



INGENIERÍA EN DESARROLLO Y GESTIÓN DE SOFTWARE EN COMPETENCIAS PROFESIONALES



ASIGNATURA DE INTEGRADORA

1. Competencias	Construir soluciones de software y sistemas inteligentes mediante la gestión de proyectos, integración de metodologías, modelos y herramientas de desarrollo bajo la normatividad aplicable para la optimización de proyectos de investigación, innovación, desarrollo tecnológico y de emprendimiento.
2. Cuatrimestre	Décimo
3. Horas Teóricas	4
4. Horas Prácticas	26
5. Horas Totales	30
6. Horas Totales por Semana Cuatrimestre	2
7. Objetivo de aprendizaje	El alumno desarrollará proyectos de emprendimiento de TI mediante la elaboración de modelos de negocio y la implementación de la metodología Lean Startup para satisfacer las necesidades de un segmento de clientes.

Unidades de Aprendizaje	Horas		
	Teóricas	Prácticas	Totales
I. Business Model Canvas	2	10	12
II. Lean Startup	2	16	18
Totales	4	26	30


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	

INTEGRADORA

UNIDADES DE APRENDIZAJE

1. Unidad de aprendizaje	I. Business Model Canvas
2. Horas Teóricas	2
3. Horas Prácticas	10
4. Horas Totales	12
5. Objetivo de la Unidad de Aprendizaje	El alumno elaborará modelos de negocio de TI para atender las necesidades de un segmento de clientes.


Temas	Saber	Saber hacer	Ser
Idea de Negocio.	<p>Describir el concepto de idea de negocio.</p> <p>Identificar el concepto de propuesta de valor.</p>	Establecer ideas de negocio.	<p>Analítico</p> <p>Organizado</p> <p>Sistemático</p> <p>Sentido de planificación</p> <p>Hábil para interpretar información</p> <p>Observador</p> <p>Ético</p>
Modelo de Negocio.	Reconocer las características y componentes de Business Model Canvas	<ul style="list-style-type: none"> - Determinar segmentos de clientes. - Elaborar propuestas de valor. - Elegir canales de distribución. - Determinar relaciones con los clientes. - Establecer flujos de ingresos. - Determinar recursos clave. - Establecer actividades clave. - Determinar aliados estratégicos. - Establecer estructuras de costos. - Elaborar modelos de negocio. 	<p>Analítico</p> <p>Organizado</p> <p>Sistemático</p> <p>Sentido de planificación</p> <p>Hábil para interpretar información</p> <p>Observador</p> <p>Ético</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	

INTEGRADORA

PROCESO DE EVALUACIÓN

Resultado de aprendizaje	Secuencia de aprendizaje	Instrumentos y tipos de reactivos
Integrará un documento, a partir de un caso de estudio, que incluya: -Justificación de la idea de negocio. -Modelo de negocio Canvas.	1. Identificar los conceptos de idea de negocio y propuesta de valor. 2. Analizar las diferencias entre planes de negocio y modelos de negocio. 3. Identificar las características de Business Model Canvas. 4. Comprender los componentes de Business Model Canvas. 5. Relacionar ideas de negocio con modelos de negocio.	- Proyecto. - Lista de cotejo.

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


INTEGRADORA

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	

INTEGRADORA

UNIDADES DE APRENDIZAJE

1. Unidad de aprendizaje	II. Lean Startup
2. Horas Teóricas	2
3. Horas Prácticas	16
4. Horas Totales	18
5. Objetivo de la Unidad de Aprendizaje	El alumno implementará la metodología Lean Startup para el lanzamiento de productos y servicios de TI.


Temas	Saber	Saber hacer	Ser
Antecedentes de Lean Startup.	Reconocer los principios Lean. Reconocer el concepto de startups.		Analítico Organizado Sistemático Sentido de planificación Hábil para interpretar información Observador Ético
Principios de Lean Startup.	Reconocer las características de Lean Startup.	Establecer la visión de empresas. Determinar productos mínimos viables de TI. Establecer indicadores de productos y servicios del proceso Crear-Medir-Aprender. Evaluar el lanzamiento al mercado de productos mínimos viables.	Analítico Organizado Sistemático Sentido de planificación Hábil para interpretar información Observador Ético

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	

INTEGRADORA

PROCESO DE EVALUACIÓN

Resultado de aprendizaje	Secuencia de aprendizaje	Instrumentos y tipos de reactivos
<p>Integrará un documento, a partir de un caso de estudio, que incluya:</p> <ul style="list-style-type: none">-Visión de la empresa.-Definición del proyecto.-Justificación del producto mínimo viable.-Reporte de seguimiento del lanzamiento del producto mínimo viable. <p>El alumno integrará un proyecto de emprendimiento de TI, a partir de un caso de estudio.</p>	<ol style="list-style-type: none">1. Identificar los antecedentes y características de Lean Startup.2. Analizar los elementos de la visión de una empresa.3. Identificar el concepto de producto mínimo viable.4. Explicar el proceso crear-medir-aprender y el concepto de indicadores.5. Comprender el proceso de evaluación del lanzamiento de productos mínimos viables.	<ul style="list-style-type: none">- Proyecto.- Lista de cotejo.

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


INTEGRADORA

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	


INTEGRADORA

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.
Establecer metodologías y herramientas de gestión con base en el tipo y características del proyecto identificando las normas, estándares, leyes y regulaciones aplicables para el cumplimiento de los requerimientos establecidos	Entrega un documento que incluya: <ul style="list-style-type: none"> - Justificación de la metodología seleccionada. - Justificación de las herramientas de gestión. - Listado de las normas, estándares, leyes y regulaciones aplicables.
Implementar el plan de trabajo del proyecto mediante la metodología y herramientas establecidas incorporando normas y estándares aplicables para la optimización de recursos.	Entrega un plan de trabajo del proyecto que incluye: <ul style="list-style-type: none"> - Descripción de actividades. - Análisis de recursos. - Descripción de roles y responsabilidades. - Definición de etapas y tiempos. - Descripción de los riesgos identificados.
Controlar la ejecución del proyecto mediante el seguimiento y evaluación del plan de trabajo para asegurar el cumplimiento de los objetivos establecidos y de las normas, estándares, leyes y regulaciones aplicables.	Entrega un documento que incluya: <ul style="list-style-type: none"> - Bitácoras. - Reportes de seguimiento. - Plan de gestión del proyecto.
Determinar metodologías y herramientas de colaboración mediante el análisis del problema de acuerdo a las normas y estándares aplicables para una administración eficiente del proceso de desarrollo de software bajo un esquema de integración continua.	Entrega un documento que incluya: <ul style="list-style-type: none"> - Objetivos del desarrollo software. - Metas y alcances del desarrollo de software. - Requerimientos funcionales y no funcionales del desarrollo de software. - Justificación de la metodología seleccionada. - Justificación de las herramientas para el proceso de comunicación, desarrollo, integración continua y monitoreo del software. - Listado de las normas, estándares, leyes y regulaciones aplicables al proceso de desarrollo de software del proyecto a realizar.


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 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"> 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. 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.
<p>Gestionar el desarrollo de software mediante el seguimiento de la codificación y ejecución de pruebas para la integración continua del producto.</p>	<ol style="list-style-type: none"> Entrega los archivos de configuración del entorno de producción. 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.
<p>Realizar la liberación de unidades estables de software mediante un plan de entregas continuas para contribuir al cumplimiento de los objetivos del proyecto.</p>	<ol style="list-style-type: none"> Entrega los resultados de las pruebas de sistema en entorno de operación. 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. Entrega archivos que incluya: <ul style="list-style-type: none"> - Unidades estables de software a producción.
<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>Diseñar interfaces de usuario mediante el uso de herramientas y principios de usabilidad para optimizar la experiencia del usuario</p>	<ol style="list-style-type: none"> 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.

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	

	2. Entrega archivos de las interfaces.
Implementar modelos de datos mediante herramientas y sistemas gestores para garantizar la disponibilidad e integridad de la información	<ol style="list-style-type: none"> Entrega un documento que incluya: <ul style="list-style-type: none"> Justificación de las herramientas a utilizar para los modelos de datos. Modelos de datos. Descripción de metadatos. Entrega archivos, credenciales de registro y secuencia de configuración para la creación de los modelos de datos.
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	<ol style="list-style-type: none"> Entrega un documento que incluya: <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. Entrega archivos de código fuente y configuración de los componentes desarrollados. Entrega evidencia de almacenamiento de archivos de código fuente y actualizaciones en la documentación del proyecto en la plataforma de versionamiento seleccionada.
Ejecutar pruebas a componentes de software a través del uso de técnicas y herramientas de testing para asegurar su correcto funcionamiento	<ol style="list-style-type: none"> Entrega un documento que incluya: <ul style="list-style-type: none"> Justificación de las técnicas y herramientas utilizadas. Descripción de las pruebas. Informe de resultados del plan de pruebas. Entrega archivos de código fuente y configuración del componente.
Implementar soluciones de software mediante la integración de componentes de acuerdo a la arquitectura definida para su liberación en un ambiente de producción	<ol style="list-style-type: none"> Entrega manuales del software establecidos en el plan de desarrollo. Entrega la versión estable del software integrando los componentes de acuerdo al plan de entregas continuas. Entrega evidencia de almacenamiento de archivos de código fuente y actualizaciones de la documentación del proyecto en la plataforma de versionamiento establecida.


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

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	


INTEGRADORA

FUENTES BIBLIOGRÁFICAS

Autor	Año	Título del Documento	Ciudad	País	Editorial
Francisco Rivera Martínez – Gisel Hernández Chávez	2016 ISBN-13 : 978-6073227919	<i>Administración De Proyectos. Guía Para El Aprendizaje</i>	México	México	Pearson Educación
HARPER-SMITH, PATRICK	ISBN: 9786071712646	<i>ADMINISTRACIÓN DE PROYECTOS SERIE: FAST TRACK TO SUCCESS</i>	México	México	Editorial Trilla
Project Management Institute, Inc.	2017 ISBN: 978-1-62825-382-5	<i>Guía de los Fundamentos Para la Dirección de Proyectos: Guía del Pmbok</i>	Pennsylvania	EEUU	Project Management Institute, Inc.
Albert Garriga Rodríguez	2019 ISBN: 9788409155064	<i>Guía práctica en gestión de proyectos: Aprende a aplicar las técnicas de gestión de proyectos a proyectos reales</i>		EEUU	Albert Garriga Rodriguez
Pablo Lledó	2017 ISBN: 9780986409691	<i>Administración de Proyectos: El ABC para un Director de proyectos exitoso</i>		EEUU	Pablolledo.com LLC
Harvard Business Review	2017 ISBN-13 : 978-8494562945	<i>Gestión de proyectos</i>		EEUU	Reverté Management
Clarck, Tim; Osterwalder, Alexander; Pigneur, Yves	2012 ISBN-10 : 1118156315 ISBN-13 : 978-1118156315	<i>Business Model You: A One-Page Method for Reinventing Your Career</i>		EEUU	Wiley
TRIAS de Bes, Fernando	2020 ISBN: 9788418259111	<i>EL libro negro del emprendedor</i>		México	Empresa Activa, Ediciones Urano

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	

RIES, Erick	2016 ISBN: 0307887898	<i>El método Lean Startup</i>		México	DEUSTO
Alexander Osterwalder, Yves Pigneur, et ál.	2015 ISBN-10 : 8423419517 ISBN-13 : 978-8423419517	<i>Diseñando la propuesta de valor</i>		EEUU	Deusto
Miguel Macías Rodríguez	2017 ASIN : B076PDG6X8	<i>El camino para innovar: Cómo pasar de la idea al modelo de negocio creando</i>		EEU	Deusto
Scott Grossman	2017 ISBN 154465832X, 9781544658322	<i>Minimum Viable Product: Master Early Learning and Develop an MVP with Scrum</i>			CreateSpace Independent Publishing Platform

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	