



**SEP**

SECRETARÍA DE  
EDUCACIÓN PÚBLICA



TECNOLÓGICO  
NACIONAL DE MÉXICO®

# **INSTITUTO TECNOLÓGICO DE MINATITLÁN**

**INGENIERÍA EN SISTEMAS COMPUTACIONALES**

**“MANUAL DE PRÁCTICAS “**

**MATERIA**

**FUNDAMENTO DE INGENIERÍA DE SOFTWARE**

**MINATITLÁN, VER. AGOSTO DEL 2023**



## 3.2 ÍNDICE DEL MANUAL DE PRÁCTICAS

### ÍNDICE

3.2 ÍNDICE DEL MANUAL DE PRÁCTICAS .....	2
3.1 INTRODUCCIÓN .....	5
3.2 JUSTIFICACIÓN .....	5
3.3 OBJETIVO GENERAL DEL MANUAL DE PRÁCTICAS .....	6
3.4 DESARROLLO .....	6
3.4.1 Práctica 1 Revisar un estudio de casos, donde se identifique una metodología de desarrollo de software y el estudiante justifique su uso en ese caso particular. ....	6
3.4.1.1 Objetivo .....	6
3.4.1.2 Introducción .....	6
3.4.1.3 Correlación Los Temas Y Subtemas Del Programa De Estudio Vigente. ....	7
3.4.1.4 Material Y Equipo Necesario .....	7
3.4.1.5 Metodología .....	7
3.4.1.6 Sugerencias Didácticas .....	8
3.4.1.7 Reporte Del Alumno .....	8
3.4.1.8 Bibliografías .....	8
3.4.2 Práctica 2 Revisar un estudio de casos, e identificar las métricas que puede cumplir el software en ese caso particular. ....	9
3.4.2.1 Objetivo .....	9
3.4.2.2 Introducción .....	9
3.4.2.3 Correlación Con Los Temas Y Subtemas Del Programa De Estudio Vigente. ....	9
3.4.2.4 Material Y Equipo Necesario .....	9
3.4.2.5 Metodología .....	9
3.4.2.6 Sugerencias Didácticas .....	10
3.4.2.7 Reporte Del Alumno .....	10
3.4.2.8 Bibliografías .....	10
3.4.3 práctica 3 Realizar un estudio de factibilidad (técnica, económica y operacional) de un caso real para el desarrollo de software. ....	11
3.4.3.1 Objetivo .....	11
3.4.3.2 Introducción .....	11
3.4.3.3 Correlación Con Los Temas Y Subtemas Del Programa De Estudio Vigente. ....	11
3.4.3.4 Material Y Equipo Necesario .....	11

3.4.3.5 Metodología .....	11
3.4.3.6 Sugerencias Didácticas .....	12
3.4.3.7 Reporte Del Alumno .....	12
3.4.3.8 Bibliografías .....	12
3.4.4 Práctica 4. Elaborar el modelado de negocios utilizando la notación de un estándar para una empresa de contexto .....	13
3.4.4.1 Objetivo .....	13
3.4.4.2 Introducción .....	13
3.4.4.3 Especificar La Correlación Con El O Los Temas Y Subtemas Del Programa De Estudio Vigente. ....	13
3.4.4.4 Material Y Equipo Necesario .....	13
3.4.4.5 Metodología .....	14
3.4.4.6 Sugerencias Didácticas .....	14
3.4.4.7 Reporte Del Alumno .....	14
3.4.4.8 Bibliografías .....	14
3.4.5 Práctica 5 Elaborar diagramas de casos de uso para el modelado de negocios sobre situaciones planteadas por el docente, utilizando herramientas CASE.....	15
3.4.5.1 Objetivo .....	15
3.4.5.2 Introducción .....	15
3.4.5.3 Especificar La Correlación Con El O Los Temas Y Subtemas Del Programa De Estudio Vigente. ....	15
3.4.5.4 Material Y Equipo Necesario .....	15
3.4.5.5 Metodología .....	15
3.4.5.6 Sugerencias Didácticas .....	16
3.4.5.7 Reporte Del Alumno .....	16
3.4.5.8 Bibliografías .....	16
3.4.6 Práctica 6 Elaborar algunas llamadas al sistema y distinguir el mecanismo de las interrupciones en el contexto de los Procesos .....	17
3.4.6.1 Objetivo .....	17
3.4.6.2 Introducción .....	17
3.4.6.3 Especificar La Correlación Con El O Los Temas Y Subtemas Del Programa De Estudio Vigente. ....	17
3.4.6.4 Material Y Equipo Necesario .....	17
3.4.6.5 Metodología .....	17
3.4.6.6 Sugerencias Didácticas .....	18

3.4.6.7 Reporte Del Alumno.....	19
3.4.6.8 Bibliografías.....	19
3.4.7 Práctica 7 Revisar un estudio de casos que involucre un producto de software y determinar el nivel de madurez de acuerdo a diferentes estándares como SPICE, CMMI, MoProSoft o ISO 29110.....	20
3.4.7.1 Objetivo .....	20
3.4.7.2 Introducción .....	20
3.4.7.3 Especificar La Correlación Con El O Los Temas Y Subtemas Del Programa De Estudio Vigente. ....	20
3.4.7.4 Material Y Equipo Necesario .....	20
3.4.7.5 Metodología .....	21
3.4.7.6 Sugerencias Didácticas .....	21
3.4.7.7 Reporte Del Alumno .....	22
3.4.7.8 Bibliografías.....	22

### **3.1 INTRODUCCIÓN**

El presente manual dará a conocer las prácticas relacionadas con los temas de la materia Sistemas Operativos, los cuales están divididos en 7 prácticas con respecto al temario de la materia:

1. Revisar un estudio de casos, donde se identifique una metodología de desarrollo de software y el estudiante justifique su uso en ese caso particular.
2. Revisar un estudio de casos, e identificar las métricas que puede cumplir el software en ese caso particular.
3. Realizar un estudio de factibilidad (técnica, económica y operacional) de un caso real para el desarrollo de software.
4. Elaborar el modelado de negocios utilizando la notación de un estándar para una empresa de contexto.
5. Elaborar diagramas de casos de uso para el modelado de negocios sobre situaciones planteadas por el docente, utilizando herramientas CASE.
6. Realizar la obtención de requisitos específicos (funcionales y no funcionales) de un caso prácticos, con la aplicación de herramientas CASE, presentando los resultados a través de un informe técnico utilizando diagramas de Casos de Uso y de Actividades.
7. Revisar un estudio de casos que involucre un producto de software y determinar el nivel de madurez de acuerdo con diferentes estándares como SPICE, CMMI, MoProSoft o ISO 29110.

### **3.2 JUSTIFICACIÓN**

Un Manual de prácticas puede definirse como un compendio de documentos que contemplan una serie de aportes a la práctica científica y social de los alumnos que se encuentren realizando dicha práctica, las cuales también incluyen las normas y procedimientos que orientarán el desempeño del alumno y facilitarán la integración de la teoría con la práctica, en un contexto real de aprendizaje.

Este manual de prácticas está basado según el contenido de “el libro Guía para la elaboración y registro de textos o trabajos académicos”, con el que cuenta el Tecnológico Nacional de México.

El manual de prácticas servirá como apoyo de aprendizaje para los alumnos de la materia de Fundamentos de Ingeniería de Software, así como apoyo didáctico para los maestros de dicha materia, ya que se presentarán consejos y sugerencias para dicha realización de las prácticas, también se dará materia de apoyo para estas mismas.

### **3.3 OBJETIVO GENERAL DEL MANUAL DE PRÁCTICAS**

El objetivo principal es proporcionar a los estudiantes una comprensión sólida de los principios fundamentales, conceptos y técnicas relacionados con el proceso de desarrollo de software. Esto incluye la planificación, diseño, implementación, prueba, mantenimiento y gestión de software.

### **3.4 DESARROLLO**

**3.4.1 Práctica 1 Revisar un estudio de casos, donde se identifique una metodología de desarrollo de software y el estudiante justifique su uso en ese caso particular.**

#### **3.4.1.1 Objetivo**

El objetivo de esta práctica es que los estudiantes adquieran la capacidad de analizar un estudio de caso relacionado con el desarrollo de software, identificar la metodología utilizada y justificar su aplicabilidad en el contexto particular del caso estudiado.

#### **3.4.1.2 Introducción**

La ingeniería de software desempeña un papel fundamental en el desarrollo de aplicaciones y sistemas informáticos. Para comprender cómo se aplican las metodologías de desarrollo de software en situaciones del mundo real, es esencial analizar estudios de casos concretos. Esta práctica permite a los estudiantes examinar un caso y evaluar la elección de la metodología de desarrollo de software,

considerando aspectos como la naturaleza del proyecto, los requisitos, el equipo de desarrollo y los objetivos.

#### **3.4.1.3 Correlación Los Temas Y Subtemas Del Programa De Estudio Vigente.**

1.1. Conceptos básicos

1.2. Fases de la Ingeniería de software.

1.3 Metodologías de desarrollo de software

1.3.1 Clásicas

1.3.2 Agiles

1.3.3 Otras filosofías.

#### **3.4.1.4 Material Y Equipo Necesario**

1. Estudio de caso relacionado con el desarrollo de software.
2. Acceso a recursos académicos y bibliográficos.
3. Papel y bolígrafo o dispositivos electrónicos para tomar notas.
4. Acceso a una plataforma de presentación o elaboración de informes.

#### **3.4.1.5 Metodología**

- Selección de un estudio de caso: Los estudiantes deben elegir un estudio de caso relacionado con el desarrollo de software. Puede ser un proyecto real o ficticio, pero debe estar bien documentado.
- Identificación de la metodología: Analizar el estudio de caso para identificar la metodología de desarrollo de software utilizada. Esto implica revisar la documentación, entrevistas o cualquier fuente relevante.
- Justificación de la metodología: Los estudiantes deben justificar por qué se seleccionó esa metodología en el estudio de caso. Deben considerar aspectos como la complejidad del proyecto, los plazos, los requisitos del cliente y otros factores que influyeron en la elección.
- Análisis de resultados: En esta sección, se espera que los estudiantes analicen los resultados de su investigación y presenten sus conclusiones.

Deben destacar cómo la metodología afectó el éxito o fracaso del proyecto y qué lecciones se pueden extraer.

#### **3.4.1.6 Sugerencias Didácticas**

- Fomentar la discusión y el intercambio de ideas entre los estudiantes para enriquecer el análisis.
- Promover la investigación y la consulta de fuentes adicionales para respaldar la justificación de la metodología.
- Alentar a los estudiantes a aplicar conceptos y teorías previamente aprendidos en la materia de Ingeniería de Software.

#### **3.4.1.7 Reporte Del Alumno**

El alumno debe de realizar la actividad detallando paso a paso la elaboración de esta, incluyendo capturas, mediante el formato de un reporte de prácticas dando detalle de los resultados obtenidos, así como su conclusión y aprendizajes obtenidos.

#### **3.4.1.8 Bibliografías**

- [itd.upm.es/wp-content/uploads/2014/06/metodologia\\_estudios\\_de\\_caso.pdf](http://itd.upm.es/wp-content/uploads/2014/06/metodologia_estudios_de_caso.pdf)
- [La elección del estudio de caso en investigación educativa \(ugr.es\)](http://La%20elecci3n%20del%20estudio%20de%20caso%20en%20investigaci3n%20educativa%20(ugr.es))



### **3.4.2 Práctica 2 Revisar un estudio de casos, e identificar las métricas que puede cumplir el software en ese caso particular.**

#### **3.4.2.1 Objetivo**

El objetivo de esta práctica es que los estudiantes desarrollen la capacidad de analizar un estudio de caso relacionado con el desarrollo de software y, a partir de dicho análisis, identificar las métricas y criterios de medición aplicables al software en ese contexto específico.

#### **3.4.2.2 Introducción**

La evaluación y medición del software son aspectos críticos en la ingeniería de software. Esta práctica tiene como finalidad que los estudiantes comprendan la importancia de definir métricas específicas y aplicables a proyectos de software concretos. Para ello, se enfocarán en el estudio de casos reales o ficticios para identificar las métricas pertinentes.

#### **3.4.2.3 Correlación Con Los Temas Y Subtemas Del Programa De Estudio Vigente.**

2.1 Definición

2.2 Componentes

2.3 Estándares

2.4 Diagramas.

#### **3.4.2.4 Material Y Equipo Necesario**

1. Estudio de caso relacionado con el desarrollo de software.
2. Acceso a recursos académicos y bibliográficos sobre métricas de software.
3. Papel y bolígrafo o dispositivos electrónicos para tomar notas.
4. Acceso a una plataforma de presentación o elaboración de informes.

#### **3.4.2.5 Metodología**

- Selección del estudio de caso: Los estudiantes deben elegir un estudio de caso relacionado con el desarrollo de software. Este caso puede ser real o ficticio, pero debe estar documentado de manera adecuada.

- **Análisis del estudio de caso:** Los estudiantes deben examinar a fondo el estudio de caso para comprender el contexto, los objetivos del software, los usuarios finales y otros factores relevantes.
- **Identificación de métricas:** A partir del análisis, los estudiantes deben identificar y seleccionar las métricas que son aplicables y relevantes para evaluar el software en ese caso particular. Estas métricas pueden incluir la eficiencia, la calidad del código, la usabilidad, el rendimiento y otras métricas específicas del dominio.
- **Justificación de las métricas:** Los estudiantes deben explicar por qué eligieron estas métricas en función de la naturaleza del proyecto, los objetivos y las necesidades del cliente. Deben destacar cómo estas métricas ayudarán a evaluar el software de manera efectiva.

#### **3.4.2.6 Sugerencias Didácticas**

- Fomentar la discusión entre los estudiantes para compartir diferentes perspectivas y enfoques en la identificación de métricas.
- Promover la investigación y consulta de fuentes de referencia relacionadas con la medición de software.
- Proporcionar ejemplos de casos de estudio variados para enriquecer la experiencia de los estudiantes.

#### **3.4.2.7 Reporte Del Alumno**

El alumno debe de realizar la actividad detallando paso a paso la elaboración de esta, incluyendo capturas, mediante el formato de un reporte de prácticas dando detalle de los resultados obtenidos, así como su conclusión y aprendizajes obtenidos.

#### **3.4.2.8 Bibliografías**

- [scaridad.com/static/descargas/tema\\_2\\_mtricas\\_y\\_modelos\\_de\\_estimacion\\_del\\_software.pdf](https://scaridad.com/static/descargas/tema_2_mtricas_y_modelos_de_estimacion_del_software.pdf).
- [Métricas de prueba de software: qué es, tipos y todo lo que necesitas saber - Atentus: Servicio de Monitoreo](#)

### **3.4.3 práctica 3 Realizar un estudio de factibilidad (técnica, económica y operacional) de un caso real para el desarrollo de software.**

#### **3.4.3.1 Objetivo**

El objetivo de esta práctica es que los estudiantes adquieran la habilidad de llevar a cabo un estudio de factibilidad integral (técnica, económica y operacional) para evaluar la viabilidad de un proyecto de desarrollo de software en un caso real.

#### **3.4.3.2 Introducción**

Los proyectos de desarrollo de software requieren un análisis cuidadoso antes de su ejecución. Este proceso implica evaluar la factibilidad desde tres perspectivas clave: técnica, económica y operacional. La práctica tiene como objetivo que los estudiantes comprendan la importancia de esta etapa en el ciclo de vida del desarrollo de software.

#### **3.4.3.3 Correlación Con Los Temas Y Subtemas Del Programa De Estudio Vigente.**

2.1 Definición

2.2 Componentes

2.3 Estándares

2.4 Diagramas

#### **3.4.3.4 Material Y Equipo Necesario**

1. Caso real o ficticio para el desarrollo de software.
2. Acceso a recursos académicos y fuentes de referencia sobre estudios de factibilidad.
3. Hojas de cálculo o herramientas para análisis económico.
4. Documentación relacionada con el caso.

#### **3.4.3.5 Metodología**

1. Selección del caso de estudio: Los estudiantes deben elegir un caso de desarrollo de software, real o ficticio, sobre el cual llevar a cabo el estudio de factibilidad. Este caso debe estar documentado adecuadamente.

2. Análisis de factibilidad técnica: Evaluación de la viabilidad desde un punto de vista técnico. Esto implica determinar si la tecnología necesaria está disponible, si el conocimiento y las habilidades requeridas existen y si hay barreras técnicas significativas.
3. Análisis de factibilidad económica: Estimación de los costos y beneficios del proyecto. Los estudiantes deben calcular los costos iniciales y continuos, así como los posibles ingresos o ahorros asociados al proyecto.
4. Análisis de factibilidad operacional: Evaluación de la viabilidad desde una perspectiva operativa. Esto implica considerar si el proyecto se puede integrar en los procesos existentes y si cumple con los requisitos operativos del cliente.
5. Presentación de resultados: Los estudiantes deben resumir los resultados de los análisis técnicos, económicos y operativos en un informe que destaque la viabilidad general del proyecto.

#### **3.4.3.6 Sugerencias Didácticas**

- Fomentar la colaboración entre los estudiantes para enriquecer el análisis de factibilidad.
- Proporcionar ejemplos de casos de estudio diversos para permitir a los estudiantes abordar situaciones variadas.
- Enfatizar la importancia de la toma de decisiones informadas en la etapa de planificación de proyectos de software.

#### **3.4.3.7 Reporte Del Alumno**

El alumno debe de realizar la actividad detallando paso a paso la elaboración de esta, incluyendo capturas, mediante el formato de un reporte de prácticas dando detalle de los resultados obtenidos, así como su conclusión y aprendizajes obtenidos.

#### **3.4.3.8 Bibliografías**

- [repopib.ubiobio.cl/jspui/bitstream/123456789/2433/1/Lazo Vallejos Marcelo Andres.pdf](http://repopib.ubiobio.cl/jspui/bitstream/123456789/2433/1/Lazo_Vallejos_Marcelo_Andres.pdf)

- [repositorio.puce.edu.ec/bitstream/handle/22000/10007/Tesis\\_Maestria\\_GTI\\_MFSanchez.pdf?sequence=1&isAllowed=y](https://repositorio.puce.edu.ec/bitstream/handle/22000/10007/Tesis_Maestria_GTI_MFSanchez.pdf?sequence=1&isAllowed=y)

### **3.4.4 Práctica 4. Elaborar el modelado de negocios utilizando la notación de un estándar para una empresa de contexto**

#### **3.4.4.1 Objetivo**

El objetivo de esta práctica es que los estudiantes adquieran habilidades en la elaboración del modelado de negocios de una empresa utilizando una notación estándar. Esto implica representar gráficamente los procesos, estructuras y flujos de información de la empresa para comprender y comunicar su funcionamiento.

#### **3.4.4.2 Introducción**

El modelado de negocios es una herramienta esencial en la gestión empresarial que permite comprender y analizar los procesos y la estructura de una empresa. La práctica tiene como finalidad que los estudiantes apliquen estos conceptos utilizando una notación estándar.

#### **3.4.4.3 Especificar La Correlación Con El O Los Temas Y Subtemas Del Programa De Estudio Vigente.**

3.1 Características de los requisitos.

3.2 Tipos de requisitos.

3.2.1 Funcionales.

3.2.2 No funcionales.

#### **3.4.4.4 Material Y Equipo Necesario**

1. Documentación sobre la empresa de contexto (descripción de procesos, organigrama, datos relevantes, etc.).
2. Acceso a una herramienta de modelado, como Microsoft Visio, Lucidchart o herramientas similares.
3. Plantillas de notación estándar para el modelado de negocios, como BPMN (Business Process Model and Notation) o UML (Unified Modeling Language).

#### **3.4.4.5 Metodología**

1. Selección de la empresa de contexto: Los estudiantes deben elegir una empresa o entidad específica a la que aplicarán el modelado de negocios.
2. Recopilación de información: Los estudiantes deben recopilar información sobre la empresa, incluyendo sus procesos, estructura organizativa, flujos de trabajo, y otros datos relevantes.
3. Elaboración del modelado: Utilizando la notación estándar seleccionada, los estudiantes deben crear diagramas que representen los procesos, actividades, flujos de información y la estructura organizativa de la empresa.
4. Análisis de resultados: Los estudiantes deben analizar el modelado de negocios para identificar áreas de mejora, ineficiencias o posibles oportunidades de optimización en la empresa de contexto.

#### **3.4.4.6 Sugerencias Didácticas**

- Fomentar la colaboración entre los estudiantes para enriquecer el análisis y el modelado.
- Proporcionar ejemplos de empresas de contextos diversos para permitir a los estudiantes abordar situaciones variadas.
- Promover la discusión y el intercambio de ideas sobre cómo la notación estándar facilita la comprensión y comunicación de los procesos empresariales.

#### **3.4.4.7 Reporte Del Alumno**

El alumno debe de realizar la actividad detallando paso a paso la elaboración de esta, incluyendo capturas, mediante el formato de un reporte de prácticas dando detalle de los resultados obtenidos, así como su conclusión y aprendizajes obtenidos.

#### **3.4.4.8 Bibliografías**

- [¿Qué es la notación de modelado de procesos de negocio? | Lucidchart.](#)
- [Modelo y notación de procesos de negocio \(BPMN\) | Microsoft Visio](#)

### **3.4.5 Práctica 5 Elaborar diagramas de casos de uso para el modelado de negocios sobre situaciones planteadas por el docente, utilizando herramientas CASE.**

#### **3.4.5.1 Objetivo**

El objetivo de esta práctica es desarrollar diagramas de casos de uso para el modelado de negocios, abordando situaciones planteadas por el docente, utilizando herramientas CASE, y posteriormente, analizar los resultados obtenidos.

#### **3.4.5.2 Introducción**

Los diagramas de casos de uso son una herramienta esencial en el proceso de modelado de negocios. Ayudan a representar la interacción entre actores y el sistema en un escenario particular, identificando los requisitos funcionales clave. Esta práctica tiene como objetivo fomentar la comprensión y habilidades en el diseño de sistemas, lo cual es fundamental para profesionales en el campo de la informática y la ingeniería de software.

#### **3.4.5.3 Especificar La Correlación Con El O Los Temas Y Subtemas Del Programa De Estudio Vigente.**

3.4 Obtención de requisitos.

3.4.1 Técnicas de recopilación de información.

3.4.2 Representación de requisitos.

3.5 Herramientas CASE para la ingeniería de requisitos.

#### **3.4.5.4 Material Y Equipo Necesario**

1. Herramientas CASE (Computer-Aided Software Engineering) para la creación de diagramas de casos de uso.
2. Acceso a la descripción de situaciones planteadas por el docente.

#### **3.4.5.5 Metodología**

- El docente proporcionará situaciones específicas de negocios que deben ser modeladas.
- Utilizando una herramienta CASE, los estudiantes crearán diagramas de casos de uso que reflejen la interacción entre actores y el sistema en cada escenario.

- Se documentarán los actores involucrados, los casos de uso, y las relaciones entre ellos.
- Los estudiantes presentarán sus diagramas y discutirán sus decisiones de modelado con el docente y los compañeros.
- Se llevará a cabo un análisis de los resultados, identificando posibles mejoras en la representación de los casos de uso y la comprensión de los requisitos.

#### **3.4.5.6 Sugerencias Didácticas**

- Fomentar la colaboración entre los estudiantes para discutir y validar los diagramas de casos de uso.
- Enfatizar la importancia de la comunicación efectiva con los stakeholders y la comprensión precisa de las necesidades del negocio.
- Incentivar la exploración de herramientas CASE populares y su integración en el proceso de desarrollo de software.
- Promover la reflexión crítica sobre las decisiones de modelado y cómo pueden impactar en el éxito del proyecto.

#### **3.4.5.7 Reporte Del Alumno**

El alumno debe de realizar la actividad detallando paso a paso la elaboración de esta, incluyendo capturas, mediante el formato de un reporte de prácticas dando detalle de los resultados obtenidos, así como su conclusión y aprendizajes obtenidos.

#### **3.4.5.8 Bibliografías**

- [Caso4.pdf \(scalahed.com\).](#)
- [Nuevo Enfoque para la Enseñanza del Paradigma MDD: Ingeniería de Requerimientos Basada en Modelos apoyada por Tecnologías CASE \(scielo.org.ar\)](#)



### **3.4.6 Práctica 6 Elaborar algunas llamadas al sistema y distinguir el mecanismo de las interrupciones en el contexto de los Procesos**

#### **3.4.6.1 Objetivo**

El objetivo de esta práctica es aprender a obtener requisitos específicos (funcionales y no funcionales) para un caso práctico mediante el uso de herramientas CASE y presentar los resultados a través de un informe técnico que incluye diagramas de Casos de Uso y de Actividades. Además, se busca realizar un análisis de los resultados para evaluar su adecuación y viabilidad.

#### **3.4.6.2 Introducción**

La obtención de requisitos es una fase crítica en el desarrollo de software. En esta práctica, se simula la obtención de requisitos para un caso práctico con herramientas CASE. Los requisitos funcionales y no funcionales son documentados en diagramas de Casos de Uso y de Actividades, lo que facilita la comprensión y comunicación de las necesidades del sistema.

#### **3.4.6.3 Especificar La Correlación Con El O Los Temas Y Subtemas Del Programa De Estudio Vigente.**

##### **4.1. Clases**

##### 4.2. Objetos

##### 4.3. Modelo de requisitos

##### 4.4. Modelo de casos de uso

##### 4.5. Modelo de dominio.

#### **3.4.6.4 Material Y Equipo Necesario**

1. Herramientas CASE (por ejemplo, Enterprise Architect, Lucidchart, o herramientas similares).
2. Descripción del caso práctico proporcionada por el docente.

#### **3.4.6.5 Metodología**

##### **Entendimiento del Caso Práctico:**

- Los estudiantes deben analizar y comprender el caso práctico proporcionado por el docente, identificando los actores y los objetivos del sistema.

### **Utilización de Herramientas CASE:**

- Utilizando una herramienta CASE, los estudiantes deben crear diagramas de Casos de Uso para representar la interacción entre actores y el sistema.
- Para los requisitos no funcionales, se deben utilizar diagramas de Actividades para ilustrar procesos clave del sistema.

### **Documentación de Requisitos:**

- Se deben documentar los requisitos funcionales y no funcionales identificados durante el proceso, proporcionando una breve descripción de cada uno.

### **Presentación de Resultados:**

- Los estudiantes deben presentar los diagramas de Casos de Uso y de Actividades, junto con una explicación de los requisitos obtenidos.

**ANÁLISIS DE RESULTADOS:** En esta etapa, se realiza un análisis crítico de los resultados obtenidos:

- Se evalúa si los requisitos funcionales y no funcionales reflejan con precisión las necesidades del sistema.
- Se discuten posibles desafíos o ambigüedades en los requisitos identificados.
- Se considera la viabilidad técnica y económica de implementar los requisitos.

#### **3.4.6.6 Sugerencias Didácticas**

- Fomentar la colaboración entre estudiantes para discutir y validar los requisitos obtenidos.
- Enfatizar la importancia de la comunicación efectiva con los stakeholders y la comprensión precisa de las necesidades del sistema.
- Promover la utilización de herramientas CASE populares y su integración en el proceso de obtención de requisitos.

- Destacar la importancia del análisis de requisitos como un paso crítico en el desarrollo de software.

#### **3.4.6.7 Reporte Del Alumno**

El alumno debe de realizar la actividad detallando paso a paso la elaboración de esta, incluyendo capturas, mediante el formato de un reporte de prácticas dando detalle de los resultados obtenidos, así como su conclusión y aprendizajes obtenidos.

#### **3.4.6.8 Bibliografías**

- [75204.pdf \(pucesa.edu.ec\)\)](https://pucesa.edu.ec/75204.pdf)
- [earchivo.uc3m.es/bitstream/handle/10016/26008/PFC Dalila Garcia Notario.pdf](https://earchivo.uc3m.es/bitstream/handle/10016/26008/PFC_Dalila_Garcia_Notario.pdf)

### **3.4.7 Práctica 7 Revisar un estudio de casos que involucre un producto de software y determinar el nivel de madurez de acuerdo a diferentes estándares como SPICE, CMMI, MoProSoft o ISO 29110.**

#### **3.4.7.1 Objetivo**

El objetivo de esta práctica es revisar un estudio de casos que involucre un producto de software y determinar su nivel de madurez de acuerdo a diferentes estándares de calidad, tales como SPICE, CMMI, MoProSoft o ISO 29110. La práctica busca desarrollar habilidades en la evaluación de la calidad del software y su conformidad con estándares reconocidos.

#### **3.4.7.2 Introducción**

La calidad del software es crucial para el éxito de los proyectos y la satisfacción del cliente. Los estándares de calidad, como SPICE, CMMI, MoProSoft e ISO 29110, ofrecen marcos de referencia para evaluar la madurez de los procesos de desarrollo de software. Esta práctica se enfoca en la revisión de un estudio de casos de software y la determinación de su nivel de madurez siguiendo estos estándares.

#### **3.4.7.3 Especificar La Correlación Con El O Los Temas Y Subtemas Del Programa De Estudio Vigente.**

5.1 Definición de calidad.

5.2 Importancia de la calidad.

5.3 Factores de calidad.

5.4 Aseguramiento de la calidad.

5.5 Estándares y métricas de calidad.

5.6 Modelos de madurez

#### **3.4.7.4 Material Y Equipo Necesario**

- Estudio de casos de un producto de software.
- Documentación relacionada con el estudio de casos.
- Acceso a los estándares de calidad, como SPICE, CMMI, MoProSoft e ISO 29110.

#### **3.4.7.5 Metodología**

Selección del Estudio de Casos:

1. Los estudiantes seleccionan un estudio de casos que involucre un producto de software y su documentación relacionada.

Revisión del Estudio de Casos:

2. Se revisa minuciosamente el estudio de casos, prestando especial atención a los procesos de desarrollo y calidad del software.

Evaluación del Nivel de Madurez:

3. Se determina el nivel de madurez del estudio de casos de acuerdo a uno o varios de los estándares mencionados (SPICE, CMMI, MoProSoft o ISO 29110).

Documentación de Resultados:

4. Se documentan los hallazgos relacionados con el nivel de madurez, incluyendo fortalezas y áreas de mejora.

**ANÁLISIS DE RESULTADOS:** En esta etapa, se realiza un análisis crítico de los resultados:

5. Se evalúa si el estudio de casos cumple con los requisitos y procesos establecidos por los estándares de calidad.
6. Se identifican posibles beneficios de la implementación de mejores prácticas y procesos de madurez.
7. Se discuten las implicaciones de los resultados en términos de mejora de calidad y eficiencia.

#### **3.4.7.6 Sugerencias Didácticas**

- Fomentar la investigación y la experimentación en el sistema operativo simulado para que los participantes profundicen en su comprensión.
- Promover la documentación detallada de los pasos realizados y de las observaciones relevantes.

- Facilitar la discusión y el intercambio de conocimientos entre los participantes para abordar desafíos y descubrimientos.

#### **3.4.7.7 Reporte Del Alumno**

El alumno debe de realizar la actividad detallando paso a paso la elaboración de esta, incluyendo capturas, mediante el formato de un reporte de prácticas dando detalle de los resultados obtenidos, así como su conclusión y aprendizajes obtenidos.

#### **3.4.7.8 Bibliografías**

- [Análisis de experiencias de mejora de procesos de desarrollo de software en PYMEs \(redalyc.org\)\)](https://redalyc.org/)
- [cimat.repositorioinstitucional.mx/jspui/bitstream/1008/535/1/ZAC TE 53.pdf](http://cimat.repositorioinstitucional.mx/jspui/bitstream/1008/535/1/ZAC%20TE%2053.pdf)
- [Tesis.pdf \(unam.mx\)](https://tesis.unam.mx/)