



UASLP
Universidad Autónoma
de San Luis Potosí



**FACULTAD DE
INGENIERÍA**
Área de Ciencias
de la Computación

2050 – PROYECTOS COMPUTACIONALES I

Clave de la materia: 2050
Clave Facultad: 2050
Clave U.A.S.L.P.: ----
Clave CACEI: IA
Nivel del Plan de Estudios: I.S.I.: 7; I.C.: 8
No. de créditos: 8
Horas/Clase/Semana: 4
Horas totales/Semestre: 64
Horas/Práctica (y/o Laboratorio): 0
Prácticas complementarias: 0
Trabajo extra-clase Horas/Semana: 4
Carrera/Tipo de materia: I.S.I., I.C./Obligatoria
No. de créditos aprobados: Haber aprobado todas las materias obligatorias del núcleo básico
Fecha última de Revisión Curricular: 11-noviembre-2023
Materia y clave de la materia requisito: 2053 – Administración de Proyectos I

OBJETIVO DEL CURSO

Conocer y aplicar los fundamentos de las dos primeras etapas del desarrollo de software que corresponden al análisis y diseño utilizando las diferentes técnicas de

captura de requerimientos, herramientas de modelado y la consideración de los riesgos que implica el desarrollo de software.

CONTENIDO TEMÁTICO

1. INTRODUCCIÓN, OBTENCIÓN Y ESPECIFICACIÓN DE REQUERIMIENTOS

Tiempo Estimado: 14 hrs.

Objetivo: Conocer y aplicar el proceso de Ingeniería de Requerimientos (IR) dentro del proceso de desarrollo de software, con el propósito de analizar y redactar los requerimientos apegados a las necesidades de los usuarios mediante diferentes técnicas de captura de requerimientos.

- 1.1. Conceptos básicos de ingeniería de requerimientos
- 1.2. Proceso de ingeniería de requerimientos: Obtención de requerimientos, especificación, análisis y gestión
- 1.3. Tipos de requerimientos: Atributos de calidad funcionales y no funcionales
- 1.4. Necesidades, objetivos y requerimientos
- 1.5. Clientes y stakeholders
- 1.6. Entrevistas y observaciones
- 1.7. Notaciones, lenguajes textuales y gráficos (UML, notación de requerimientos)
- 1.8. Casos de uso
- 1.9. Técnicas para escribir requerimientos de alta calidad
- 1.10. Normas de documentación
- 1.11. Especificación formal

2. MODELADO DE REQUERIMIENTOS

Tiempo estimado: 18 hrs.

Objetivo: Conocer y aplicar íntegramente el proceso unificado de desarrollo de software en sus fases de diseño, así como la notación utilizada en sus artefactos para producir el modelado de requerimientos con base

en el análisis realizado.

- 2.1. Diagramas de casos de uso UML
- 2.2. Diagramas de clases UML
- 2.3. Diagramas de secuencia UML
- 2.4. Diagramas de estados UML
- 2.5. Diagramas de actividades UML
- 2.6. Inspección, validación, integridad, detección de inconsistencias de conflictos, análisis de interacción de características y resolución

3. TÓPICOS COMPLEMENTARIOS DE BASES DE DATOS

Tiempo Estimado: 21 hrs.

Objetivo: Implementar la especificación de requerimientos en la creación del diagrama entidad relación y el diccionario de datos apegado a las necesidades de los usuarios para el desarrollo del software dentro de un proyecto.

- 3.1. Repaso de funciones básicas en SQL
- 3.2. Funciones de agregación en SQL (*Count, Min, Max, Sum, Avg*) y sentencias *Group by* y *Having*
- 3.3. Formas normales: 1ra., 2da., 3er., y BOYCE CODD
- 3.4. Conexión con la base de datos desde un lenguaje de programación
- 3.5. Procedimientos almacenados (*store procedures*) y disparadores (*triggers*)
- 3.6. JSON y bases de datos NoSQL

4. GESTIÓN DE RIESGOS

Tiempo estimado: 5 hrs.

Objetivo: Identificar y considerar la redacción de los riesgos que se pueden producir en el desarrollo de software buscando mitigar retrasos en el desarrollo de un proyecto, así como la consideración de la gestión de control de versiones que llevarán a una entrega adecuada del desarrollo

- 4.1. Trazabilidad, prioridades, cambios, líneas de base y soporte de herramientas
- 4.2. Negociación de requerimientos
- 4.3. Organización, gestión y priorización de riesgos

5. PROYECTO DE ANÁLISIS Y DISEÑO DE UN SISTEMA

Tiempo estimado: 6 hrs.

Objetivo: Aplicar lo aprendido durante el curso en el análisis y diseño de un sistema computacional real.

- 5.1. Compilación del análisis y diseño del sistema en un solo documento
- 5.2. Exposición de proyectos

METODOLOGÍA

Exposición de los temas por parte del profesor utilizando material audiovisual.

Se utilizará el método de aula invertida, en donde el alumno tiene el compromiso de leer acerca del tema antes de cada clase.

Por la naturaleza de la materia, se requiere que el profesor haga uso de diversos casos de proyectos reales, para que el alumno comprenda de mejor manera los conceptos y los pueda aplicar en su proyecto.

Fomentar la participación de los alumnos mediante discusiones en clase, así como ejercicios que el estudiante deberá resolver con la guía del profesor, siguiendo el método de aprendizaje basado en problemas.

En lo posible se fomentará el trabajo en equipo.

En la medida de lo posible se recomienda usar el idioma inglés para las presentaciones *Powerpoint*, los apuntes e incluso las tareas, esto con el fin de que los alumnos practiquen el idioma desde un punto de vista técnico.

El entregable de esta materia involucra la definición de requerimientos técnicos, basados en los requerimientos del proceso que se entregaron en el Acta Constitutiva del proyecto; así como el modelado del software bajo la metodología de UML.

EVALUACIÓN

Se realizarán tres exámenes parciales de forma colegiada en las fechas establecidas por la Facultad, de acuerdo con el Reglamento de Exámenes. La calificación de los dos primeros exámenes parciales estará compuesta por la evaluación del examen parcial (40%), el avance del proyecto (30%) y por otras actividades, como: tareas, investigaciones, resolución de problemas,

ejercicios, etc. (30%). La calificación del tercer examen parcial estará compuesta por la evaluación del examen parcial (40%), la entrega final del proyecto (50%) y por otras actividades, como: tareas, investigaciones, resolución de problemas, ejercicios, etc. (10%). La calificación del examen ordinario es el promedio de los tres parciales.

BIBLIOGRAFÍA

Bibliografía Básica

Date, C., *SQL and Relational Theory: How to Write Accurate SQL Code*. O'Reilly Media, Inc., 2015, Third Edition.

Date, C., *Database Design and Relational Theory: Normal Forms and All That Jazz*, O'Reilly Media, Inc., 2012.

Silberschatz, F., Korth, H., Sudarshan, S., *Database System Concepts*, 7th Edition, McGraw Hill, 2020.

Harrison et al. *Next Generation Databases: NoSQL, NewSQL, and Big Data*. 2016. Apress.

James Rumbaugh, Ivar Jacobson y Grady Booch *Proceso Unificado de Desarrollo de Software*, Addison Wesley, 2018

Horine, G. *Project Management Absolute Beginner's Guide. 4th Edition*, Que Publishing, 2017.

Kerzner, H. *Project Management: A Systems Approach to Planning, Scheduling, and Controlling*. 12th Edition, Wiley, 2017.

Wiegers, Karl, and Joy Beatty. *Software Requirements*. Pearson Education, 2013.

Bibliografía Complementaria

Carter, M., Price, C. *Operations Research: A Practical Introduction*. CRC Press, 2017.

Chen, Anthony, and Joy Beatty. *Visual Models for Software Requirements*. Pearson Education, 2012.

McDonald, Kent J. *Beyond requirements: Analysis with an Agile Mindset*. Addison-Wesley Professional, 2015.

Schwalbe, K. *An Introduction to Project Management*. 6th Edition, CreateSpace Independent Publishing Platform, 2017.

Meredith, J., et al. *Project Management in Practice*. 6th Edition, Wiley, 2016.

Sitios de internet

Project Management Institute: <https://www.pmi.org/>

Creativa UASLP: <https://creativa.uaslp.mx/>

Tutorial SQL:

<https://www.w3schools.com/sql/default.asp>