

Introducción al ejercicio

Una empresa de software funciona como una cooperativa con 30 años de antigüedad. Cuenta con una cartera de 30 clientes que utilizan diversos sistemas y soluciones consolidadas. La empresa no aplica ninguna de las metodologías de trabajo mencionadas en este curso sobre: - Ciclo de vida del desarrollo de software en cascada,

- Cascada con testing en cada etapa,
- Xtreme Programming,
- Metodología Lean,
- TDD,
- Proceso Unificado,
- RUP, Etc.

La empresa desea modernizar su stack tecnológico para incorporar nuevas funcionalidades y mejoras a sus productos. Además, busca adquirir conocimientos en herramientas, técnicas de desarrollo e inteligencia artificial para mejorar la eficiencia y la calidad del software.

Estas son las principales problemáticas que se requiere solucionar:

- Falta de planificación y organización: La empresa no utiliza metodologías de trabajo, lo que puede generar problemas de planificación, organización y seguimiento del desarrollo de software.
- Dificultad para adaptarse a los cambios: La falta de flexibilidad en los procesos de desarrollo puede dificultar la adaptación a los cambios en las necesidades del cliente o del mercado.
- Baja calidad del software: La ausencia de pruebas y control de calidad puede generar un software con errores y problemas de rendimiento.
- Dificultad para la colaboración y la comunicación: La falta de metodologías de trabajo puede dificultar la colaboración y la comunicación entre los miembros del equipo de desarrollo.
- Baja productividad: La falta de eficiencia en los procesos de desarrollo puede afectar negativamente a la productividad del equipo.
- Falta de conocimientos en herramientas, técnicas de desarrollo e inteligencia artificial: La empresa no está familiarizada con las últimas

tecnologías y herramientas, lo que puede limitar su capacidad para desarrollar software moderno y eficiente.

Metodología Ágiles - Parcial 1

Ejercicios:

1. Identifique tres nuevas problemáticas de las enunciadas anteriormente
2. Para cada problemática debe indicar con cuales principios y prácticas de las metodologías se podría abordar.

Se sugiere hacer una tabla con las siguientes columnas:

- a. Problema
 - b. Principios
 - c. Prácticas
 - d. Metodología
3. Diseñe un tablero Kanban para que la empresa lo pueda comenzar a usar. Por cada sección del tablero (o grupos de secciones) del tablero describa qué aspecto y/o problemáticas está abordando.

Tiempo estimado de resolución: 2hs

Formato de entrega del ejercicio

El alumno lo puede entregar el examen en los siguientes formatos:

- a. en formato digital a través del repositorio en la carpeta examen/parcial
- b. en formato papel

1) Las 3 problemáticas nuevas que pude encontrar son:

- Seguridad insuficiente:

Una de las prioridades en el desarrollo de software tiene que ser la seguridad ya que una mala seguridad lleva a ignorar las vulnerabilidades teniendo así graves consecuencias como la pérdida o robo de datos.

- Falta de diseño escalable:

Si no se diseñan los sistemas actuales teniendo en cuenta la escalabilidad, podrían surgir problemas al aumentar la carga o el número de usuarios. Es importante considerar patrones de diseño escalables, como la distribución de carga y el uso de caché.

- Falta de pruebas sistemáticas:

La empresa carece de un enfoque estructurado para realizar pruebas de software. La ausencia de pruebas puede resultar en un software con errores y problemas de rendimiento.

2)

Problema	Principios	Prácticas	Metodologías
Falta de planificación y organización.	Desarrollo iterativo e incremental	Iteraciones, modelos y diagramas	Proceso unificado
Dificultad para adaptarse a los cambios	Reflexión, oportunidad, diversidad.	Whole team, Sit together, informative workspace	XP
Baja calidad de software	Calidad de software a través de pruebas continuas.	Refactoring	TDD
Dificultad para la colaboración y la comunicación.	Respetar a la gente, potenciar el equipo, optimizar el conjunto.	Stand-up meetings, visual management	Lean
Baja productividad.	Control de calidad en cada fase.	Verificación y validación.	Cascada con testing en cada etapa

Falta de conocimientos en herramientas.	Formación y documentación estructurada.	Fases detalladas, manual de usuario y técnico	Ciclo de vida del desarrollo en cascada
Seguridad ineficiente.	Seguridad desde el inicio	Análisis de riesgos, pruebas de seguridad	Proceso unificado
Falta de diseño escalable.	Diseño simple, mejora continua	simple design, pair programming, continuous integration.	XP
Falta de pruebas de software.	Pruebas continuas y sistemáticas.	Desarrollo guiado por pruebas, automatización de pruebas.	TDD

3)

Pendiente	En proceso	Pruebas	Desplegado	Corrección	Archivado
seguridad insuficiente. diseño escalable.	Falta de conocimientos en herramientas.	Falta de pruebas software baja calidad de software	Falta de planificación y organización. Dificultad para la colaboración y la comunicación..	baja productividad Dificultad para adaptarse a los cambios	

Políticas de paso de sección:

Pendiente: se encuentran las tareas que todavía no empezaron, las tareas se mueven a “En Progreso” cuando se asignan a un miembro del equipo.

En progreso: tareas que se están trabajando actualmente, se mueve a “Pruebas” cuando se completa el progreso.

Pruebas: tareas listas para poner a prueba, si las pruebas son exitosas se mueven a “Desplegado”, si no son exitosas se pasan a “Corrección”.

Desplegado: son tareas que están en producción, si surge algún problema se pasan las tareas a “Corrección”.

Corrección: tareas que requieren solución, si el problema no se encuentra solución se pasa a “Archivado”, si no se vuelve a la parte de “Pruebas”.

Seguridad Insuficiente y Diseño escalable (Pendiente):

Ya que requiere una planificación y organización grande, se encuentran en la sección “Pendiente” para su análisis y planificación.

Falta de Conocimientos en Herramientas (En Progreso):

Ya que siempre se puede incorporar nuevos conocimientos se está en proceso.

Falta de Pruebas Software (Pruebas):

Se encuentra en pruebas debido a la gran cantidad de pruebas unitarias que se deben de hacer.

Baja Calidad del Software (Pruebas):

Se encuentra en búsqueda y prueba de la calidad del software.

Falta de Planificación y Organización (Desplegado):

Se pudo realizar una planificación y organización gracias a las metodologías aplicadas.

Dificultad para Adaptarse a los Cambios (Corrección):

Debido a que la empresa estuvo trabajando hace más de 30 años de la misma manera y ahora estamos en proceso de cambiar las metodologías tenemos cosas que seguir corrigiendo.

Baja Productividad (Corrección):

La baja productividad se aborda en la sección “Corrección”. Si se identifican problemas que afectan la productividad, se toman medidas correctivas.