

CARRERA DE COMPUTACIÓN

1. Datos Informativos

- 1.1. Módulo:
- 1.2. Nivel:
- 1.3. Apellidos y Nombres: Samira Narvaez – Estiven Mendez -Almer
Paguay
- 1.4. Tema: Aplicación del Modelo CMMI en el Desarrollo de Software
- 1.5. Fecha: 16-10-2025

2. Objetivo

Aplicar el modelo CMMI en el desarrollo de software para mejorar la calidad de los procesos, optimizar la gestión de proyectos y promover una cultura de mejora continua dentro de las organizaciones, garantizando productos más confiables, eficientes y alineados con las necesidades del cliente.

3. Contenido

Taller: Aplicación del Modelo CMMI en el Desarrollo de Software

1. Datos Generales

Modalidad: Presencial o virtual

Dirigido a: Estudiantes de ingeniería de software, sistemas o computación

Facilitador: Jorge Humberto Miranda Realpe

2. Objetivo General

Comprender los niveles de madurez del modelo CMMI y su aplicación práctica en procesos de desarrollo de software, mediante el análisis de casos y actividades colaborativas.

3. Objetivos Específicos

- Identificar los cinco niveles de madurez del modelo CMMI.
- Analizar las características y prácticas asociadas a cada nivel.
- Aplicar los principios del CMMI a un caso real o simulado de desarrollo de software.

4. Contenidos

Unidad 1. Introducción al CMMI

Demoras constantes en el desarrollo:

Los equipos rara vez cumplen con las fechas pactadas. Cada proyecto parece tomar el doble del tiempo estimado inicialmente, y nadie sabe exactamente por qué sucede esto.

Errores frecuentes:

Los bugs aparecen después del lanzamiento, cuando los usuarios ya están utilizando las aplicaciones. Muchos de estos problemas pudieron haberse detectado antes, pero simplemente nadie los vio.

Clientes insatisfechos:

Las expectativas de los clientes no coinciden con lo que reciben. Hay desconexión entre lo que esperan y lo que el equipo entrega, generando frustración en ambas partes.

Preguntas guía para la discusión:

- ¿Qué patrones observan en estos problemas?
- ¿Creen que hay procesos establecidos en esta empresa?
- ¿Cómo creen que trabaja el equipo día a día?

Unidad 2. Niveles de madurez del CMMI

• Nivel 1: Inicial

¿Por qué llegamos a esta conclusión?

Observando la realidad de SoftEdu, podemos ver que la empresa opera en modo "apagar incendios". No existen procesos documentados ni estándares que el equipo siga de manera consistente. Cada desarrollador trabaja a su manera, haciendo lo mejor que puede con los recursos que tiene, pero sin una guía clara. Cuando surge un problema, todo el equipo entra en pánico y trata de solucionarlo sobre la marcha. Esta forma de trabajar es insostenible y explica perfectamente los síntomas que presenta la empresa: imprevisibilidad total, falta de control y resultados inconsistentes.

En este nivel, el éxito depende más de héroes individuales que de procesos sólidos. Si alguien clave se enferma o renuncia, todo se desmorona porque el conocimiento está en las cabezas de las personas, no en sistemas organizacionales.

- Nivel 2: Gestionado
 - Nivel 3: Definido
 - Nivel 4: Cuantitativamente gestionado
 - Nivel 5: Optimización continua
- Unidad 3. Aplicación práctica
- Identificación del nivel de madurez en un caso empresarial.
 - Propuesta de mejora de procesos.

5. Metodología

- Trabajo en grupo
- Conclusión y autoevaluación.

6. Actividad Práctica

Caso: Empresa “SoftEdu” que desarrolla aplicaciones educativas, pero presenta demoras, errores y baja satisfacción del cliente.

Imaginen que trabajan como consultores para "SoftEdu", una empresa que desarrolla aplicaciones educativas. Últimamente, la empresa está atravesando situaciones complicadas: los proyectos se entregan tarde, aparecen errores inesperados en producción y los clientes expresan su insatisfacción constantemente. El equipo directivo les ha contratado para diagnosticar qué está pasando y proponer soluciones viables.

Tareas:

1. Situación Actual de SoftEdu

Trabajen en equipos de 3-4 personas y analicen los siguientes síntomas que presenta SoftEdu:

Demoras constantes en el desarrollo:

Los equipos rara vez cumplen con las fechas pactadas. Cada proyecto parece tomar el doble del tiempo estimado inicialmente, y nadie sabe exactamente por qué sucede esto.

Errores frecuentes:

Los bugs aparecen después del lanzamiento, cuando los usuarios ya están

utilizando las aplicaciones. Muchos de estos problemas pudieron haberse detectado antes, pero simplemente nadie los vio.

Clientes insatisfechos:

Las expectativas de los clientes no coinciden con lo que reciben. Hay desconexión entre lo que esperan y lo que el equipo entrega, generando frustración en ambas partes.

Preguntas guía para la discusión:

- ¿Qué patrones observan en estos problemas?
- ¿Creen que hay procesos establecidos en esta empresa?
- ¿Cómo creen que trabaja el equipo día a día?

2. Identificación del Nivel CMMI Actual

Nuestra evaluación:

Nivel 1 - Inicial (Caótico)

Con base en lo analizado, determinen en qué nivel de madurez CMMI se encuentra actualmente SoftEdu.

¿Por qué llegamos a esta conclusión?

Observando la realidad de SoftEdu, podemos ver que la empresa opera en modo "apagar incendios". No existen procesos documentados ni estándares que el equipo siga de manera consistente. Cada desarrollador trabaja a su manera, haciendo lo mejor que puede con los recursos que tiene, pero sin una guía clara. Cuando surge un problema, todo el equipo entra en pánico y trata de solucionarlo sobre la marcha. Esta forma de trabajar es insostenible y explica perfectamente los síntomas que presenta la empresa:

imprevisibilidad total, falta de control y resultados inconsistentes.

En este nivel, el éxito depende más de héroes individuales que de procesos sólidos. Si alguien clave se enferma o renuncia, todo se desmorona porque el conocimiento está en las cabezas de las personas, no en sistemas organizacionales.

Propuesta 3: Incorporar Revisiones de Calidad en el Proceso

¿Qué haremos específicamente?

Introduciremos momentos específicos donde el equipo revisa el trabajo antes de considerarlo terminado. Esto incluye revisiones de código entre compañeros (code reviews) donde otro desarrollador verifica que el código sea claro, funcione correctamente y siga buenas prácticas. También estableceremos pruebas de aceptación donde verificamos que cada funcionalidad cumpla con los requisitos definidos.

Crearemos una checklist simple de calidad: ¿funciona según lo especificado? ¿maneja casos de error? ¿es eficiente? ¿está documentado? Solo cuando pase estas verificaciones, se marca como completo.

¿Por qué esto ayuda?

Los bugs llegan a producción porque nadie los detectó antes. Al incorporar revisiones sistemáticas, atrapamos los errores en etapas tempranas, cuando son fáciles y baratos de corregir. Además, el equipo aprende continuamente de estas revisiones, mejorando sus habilidades con cada iteración.

Propuestas de Mejora para Avanzar al Nivel 2 (Gestionado)

Acción 1: Establecer Planificación y Seguimiento de Proyectos

Descripción: Definir planes detallados (alcance, cronograma, recursos).

Junto con la planificación realista de los proyectos.

Justificación:

- Reducir las demoras mediante planificación realista
- Minimización de errores por mala interpretación de requisitos
- Mejora de la satisfacción al cumplir expectativas del cliente

Acción 2: Implementar Gestión de control de calidad y gestión de configuración

Descripción:

Implementar revisiones de código, pruebas sistemáticas y control de versiones del software

Controlar cambios mediante un proceso formal.

Justificación:

- Minimiza errores por malentendidos en requisitos.

- Detección de errores tempranamente reduciendo costos de **corrección**

- Facilitación de mantenimiento y actualizaciones
- Mejora la satisfacción al alinear expectativas.

Acción 3: Introducir Revisiones de Calidad

Descripción:

Implementar reuniones periódicas de seguimiento, métricas básicas de progreso y mecanismo de control de cambios.

Justificación:

- Permiso para identificar desviaciones tempranamente
- Mejora de la predictibilidad de entregas
- Incremento de transparencia con el cliente
- Eleva la calidad del producto final.

Beneficios esperados con la implementación:

- Mejorar en calidad del software
 1. Reducción de errores
 2. Cumplimiento de plazos
 3. Satisfacción con el cliente
 4. Procesos repetibles -> base para una mejora continua

Impacto en la organización:

- Transición de procesos ad-hoc a procesos gestionados
- Mayor predictibilidad en resultados
- Mejora en la reputación de la empresa
- Reducción de costos por retrabajos

7. Recursos

Para que este taller sea efectivo y todos puedan participar activamente, necesitaremos preparar lo siguiente:

Materiales de presentación:

Prepararemos una presentación visual clara y dinámica con los conceptos clave del CMMI. No queremos saturar con texto, sino usar diagramas, ejemplos visuales y casos breves que ilustren cada nivel de madurez. La idea es que los conceptos se entiendan fácilmente, no que memoricen definiciones.

Documentos de trabajo:

Cada equipo recibirá una guía impresa o digital con el caso de SoftEdu detallado, incluyendo espacio para que anoten sus observaciones, diagnóstico y propuestas. También incluiremos una hoja de referencia rápida con las características de cada nivel CMMI para que puedan consultarla durante el análisis.

Herramientas tecnológicas:

Si trabajamos presencialmente, necesitaremos un proyector, pizarra y marcadores para que los grupos compartan sus ideas. Si es virtual, usaremos una plataforma colaborativa donde puedan trabajar en equipos pequeños (Zoom con breakout rooms, Google Meet, etc.) y un documento compartido (Google Docs o Miro) donde todos puedan aportar simultáneamente.

Material de apoyo adicional:

Proporcionaremos acceso a recursos digitales complementarios: un documento PDF con la descripción completa del modelo CMMI, enlaces a artículos cortos sobre casos de éxito, y quizás un video breve (5-10 minutos) que muestre cómo una empresa real implementó mejoras basadas en CMMI.

8. Evaluación

La evaluación de este taller será integral y reflejará tanto el proceso como los resultados. Queremos valorar no solo lo que aprendieron, sino cómo lo aplicaron.

Criterio 1: Participación Activa y Colaboración (30%)

Aquí evaluaremos cómo se involucraron en la actividad. ¿Aportaron ideas durante las discusiones en equipo? ¿Escucharon a sus compañeros y construyeron sobre sus propuestas? ¿Hicieron preguntas relevantes o compartieron experiencias propias que enriquecieron el análisis?

No buscamos que hablen todo el tiempo, sino que cuando lo hagan, sus aportes demuestren que están pensando críticamente sobre el caso. La colaboración genuina es clave: ayudar a otros a entender conceptos, mediar

cuando hay desacuerdos, integrar diferentes perspectivas en una propuesta coherente.

Indicadores concretos:

- Contribuye con ideas originales o perspectivas útiles
- Se involucra en las discusiones sin dominarlas ni ausentarse
- Ayuda a su equipo a avanzar cuando se atascan
- Hace conexiones entre la teoría CMMI y el caso práctico

Criterio 2: Análisis del Caso (40%)

Este es el componente más importante. Evaluaremos qué tan bien diagnosticaron la situación de SoftEdu. ¿Identificaron correctamente que la empresa está en el Nivel 1? Más importante aún: ¿su justificación tiene sentido? No queremos respuestas memorizadas, sino razonamiento lógico. Buscaremos que establezcan conexiones claras entre los síntomas observados (demoras, errores, insatisfacción) y las características del nivel de madurez identificado. ¿Pueden explicar por qué estos problemas son típicos de una organización en el Nivel Inicial? ¿Entienden la diferencia entre estar en Nivel 1 versus estar en Nivel 2?

Indicadores concretos:

- Identifica correctamente el nivel de madurez actual
- Proporciona evidencias del caso que respaldan su diagnóstico
- Explica con claridad por qué la empresa está en ese nivel
- Demuestra comprensión de las diferencias entre niveles

Criterio 3: Propuesta de Mejora (30%)

Finalmente, evaluaremos la calidad y viabilidad de sus propuestas. ¿Son acciones concretas o ideas vagas? ¿Podrían realmente implementarse en una empresa pequeña como SoftEdu? ¿Atacan las causas raíz de los problemas o solo los síntomas?

Las mejores propuestas serán aquellas que sean específicas (no "mejorar la comunicación" sino "implementar reuniones semanales de 30 minutos con agenda definida"), realistas (considerando recursos limitados) y alineadas con el objetivo de avanzar al Nivel 2.

Indicadores concretos:

- Las acciones propuestas son específicas y detalladas
- Explican cómo cada acción ayuda a resolver problemas específicos

- Las propuestas son realistas y aplicables
- Demuestran comprensión de lo que implica el Nivel 2

9. Resultados Esperados

Al finalizar este taller, esperamos que los participantes hayan logrado lo siguiente:

Comprensión profunda del modelo CMMI:

Los estudiantes no solo conocerán los cinco niveles de madurez de memoria, sino que entenderán genuinamente qué representa cada uno. Podrán explicar con sus propias palabras la diferencia entre una organización caótica (Nivel 1) y una que tiene procesos gestionados (Nivel 2), o entre tener procesos definidos (Nivel 3) y optimizarlos continuamente (Nivel 5). Más allá de las definiciones técnicas, comprenderán la evolución natural que atraviesan las organizaciones de software.

Capacidad de análisis aplicado:

Desarrollarán la habilidad de observar una situación real de desarrollo de software, identificar sus problemas subyacentes y diagnosticar en qué nivel de madurez se encuentra. Esta competencia es tremendamente valiosa porque les permitirá, en su futura vida profesional, evaluar rápidamente la salud de los procesos en cualquier proyecto u organización donde trabajen.

Pensamiento orientado a la mejora continua:

Habrán internalizado que los problemas en el desarrollo de software raramente se resuelven con esfuerzo heroico individual, sino con procesos sólidos y mejora sistemática. Entenderán que avanzar en madurez no es un lujo sino una necesidad para ser competitivos y entregar valor real a los clientes. Esta mentalidad de mejora continua es quizás el aprendizaje más importante del taller.

Habilidades prácticas de propuesta:

Sabrán diseñar acciones concretas de mejora que sean realistas, específicas y

alineadas con los principios del CMMI. Podrán justificar por qué ciertas acciones son prioritarias y cómo contribuyen a resolver problemas específicos. Esta capacidad de proponer soluciones fundamentadas es esencial para cualquier ingeniero de software que aspire a roles de liderazgo.

Valoración de la disciplina en ingeniería de software:

Finalmente, esperamos que desarrollen un genuino aprecio por la importancia de los procesos en la ingeniería de software. Entenderán que la disciplina, la documentación y los estándares no son burocracia innecesaria, sino herramientas que permiten construir software de calidad de manera sostenible. Saldrán del taller con la convicción de que la mejora de procesos es tan importante como la habilidad técnica para programar.

4. Conclusiones

La aplicación del modelo **CMMI** en el desarrollo de software representa una herramienta clave para fortalecer la calidad, la eficiencia y la organización de los procesos dentro de una empresa. Su enfoque basado en niveles de madurez permite identificar debilidades y establecer estrategias de mejora continua que impactan positivamente en los resultados. Implementar CMMI no solo ayuda a cumplir plazos y reducir errores, sino que también promueve una cultura de trabajo más planificada, colaborativa y orientada a la satisfacción del cliente. En definitiva, este modelo impulsa a las organizaciones a evolucionar de manera constante hacia la excelencia en la ingeniería de software.