

La convergencia entre Gestión de Proyectos Y Usabilidad

Objetivo: Entender cómo integrar la usabilidad dentro de la administración de proyectos informáticos para minimizar el riesgo de rechazo del usuario y maximizar el ROI(Retorno de inversión).

I. Administración de Proyectos Informáticos: Más allá del Diagrama de Gantt

En semestres anteriores, aprendieron sobre la "Triple Restricción" (Tiempo, Costo, Alcance). A este nivel, debemos analizar la administración de proyectos desde una perspectiva **estratégica y de riesgos**.

1. Del modelo en Cascada a la Agilidad.

El modelo en cascada (Waterfall) es útil para la teoría, pero en sistemas de información modernos, la incertidumbre es alta.

- **Gestión de la Incertidumbre:** En un proyecto de software, rara vez conocemos el 100% de los requisitos al inicio.
- **Iteración:** Utilizamos marcos como **Scrum** o **Kanban**. Aquí, el rol del *Project Manager* (PM) se transforma a menudo en *Scrum Master* o *Product Owner*.



2. El concepto de “Deuda Técnica” y “Deuda de Gestión”

Como administradores, deben vigilar dos métricas invisibles:

- **Deuda Técnica:** El costo de elegir una solución fácil ahora en lugar de un mejor enfoque que tomaría más tiempo (ej. hardcodear variables).
- **Gold Plating (Bañado en oro):** El error común del desarrollador de agregar funcionalidades que el cliente no pidió, inflando el costo y el riesgo. **Como PMs, su trabajo es decir "No" a lo innecesario.**

3. Estimación y Métricas

A nivel directivo, no medimos en "horas de programación", medimos en:

- Velocity: Puntos de historia completados por sprint.
- Burn-down Charts: Visualización del trabajo restante.

II. Usabilidad: Ingeniería, no "Arte"

Muchos ingenieros cometen el error de pensar que la usabilidad es "hacer que se vea bonito" (UI). **Falso.** La usabilidad es un atributo de calidad medible y objetivo.

1. Definición Formal (ISO 9241-11)

La usabilidad se define por tres pilares que deben memorizar para su vida profesional:

1. **Eficacia:** ¿El usuario puede completar la tarea? (Si no puede, el sistema falla).
2. **Eficiencia:** ¿Cuántos recursos (tiempo, clics, esfuerzo mental) le toma hacerlo?
3. **Satisfacción:** ¿Qué tan cómodo se siente el usuario con el proceso?

2. Heurísticas de Nielsen

No necesitamos reinventar la rueda. Jakob Nielsen definió 10 principios generales. Para sistemas de información, destaco tres críticos:

- **Visibilidad del estado del sistema:** El sistema siempre debe informar al usuario qué está pasando (ej. una barra de carga, un mensaje de éxito).

- **Prevención de errores:** Es mejor un diseño que prevenga un problema a uno que dé un buen mensaje de error.
- **Reconocer antes que recordar:** Minimizar la carga de memoria del usuario.

2. El ROI de la Usabilidad

Si un Stakeholder les pregunta por qué invertir dinero en pruebas de usabilidad, la respuesta es financiera:

- Regla 1:10:100: Corregir un problema de usabilidad en diseño cuesta \$1. En desarrollo cuesta \$10. En producción (mantenimiento) cuesta \$100.

III. El Cruce: Gestión de Proyectos + Usabilidad

Aquí es donde fallan la mayoría de los proyectos de titulación y startups. El PM quiere velocidad, el UX Designer quiere perfección. ¿Cómo lo resolvemos?

1. La Paradoja de la Usabilidad en Agile

Scrum pide entregar software funcionando cada 2 semanas. La usabilidad requiere investigación y pruebas con usuarios.

- **Solución: Dual Track Agile.**

- Hay un *Track de Descubrimiento (Diseño/UX)* que va un sprint adelante validando prototipos.
- Hay un *Track de Entrega (Desarrollo)* que codifica lo que ya fue validado.

2. Pruebas de Usabilidad como Hito del Proyecto

No dejen las pruebas para el final (UAT - User Acceptance Testing).

- **Pruebas de Guerrilla:** En etapas tempranas, prueben con 5 usuarios. Nielsen demuestra que 5 usuarios encuentran el 85% de los problemas de usabilidad.
- Esto debe estar cronometrado en su **WBS (Work Breakdown Structure)**.

3. Métricas Unificadas

Para evaluar el éxito del proyecto, usen métricas compuestas:

- **SUS (System Usability Scale):** Un cuestionario estándar de 10 preguntas.
- **Time on Task:** Tiempo promedio para completar una tarea crítica.

Ejercicio de Análisis (Caso Práctico)

- Imaginen que son los **Líderes de Proyecto** para el nuevo portal de inscripciones de la universidad.
- **El conflicto:** El Rector (Cliente) exige que el sistema tenga una contraseña de 16 caracteres alfanuméricos con símbolos para "máxima seguridad". El equipo de UX advierte que esto causará que el 40% de los alumnos bloqueen su cuenta en la primera semana (baja Eficiencia y Satisfacción).
- **Pregunta para la clase:** Como gestores del proyecto, ¿qué decisión toman? A) Obedecer al Rector (es el cliente) y asumir el costo de soporte técnico. B) Ignorar al Rector e implementar una clave simple (riesgo de seguridad). C) Negociar una solución técnica alternativa.
- *Respuesta esperada (Criterio Ingenieril):* La opción C. Se busca un equilibrio. Implementar **SSO (Single Sign-On)** con el correo institucional o **Autenticación de Dos Factores (2FA)**. Así se cumple el requisito de seguridad del proyecto sin destruir la usabilidad del sistema.

Tarea para la próxima sesión

Para consolidar esto, quiero que seleccionen el proyecto de software que están desarrollando para su tesis/proyecto final y me traigan lo siguiente:

1. Su **Backlog** actual priorizado.
2. Un análisis de **una pantalla crítica** de su sistema aplicando 3 heurísticas de Nielsen.
3. Identificar dónde hay **fricción**: ¿Dónde el sistema es lento o confuso para el usuario?