

CARRERA DE COMPUTACIÓN

1. Datos Informativos

- 1.1. Módulo: 1
- 1.2. Nivel: 7
- 1.3. Apellidos y Nombres: Solis Ramírez Aida Milena – Valdivieso
Aslalema Jeremy – Valenzuela Quisiquinga Cristian Sebastian
- 1.4. Tema: Taller: Aplicación del Modelo CMMI en el Desarrollo de Software
- 1.5. Fecha: 16-10-2025

2. Objetivo

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.

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.

3. Contenido

Unidad 1. Introducción al CMMI

Origen y Propósito del Modelo en el Contexto del Proyecto

El CMMI, que significa Modelo de Madurez de Capacidades Integrado, es un marco que se utiliza para mejorar los procesos, como lo hace la norma ISO 9001:2015 que basa tu proyecto. Su principal meta es estandarizar y perfeccionar la manera en que se desarrolla software. En el caso de la "Plataforma Integral de Gestión Médica en Línea", la intención de aplicar un modelo de madurez es transformar los desafíos en el ámbito digital de la salud en oportunidades de calidad y confianza elevadas. Se busca asegurar que cada etapa, desde la planificación hasta el soporte, cuente con un control adecuado y esté alineada con los objetivos de calidad, garantizando así la prestación de un servicio confiable, seguro y centrado en el usuario.

Beneficios de su Implementación para la Plataforma Médica

Incorporar un marco de madurez como el CMMI, fundamentado en los principios de la norma ISO 9001, no es simplemente un procedimiento administrativo, sino una decisión estratégica crucial. En cuanto a tu proyecto, los beneficios inmediatos incluyen:

Mejora en la Calidad y Seguridad: Poner énfasis en la seguridad del paciente y la protección de datos mediante procesos controlados, como revisiones de código obligatorias y pruebas de seguridad cada tres meses.

Disminución de Costos y Riesgos: Planificar teniendo en cuenta los riesgos ayuda a anticipar y resolver problemas críticos, como filtraciones de datos o interrupciones en el servicio, lo que reduce el gasto en correcciones y retrabajos.

Aumento de la Satisfacción del Cliente: Al centrar la atención en las necesidades de los pacientes, médicos y centros de salud y evaluar la satisfacción a través de métricas como el Net Promoter Score (NPS), se garantiza que el producto final satisfaga e idealmente supere sus expectativas.

Mejoramiento de la Eficiencia Operativa: La estandarización de procesos, como la implementación de la metodología Scrum y la automatización de despliegues (CI/CD), optimiza el tiempo y los recursos del equipo.

Unidad 2. Niveles de Madurez del CMMI

Estos niveles muestran la evolución de los métodos de desarrollo de la plataforma, desde un comienzo incierto hasta una fase optimizada.

Nivel 1: Inicial

En esta fase, el proceso de desarrollo de la plataforma se caracterizaría por su inestabilidad y falta de orden. No habría procedimientos bien definidos, y el éxito dependería del esfuerzo individual de cada programador. Las exigencias de los pacientes y médicos no se documentarían de manera formal, lo que aumentaría el riesgo de incumplimiento con normativas como la HIPAA o la LOPDGDD, resultando en una calidad del software final inconsistente.

Nivel 2: Gestionado

En este nivel, la administración de la plataforma empieza a estar controlada a nivel de proyecto. Se crearían planes básicos para cada módulo (como la programación de citas y servicios de telemedicina). Se utilizarían

herramientas como Jira para registrar necesidades y se realizarían seguimientos para observar el avance. Aunque hay una mejora en comparación con el nivel inicial, los procedimientos pueden variar significativamente entre diferentes equipos o módulos.

Nivel 3: Definido

Este es el nivel en que la organización comienza a estandarizar sus procedimientos. Para la plataforma médica, esto implica que todos los equipos de desarrollo adoptarían la misma metodología (Scrum), la misma estrategia de control de código (GitFlow) y los mismos criterios de calidad, como la "Definición de Hecho". Los procesos de diseño, pruebas y despliegue estarían bien documentados y todo el personal recibiría capacitación para seguirlos, asegurando así la consistencia en la calidad de toda la plataforma.

Nivel 4: Cuantitativamente Gestionado

La administración de la plataforma se convierte en un proceso basado en datos. No es suficiente con seguir los procedimientos; la organización mide su rendimiento usando métricas específicas. Se establecen y supervisan objetivos cuantitativos SMART, como lograr un 99.9% de disponibilidad, mantener el tiempo de carga de historiales clínicos por debajo de 2 segundos, o resolver tickets urgentes en menos de 4 horas. Se realizan análisis estadísticos para supervisar los procesos y anticipar su rendimiento.

Nivel 5: Optimización Continua

En el nivel máximo de madurez, la organización se enfoca en la innovación y en la mejora continua de los procesos de la plataforma. Se investigan las causas raíz de los problemas para perfeccionar el sistema que los genera. Nuevas tecnologías, como la inteligencia artificial para diagnósticos asistidos, se evalúan e implementan de manera proactiva, y se utiliza la retroalimentación de los usuarios no solo para hacer correcciones, sino también para innovar y buscar la excelencia de forma permanente.

Unidad 3. Aplicación Práctica

Caso Empresarial: "Plataforma Integral de Gestión Médica en Línea"

1. Identificación del nivel de madurez en un caso empresarial

Al analizar el diseño del Sistema de Gestión de Calidad (SGC) registrado, se concluye que el proyecto está estructurado para funcionar en un Nivel 3.

Definido, con metas y componentes claramente definidos para alcanzar el

Nivel 4: Gestionado Cuantitativamente.

Justificación pormenorizada:

Evidencia de Nivel 3 (Definido):

Procesos Normalizados y Organizativos: El documento ofrece un enfoque colectivo y obligatorio para todos los participantes, que incluye la aplicación de Scrum en ciclos de dos semanas, una estrategia de ramificación GitFlow que asegura que la rama principal esté siempre lista para su implementación, así como la necesidad de realizar revisiones de código durante todo el proceso de desarrollo.

Documentación y Metodología Común: Se han elaborado criterios normalizados por escrito, como la "Definición de Listo" y la "Definición de Terminado" para cada historia de usuario.

Enfoque en Capacitación: El plan contempla capacitaciones obligatorias anuales acerca del manejo de datos sensibles y seguridad, manteniendo un registro que refleja las competencias del equipo.

Evidencia de Nivel 4 (Gestionado Cuantitativamente):

Definición de Objetivos Cuantitativos: El proyecto presenta claramente los Objetivos de Calidad SMART. Ejemplos específicos incluyen:

Disponibilidad: Lograr un 99.9% de tiempo de operación en los servicios esenciales.

Seguridad: No deben encontrarse vulnerabilidades críticas en las auditorías que se realizan cada trimestre.

Rendimiento: El 95% de las solicitudes para cargar historiales clínicos deben completarse en menos de 2 segundos.

Satisfacción del Usuario: Alcanzar una puntuación NPS mayor a 50.

Medición y Análisis: Se prevé hacer un seguimiento en tiempo real de las métricas de la plataforma (tiempo de actividad, latencia, tasa de errores), métricas de usuarios (encuestas CSAT) y métricas de seguridad.

2. Propuesta de mejora de procesos

El proyecto tiene un esquema muy sólido. Las siguientes recomendaciones buscan reforzar el Nivel 3 y acelerar el avance hacia los Niveles 4 y 5.

Acción 1 (Consolidar Nivel 3): Formalizar y Ejecutar el Programa de Auditorías Internas de Procesos.

Propuesta Específica: El plan ya incluye auditorías internas cada seis meses. Para fortalecer esto, es importante crear una "Lista de Verificación de Cumplimiento de Procesos Definidos" basada en el SGC. Esta lista verificaría aspectos concretos mencionados en el documento, como: ¿Se llevaron a cabo todas las ceremonias de Scrum? ¿Todas las historias de usuario cumplen con la "Definición de Terminado"? ¿Existen pruebas de las revisiones de código para cada combinación en la rama principal? Los resultados generarían un informe de "no conformidades" junto a un plan de acción, garantizando que los procesos establecidos se ejerciten de manera rigurosa.

Mejora en Calidad: Esto asegura la consistencia y la disciplina en la implementación, eliminando variantes que suelen ser la causa de errores y asegurando una calidad homogénea en toda la plataforma.

Acción 2 (Progresar a Nivel 4): Implantar Control Estadístico de Procesos (SPC) para las Métricas Principales.

Propuesta Detallada: En lugar de limitarse a verificar si el tiempo de operación alcanza el 99. 9% (un solo dato), es fundamental utilizar gráficos de control para evaluar la estabilidad del proceso que genera dicho resultado. Por ejemplo, se podría crear un gráfico diario que represente la latencia de la API. Un gráfico estadístico de control revelaría tanto el promedio como los límites establecidos. Si un dato se aleja de esos límites, esto indica una variación que debe ser revisada de inmediato, permitiendo al equipo comprender mejor la verdadera capacidad de sus procesos y predecir su desempeño.

Mejora en Calidad: Permite adoptar un enfoque preventivo en lugar de uno reactivo. El equipo puede identificar tendencias negativas y actuar antes de que se afecte un estándar de calidad, lo que aumenta de manera significativa la confianza y la eficacia del servicio.

Acción 3 (Avanzar hacia el Nivel 5): Establecer un "Comité de Optimización e Innovación" Basado en Datos.

Propuesta Detallada: Organizar el ciclo de mejora continua mencionado en la cláusula 10 del proyecto. Se sugiere crear un comité que se reúna cada

tres meses, bajo la dirección del "Director del Proyecto", e incluyendo al Líder

Técnico, al Líder de Calidad y al Líder de Soporte. La agenda incluirá la evaluación de los resultados del Nivel 4 (informes de auditoría, gráficos de SPC, análisis de causas raíz de incidentes) para sugerir e implementar mejoras innovadoras, como las oportunidades comentadas anteriormente (por ejemplo, usar IA para pre-diagnóstico), asignando recursos para su desarrollo.

Mejora en Calidad: Esto establece la innovación y la optimización como parte de la cultura organizacional. Garantiza que la entidad no se detenga, sino que utilice sus propios datos para avanzar de forma inteligente y sostenible, lo que refleja el más alto grado de madurez organizacional.

MÉTRICAS DE SEGUIMIENTO CMMI

Categoría	Métrica	Objetivo	Frecuencia	Responsable
Calidad del Producto	Defectos por Sprint	< 5 defectos críticos por sprint	Semanal	QA/Tester
	Cobertura de Pruebas	> 80% cobertura de código	Por Sprint	Desarrolladores
	Vulnerabilidades de Seguridad	0 vulnerabilidades críticas	Por Sprint	Scrum Master
Desempeño de Proceso	Velocidad del Equipo	$\pm 10\%$ de variación entre sprints	Por Sprint	Scrum Master
	Cumplimiento de Fechas	> 90% de tareas completadas a tiempo	Semanal	Product Owner
	Tiempo de Resolución de Bugs	< 8 horas para bugs críticos	Diario	Desarrolladores
Satisfacción del Cliente	NPS Interno	> 40 puntos	Mensual	Product Owner
	Feedback de Usuarios Clave	> 4/5 puntos de satisfacción	Por Sprint	Equipo Completo

PLANIFICACIÓN DETALLADA POR SPRINT - ENFOQUE CMMI

SPRINT 0: INICIO Y PREPARACIÓN (22 Días)

Objetivo CMMI: Establecer procesos gestionados y planificación formal

Tarea	Duración	Entregables CMMI	Prácticas CMMI Aplicadas
Constituir equipo Scrum	3 días	Matriz de responsabilidades	Gestión de Proyectos
Definir Product Backlog	5 días	Backlog priorizado y estimado	Planificación de Proyectos
Establecer entorno desarrollo	7 días	Ambiente configurado y documentado	Gestión de Configuración
Diseñar arquitectura técnica	5 días	Documento de arquitectura	Definición de Procesos
Planificar Sprint 1	2 días	Sprint Backlog detallado	Planificación y Seguimiento

Criterios de Aceptación CMMI:

- Todos los procesos documentados
- Roles y responsabilidades definidos
- Herramientas de gestión implementadas

SPRINT 1: MÓDULO DE AUTENTICACIÓN Y USUARIOS (12 Días)

Objetivo CMMI: Implementar gestión de requisitos y garantía de calidad

Tarea	Duración	Entregables CMMI	Métricas Clave
HU001 - Registro de usuarios	4 días	Formularios validados y seguros	< 3 defectos por funcionalidad
HU002 - Login y recuperación	4 días	Sistema de autenticación robusto	100% pruebas de seguridad
Pruebas de integración	2 días	Reporte de pruebas ejecutadas	> 90% cobertura de pruebas
Revisión y retrospectiva	2 días	Acta de mejora de procesos	100% issues documentados

Prácticas CMMI Aplicadas:

- Gestión de Requisitos (REQM)
- Garantía de Calidad del Proceso y Producto (PPQA)
- Medición y Análisis (MA)

SPRINT 2: PERFILES Y PANEL PRINCIPAL (9 Días)

Objetivo CMMI: Establecer gestión de configuración y seguimiento de métricas

Tarea	Duración	Entregables CMMI	Métricas Clave
Crear/editar perfiles	3 días	Componentes reutilizables	< 2 defectos de UI/UX
Dashboard de inicio	4 días	Panel con métricas operativas	100% componentes versionados
Pruebas de usabilidad	1 día	Reporte de feedback usuarios	> 4/5 satisfacción
Control de versiones	1 día	Release estable etiquetado	0 conflictos de integración

Prácticas CMMI Aplicadas:

- Gestión de Configuración (CM)
- Medición y Análisis (MA)
- Verificación y Validación (VER/VAL)

SPRINT 3: SISTEMA DE CITAS MÉDICAS (10 Días)

Objetivo CMMI: Implementar gestión cuantitativa de proyectos

Tarea	Duración	Entregables CMMI	Métricas Clave
Solicitud de citas (pacientes)	4 días	Flujo completo de agendamiento	< 5% tasa de error
Gestión de agenda (doctores)	4 días	Interface de gestión profesional	100% disponibilidad funcional
Pruebas de carga	1 día	Reporte de performance	< 2s tiempo de respuesta
Análisis métricos	1 día	Dashboard de rendimiento	Todas métricas visibles

Prácticas CMMI Aplicadas:

- Gestión Cuantitativa de Proyectos (QPM)
- Análisis y Decisión de Resolución (DAR)
- Gestión de Riesgos (RSKM)

SPRINT 4: HISTORIAL MÉDICO BÁSICO (5 Días)

Objetivo CMMI: Establecer mejora continua de procesos

Tarea	Duración	Entregables CMMI	Métricas Clave
Visualización historial	2 días	Interface segura y responsive	0 violaciones de seguridad
Registro notas médicas	2 días	Formularios validados	100% datos encriptados
Auditoría de seguridad	1 día	Reporte de cumplimiento	100% requisitos HIPAA

Prácticas CMMI Aplicadas:

- Mejora de Procesos Organizativos (OPF)
- Definición de Procesos Organizativos (OPD)
- Gestión de Desempeño Organizativo (OPM)

FASE FINAL: LANZAMIENTO (3 Días)

Objetivo CMMI: Validación completa y transición a operación

Tarea	Duración	Entregables CMMI	Criterios de Éxito
Despliegue producción	1 día	Sistema en ambiente live	100% funcionalidades operativas
Pruebas finales humo	1 día	Checklist de validación	0 defectos críticos
Lanzamiento oficial	1 día	Comunicado y documentación	100% usuarios capacitados

TABLERO TRELLO CMMI - ESTRUCTURA PROPUESTA

Listas Específicas para CMMI:

1. **BACKLOG CMMI** - Requisitos de proceso pendientes
2. **PLANIFICACIÓN** - Tareas de documentación de procesos
3. **DESARROLLO** - Implementación de prácticas CMMI
4. **VERIFICACIÓN** - Auditorías internas de procesos
5. **VALIDACIÓN** - Pruebas de cumplimiento CMMI
6. **MEJORA CONTINUA** - Acciones de optimización

Etiquetas CMMI:

- Nivel 2** - Gestionado
- Nivel 3** - Definido
- Nivel 4** - Cuantitativo
- Nivel 5** - Optimización
- Auditoría Interna**
- Mejora de Proceso**

CHECKLIST DE VERIFICACIÓN CMMI POR SPRINT

Al Inicio de Cada Sprint:

- Revisar métricas del sprint anterior
- Actualizar documentación de procesos
- Validar cumplimiento de estándares
- Asignar responsables de calidad

Durante el Sprint:

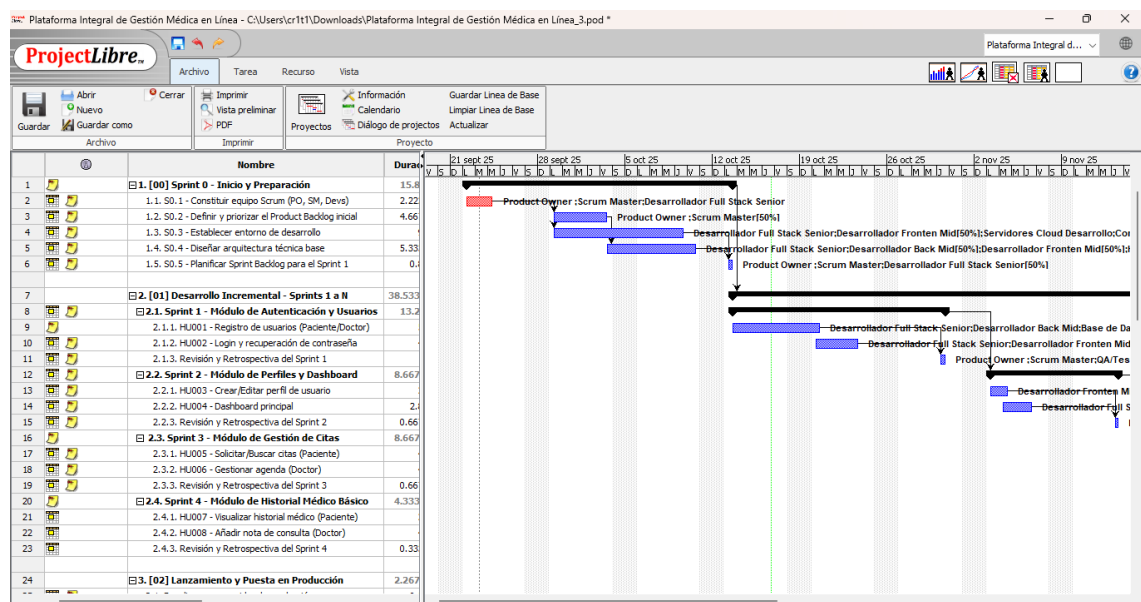
- Ejecutar auditorías internas semanales
- Actualizar dashboard de métricas
- Documentar desviaciones de proceso
- Realizar revisiones de pares

Al Final de Cada Sprint:

- Generar reporte de cumplimiento CMMI
- Actualizar base de conocimientos
- Identificar oportunidades de mejora
- Planificar acciones correctivas

Esta planificación integra completamente los principios CMMI dentro de la metodología ágil, asegurando que cada sprint no solo entregue valor al cliente, sino que también contribuya a la madurez de procesos de la organización, específicamente avanzando del Nivel 1 (Inicial) al Nivel 2 (Gestionado) del modelo CMMI.

Metodología (Trabajo en grupo)



		Nombre	Duración	Inicio	Terminado	Predecesores	Nombres del Recurso
1		1. [00] Sprint 0 - Inicio y Preparación	15.8 days	22/09/25, 08:00	13/10/25, 15:24		
2		1.1. S0.1 - Constituir equipo Scrum (PO, SM, Devs)	2.222 days	22/09/25, 08:00	24/09/25, 09:46		Product Owner ;Scrum Mast...
3		1.2. S0.2 - Definir y priorizar el Product Backlog inicial	4.667 days	29/09/25, 08:00	03/10/25, 14:20	2	Product Owner ;Scrum Mast...
4		1.3. S0.3 - Establecer entorno de desarrollo	9 days	29/09/25, 08:00	09/10/25, 17:00	2	Desarrollador Full Stack Seni...
5		1.4. S0.4 - Diseñar arquitectura técnica base	5.333 days	03/10/25, 14:20	10/10/25, 17:00	3	Desarrollador Full Stack Seni...
6		1.5. S0.5 - Planificar Sprint Backlog para el Sprint 1	0.8 days	13/10/25, 08:00	13/10/25, 15:24	4;5	Product Owner ;Scrum Mast...
7		2. [01] Desarrollo Incremental - Sprints 1 a N	38.533 days	13/10/25, 15:24	05/12/25, 10:40	1	
8		2.1. Sprint 1 - Módulo de Autenticación y Usuarios	13.2 days	13/10/25, 15:24	30/10/25, 17:00		
9		2.1.1. HU001 - Registro de usuarios (Paciente/Doctor)	5 days	13/10/25, 15:24	20/10/25, 15:24		Desarrollador Full Stack Seni...
10		2.1.2. HU002 - Login y recuperación de contraseña	4 days	20/10/25, 08:00	23/10/25, 17:00		Desarrollador Full Stack Seni...
11		2.1.3. Revisión y Retrospectiva del Sprint 1	1 day	30/10/25, 08:00	30/10/25, 17:00	9;10	Product Owner ;Scrum Mast...
12		2.2. Sprint 2 - Módulo de Perfiles y Dashboard	8.667 days	03/11/25, 08:00	13/11/25, 14:20	8	
13		2.2.1. HU003 - Crear/Editar perfil de usuario	2 days	03/11/25, 08:00	04/11/25, 17:00		Desarrollador Fronten Mid;D...
14		2.2.2. HU004 - Dashboard principal	2.8 days	04/11/25, 08:00	06/11/25, 15:24		Desarrollador Full Stack Seni...
15		2.2.3. Revisión y Retrospectiva del Sprint 2	0.667 days	13/11/25, 08:00	13/11/25, 14:20	13;14	Product Owner ;Scrum Mast...
16		2.3. Sprint 3 - Módulo de Gestión de Citas	8.667 days	17/11/25, 08:00	27/11/25, 14:20	12	
17		2.3.1. HU005 - Solicitar/Buscar citas (Paciente)	4 days	17/11/25, 08:00	20/11/25, 17:00		Desarrollador Fronten Mid;D...
18		2.3.2. HU006 - Gestionar agenda (Doctor)	4 days	17/11/25, 08:00	20/11/25, 17:00		Desarrollador Full Stack Seni...
19		2.3.3. Revisión y Retrospectiva del Sprint 3	0.667 days	27/11/25, 08:00	27/11/25, 14:20	17;18	Product Owner ;Scrum Mast...
20		2.4. Sprint 4 - Módulo de Historial Médico Básico	4.333 days	01/12/25, 08:00	05/12/25, 10:40	16	
21		2.4.1. HU007 - Visualizar historial médico (Paciente)	2 days	01/12/25, 08:00	02/12/25, 17:00		Desarrollador Fronten Mid;D...
22		2.4.2. HU008 - Añadir nota de consulta (Doctor)	4 days	01/12/25, 08:00	04/12/25, 17:00		Desarrollador Back Mid;Des...
23		2.4.3. Revisión y Retrospectiva del Sprint 4	0.333 days	05/12/25, 08:00	05/12/25, 10:40	21;22	Product Owner ;Scrum Mast...
24		3. [02] Lanzamiento y Puesta en Producción	2.267 days	08/12/25, 08:00	10/12/25, 10:08	7	

4. Actividad Práctica

Caso: Empresa SoftEdu

Análisis de la Situación Actual:

- Nivel CMMI Identificado: Nivel 1 – Inicial

Justificación:

La empresa SoftEdu presenta características típicas del Nivel 1:

- Procesos caóticos: Las demoras constantes indican ausencia de planificación formal.
- Dependencia individual: El éxito depende del esfuerzo de cada programador sin procedimientos estandarizados.
- Sin documentación: Los errores frecuentes evidencian falta de documentación de requisitos.
- Calidad impredecible: La baja satisfacción del cliente refleja resultados inconsistentes.

Propuesta de Tres Acciones para Avanzar al Nivel 2: Gestionado

Acción	Descripción	Implementación Específica
1. Gestión de Requisitos	Establecer proceso formal para capturar, documentar y validar requisitos del cliente	<ul style="list-style-type: none"> • Crear plantillas de requisitos • Usar herramienta de gestión (Jira/Trello) • Validar requisitos con el cliente antes de desarrollar • Mantener trazabilidad requisito-funcionalidad
2. Planificación y Seguimiento de Proyectos	Implementar planes de proyecto con cronogramas, asignación de recursos y control de avance	<ul style="list-style-type: none"> • Definir roles y responsabilidades • Crear cronogramas con hitos claros • Realizar reuniones semanales de seguimiento • Generar reportes de estado del proyecto
3. Aseguramiento de la Calidad	Establecer procesos de revisión de código, pruebas sistemáticas y control de versiones	<ul style="list-style-type: none"> • Implementar sistema de control de versiones (Git) • Establecer revisiones de código obligatorias • Crear plan de pruebas por funcionalidad • Definir criterios de aceptación para cada entrega

Justificación: Mejora en la Calidad del Software

Acción 1 - Gestión de Requisitos:

- Reduce errores: Al documentar formalmente lo que el cliente necesita, se eliminan malentendidos que causan retrabajo.
- Mejora satisfacción: El cliente recibe exactamente lo que solicitó, aumentando su confianza en la empresa.
- Previene cambios no controlados: La trazabilidad permite gestionar modificaciones de manera ordenada.

Acción 2 - Planificación y Seguimiento:

- Elimina demoras: Los cronogramas realistas y el seguimiento continuo permiten identificar problemas tempranamente.
- Optimiza recursos: La asignación planificada evita sobrecarga de trabajo y mejora la productividad.
- Aumenta previsibilidad: Los clientes reciben estimaciones confiables, mejorando su percepción del servicio.

Acción 3 - Aseguramiento de Calidad:

- Reduce defectos: Las revisiones de código y pruebas detectan errores antes de llegar al cliente.
- Facilita mantenimiento: El control de versiones permite rastrear cambios y revertir errores fácilmente.
- Estandariza calidad: Las revisiones educan al equipo y reducen la dependencia de individuos específicos.

5. Conclusiones

Con este taller, hemos conseguido entender la relevancia de aplicar un modelo de madurez como CMMI en el desarrollo de software. Detectamos que el proyecto de la "Plataforma Integral de Gestión Médica en Línea" está situado entre los niveles 3 (Definido) y 4 (Gestionado Cuantitativamente), con posibilidades evidentes para progresar hacia la optimización continua a través del empleo de métricas, auditorías y comités de innovación. Por otro lado, el análisis de la compañía "SoftEdu" nos permitió reconocer una entidad en el Nivel 1 (Inicial) y sugerir tres medidas específicas para progresar al Nivel 2 (Gestionado), evidenciando que hasta las organizaciones con procesos poco desarrollados tienen la capacidad de cambiar a través de la aplicación sistemática de prácticas CMMI.

Como equipo, llegamos a la conclusión de que esta experiencia confirma la importancia de CMMI como un marco estratégico global, que puede ser utilizado tanto en proyectos críticos del sector salud como en compañías educativas en sus primeras fases. El taller nos ha brindado la oportunidad de adquirir habilidades prácticas para examinar y optimizar los procesos de desarrollo de software, teniendo en cuenta que la implementación de estos niveles de madurez posibilita la estandarización de procesos, la medición de resultados y el perfeccionamiento constante del producto final, lo cual asegura un software confiable, seguro y orientado al usuario.

6. Referencia Bibliografía

- Soto Lopez, D., Garshasbi, M., Kabir, G., Bari, A. B. M. M., & Ali, S. M. (2022). Evaluating interaction between internal hospital supply chain performance indicators: a rough-DEMATEL-based approach. *International Journal of Productivity and Performance Management*, 71(6), 2087-2113. <https://doi.org/10.1108/ijppm-02-2021-0085>
- Castillo, R., Gómez, M., & Pérez, J. (2021). Calidad en el desarrollo de software: Aplicación de la norma ISO 9001 en proyectos de telemedicina. *Revista Iberoamericana de Ingeniería de Software*, 15(2), 45-60.
- ISO 9001:2015. (2023). ISO. <https://www.iso.org/standard/62085.html>
- ISO 13485:2016. (2025). ISO. <https://www.iso.org/standard/59752.html>
- López, A., Fernández, C., & Díaz, K. (2020). Gestión de riesgos en historia clínica electrónica: Un enfoque desde la ISO 31000. *Revista de Informática en la Salud*, 8(1), 22-35.
- Rodríguez, P.; Silva, L. & González, E. (2022). Desafíos en la implementación de sistemas de gestión de calidad ISO 9001 en startups de salud digital. *Journal of Health Informatics Research*, 6(4), 78-95. <https://doi.org/10.1007/s41666-022-00128-9>
- Documento proporcionado por el Docente CMMI_Niveles_de_Madurez_y_Ejemplos.

7. Referencia linkografía