

Ejercicio Individual

sábado, 12 de noviembre de 2022 9:50

Tema	¿Cómo lo voy a usar? Cuándo? Dónde?
1. La triple restricción	Siempre buscar el balance de tiempo, alcance y costo. Cuando esté implementando código, ya que muchas veces busco que esté perfectamente bien hecho y organizado, y por lo tanto, probablemente no estoy balanceando el alcance y costo del fin del programa, sólo el tiempo que le estoy dedicando.
2. Escalera de Tuckman	Las distintas etapas normales del equipo. Lo usaría cuando me encuentro en la etapa del storming en cada equipo nuevo, al saber que es normal y natural pasar por etapas donde el equipo no se siente tan unido; y donde de hecho, para pasar al performing, uno tiene que pasar por storming y norming, donde aceptas nuevos puntos de vista como parte de la verdad.
3. Gestión de la configuración	Lo voy a usar en todo momento del desarrollo de software, pero en los momentos de inicio y cambios: muchas librerías, módulos o frameworks tienen versiones con bugs, y a veces tiendo a actualizarlas sin documentar cuál versión usaba antes. Con las actualizaciones, a veces los proyectos ya no funcionan y sin documentar CIs, se vuelve difícil tener un software confiable.
4. Project Work Management	Usualmente me gusta hacer diagramas de las funcionalidades del software que voy implementando. Entonces, el Management del Project Work lo voy a usar para seguir implementando estos diagramas de forma personal, para mejorar mi aprendizaje durante el desarrollo de software y así ser más rápida en el futuro cuando una funcionalidad parecida emerja de nuevo.
5. Juntas de trabajo	Lo voy a usar cuando en mi equipo de la Startup tenemos juntas, lo que es cada 2 semanas: estas juntas he notado que tienen poca agenda de decisiones a tomar, por lo que a veces tenemos ratos muertos donde cada quien busca algún problema que quiera tratar. Estas juntas mejorarían si existe la preparación de la agenda en estos temas

6. Negociaciones	Lo aplicaré en cualquier negociación de requerimientos de software, ya que los requerimientos son procesos de ajuste entre el usuario y el desarrollador. Buscaría no traspasar temas emocionales para evitar que se puedan dar discusiones fuera del fin común que busca el software, y saber que ante cualquier propuesta fuera de lugar, lo primero que debo hacer es no decir nada e irme. Se me hizo una buena práctica para no terminar en problemas grandes si dices algo ante amenazas o cosas del estilo.
7. Conversaciones auténticas	Lo buscaré implementar en mi día a día: buscar que en una conversación yo reviva experiencias a través de lo que la otra persona platica, para poder adentrarme en lo que habla y no buscar como decirle mi experiencia como contraste a la suya.
8. Gestión de riesgos	Los riesgos son más comunes de lo que yo pensaba. Lo aplicaré cuando tenga cualquier inicio de implementación grande, donde cualquier oportunidad podría también ser un riesgo. El hecho de simplemente anotarlos e identificarlos ayuda a que cuando se presenten, sea algo que yo ya había mínimo pensado.

