



**MINUTA DE REUNIÓN DEL CONSEJO DE PROGRAMAS DOCENTES DEL
CENTRO DE INVESTIGACIÓN EN MATEMÁTICAS, A. C.,
CELEBRADA EL DÍA 10 DE AGOSTO DE 2020.**

En el Centro de Investigación en Matemáticas, A.C. siendo las 11:00 horas del día 10 de agosto de 2020, se reunieron los integrantes del Consejo de Programas Docentes del Centro de Investigación en Matemáticas, A.C., con objeto de celebrar una reunión extraordinaria y atendiendo a la convocatoria que para este propósito formulara el Coordinador de Formación Académica.

El Dr. Ignacio Barradas, solicitó iniciar la sesión, encontrándose presentes, por videoconferencia:

Dr. Héctor Manuel Becerra Ferrín	Coordinador del Posgrado en Computación
Dr. José Ignacio Barradas Briblesca	Presidente del Consejo y Coordinador de Formación Académica
Dr. Marcos Aurelio Capistrán Ocampo	Coordinador del Posgrado en Matemáticas Aplicadas
Dr. Pedro Luis del Ángel Rodríguez	Coordinador del Posgrado en Matemáticas Básicas
Dr. José Luis Ángel Pérez Carmendis	Coordinador del Posgrado en Probabilidad y Estadística
Dr. Humberto Martínez Baufista	Coordinador de la Maestría en Modelación y Optimización de Procesos (Unidad Aguascalientes)
MCE Jannet Vega Gutiérrez	Secretaria del Consejo y Jefa de Servicios Escolares
Dr. Hugo Arnoldo Mitre Hernández	Coordinador de la Maestría en Ingeniería de Software (Unidad Zacatecas)
Dr. Rodrigo Macías Páez	Coordinador de la Maestría en Cómputo Estadístico (Unidad Monterrey)
Invitados	
Dr. Carlos Segura González	Investigador Ciencias de la Computación

ORDEN DEL DÍA

1. Revisión y en su caso aprobación del Plan de Estudios y Lineamientos 2020 de la Maestría en Ingeniería de Software.

El Coordinador de la Maestría en Ingeniería de Software (MIS), presenta el documento con la propuesta de modificación al Plan de Estudios y Lineamientos, el cual incluye una explicación detallada de los cambios con respecto a la versión vigente.

Luego de comentar el tema, el Consejo de Programas Docentes llegó al siguiente acuerdo:





ACUERDO CPD/EX/2020/VII/01:

Se aprueba el nuevo Plan de Estudios y Lineamientos de la Maestría en Ingeniería de Software.

No habiendo más asuntos que tratar, siendo las 12:00 hrs se da por concluida la sesión. Para constancia y efectos correspondientes, firman la presente minuta:

Dr. José Ignacio Barradas Briblesca
Presidente

MCE Janjet Vega Gutiérrez
Secretario

Dr. Marcos Aurelio Capistrán Ocampo
Coordinador del Posgrado en Matemáticas Aplicadas

Dr. José Luis Ángel Pérez Garmendia
Coordinador del Posgrado en Probabilidad y Estadística

Dr. Héctor Manuel Becerra Ferrín
Coordinador del Posgrado en Computación

Dr. Hugo Arnoldo Nitre Hernández
Coordinador de la Maestría en Ingeniería de Software
(Unidad Zacatecas)

Dr. Pedro Luis del Ángel Rodríguez
Coordinador del Posgrado en Matemáticas Básicas

Dr. Humberto Martínez Bautista
Coordinador de la Maestría en Modelación
y Optimización de Procesos
(Unidad Aguascalientes)

Dr. Rodrigo Macías Páez
Coordinador de la Maestría en Cómputo Estadístico
(Unidad Monterrey)



2020
LEONORA VICARIO



MAESTRÍA EN INGENIERÍA DE SOFTWARE

Centro de Investigación en Matemáticas, A.C.

Plan de estudios 2020

2020 -

VIGENCIA

Para ingresar al programa es necesario contar con estudios de licenciatura en Ciencias Exactas, Ciencias Naturales o Ingeniería. Pueden ser profesionistas o profesores universitarios que cuenten con capacidades de sistematización e integración mediante el uso de fórmulas, reglas o teorías, competencias de interpretación y aplicación para realizar inferencias, derivar conclusiones y solucionar problemas, habilidades en lenguajes de programación, un inglés con un rango mínimo del nivel B1 según el Marco Común Europeo de Referencia (MCER) y una buena motivación de estudios en ingeniería de software.

MODALIDAD Escolarizada con orientación profesional

DURACIÓN DEL CICLO 4 semestres de 14 a 15 semanas efectivas de clase

CLAVE DEL PLAN DE ESTUDIOS 2020

OBJETIVOS GENERALES DEL PLAN DE ESTUDIOS

El objetivo general de la Maestría en Ingeniería de Software (MIS) es atender las necesidades de los profesionales de la Industria de Software de elevar sus capacidades y conocimientos para el desarrollo de Software, además de transmitir principios, mejores prácticas, y tecnologías avanzadas de la Ingeniería de Software.

Los objetivos específicos son:

1. Que los estudiantes adquieran la capacidad de contribuir con la innovación en la industria del software a través de investigaciones originales, la adquisición de competencias de investigación y la publicación de trabajos de índole técnico y científico.
2. Que los estudiantes generen las habilidades para identificar y aplicar, a través de conocimientos teóricos-prácticos, las metodologías más modernas disponibles para la solución de problemas en un entorno de ingeniería de software.
3. Que los estudiantes tengan habilidad en el desarrollo de nuevas metodologías y mejores prácticas para la solución científica de los problemas técnicos derivados de fenómenos inmersos en un contexto de ingeniería de software.
4. Que los estudiantes desarrollen habilidades para generar y transmitir conocimientos científicos y tecnológicos al sector público y privado a nivel nacional e internacional.





PERFIL DEL EGRESADO

Al término de sus estudios, el egresado de la Maestría será capaz de:

- Gestionar proyectos de software con las habilidades para mejorar las prácticas en el área de trabajo en la cual se desempeña.
- Entender y aplicar las mejores prácticas de la Ingeniería de Software en el desarrollo de soluciones competitivas a los problemas que se enfrente en su trayectoria profesional.
- Desempeñar diversos roles tales como:
 - Emprendedor de base tecnológica.
 - Líder de equipo de proyectos de software.
 - Jefe de Procesos y Calidad de empresas de desarrollo de software.
 - Consultor de organizaciones que desarrollan software.
 - Instructor que transfiere técnicas y métodos avanzados de Ingeniería de Software a sus alumnos.
 - Estudiante de doctorado en Universidades o Centros de Investigación Nacional o Internacional que ofrezcan programas en Ciencias de la Computación, Informática, Sistemas de Información, Administración de Sistemas Computacionales, Ingeniería de Software, etc.
- Tendrá un inglés intermedio avanzado nivel B2 según el Marco Común Europeo de Referencia (MCER).

Semestre	Lista de asignaturas o unidades de aprendizaje	Clave	Seriación	Horas con docente	Horas independientes	Créditos	Instalaciones A=aula, L=Laboratorio, T=Taller, O=otros
1	Probabilidad y estadística	20PYI01		48	48	6	A
1	Ingeniería de Software	20IS01		48	48	6	A
1	Ingeniería de Diseño de Software	20IDS01		48	48	6	A
1	Metodología de la Investigación	20MI01		48	48	6	A
2	Administración del desarrollo de software	20ADS01	20IS01, 20IDS01	48	48	6	A
2	Aseguramiento de la Calidad de Software	20ACS01	20IS01, 20IDS01	48	48	6	A
2	Proyecto con la Industria I	20PI01		32	128	10	A,O
3	Proyecto con la Industria II	20PI02	20PI01	32	128	10	A,O
3	Seminario de Tesis I	20ST01		60	356	20	A,O
4	Proyecto con la Industria III	20PI03	20PI02	32	128	10	A,O
4	Seminario de Tesis II	20ST02		60	356	20	A,O
				SUMA	SUMA	SUMA	
				504	1384	106	





Lista de asignaturas o unidades de aprendizaje OPTATIVAS	Clave	Seriación	Horas con docente	Horas independientes	Créditos	Instalaciones A=aula L=Laboratorio T=Taller O=otros
Definición de Procesos de Software	20DPS01		48	48	6	A, L, O
Integración de equipos de desarrollo de software	20IEDS01		48	48	6	A, L, O
Optimización de procesos	20ODP01		48	48	6	A, L, O
Tópicos Selectos de Calidad	20C01		48	48	6	A, L, O
Seguridad en Sistemas de Información	20SSI01		48	48	6	A, L, O
Tópicos Selectos de Aseguramiento del Software	20AS01		48	48	6	A, L, O
Programación Segura	20PS01		48	48	6	A, L, O
Inteligencia de Negocios	20IN01		48	48	6	A, L, O
Tópicos selectos de análisis de datos	20AD01		48	48	6	A, L, O
Tópicos selectos de aplicaciones en entornos interactivos	20AEI01		48	48	6	A, L, O
Tópicos selectos de Inteligencia artificial	20IA01		48	48	6	A, L, O
Tópicos selectos de Interacción Humano Computador	20IHC01		48	48	6	A, L, O
Tópicos selectos de visión computacional	20VC01		48	48	6	A, L, O
Tópicos selectos de programación en robótica	20PR01		48	48	6	A, L, O
			SUMA	SUMA	SUMA	
			672	672	84	





NÚMERO MÍNIMO DE HORAS QUE SE DEBERÁN ACREDITAR EN LAS ASIGNATURAS OPTATIVAS, BAJO LA CONDUCCIÓN DE UN DOCENTE

192

NÚMERO MÍNIMO DE CRÉDITOS QUE SE DEBERÁN ACREDITAR EN LAS ASIGNATURAS OPTATIVAS

24

PROPUESTA DE EVALUACIÓN Y ACTUALIZACIÓN PERIÓDICA DEL PLAN DE ESTUDIOS

El CIMAT designará un **Comité Académico de Posgrado** (CAP) integrado por investigadores adscritos al CIMAT Unidad Zacatecas. Este comité estará a cargo de los aspectos académicos del programa incluyendo la planeación académica, evaluación y seguimiento del programa. Sus decisiones se tomarán de manera colegiada, siguiendo estos lineamientos para la Maestría en Ingeniería de Software y la normativa interna de CIMAT para sus programas académicos.

OPCIONES DE TITULACIÓN

Para obtener el grado de Maestría, el alumno deberá cumplir los siguientes requisitos:

1. Cubrir la totalidad de 130 créditos del plan de estudios (106 obligatorios y 24 optativos).
2. Cumplir algunas de las siguientes alternativas para cubrir el requisito del idioma inglés:
 - a. Tomar los cursos propuestos por el CIMAT Unidad Zacatecas que acrediten el nivel solicitado en el inciso b.
 - b. Mostrar documento oficial que avale el Nivel B2 o mayor en el estándar internacional del Marco Común Europeo de Referencia para las lenguas, MCER (CEFR en inglés). El documento debe estar respaldado por una institución de prestigio a juicio del CPD.
3. Realizar un trabajo de Tesis bajo la supervisión de un asesor y defenderla ante un jurado. El tema será seleccionado por el estudiante y su asesor y deberá contar con la aprobación del CAP. La tesis consistirá en el desarrollo de un proyecto de investigación corto sobre un tema en relación con una de las LGAC de la maestría en Ingeniería de Software: Gestión de procesos y aseguramiento de la información, y Computación centrada en el humano y robótica. El trabajo de tesis se realizará durante el tercer y cuarto semestre, el cual será evaluado periódicamente en el Seminario de Tesis.

Dr. Víctor Manuel Rivero Mercado
Director General





PROGRAMAS DE ESTUDIOS

PROBABILIDAD Y ESTADÍSTICA

CICLO

CLAVE DE LA ASIGNATURA

SEMESTRE 1

20PYI01

DESCRIPCIÓN

Este es un curso práctico con la intención de desarrollar en el estudiante la capacidad de realizar análisis cuantitativos, representarlos e interpretarlos.

OBJETIVO(S) GENERAL(ES) DE LA ASIGNATURA

Al finalizar este curso el estudiante será capaz de comprender y aplicar las herramientas de desarrollo de software (e.g. lenguaje R) y metodologías básicas de la estadística y la probabilidad.

Los objetivos específicos son formar a estudiantes para:

- Presentar resultados descriptivos en situaciones diversas según el comportamiento de los datos
- Programar ecuaciones para análisis de conjuntos de datos,
- Realizar validaciones de hipótesis con pruebas paramétricas y no paramétricas según sea el caso.
- Realizar distintos tipos de análisis de correlaciones.
- Realizar representaciones gráficas de resultados.
- Particionar datos en grupos.

TEMAS Y SUBTEMAS

- I. Estadística descriptiva básica
- II. Programación de ecuaciones para conjuntos de datos
 - A. Programación de ecuaciones
 - B. Almacenamiento, recuperación y cambios de valores de datos
- III. Pruebas de hipótesis
 - A. Pruebas paramétricas
 - A. Pruebas no paramétricas
- IV. Exploración y representación visual de datos
 - A. Estructuras de datos para graficar
 - A. Gráficos de barras, líneas y dispersión



- B. Distribuciones de datos con histogramas, curvas de densidad, cajas, y otros
 - C. Interpretación de gráficos
- V. Análisis estadístico
- . Análisis de correlaciones
 - A. Análisis de grupos de datos
 - B. Análisis de series de tiempo

ACTIVIDADES DE APRENDIZAJE

- Asistencia
- Presentaciones
- Ejercicios
- Proyecto Final

CRITERIOS Y PROCEDIMIENTOS DE EVALUACION Y ACREDITACION

ASPECTO A EVALUAR	PORCENTAJE
Asistencia	10%
Presentaciones	25%
Ejercicios	25%
Proyecto final	40%

BIBLIOGRAFÍA

No.	Título	Autor	Editorial	Año
1	Hands-On Programming with R: Write Your Own Functions and Simulations	Garrett Golemund	O'Reilly Media	2014
2	R in Action: Data Analysis and Graphics with R	Robert Kabacoff	Manning Publications	2015
3	R Graphics Cookbook: Practical Recipes for Visualizing Data	Winston Chang	O'Reilly Media	2013





INGENIERÍA DE SOFTWARE

CICLO

SEMESTRE 1

CLAVE DE LA ASIGNATURA

20IS01

DESCRIPCIÓN

Esta asignatura proporciona información al estudiante que le permite crear correctamente requerimientos pasando por el proceso de obtención, análisis, especificación y validación. Para la creación de requerimientos también se presenta el cómo modelarlos gráficamente con el modelado de lenguaje unificado, es aquí donde comienza el largo camino de la calidad y donde puede evitarse trabajo en vano en la etapa de construcción del software. También se presenta material para elegir y usar una metodología adecuada a un proyecto dependiendo de la naturaleza del mismo, así como la aplicación de conceptos básicos de ingeniería de software, permitiendo establecer un marco de trabajo para construir software de mayor calidad.

OBJETIVO(S) GENERAL(ES) DE LA ASIGNATURA

El objetivo general es crear, modelar y controlar los requerimientos de un proyecto de software, además de proporcionar visión integral de los conceptos básicos de ingeniería de software así como las prácticas más reconocidas de desarrollo de software, analizando técnicas e identificando diferencias de modelos y metodologías de desarrollo de software actuales, para poder elegir y combinar las prácticas de gestión más eficientes según el contexto y naturaleza de un proyecto.

Los objetivos específicos son formar a estudiantes para:

- Diseñar el negocio de un producto o servicio para su posterior diseño en proyecto de software.
- Describir el proceso de Ingeniería de Requerimientos.
- Caracterizar y definir requerimientos en el entorno tradicional y ágil de software.
- Modelar en UML los requerimientos pertinentes.
- Extrapolar el diseño del proyecto de software con sus requerimientos en un documentos de especificación de requerimientos.
- Identificar y definir atributos de calidad de proyectos de software (requerimientos no funcionales)
- Conocer y aplicar conceptos de Ingeniería de Software para la gestión del desarrollo del software.



- Conocer las metodologías y marcos de trabajos más empleados en la industria de TI en el desarrollo de software y desarrollar habilidades para su correcta selección.
- Identificar las principales prácticas de las metodologías de gestión de proyectos de desarrollo de software.
- Desarrollar la habilidad para identificar y aplicar la metodología adecuada para el desarrollo de diferentes productos de software.

TEMAS Y SUBTEMAS

I. Introducción a la Ingeniería de Software

- A. Importancia del Software
- B. Características y tipos de software
- C. La crisis del software
- D. Definición e Importancia de la Ingeniería de Software

II. Lo esencial de los requerimientos de software

- . ¿Por qué son importantes los requerimientos?
- A. Requerimientos de software definidos
- B. Niveles de requerimientos
- C. Qué no es un requerimiento
- D. Ingeniería de requerimientos
- E. Cuando se crean malos requerimientos
- F. Características de un excelente requerimiento
- G. Requerimientos desde la perspectiva del cliente

III. Ingeniería de requerimientos: Qué, por qué y quién?

- . Proceso de gestión de requerimientos
- A. Roles en el proceso de ingeniería de requerimientos

IV. Licitación de requerimientos

- . Talleres de licitación
- A. Encontrando los requerimientos perdidos
- B. Matriz CRUD
- C. Cuando la licitación ha terminado

V. Modelo de negocio antes de requerimientos (CANVAS)

VI. Diseño y modelado de requerimientos

- . Requerimientos de métodos ágiles
- A. Requerimientos de negocio, de usuario y funcionales
- B. Especificación de requerimientos de software (estándar IEEE 830-1998)
- C. Modelado de requerimientos en Lenguaje Unificado de Modelado

VII. Requerimientos no funcionales

- . Atributos de calidad en el software

VIII. Modelos de desarrollo de software

- . Modelo en cascada
- A. Modelo Incremental
- B. Modelo Iterativo



- C. Modelo en espiral
 - D. Modelo de desarrollo de componentes
- IX. Metodología tradicional y Metodología Ágil
- . Introducción a las Metodologías
 - A. Metodologías tradicionales
 - B. Metodologías ágiles
 - C. Análisis de semejanzas y diferencias entre metodologías tradicionales y ágiles
 - D. ágiles
 - E. Metodologías tradicionales y ágiles más empleadas

ACTIVIDADES DE APRENDIZAJE

- Asistencia
- Trabajos
- Exposiciones
- Proyecto Final

CRITERIOS Y PROCEDIMIENTOS DE EVALUACION Y ACREDITACION

ASPECTO A EVALUAR	PORCENTAJE
Asistencia	10%
Trabajos	25%
Exposiciones	25%
Proyecto final	40%

BIBLIOGRAFÍA

No.	Título	Autor	Editorial	Año
1	Software Requirements Business Modeling with UML: Business Patterns at Work	WIEGERS, KARL E.	Microsoft Press	2003
2	Rapid Development	McConnell, S	Microsoft Press	1999
	Quality Attributes CMU/SEI-95-TR-021. ESC-TR-95-021	Mario Barbacci, Mark H. Klein, Thomas A. Longstaff. Charles B. Weinstock.	Microsoft Press	1995
3	IEEE Recommended Practice for Software Requirements Specifications. ISBN 0-7381-0332-2	IEEE	IEEE	1998
4	Agile software requirements: Lean requirements practices for teams, programs, and enterprise	Dean Leffingwell	Addison-Wesley	2011
5	Ingeniería del software: un enfoque práctico. 6ta Edición	Roger S. Pressman Traducción de Víctor	McGraw Hill.	2010





		Campos Olgúin, Javier Villegas Quezada. (7th. Edition.		
6	Software engineering (7 ^a ed.)	Sommerville, I.	Pearson	2004
7	El lenguaje Unificado de Modelado, UML 2.0, Guia de Usuario. 1 ^a . Edición	Booch G.	Pearson ADDISON- WESLEY	2006
8	Software Engineering. Principles and Practice (3a ed)	Hans Van Vliet	Wiley	2007





INGENIERÍA DE DISEÑO DE SOFTWARE

CICLO

SEMESTRE 1

CLAVE DE LA ASIGNATURA

20IDS01

DESCRIPCIÓN

La asignatura de Ingeniería de Diseño de Software, el alumno conocerá y aplicará las reglas generales de los modelos prescriptivos de desarrollo de software, los marcos de referencia de calidad y las técnicas de medición de software más utilizados. Además, de conocer y analizar las metodologías de desarrollo de software aplicadas en la industria. Conocerá el desarrollo de la arquitectura tiene que ver con la estructuración de un sistema para satisfacer los requerimientos de clientes y otros involucrados, en especial los requerimientos de atributos de calidad.

OBJETIVO(S) GENERAL(ES) DE LA ASIGNATURA

Al término de este curso el alumno tendrá conocimiento de las diversas metodologías de desarrollo de software y arquitectura de software en el contexto de desarrollo de sistemas. Siendo capaz de aplicar los conocimientos en diversos proyectos de software que se puedan presentar.

Los objetivos específicos son formar al estudiante para:

- Identificar y comprender los conceptos relevantes de los inicios y la evolución de la ingeniería de software.
- Identificar y comprender las diferencias y alcance de los marcos de referencia de Calidad de software vigentes.
- Identificar y comprender el concepto de arquitectura de software.
- Identificar y comprender el proceso de desarrollo de la arquitectura de software.
- Identificar y comprender los conceptos de objetivo de negocio, requisitos arquitectónicos y su relación con el diseño de la arquitectura de software.
- Identificar la noción de concepto de diseño y comprender algunos relevantes al diseño de la arquitectura.
- Reconocer y aplicar algunos métodos de desarrollo de la arquitectura de software a través de ejercicios prácticos.



TEMAS Y SUBTEMAS

I. Presentación y Descripción del Curso

II. Fundamentos de metodologías de desarrollo

- A. Producto de Software ventajas y problemas.
- B. Sistema de Información ventajas y problemas
- C. Evolución histórica de las metodologías de software
- D. Estructura de Procesos de Desarrollo Tradicionales
 - 1. Cascada
 - 2. Prototipos
 - 3. RAD
 - 4. Espiral
- E. Modelos prescriptivos vs. Modelos ágiles
- F. Marcos de referencia calidad de software
 - 1. ISO 9126
 - 2. CMMi
 - 3. ISO/IEC 15504 (SPICE)
 - 4. PMBOK

III. Introducción a la Administración de Proyectos de Software

- . Análisis de Riesgos
- A. Plan de Desarrollo del Producto
- B. Monitoreo y Control

IV. Introducción a la Arquitectura de Software

- . Definición de Arquitectura de Software
- A. Objetivos de Negocio
- B. Proceso de Desarrollo de Arquitectura

V. Identificación de Requisitos Arquitectónicos

- . Requisitos Arquitectónicos
- A. Métodos de Identificación de Requisitos Arquitectónicos

VI. Diseño de la Arquitectura

- . Principios de Diseño de Sistemas
- A. Conceptos de Diseño de Arquitectura: Patrones, Tácticas y Tecnologías.
- B. Métodos de Diseño de Arquitectura

VII. Documentación de la Arquitectura

- . 5.1. Vistas Arquitectónicas
- A. 5.2. Notaciones
- B. 5.3. Métodos de Documentación de la Arquitectura

VIII. Evaluación de Arquitectura

- . Tipos de Evaluación
- A. Métodos de Evaluación de Arquitectura





IX. Tópicos Avanzados

- . Diseñando para ...
- A. Métodos Ágiles
- B. Investigación y Desarrollo en Arquitectura de Software

X. Conclusiones y Cierre del Curso

ACTIVIDADES DE APRENDIZAJE

- Exposiciones por parte del profesor
- Exposiciones por parte del alumno
- Realización de tareas por parte del alumno
- Realización de lecturas por parte del alumno
- Desarrollo de prácticas
- Desarrollo de un proyecto integrador
- Exámenes

CRITERIOS Y PROCEDIMIENTOS DE EVALUACION Y ACREDITACION

ASPECTO A EVALUAR	PORCENTAJE
Habrá un proyecto de diseño con varios entregables y presentaciones durante el curso.	30%
Habrá varias tareas de programación en el curso.	30%
Habrá varias lecturas en el curso para las cuales se pide contestar cuestionarios	20%
Habrá dos exámenes parciales	20%

BIBLIOGRAFÍA

No.	Título	Autor	Editorial	Año
1	Software Architecture in Practice	Len Bass, Paul Clements, and Rick Kazman	Addison Wesley	2012
2	Architecting Software Intensive Systems: A Practitioner's Guide	Anthony J. Lattanze	Taylor and Francis/Auerbach	2008
3	Software Architecture: Foundations, Theory and Practice	Richard N. Taylor, Nenad Medvidovic, and Eric M	Addison Wesley	2007
4	Documenting Software Architectures: Views and Beyond	Paul Clements, Felix Bachmann, Len Bass, David Garlan, James Ivers, Reed Little and Robert Nord	Addison Wesley	2011
5	The Performance of Open	Tavish Armstrong	Lulu.com	2013





	Source Applications			
6	The Architecture Of Open Source Applications	Amy Brown and Greg Wilson	Lulu.com	Vol I, 2011 Vol II, 2012
7	Ingeniería de Software. Un enfoque práctico.	Pressman Roger	Mc. Graw Hill	5 ^a edición
8	Ingeniería de Software	Ian Sommerville	Pearson Education	7 ^a edición
9	Ingeniería de Software Teoría y Práctica	Shari Lawrence Pfleeger	Prentice Hall	2002





METODOLOGÍA DE LA INVESTIGACIÓN

CICLO

CLAVE DE LA ASIGNATURA

SEMESTRE 1

20MI01

DESCRIPCIÓN

Esta materia está orientada a que el alumno aprenda de manera práctica los principios, técnicas y herramientas necesarias para desarrollar trabajos de investigación, tales como artículos científicos, reportes técnicos y su tesis de maestrías.

OBJETIVO(S) GENERAL(ES) DE LA ASIGNATURA

En esta materia se plantea establecer las bases que permitan a los estudiantes de maestría abordar sus trabajos de investigación de manera satisfactoria incrementando su eficiencia y productividad. El objetivo principal es cubrir aspectos relacionados con la forma de llevar a cabo una investigación de modo que revierta en bien de la comunidad científica en términos de buenas publicaciones de resultados, realización de informes técnicos y elaboración de tesis.

Los objetivos particulares son formar a estudiantes para:

- Conocer el significado de la investigación científica.
- Aprender a orientar un trabajo de investigación.
- Aprender los tipos genéricos de publicaciones científicas y técnicas que existen.
- Conocer de manera general como se puede estructurar un reporte técnico y una tesis.
- Conocer las líneas de investigación más actuales y los referentes nacionales e internacionales en el ámbito de la Ingeniería del Software.
- Conocer algunas de las técnicas de experimentación más utilizadas en la Ingeniería del Software.

TEMAS Y SUBTEMAS

I. Introducción y Motivación

- A. Reflexiones sobre la investigación
- B. Investigación en ingeniería del software: La vida más allá de la programación
- C. Áreas y tópicos de investigación en la ingeniería del software

II. Cómo centrar un tema de investigación

- Consideraciones y recomendaciones generales para centrar un tema de investigación





- A. Guía práctica para definir un tema de investigación
- B. Compromisos de tutores, directores y alumnos una vez que se define un tema de investigación
- III. Búsqueda de información y revisión crítica de trabajos científicos
 - . Bases de datos y motores de búsqueda especializados
 - A. Guía práctica para la revisión crítica de trabajos científicos
 - B. Proceso de revisión literaria
 - C. Proceso de revisión sistemática
- IV. Herramientas tecnológicas de apoyo a la investigación
 - . Equipo de cómputo
 - A. Sistemas de almacenamiento físicos y virtuales
 - B. Procesadores de texto y sistemas de composición de texto (TeX y LaTeX)
 - C. Hojas de cálculo, programas estadísticos y lenguajes de programación orientados al análisis estadístico
 - D. Gestores de referencias
- V. Recomendaciones generales para escribir artículos científicos y reportar resultados
 - . Consejos prácticos para la escritura de artículos científicos, reportes técnicos y bitácoras de trabajo
 - A. Estructura general de un trabajo científico
 - B. Principales guías de estilo para referencias y trabajos
 - C. Tipos de publicaciones científicas
- VI. Técnicas para redactar y estructurar un informe técnico y una tesis
 - . Técnicas y estructura para redactar un informe técnico
 - A. Método general para estructurar y escribir una tesis

ACTIVIDADES DE APRENDIZAJE

- Trabajos
- Revisiones de artículos
- Anteproyecto de tesis

CRITERIOS Y PROCEDIMIENTOS DE EVALUACION Y ACREDITACION

ASPECTO A EVALUAR	PORCENTAJE
Asistencia (al menos 80% de las clases)	10%
Trabajo sobre definición de tema de investigación	20%
Revisión y análisis crítico de artículo científico	10%
Elaboración de anteproyecto de tesis	60%

BIBLIOGRAFÍA

No.	Título	Autor	Editorial	Año
1	Research Methodology: A Step-by-Step Guide	Ranjit Kumar	SAGE Publications	2014





	for Beginners, 4th Edition		Ltd	
2	Academic Writing for Graduate Students, 3rd Edition: Essential Tasks and Skills	John M. Swales, Christine Feak	University of Michigan Press	2012
3	A Manual for Writers of Research Papers, Theses, and Dissertations, Eighth Edition: Chicago Style for Students and Researchers (Chicago Guides to Writing, Editing, and Publishing)	Kate L. Turabian	University Of Chicago Press	2013
4	Métodos y Técnicas Orientadas al Desarrollo de Trabajos de Investigación	Maria-Isabel Sanchez-Segura, Fuensanta Medina-Dominguez, Arturo Mora-Soto	CreateSpace Independent Publishing Platform	2013

NOTA: El docente y el alumno pueden hacer uso de la biblioteca digital [http://www.cimat.mx/es/Catalogos Servicios en Linea](http://www.cimat.mx/es/Catalogos_Servicios_en_Linea) pueden acceder utilizando correo institucional, utilizando su cuenta y contraseña.





ADMINISTRACIÓN DEL DESARROLLO DE SOFTWARE

CICLO

CLAVE DE LA ASIGNATURA

SEMESTRE 2

20ADS01

DESCRIPCIÓN

La carencia de una buena gestión de proyectos continúa siendo la causa principal del fracaso de los proyectos, repitiéndose constantemente en las organizaciones el incumplimiento en tiempo y en coste aceptable de los proyectos, en productos liberados libres de defectos, por lo que se resalta el aumento de la necesidad de gestionar los proyectos de manera exitosa. Como resultado a esta necesidad, la gestión de proyectos ha ido adquiriendo mayor importancia en las organizaciones, llevando a la industria del software a enfocarse en la gestión de proyectos.

OBJETIVO(S) GENERAL(ES) DE LA ASIGNATURA

Desarrollar un plan de control integral de un proyecto de software, que considere las actividades de gestión del proyecto, el alcance, el tiempo de duración, los costos, los recursos, los riesgos y la comunicación.

Los objetivos particulares son formar a estudiantes para:

- Ofrecer una visión integral sobre la Gestión de proyectos
- Comprender modelos de buenas prácticas del proceso software, teniendo en cuenta factores como la Tecnología, la Organización y el propio negocio.
- Desarrollar la habilidad para abordar proyectos de evaluación y mejora de procesos teniendo en cuenta los factores que inciden en los mismos.
- Desarrollar la habilidad para gestionar el proceso software.
- Desarrollar la habilidad para liderar grupos de procesos de ingeniería de software .

TEMAS Y SUBTEMAS

I. Introducción a la Gestión de Proyectos

A. Concepto básicos de la Gestión de Proyectos



- B. Importancia de la Gestión de Proyectos
- C. Problemática de la gestión de proyectos
- D. Beneficios de la gestión de Proyectos
- E. Implicados en la gestión de Proyectos
- II. Tareas de la gestión de proyectos
 - . Selección de recursos
 - A. Estimación de costos y productividad
 - B. Calendarización del proyecto
 - C. Simulación de alternativas
 - D. Seguimiento del proyecto
 - E. Indicadores financieros
- III. Procesos relacionados con la Gestión de Proyectos
 - . Gestión de Proyectos
 - A. Seguimiento y Control de Proyectos
 - B. Gestión de Riesgos
 - C. Gestión de la Calidad
 - D. Gestión de la Configuración
- IV. Tendencias en la Gestión de Proyectos

ACTIVIDADES DE APRENDIZAJE

- Trabajos
- Revisiones de artículos
- Anteproyecto de tesis

CRITERIOS Y PROCEDIMIENTOS DE EVALUACION Y ACREDITACION

ASPECTO A EVALUAR	PORCENTAJE
Asistencia (al menos 90% de las clases)	10%
Trabajos	20%
Exposiciones	30%
Proyecto Final	40%

BIBLIOGRAFÍA

No.	Título	Autor	Editorial	Año
1	CMMI for Development: Guidelines for Process Integration and Product Improvement (3rd Edition) (SEI Series in Software Engineering)	by Mary Beth Chrissis (Author), Mike Konrad (Author), Sandra Shrum (Author)	Addison-Wesley Professional;	2011





2	A Discipline for Software Engineering.	Humphrey,W	AddisonWesley	1995
3	Humphrey,W	Humphrey,W	AddisonWesley	2000
4	Gestión del Proceso Software	Gonzalo Cuevas Agustín	Editorial Universitaria Ramón Areces	2002
5	A Guide to the Project Management Body of Knowledge	Project Management Institute	PMI	2000/2005

NOTA: La materia se apoyará con el uso de artículo científicos relacionados con los temas, por lo tanto, el docente y el alumno pueden hacer uso de la biblioteca digital [http://www.cimat.mx/es/Catalogos Servicios en Linea](http://www.cimat.mx/es/Catalogos_Servicios_en_Linea) pueden acceder utilizando correo institucional, utilizando su cuenta y contraseña.





ASEGURAMIENTO DE LA CALIDAD DE SOFTWARE

CICLO

CLAVE DE LA ASIGNATURA

SEMESTRE 2

20ACS01

DESCRIPCIÓN

Las organizaciones dependen cada vez más del software, debido a que éste facilita la adaptación rápida de productos y servicios a diferentes sectores del mercado. Por lo tanto, asegurar la calidad del software se ha convertido en un aspecto crítico, siendo necesario para las organizaciones de desarrollo de software saber definir adecuadamente la calidad del software y cómo debe ser evaluada dicha calidad. Además, para considerar que un software es de calidad debe ser analizada la seguridad, de lo contrario un software sin seguridad se considera un software sin calidad.

OBJETIVO(S) GENERAL(ES) DE LA ASIGNATURA

Proporcionar una visión práctica sobre la calidad del software, brindando las herramientas necesarias para implementar y mantener métricas adecuadas para asegurar la calidad y seguridad durante el desarrollo de software y así lograr software de alta calidad.

Los objetivos particulares son formar a estudiantes para:

- Conocer la importancia de la Calidad y Seguridad
- Conocer los mayores errores que se cometen en el desarrollo del software que afectan a la calidad y la seguridad.
- Conocer nuevas actividades a los procesos que garanticen la calidad y seguridad adecuada en el software que se desarrolla.
- Aprender a establecer métricas que permitan medir la calidad y la seguridad del software.

TEMAS Y SUBTEMAS

I. Introducción

- A. Historia
- B. Errores típicos en la calidad y seguridad.

II. Que es la Calidad y Seguridad.

III. Gestión de la calidad y seguridad

- . Modelos y estándares de Calidad
- A. Modelos y estándares de seguridad



- B. Metodologías de calidad y seguridad
- IV. Qué son las métricas y medidas en la calidad y seguridad
 - . Goal Question Metrics
- V. Herramientas para la calidad y seguridad en el software
- VI. Establecimiento de plan de calidad y seguridad en el desarrollo de software

ACTIVIDADES DE APRENDIZAJE

- Trabajos
- Exposiciones
- Proyecto final

CRITERIOS Y PROCEDIMIENTOS DE EVALUACION Y ACREDITACION

ASPECTO A EVALUAR	PORCENTAJE
Asistencia	5%
Trabajos	15%
Exposiciones	20%
Proyecto Final	60%

BIBLIOGRAFÍA

No.	Título	Autor	Editorial	Año
1	CMMI for Development: Guidelines for Process Integration and Product Improvement (3rd Edition) (SEI Series in Software Engineering)	by Mary Beth Chrissis (Author), Mike Konrad (Author), Sandra Shrum (Author)	Addison-Wesley Professional;	2011
2	The Goal Question Metric Approach	Victor Basili, Gianluigi Caldiera, Dieter Rombach.		1994
3	Ingeniería de Software. Un enfoque práctico (Quinta Edición)	Roger S. Pressman	Mc Graw Hill	2002
4	A Discipline for Software Engineering.	Humphrey, W	AddisonWesley	1995
5	Building Secure Software: How to Avoid Security Problems the Right Way (paperback)	Viega John & McGraw Gary	Addison-Wesley Professional Computing Series	2011

NOTA: La materia se apoyará con el uso de artículo científicos relacionados con los temas, por lo tanto, el docente y el alumno pueden hacer uso de la biblioteca digital

http://www.cimat.mx/es/Catalogos_Servicios_en_Linea pueden acceder utilizando correo institucional, utilizando su cuenta y contraseña.





PROYECTO CON LA INDUSTRIA I, II Y III

CICLO	CLAVE DE LA ASIGNATURA
SEMESTRE II	20PI01
SEMESTRE III	20PI02
SEMESTRE IV	20PI03

DESCRIPCIÓN

La serie de materias de Proyecto con la Industria tiene por objetivo la aplicación de conocimientos, habilidades, destrezas y técnicas avanzadas de Ingeniería de Software para resolver algún problema en específico o una necesidad de la industria del software o de la investigación que esté realizando el estudiante. Durante el curso, el alumno aprenderá a definir un proyecto aplicado a la industria, planificar una serie de actividades interrelacionadas, plantear objetivo de acuerdo al tipo de proyecto y los recursos disponibles y llevar a cabo el proyecto durante un periodo definido. Se espera que al finalizar el curso, los alumnos hayan creado un producto, servicio, proceso o modelo que agregue valor tecnológico o científico, además se espera que hayan incorporado las técnicas aprendidas durante su estancia en el programa.

OBJETIVO(S) GENERAL(ES) DE LA ASIGNATURA

El objetivo general es que el alumno desarrolle software o proyectos relacionados en el ámbito empresarial, emprendedor o académico y que incorpore las técnicas aprendidas durante su estancia en el programa.

Los objetivos particulares son formar a estudiantes para:

- Desarrollar experiencia en participar en proyectos de software.
- Implementar técnicas avanzadas de ingeniería de software para desarrollar producto, servicio, proceso o modelo de manera iterativa.
- Generar proyectos entregando materiales de calidad y replicables para alumnos de generaciones posteriores.
- Definir la técnica de gestión de proyectos a utilizar durante el desarrollo del proyecto y se dé un seguimiento puntual al mismo.
- Equilibrar la obtención de resultados y la administración del proyecto.
- Comprender y aplicar las mejores prácticas de gestión de proyectos.
- Avanzar en el proyecto con la industria de tal forma que pueda dar pie al desarrollo de un tema de tesis.

TEMAS Y SUBTEMAS

- Planeación del proyecto
- Diseño de propuesta



- Desarrollo del proyecto
- Reporte del proyecto

ACTIVIDADES DE APRENDIZAJE

Trabajos periódicos

Reporte de proyecto final

Presentación de proyecto final

CRITERIOS Y PROCEDIMIENTOS DE EVALUACION Y ACREDITACION

ASPECTO A EVALUAR	PORCENTAJE
Trabajos periódicos	40%
Reporte de proyecto final	50%
Presentación de proyecto final	10%

BIBLIOGRAFÍA

No.	Título	Autor	Editorial	Año
1	Kanban: Successful Evolutionary Change for Your Technology Business	David J Anderson	Blue Hole Press	2010
1	Scrum from the Trenches	Herki Kniberg	InfoQ	2007
2	Project Managment Body Of Knowledge. Fifth edition.	Project Management Institute	Project Management Institute	2013

NOTA: El docente y el alumno pueden hacer uso de la biblioteca digital [http://www.cimat.mx/es/Catalogos Servicios en Linea](http://www.cimat.mx/es/Catalogos_Servicios_en_Linea) pueden acceder utilizando correo institucional, utilizando su cuenta y contraseña.





SEMINARIO DE TESIS I Y II

CICLO

SEMESTRE III
SEMESTRE IV

CLAVE DE LA ASIGNATURA

20ST01
20ST02

DESCRIPCIÓN

Esta serie de tres materias está orientada a que el alumno adquiera las bases de investigación científica que le permitan el desarrollo de su trabajo de tesis para la obtención del grado de Maestro en Ingeniería de Software. Durante el transcurso de estos cursos el alumno establecerá un estado del arte sobre un tema en particular de acuerdo a las líneas de trabajo de los investigadores del área en la que desarrolle su trabajo de tesis. Los alumnos deberán tener reuniones de seguimiento con el investigador líder resumiendo sus actividades y el avance logrado en su investigación.

OBJETIVO(S) GENERAL(ES) DE LA ASIGNATURA

Proporcionar una ayuda, razonada y concreta, que permita una ordenada elaboración y presentación de un trabajo de tesis con suficiente rigor técnico y/o científico para que pueda ser presentado como proyecto de tesis en forma escrita y oral.

Los objetivos particulares son formar a estudiantes para:

- Comprender el significado de la investigación científica y tecnológica en el ámbito de la ingeniería de software.
- Desarrollar las habilidades básicas que le permitan conocer la fundamentación de una investigación.
- vincular los elementos básicos del conocimiento científico con el diseño de una investigación.
- Utilizar las herramientas proporcionadas en este curso y en los anteriores para elaborar su proyecto de tesis.
- Recibir conocimiento de su asesor en el aspecto formal de metodología de la investigación en el caso particular de la tesis que el alumno desarrollará.

TEMAS Y SUBTEMAS

No aplica.



ACTIVIDADES DE APRENDIZAJE

De manera particular estos cursos no tienen un temario específico con la finalidad de dar libertad a cada asesor de tesis de plantear los objetivos de aprendizaje de manera concreta al proyecto de tesis que defina con el alumno; sin embargo, como resultado de estos cursos se espera que el alumno desarrolle los siguiente productos de trabajo:

- Plan de trabajo para el desarrollo de la tesis.
- Protocolo de tesis.
- Presentación de propuesta de tesis en los seminarios de la Maestría en Ingeniería de Software.
- Documento de tesis.

CRITERIOS Y PROCEDIMIENTOS DE EVALUACION Y ACREDITACION

ASPECTO A EVALUAR	PORCENTAJE
Elaboración de protocolo de tesis	20%
Participación en seminario de tesis	20%
Participación en revisiones periódicas de la Maestría	20%
Elaboración de entregables parciales para el avance de su tesis	40%





DEFINICIÓN DE PROCESOS DE SOFTWARE

CICLO

CLAVE DE LA ASIGNATURA

20DPS01

DESCRIPCIÓN

La capacidad de las organizaciones y sus productos, sistemas y servicios que les permite competir, adaptarse y sobrevivir depende cada vez más del software, ya que facilita la adaptación rápida de productos y servicios a diferentes sectores del mercado, por lo tanto, es indispensable garantizar la calidad del mismo. Basado en la perspectiva de que la calidad del software está directamente relacionada con la calidad de los procesos utilizados para su desarrollo, las organizaciones necesitan concretar “el CÓMO” definen y despliegan sus procesos. Por lo tanto, es necesario conocer técnicas y herramientas para modelar los procesos, además del conocimiento de cómo seleccionar las más adecuadas al entorno de la organización.

OBJETIVO(S) GENERAL(ES) DE LA ASIGNATURA

Proporcionar una visión integral de los procesos y su importancia para la madurez y capacidad de las organizaciones, además de brindar las bases para modelados de procesos a través de la selección y uso de técnicas y herramientas existentes para tal fin.

Los objetivos particulares son formar a estudiantes para:

- Ofrecer una visión integral de la importancia de los procesos en las organizaciones de desarrollo de software.
- Comprender la importancia de la correcta definición de los procesos software.
- Identificar mejores prácticas en la definición de procesos software
- Desarrollar las habilidades para conformar y liderar grupo de procesos de ingeniería de software (SEPG) .

TEMAS Y SUBTEMAS

- I. Establecimiento de la cultura de procesos en las organizaciones.
- II. Conceptos Generales de la mejora de procesos.



- A. introducción a la mejora de procesos.
 - B. Modelos de mejora de procesos software.
 - C. Uso de entornos multimodelo.
 - D. Evaluación y mejora de procesos de software.
 - E. Herramientas para implementar mejora de procesos.
- III. La relación entre la calidad de procesos y la calidad de productos y servicios de software.
- IV. Modelos y estándares de calidad.
- V. Uso de Goal Question Metric en la definición de procesos.
- VI. Entendiendo el entorno empresarial para la definición de procesos.
- VII. Herramientas para el modelado de procesos.
- VIII. Extracción del Conocimiento Tácito Organizacional.
- IX. Trazabilidad entre objetivos de negocio y procesos organizacionales.
- X. Aplicación de estudio de caso.
- . Establecimiento de equipos.
 - A. Identificación de necesidades.
 - B. Propuesta de Objetivos de negocio y métricas.
 - C. Extracción de conocimiento tácito.
 - D. Propuesta de modelado de procesos.
 - E. Trazabilidad entre objetivos de negocio y procesos.
 - F. Propuesta de mejora para la organización.

ACTIVIDADES DE APRENDIZAJE

- Participación
- Trabajos
- Exposiciones
- Proyecto final

CRITERIOS Y PROCEDIMIENTOS DE EVALUACION Y ACREDITACION

ASPECTO A EVALUAR	PORCENTAJE
Asistencia (al menos 90% de las clases)	10%
Trabajos	20%
Exposiciones	20%
Proyecto Final	40%

BIBLIOGRAFÍA

No.	Título	Autor	Editorial	Año
1	CMMI for Development: Guidelines for Process	by Mary Beth Chrissis (Author),	Addison-Wesley Professional	2011





	Integration and Product Improvement (3rd Edition) (SEI Series in Software Engineering)	Mike Konrad (Author), Sandra Shrum (Author)		
2	CMMI para desarrollo: Guía para la integración de procesos y la mejora del producto (Tercera edición)	by Mary Beth Chrissis (Author), Mike Konrad (Author), Sandra Shrum (Author) Traducción Cátedra MPSEI	Editorial Universitaria Ramón Areces	2011
3	CMM Implementation Guide: Choreographing Software Process Improvement	Kim Caputo	Addison-Wesley Professional	1998
4	Gestión del Proceso Software	Gonzalo Cuevas Agustín	Editorial Universitaria Ramón Areces	2002
5	Metodología Multimodelo para Implementar Mejoras de Procesos Software	Mirna Muñoz; Gonzalo Cuevas; Tomás San Feliu	Editorial Académica Española	2012
6	Goal Question Metric	Victor Basili	A methodology for collecting valid software engineering data. Basili, V.R., D.M. Weiss	1984
7	SCAMPI Standard CMMI Appraisal Method for Process Improvement	SEI/CMU		2004
8	Process Assessment and ISO/IEC 15504: A Reference Book	Han van Loon	Springer	2nd ed. 2007 edition (November 16, 2014)

NOTA: La materia se apoyará con el uso de artículo científicos relacionados con los temas, por lo tanto, el docente y el alumno pueden hacer uso de la biblioteca digital http://www.cimat.mx/es/Catalogos_Servicios_en_Linea pueden acceder utilizando correo





institucional, utilizando su cuenta y contraseña.





INTEGRACIÓN DE EQUIPOS DE DESARROLLO DE SOFTWARE

CICLO

CLAVE DE LA ASIGNATURA

20IEDS01

DESCRIPCIÓN

En la actualidad el software es desarrollado por equipos de personas, por lo tanto, los profesionales en TI deben de estructurarse como equipos, lo que significa que deben comprender su propio rendimiento y aprender de su experiencia. Por lo tanto, un aspecto clave para lograr ser un equipo de trabajo real es que las personas sean capaces de establecer un buen análisis de los equipos de trabajo y una comunicación adecuada, así como, tener la habilidad para planificar y estimar su trabajo, que se verá reflejado en el cumplimiento de sus compromisos y una mejora en su productividad y calidad.

OBJETIVO(S) GENERAL(ES) DE LA ASIGNATURA

Proporcionar una visión integral de la importancia de la conformación de equipos de desarrollo de software mediante la impartición de conocimientos y habilidades necesarias para integrar, liderar y reforzar la colaboración de manera efectiva de equipos de desarrollo de software, de tal manera que el individuo identifique sistema del que forman parte, se conozca y conozca cómo funcionan los equipos dentro del macrosistema. Además, de proporcionar conocimiento de cómo gestionar cuantitativamente proyectos de software usando medidas de calidad, costo y cronograma.

Los objetivos particulares son formar a estudiantes para:

- Lograr pensar en sistemas y aplicar en una situación específica el pensamiento sistémico que aporte a la solución del problema organizacional y/o personal.
- Ofrecer una visión integral de la importancia del factor humano en el desarrollo de software.
- Comprender la importancia de la formación correcta de los equipos.
- Adquirir conocimiento para el establecimiento de una comunicación adecuada entre el equipo.



- Desarrollar las habilidades para conformar y liderar equipos de desarrollo de software de manera efectiva.
- Identificar mejores prácticas para la gestión de un equipo de desarrollo de software.

TEMAS Y SUBTEMAS

- I.El pensamiento sistémico ¿qué es y para qué sirve?
 - A. Elementos clave del pensamiento sistémico.
 - B. Teoría general de los sistemas.
 - C. Tipos de Sistemas (abiertos y cerrados)
 - D. Niveles de Complejidad (subsistemas, sistemas, suprasistemas)
 - E. Puntos de apalancamiento.
 - F. Aplicando sistemas en mi contexto: Genograma / Organigrama
- II.El pensamiento sistémico en la vida cotidiana y las organizaciones
 - . Axiomas de la comunicación humana
 - A. Del pensamiento lineal al pensamiento circular
 - B. Causas lineales vs. causas circulares ejemplos
 - C. Pautas y secuencias
 - D. Cambios de primer y segundo orden
- III.Pensar en equipos es pensar en sistemas
 - . Hipótesis sistémicas
 - A. Soluciones intentadas fracasadas, soluciones acertadas
 - B. Revisión de un caso, trabajar con equipos en las organizaciones
- IV.Introducción a la Formación de Equipos
 - . Perfiles de personalidad
 - A. Roles necesarios para el cambio
 - B. Equipos vs grupos
 - C. Razones de fracaso de un equipo
 - D. Características comunes a los equipos eficaces
- V.Modelos de crecimiento del equipo
 - . Actitudes y comportamiento
 - A. Etapas del modelo de crecimiento
- VI.Diagnóstico de equipos de trabajo y proyección de los mismos
- VII.Conclusiones de la primer parte del curso
- VIII.Facilitando reuniones de equipos
 - . Dirigir reuniones productivas
 - A. Proceso de reuniones
 - B. Técnicas de decisión de equipos
 - C. Habilidades para interactuar en equipo
- IX.Proceso de desarrollo de software en equipo (TSP)



2020
LEONA VICARIO



- . Introducción al TSP
- A. Estructura de TSP
- B. Roles de TSP
- C. Fases de TSP
- X. Equipos de desarrollo en metodologías ágiles
 - . Introducción a equipos auto dirigidos
 - A. Roles en metodologías ágiles
 - B. Comparativa entre roles TSP y roles de metodologías ágiles
- XI. Conclusiones, Evaluación y Cierre.

ACTIVIDADES DE APRENDIZAJE

- Participación
- Ejercicios
- Trabajos y reportes
- Proyecto
- Examen

CRITERIOS Y PROCEDIMIENTOS DE EVALUACION Y ACREDITACION

ASPECTO A EVALUAR	PORCENTAJE
Asistencia (al menos 90% de las clases)	10%
Trabajos / Reporte de lecturas	30%
Exposiciones	20%
Proyecto Final	40%

BIBLIOGRAFÍA

No.	Título	Autor	Editorial	Año
1	Pensamiento sistémico: Nuevas perspectivas y contextos de intervención	Raúl Medina, Esteban Laso y Eduardo Hernández	Litteris psicología	2014
2	La quinta disciplina	Peter Senge	Garnica	2004
3	Teoría general de los sistemas	Ludwig Von Bertalanffy	Fondo de Cultura Económica	1989
4	Introduction to the Team Software Process	Watts S. Humphrey	Addison-Wesley Professional	1999
5	TSP: Coaching Development Teams (The SEI Series in Software Engineering)	Watts S. Humphrey	Addison-Wesley Professional	2006
6	The People CMM: A Framework for	Bill Curtis and	Addison-Wesley	2009





	Human Capital Management (2nd Edition)	William E. Hefley	Professional	
7	The Agile Team Handbook	Jan Beaver	CreateSpace Independent Publishing Platform	2013
8	Coaching Agile Teams: A Companion for ScrumMasters, Agile Coaches, and Project Managers in Transition	Lyssa Adkins	(Addison-Wesley Signature Series (Cohn))	2010





OPTIMIZACIÓN DE PROCESOS

CICLO

CLAVE DE LA ASIGNATURA

200DP01

DESCRIPCIÓN

Los mercados y el mundo empresarial actual presentan como característica más notoria la necesidad a la que se enfrentan las organizaciones para optimizar y hacer más efectivos los distintos procesos de producción con el objetivo de conseguir el máximo rendimiento y la mayor productividad generando la mínima cantidad de costos enfocando en tiempo, dinero y esfuerzo.

OBJETIVO(S) GENERAL(ES) DE LA ASIGNATURA

Brindar el conocimiento para identificar problemas de optimización y proporcionar soluciones basadas en el desarrollo de modelos matemáticos adecuados, selección e implementación de las técnicas más adecuadas para resolverlos, e interpretación de los resultados obtenidos.

Los objetivos particulares son formar a estudiantes para:

- Adquirir conocimiento en los procesos que deben implementarse para poder optimizar procesos en las organizaciones.
- Conocer técnicas y herramientas que puedan ser utilizadas en la optimización de procesos.
- Conocer diferentes técnicas para representación de problemas de optimización
- Desarrollar la habilidad para construir o adaptar modelos matemáticos para representar problemas de optimización.
- Diseñar, analizar e implementar algoritmos para problemas específicos de optimización.

TEMAS Y SUBTEMAS

- I. Introducción a la Optimización de procesos.
- II. CMMI y las áreas de proceso enfocadas en la mejora y optimización de procesos.
- III. Técnicas y herramientas utilizadas en la optimización de procesos.
 - a. Definición de Métricas.



- b. Modelado de estándares y/o modelos para su implementación en empresas
- IV. Métodos numéricos.
- V. Investigación de operaciones.
- VI. Métodos estadísticos.
- VII. Simulación.

ACTIVIDADES DE APRENDIZAJE

- Participación
- Trabajos
- Exposiciones
- Proyecto final

CRITERIOS Y PROCEDIMIENTOS DE EVALUACION Y ACREDITACION

ASPECTO A EVALUAR	PORCENTAJE
Asistencia (al menos 90% de las clases)	10%
Trabajos	15%
Exposiciones	15%
Proyecto Final	40%

BIBLIOGRAFÍA

No	Título	Autor	Editorial	Año
1	Process Dynamics and control (2nd Edition)	Seborg D.E., Edgar T.F., Mellichamp D.A.	John Wiley and Sons Inc.	2004
2	Engineering Optimization	G.V. Reklaitis, A. Ravindran, K.M. Ragsdell	J. Wiley	1983
3	Practical Methods of Optimization (2nd edición)	R. Fletcher.	J. Wiley	1991
4	Model Building in Mathematical Programming (4th edition)	H.P. Williams	J. Willey	2002





5	Optimization, Foundations and Applications	R. E. Miller	J. Wiley	2000
6	CMMI for Development: Guidelines for Process Integration and Product Improvement (3rd Edition) (SEI Series in Software Engineering)	by Mary Beth Chrissis (Author), Mike Konrad (Author), Sandra Shrum (Author)	Addison-Wesley Professional;	2011
7	Goal Question Metric	Victor Basili	A methodology for collecting valid software engineering data. Basili, V.R., D.M. Weiss.	1984
8	Process Dynamics and control (2nd Edition)	Seborg D.E., Edgar T.F., Mellichamp D.A.	John Wiley and Sons Inc.	2004

NOTA: La materia se apoyará con el uso de artículo científicos relacionados con los temas, por lo tanto, el docente y el alumno pueden hacer uso de la biblioteca digital http://www.cimat.mx/es/Catalogos_Servicios_en_Linea pueden acceder utilizando correo institucional, utilizando su cuenta y contraseña.





TÓPICOS SELECTOS DE CALIDAD

CICLO	CLAVE DE LA ASIGNATURA
	20C01

DESCRIPCIÓN

En años recientes, la mejora de procesos software ha emergido como un enfoque dominante para el incremento continuo de la efectividad y eficiencia de los procesos de desarrollo y por consiguiente la calidad de los productos software. Para implementar la mejora de procesos de manera adecuada un aspecto clave es el entendimiento de elementos que refuerzan la correcta gestión de proceso y del cambio que conllevan las actividades de mejora, las cuales apoyan en el incremento de la calidad de producto y la reducción de costos y tiempo de desarrollo.

OBJETIVO(S) GENERAL(ES) DE LA ASIGNATURA

Brindar el conocimiento de temas que apoyen en la correcta implementación de iniciativas de mejora así como en elementos clave para el seguimiento y control de este tipo de actividades.

Los objetivos particulares son formar a estudiantes para:

- Conocer temas avanzados en el área de mejora de procesos.
- Desarrollar la habilidad para gestionar de manera adecuada los proyectos de implementación de mejora.
- Introducir temas para el control y seguimiento adecuado en la implementación de mejoras.
- Reforzar conocimiento para liderar grupos de procesos de ingeniería de software.

TEMAS Y SUBTEMAS

VIII. Gestión del proceso software.

IX. Gestión de proyectos.

X. Métricas en el software.

XI. Gestión de riesgos.

XII. Aseguramiento de la calidad del producto y el proceso.



XIII. Gestión del cambio.

XIV. Tendencias en Calidad del software.

ACTIVIDADES DE APRENDIZAJE

Participación
Trabajos
Exposiciones
Proyecto final

CRITERIOS Y PROCEDIMIENTOS DE EVALUACION Y ACREDITACION

ASPECTO A EVALUAR	PORCENTAJE
Asistencia (al menos 90% de las clases)	10%
Trabajos	15%
Exposiciones	15%
Proyecto Final	40%

BIBLIOGRAFÍA

No.	Título	Autor	Editorial	Año
1	Agile Estimation Techniques and innovative approaches to software process improvement	Ricardo Colomo-Palacios, Jose Antonio Calvo-Manzano Villalón, Antonio Amescua Seco, Tomás San Feliu Gilabert	IGI-Global	2014
2	Gestión del Proceso Software	Gonzalo Cuevas Agustín	Editorial Universitaria Ramón Areces	2002
3	CMMI for Development: Guidelines for Process Integration and Product Improvement (3rd Edition) (SEI Series in Software Engineering)	by Mary Beth Chrissis (Author), Mike Konrad (Author), Sandra Shrum (Author)	Addison-Wesley Professional;	2011
4	CMMI para desarrollo: Guía para la integración de procesos y la mejora del producto (Tercera	by Mary Beth Chrissis (Author), Mike Konrad (Author), Sandra Shrum	Editorial Universitaria Ramón Areces	2011



	edición)	(Author) Traducción Cátedra MPSEI		
5	Making Process Improvement Work: A Concise Action Guide for Software Managers and Practitioners	Neil Potter and Mary Sakr	Addison-Wesley Professional	2002
6	Project Management Success with CMMI: Seven CMMI Process Areas	James Persse	Prentice Hal	2007

NOTA: La materia se apoyará con el uso de artículo científicos relacionados con los temas, por lo tanto, el docente y el alumno pueden hacer uso de la biblioteca digital http://www.cimat.mx/es/Catalogos_Servicios_en_Linea pueden acceder utilizando correo institucional, utilizando su cuenta y contraseña.





SEGURIDAD EN SISTEMAS DE INFORMACIÓN

CICLO

CLAVE DE LA ASIGNATURA

20SSIO1

DESCRIPCIÓN

El auge de las redes computacionales, el surgimiento de múltiples plataformas tecnológicas, el uso de diferentes sistemas operativos y la interconexión entre todos estos elementos, si bien han favorecido el desarrollo operativo, comercial y han mejorado la productividad de las empresas, representan también el surgimiento de nuevas amenazas, muchas de las cuales cada vez son más sofisticadas dentro de Internet. Por lo tanto se hace necesaria la implementación de metodologías, modelos y estándares enfocados en el establecimiento de controles de gobernanza y gestión para la seguridad de tecnologías de Información y Comunicación encaminados hacia el logro de una gestión de la seguridad eficiente y efectiva.

OBJETIVO(S) GENERAL(ES) DE LA ASIGNATURA

Brindar el conocimiento de la importancia de la seguridad de la información para las organizaciones, sus métodos de gestión y establecimiento de objetivos de control así como conocimiento relacionado con el establecimiento de políticas, procedimientos y controles de seguridad informática dentro de una organización aplicando los estándares y mejores prácticas ampliamente aceptados y utilizados por las organizaciones en la actualidad.

Los objetivos particulares son formar a estudiantes para:

- Conocer la importancia de la seguridad de la información para las organizaciones.
- Desarrollar las habilidades y capacidades necesarias para resolver problemas relacionados con aspectos de la gestión de la seguridad informática.
- Conocer buenas prácticas de modelos y estándares que permiten el establecimiento de mecanismos de control y comunicación.
- Conocer cómo establecer lineamientos para la instalación, operación y mantenimiento de políticas y controles de seguridad informática dentro de una organización.

TEMAS Y SUBTEMAS





I.Introducción a SGSI

- A. ¿Qué es un SGSI?
- B. El enfoque basado en procesos.
- C. ¿Por qué es importante un SGSI?
- D. Establecer, supervisar, mantener y mejorar el SGSI.
- E. Factores críticos de éxito de un SGSI.
- F. Beneficios de la familia de normas de SGSI de la calidad del producto y el procesos.

II.Contexto de la Organización.

- . Comprensión de la organización y de su contexto.
- A. Política de seguridad.
- B. Gestión de activos.
- C. Seguridad ligada a los recursos humanos.
- D. Seguridad física y del entorno.
- E. Gestión de comunicaciones y operaciones.
- F. Controles de acceso.

III.Tratamiento de los riesgos de seguridad de información.

- . Acciones para tratar riesgos y oportunidades
- A. Gestión de incidentes de seguridad de la información

IV.Modelos y estándares de Seguridad

- . CERT Resilience Management Model
- A. ISO 17799, 27000, 27001, 27002
- B. ITIL
- C. COBIT
- D. Otros

V.Metodologías para establecimiento de seguridad

- . Adquisición, desarrollo y mantenimiento de los sistemas de información.

ACTIVIDADES DE APRENDIZAJE

- Participación
- Trabajos
- Exposiciones
- Prácticas
- Proyecto final

CRITERIOS Y PROCEDIMIENTOS DE EVALUACION Y ACREDITACION

ASPECTO A EVALUAR	PORCENTAJE
Asistencia (al menos 90% de las clases)	10%
Trabajos	20%





Exposiciones	20%
Proyecto Final	50%

BIBLIOGRAFÍA

No.	Título	Autor	Editorial	Año
1	CERT Resilience Management Model	Richard A. Caralli; Julia H. Allen; David W. White	Adison-Wesley	2011
2	Implantar Controles de Seguridad de la Información: Implantación de Controles de Seguridad de la Información en un CSIRT/CERT (Spanish Edition)	Carlos Solís Salazar	Editorial Académica Española	2012
3	Transforming Cybersecurity: Using COBIT 5	Isaca	Isaca	2013
4	Software Vulnerability: Identification and Minimization	Alka Agrawal, Raees A Khan	Scholars' Press	2014
5	Estándares ISO 27000 27001 y 27002			

NOTA: La materia se apoyará con el uso de artículo científicos relacionados con los temas, por lo tanto, el docente y el alumno pueden hacer uso de la biblioteca digital http://www.cimat.mx/es/Catalogos_Servicios_en_Linea pueden acceder utilizando correo institucional, utilizando su cuenta y contraseña.





TÓPICOS SELECTOS DE ASEGURAMIENTO DEL SOFTWARE

CICLO

CLAVE DE LA ASIGNATURA

20AS01

DESCRIPCIÓN

Actualmente existe un crecimiento importante en el aseguramiento del software, principalmente debido al crecimiento potencial de los ataques a sistemas de software y a la severidad de las consecuencias en caso de fallos en los sistemas software. Se hace cada vez más necesario adquirir y dominar el conocimiento en temas relacionados con el aseguramiento del software.

El aseguramiento del software surge como una disciplina que proporciona requisitos de niveles de dependencia y seguridad en el desarrollo, adquisición y operación del software, abarcando actividades enfocadas en asegurar que los procesos del ciclo de vida del software y sus productos se adecuan a unos requerimientos, estándares y procedimientos de seguridad establecidos en las organizaciones.

OBJETIVO(S) GENERAL(ES) DE LA ASIGNATURA

Brindar el conocimiento para en distintas metodologías y técnicas emergentes orientadas a mejorar el control de las Tecnologías de la información, como frameworks para desarrollo de software seguro, Informática forense y técnicas y herramientas de soporte para prevención y gestión de amenazas en TI.

Los objetivos particulares son formar a estudiantes para:

- Conocer la relación e importancia entre calidad y seguridad en el desarrollo de software seguro.
- Desarrollar habilidades para el uso de métodos y prácticas para el desarrollo de software seguro.
- Desarrollar habilidades en temas relacionados con establecimiento de control en sistemas e informática forense.
- Desarrollar habilidades para el aseguramiento de TICs mediante el uso de herramientas especializadas.



TEMAS Y SUBTEMAS

- VI. Calidad y seguridad del software
 - A. Calidad y Seguridad.
 - B. El nuevo ciclo de vida de desarrollo del software orientado a la seguridad.
 - C. Los requisitos de software y la seguridad
 - D. La gestión del riesgo durante el desarrollo del software
 - E. Las pruebas de seguridad del software
 - F. Plan de calidad del software seguro
- VII. Auditoría Informática
 - . El marco de objetivos de control CobiT
 - A. Conceptos de Informática Forense
 - B. Nuevas tendencias.
- VIII. Laboratorio de seguridad Informática
 - . Frameworks para desarrollo de software seguro

ACTIVIDADES DE APRENDIZAJE

Participación
Trabajos
Exposiciones
Prácticas
Proyecto final

CRITERIOS Y PROCEDIMIENTOS DE EVALUACION Y ACREDITACION

ASPECTO A EVALUAR	PORCENTAJE
Asistencia (al menos 90% de las clases)	5%
Trabajos	10%
Exposiciones	10%
Prácticas	25%
Proyecto Final	50%

BIBLIOGRAFÍA

No.	Título	Autor	Editorial	Año
1	CERT Resilience Management Model	Richard A. Caralli; Julia H. Allen; David W. White	Adison-Wesley	2011
2	The Web Application Hacker's	Stuttard, Dafydd, Pinto,	Wiley	2011





	Handbook: Finding and Exploiting Security Flaws (2 edition)	Marcus		
3	Software Vulnerability: Identification and Minimization	Alka Agrawal, Raees A Khan	Scholars' Press	2014
4	The Hacker Playbook: Practical Guide To Penetration Testing	Kim, Peter	CreateSpace Independent Publishing Platform	2014
5	The Database Hacker's Handbook: Defending Database Servers	Litchfield, David, Anley, Chris, Heasman, John, Grindlay, Bill	Wiley	2005
6	Estándares ISO 27000 27001 y 27002			

NOTA: La materia se apoyará con el uso de artículo científicos relacionados con los temas, por lo tanto, el docente y el alumno pueden hacer uso de la biblioteca digital http://www.cimat.mx/es/Catalogos_Servicios_en_Linea pueden acceder utilizando correo institucional, utilizando su cuenta y contraseña.





PROGRAMACIÓN SEGURA

CICLO

CLAVE DE LA ASIGNATURA
20PS01

DESCRIPCIÓN

Actualmente el uso de productos y servicios de software en cualquier entorno es una realidad y su uso frecuente es cada vez más intrínseco a la actividad humana. Sin embargo, de acuerdo a un estudio de veracode acerca del estado de seguridad de estos productos de software se ha identificado que el 77% de las aplicaciones muestran al menos una vulnerabilidad. Esto se afirma en el informe integrando, la seguridad en el ADN del ciclo de vida del Software donde el 74% de los entrevistados coinciden en que las amenazas de seguridad son una preocupación creciente debido a los problemas de software y su código. Por lo tanto, se hace necesaria la implementación de una cultura de desarrollo de software seguro.

OBJETIVO(S) GENERAL(ES) DE LA ASIGNATURA

Establecer las bases de una cultura de desarrollo de software seguro, brindando el conocimiento de las mejores prácticas para evitar o mitigar los errores en los productos de software.

Los objetivos particulares son formar a estudiantes para:

- Conocer las mejores prácticas para el desarrollo de software seguro.
- Identificar las principales vulnerabilidades en productos de software.
- Desarrollar estrategias y buenas prácticas para el desarrollo de software seguro.

TEMAS Y SUBTEMAS

- I. Introducción a la seguridad de la información.
- II. Conceptos básicos acerca de la seguridad en productos de software.
 - A. Vulnerabilidad.
 - B. Vulnerabilidad 0day.
 - C. Exploit.
 - D. Payload.
 - E. Pivoting.
 - F. OWASP.





- G. Inyección.
 - H. Inyección SQL.
 - I. Pérdida de autenticación y gestión de sesiones.
 - J. XSS.
 - K. DNS Spoofing.
 - L. DNS Smurf.
 - M. Análisis de puertos.
 - N. Banner Grabbing.
- III. Principios de Diseño de software Seguro.
- . Mínimo privilegio
 - A. Código limpio - KISS
 - B. Mantenimiento y actualización de recursos externos
 - C. Cifrado de comunicaciones
 - D. Requerimientos para nuevas funcionalidades
 - E. Data at rest o Documentación de cambios
 - F. Separación de tareas
 - G. Falla segura
 - H. Economía de mecanismo
 - I. Mediación completa
 - J. Diseño abierto
 - K. Mecanismo menos común
 - L. Aceptabilidad psicológica
 - M. Eslabón más débil
- IV. Metodologías o estándares de desarrollo de software seguro.
- . SEI (Software Engineering Institute) CERT Secure Coding Standards
 - A. Mozilla WebAppSec, entre otros.
- V. Vulnerabilidades
- . Proyecto OWASP y las vulnerabilidades más importantes.
 - A. Cómo mitigar las principales vulnerabilidades.
- VI. La estructura de una arquitectura.
- VII. Seguridad en aplicaciones web.
- VIII. Herramientas de análisis.

ACTIVIDADES DE APRENDIZAJE

Participación
Trabajos
Exposiciones
Prácticas
Proyecto final

CRITERIOS Y PROCEDIMIENTOS DE EVALUACION Y ACREDITACION





ASPECTO A EVALUAR	PORCENTAJE
Asistencia (al menos 90% de las clases)	10%
Trabajos	20%
Exposiciones	20%
Proyecto Final	50%

BIBLIOGRAFÍA

No.	Título	Autor	Editorial	Año
1	CERT Resilience Management Model	Richard A. Caralli; Julia H. Allen; David W. White	Adison-Wesley	2011
2	Implantar Controles de Seguridad de la Información: Implantación de Controles de Seguridad de la Información en un CSIRT/CERT (Spanish Edition)	Carlos Solís Salazar	Editorial Académica Española	2012
3	Transforming Cybersecurity: Using COBIT 5	Isaca	Isaca	2013





4	Software Vulnerability: Identification and Minimization	Alka Agrawal, Raees A Khan	Scholars' Press	2014
---	---	-------------------------------	-----------------	------

NOTA: La materia se apoyará con el uso de artículo científicos relacionados con los temas, por lo tanto, el docente y el alumno pueden hacer uso de la biblioteca digital http://www.cimat.mx/es/Catalogos_Servicios_en_Linea pueden acceder utilizando correo institucional, utilizando su cuenta y contraseña.





INTELIGENCIA DE NEGOCIOS

CICLO

CLAVE DE LA ASIGNATURA

20IN01

DESCRIPCIÓN

Business Intelligence (BI) o Inteligencia de Negocios, es la gestión y el análisis de una gran cantidad de datos que son transformados en conocimientos y recomendaciones. Se trata de transformar datos almacenados en información concreta, esta información concreta es una herramienta clave que es utilizada para la toma de decisiones de cualquier compañía. La inteligencia de negocios incluye: la captura de datos, la gestión del almacén de datos, el análisis de datos, la recopilación de información y los informes. La inteligencia de negocios, utiliza tecnologías de extracción de datos, herramientas y otras técnicas para transformar datos provenientes de múltiples fuentes internas y externas. Estos datos, son transformados en métricas procesables que se pueden utilizar en operaciones, planificación, desarrollo de productos, administración estratégica y otras divisiones de la empresa. Existen diferentes herramientas para obtener información de los datos como: Tableau,, R y Python. Los negocios internacionales y la administración de empresas son áreas donde se pueden usar la inteligencia de negocios. Sin embargo, esta disciplina puede ser aplicada en cualquier industria.

OBJETIVO(S) GENERAL(ES) DE LA ASIGNATURA

Implementar diversas herramientas tecnológicas para la inteligencia de negocios como R, Python y Tableau centradas en la visualización y creación de tableros de control para mejorar la toma de decisiones empresariales.

Los objetivos particulares son formar a estudiantes para:

- Explorar cómo visualizar datos con R usando librerías como ggplot2 y shinny.
- Aprender la visualización de datos con Python usando librerías como Seaborn y Boken.
- Implementar tableros de control y visualización interactiva con Tableau.
- Desarrollar habilidades para la gestión de proyectos en inteligencia de negocios.



TEMAS Y SUBTEMAS

I. Visualización de Datos con R

- A. Introducción a ggplot2
- B. Datos
- C. Aeroestética
- D. Geometría
- E. Estadísticas
- F. Coordenadas
- G. Segmentación
- H. Temas
- I. Visualización de datos interactivos con Shiny
- J. Casos de Estudio

II. Visualización de Datos con Python

- . Personalizando gráficos
- A. Graficación en 2D
- B. Gráficos estadísticos con Seaborn
- C. Análisis de Series de Tiempo e imágenes
- D. Visualizando dos variables cuantitativas
- E. Visualizando una categórica y otra cuantitativa
- F. Visualización de datos geoespaciales
- G. Visualización de datos interactivos con Boken

III. Introducción a Tableau

- . Iniciando con Tableau
- A. Construyendo y personalizando visualizaciones
- B. Visualizando datos geográficos con Deeper
- C. Presentación de datos interactivos

ACTIVIDADES DE APRENDIZAJE

- Actividades en línea
- Solución de Casos Aplicados
- Proyecto final

CRITERIOS Y PROCEDIMIENTOS DE EVALUACION Y ACREDITACION

ASPECTO A EVALUAR	PORCENTAJE
Actividades en línea	20%
Solución de Casos Aplicados	20%
Examen en línea	20%





Proyecto Final	40%
----------------	-----

BIBLIOGRAFÍA

No.	Título	Autor	Editorial	Año
1	Learning Tableau	Joshua N Milligan	Parckt Publishing	2019
2	Storytelling with Data: A data visualization guide for business professional	Cole Nussbaumer Kanaflic	Wiley	2015
3	Python Business Inteligence Cookbook	Robert Dempsey	Parckt Publishing	2015
4	Introduction to R for Business Inteligence	Jay Gendron	Impactk Publishing	2016

NOTA: La materia se apoyará con el uso de artículo científicos relacionados con los temas, por lo tanto, el docente y el alumno pueden hacer uso de la biblioteca digital http://www.cimat.mx/es/Catalogos_Servicios_en_Linea pueden acceder utilizando correo institucional, utilizando su cuenta y contraseña.





TOPICOS SELECTOS DE ANÁLISIS DE DATOS

CICLO

CLAVE DE LA ASIGNATURA
20AD01

DESCRIPCIÓN

Un analista de datos utiliza técnicas de visualización y manipulación de datos para descubrir patrones ocultos en los datos que ayudan a las organizaciones mejorar la toma de decisiones y acciones en la práctica. Se centra en la tecnología Python para aplicar herramientas de ciencia de datos, importación y limpieza de datos, con el uso de pensamiento estadístico.

OBJETIVO(S) GENERAL(ES) DE LA ASIGNATURA

Implementar con Python la importación de datos, minería de datos, explotación de datos, análisis y visualización de datos centrados en la aplicación del pensamiento estadístico y la ciencia de datos.

Los objetivos particulares son formar a estudiantes para:

- Implementar herramientas básicas de ciencia de datos.
- Importación y limpieza de datos en Python.
- Usar la minería de datos con Python.
- Aplicación del pensamiento estadístico para el análisis de los datos.

TEMAS Y SUBTEMAS

I.Ciencia de Datos con Python

- A. Introducción a ciencia de datos con Python
- B. Python intermedio aplicado a ciencia de datos
- C. Herramientas en ciencia de datos

II.Minería de Datos con Python

- . Importación de Datos
- A. Limpieza de Datos
- B. Fundamentos de Pandas



- C. Manipulación de Datos
 - D. Combinación de Datos
 - E. Visualización de Datos
- III. Pensamiento Estadístico Aplicado en Python
- . Análisis de Datos Exploratorio
 - A. Análisis de Datos Exploratorio Cuantitativo
 - B. Pensamiento Probabilista: Continuo y Discreto
 - C. Estimación de Parámetros por optimización
 - D. Intervalos de Confianza por Bootstrap
 - E. Pruebas de hipótesis en Casos de Estudio

ACTIVIDADES DE APRENDIZAJE

- Actividades en línea
- Solución de Casos Aplicados
- Proyecto final

CRITERIOS Y PROCEDIMIENTOS DE EVALUACION Y ACREDITACION

ASPECTO A EVALUAR	PORCENTAJE
Actividades en línea	20%
Solución de Casos Aplicados	20%
Examen en línea	20%
Proyecto Final	40%

BIBLIOGRAFÍA

No.	Título	Autor	Editorial	Año
1	Practical Data Science with Python 3	Ervin Varga	Springer	2019
2	Learn Data Analysis with Python	A.J. Henley, Dave Wolf	Springer	2018
3	Python Data Analytics	FABIO NELLI	Springer	2018
4	Python Data Science Handbook	Jake Vanderplas	O'Reilly	2016
5	Python for Data Analysis	Wes Mackinney	O'Reilly	2017





NOTA: La materia se apoyará con el uso de artículo científicos relacionados con los temas, por lo tanto, el docente y el alumno pueden hacer uso de la biblioteca digital http://www.cimat.mx/es/Catalogos_Servicios_en_Linea pueden acceder utilizando correo institucional, utilizando su cuenta y contraseña.





TÓPICOS SELECTOS DE APLICACIONES EN ENTORNOS INTERACTIVOS

CICLO

CLAVE DE LA ASIGNATURA

20AEI01

DESCRIPCIÓN

La asignatura de Tópicos Selectos de Aplicaciones en Entornos Interactivos el alumno conocerá y aplicará las reglas generales de cómo diseñar entornos interactivos, además de reconocer las áreas de trabajo necesarias para la realización de este tipo de proyectos. Conocerá las herramientas y los conocimientos generales para desarrollar proyectos de entornos interactivos. El alumno analizará los factores que influyen en la Interacción Humano-Computadora (IHC) con la finalidad de proponer entornos interactivos que permitan una interacción óptima con el usuario. Además, realizar un análisis y diseño utilizando metodologías centradas en el usuario lo cual le llevará a desarrollar sistemas interactivos usables. Al finalizar el curso el alumno aplicará metodologías de evaluación para medir la usabilidad de un entorno interactivo.

OBJETIVO(S) GENERAL(ES) DE LA ASIGNATURA

El alumno conocerá las metodologías de análisis y diseño centradas en el usuario que le permitan el desarrollar entornos interactivos en usables. El alumno conocerá los principios de la IHC con la finalidad proponer entornos interactivos factibles.

Los objetivos específicos son formar al estudiante para:

- El alumno conocerá metodologías de diseño centradas en el usuario, además de principios, guías y estándares que le permitan desarrollar sistemas interactivos usables.
- El alumno conocerá principios de la IHC que le permitan proponer interfaces óptimas en el desarrollo de sus sistemas interactivos.
- El alumno conocerá la importancia de la usabilidad en los sistemas interactivos, así como métodos para su evaluación.
- El alumno conocerá las tecnologías para el desarrollo de entornos interactivos que pueden ser aprovechadas en diversos ámbitos como el entretenimiento, la educación y la medicina, entre otros.



- El alumno desarrollará entornos interactivos utilizando los motores de juegos, además conocerá las capacidades que ofrecen para la integración de dispositivos y herramientas como son: Vuforia, HTC Vive, Oculus Rift, HCT Vive, Cardboard, entre otros.

TEMAS Y SUBTEMAS

I. Introducción a la Interacción Humano Computadora.

- A. Disciplinas participantes.
- B. Los aspectos del humano.
- C. Los aspectos del sistema.
- D. Los aspectos de la interacción.

II. Introducción al diseño de entornos interactivos.

- A. Desarrollo de un sistema interactivo.
 - 1. Ciclo de desarrollo del software.
 - 2. Proceso del diseño de la interfaz.
- B. Diseño centrado en el usuario.
- C. Análisis y modelado de usuarios, tareas y entornos.
 - 1. Métodos de análisis de tareas.
- D. Reglas para el diseño de interfaces de usuario.
- E. Principios, guías, estándares.
- F. Prototipado.
- G. Desarrollo de interfaces de usuario.

III. Introducción a la Usabilidad en los entornos interactivos.

- A. Objetivos de la usabilidad en los entornos interactivos.
- B. Importancia de la usabilidad.
- C. Evaluación de la Usabilidad en los entornos interactivos.
- D. Métodos de evaluación de la usabilidad en los entornos interactivos.
 - 1. Métodos de inspección.
 - 2. Métodos de indagación.
 - 3. Métodos de test.
 - 4. Laboratorios de Usabilidad.

IV. Tecnologías para el desarrollo de entornos interactivos en realidad virtual.

- A. Conceptos básicos y fundamentos de Realidad aumentada (AR), Realidad Virtual (VR) y Realidad extendida (XR).
- B. Áreas de conocimiento.
- C. Percepción sensorial en el ser humano.
- D. Arquitectura de un sistema de realidad virtual.
- E. Niveles de interacción e inmersión.
- F. Interfaces de usuarios espaciales.
- G. Interfaces de usuarios diagéticas.



- H. Interacción con elementos de la interfaz de usuario.

- V. Desarrollo de aplicaciones en realidad virtual y aumentada.
- A. Los dispositivos para realidad virtual.
 - 1. Herramientas para el desarrollo.
 - 2. Ejemplos prácticos en unity.
- B. Introducción a la realidad aumentada.
 - 1. Herramientas para el desarrollo.
 - 2. Desarrollo de aplicaciones para dispositivos móviles
 - 3. Ejemplos prácticos en unity

ACTIVIDADES DE APRENDIZAJE

- Exposiciones por parte del profesor
- Exposiciones por parte del alumno
- Realización de tareas por parte del alumno
- Realización de lecturas por parte del alumno
- Análisis de casos de estudio por parte del alumno
- Desarrollo de prácticas
- Desarrollo de un proyecto integrador
- Exámenes

CRITERIOS Y PROCEDIMIENTOS DE EVALUACION Y ACREDITACION

ASPECTO A EVALUAR	PORCENTAJE
Participación y asistencia (90%)	10%
Prácticas	30%
Proyecto integrador	30%
Examen en cada módulo	30%

BIBLIOGRAFÍA

No.	Título	Autor	Editorial	Año
1	Human-Computer Interaction	Alan Dix	Pearson Prentice Hall	2003
2	Usability Engineering	Jakob Nielsen	Morgan Kaufmann Publisher	1993
3	Th VR Book: Human-Centered Design for Virtual Reality	Jason Jerald	Morgan & Claypool	2015
4	Virtual Reality	Steven M. LaValle	Cambridge University Press	2016
5	VR UX: Learn VR UX, Storytelling &	Casey Fictum	Creater Space Independent	2016



	Design		Publishing Platform	
6	Unity Game Development Cookbook: Essentials for Every Game	Paris Buttfield-Addison	O'Reilly	2019
7	Virtual Reality	Samuel Greengard	Mit Press	2019





TÓPICOS SELECTOS DE INTELIGENCIA ARTIFICIAL

CICLO

CLAVE DE LA ASIGNATURA

20IA01

DESCRIPCIÓN

El propósito de este curso es introducir los problemas básicos de Inteligencia Artificial (IA) así como los algoritmos usuales empleados para resolverlos. Para ello, el curso está compuesto de una introducción a la IA, y de temas prácticos y aplicables en programación los distintos modelos de inteligencia artificial correspondiente a cada problema según su representación en datos o imágenes.

OBJETIVO(S) GENERAL(ES) DE LA ASIGNATURA

El objetivo general es formar habilidades prácticas a los estudiantes en la selección, entrenamiento y ajuste de modelos de inteligencia artificial dependiendo del contexto de la representación de los datos o imágenes.

TEMAS Y SUBTEMAS

Algunos temas sugeridos (pero no limitados) son:

I. Introducción a la inteligencia artificial

- A. Historia de la inteligencia artificial
- B. Introducción del aprendizaje de máquina
- C. Introducción a las redes neuronales

II. Aprendizaje de máquina

- . A. Aprendizaje supervisado
- B. Aprendizaje no supervisado
- C. Sistemas de Regresión de Aprendizaje de Máquina
- D. Sistemas de Clasificación de Aprendizaje de Máquina
- E. Redes neuronales

III. Aprendizaje profundo

- . A. Aplicaciones prácticas con modelos de aprendizaje profundo



B. Evaluación de modelos
 IV. Temas selectos de inteligencia artificial

ACTIVIDADES DE APRENDIZAJE

- Asistencia
- Participación
- Ejercicios
- Proyecto

CRITERIOS Y PROCEDIMIENTOS DE EVALUACION Y ACREDITACION

ASPECTO A EVALUAR	PORCENTAJE
Asistencia	10%
Participación	10%
Ejercicios	40%
Proyecto	40%

BIBLIOGRAFÍA

No.	Título	Autor	Editorial	Año
1	Hands-on machine learning with Scikit-Learn and TensorFlow: concepts, tools, and techniques to build intelligent systems.	Géron, A.	O'Reilly Media, Inc.	2017
2	Rosebrock, A. (2017). Deep Learning for Computer Vision with Python	Rosebrock, A.	PyImageSearch	2017
3	Deep Learning with Python	FRANÇOIS CHOLLET	Manning Publications Co.	2017





TÓPICOS SELECTOS DE INTERACCIÓN HUMANO COMPUTADOR

CICLO

CLAVE DE LA ASIGNATURA

20IHC01

DESCRIPCIÓN

La Interacción Humano-Computador (IHC) es una disciplina que estudia el diseño, evaluación e implementación de sistemas computacionales interactivos para el uso humano y el estudio de los fenómenos principales que los rodean. Existen muchas características a estudiar de las personas en los sistemas interactivos como lo son: percepción, experiencia, estados cognitivos, emociones, usabilidad. En este curso se discuten los orígenes de IHC, los aspectos básicos del humano y del computador, la evaluación de experiencia de usuario, sensores fisiológicos, los fundamentos para el diseño centrado al usuario, y el cómo abstraer y perfeccionar el entendimiento del usuario.

OBJETIVO(S) GENERAL(ES) DE LA ASIGNATURA

El objetivo general es proporcionar los fundamentos de IHC para formar estudiantes que logren comprender los elementos humanos, de computador, y de interacción para lograr capturar y perfeccionar el entendimiento del usuario en el diseño de sistemas IHC.

los objetivos específicos son formar a estudiantes para:

- Describir la historia de la disciplina Interacción humano-computador (IHC).
- Describir los aspectos básicos de IHC como los factores humanos, los elementos de interacción, la experiencia del usuario y sensores fisiológicos.
- Seguir un proceso de diseño de productos centrado en el usuario.
- Extrapolar los métodos, herramientas y técnicas para abstraer el conocimiento del usuario.

TEMAS Y SUBTEMAS

Algunos temas sugeridos (pero no limitados) son:

- I.Historia de la disciplina Interacción Hombre-Computador (IHC)
- II.El factor humano
- III.Sensores fisiológicos
- IV.Elementos de interacción
- V.Experiencia del Usuario (EU) y su evaluación
- VI.Modelos de evaluación emocional



- VII. Diseño centrado en el usuario
- VIII. Aprendiendo del producto y del usuario
- IX. Métodos de obtención de información de los usuarios

ACTIVIDADES DE APRENDIZAJE

- Asistencia
- Participación
- Presentaciones
- Proyecto

CRITERIOS Y PROCEDIMIENTOS DE EVALUACION Y ACREDITACION

ASPECTO A EVALUAR	PORCENTAJE
Asistencia	10%
Participación	10%
Presentaciones	40%
Proyecto	40%

BIBLIOGRAFÍA

No.	Título	Autor	Editorial	Año
1	Human-Computer Interaction: An Empirical Research Perspective	I. Scott MacKenzie	Elsevier	2013
2	The UXBook: Process and guidelines for ensuring a quality user experience	Rex Hartson and Pardha Pyla	Elsevier	2012
3	Understanding your users: A practical guide to user requirements: Methods, Tools and Requirements	Catherine Courage and Kathy Baxter	Elsevier	2005
4	Emotions at School	Reinhard Pekrun, Krista R Muis, Anne C Frenzel, Thomas Goetz	Routledge	2017





TÓPICOS SELECTOS DE VISIÓN COMPUTACIONAL

CICLO	CLAVE DE LA ASIGNATURA
	20VC01

DESCRIPCIÓN

La percepción del ambiente juega un papel preponderante en un sinnúmero de actividades, tanto para seres vivos como para entes artificiales. En particular, la mayoría de los animales basan gran parte de esa percepción en información visual. La visión por computadora busca emular dicha capacidad de percepción en tiempo real a partir de información visual, donde el uso de cámaras nos permite obtener gran cantidad de información de manera rápida, compacta y a bajo costo. Este curso contempla de estudio desde un enfoque tanto práctico como teórico los fundamentos necesarios para entender y aplicar las técnicas existentes de visión por computadora, así como explorar las últimas tendencias.

OBJETIVO(S) GENERAL(ES) DE LA ASIGNATURA

Proporcionar los fundamentos teóricos y prácticos más pertinentes para entender y desarrollar algoritmos de visión computacional, así como explorar las últimas tendencias en el estado del arte concernientes al desarrollo de algoritmos y sus aplicaciones a problemas reales.

Objetivos específicos:

- Proporcionar los fundamentos físicos y matemáticos necesarios para entender la formación de la imagen en una cámara, y su interacción con una escena 3-D.
- Aprender a utilizar lo último en librerías especializadas en visión por computadora, tales como OpenCV.
- Entender aplicar técnicas de procesamiento de imágenes.
- Investigar y familiarizarse con las últimas tendencias de investigación en el área de visión por computadora, como son las técnicas de aprendizaje de máquina y seguimiento mediante filtros bayesianos.
- Ser capaces de aplicar las técnicas y algoritmos existentes para resolver problemas reales en la sociedad.

TEMAS Y SUBTEMAS

Algunos temas sugeridos (pero no limitados) son:



- I.Principios matemáticos (álgebra lineal, transformada de cuerpo rígido, etc.)
- II.Procesamiento de Imágenes
- III.Librerías de OpenCV
- IV.Formación de la imagen
- V.Modelos de cámara
- VI.Calibración de cámaras
- VII.Puntos característicos y correspondencia
- VIII.Reconstrucción a partir de dos o más vistas
- IX.Visión 3-D
- X.Control basado en imagen (visual servoing)
- XI.Filtros de Kalman para seguimiento visual
- XII.Aprendizaje profundo para detección
- XIII.Localización y mapeo simultáneos (SLAM)

ACTIVIDADES DE APRENDIZAJE

- Asistencia
- Participación
- Presentaciones
- Tareas
- Examen
- Proyecto

CRITERIOS Y PROCEDIMIENTOS DE EVALUACION Y ACREDITACION

ASPECTO A EVALUAR	PORCENTAJE
Asistencia	10%
Participación	10%
Presentaciones	10%
Tareas	20%
Examen	20%
Proyecto	30%

BIBLIOGRAFÍA

No.	Título	Autor	Editorial	Año
1	An Invitation to 3-D Vision	Yi Ma, Stefano Soatto, Jana Kosecka and Shankar Sastry.	Springer	2004
2	Learning OpenCV	Gary Bradski and Adrian Kaehler	O'Reilly	2008





3	Deep Learning	Ian Goodfellow, Yoshua Bengio and Aaron Courville	MIT Press	2016
4	Multiple View Geometry in Computer Vision	Richard Hartley and Andrew Zisserman	Cambridge	2004





TÓPICOS SELECTOS DE PROGRAMACIÓN EN ROBÓTICA

CICLO

CLAVE DE LA ASIGNATURA

20PR01

DESCRIPCIÓN

La automatización y uso de agentes inteligentes cobra cada día más importancia en la sociedad actual, donde la robótica y los vehículos autónomos juegan un papel preponderante. Este curso aborda, con un enfoque práctico, los aspectos más relevantes para el desarrollo de aplicaciones de software orientadas a robótica, el uso de las librerías más utilizadas para la programación de robots, así como la implementación de algoritmos en el estado del arte para la percepción, navegación autónoma, planificación de movimiento y control de robots y vehículos autónomos.

OBJETIVO(S) GENERAL(ES) DE LA ASIGNATURA

Explorar desde un enfoque práctico las últimas tendencias para el desarrollo de aplicaciones en robótica y desarrollo de software para plataformas robóticas existentes, así como adquirir los conocimientos fundamentales para entender e implementar los últimos algoritmos en el estado del arte para la percepción, navegación, planificación de movimiento y control de robots. En particular se contempla emplear librerías de software libre ampliamente utilizadas en la comunidad robótica como son el sistema operativo de robótica (ROS: Robot Operative System), y las librerías de autopilotos para vehículos autónomos de ArduPilot, entre otras.

Objetivos específicos:

- Proporcionar los fundamentos físicos y matemáticos necesarios para entender los algoritmos pertinentes de robótica.
- Aprender a utilizar lo último en librerías de software especializadas en robótica, tales como ROS, ArduPilot, etc.
- Ser capaces de programar las plataformas robóticas existentes.
- Aplicar las técnicas aprendidas para solucionar problemas reales.

TEMAS Y SUBTEMAS

Algunos temas sugeridos (pero no limitados) son:

I. Fundamentos de robótica.



- II.El sistema operativo de robótica ROS
- III.Sensores y actuadores
- IV.Programación embebida
- V.Interfaces gráficas de usuario
- VI.Programación de autopilotos
- VII.Implementación de algoritmos de percepción, planificación, navegación y control
- VIII.Desarrollo de simuladores
- IX.Últimas tendencias en robótica

ACTIVIDADES DE APRENDIZAJE

- Asistencia
- Participación
- Tareas
- Proyecto

CRITERIOS Y PROCEDIMIENTOS DE EVALUACION Y ACREDITACION

ASPECTO A EVALUAR	PORCENTAJE
Asistencia	10%
Participación	10%
Tareas	30%
Proyecto	50%

BIBLIOGRAFÍA

No.	Título	Autor	Editorial	Año
1	https://www.ros.org/	Open Source Robotics Foundation	Open Source Robotics Foundation	2019
2	http://ardupilot.org/	ArduPilot	ArduPilot	2016
3	Modeling and Control of Mini Flying Machines	Pedro Castillo, Rogelio Lozano and Alejandro Dzul	Springer	2005
4	Robot Modeling and Control	Mark W. Spong, Seth Hutchinson, and M. Vidyasagar	JOHN WILEY & SONS, INC.	2005

