio III 47

**6.1.4 Prácticas Profesionalizantes 48**

6.1.4.1 Unidad curricular: Práctica Profesionalizante I 49

6.1.4.2 Unidad curricular: Práctica Profesionalizante II 51

6.1.4.3 Unidad curricular: Práctica Profesionalizante III 52

**CAPÍTULO VII\_BIBLIOGRAFÍA GENERAL 54**

**CAPÍTULO VIII\_GLOSARIO TÉCNICO - DevOps 58**

## TECNICATURA SUPERIOR EN

# DevOps

### CAPÍTULO I\_MARCO DE LA POLÍTICA EDUCATIVA PROVINCIAL Y NACIONAL PARA LA EDUCACIÓN TÉCNICO PROFESIONAL

#### 1.1 La Educación Técnico Profesional en la provincia de Río Negro. Antecedentes y nuevos contextos.

La Ley de Educación Nacional N° 26.206 define el Sistema Educativo Nacional en cuatro niveles –Educación Inicial, Educación Primaria, Educación Secundaria y Educación Superior– y ocho modalidades, entre las cuales se ubica como tal la Educación Técnico Profesional. Esta última se rige por la Ley de Educación Técnico Profesional N° 26058 que enmarca los tres niveles de Educación Técnico Profesional: Educación Secundaria, la Educación Superior, responsable de la formación de Técnicos Secundarios y Técnicos Superiores en áreas ocupacionales específicas y de la Formación Profesional para formar recursos humanos en puestos de trabajo específicos.

En el ámbito provincial la sanción de la Ley Orgánica de Educación F N° 4819, establece en el Título 4, Cap. III, Art. 45, inc. b) “La habilitación de mecanismos administrativos e institucionales que permitan la elaboración de propuestas de formación de técnicos (...) superiores no universitarios como también de formación profesional en las diferentes áreas de la producción y los servicios, de acuerdo a las necesidades y potencialidades de desarrollo de cada región de la provincia”.

Toda Educación Técnico Profesional en el Nivel Superior tiene la intencionalidad de diseñar itinerarios profesionalizantes que permitan iniciar o continuar procesos formativos sistemáticos en diferentes áreas ocupacionales.

El ámbito y el nivel de Educación Superior se caracterizan por los rasgos peculiares de sus instituciones. Ellas deben no sólo desarrollar funciones relativas a la enseñanza sino también las concernientes a investigación, desarrollo, extensión y vinculación tecnológica, e innovación.

Dada la pluralidad de instituciones de educación superior que actualmente brindan, como parte de su oferta formativa, la modalidad de Educación Técnico Profesional, se marca como orientación la progresiva especificidad de tales instituciones, de modo de asegurar las condiciones institucionales

necesarias para que la expectativa señalada en el apartado anterior pueda concretarse.

En las últimas décadas, en la Provincia de Río Negro, se ha producido un incremento de propuestas formativas de Educación Superior Técnico Profesional que se implementan desde los Institutos Superiores, con anclaje territorial en diferentes puntos de la geografía provincial dando respuestas a las necesidades de formación e inserción laboral. En consonancia con lo que establece la Ley de Educación F N° 4819, en su Art. 55 establece que: “Las instituciones que brindan Formación Profesional deben reflejar en su propuesta de formación una estrecha vinculación con el medio productivo local y regional en el cual se encuentran insertas para dar respuesta a las demandas de calificación en aquellos sectores con crecimiento sostenido”.

Estas instituciones han transitado diferentes contextos, en cuanto a sus inicios y desarrollo, destacándose las acciones destinadas a cubrir el territorio y satisfacer las demandas de formación específica para el sector socio – productivo, promoviendo una adecuada diversificación, que atiende a las expectativas y requerimientos de la estructura productiva, aprovechando en forma integral los recursos humanos, incrementando las posibilidades de actualización y reconversión para los integrantes del sistema y sus egresados. De este modo identificamos momentos históricos vinculados a las primeras acciones, contándose con establecimientos que surgieron por medio de políticas nacionales (dependientes de la Ex DINEA), transfiriéndose luego al ámbito provincial; otros que se iniciaron a través de iniciativas locales (acuerdos interinstitucionales); aquellos que surgieron a partir de la reconversión institucional (establecimientos que constituían el sistema de formación Docente) y finalmente los establecimientos de creación reciente, que en muchos casos son propuestas ensambladas en los Institutos Superiores de Formación Docente. Asimismo cabe destacar el desarrollo de la Educación Privada cuya oferta educativa representa otra alternativa de Educación Superior Técnico Profesional ligada a diferentes campos del conocimiento, orientadas a la enseñanza aplicada y su vinculación con el empleo, y enfocada en el aumento de la demanda de formación, por parte de una población estudiantil cada vez más heterogénea, con particularidades asociadas al desarrollo económico y social.

Siguiendo los lineamientos LINEAMIENTOS PARA LA ORGANIZACIÓN DE LA OFERTA FORMATIVA PARA LA EDUCACIÓN TÉCNICO PROFESIONAL DE NIVEL SUPERIOR, a saber Resol CFE N° 295/16 y Anexos: En el ámbito de Educación Técnica de nivel superior, la formación de técnicos podrá adoptar carácter diversificado o de especialización en un determinado sector profesional.

La diversificación es entendida como una trayectoria formativa de carácter inicial en una profesión técnica cuyo perfil profesional tiene amplio espectro ocupacional. Implica un plan de estudios que debe asegurar el desarrollo del conjunto de capacidades profesionales propias del perfil profesional elegido y el nivel de complejidad necesario como para permitir altos niveles de autonomía y responsabilidad. La condición de acceso es haber cumplimentado la educación obligatoria en cualquiera de sus formas (o las condiciones establecidas en el art. 7mo. de la Ley de Educación Superior Nro. 24.521).

La especialización alude a una misma trayectoria de profesionalidad de articulación que comprende al nivel secundario, al nivel superior y la formación profesional dentro de la modalidad técnico profesional. Así, su propósito es el de profundizar la ETP inicial alcanzada en la formación profesional, en el nivel de educación secundaria por medio de la continuación del desarrollo de las capacidades profesionales de un determinado perfil profesional en el nivel de educación superior.

## 1.2 Marco Normativo para la Educación Técnico Profesional

El siguiente marco regulatorio configura los lineamientos y criterios para el desarrollo de la Educación Técnico Profesional.

- Ley de Educación Técnico Profesional N° 26.058; que define a la Formación Profesional como ámbito específico de la Modalidad de Educación Técnico Profesional.
- Resolución N° 261/06 CFE, que define el proceso de homologación y marcos de referencia de títulos y certificaciones de Educación Técnico Profesional.
- La Resolución del CFE 13/07, que define y ordena los tipos de títulos y certificaciones de la Educación Técnico Profesional y la Formación Profesional Inicial y Continua, sus niveles certificación y referenciales de ingreso.
- La Resolución del CFE 115/10, que establece los lineamientos y criterios de organización institucional y curricular de la Formación Profesional.
- La Resolución del CFE 287/16, que explicita los marcos para la elaboración de Diseños Curriculares Jurisdiccionales para la Formación Profesional.
- La Resolución del CFE 288/16, que explicita las orientaciones y criterios de organización para la Formación Profesional Continua y la Capacitación Laboral.

- La Resolución del CFE 295/16, que establece los criterios para la organización institucional y lineamientos para la organización de la oferta formativa para la Educación Técnico Profesional de nivel superior.
- Resolución CFE 451/22 “Procedimiento y componentes para la validez nacional”, que establece los establece los criterios y procedimientos para la tramitación de la validez nacional de títulos y/o certificados de los estudios correspondientes a las opciones pedagógicas presenciales y a distancia de todos los niveles y modalidades del Sistema Educativo Nacional previstos en la Ley de Educación Nacional N° 26.206.

## CAPÍTULO II\_ LA PROPUESTA DE FORMACIÓN

### 2.1. Finalidades de la Tecnicatura Superior en DevOps: aproximaciones a su campo de estudio, conocimiento.

El presente diseño curricular tiene como objetivo continuar y profundizar la formación de perfiles profesionales correspondientes al nivel secundario de la modalidad técnico profesional en el área de la informática, específicamente en: software, soporte, redes, sistemas y programación.

A través de la formación técnica especializada en DevOps, se busca ampliar los itinerarios de la formación mediante el dominio de conocimientos, habilidades, destrezas, valores y actitudes. Estos elementos son fundamentales y sólo pueden desarrollarse mediante procesos sistemáticos y prolongados de formación en el Nivel Superior.

Así, su propósito es el de profundizar y profesionalizar la formación de los/las egresados/as de la educación secundaria de la modalidad técnico profesional ampliando el desarrollo de sus habilidades y capacidades profesionales. Este enfoque promueve la construcción de múltiples competencias cognitivas, socioemocionales y técnico-profesionales en el área de TI, cimentando su futuro rol profesional y asegurando la articulación entre el sistema educativo y el sistema productivo, tanto a nivel regional como nacional.

Se considera la formación como una herramienta para potenciar a los individuos, mejorando su calidad de vida y ofreciendo una posibilidad concreta de equidad social. Esto ayuda a reducir las brechas de vulnerabilidad ante la alta competitividad y la complejidad del mundo laboral. La formación ya no puede ser concebida como una adquisición estática y puntual de conocimientos que van a utilizarse para siempre, sino como un proceso continuo que se construye, se renueva y actualiza a lo largo de toda la vida profesional del individuo. De esta manera, la formación inicial requiere prolongarse en una formación continua, permitiendo que los/as técnicos/as adapten sus cualificaciones profesionales a las nuevas demandas derivadas de la evolución y el cambio tecnológico.

DevOps representa una evolución del rol tradicional de SysAdmin (administrador de sistemas) al integrar, transformar y ampliar sus responsabilidades y habilidades hacia un enfoque más colaborativo y automatizado; busca integrar las funciones de operaciones de TI y desarrollo de software en un solo flujo de trabajo. Esta evolución responde a la necesidad de adaptarse a entornos tecnológicos más complejos y a la demanda de una entrega más rápida y confiable de software.

El campo de estudio y conocimiento de un Técnico Superior en DevOps abarca diversas áreas interrelacionadas que combinan conceptos técnicos, metodológicos y culturales. Estas áreas proporcionan una visión integral del ciclo de vida del desarrollo y operación de software, permitiendo a los profesionales implementar soluciones eficaces y eficientes. La formación se sustenta en sólidos conocimientos de los fundamentos de los Sistemas Operativos Libres y de Código Abierto, Cloud Networking (redes en la nube), Programación y Scripting, Herramientas de Automatización y Orquestación, Integración Continua y Entrega Continua (CI/CD), infraestructura como Código (IaC), Cloud Computing (computación en la nube), Monitoreo y Observabilidad, Seguridad Informática y Ciberseguridad, Metodologías Ágiles, Gestión de Proyectos, Habilidades Blandas y Ética Profesional.

Estas disciplinas forman la base del campo de estudio para un Técnico Superior en DevOps, proporcionando una preparación integral y multidisciplinaria que permite a los/las profesionales enfrentar con confianza y competencia los desafíos de la industria tecnológica.

La formación de los/las Técnicos/as Superiores en DevOps se sustenta atendiendo los Lineamientos y Criterios para la organización institucional y curricular de la Educación Técnico Profesional correspondientes a la Educación Secundaria y la Educación Superior aprobados por Resolución del CFE N° 295/16.

El presente diseño curricular para la formación de Técnicos/as Superiores en DevOps se fundamenta en los Marcos de Referencia aprobados por resolución del CFE N° 129/11, 351/19, 352/19, 107/10, 148/11.

## 2.2 Descripción de la Carrera

La carrera de Técnico Superior en DevOps está diseñada para formar profesionales altamente capacitados en la integración y optimización de los procesos de desarrollo y operaciones de software. Este programa educativo combina una sólida base teórica con una intensa formación práctica, permitiendo a los/las estudiantes adquirir las competencias necesarias para implementar y gestionar soluciones tecnológicas innovadoras, eficientes y seguras.

DevOps es una metodología y un conjunto de prácticas que combinan el desarrollo de software (Dev) y las operaciones de TI (Ops) para mejorar la colaboración y productividad entre estos dos equipos tradicionalmente separados. El objetivo principal de DevOps es acelerar el ciclo de vida del

desarrollo de software, desde la creación y prueba hasta el despliegue y monitoreo, proporcionando a las empresas una mayor capacidad de entrega continua y una mejor calidad del producto final.

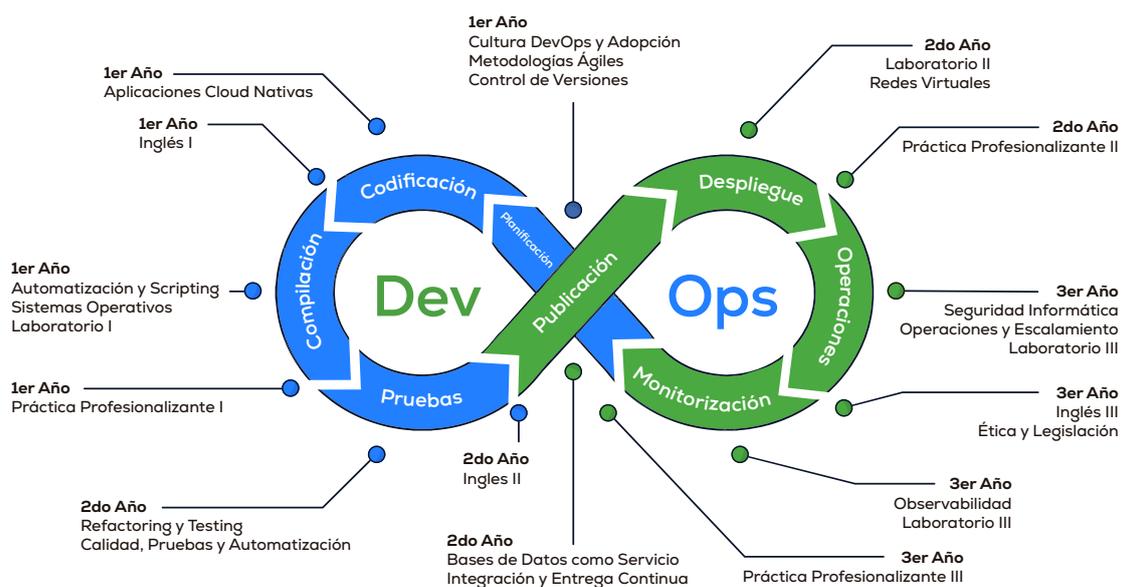
En resumen, DevOps es una práctica integral que transforma la forma en que las organizaciones desarrollan, entregan y operan software, permitiendo una mayor agilidad, calidad y eficiencia en el ciclo de vida del desarrollo de las aplicaciones.

Debido al carácter continuo de DevOps, se usa un bucle infinito para representar que las fases del ciclo de vida de DevOps se relacionan entre sí. A pesar de que parece fluir de forma secuencial, este bucle simboliza la necesidad de colaboración constante y mejora iterativa a lo largo de todo el ciclo de vida.

El ciclo de vida de DevOps se divide en ocho fases, que representan los procesos, capacidades y herramientas necesarios para el desarrollo (en la parte izquierda del bucle) y para las operaciones (en la parte derecha).

A lo largo de cada una de estas fases, los equipos trabajan juntos y mantienen la comunicación para ir coordinados y ajustarse a la misma velocidad y calidad.

### Trayectoria Técnico Superior en DevOps - ITS Cipolletti



La IA tiene un impacto significativo en el ciclo de vida de DevOps al automatizar tareas, mejorar el monitoreo y la detección de anomalías, optimizar el rendimiento y la eficiencia, y fortalecer la seguridad y el cumplimiento. Sin embargo, es importante tener en cuenta que la IA es una herramienta complementaria y no reemplaza por completo la experiencia y el juicio humano. La integración efectiva de la IA en el ciclo de vida de DevOps requiere una combinación de habilidades técnicas y conocimientos específicos de DevOps para aprovechar al máximo sus beneficios.

La mayoría de las herramientas que utiliza un Técnico Superior en DevOps incluyen en mayor o menor medida modelos fundacionales de inteligencia artificial.

### 2.2.1 Objetivos de la Carrera

#### Objetivo General

Promover el acceso a estudios superiores en la Educación Técnico Profesional para la formación de Técnicos Superiores en DevOps, facilitando la adquisición de conocimientos, habilidades y competencias integrales. Este trayecto formativo busca asegurar la progresión académica y la permanencia de los estudiantes, preparándolos para una inserción exitosa en los sectores de Informática y Nuevas Tecnologías, donde podrán contribuir de manera efectiva a la optimización del ciclo de vida del software y la infraestructura de TI en diversas organizaciones.

#### Objetivos Específicos

##### 1) Desarrollar Competencias Técnicas:

- Proporcionar conocimientos sólidos sobre sistemas operativos, redes, bases de datos y seguridad informática, esenciales para el entorno DevOps.
- Enseñar el uso de herramientas y tecnologías clave de DevOps, incluyendo CI/CD, contenedores, orquestación, y plataformas cloud.

##### 2) Fomentar la Automatización y Eficiencia:

- Capacitar a los/las estudiantes en la implementación de procesos de automatización que mejoren la eficiencia y la calidad del desarrollo y la operación de software.
- Instruir en el uso de técnicas y herramientas de testing y aseguramiento de calidad, integradas en pipelines de CI/CD.

- 3) Promover la Cultura DevOps:
  - Impulsar la adopción de una cultura colaborativa entre equipos de desarrollo y operaciones, enfatizando la comunicación, la integración y la cooperación continua.
  - Desarrollar habilidades en metodologías ágiles y gestión de proyectos, facilitando la adaptación y la mejora continua en entornos cambiantes.
- 4) Garantizar la Seguridad y Confiabilidad:
  - Enseñar principios y prácticas de ciberseguridad, asegurando la implementación de medidas de protección en todas las fases del ciclo de vida del software.
  - Capacitar en la identificación y mitigación de riesgos, así como en la respuesta a incidentes de seguridad.
- 5) Facilitar la Escalabilidad y Disponibilidad:
  - Instruir en la planificación y gestión de infraestructuras escalables, tanto on-premise como en la nube, asegurando la alta disponibilidad y el rendimiento óptimo de las aplicaciones.
  - Formar en técnicas de monitoreo y observabilidad, permitiendo la detección y resolución proactiva de problemas en sistemas productivos.
- 6) Integrar Teoría y Práctica:
  - Proporcionar una formación equilibrada que combine conocimientos teóricos con experiencias prácticas, facilitando la transferencia de lo aprendido a situaciones reales.
  - Ofrecer prácticas profesionalizantes en entornos reales, permitiendo a los estudiantes aplicar y consolidar sus habilidades en proyectos concretos.
- 7) Desarrollar Habilidades Blandas:
  - Fomentar el desarrollo de habilidades de comunicación, trabajo en equipo y liderazgo, esenciales para el desempeño efectivo en entornos DevOps.
  - Promover la ética profesional y el cumplimiento de la legislación vigente, asegurando una práctica responsable y acorde a los estándares de la industria.
- 8) Actualizar y Adaptar Conocimientos:
  - Mantener un enfoque actualizado y flexible, incorporando las últimas tendencias y avances tecnológicos en el currículum.

- Preparar a los/las estudiantes para el aprendizaje continuo, equipándolos/as con las herramientas y la mentalidad necesarias para adaptarse a los cambios en el ámbito tecnológico.
- 9) Desarrollar competencias comunicativas en idioma extranjero Inglés:
- Capacitar a los estudiantes en inglés conversacional, asegurando una comunicación efectiva en equipos globales y en el uso de documentación técnica.
  - Desarrollar habilidades para la participación en reuniones, discusiones, y presentaciones en inglés, facilitando la integración en equipos multiculturales y globales.
  - Fomentar la comprensión de la diversidad cultural y la adaptación a diferentes contextos laborales internacionales, esenciales para el éxito en un entorno de trabajo global e interconectado.

### 2.3. Ficha Técnica

- Sector/es de actividad socio productivo: Informática/ Nuevas Tecnologías
- Denominación del perfil profesional: DevOps
- Familia profesional: Informática/ Nuevas Tecnologías
- Denominación del título: Técnico Superior en DevOps
- Nivel y ámbito de la trayectoria formativa: EDUCACIÓN TÉCNICA SUPERIOR
- Tipo de certificación: TÍTULO DE EDUCACIÓN TÉCNICA SUPERIOR

### 2.4. Perfil Profesional

El Técnico Superior en DevOps es un profesional capacitado para:

- Implementar y gestionar procesos de integración y entrega continua.
- Automatizar procesos de despliegue y monitoreo de aplicaciones.
- Administrar infraestructuras en la nube y herramientas de contenedorización.
- Asegurar la calidad y seguridad en el desarrollo y operación de software.
- Colaborar en equipos multidisciplinarios aplicando metodologías ágiles.
- Monitorear y optimizar el rendimiento y la disponibilidad de sistemas y aplicaciones.

### 2.4.1 Alcance del perfil profesional

El Técnico Superior en DevOps estará habilitado para:

- Implementar y gestionar pipelines de CI/CD.
- Administrar sistemas operativos y servicios en la nube.
- Configurar y utilizar herramientas de contenedorización y orquestación.
- Desarrollar y ejecutar pruebas automatizadas.
- Implementar y supervisar medidas de seguridad en entornos DevOps.
- Realizar monitoreo y logging de aplicaciones para detectar y solucionar problemas.
- Colaborar en la adopción de la cultura DevOps en organizaciones.

### 2.4.2. Área Ocupacional

El área ocupacional del Técnico Superior en DevOps incluye:

- Empresas de desarrollo de software.
- Proveedores de servicios en la nube.
- Consultoras tecnológicas.
- Empresas de telecomunicaciones.
- Departamentos de IT en cualquier organización.
- Startups tecnológicas.
- Proyectos de software libre y comunidades de código abierto.

### 2.4.3. Capacidades Profesionales

El proceso de formación habrá de organizarse en torno al desarrollo y acreditación de un conjunto de capacidades profesionales que se corresponden con los desempeños descritos en el perfil profesional.

Las capacidades se desarrollan de manera exhaustiva en las unidades curriculares del presente diseño.

## 2.5. Modalidad de Cursado

Presencial

## 2.6. Duración de la Carrera

3 años

## 2.7. Carga horaria total de la carrera

2.304 horas reloj.

## 2.8. Condiciones de ingreso

La Tecnicatura Superior en DevOps es una especialización de nivel superior dentro de la Educación Técnico Profesional (ETP). La condición de acceso a este tipo de trayectoria formativa es poseer un título de técnico de nivel secundario en una especialidad afín.

Los conocimientos previos solicitados son básicos, se obtienen en un secundario con orientación en el área de informática tanto técnico como ESRN de acuerdo a los Diseños Curriculares:

- Redes: Comprender conceptos esenciales como direccionamiento IP, subredes, enrutamiento, protocolos TCP/IP, topologías de red.
- Programación: Se recomienda Python: Comprender y dominar la sintaxis del lenguaje, las estructuras de datos, el flujo de control, las funciones y las bibliotecas.
- Bases de datos: Bases de datos SQL/NoSQL, modelado de datos maestros, consultas, transacciones y gestión de bases de datos para un almacenamiento y recuperación de datos eficientes.

Las condiciones de ingreso se establecen principalmente para informar y orientar a los/las aspirantes de otras modalidades sobre los conocimientos necesarios para transitar esta tecnicatura con éxito, no son excluyentes pero se espera que los/las estudiantes posean esos conocimientos previos, ya que son esenciales para abordarla.

Asimismo, las condiciones de ingreso generales están especificadas en el Reglamento Académico Marco de la Jurisdicción, Resolución N° 4077/14-CPE, a saber:

- Artículo 5°: Ingreso.- A las instituciones de Educación Superior dependientes de la Provincia de Río Negro se ingresa de manera directa, atendiendo a la igualdad de oportunidades y la no discriminación.

- Artículo 6°: Inscripción.- Para inscribirse en una institución de Educación Superior es necesario presentar la siguiente documentación:
  - a) Solicitud de inscripción.
  - b) Constancia de estudios secundarios completos, acreditados con la presentación del título en original y copia, o constancia de título en trámite o constancia de finalización de cursado del secundario con materias adeudadas.
  - c) Fotocopia autenticada del documento de identidad (datos de identificación y domicilio).
  - d) Partida de Nacimiento actualizada, original y copia autenticada.
  - e) CUIL.

## CAPÍTULO III\_FUNDAMENTOS PEDAGÓGICOS DE LA PROPUESTA CURRICULAR

### 3.1 Acerca del Currículum, el Conocimiento, la Enseñanza y el Aprendizaje.

La Educación Técnico-Profesional se constituye en un ámbito relevante de crecimiento y profesionalización en tiempos de avance científico-tecnológico. La formación de los profesionales se presenta como una necesidad. Esta modalidad establece una estrecha vinculación entre el campo educativo y el campo laboral y requiere de propuestas curriculares abiertas flexibles en permanente actualización vinculadas con los contextos de actuación.

La Educación Técnico Profesional pretende una sólida formación teórico-práctica que incorpore el desarrollo de capacidades, destrezas en el manejo de la especificidad técnica para la que se forma y el aprendizaje de capacidades más integradoras que vinculen conocimientos con habilidades que apunten a la puesta en escena de herramientas de análisis en contextos, para la toma de decisiones. Esta visión intenta articular la formación de un profesional que aprenda no sólo aspectos instrumentales y técnicos sino contextuales frente a las demandas ocupacionales de la región. En este sentido la Ley de Educación Superior Nacional 24.521 en su Título II Capítulo 1 Artículo 3 explicita que: “la educación superior tiene por finalidad proporcionar formación científica, profesional, humanística y técnica en el más alto nivel, contribuir a la preservación de la cultura nacional, promover la generación y desarrollo del conocimiento en todas sus formas, y desarrollar las actitudes y valores que requiere la formación de personas responsables, con conciencia ética y solidaria, reflexivas, críticas, capaces de mejorar la calidad de vida, consolidar el respeto al medio ambiente, a las instituciones de la República y a la vigencia del orden democrático”.

En este sentido se estructuran los Diseños Curriculares para la Formación Técnico Profesional y para comprenderlo es necesario explicitar las concepciones de currículum, conocimiento, enseñanza y aprendizaje que orientan el modo que asume el proceso formativo.

Todo currículum explicita ideas, conocimientos, posiciones acerca del hombre, la cultura y la sociedad. Se presenta como representación y concreción de un proyecto político-social-cultural en el que se manifiestan las intencionalidades educativas. El currículum se considera una herramienta de profesionalización en el sentido de promover capacidades en los ciudadanos para lograr progresos personales, profesionales y sociales. El currículum se concibe aquí como: “la expresión de un proyecto político, pedagógico y cultural, escenario de conflictos entre múltiples intereses e intencionalidades, es un eslabón entre

la teoría educativa y la práctica pedagógica, entre lo que se afirma que puede ser la educación y lo que finalmente es. Por eso debe ser concebido como un proceso que involucra la práctica pedagógica misma como uno de sus componentes básicos." (Diseño Curricular de la Formación Docente - 1988)

En el Currículum se seleccionan y sintetizan conocimientos que una sociedad define como válidos a ser transmitidos. Construidos y producidos socialmente que deben ser comprendidos en su carácter histórico y provisional. El conocimiento se presenta como un producto objetivado y contradictorio de procesos históricos sociales, culturales que son interpelados permanentemente por relaciones de poder.

En relación a los conocimientos a transmitir concebimos a la enseñanza como una práctica compleja que se sitúa en contextos configurados por múltiples dimensiones: institucionales, pedagógicas, didácticas, éticas, políticas. Su particularidad está definida por su relación específica con los conocimientos y los contextos de prácticas donde se producen.

La enseñanza implica siempre un intento deliberado y con relativa sistematicidad de transmisión de conocimientos. Toda enseñanza al ser una acción deliberada no es neutral sino que siempre se evidencia como una actividad política. Como toda acción intencional involucra aspectos personales, sociales y contextuales. En este sentido los aportes de las teorías psicológicas y sociológicas que toman lo grupal y lo social como intervinientes en el aprendizaje son un aporte relevante.

La enseñanza es un campo de prácticas que articula distintos ámbitos de decisión; el político, ético, epistemológico, el técnico y el de los contextos específicos de enseñanza. El desarrollo de una enseñanza situada requiere incorporar la dialéctica permanente entre los conocimientos y su transferibilidad en el ámbito de acción, esto exige la incorporación de diversos abordajes metodológicos según los campos de conocimientos que se involucren.

La intervención en la enseñanza involucra la relación con un aprendiz con sus características y las particularidades del conocimiento a transmitir. El aprendizaje hace referencia a los procesos a través de los cuales se adquieren los conocimientos. La enseñanza influye en el aprendizaje desde la tarea de aprendizaje que realiza el propio estudiante. Las situaciones de enseñanza que se proponen promueven procesos de aprendizaje y construcción de significados por parte del estudiante. Planteamos una concepción de aprendizaje como proceso de construcción de conocimientos que promueven

la transferencia y comprensión de los mismos en situaciones cotidianas. Ubica al estudiante como un sujeto activo intelectualmente con posibilidades de aprendizaje personal y formado a través de la mediación con otros.

### 3.2. Acerca de la evaluación en la ETP

Desde una racionalidad práctica y crítica, la evaluación constituye un campo conflictivo que nos involucra intersubjetivamente en tanto excede las cuestiones técnicas-pedagógicas por estar vinculada a cuestiones éticas, políticas, sociales e ideológicas. De allí la importancia de analizar y comprender los usos y sentidos de la evaluación, la finalidad que persigue, los intereses a los que responde y principios a los que adscribe.

Fernández Sierra (Fernández Sierra, J. - 1994 "Evaluación del Currículum: perspectivas curriculares y enfoques en su evaluación"; en: Teoría del desarrollo del currículum. Málaga: Aljibe) plantea que la evaluación condiciona y determina la vida académica dentro y fuera del aula, involucrando todos los estamentos educativos, desde el trabajo de estudiantes y profesores hasta las decisiones políticas de más alto nivel.

Todo proceso de evaluación responde a múltiples dimensiones: las características de la institución, los proyectos institucionales, los estilos de gestión, las propuestas editoriales y curriculares, las particularidades de los docentes y de los estudiantes, entre otros.

En tal sentido, concebimos la evaluación como una práctica democrática y participativa abierta a la interrogación, la problematización, la búsqueda de entendimiento, la producción de conocimiento y la emancipación de todos los sujetos involucrados.

"La evaluación aplicada a la enseñanza y el aprendizaje consiste en un proceso sistemático y riguroso de recogida de datos, incorporado al proceso educativo desde su comienzo, de manera que sea posible disponer de información continua y significativa para conocer la situación, formar juicios de valor con respecto a ella y tomar las decisiones adecuadas para proseguir la actividad educativa mejorándola progresivamente." (Casanova, M. A. (1995). Manual de evaluación educativa. La Muralla, Madrid.)

Esto nos lleva a afirmar la necesidad de revisar la definición de evaluación que suele sostenerse en los ámbitos educativos y que sólo la ligan a la constatación de conocimientos aprendidos. La evaluación debe concebirse desde su inclusión permanente y constante en nuestra cotidianidad áulica y como una responsabilidad compartida.

La evaluación como práctica de aprendizaje y de enseñanza promueve instancias de auto, co y heteroevaluación y no actividades instrumentales que generan medición y clasificación de los aprendizajes en los sujetos.

La evaluación como parte del proceso didáctico implica para los estudiantes una toma de conciencia de los aprendizajes adquiridos y, para los docentes, una interpretación de las implicancias de la enseñanza en esos aprendizajes. De este modo retroalimenta el proceso de enseñanza e informa a los estudiantes los progresos en sus aprendizajes. Será siempre formativa, motivadora, orientadora y al servicio de los protagonistas.

### **3.3 Acerca de las prácticas profesionalizantes**

Las prácticas profesionalizantes propician una aproximación progresiva al campo ocupacional, de un determinado profesional, hacia el cual se orienta la formación. Favorecen la integración y consolidación de los saberes a los cuales se refiere ese campo ocupacional y la profesión, poniendo a los estudiantes en contacto con diferentes situaciones y problemáticas que permitan tanto la identificación del objeto de la práctica profesional como la del conjunto de procesos técnicos, tecnológicos, científicos, culturales, sociales y jurídicos que se involucran en la diversidad de situaciones socioculturales y productivas que se relacionan con un posible desempeño profesional (Resolución N° 295/16-CFE).

## CAPÍTULO IV ORGANIZACIÓN CURRICULAR

### 4.1 Definición y caracterización de los Campos de la Formación y sus relaciones

El Plan de Estudios se organiza en torno a cuatro campos de formación establecidos por la Resolución CFE N°295/16-CFE.

- Campo de la formación general:

Destinado a abordar los saberes que posibiliten la participación activa, reflexiva y crítica en los diversos ámbitos de la vida laboral y sociocultural, y, el desarrollo de una actitud ética respecto del continuo cambio tecnológico y social.

- Campo de la formación del fundamento:

Destinado a abordar los saberes científico tecnológicos y socioculturales que otorgan sostén a los conocimientos, habilidades, destrezas, valores y actitudes propios del campo profesional en cuestión.

- Campo de la formación específica:

Dedicado a abordar los saberes propios de cada campo profesional, así como también la contextualización de los desarrollados en la formación de fundamento.

- Campo de la formación en la práctica profesional:

Destinado a posibilitar la integración y contrastación de los saberes construidos en la formación de los campos descriptos, y garantizar la articulación teoría-práctica en los procesos formativos a través del acercamiento de los estudiantes a situaciones reales de trabajo.

### 4.2 Carga horaria por campo de formación

Campos de Formación	% en Plan de estudios	% Actividades teóricas	% Actividades prácticas formativas	Total de hs de la carrera: <b>2304 Hs reloj</b>
Formación general	19,44%	60%	40%	
Formación de Fundamento	22,22%	50%	50%	
Formación específica	44,44%	40%	60%	
Prácticas Profesionalizantes	13,88%	20%	80%	

\* Según lo establecido por la Resolución N° 295/16 del Consejo Federal de Educación.

### 4.3 Definición de los Formatos Curriculares que integran la propuesta

El desarrollo de las trayectorias formativas, contempla una organización curricular flexible, en donde los **Espacios Curriculares**, deben estar claramente definidos en cada uno de los campos profesionales, dando unidad y significado a los contenidos y actividades con un enfoque pluridisciplinario, y que garanticen una lógica de progresión que organice los procesos de enseñanza y de aprendizaje en un orden de complejidad creciente. Para hacerlo posible se combinan espacios curriculares de diferente tipo, disciplinas organizadas de una manera convencional, espacios curriculares que se organicen en torno a problemáticas propias de la profesión, espacios para proyectos con diferente grado de autonomía por parte de los estudiantes, para actividades a realizar fuera de la institución, entre otras propuestas de trabajo, las cuales conforman espacios de integración, como por ejemplo: proyectos, seminarios, seminarios de intercambio y debate de experiencias, congresos jornadas y talleres, conferencias y coloquios, entre otros formatos pedagógicos, que forman parte constitutiva del diseño, organizan la enseñanza y los distintos contenidos de la formación y deben ser acreditadas por los estudiantes.

- **Las asignaturas:** son unidades curriculares definidas por la enseñanza de marcos disciplinares o multidisciplinares y sus derivaciones metodológicas para la intervención educativa. Son de valor troncal para la formación y se caracterizan por brindar conocimientos, modos de pensamiento y modelos explicativos. Permiten el análisis de problemas, la investigación documental, la preparación de informes, el desarrollo de la comunicación oral y escrita y la aproximación a métodos de trabajo intelectual transferibles a la acción profesional. En cuanto al tiempo y ritmo, pueden adoptar la periodización anual o cuatrimestral, incluyendo su secuencia en cuatrimestres sucesivos.
- **El trabajo de campo:** propone un acercamiento real al contexto, a la cultura de la comunidad, a las instituciones y los sujetos en los que acontecen las experiencias de práctica. Los Trabajos de Campo constituyen espacios sistemáticos de trabajos de indagación en terreno e intervenciones en espacios acotados, desarrollo de micro experiencias, prácticas sistemáticas en contextos específicos y diversos.
- **Los Talleres** son unidades curriculares que promueven la resolución práctica de situaciones que requieren de un hacer creativo y reflexivo, poniendo en juego marcos conceptuales disponibles, también posibilita la búsqueda de otros marcos necesarios para orientar, resolver o interpretar los desafíos de la producción. Como modalidad pedagógica, el taller apunta al desarrollo de capacidades para el análisis de casos y de

alternativas de acción, la toma de decisiones y la producción de soluciones e innovaciones. En este proceso, se estimula la capacidad de intercambio, la búsqueda de soluciones originales y la autonomía del grupo. Su organización es adaptable a los tiempos cuatrimestrales.

- **Los seminarios** son unidades que se organizan en torno a casos, problemas, temas o corrientes de pensamientos para cuyo análisis se requiere de una producción específica, la contraposición de enfoques, posiciones y debate. Implican instancias académicas de estudio de problemas relevantes para la formación profesional, a través de la reflexión crítica de las concepciones o supuestos previos sobre tales problemas, que los estudiantes tienen incorporados como resultado de su propia experiencia, para luego profundizar su comprensión a través de la lectura y el debate de materiales bibliográficos o de investigación. Estas unidades permiten el cuestionamiento del "pensamiento práctico" y ejercitan en el trabajo reflexivo y el manejo de literatura específica, como usuarios activos de la producción del conocimiento. Los seminarios se adaptan bien a la organización cuatrimestral, atendiendo a la necesidad de organizarlos por temas/ problemas. Asimismo estos espacios incluyen dispositivos que enriquecen el proceso formativo, como propuestas de opcionalidad académica que supondrán la articulación entre diferentes instituciones (sociales, académicas, políticas, etc.).
- **Conferencias y coloquios** conforman encuentros de aprendizaje con especialistas especialmente invitados, sobre temáticas relativas a los contenidos que se están desarrollando en los distintos cursos para resignificar, ampliar y profundizar los marcos interpretativos.
- **Seminarios de intercambio y debate de experiencias:** encuentros de presentación de experiencias, de informes de estudios de campo, de trabajos monográficos, posters, y otras modalidades, con debate de sus desarrollos y conclusiones con el propósito de valorizar, producir, sistematizar y socializar conocimientos, investigaciones operativas llevadas a cabo por los estudiantes durante su proceso de formación.
- **Congresos, Jornadas, Talleres:** actividades académicas sistematizadas que organizadas por los Institutos Superiores u otro tipo de instituciones reconocidas permiten, aún antes del egreso, vincular a los estudiantes con el mundo técnico-profesional.

**CAPÍTULO V\_ESTRUCTURA CURRICULAR** 5.1. Mapa Curricular

TECNICATURA SUPERIOR EN DevOps							
FORMACIÓN GENERAL		FORMACIÓN DE FUNDAMENTO		FORMACIÓN ESPECÍFICA		PRÁCTICAS PROFESIONALIZANTES	
<b>PRIMER AÑO</b>							
1° Cuatrimestre	2° Cuatrimestre	1° Cuatrimestre	2° Cuatrimestre	1° Cuatrimestre	2° Cuatrimestre	1° Cuatrimestre	2° Cuatrimestre
Inglés I (4hs/128hs)		Cultura DevOps y Adopción (4hs/64hs) Metodologías Ágiles (4hs/64hs)	Control de Versiones (4hs/64hs)	Aplicaciones Cloud Nativas (4hs/128hs) Sistemas Operativos (4hs/128hs) Automatización y Scripting (4hs/64hs)	Laboratorio I (4hs/64hs)		Práctica Profesionalizante I (4hs/64hs)
<b>SEGUNDO AÑO</b>							
Inglés II (4hs/128hs)		Bases de Datos como Servicio (4hs/64hs)	Refactoring y Testing (4hs/64hs)	Calidad, Pruebas y Automatización (4hs/128hs) Laboratorio II (4hs/128hs) Redes Virtuales (4hs/64hs)	Integración y Entrega Continua (4hs/64hs)		Práctica Profesionalizante II (4hs/128hs)
<b>TERCER AÑO</b>							
Inglés III (4hs/128hs)		Seguridad Informática (4hs/128hs)		Operaciones y Escalamiento (4hs/128hs) Laboratorio III (4hs/128hs)			Práctica Profesionalizante III (4hs/128hs)
	Ética y Legislación (4hs/64hs)	Observabilidad (4hs/64hs)					
<b>448 hs</b>		<b>512 hs</b>		<b>1024 hs</b>		<b>320 hs</b>	
<b>TOTAL HORAS DE LA CARRERA: 2.304 hs</b>							

## CAPÍTULO VI\_UNIDADES CURRICULARES

### 6.1 Presentación de las Unidades Curriculares. Componentes básicos.

En la presentación de las unidades curriculares se explicitan los formatos y las orientaciones para la selección de contenidos.

**Formatos:** Implican no sólo un determinado modo de transmisión del conocimiento, sino también una forma particular de intervención en los modos de pensamiento, en las formas de indagación, en los hábitos que se construyen para definir la vinculación con un objeto de conocimiento(Resolución CFE N° 24/07).

**Finalidades formativas de una unidad curricular:** Las finalidades formativas de la unidad curricular son un componente del Encuadre Didáctico de los Diseños Curriculares Jurisdiccionales. Son un tipo particular de propósitos que refieren a aquellos saberes que los y las estudiantes deben acreditar al finalizar el curso y que están vinculados a las prácticas profesionalizantes.

**Capacidades vinculadas con la Unidad Curricular:** Son recursos para actuar de manera competente de acuerdo a la unidad curricular. Estos recursos son conocimientos, habilidades y actitudes que los/as estudiantes utilizan para afrontar una situación determinada. Están directamente vinculadas con las capacidades a desarrollar en el/la futuro/a profesional técnico/a y son un componente del Encuadre Didáctico de los Diseños Curriculares Provinciales.

**Ejes de contenidos - descriptores:** Adscribiendo a la concepción de los diseños curriculares como un “marco de organización y de actuación y no un esquema rígido de desarrollo”, el presente diseño curricular incorpora criterios de apertura y flexibilidad para que “el currículum en acción” adquiera una fluida dinámica, sin que sea una rígida e irreflexiva aplicación del diseño curricular o un requerimiento burocrático a ser evitado. En ese encuadre se presentan los ejes de contenidos, concebidos como las nociones más generales y abarcadoras que constituirán la unidad curricular con la función de delimitar, definir y especificar los campos conceptuales que la integran.

#### 6.1.1\_FORMACIÓN GENERAL

##### 6.1.1.1 Unidad curricular: Inglés I

Formato: Asignatura

Régimen de cursada: Anual

Ubicación en el diseño curricular: 1er año

Asignación de horas semanales: 4 hs.

Total de horas: 128 hs.

### Finalidades formativas y capacidades a desarrollar

Desarrollar competencias comunicativas en la lengua inglesa, tanto orales como escritas, que permitan al/la estudiante comprender y expresarse de manera efectiva en situaciones cotidianas, académicas y profesionales. Adquirir vocabulario técnico y terminología específica del campo de las Tecnologías de la Información y la Comunicación (TIC) en inglés. Fortalecer la capacidad de comprensión de textos y discursos en inglés relacionados con las TIC. Fomentar el interés por el aprendizaje continuo del inglés como herramienta clave para el desarrollo profesional y personal.

### Ejes de contenidos. Descriptores

Gramática y estructuras básicas de la lengua inglesa: tiempos verbales, sustantivos, adjetivos, pronombres, preposiciones, entre otros. Comprensión lectora de textos generales y específicos del ámbito de las TIC. Producción escrita de párrafos, correos electrónicos y documentos sencillos. Comprensión auditiva y expresión oral en situaciones cotidianas y académicas. Vocabulario general y específico relacionado con las TIC. Aspectos culturales y de comunicación intercultural.

#### 6.1.1.2 Unidad curricular: Inglés II

Formato: Asignatura

Régimen de cursada: Anual

Ubicación en el diseño curricular: 2do año

Asignación de horas semanales: 4 hs.

Total de horas: 128 hs.

### Finalidades formativas y capacidades a desarrollar

Consolidar y ampliar las competencias comunicativas en inglés adquiridas en Inglés I, tanto en la expresión oral como escrita. Desarrollar habilidades de comprensión y análisis de textos técnicos y académicos en el ámbito de las TIC. Fortalecer la capacidad de comunicación efectiva en situaciones profesionales y laborales relacionadas con las TIC. Promover el interés por la cultura y la diversidad lingüística como elementos clave para el desempeño profesional en un contexto globalizado.

### Ejes de contenidos. Descriptores

Gramática y estructuras intermedias de la lengua inglesa: voz pasiva, condicionales, cláusulas relativas, entre otras. Comprensión lectora de textos

técnicos, manuales y documentos académicos relacionados con las TIC. Producción escrita de informes, propuestas y documentos profesionales. Comprensión auditiva y expresión oral en situaciones laborales y profesionales. Vocabulario técnico y terminología específica del campo de las TIC. Aspectos culturales y de comunicación intercultural en el ámbito profesional.

### 6.1.1.3 Unidad curricular: Inglés III

Formato: Asignatura

Régimen de cursada: Anual

Ubicación en el diseño curricular: 3er año

Asignación de horas semanales: 4 hs.

Total de horas: 128 hs.

#### Finalidades formativas y capacidades a desarrollar

Profundizar las competencias comunicativas en inglés, tanto orales como escritas, alcanzando un nivel avanzado que permita al estudiante desempeñarse de manera efectiva en contextos profesionales y académicos. Desarrollar habilidades de comprensión y análisis crítico de textos científicos y académicos en el campo de las TIC. Fomentar la capacidad de expresión oral y escrita en inglés para la presentación de proyectos, informes y trabajos de investigación. Promover el interés por el aprendizaje continuo del inglés como herramienta clave para el desarrollo profesional y la actualización constante en el ámbito de las TIC.

#### Ejes de contenidos. Descriptores

Gramática y estructuras avanzadas de la lengua inglesa: voz pasiva compleja, cláusulas condicionales mixtas, discurso indirecto, entre otras. Comprensión lectora de textos científicos, artículos académicos y trabajos de investigación relacionados con las TIC. Producción escrita de informes técnicos. Comprensión auditiva y expresión oral en presentaciones, conferencias y eventos académicos y profesionales. Vocabulario técnico y terminología específica avanzada del campo de las TIC. Aspectos culturales y de comunicación intercultural en el ámbito académico y profesional.

#### 6.1.1.4 Unidad curricular: Ética y Legislación

Formato: Asignatura

Régimen de cursada: Cuatrimestral

Ubicación en el diseño curricular: 3er año 2do Cuatrimestre

Asignación de horas semanales: 4 hs.

Total de horas: 64 hs.

#### Finalidades formativas y capacidades a desarrollar

Comprender los principios éticos y los marcos normativos que rigen el ejercicio profesional en el ámbito de las Tecnologías de la Información y la Comunicación (TIC). Desarrollar una conciencia crítica y reflexiva sobre las implicaciones éticas, legales y sociales del uso de las TIC. Fomentar una actitud responsable y comprometida con el respeto a los derechos y libertades fundamentales en el desarrollo y aplicación de las TIC. Adquirir conocimientos sobre la legislación vigente en materia de propiedad intelectual, privacidad y seguridad de la información, y su impacto en el ejercicio profesional.

#### Ejes de contenidos. Descriptores

Ética y responsabilidad profesional en el ámbito de las TIC. Principios éticos y valores fundamentales: integridad, honestidad, respeto, confidencialidad, entre otros. Impacto social y ambiental de las TIC. Legislación vigente en materia de propiedad intelectual, derechos de autor y patentes y delitos informáticos. Normativa sobre protección de datos personales y privacidad. Derecho laboral y teletrabajo. Seguridad de la información y ciberseguridad. Regulaciones y estándares internacionales en el ámbito de las TIC. Ética y dilemas éticos en el desarrollo y aplicación de las TIC: inteligencia artificial, big data, entre otros.

### 6.1.2\_FORMACIÓN DE FUNDAMENTOS

#### 6.1.2.1 Unidad curricular: Cultura DevOps y Adopción

Formato: Asignatura

Régimen de cursada: Cuatrimestral

Ubicación en el diseño curricular: 1er año 1er Cuatrimestre

Asignación de horas semanales: 4 hs.

Total de horas: 64 hs.

### Finalidades formativas y capacidades a desarrollar

Comprender los principios fundamentales de la cultura DevOps y su importancia en la entrega de software de alta calidad de manera ágil y eficiente. Analizar los beneficios y desafíos de la adopción de DevOps en organizaciones, identificando las estrategias y prácticas clave para una implementación exitosa. Desarrollar habilidades para fomentar la colaboración, la comunicación efectiva y la responsabilidad compartida entre los equipos de desarrollo y operaciones.

### Ejes de contenidos. Descriptores

Principios fundamentales de la cultura DevOps: colaboración, automatización, monitoreo continuo y entrega continua. Beneficios de DevOps: mayor velocidad de entrega, calidad mejorada, reducción de riesgos y costos. Desafíos en la adopción de DevOps: resistencia al cambio, falta de habilidades, integración de herramientas y procesos. Estrategias para la adopción de DevOps: liderazgo y patrocinio ejecutivo, formación y capacitación, métricas y mediciones. Prácticas clave de DevOps: integración continua, entrega continua, infraestructura como código, monitoreo y registro.

### Prácticas Formativas

Para poder desarrollar la unidad curricular se realizan Prácticas Formativas mediante la instalación, configuración y uso de Herramientas, Servicios y Recursos comúnmente utilizados:

- herramientas de control de versiones (Git, GitHub, GitLab),
- herramientas de integración continua (Jenkins, Travis CI, CircleCI),
- herramientas de infraestructura como código (Terraform, Ansible, Puppet),
- herramientas de monitoreo (Prometheus, Grafana, ELK Stack),
- herramientas de registro (Splunk, Graylog),
- plataformas de colaboración (Slack, Microsoft Teams).

### 6.1.2.2 Unidad curricular: Metodologías Ágiles

Formato: Asignatura

Régimen de cursada: Cuatrimestral

Ubicación en el diseño curricular: 1er año 1er Cuatrimestre

Asignación de horas semanales: 4 hs.

Total de horas: 64 hs.

#### Finalidades formativas y capacidades a desarrollar

Comprender los principios y prácticas de las metodologías ágiles, su importancia en el desarrollo de software y su relación con la cultura DevOps. Adquirir conocimientos sobre los marcos de trabajo ágiles más populares, como Scrum y Kanban, y su aplicación en entornos de desarrollo y operaciones. Desarrollar habilidades para la gestión de proyectos ágiles, la planificación de entregas, la colaboración en equipos multidisciplinarios y la adaptación al cambio.

#### Ejes de contenidos. Descriptores

Introducción a las metodologías ágiles: valores y principios del Manifiesto Ágil. Scrum: roles, eventos, artefactos y prácticas. Kanban: principios, flujo de trabajo y métricas. Planificación de entregas ágiles: historias de usuario, estimación, priorización y gestión del backlog. Trabajo en equipos multidisciplinarios: colaboración, comunicación y toma de decisiones. Adaptación al cambio: retrospectivas, mejora continua y aprendizaje.

#### Prácticas Formativas

Para poder desarrollar la unidad curricular se realizan Prácticas Formativas mediante la instalación, configuración y uso de Herramientas, Servicios y Recursos comúnmente utilizados:

- herramientas de gestión de proyectos ágiles (Jira, Trello, Azure DevOps),
- herramientas de planificación y seguimiento (Sprint.ly, Pivotal Tracker, ZenHub),
- herramientas de colaboración (Slack, Microsoft Teams),
- herramientas de visualización de flujo de trabajo (Kanban Tool, Kanban Flow),
- recursos en línea
- comunidades de práctica.

### 6.1.2.3 Unidad curricular: Control de Versiones

Formato: Asignatura

Régimen de cursada: Cuatrimestral

Ubicación en el diseño curricular: 1er año 2do Cuatrimestre

Asignación de horas semanales: 4 hs.

Total de horas: 64 hs.

#### Finalidades formativas y capacidades a desarrollar

Comprender la importancia y los beneficios del control de versiones en el desarrollo de software, así como su papel fundamental en la cultura DevOps. Adquirir conocimientos y habilidades prácticas en el uso de sistemas de control de versiones distribuidos, como Git, para la gestión eficiente del código fuente y la colaboración en equipos de desarrollo.

#### Ejes de contenidos. Descriptores

Fundamentos del control de versiones: conceptos clave, beneficios y flujos de trabajo. Sistemas de control de versiones distribuidos: Git, principios y arquitectura. Repositorios Git: creación, clonación y configuración. Flujo de trabajo básico en Git: commit, push, pull, merge y branch. Gestión de ramas y fusiones. Resolución de conflictos. Trabajo colaborativo en equipos: revisiones de código, solicitudes de extracción (pull requests) y políticas de rama.

#### Prácticas Formativas

Para poder desarrollar la unidad curricular se realizan Prácticas Formativas mediante la instalación, configuración y uso de Herramientas, Servicios y Recursos comúnmente utilizados:

- Git, GitHub, GitLab, Bitbucket,
- herramientas de revisión de código (Gerrit, Review Board),
- herramientas de integración continua (Jenkins, Travis CI, CircleCI),
- recursos en línea
- comunidades de práctica.

#### 6.1.2.4 Unidad curricular: Bases de Datos como Servicio

Formato: Asignatura

Régimen de cursada: Cuatrimestral

Ubicación en el diseño curricular: 2do año 1er Cuatrimestre

Asignación de horas semanales: 4 hs.

Total de horas: 64 hs.

#### Finalidades formativas y capacidades a desarrollar

Comprender el concepto de bases de datos como servicio (DBaaS) y su importancia en entornos de nube y DevOps. Adquirir conocimientos sobre los diferentes tipos de bases de datos, su escalabilidad, disponibilidad y rendimiento en entornos de nube. Desarrollar habilidades para el diseño, implementación y administración de bases de datos en servicios de nube, utilizando herramientas y prácticas de DevOps.

#### Ejes de contenidos. Descriptores

Bases de datos como servicio (DBaaS): conceptos, beneficios y desafíos. Tipos de bases de datos: relacionales, NoSQL, bases de datos en memoria. Escalabilidad, disponibilidad y rendimiento de bases de datos en la nube. Servicios de bases de datos en la nube (AWS RDS, Google Cloud SQL, Azure SQL Database). Aprovisionamiento y configuración de bases de datos en la nube. Automatización de tareas de administración de bases de datos. Monitoreo y optimización del rendimiento de bases de datos.

#### Prácticas Formativas

Para poder desarrollar la unidad curricular se realizan Prácticas Formativas mediante la instalación, configuración y uso de Herramientas, Servicios y Recursos comúnmente utilizados:

- servicios de bases de datos en la nube (AWS RDS, Google Cloud SQL, Azure SQL Database),
- herramientas de automatización (Terraform, Ansible),
- herramientas de monitoreo (Prometheus, Datadog),
- herramientas de optimización de rendimiento (pgBadger, MySQL Tuner),
- recursos en línea
- comunidades de práctica.

### 6.1.2.5 Unidad curricular: Refactoring y Testing

Formato: Asignatura

Régimen de cursada: Cuatrimestral

Ubicación en el diseño curricular: 2do año 2do Cuatrimestre

Asignación de horas semanales: 4 hs.

Total de horas: 64 hs.

#### Finalidades formativas y capacidades a desarrollar

Comprender la importancia del refactoring y las pruebas (testing) en el desarrollo de software de alta calidad y su relación con las prácticas de DevOps. Adquirir conocimientos y habilidades prácticas para aplicar técnicas de refactoring y diferentes tipos de pruebas en el ciclo de desarrollo de software, con el objetivo de mejorar la calidad, mantenibilidad y eficiencia del código.

#### Ejes de contenidos. Descriptores

Refactoring: principios, beneficios y técnicas. Código limpio y buenas prácticas de programación. Pruebas unitarias: frameworks de pruebas, cobertura de código e integración continua. Pruebas de integración y de extremo a extremo. Pruebas de rendimiento y carga. Pruebas de seguridad y penetración. Automatización de pruebas y pipelines de CI/CD. Informes de pruebas y métricas de calidad.

#### Prácticas Formativas

Para poder desarrollar la unidad curricular se realizan Prácticas Formativas mediante la instalación, configuración y uso de Herramientas, Servicios y Recursos comúnmente utilizados:

- herramientas de refactoring (IDE, herramientas de análisis de código),
- frameworks de pruebas unitarias (JUnit, Mocha, Pytest),
- herramientas de cobertura de código (Istanbul, Coveralls),
- herramientas de pruebas de rendimiento (JMeter, Gatling),
- herramientas de pruebas de seguridad (OWASP ZAP, Burp Suite),
- herramientas de integración continua (Jenkins, CircleCI, GitHub Actions),
- recursos en línea
- comunidades de práctica.

### 6.1.2.6 Unidad curricular: Observabilidad

Formato: Asignatura

Régimen de cursada: Cuatrimestral

Ubicación en el diseño curricular: 3er año 1er Cuatrimestre

Asignación de horas semanales: 4 hs.

Total de horas: 64 hs.

#### Finalidades formativas y capacidades a desarrollar

Comprender el concepto de observabilidad y su importancia en entornos de DevOps y operaciones de software. Adquirir conocimientos sobre las técnicas y herramientas utilizadas para monitorear, registrar y analizar el comportamiento de aplicaciones y sistemas en producción, con el objetivo de identificar y resolver problemas de manera eficiente.

#### Ejes de contenidos. Descriptores

Observabilidad: definición, principios y beneficios. Monitoreo: métricas, alertas y paneles de control. Registro (logging): recopilación, almacenamiento y análisis de registros. Trazabilidad distribuida (distributed tracing): seguimiento de solicitudes a través de servicios distribuidos. Métricas de aplicación y de infraestructura. Alertas y notificaciones. Análisis de registros y detección de anomalías. Correlación de datos de observabilidad.

#### Prácticas Formativas

Para poder desarrollar la unidad curricular se realizan Prácticas Formativas mediante la instalación, configuración y uso de Herramientas, Servicios y Recursos comúnmente utilizados:

- herramientas de monitoreo (Prometheus, Datadog, New Relic),
- herramientas de registro (Elasticsearch, Logstash, Kibana, Graylog),
- herramientas de trazabilidad distribuida (Jaeger, Zipkin, AWS X-Ray),
- herramientas de análisis de registros (Splunk, Sumo Logic),
- herramientas de visualización (Grafana, Kibana),
- recursos en línea
- comunidades de práctica.

### 6.1.2.7 Unidad curricular: Seguridad Informática

Formato: Asignatura

Régimen de cursada: Anual

Ubicación en el diseño curricular: 3er año

Asignación de horas semanales: 4 hs.

Total de horas: 128 hs.

#### Finalidades formativas y capacidades a desarrollar

Comprender la importancia de la seguridad informática en el desarrollo y operación de aplicaciones y sistemas, y su relación con las prácticas de DevOps. Adquirir conocimientos sobre amenazas, vulnerabilidades, principios de seguridad y mejores prácticas para proteger los sistemas y la información. Desarrollar habilidades para implementar controles de seguridad, evaluar riesgos y responder a incidentes de seguridad en entornos DevOps.

#### Ejes de contenidos. Descriptores

Fundamentos de la seguridad informática: amenazas, vulnerabilidades y riesgos. Principios de seguridad: confidencialidad, integridad y disponibilidad. Seguridad en aplicaciones web y servicios en la nube. Seguridad en contenedores y Kubernetes. Controles de seguridad: autenticación, autorización, cifrado y firmas digitales. Análisis de vulnerabilidades y pruebas de penetración. Respuesta a incidentes de seguridad y gestión de riesgos. Seguridad en DevOps: integración de prácticas de seguridad en el ciclo de vida del desarrollo de software.

#### Prácticas Formativas

Para poder desarrollar la unidad curricular se realizan Prácticas Formativas mediante la instalación, configuración y uso de Herramientas, Servicios y Recursos comúnmente utilizados:

- herramientas de análisis de vulnerabilidades (OWASP ZAP, Nessus),
- herramientas de pruebas de penetración (Metasploit, Burp Suite),
- herramientas de seguridad en contenedores (Docker Bench, Falco),
- herramientas de gestión de identidades y accesos (Okta, Auth0),
- herramientas de cifrado y firmas digitales (OpenSSL, GPG),
- herramientas de gestión de incidentes de seguridad (TheHive, CIRTKit),

- recursos en línea
- comunidades de práctica sobre seguridad en DevOps y DevSecOps.

### 6.1.3\_FORMACIÓN ESPECÍFICA

#### 6.1.3.1 Unidad curricular: Aplicaciones Cloud Nativas

Formato: Asignatura

Régimen de cursada: Anual

Ubicación en el diseño curricular: 1er año

Asignación de horas semanales: 4 hs.

Total de horas: 128 hs.

#### Finalidades formativas y capacidades a desarrollar

Comprender los conceptos y principios fundamentales de las aplicaciones nativas de la nube (cloud-native), su importancia en el desarrollo de software moderno y su relación con la cultura DevOps. Adquirir conocimientos sobre arquitecturas de aplicaciones escalables, resilientes y distribuidas, utilizando patrones de diseño y tecnologías específicas para entornos de nube. Desarrollar habilidades para el diseño, implementación y operación de aplicaciones nativas de la nube, aprovechando las ventajas de la infraestructura en la nube y las prácticas de DevOps.

#### Ejes de contenidos. Descriptores

Conceptos y principios de las aplicaciones nativas de la nube: escalabilidad, resiliencia, distribución y automatización. Arquitecturas de microservicios y patrones de diseño. Contenedores y orquestación de contenedores (Docker, Kubernetes). Servicios gestionados en la nube (serverless, bases de datos, colas, etc.). Infraestructura como código y aprovisionamiento de recursos en la nube. Monitoreo y observabilidad de aplicaciones nativas de la nube. Prácticas de DevOps aplicadas al ciclo de vida de aplicaciones nativas de la nube.

#### Prácticas Formativas

Para poder desarrollar la unidad curricular se realizan Prácticas Formativas mediante la instalación, configuración y uso de Herramientas, Servicios y Recursos comúnmente utilizados:

- Docker,
- Kubernetes,
- servicios de nube (AWS, Google Cloud, Azure),
- herramientas de infraestructura como código (Terraform, Ansible),
- herramientas de monitoreo (Prometheus, Grafana),
- herramientas de observabilidad (Jaeger, Zipkin),
- entornos de desarrollo locales (Minikube, Docker Desktop),
- recursos en línea
- comunidades de práctica.

### 6.1.3.2 Unidad curricular: Automatización y Scripting

Formato: Asignatura

Régimen de cursada: Cuatrimestral

Ubicación en el diseño curricular: 1er año 1er Cuatrimestre

Asignación de horas semanales: 4 hs.

Total de horas: 64 hs.

#### Finalidades formativas y capacidades a desarrollar

Adquirir conocimientos y habilidades en el lenguaje de programación Python enfocado en su aplicación en tareas de automatización y scripting en entornos DevOps. Desarrollar la capacidad de escribir scripts y programas en Python para automatizar tareas repetitivas, procesar datos e interactuar con herramientas y servicios comúnmente utilizados en DevOps. Fomentar la comprensión de los conceptos de programación y algoritmos para su aplicación en la resolución de problemas y la mejora de la eficiencia en las operaciones de software.

#### Ejes de contenidos. Descriptores

Conceptos básicos de Python: sintaxis, tipos de datos, estructuras de control, funciones y módulos. Manejo de archivos y sistemas de archivos en Python. Scripting y automatización de tareas con Python. Interacción con APIs y servicios en la nube. Procesamiento de datos y análisis con librerías de Python (Pandas, NumPy). Programación orientada a objetos en Python. Pruebas unitarias y buenas prácticas de programación.

### Prácticas Formativas

Para poder desarrollar la unidad curricular se realizan Prácticas Formativas mediante la instalación, configuración y uso de Herramientas, Servicios y Recursos comúnmente utilizados:

- Python y sus librerías estándar,
- entornos de desarrollo integrados (IDEs) para Python (PyCharm, Visual Studio Code),
- gestores de paquetes (pip, conda),
- herramientas de pruebas unitarias (pytest, unittest),
- recursos en línea
- comunidades de práctica.

#### 6.1.3.3 Unidad curricular: Sistemas Operativos

Formato: Asignatura

Régimen de cursada: Anual

Ubicación en el diseño curricular: 1er año

Asignación de horas semanales: 4 hs.

Total de horas: 128 hs.

#### Finalidades formativas y capacidades a desarrollar

Comprender los conceptos fundamentales de los sistemas operativos, su importancia en la gestión de recursos y su papel en entornos DevOps. Adquirir conocimientos sobre los principales sistemas operativos utilizados en infraestructuras de TI, GNU-Linux, así como sus características, arquitectura y administración. Desarrollar habilidades prácticas en la configuración, mantenimiento y solución de problemas en sistemas operativos, con énfasis en tareas relevantes para DevOps, como la automatización, la virtualización y la seguridad.

#### Ejes de contenidos. Descriptores

Fundamentos de sistemas operativos: componentes, funciones y tipos. Sistemas de archivos y gestión de almacenamiento. Bash. Línea de comandos y comandos de Shell. Gestión de procesos y memoria. Administración de usuarios y permisos. Redes y comunicaciones. Package Manager.

Automatización de tareas con scripts y herramientas de configuración. Monitoreo y solución de problemas en sistemas operativos. Seguridad y endurecimiento de sistemas operativos. Virtualización y contenedores. S.O. de Nube. Virtual Private Cloud. IAM. Security Groups. Docker: Contenedores vs. Máquinas Virtuales, Imágenes de Docker, Docker Engine, Instalación, Configuración, Beneficios, Desarrollo y Pruebas de Software, Comandos Básicos, Logs, Docker Registry, Despliegue de Aplicaciones: Docker Compose. Escalabilidad y Gestión de Infraestructura. Orquestador: Kubernetes. Microservicios. Integración Continua y Entrega Continua (CI/CD).

### Prácticas Formativas

Para poder desarrollar la unidad curricular se realizan Prácticas Formativas mediante la instalación, configuración y uso de Herramientas, Servicios y Recursos comúnmente utilizados:

- Distribuciones de Linux (Ubuntu, CentOS, Debian),
- herramientas de línea de comandos (bash, PowerShell),
- herramientas de automatización (Ansible, Puppet),
- herramientas de virtualización (VirtualBox, VMware),
- herramientas de monitoreo (Nagios, Zabbix),
- herramientas de funcionalidad y gestión (Prometheus y Grafana),
- recursos en línea: (Docker Hub, Docker Compose y Docker Swarm) ELK (Elasticsearch, Logstash, Kibana),
- seguridad (Docker Bench for Security y Clair)
- AWS ECS (Elastic Container Service)
- Google Kubernetes Engine (GKE).

#### 6.1.3.4 Unidad curricular: Laboratorio I

Formato: Taller

Régimen de cursada: Cuatrimestral

Ubicación en el diseño curricular: 1er año 2do Cuatrimestre

Asignación de horas semanales: 4 hs.

Total de horas: 64 hs.

### Finalidades formativas y capacidades a desarrollar

Integrar y ampliar sus conocimientos teórico-prácticos de los conocimientos y habilidades adquiridos en las unidades curriculares previas a través de proyectos prácticos relacionados con DevOps. Fomentar la capacidad de trabajar en equipo, la resolución de problemas y la toma de decisiones en situaciones reales. Promover la investigación, el aprendizaje autónomo y la adaptación a nuevas tecnologías y herramientas en el campo de DevOps.

### Ejes de contenidos. Descriptores

Implementación de proyectos prácticos relacionados con las temáticas de la cultura DevOps, metodologías ágiles, control de versiones, bases de datos como servicio, automatización y scripting, y sistemas operativos. Trabajo en equipo y colaboración. Resolución de problemas y toma de decisiones. Investigación y aprendizaje autónomo. Adaptación a nuevas tecnologías y herramientas.

### Prácticas Formativas

Para poder desarrollar la unidad curricular se realizan Prácticas Formativas mediante la instalación, configuración y uso de Herramientas, Servicios y Recursos comúnmente utilizados:

- Git, GitHub, GitLab; Jira, Trello, Azure DevOps,
- Python y librerías estándar
- distribuciones de Linux (Ubuntu, CentOS, Debian)
- herramientas de línea de comandos (bash, PowerShell)
- herramientas de automatización (Ansible, Puppet)
- servicios de bases de datos en la nube (AWS RDS, Google Cloud SQL, Azure SQL Database)
- recursos en línea,
- documentación técnica
- comunidades de práctica.

### 6.1.3.5 Unidad curricular: Calidad, Pruebas y Automatización

Formato: Asignatura

Régimen de cursada: Anual

Ubicación en el diseño curricular: 2do año

Asignación de horas semanales: 4 hs.

Total de horas: 128 hs.

#### Finalidades formativas y capacidades a desarrollar

Comprender la importancia de la calidad del software y las pruebas en el ciclo de vida del desarrollo de aplicaciones, y su relación con las prácticas de DevOps. Adquirir conocimientos sobre diferentes tipos de pruebas, estrategias de pruebas y técnicas de automatización de pruebas. Desarrollar habilidades para el diseño, implementación y ejecución de pruebas automatizadas, así como gestionar estrategias para mejorar la eficiencia, confiabilidad y mantenibilidad de los sistemas.

#### Ejes de contenidos. Descriptores

Fundamentos de la calidad del software y las pruebas. Tipos de pruebas: unitarias, de integración, de extremo a extremo, de rendimiento, de seguridad, entre otras. Estrategias y técnicas de pruebas. Automatización de pruebas: frameworks, herramientas y prácticas. Integración de pruebas en el ciclo de vida de desarrollo de software y en las pipelines de CI/CD. Monitoreo y generación de informes de calidad. Pruebas exploratorias y basadas en comportamiento. Cultura de calidad y pruebas en DevOps.

#### Prácticas Formativas

Para poder desarrollar la unidad curricular se realizan Prácticas Formativas mediante la instalación, configuración y uso de Herramientas, Servicios y Recursos comúnmente utilizados:

- Frameworks de pruebas unitarias (JUnit, Pytest, Mocha),
- frameworks de pruebas de extremo a extremo (Selenium, Cypress),
- herramientas de pruebas de rendimiento (JMeter, Gatling),
- herramientas de integración continua (Jenkins, CircleCI, GitHub Actions),
- herramientas de monitoreo y generación de informes (SonarQube, Coveralls),
- herramientas de pruebas exploratorias (Exploratory Testing),

- recursos en línea
- comunidades de práctica.

### 6.1.3.6 Unidad curricular: Redes Virtuales

Formato: Asignatura

Régimen de cursada: Cuatrimestral

Ubicación en el diseño curricular: 2do año 1er Cuatrimestre

Asignación de horas semanales: 4 hs.

Total de horas: 64 hs.

#### Finalidades formativas y capacidades a desarrollar

Comprender los conceptos y tecnologías relacionados con las redes virtuales y su importancia en entornos de nube y DevOps. Adquirir conocimientos sobre la virtualización de redes, redes definidas por software (SDN) y redes virtuales superpuestas. Desarrollar habilidades para el diseño, implementación y gestión de redes virtuales, aprovechando las ventajas de la automatización y la programabilidad.

#### Ejes de contenidos. Descriptores

Fundamentos de redes virtuales y Cloud Networking. Redes definidas por software (SDN) y su arquitectura. Redes virtuales superpuestas y túneles. Servicios de red virtuales en la nube. Enrutamiento y direccionamiento en redes virtuales. Subnets, Security Groups, Internet Gateway, Networking, Route Table, AWS CLI. Seguridad y aislamiento en redes virtuales. Automatización y orquestación de redes virtuales.

#### Prácticas Formativas

Para poder desarrollar la unidad curricular se realizan Prácticas Formativas mediante la instalación, configuración y uso de Herramientas, Servicios y Recursos comúnmente utilizados:

- Plataformas de SDN (Open Daylight, ONOS),
- servicios de red virtual en la nube (AWS Virtual Private Cloud, Azure Virtual Network),
- herramientas de automatización (Ansible, Terraform),
- herramientas de monitoreo de redes (Prometheus, Grafana),

- recursos en línea
- comunidades de práctica.

### 6.1.3.7 Unidad curricular: Integración y Entrega Continua

Formato: Asignatura

Régimen de cursada: Cuatrimestral

Ubicación en el diseño curricular: 2do año 2do Cuatrimestre

Asignación de horas semanales: 4 hs.

Total de horas: 64 hs.

#### Finalidades formativas y capacidades a desarrollar

Comprender los conceptos y prácticas de la integración continua (CI) y la entrega continua (CD), y su importancia en el desarrollo y operación de software de alta calidad. Adquirir conocimientos sobre la automatización de los procesos de compilación, pruebas y despliegue, así como la creación de pipelines de CI/CD. Desarrollar habilidades para diseñar, implementar y mantener pipelines de CI/CD utilizando herramientas y servicios modernos.

#### Ejes de contenidos. Descriptores

Fundamentos de la integración continua y la entrega continua. Automatización de procesos de compilación, pruebas y despliegue. Pipelines de CI/CD: arquitectura y flujos de trabajo. Herramientas y servicios de CI/CD. Integración con sistemas de control de versiones y gestión de artefactos. Despliegues automáticos y estrategias de implementación. Monitoreo y retroalimentación en pipelines de CI/CD.

#### Prácticas Formativas

Para poder desarrollar la unidad curricular se realizan Prácticas Formativas mediante la instalación, configuración y uso de Herramientas, Servicios y Recursos comúnmente utilizados:

- herramientas de integración continua (Jenkins, Travis CI, CircleCI),
- servicios de CI/CD en la nube (AWS CodePipeline, Azure DevOps),
- gestores de artefactos (Artifactory, Nexus),
- herramientas de contenedores (Docker, Kubernetes),
- herramientas de monitoreo (Prometheus, Grafana),

- recursos en línea
- comunidades de práctica.

### 6.1.3.8 Unidad curricular: Laboratorio II

Formato: Taller

Régimen de cursada: Anual

Ubicación en el diseño curricular: 2do año.

Asignación de horas semanales: 4 hs.

Total de horas: 128 hs.

#### Finalidades formativas y capacidades a desarrollar

Aplicar e integrar los conocimientos y habilidades adquiridos en las unidades curriculares previas incluyendo las de primer año a través del desarrollo de proyectos prácticos más complejos relacionados con DevOps. Fomentar la capacidad de trabajar en equipo, la resolución de problemas y la toma de decisiones en situaciones reales. Promover la investigación, el aprendizaje autónomo y la adaptación a nuevas tecnologías y herramientas en el campo de DevOps.

#### Ejes de contenidos. Descriptores

Desarrollo de proyectos prácticos relacionados con las temáticas de calidad, pruebas y automatización, redes virtuales, integración y entrega continua, y aplicaciones cloud nativas. Trabajo en equipo y colaboración. Resolución de problemas y toma de decisiones. Investigación y aprendizaje autónomo. Adaptación a nuevas tecnologías y herramientas.

#### Prácticas Formativas

Para poder desarrollar la unidad curricular se realizan Prácticas Formativas mediante la instalación, configuración y uso de Herramientas, Servicios y Recursos comúnmente utilizados:

- Docker, Kubernetes,
- servicios de nube (AWS, Google Cloud, Azure),
- herramientas de infraestructura como código (Terraform, Ansible),
- herramientas de monitoreo (Prometheus, Grafana)
- herramientas de pruebas unitarias (JUnit, Pytest, Mocha)

- herramientas de pruebas de rendimiento (JMeter, Gatling)
- herramientas de integración continua (Jenkins, CircleCI, GitHub Actions)
- recursos en línea,
- documentación técnica
- comunidades de práctica.

### 6.1.3.9 Unidad curricular: Operaciones y Escalamiento

Formato: Asignatura

Régimen de cursada: Anual

Ubicación en el diseño curricular: 3er año

Asignación de horas semanales: 4 hs.

Total de horas: 128 hs.

#### Finalidades formativas y capacidades a desarrollar

Comprender los conceptos y prácticas relacionados con las operaciones de software y el escalamiento de aplicaciones en entornos de nube y DevOps. Adquirir conocimientos sobre monitoreo, registro, escalamiento automático, gestión de configuraciones y solución de problemas en entornos de producción. Desarrollar habilidades para diseñar, implementar y mantener operaciones eficientes y escalables para aplicaciones en la nube.

#### Ejes de contenidos. Descriptores

Fundamentos de las operaciones de software y el escalamiento. Monitoreo y registro en entornos de producción. Escalamiento automático y gestión de recursos. Gestión de configuraciones y cambios. Solución de problemas y resolución de incidentes. Prácticas de operaciones en DevOps y Site Reliability Engineering (SRE). Resiliencia y alta disponibilidad en aplicaciones escalables.

#### Prácticas Formativas

Para poder desarrollar la unidad curricular se realizan Prácticas Formativas mediante la instalación, configuración y uso de Herramientas, Servicios y Recursos comúnmente utilizados:

- herramientas de monitoreo (Prometheus, Datadog, New Relic),
- herramientas de registro (Elasticsearch, Logstash, Kibana),

- herramientas de escalamiento automático en la nube (AWS Auto Scaling, Kubernetes Horizontal Pod Autoscaler),
- herramientas de gestión de configuraciones (Ansible, Puppet),
- herramientas de gestión de incidentes (PagerDuty, OpsGenie),
- herramientas de pruebas de caos (Chaos Monkey, Gremlin),
- recursos en línea
- comunidades de práctica sobre SRE y operaciones de DevOps.

### 6.1.3.10 Unidad curricular: Laboratorio III

Formato: Taller

Régimen de cursada: Anual

Ubicación en el diseño curricular: 3er año

Asignación de horas semanales: 4 hs.

Total de horas: 128 hs.

#### Finalidades formativas y capacidades a desarrollar

Aplicar y ampliar los conocimientos y habilidades adquiridos en las unidades curriculares previas incluyendo las de primer y segundo año mediante la construcción y desarrollo de proyectos prácticos avanzados y complejos relacionados con DevOps. Fortalecer la capacidad de trabajar en equipo, la resolución de problemas y la toma de decisiones en situaciones reales. Promover la investigación, el aprendizaje autónomo y la adaptación a nuevas tecnologías y herramientas en el campo de DevOps. Preparar a los estudiantes para la inserción laboral en el ámbito de DevOps.

#### Ejes de contenidos. Descriptores

Puesta en marcha de proyectos prácticos relacionados con las temáticas de operaciones y escalamiento, seguridad informática, y otras áreas avanzadas de DevOps. Trabajo en equipo y colaboración. Resolución de problemas y toma de decisiones. Investigación y aprendizaje autónomo. Adaptación a nuevas tecnologías y herramientas. Preparación para la inserción laboral en el ámbito de DevOps.

#### Prácticas Formativas

Para poder desarrollar la unidad curricular se realizan Prácticas Formativas mediante la instalación, configuración y uso de Herramientas, Servicios y Recursos comúnmente utilizados:

- herramientas de monitoreo (Prometheus, Datadog, New Relic),
- herramientas de registro (Elasticsearch, Logstash, Kibana)
- herramientas de escalamiento automático en la nube (AWS Auto Scaling, Kubernetes HPA),
- herramientas de gestión de configuraciones (Ansible, Puppet),
- herramientas de análisis de vulnerabilidades (OWASP ZAP, Nessus),
- herramientas de pruebas de penetración (Metasploit, Burp Suite),
- herramientas de seguridad en contenedores (Docker Bench, Falco),
- recursos en línea,
- documentación técnica
- comunidades de práctica sobre DevOps y DevSecOps.

#### 6.1.4\_PRÁCTICAS PROFESIONALIZANTES

Las Prácticas Profesionalizantes de la carrera Técnico Superior en DevOps se desarrollan en un entorno híbrido que combina encuentros presenciales y virtuales dentro de la órbita institucional, aprovechando las ventajas de la tecnología y las plataformas digitales. Este enfoque permite replicar de manera efectiva la naturaleza remota y virtual inherente a la cultura DevOps y a los campos ocupacionales relacionados.

Durante estas prácticas, los/las estudiantes cuentan con plataformas cloud privadas y públicas, así como software y herramientas específicas utilizadas exclusivamente en el ámbito de DevOps. Estas herramientas les permiten familiarizarse con las tecnologías y los flujos de trabajo más actuales, preparándolos de manera óptima para los desafíos del mundo laboral.

La modalidad remota es fundamental para estas Prácticas Profesionalizantes, ya que refleja fielmente la naturaleza de los campos ocupacionales relacionados con DevOps, los cuales se caracterizan por el trabajo virtual y la colaboración remota entre equipos distribuidos geográficamente. Al sumergirse en este entorno durante su formación, los/las estudiantes adquieren habilidades y experiencias invaluableles que los ayudarán a adaptarse con facilidad a los desafíos y dinámicas de trabajo propios de la cultura DevOps.

Además, el enfoque remoto fomenta el desarrollo de habilidades esenciales, como la comunicación efectiva en entornos virtuales, la gestión de proyectos de manera remota, y la capacidad de trabajar de forma autónoma y autorregulada. Estas competencias son fundamentales para el éxito en el ámbito de DevOps, donde la agilidad, la colaboración y la adaptabilidad son pilares clave.

En resumen, las Prácticas Profesionalizantes de la carrera Técnico Superior en DevOps aprovechan las tecnologías y plataformas digitales para crear un entorno de aprendizaje práctico y realista, que prepara a los/las estudiantes para los desafíos y oportunidades de los campos ocupacionales relacionados con DevOps, reconociendo la importancia de la virtualidad y el trabajo remoto en esta disciplina.

#### 6.1.4.1 Unidad curricular: Práctica Profesionalizante I

Formato: Prácticas Profesionales

Régimen de cursada: Cuatrimestral

Ubicación en el diseño curricular: 1er año 2do Cuatrimestre

Asignación de horas semanales: 4 hs.

Total de horas: 64 hs.

#### Finalidades formativas y capacidades a desarrollar

La Práctica Profesionalizante I tiene como finalidad principal proporcionar a los/las estudiantes una introducción sólida al entorno profesional del campo de DevOps. Durante esta primera etapa, se busca que los/las estudiantes desarrollen una comprensión básica de las operaciones de TI y la cultura DevOps, integrando conocimientos teóricos con experiencias prácticas iniciales en escenarios controlados. Esta unidad curricular se orienta hacia el desarrollo de habilidades fundamentales en la automatización de procesos, la integración continua y la colaboración efectiva en equipos multidisciplinarios. Asimismo, se pretende que los/las estudiantes adquieran una perspectiva crítica sobre el impacto de las prácticas DevOps en la mejora de la eficiencia operativa y en la aceleración del ciclo de vida del desarrollo de software.

#### Ejes de contenidos. Descriptores

**El primer eje** de contenidos se centra en la Introducción a la Cultura DevOps y abarca los principios fundamentales de DevOps, su historia y evolución, así como la importancia de la colaboración y la comunicación entre equipos de

desarrollo y operaciones. Se explorarán los conceptos básicos de integración continua y entrega continua (CI/CD), y se estudiarán herramientas básicas utilizadas en estas prácticas.

**El segundo eje** se focaliza en la Automatización de Procesos Iniciales, donde los/las estudiantes aprenderán sobre la automatización de tareas repetitivas y la creación de scripts básicos utilizando lenguajes de programación comúnmente empleados en DevOps. Los descriptores incluyen la creación de scripts para la gestión de infraestructuras y la automatización de despliegues iniciales.

**El tercer eje** aborda la Colaboración en Entornos Controlados, donde se busca que los/las estudiantes trabajen en pequeños proyectos en equipo, aplicando los conocimientos adquiridos en un entorno simulado que refleje las dinámicas reales de un equipo DevOps.

### **Aspectos Metodológicos**

La metodología de enseñanza se basa en un enfoque práctico y experiencial. Se emplearán simulaciones como principales herramientas pedagógicas, permitiendo a los/las estudiantes aplicar los conceptos teóricos en situaciones prácticas controladas.

Las clases se estructurarán combinando sesiones teóricas con ejercicios prácticos intensivos, promoviendo la reflexión y el análisis crítico de los resultados obtenidos.

Los/las estudiantes participarán en proyectos colaborativos que simulen entornos de trabajo reales, fomentando el desarrollo de habilidades de trabajo en equipo y comunicación.

Además, se utilizarán estudios de caso para ilustrar la aplicación de principios DevOps en escenarios del mundo real, facilitando la conexión entre teoría y práctica.

### 6.1.4.2 Unidad curricular: Práctica Profesionalizante II

Formato: Prácticas Profesionales

Régimen de cursada: Anual

Ubicación en el diseño curricular: 2do año

Asignación de horas semanales: 4 hs.

Total de horas: 128 hs.

#### Finalidades formativas y capacidades a desarrollar

La Práctica Profesionalizante II tiene como objetivo profundizar en la experiencia práctica de los/las estudiantes, consolidando sus habilidades y conocimientos en entornos de trabajo más complejos y reales.

Durante esta fase, se espera que los/las estudiantes integren de manera más avanzada los conceptos y herramientas de DevOps en proyectos reales, abordando desafíos técnicos y colaborativos que reflejan la dinámica del sector.

La unidad se centra en el desarrollo de competencias avanzadas en automatización, monitoreo y gestión de infraestructuras, así como en la optimización de procesos y la mejora continua.

#### Ejes de contenidos. Descriptores

**El primer eje** se centra en la Automatización Avanzada y Gestión de Infraestructuras, abarcando técnicas y herramientas para la gestión avanzada de infraestructuras como código, el uso de plataformas de orquestación y la implementación de automatizaciones complejas. Los descriptores incluyen el uso de herramientas como Terraform, Ansible y Kubernetes.

**El segundo eje** aborda el Monitoreo y Gestión de Desempeño, donde los/las estudiantes aprenderán a implementar y gestionar sistemas de monitoreo y alertas para infraestructuras y aplicaciones, utilizando herramientas como Prometheus, Grafana y ELK Stack.

Se estudiará la importancia del monitoreo en la detección temprana de problemas y en la optimización del rendimiento.

**El tercer eje** se focaliza en la Optimización de Procesos y Mejora Continua, integrando prácticas de retroalimentación y análisis para la mejora constante

de los procesos DevOps. Los descriptores incluyen el uso de metodologías ágiles y lean, y la aplicación de análisis de post-mortem para la resolución de problemas y la mejora continua.

### **Aspectos Metodológicos**

La metodología se fundamenta en la participación activa y el aprendizaje basado en proyectos.

Los/las estudiantes se involucrarán en proyectos reales o simulados de mayor complejidad, donde deberán aplicar y profundizar en los conocimientos adquiridos.

Se promoverá el aprendizaje colaborativo mediante la formación de equipos multidisciplinarios que trabajen en la resolución de problemas reales, facilitando así el desarrollo de habilidades interpersonales y de comunicación. Además, se utilizarán plataformas de colaboración y herramientas de gestión de proyectos para replicar las condiciones de trabajo en la industria. La retroalimentación continua y las sesiones de revisión serán fundamentales para el aprendizaje, permitiendo a los estudiantes reflexionar sobre sus prácticas y mejorar continuamente.

#### **6.1.4.3 Unidad curricular: Práctica Profesionalizante III**

Formato: Prácticas Profesionales

Régimen de cursada: Anual

Ubicación en el diseño curricular: 3er año

Asignación de horas semanales: 4 hs.

Total de horas: 128 hs.

#### **Finalidades formativas y capacidades a desarrollar**

La Práctica Profesionalizante III tiene como finalidad culminar la formación práctica de los/las estudiantes, preparándolos/as para su inserción profesional en el campo de DevOps. Esta etapa se orienta a la aplicación integral y autónoma de todos los conocimientos y habilidades adquiridos a lo largo de la carrera, en proyectos complejos y de gran envergadura.

Se espera que los/as estudiantes demuestren competencias avanzadas en la gestión de infraestructuras, la automatización, el monitoreo, la seguridad y la

mejora continua, aplicando las mejores prácticas y adaptándose a diferentes contextos profesionales.

### Ejes de contenidos. Descriptores

**El primer eje** se focaliza en la Implementación Integral de Soluciones DevOps, abarcando la planificación, implementación y gestión de proyectos DevOps de principio a fin. Los descriptores incluyen la integración de todas las etapas del ciclo de vida de DevOps, desde el desarrollo y las pruebas hasta el despliegue y el mantenimiento.

**El segundo eje** se centra en la Seguridad en Entornos DevOps (DevSecOps), donde se estudiarán los principios y prácticas de seguridad en todas las etapas del ciclo de vida del software. Los descriptores incluyen la implementación de prácticas de seguridad desde el diseño hasta la producción, utilizando herramientas y técnicas como análisis de vulnerabilidades, pruebas de penetración y monitoreo continuo de seguridad.

**El tercer eje** aborda la Gestión de Proyectos y Liderazgo en Equipos DevOps, fomentando el desarrollo de habilidades de liderazgo y gestión de proyectos. Los descriptores incluyen la planificación y gestión de proyectos, la coordinación de equipos y la implementación de metodologías ágiles para la gestión efectiva de proyectos DevOps.

### Aspectos Metodológicos

La metodología de enseñanza se centrará en proyectos capstone, donde los/las estudiantes deberán planificar y ejecutar proyectos completos que integren todos los aspectos de DevOps. Estos proyectos serán supervisados por profesores profesionales de la industria, proporcionando una retroalimentación continua y oportunidades para la mejora. Se promoverá la autonomía y la responsabilidad en la gestión de los proyectos, preparando a los/las estudiantes para enfrentar desafíos profesionales de manera independiente. Además, se fomentará la participación en comunidades profesionales y eventos de la industria, facilitando el networking y la actualización continua de conocimientos. La evaluación se basará en la calidad y complejidad de los proyectos realizados, así como en la capacidad de los estudiantes para aplicar de manera crítica y creativa los conocimientos adquiridos.

## CAPÍTULO VII\_BIBLIOGRAFÍA GENERAL

- "DevOps: A Handbook for the Perplexed" de Gene Kim y Jez Humble (2010)
- "Continuous Delivery: Reliable Software Releases through Build, Test, and Deployment Automation" de Jez Humble y David Farley (2010)
- "The Phoenix Project: A Novel about IT, DevOps, and Helping Your Business Win" de Gene Kim, Kevin Behr, y George Spafford (2013)
- "The DevOps Handbook: How to Create World-Class Agility, Reliability, and Security in Technology Organizations" de Gene Kim y Jez Humble (2016)
- "DevOps: The Agile Way to Deliver Software" en Medium (2020)
- "What is DevOps?" en DevOps.com (2020)
- "DevOps: A Guide to Implementing DevOps in Your Organization" en AWS (2020)
- "DevOps: Best Practices for Implementing DevOps" en IBM (2020)
- "Operating Systems: Three Easy Pieces" by Remzi H. Arpaci-Dusseau and Andrea C. Arpaci-Dusseau (2015)
- "Linux Command Line and Shell Scripting Bible" by Richard Blum and Christine Bresnahan (2008)
- "Infrastructure as Code: Automate Your IT Infrastructure" por Arjen Lentjes (2012)
- "DevOps Culture and Practice with OpenShift" por David Worthington (2017)
- "Scrum: The Art of Doing Twice the Work in Half the Time" by Jeff Sutherland (2014)
- "Automate the Boring Stuff with Python" by Al Sweigart (2019)
- "Learning Python" by Mark Lutz: O'Reilly Media (2013)
- PROGRAMACION CON MICROSERVICIOS, FRANCISCO MIGUEL GARCIA MUÑOZ

- Microservices: Flexible Software Architecture, Eberhard Wolff (2016)
- Learn Microservices with Spring Boot, Moises Macero (2017)
- Advanced Microservices, Thomas Hunter II (2017)
- Redes de Datos y sus Protocolos, Mónica Cristina Liberatori (2018)
- Control de Versiones de Software con GIT, de Arturo Paz Arias (2015)
- Sistemas para el Control de Versiones, Jesús (2011)
- Scripting Python en Linux, Christophe BONNET (2021)
- Arquitecturas Cloud Nativas, Tom Laszewski, Kamal Arora, Erik Farr (2018)
- Un modelo de madurez para aplicaciones cloud nativas, Ignacio G.R. Gavilán (2019)
- Una introducción a las bases de datos en la nube, Wendy A. Neu, Vlad Vlasceanu, Andy Oram y Sam Alapati (2019)
- Refactoring: Improving the Design of Existing Code, de Martin Fowler, Kent Beck, John Brant, William Opdyke (1999)
- Software Testing Automation, Saeed Parsa (2022)
- Refactoring, Martin Fowler (1997)
- The Art of Software Testing, Glenford J. Myers, Corey Sandler y Tom Badgett (2011)
- Agile Testing: A Practical Guide for Testers and Agile Teams, Lisa Crispin y Janet Gregory (2008)
- DevOps en el desarrollo de software: integración y entrega continua, Cristian David Muñoz (2023)
- Entrega Continua, Katia (2020)  
Cómo implantar un proceso de integración continua, Mariano Minoli (2014)
- Hacking: The Art of Exploitation, 2nd Edition, Jon Erickson (2008)

- Big Breaches: Cybersecurity Lessons for Everyone, Neil Daswani, Moudy Elbayadi (2021)
- The Hacker Playbook 3: Practical Guide To Penetration Testing, Peter Kim (2018)
- Observabilidad: Todo lo que hay que ver, Isaac Ruiz Guerra (2020)
- Observabilidad y controlabilidad en modelos de evolución, Inmaculada López García (2004)
- Architecting for Scale: High Availability for Your Growing Applications, Lee Atchison (2018)
- The DevOps 2.0: Automating the Continuous Deployment Pipeline with Containerized Microservices, Viktor Farcic (2016)
- The Docker Book: Containerization is the new virtualization, James Turnbull (2019)
- Scalability Rules: Principles for Scaling Web Sites, Martin Abbott, Michael Fisher (2016)

### Referencias y Documentación web:

- "The DevOps Journal" (<https://www.sciencedirect.com/topics/computer-science/devops>)
- "InfoQ" (<https://www.infoq.com/>)
- "TechBeacon" (<https://techbeacon.com/>)
- "Red Hat Developer" (<https://developers.redhat.com/>)
- "Container Journal" (<https://cloudnativenow.com/features/welcome-to-container-journal/>)
- "DevOps.com" (<https://devops.com/>)
- "Atlassian DevOps" (<https://www.atlassian.com/solutions/devops>)
- "CloudBees" (<https://www.cloudbees.com/>)

- "CircleCI" (<https://circleci.com/>)
- "Jenkins" (<https://www.jenkins.io/>)
- "DevOps Roadmap" por ThoughtWorks  
(<https://www.thoughtworks.com/radar/techniques/devops>)
- "The State of DevOps 2023" por DORA (<https://dora.dev/>)
- "Gartner DevOps Magic Quadrant"  
(<https://www.atlassian.com/gartner/magic-quadrant-devops>)
- "DevOps Glossary" por Red Hat  
(<https://www.redhat.com/en/topics/devops>)
- "DevOps Tools Landscape" por TechRadar Pro  
(<https://www.techradar.com/best/best-devops-tools>)
- Herramientas de DevOps a Tener en Cuenta en 2024  
<https://kinsta.com/es/blog/herramientas-devops/#cules-son-las-herramientas-devops>
- ¿Qué es DevOps?  
<https://aws.amazon.com/es/devops/what-is-devops/>
- Operaciones de desarrollo y AWS  
<https://aws.amazon.com/es/devops/>

## CAPÍTULO VIII\_GLOSARIO TÉCNICO - DevOps

**Agile (Metodologías Ágiles):** Conjunto de metodologías de desarrollo de software basadas en el desarrollo iterativo e incremental, donde los requisitos y soluciones evolucionan a través de la colaboración entre equipos autoorganizados.

**Automatización de Pruebas:** Uso de software para realizar pruebas repetitivas en una aplicación, reduciendo el esfuerzo manual y mejorando la cobertura de pruebas.

**Automatización y Scripting:** Uso de scripts y herramientas de automatización para realizar tareas repetitivas y gestionar la infraestructura y las aplicaciones de manera eficiente.

**Backlog:** Lista priorizada de tareas pendientes o funcionalidades por implementar en un proyecto ágil.

**BD como Servicio (Bases de Datos como Servicio):** Modelo de servicio donde el proveedor ofrece una base de datos completa como un servicio gestionado, permitiendo a los usuarios evitar la administración física del hardware y software.

**Bitbucket:** Plataforma de gestión de repositorios Git que facilita la colaboración en proyectos de software.

**CI/CD (Integración Continua / Despliegue Continuo):** Prácticas de integración y entrega continua de cambios de código, asegurando que el software pueda ser desplegado automáticamente en cualquier momento.

**Colaboración:** Trabajo conjunto entre equipos de desarrollo y operaciones para lograr objetivos comunes en el ciclo de vida del software.

**Comunicación Abierta:** Flujo constante de información entre equipos y partes interesadas para asegurar que todos estén alineados y puedan colaborar eficazmente.

**Configuración Manual:** Proceso de establecer manualmente las configuraciones de hardware y software en lugar de utilizar archivos de configuración automatizados.

**Contenedores (Docker, Kubernetes):** Tecnologías que permiten empaquetar y ejecutar aplicaciones de manera consistente en diferentes entornos.

**Control de Versiones:** Sistema que permite gestionar y registrar los cambios realizados en el código fuente y otros documentos, facilitando la colaboración entre desarrolladores y el mantenimiento del historial del proyecto.

**Despliegue Automático:** Proceso de lanzar nuevas versiones de software a producción sin intervención manual, generalmente utilizando pipelines de CI/CD.

**DevOps:** Conjunto de prácticas que unifican el desarrollo de software (Dev) y las operaciones de TI (Ops), con el objetivo de mejorar y acelerar la entrega de aplicaciones y servicios.

**DevSecOps:** Integración de prácticas de seguridad en todas las etapas del ciclo de vida del desarrollo de software, asegurando que la seguridad esté presente desde el diseño hasta la producción.

**Entrega Continua:** Práctica de ingeniería de software donde los equipos producen software en ciclos cortos, asegurando que el software pueda ser liberado en cualquier momento.

**Escalamiento (Operaciones y Escalamiento):** Capacidad de un sistema para manejar una cantidad creciente de trabajo o su potencial para ser ampliado para acomodar dicho crecimiento.

**Git:** Sistema de control de versiones distribuido que permite a múltiples desarrolladores trabajar en el mismo proyecto de manera eficiente.

**GitHub:** Plataforma de hospedaje de repositorios Git, utilizada para la gestión de código y la colaboración en proyectos de software.

**GitLab:** Plataforma que ofrece repositorios Git y herramientas para la integración continua y la entrega continua (CI/CD).

**Herramientas de Infraestructura como Código (Terraform, Ansible, Puppet):** Herramientas que permiten gestionar la infraestructura mediante definiciones de código, facilitando la automatización y repetibilidad.

**Herramientas de Integración Continua (Jenkins, Travis CI, CircleCI):** Plataformas que facilitan la automatización de la integración y prueba continua del código fuente.

**Herramientas de Monitoreo (Prometheus, Grafana, ELK Stack):** Herramientas que permiten la recopilación, visualización y análisis de métricas de rendimiento y logs de sistemas y aplicaciones.

**Herramientas de Registro (Splunk, Graylog):** Sistemas que centralizan y analizan los logs generados por aplicaciones y sistemas para detectar y resolver problemas.

**Infraestructura como Código (IaC):** Práctica de gestionar y aprovisionar la infraestructura mediante archivos de configuración, en lugar de configuraciones manuales.

**Integración Continua (CI):** Práctica de integrar regularmente el código de trabajo en un repositorio compartido, seguido de pruebas automáticas.

**Kanban:** Método de gestión de proyectos ágiles que visualiza el flujo de trabajo y limita el trabajo en progreso para mejorar la eficiencia.

**Monitoreo y Observabilidad:** Conjunto de prácticas y herramientas que permiten entender el estado interno de un sistema basado en los datos de los logs, métricas y trazas.

**Plataformas de Colaboración (Slack, Microsoft Teams):** Herramientas que facilitan la comunicación y colaboración entre equipos, especialmente en entornos distribuidos.

**Práctica Profesionalizante:** Experiencia de aprendizaje que proporciona a los estudiantes la oportunidad de aplicar y profundizar los conocimientos adquiridos en un entorno profesional real.

**Refactoring:** Técnica de mejora del código fuente que implica modificar su estructura interna sin cambiar su comportamiento externo, con el objetivo de mejorar su legibilidad y mantenibilidad.

**Scrum:** Marco de trabajo ágil que se basa en equipos pequeños que trabajan en ciclos cortos (sprints) para entregar incrementos de producto funcionales.  
**Seguridad Informática:** Prácticas y tecnologías diseñadas para proteger los sistemas informáticos y la información contra el acceso no autorizado, daños o ataques.

**Sistemas Operativos:** Software que administra el hardware de una computadora y proporciona servicios para los programas de aplicación.

**Testing (Pruebas):** Proceso de evaluación de un sistema o sus componentes con el propósito de encontrar si satisface los requisitos especificados o identificar diferencias entre las condiciones esperadas y las reales.



Río  
Negro