

1. Competencias	Construir soluciones de software y sistemas inteligentes mediante la gestión de proyectos, integración de metodologías, modelos y herramientas de desarrollo bajo la normatividad aplicable para la optimización de proyectos de investigación, innovación, desarrollo tecnológico y de emprendimiento.
2. Cuatrimestre	Noveno
3. Horas Teóricas	29
4. Horas Prácticas	61
5. Horas Totales	90
6. Horas Totales por Semana Cuatrimestre	6
7. Objetivo de aprendizaje	El alumno optimizará el proceso, desarrollo y liberación de componentes de software mediante la integración de metodologías, modelos, herramientas y servicios para la implementación de aplicaciones WEB empresariales seguras.

Unidades de Aprendizaje	Horas		
	Teóricas	Prácticas	Totales
I. Definición del proceso de desarrollo WEB	6	6	12
II. Control de Versiones	5	12	17
III. Integración de componentes de software para aplicaciones Web	10	25	35
IV. Pruebas y liberación del desarrollo WEB	8	18	26
Totales	29	61	90

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

DESARROLLO WEB INTEGRAL

UNIDADES DE APRENDIZAJE

1. Unidad de aprendizaje	I. Definición del proceso de desarrollo WEB
2. Horas Teóricas	6
3. Horas Prácticas	6
4. Horas Totales	12
5. Objetivo de la Unidad de Aprendizaje	El alumno determinará las metodologías, arquitecturas, patrones de diseño y frameworks para definir los procesos de desarrollo WEB.

Temas	Saber	Saber hacer	Ser
Selección de Metodología ágil.	Reconocer las características de las metodologías de desarrollo ágiles.	<p>Seleccionar la metodología de desarrollo ágil.</p> <p>Definir el esquema de pruebas.</p> <p>Planear el proceso de desarrollo WEB de acuerdo a la metodología seleccionada.</p>	<p>Comunicación eficiente</p> <p>Analítico</p> <p>Sentido de la planificación</p> <p>Razonamiento lógico</p>
Especificación de la Arquitectura.	<p>Reconocer los modelos de arquitecturas de software.</p> <p>Reconocer las herramientas de modelado de software.</p>	<p>Justificar la arquitectura de software del desarrollo WEB.</p> <p>Esquematizar la arquitectura definida.</p>	<p>Comunicación eficiente</p> <p>Analítico</p> <p>Sentido de la planificación</p> <p>Razonamiento lógico</p>
Selección de Patrones de diseño.	Reconocer las características de los patrones de diseño.	Proponer los patrones de diseño estructurales, de comportamiento, creacionales y emergentes a utilizar en el desarrollo WEB.	<p>Comunicación eficiente</p> <p>Analítico</p> <p>Sentido de la planificación</p> <p>Razonamiento lógico</p>
Selección de Frameworks.	Reconocer los principales Frameworks de desarrollo WEB.	Justificar los Frameworks del desarrollo WEB a utilizar .	<p>Comunicación eficiente</p> <p>Analítico</p> <p>Sentido de la planificación</p>

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

		Realizar la instalación y configuración del entorno de desarrollo WEB	Razonamiento lógico
--	--	---	---------------------

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

DESARROLLO WEB INTEGRAL

PROCESO DE EVALUACIÓN

Resultado de aprendizaje	Secuencia de aprendizaje	Instrumentos y tipos de reactivos
<p>Genera un documento a partir de un caso de estudio el cual incluya:</p> <ul style="list-style-type: none">- Plan del proceso de desarrollo WEB de acuerdo a la metodología ágil seleccionada.- Justificación de la arquitectura.- Diagrama de la arquitectura.- Propuesta de los patrones de diseño.- Justificación de los frameworks de desarrollo WEB a utilizar.- Esquema de pruebas. <p>Genera un reporte con la configuración del entorno que incluya:</p> <ul style="list-style-type: none">- Lista de herramientas utilizadas.- Parámetros de configuración.	<ol style="list-style-type: none">1. Identificar la metodología ágil adaptable al desarrollo.2. Analizar las características de la arquitectura de software aplicable.3. Comprender los patrones de diseño aplicables.4. Comprender las características de frameworks de desarrollo.5. Relacionar los componentes con la definición del proceso de desarrollo WEB.	<ol style="list-style-type: none">1. Estudio de casos.2. Listas de cotejo.

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

DESARROLLO WEB INTEGRAL

PROCESO ENSEÑANZA APRENDIZAJE

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

ESPACIO FORMATIVO

Aula	Laboratorio / Taller	Empresa
X	X	

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

DESARROLLO WEB INTEGRAL

UNIDADES DE APRENDIZAJE

1. Unidad de aprendizaje	II. Control de Versiones
2. Horas Teóricas	5
3. Horas Prácticas	12
4. Horas Totales	17
5. Objetivo de la Unidad de Aprendizaje	El alumno estructurará un repositorio de código fuente para optimizar el control de versiones de software.

Temas	Saber	Saber hacer	Ser
Versionamiento .	<p>Definir el concepto de versionamiento de software.</p> <p>Identificar las características de los sistemas de control de versiones:</p> <ul style="list-style-type: none"> - Git - SVN <p>Identificar las plataformas y herramientas de versionamiento:</p> <ul style="list-style-type: none"> - GitHub. - GitLab. - bitbucket. 	<p>Determinar las plataformas y herramientas de versionamiento a utilizar.</p> <p>Realizar la configuración de las plataformas y herramientas de versionamiento.</p>	<p>Razonamiento lógico</p> <p>Razonamiento cognitivo</p> <p>Sentido de la planificación</p> <p>Organizado</p> <p>Capacidad de autoaprendizaje</p>
Flujos de trabajo.	<p>Definir el concepto de flujo de trabajo para el control de versiones.</p> <p>Identificar los flujos de trabajo para el control de versiones:</p> <ul style="list-style-type: none"> - Git Flow. - GitHub Flow. - GitLab Flow. - One Flow. 	<p>Establecer estrategias de:</p> <ul style="list-style-type: none"> - Nomenclatura de Ramas. - Acceso a usuarios. - Políticas de combinación de ramas. - Políticas de peticiones Pull request. <p>Controlar las versiones de código fuente:</p>	<p>Razonamiento lógico</p> <p>Razonamiento cognitivo</p> <p>Sentido de la planificación</p> <p>Organizado</p> <p>Capacidad de autoaprendizaje</p>

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

	Describir las características de los flujos de trabajo.	<ul style="list-style-type: none"> - Línea de comandos. - Plataforma de versionamiento. - Herramientas integradas en el entorno de desarrollo (IDE). - Herramientas gráficas. 	
--	---	---	--

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

DESARROLLO WEB INTEGRAL

PROCESO DE EVALUACIÓN

Resultado de aprendizaje	Secuencia de aprendizaje	Instrumentos y tipos de reactivos
<p>Genera un documento a partir de un caso de estudio el cual incluya:</p> <ul style="list-style-type: none">- Justificación de las plataformas y herramientas de versionamiento.- Flujo de trabajo del control de versiones.- Parámetros de configuración de las plataformas y herramientas de versionamiento.- Enlace del repositorio en funcionamiento con la estructura del flujo de trabajo.	<ol style="list-style-type: none">1. Identificar el concepto de versionamiento.2. Identificar las plataformas y herramientas de versionamiento a utilizar.3. Comprender el flujo de trabajo.4. Comprender el control de las versiones de código fuente.	<ol style="list-style-type: none">1. Estudio de casos.2. Listas de cotejo.

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

DESARROLLO WEB INTEGRAL

PROCESO ENSEÑANZA APRENDIZAJE

Métodos y técnicas de enseñanza	Medios y materiales didácticos
<ul style="list-style-type: none">- Análisis de casos.- Prácticas en laboratorio.- Equipos colaborativos.	<ul style="list-style-type: none">- Equipos de cómputo.- Proyector.- Internet.- Pizarrón y marcadores.- Plataformas virtuales.- Ejercicios prácticos.- Frameworks.- Entornos de desarrollo.

ESPACIO FORMATIVO

Aula	Laboratorio / Taller	Empresa
	X	

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

DESARROLLO WEB INTEGRAL

UNIDADES DE APRENDIZAJE

1. Unidad de aprendizaje	III. Integración de componentes de software para aplicaciones Web
2. Horas Teóricas	10
3. Horas Prácticas	25
4. Horas Totales	35
5. Objetivo de la Unidad de Aprendizaje	El alumno integrará componentes de software seguros para el intercambio de información entre aplicaciones.

Temas	Saber	Saber hacer	Ser
Especificación de principios de codificación segura .	<p>Reconocer los lineamientos de la protección de datos personales.</p> <p>Reconocer los principios de codificación segura.</p> <p>Identificar los puntos de vulnerabilidad en aplicaciones WEB.</p> <p>Identificar certificados de seguridad de sitios WEB: - SSL. - TLS.</p>	Implementar los mecanismos de seguridad en el desarrollo WEB.	<p>Responsabilidad</p> <p>Ética profesional</p> <p>Analítico</p> <p>Razonamiento lógico</p> <p>Organizado</p> <p>Trabajo en equipo</p> <p>Autonomía</p> <p>Comunicación eficiente</p> <p>Proactivo</p>
Implementación de WEB services en el desarrollo WEB.	<p>Reconocer los conceptos de WEB services: - SOAP. - REST.</p> <p>Identificar API's de terceros aplicables al desarrollo WEB.</p> <p>Identificar los mecanismos para el desarrollo de WEB services propios.</p>	<p>Realizar la integración de API's de terceros al desarrollo WEB.</p> <p>Realizar la implementación de WEBServices propios en el desarrollo WEB.</p> <p>Realizar la implementación de mecanismos de autenticación remota a WEB services.</p>	<p>Responsabilidad</p> <p>Ética profesional</p> <p>Analítico</p> <p>Razonamiento lógico</p> <p>Organizado</p> <p>Trabajo en equipo</p> <p>Autonomía</p> <p>Comunicación eficiente</p> <p>Proactivo</p>

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

	Identificar los mecanismos de autenticación remota de WEB services.		
--	---	--	--

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

DESARROLLO WEB INTEGRAL

PROCESO DE EVALUACIÓN

Resultado de aprendizaje	Secuencia de aprendizaje	Instrumentos y tipos de reactivos
<p>Desarrolla una aplicación WEB utilizando un repositorio de código fuente, con base en un caso de estudio de acuerdo a la metodología ágil, la arquitectura de software, los patrones de diseño, frameworks y esquema de pruebas seleccionados, que incluya:</p> <ul style="list-style-type: none">- Mecanismos de seguridad.- WEB Services de terceros.- WEB Services propios- Enlace del repositorio en funcionamiento.	<ol style="list-style-type: none">1. Explicar los principios de codificación segura.2. Explicar los conceptos de WEB services.3. Comprender el uso de API's de terceros y propios.4. Comprender los mecanismos de autenticación remota.5. Relacionar los componentes de software.	<ol style="list-style-type: none">1. Estudio de casos.2. Listas de cotejo.

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

DESARROLLO WEB INTEGRAL

PROCESO ENSEÑANZA APRENDIZAJE

Métodos y técnicas de enseñanza	Medios y materiales didácticos
<ul style="list-style-type: none">- Análisis de casos.- Prácticas en laboratorio.- Equipos colaborativos.	<ul style="list-style-type: none">- Equipos de cómputo.- Proyector.- Internet.- Pizarrón y marcadores.- Plataformas virtuales.- Ejercicios prácticos.- Frameworks.- Entornos de desarrollo.

ESPACIO FORMATIVO

Aula	Laboratorio / Taller	Empresa
	X	

10.

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

DESARROLLO WEB INTEGRAL

UNIDADES DE APRENDIZAJE

1. Unidad de aprendizaje	IV. Pruebas y liberación del desarrollo WEB
2. Horas Teóricas	8
3. Horas Prácticas	18
4. Horas Totales	26
5. Objetivo de la Unidad de Aprendizaje	El alumno realizará el despliegue de servicios y pruebas para la liberación del desarrollo WEB.

Temas	Saber	Saber hacer	Ser
Despliegue de servicios.	<p>Identificar el concepto de contenedores.</p> <p>Diferenciar las herramientas para la gestión de contenedores:</p> <ul style="list-style-type: none"> - Docker. - AWS. - Azure. - Google cloud. <p>Identificar los servicios de infraestructura para cómputo en la nube.</p>	<p>Realizar la implementación de los contenedores para desplegar los componentes WEB.</p> <p>Realizar la configuración de la infraestructura de los servicios para cómputo en la nube:</p> <ul style="list-style-type: none"> - Servidor de aplicaciones. - Servidor de base de datos. - Balanceo de cargas. - Configuración de dominio. - Certificados de seguridad. 	<p>Responsabilidad</p> <p>Analítico</p> <p>Autonomía</p> <p>Proactivo</p> <p>Sentido de la planificación</p>
Pruebas de software.	<p>Reconocer los tipos pruebas de software:</p> <ul style="list-style-type: none"> - Caja blanca. - Caja negra. - Unitarias. - Integración. - Regresión. - Rendimiento. - Esfuerzo. - Usabilidad. <p>Diferenciar las herramientas de pruebas de software:</p>	<p>Realizar la pruebas software con las herramientas seleccionadas.</p> <p>Realizar la automatización de pruebas de software.</p>	<p>Responsabilidad</p> <p>Analítico</p> <p>Autonomía</p> <p>Proactivo</p> <p>Sentido de la planificación</p>

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

Liberación del software.	<ul style="list-style-type: none"> - Identificar las políticas y normativa aplicable para la liberación del software. - Diferenciar las herramientas de liberación de software. 	Realizar la publicación del desarrollo WEB.	Responsabilidad Analítico Autonomía Proactivo Sentido de la planificación
--------------------------	---	---	---

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

DESARROLLO WEB INTEGRAL

PROCESO DE EVALUACIÓN

Resultado de aprendizaje	Secuencia de aprendizaje	Instrumentos y tipos de reactivos
<p>Genera un contenedor que incluya la Aplicación WEB desplegada a partir de un caso de estudio.</p> <p>Genera la infraestructura para el despliegue de la aplicación WEB en servicios de cómputo en la nube considerando:</p> <ul style="list-style-type: none">- Servidor de aplicaciones.- Servidor de base de datos.- Balanceo de cargas.- Configuración de dominio.- Certificados de seguridad. <p>Realiza la publicación de una aplicación WEB a partir de un caso de estudio considerando las pruebas necesarias para la liberación:</p> <ul style="list-style-type: none">- Enlace del repositorio en funcionamiento.	<ol style="list-style-type: none">1. Comprender las características de los servicios de cómputo en la nube.2. Identificar el proceso de implementación de contenedores.3. Comprender la ejecución de pruebas automatizadas.4. Comprender el proceso de liberación de unidades de software.	<ol style="list-style-type: none">1. Estudio de casos.2. Listas de cotejo.

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

DESARROLLO WEB INTEGRAL

PROCESO ENSEÑANZA APRENDIZAJE

Métodos y técnicas de enseñanza	Medios y materiales didácticos
<ul style="list-style-type: none">- Análisis de casos.- Prácticas en laboratorio.- Equipos colaborativos.	<ul style="list-style-type: none">- Equipos de cómputo.- Proyector.- Internet.- Pizarrón y marcadores.- Plataformas virtuales.- Ejercicios prácticos.- Frameworks.- Entornos de desarrollo.

ESPACIO FORMATIVO

Aula	Laboratorio / Taller	Empresa
	X	

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

DESARROLLO WEB INTEGRAL

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

Capacidad	Criterios de Desempeño
Determinar el tipo de proyecto de software mediante el análisis de un problema para determinar la forma de desarrollo.	Entrega un documento de definición del proyecto que incluya: <ul style="list-style-type: none"> - Descripción de la problemática. - Objetivos establecidos. - Metas y alcances. - Requerimientos. - Justificación del tipo de proyecto.
Gestionar el desarrollo de software mediante el seguimiento de la codificación y ejecución de pruebas para la integración continua del producto.	<ol style="list-style-type: none"> 1. Entrega los archivos de configuración del entorno de producción. 2. Entrega un documento que incluya: <ul style="list-style-type: none"> - Bitácoras del desarrollo de software. - Reportes de seguimiento del desarrollo de software. - Evidencias de control de cambios y versiones del código. - Resultado de pruebas.
Realizar la liberación de unidades estables de software mediante un plan de entregas continuas para contribuir al cumplimiento de los objetivos del proyecto.	<ol style="list-style-type: none"> 1. Entrega los resultados de las pruebas de sistema en entorno de operación. 2. Entrega un documento que incluya: <ul style="list-style-type: none"> - Evidencias del seguimiento de la ejecución del plan de entregas continuas. - Evidencia de acuerdo de aceptación. 3. Entrega archivos que incluya: <ul style="list-style-type: none"> - Unidades estables de software a producción.
Diseñar interfaces de usuario mediante el uso de herramientas y principios de usabilidad para optimizar la experiencia del usuario.	<ol style="list-style-type: none"> 1. Entrega un documento que incluya: <ul style="list-style-type: none"> - Justificación de las herramientas para el diseño de las interfaces. - Maquetado de interfaces. - Modelos de navegación. - Guía de estilo de interfaces. 2. Entrega archivos de las interfaces.

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

<p>Implementar modelos de datos mediante herramientas y sistemas gestores para garantizar la disponibilidad e integridad de la información.</p>	<p>1. Entrega un documento que incluya:</p> <ul style="list-style-type: none"> - Justificación de las herramientas a utilizar para los modelos de datos. - Modelos de datos. - Descripción de metadatos. <p>2. Entrega archivos, credenciales de registro y secuencia de configuración para la creación de los modelos de datos.</p>
<p>Desarrollar componentes mediante el uso de patrones de diseño, APIs y frameworks de desarrollo conforme a la arquitectura establecida para la integración continua de unidades del software.</p>	<p>1. Entrega un documento que incluya:</p> <ul style="list-style-type: none"> - Justificación de las herramientas para el desarrollo de componentes. - Justificación de los patrones de diseño a utilizar. - Justificación de los lenguajes de programación a utilizar. <p>2. Entrega archivos de código fuente y configuración de los componentes desarrollados.</p> <p>3. Entrega evidencia de almacenamiento de archivos de código fuente y actualizaciones en la documentación del proyecto en la plataforma de versionamiento seleccionada.</p>
<p>Implementar esquemas de seguridad mediante codificación, estándares, protocolos, herramientas e infraestructura para garantizar la privacidad y confidencialidad de la información cumpliendo con leyes y regulaciones aplicables.</p>	<p>1. Entrega un documento que incluya:</p> <ul style="list-style-type: none"> - Listado de las leyes y regulaciones aplicables al proyecto de desarrollo de software. - Descripción de acciones encaminadas a cumplir las leyes y regulaciones aplicables al proyecto de desarrollo de software. - Descripción de estándares, protocolos, herramientas e infraestructura para garantizar la privacidad y confidencialidad de la información del proyecto de desarrollo de software. - Reporte de pruebas de seguridad. <p>2. Entrega archivos de código fuente y configuración de los esquemas de seguridad.</p>
<p>Ejecutar pruebas a componentes de software a través del uso de técnicas y herramientas de testing para asegurar su correcto funcionamiento.</p>	<p>1. Entrega un documento que incluya:</p> <ul style="list-style-type: none"> - Justificación de las técnicas y herramientas utilizadas. - Descripción de las pruebas. - Informe de resultados del plan de pruebas. <p>2. Entrega archivos de código fuente y configuración del componente.</p>

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

<p>Implementar soluciones de software mediante la integración de componentes de acuerdo a la arquitectura definida para su liberación en un ambiente de producción.</p>	<ol style="list-style-type: none"> 1. Entrega manuales del software establecidos en el plan de desarrollo. 2. Entrega la versión estable del software integrando los componentes de acuerdo al plan de entregas continuas. 3. Entrega evidencia de almacenamiento de archivos de código fuente y actualizaciones de la documentación del proyecto en la plataforma de versionamiento establecida.
<p>Determinar la arquitectura de software mediante el análisis de los requerimientos, la definición de los componentes y la relación entre ellos para satisfacer los atributos de calidad del software y servir como guía en el desarrollo.</p>	<p>Entrega un documento de arquitectura de software que incluya:</p> <ul style="list-style-type: none"> - Requerimientos funcionales y no funcionales de desarrollo de software. - Justificación de la arquitectura seleccionada. - Diagrama de arquitectura de software. - Características y funciones de los componentes. - Diagramas necesarios de acuerdo a la arquitectura seleccionada.
<p>Implementar planes de desarrollo y pruebas mediante la metodología y herramientas definidas, incorporando normas y estándares aplicables para atender los requerimientos establecidos.</p>	<ol style="list-style-type: none"> 1. Entrega un plan de trabajo que incluya: <ul style="list-style-type: none"> - Cronograma de actividades. - Plan de entregas continuas. - Roles y responsabilidades. - Requisitos del entorno de desarrollo y producción. - Definición de manuales a entregar. 2. Entrega el plan de pruebas, de acuerdo a la metodología seleccionada: <ul style="list-style-type: none"> - Tipos de pruebas. - Justificación de las herramientas de pruebas. - Criterios de aceptación y liberación. - Cronograma de pruebas. - Instrumentos de pruebas. - Requisitos del entorno de pruebas.

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

DESARROLLO WEB INTEGRAL

FUENTES BIBLIOGRÁFICAS

Autor	Año	Título del Documento	Ciudad	País	Editorial
José Manuel Ortega Candel	2020 ISBN: 978-8426728005	<i>Desarrollo seguro en ingeniería del software</i>	Barcelona	España	Marcombo
Ortega Cangel, José Manuel	2018 ISBN:978-84-9964-732-6	<i>Seguridad en Aplicaciones Web Java</i>	Madrid	España	Ra-Ma
Fernando Luna, Claudio Peña Millahual, Matías Iacono	2019 ASIN:B07ZG2KV4J	<i>PROGRAMACIÓN WEB Full Stack - Versionar el desarrollo: Git y GitHub</i>			Creative Andina Corp
Iuliana Cosmina y Rob Harrop	Año: 2017 ISBN 9781484228074	<i>Pro Spring 5: An In-Depth Guide to the Spring Framework and Its Tools</i>	Dallas	Estados Unidos	Apress
Rhuan Rocha	Año: 2018 ISBN 9781788830621	<i>Java EE 8 Design Patterns and Best Practices</i>	Birmingham	Reino Unido	Packt Publishing Ltd.
Ganesan Senthilvel	Año: 2017 ISBN 9781786468888	<i>Enterprise Application Architecture with .NET Core</i>	Birmingham	Reino Unido	Packt Publishing Ltd.
Chacon, Scott; Straub, Benn	2021 ISBN: 978-1484200773	<i>Pro Git: Todo lo que necesitas saber acerca de Git (Segunda Edición)</i>		Estados Unidos	Apress

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

Caballero González, Carlos	2016 ISBN-13: 9788428396929	<i>Pruebas de funcionalidad y optimización de páginas web</i>	Madrid	España	Ediciones Paraninfo, S.A.
Edge, James	2020 ISBN-10: 1647486440 ISBN-13: 978-1647486440	<i>Agile: Una guía para la Gestión de Proyectos Agile con Scrum, Kanban y Lean</i>		Estados Unidos	Bravex Publications
Bill Laboon	2016 ISBN 9781523477371	<i>A Friendly Introduction to Software Testing</i>	NJ	Estados Unidos	CreateSpace Independent Publishing Platform
Rex Black	2016 ISBN 9781937538682	<i>Advanced Software Testing - Vol. 1,2,3, 2nd Edition: Guide to the ISTQB Advanced Certification as an Advanced Test Analyst</i>	Illinois	Estados Unidos	Rocky Nook

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