

**Tecnicatura Universitaria en Programación
(TUP-2024)**

**Metodología en Sistemas I
Profesora Petrelli Nadia**

Trabajo Práctico N°2

Tema: "Los participantes en el juego de los sistemas"

Grupo 3

Integrantes:

- Díaz Medina Tomás
- Díaz Rossini Juan José
- Gallo Genaro
- Hinojosa Alexis
- Liacoplo Gerónimo Emiliano
- Navarro Victor Leandro



Consigna:

1. Mencione por lo menos un participante adicional con el que pudiera interactuar en un proyecto de desarrollo de sistemas.
2. Describa un proyecto en el cual el analista no tenga contacto directo con el verdadero usuario. **¿Cuáles son las ventajas y desventajas de esta situación? ¿Qué otros arreglos alternos pudieran haberse hecho?**
3. **¿Se le ocurre algún otro término que pueda usarse para el usuario, además de propietario o cliente?**
4. **¿Se le ocurre alguna situación donde el analista no debiera hablar con el usuario?**
5. **¿Qué ventajas y desventajas se tendrían al ser el usuario miembro de tiempo completo del equipo del proyecto de desarrollo del sistema? ¿Se le ocurre algún proyecto específico en el que sería particularmente positivo incluir a un usuario en el equipo?**
6. **¿Cuáles son las ventajas y desventajas de que el usuario sea administrador del equipo encargado del proyecto de desarrollo del sistema? ¿Se le ocurre algún proyecto específico donde fuera muy positivo tener de administrador del proyecto a un usuario?**
7. **¿Cuáles son las ventajas y desventajas de que el usuario desarrolle el sistema de información solo? ¿Se le ocurre algún proyecto donde fuera bueno que el usuario hiciera las veces de analista, diseñador, programador y administrador?**
8. **¿Cuánto tendría que saber el usuario de computadoras y software para poder participar en un equipo de proyecto durante la fase de análisis del sistema? ¿Cuánto tendría que saber de las herramientas y técnicas del análisis de sistemas?**
9. **¿Cuánto tendría que saber un usuario acerca de las computadoras y el software para poder administrar un equipo de proyecto de desarrollo de sistemas? ¿Cuánto necesitaría saber acerca del análisis de sistemas para ser buen administrador?**
10. **¿Cuánto debe saber un usuario de computadoras y software para poder llevar a cabo el solo un proyecto de desarrollo de sistemas? ¿Cuánto debería saber acerca del análisis de sistemas?**
11. **¿Qué precauciones especiales tomaría como analista de sistemas si no tuviera contacto directo con el usuario? ¿Cree que no serían suficientes las herramientas descritas en este libro?**
12. En la sección 3.1.2 se mencionan varias de las preocupaciones que pudiera tener el usuario operacional acerca de un sistema nuevo. Mencione las tres más probables. **¿Cree que estas preocupaciones son razonables o que solo reflejan la típica falta de familiaridad del usuario con las computadoras?**

- 13.** ¿Qué responsabilidad ética y moral tiene el analista con el usuario operacional si el primero está convencido de que no causará despidos, pero el usuario se preocupa por la posibilidad de que si los cause? (Véase también la pregunta 19)
- 14.** Describa el escenario en el que los usuarios operacionales pudieran ocasionar que un nuevo sistema no funcione. ¿Cree que su escena sea realista? ¿No podría el usuario supervisor simplemente ordenar que se use el sistema?
- 15.** ¿Cuándo cree que deban discutirse con los usuarios los asuntos relacionados con la interfaz humana? ¿Al comienzo del proyecto? ¿A finales de éste? ¿Cuál es la interacción indicada?
- 16.** ¿Cree que sea poco realista que los usuarios operativos tengan solo un panorama local del sistema en el que participan? ¿Cree que sea seguro para el analista dar por hecho esto? ¿Cree que esto sea positivo? ¿Debería tratar el analista de proporcionar un panorama global a los usuarios operativos?
- 17.** Dé un ejemplo del panorama físico de un sistema o de su implantación, que podría tener el usuario operativo. ¿Le encuentra algún problema a esto?
- 18.** ¿Qué debe hacer el analista si el usuario supervisor no le permite hablar directamente con los usuarios operativos? ¿Cómo puede el analista manejar esta situación?
- 19.** ¿Qué responsabilidad ética o moral tiene el analista con el usuario supervisor si los usuarios operativos le expresan su preocupación acerca de posibles despidos ocasionados por el nuevo sistema? (Vea la pregunta 13.)
- 20.** Dé un ejemplo de un sistema en el que el usuario supervisor pueda no estar familiarizado con la política detallada de negocios a la que se estén atendiendo los usuarios
- 21.** ¿Por qué los usuarios ejecutivos típicos normalmente no se interesan por el posible ahorro que representaría la reducción de personal, lo cual se hará posible con la puesta en práctica o la implantación del nuevo sistema?
- 22.** ¿Qué tanto se debe involucrar a los usuarios ejecutivos en el desarrollo de un nuevo sistema de información?
- 23.** ¿Qué opciones tiene el analista si el usuario no entiende los modelos abstractos en el documento?
- 24.** ¿Cómo debe hacerse cargo el analista del "nuevo presuntuoso" descrito en este capítulo? ¿Qué hacer si el usuario insiste en un hardware o software específico para el nuevo sistema?
- 25.** ¿Cuánta responsabilidad debe asumir el analista por la obtención del consentimiento de los usuarios? ¿Qué tal si el analista no logra hacerlo?

26. ¿Qué riesgos cree que enfrenta el analista provenientes de la administración, según se expuso en la sección 3.2? ¿Qué puede hacer el analista para minimizar estos riesgos?
27. ¿Qué debe hacer el analista si las metas y prioridades de la administración entran en conflicto con las de los usuarios?
28. ¿Cuándo cree que debería participar en el proyecto la gente de operaciones?
29. ¿Debe la misma persona (o el mismo grupo de personas) llevar a cabo tanto el análisis como el diseño (y la programación) del sistema? ¿Cuáles son las ventajas y desventajas?

Desarrollo

1) Un participante adicional con el que se pueda interactuar en un proyecto de desarrollo de sistemas:

En caso de que fuéramos el usuario, a quien se le va a otorgar/vender el producto por parte del equipo encargado del proyecto en cuestión. Consideramos bastante ideal que podamos interactuar con el **personal de control de calidad**, su función es asegurarse de que el sistema funcione correctamente antes de su implementación, buscando errores o mejoras en el desempeño. Esto es clave porque, aunque los programadores prueben su código, siempre es bueno que alguien más valide que todo esté en orden antes de que el sistema llegue a los usuarios finales. Así, si el usuario pudiese interactuar con el personal de calidad, podrían asegurarse de que el sistema sea el que el usuario pidió, quizás para no perder tiempo al tener que hacer el sistema nuevamente si el usuario presenta rechazo al sistema presentado, y aclarar las cosas con el mismo antes de, propiamente, finalizar el contrato.

2) Proyecto en el cual el analista no tenga contacto directo con el verdadero usuario:

Un caso típico ocurre cuando el analista interactúa con el cliente (por ejemplo, el dueño o gerente de una empresa) en lugar del usuario final (empleados, clientes de la empresa, etc.). Esto sucede en proyectos donde la empresa contrata el desarrollo de un sistema que no usará directamente, sino que será utilizado por terceros.

Ejemplo:

Una empresa financiera contrata un sistema de gestión de préstamos en línea. El analista solo habla con los directivos para entender los requisitos, pero los usuarios finales serán los clientes que soliciten préstamos a través de la plataforma.

Ventajas

Mayor rapidez en la toma de decisiones: El cliente suele tener más poder de decisión que un usuario final individual.

Menos diversidad de opiniones: Se evitan conflictos entre múltiples usuarios con necesidades distintas.

Claridad en los objetivos comerciales: El cliente define el sistema en función de los objetivos estratégicos de la empresa.

Desventajas

Falta de precisión en los requerimientos: El cliente puede no conocer a fondo las necesidades y problemas reales de los usuarios finales.

Riesgo de baja adopción del sistema: Si los usuarios finales no encuentran el sistema intuitivo o útil, podrían resistirse a usarlo.

Mayor posibilidad de cambios tardíos: Si los usuarios finales no son consultados desde el principio, pueden surgir problemas cuando el sistema ya está avanzado.

Alternativas o soluciones

- **Realizar encuestas o entrevistas con los usuarios finales:** Permite validar las necesidades reales antes del diseño final.
- **Prototipos y pruebas tempranas:** Incluir a los usuarios finales en la revisión del sistema antes del desarrollo completo.
- **Representantes de usuarios:** Asignar personas dentro de la empresa cliente que actúen como intermediarios y puedan dar retroalimentación desde la perspectiva de los usuarios finales.

3) Otros términos para "usuario", además de "propietario" y "cliente":

Otro término podría ser "consumidor", Si el sistema fuera para que el cliente de una empresa de comercio lo usase, el usuario sería el que quiere obtener productos de la empresa, por ejemplo.

También podría ser "Invitado" o "Huésped", A veces, en los sistemas se le otorga una cuenta anónima, que se llama "Invitado" en varios sistemas, en los cuales solo entras como si fueras un usuario temporal o no regular, ya que la gran mayoría de veces los cambios que realices en la cuenta de invitado, no se guardarán y serán solo locales, como por ejemplo el "Invitado" en una cuenta de Youtube. En plataformas Yanquis, en general y en algunas empresas de nuestro país, se usa también el término "Huésped" o "Guest", para referirse al mismo término del "invitado", un usuario "temporal"

O quizás "Beneficiario", cuando el usuario percibe un beneficio del sistema en cuestión, como el cobro de una beca, o el descuento de alguna cosa por ejemplo, en un hospital, el usuario podría percibir un descuento por ser un jubilado.

4) Situación donde el analista no debe hablar con el usuario:

Un caso típico es cuando una empresa solicita un sistema que será utilizado por sus clientes o empleados, pero prefiere que el analista solo hable con los directivos, gerentes o representantes de la empresa en lugar de los usuarios finales.

Ejemplo:

Una empresa de telecomunicaciones quiere un nuevo sistema de atención al cliente para gestionar reclamos y consultas. Los usuarios finales serán los clientes que llaman al servicio técnico, pero la empresa decide que el analista solo se comunique con los gerentes de servicio al cliente y no con los clientes directamente.

Motivos por los que el analista no debe hablar con el usuario

1. Confidencialidad o seguridad de la empresa

- Algunas empresas manejan información sensible y no quieren que los usuarios finales sean entrevistados directamente.

2. Falta de conocimientos técnicos del usuario final

- Los usuarios pueden no comprender la parte técnica del sistema y dar información poco útil.

3. Intereses estratégicos de la empresa

- La empresa puede querer que el sistema se diseñe según su visión y no según las opiniones de los usuarios.

4. Estructura jerárquica de la empresa

- En empresas grandes, las decisiones se toman en niveles superiores, y los empleados solo reciben el sistema final.

5. Cuando los usuarios aún no existen

- En sistemas completamente nuevos, aún no hay usuarios finales con los que hablar.

5) ¿Qué ventajas y desventajas hay de incluir al usuario como miembro de tiempo completo en el equipo del proyecto?:

Según Yourdon, el usuario no solo debería ser parte del equipo del proyecto, sino que, en algunos casos, debería ser un empleado del proyecto o incluso su **dueño**. En situaciones ideales, el usuario podría hasta desarrollar el sistema por sí mismo, asegurando que cubra sus necesidades al 100%.

Ventajas:

Mejor entendimiento de los requerimientos

El usuario puede explicar exactamente qué necesita y evitar malentendidos.

Corrección de problemas en etapas tempranas

Se pueden identificar y corregir errores desde el inicio sin esperar hasta la fase de pruebas.

Mayor aceptación del sistema final

Si el usuario está involucrado, es más probable que adopte el sistema sin resistencia y lo promueva entre otros usuarios.

Menos retrabajo y cambios inesperados

Al tener al usuario dentro del equipo, se reducen los ajustes de última hora porque el diseño se valida en tiempo real.

Desventajas:

Possible falta de conocimientos técnicos

Un usuario puede no entender bien el proceso de desarrollo, lo que puede causar problemas en la comunicación con los programadores.

Poca objetividad y enfoque limitado

Un usuario puede centrarse demasiado en su propia experiencia sin considerar otros casos de uso.

Tiempo y costos adicionales

Tener a un usuario en el equipo significa pagarle o compensarle de alguna forma, lo que puede aumentar costos y demorar el desarrollo.

¿Se le ocurre algún proyecto específico en el que sería particularmente positivo incluir a un usuario en el equipo?

La gran mayoría de veces los encargados de la creación del sistema no son expertos en todo, no lo saben todo, porque sino serían "Todólogos". Por eso mismo, puede ser muy positivo tener a un Usuario Experto en el ámbito o el área en el que será desarrollado el sistema.

Por ejemplo: Se tiene que crear un sistema para simular interacciones de partículas subatómicas. El problema es que los analistas y programadores no tienen suficiente conocimiento sobre el tema, y aunque hagan una investigación previa, podrían malinterpretar conceptos o no captar completamente las necesidades del laboratorio. Para esto el Usuario experto podría guiar las decisiones del analista, para poder comunicarles al Diseñador, para que a su vez el Diseñador le dé el modelo del sistema al programador... para continuar el proceso de manera correcta, sin que hubiera errores.

Otro ejemplo, Minecraft: Es un gran ejemplo de cómo los usuarios pueden volverse parte del desarrollo de un sistema. En varias ocasiones, Mojang (la empresa detrás de Minecraft) ha permitido que los jugadores den sus opiniones de qué contenido se añadirá al juego escuchando ideas y propuestas sin afectar el desarrollo del juego.

¿Cómo los jugadores se vuelven parte del proyecto?

Votaciones de la comunidad: Mojang ha organizado eventos donde los jugadores votan por nuevos mobs (criaturas). Un ejemplo fue la votación para elegir entre el Allay, el Copper Golem y el Glare, donde el Allay ganó y fue implementado en el juego.

Feedback en versiones beta: Los jugadores pueden probar versiones experimentales del juego y dar retroalimentación para mejorar mecánicas antes del lanzamiento oficial.

Mods y contenido personalizado: Minecraft permite que los usuarios creen y comparten modificaciones (mods), lo que ha influenciado el desarrollo oficial del juego.

6) Ventajas y desventajas de que el usuario sea el administrador del proyecto de desarrollo de sistemas:

Ventajas de que el usuario sea el administrador del proyecto:

1. Conoce bien el problema:

Al ser parte del área usuaria, entiende a fondo las necesidades reales del sistema.

2. Mejor comunicación con los usuarios finales:

Puede traducir los requerimientos con mayor precisión al equipo técnico.

3. Focalización en los objetivos del negocio:

Se enfoca más en que el sistema resuelva problemas reales, no solo en que esté técnicamente bien hecho.

4. Aumenta la aceptación del sistema:

Los demás usuarios confían más en un proyecto liderado "desde adentro".

Desventajas de que el usuario sea el administrador del proyecto:

1. Falta de experiencia técnica: Puede tener dificultades para entender o gestionar aspectos técnicos o metodológicos del proyecto.

2. Riesgo de parcialidad: Podría tomar decisiones que beneficien solo a su área o punto de vista, sin considerar al sistema como un todo.

3. Falta de experiencia en gestión de proyectos: Administrar tiempos, costos, recursos y calidad puede ser complejo si no se tiene formación en gestión.

4. Posibles conflictos con el equipo técnico: La falta de lenguaje común con analistas y desarrolladores puede generar malentendidos.
¿En qué tipo de proyecto sería ideal que el usuario sea el administrador?

Un ejemplo claro: Implementación de un sistema de gestión clínica en un hospital. Si el administrador del proyecto es un médico con conocimientos básicos de informática y gestión, será ideal porque: Conoce los procesos críticos (admisión, historia clínica, emergencias).

- Puede identificar qué es prioridad para los usuarios (personal médico y administrativo).
- Asegura que el sistema cumpla los requisitos legales y éticos del sector salud.

7) Ventajas y desventajas de que el usuario realice todo el proyecto de desarrollo de sistemas:

Ventajas de que el usuario desarrolle el sistema por sí solo:

- 1. Conocimiento completo del problema:** El usuario entiende a fondo las necesidades del negocio o área específica, así que puede diseñar algo muy ajustado a la realidad.
- 2. Desarrollo rápido de soluciones específicas:** Al no haber intermediarios, puede tomar decisiones y hacer cambios rápidamente.
- 3. Reducción de costos:** No se necesita contratar a un equipo técnico o de análisis, lo cual puede ser ideal para proyectos con poco presupuesto.
- 4. Mayor flexibilidad:** Puede adaptar el sistema sobre la marcha sin pasar por procesos formales.

Desventajas de que el usuario lo haga todo:

- 1. Limitado conocimiento técnico:** Puede que el usuario no tenga la preparación suficiente en programación, diseño de bases de datos o seguridad.
- 2. Falta de buenas prácticas:** Al no tener formación profesional en desarrollo de software, el sistema puede ser difícil de mantener, escalar o integrar.
- 3. Sobrecarga de trabajo:** Una sola persona encargándose de todo puede saturarse y cometer errores importantes.
- 4. Poca visión a largo plazo:** El usuario puede enfocarse en resolver un problema inmediato, sin considerar aspectos futuros o la escalabilidad del sistema.
- 5. Riesgo de dependencia:** Si solo esa persona entiende el sistema, puede haber un gran problema si se va o no puede continuar.

¿En qué proyecto sería útil que el usuario lo hiciera todo?: Un buen ejemplo sería una pequeña hoja de cálculo o aplicación para control de inventario en un pequeño negocio familiar.

El dueño sabe exactamente qué productos tiene, qué necesita controlar y qué reportes quiere

8) ¿Cuánto tendría que saber el usuario de computadoras y software para poder participar en un equipo de proyecto durante la fase de análisis del sistema?

Para participar durante la fase de análisis del sistema, el nivel de conocimiento que un usuario debe tener sobre computadoras y software, así como sobre las herramientas y técnicas de análisis de sistemas, varía significativamente según el rol específico que desempeñe dentro del equipo y la complejidad del proyecto.

Sin embargo, podríamos establecer algunos lineamientos generales: Conocimiento del Usuario sobre Computadoras y Software:

* **Mínimo (Usuario Clave/Representante del Negocio):**

* **Uso básico de computadoras:** Debe ser capaz de utilizar una computadora para tareas cotidianas como navegar por internet, usar correo electrónico y, fundamentalmente, utilizar el software o los sistemas existentes relevantes para el proyecto.

* **Comprendión general del software:** Debe entender el propósito y la funcionalidad del software que utiliza en su trabajo diario. No necesita ser un experto técnico, pero sí comprender cómo el software apoya sus procesos de negocio.

* **Vocabulario básico:** Estar familiarizado con algunos términos informáticos comunes (archivos, carpetas, guardar, imprimir, etc.).

¿Cuánto tendría que saber de las herramientas y técnicas del análisis de sistemas?

En un ideal, el usuario debería saber:

Además de lo anterior, tener una comprensión más profunda de los flujos de trabajo digitales de su área.

* **Poder identificar problemas y oportunidades de mejora** en los sistemas actuales.

* **Ser capaz de comunicar sus necesidades y requerimientos** de manera clara, aunque no necesariamente en lenguaje técnico.

Conocimiento básico del propósito del análisis: Entender que la fase de análisis busca comprender el problema, las necesidades y los requerimientos del nuevo sistema.

Disposición a participar en las técnicas: Estar dispuesto a participar en entrevistas, reuniones, talleres y otras técnicas de recolección de información.

9) ¿Cuánto tendría que saber un usuario acerca de las computadoras y el software para poder administrar un equipo de proyecto de desarrollo de sistemas?

Un usuario que aspire a administrar un equipo de proyecto de desarrollo de sistemas no necesita ser un experto técnico en computadoras y software, pero sí necesita un sólido entendimiento conceptual y práctico de varios aspectos clave. Su rol principal será la gestión, la comunicación y la toma de decisiones estratégicas, por lo que el conocimiento técnico debe estar enfocado en permitirle realizar estas tareas de manera efectiva.

¿Cuánto necesitaría saber acerca del análisis de sistemas para ser buen administrador?

- Componentes Básicos de un Sistema Informático:** Entender la diferencia entre hardware (CPU, memoria, disco duro, etc.) y software (sistema operativo, aplicaciones).
- **Sistemas Operativos:** Conocer los conceptos básicos de cómo funcionan los sistemas operativos (Windows, macOS, Linux) y su rol en la gestión de recursos.
 - **Tipos de Software:** Distinguir entre software de sistema, software de aplicación y software de desarrollo.
 - **Conceptos de Redes:** Entender los fundamentos de las redes informáticas (LAN, WAN, internet), protocolos básicos (TCP/IP), y conceptos de seguridad (firewalls, antivirus).
 - **Almacenamiento de Datos:** Conocer diferentes formas de almacenamiento (local, en la nube), conceptos básicos de bases de datos (relacionales, no relacionales) y la importancia de la integridad y seguridad de los datos.

10) ¿Cuánto debe saber un usuario de computadoras y software para poder

llevar a cabo el solo un proyecto de desarrollo de sistemas?

Para que un usuario pueda llevar a cabo por sí solo un proyecto de desarrollo de sistemas, necesita tener conocimientos bastante amplios en varias áreas.

1. Conocimientos de computadoras y software

Un usuario que quiera llevar a cabo un proyecto completo debe saber bastante más que un "usuario común". En los tiempos actuales, un usuario que tenga como objetivo llevar a cabo por si solo un proyecto de desarrollo de sistemas, necesita saber sobre:

- **Sistemas operativos:** Entender cómo funcionan, manejar archivos, procesos y redes básicas.
- **Lenguajes de programación:** Ser competente en al menos un o dos lenguajes de alto nivel modernos (por ejemplo, Python, Java, JavaScript).
- **Bases de datos:** Saber cómo diseñar, implementar y consultar bases de datos (SQL).
- **Entornos de desarrollo (IDEs):** Usar herramientas como Visual Studio Code, Eclipse, etc.
- **Control de versiones:** Manipulación básica de Git y GitHub para trabajo organizado.
- **Testing y debugging:** Aprender a probar el sistema, encontrar errores y solucionarlos.
- **Despliegue:** Conocimientos para poner en producción el sistema (servidores, nube, seguridad básica).

2. Conocimientos de análisis de sistemas

En este apartado se necesita una comprensión clara del ciclo de vida del desarrollo de sistemas y metodologías de análisis. De manera ideal, debería conocer:

- **Definir requerimientos:** Aprender a entrevistar usuarios y documentar sus necesidades.

- **Modelado de sistemas:** Utilizar herramientas como diagramas de flujo, diagramas UML, DFDs, etc.
- **Diseño del sistema:** ¿Cómo estructurar lógicamente la solución antes de programarla.
- **Planeación y gestión del proyecto:** ¿Conocer estimar tiempos, descomponer tareas, y seguirle el rabo?
- **Evaluación de viabilidad:** Saber si el proyecto es técnicamente, económico y operativamente viable.

¿Cuánto debería saber acerca del análisis de sistemas?

El usuario debe ser de nivel alto en programación y de nivel medio a alto en análisis de sistemas. Es casi equivalente a ser un desarrollador full-stack pero con habilidades en ingeniería de software.

11) ¿Qué precauciones especiales tomaría como analista de sistemas si no tuviera contacto directo con el usuario? ¿Cree que serían suficientes las herramientas descritas en este libro?

Siempre que sea posible el analista debería establecer contacto directo con el usuario: Es importante mantener reuniones cara a cara con la persona que utilizará y recibirá el sistema. Si el usuario forma parte o es miembro activo del equipo del proyecto suele ser aún mejor, en algunos casos los usuarios podrían desempeñar el rol de gerente del proyecto.

Documentación rigurosa: Si no se puede tener un contacto directo, la documentación generada por el analista se vuelve aún más importante. Es esencial usar estas herramientas de modelado formales y rigurosas que describen el comportamiento del sistema, para evitar malentendidos costosos.

Las herramientas descritas por el libro son suficientes para la falta de comunicación directa con el usuario debido a que estas herramientas son esenciales para modelar y documentar el sistema de manera precisa y no reemplazan la interacción con el usuario.

12) Preocupaciones que pudiera tener el usuario operacional acerca de un sistema nuevo:

Tal y como se menciona en este punto, se mencionan varias preocupaciones que podría tener el usuario operacional acerca de un sistema nuevo, de los cuales, las 3 preocupaciones más probables serían:

1 • ¿Es como el teclado de la máquina de escribir que he estado usando por años?: Esta preocupación refleja la típica falta de familiaridad del usuario con las computadoras, ya que estas están en constante evolución y avance. Por lo que, los usuarios en general, deben adaptarse rápidamente al ritmo de los avances tecnológicos

2 • ¿Tendré que volver a teclear todo?: En este caso, hay varios factores en juego dependiendo de la situación. Ya que el usuario podría estar tecleando un trabajo en un documento que no haya activado la opción de autoguardado, o que no disponga de dicha opción en el peor casos.

Suponiendo que se dá el 2do caso, el usuario deberá guardar manualmente el trabajo que haya realizado hasta cierto punto, y todo este progreso podría verse afectado si se corta la energía (en caso de trabajar en una computadora), perdiendo todo el trabajo realizado hasta ese punto, o que

se quede sin batería el dispositivo donde está trabajando. Y si el usuario está trabajando en con un dispositivo con batería, podría tomar como precaución estar al pendiente de la batería del dispositivo donde está trabajando para evitar perder su trabajo cuando este se quede sin batería

3 • ¿Que tal si quiero "borrar" algo que teclee hace un momento?: Este también es otro ejemplo de la típica falta de familiaridad del usuario con las computadoras. Todos los programas están diseñados para borrar con la misma tecla empleada tanto en computadora como en otros dispositivos, y en caso de equivocarse, algunos programas cuentan con las opciones "deshacer" y "rehacer" para agregar o eliminar los cambios/ediciones/palabras y/o demás cambios que se realizaron durante el desarrollo del proyecto

¿A qué se quiere llegar con todo esto?

Existen usuarios que están familiarizados con la tecnología, siendo mayormente el caso de los jóvenes, quienes nacieron en una época moderna con "tecnología avanzada"

También están los usuarios que se vieron "forzados" a adaptarse al rápido avance de la tecnología, y esto se nota en los adultos y ancianos. Cuya parte de ambos grupos pudieron familiarizarse con estos avances tecnológicos, mientras que otra parte quedó estancada en el tiempo, a tal punto donde algunos prefieren el lápiz y papel antes que un trabajo digital.

Aquel grupo que quedó estancado en el pasado, son personas que pueden llegar a representar la falta de familiaridad del usuario con las computadoras, incluso negándose a incluir estos cambios en sus trabajos. Mientras que otros, a pesar de las dificultades que presentan estos cambios, terminan adaptándose por distintas circunstancias que se presentan en base a varios factores que pueden afectar sus trabajos.

La conclusión de todo esto, es que tarde o temprano, los usuarios deberán incluir estos "avances tecnológicos" en sus vidas laborales, y en algunos casos, llegarán a incluirlas en sus vidas cotidianas a medida que la tecnología evolucione hasta convertir el mundo en una era digital

13) ¿Qué responsabilidad ética y moral tiene el analista con el usuario operacional si el primero está convencido de que no causará despidos, pero el usuario se preocupa por la posibilidad de que sí los cause?

El analista tiene una responsabilidad ética y moral hacia el usuario operacional, incluso si está convencido de que el nuevo sistema no causará despidos. Si el usuario expresa preocupación por esa posibilidad, el analista no debe ignorarla ni minimizarla, sino abordarla con respeto, empatía y comunicación clara.

Aunque desde un punto de vista contractual el usuario sólo debería preocuparse por cumplir con el uso del sistema que se le provee, el aspecto humano del desarrollo no puede ser omitido, como menciona Yourdon en el capítulo 3. Si el usuario siente que el sistema lo hará innecesario, puede resistirse al cambio, afectar el clima

laboral o incluso sabotear el proyecto, aunque el sistema técnicamente funcione.

Por eso, parte de la ética profesional del analista incluye informar, tranquilizar y educar al usuario sobre cómo cambiará su trabajo con el nuevo sistema, buscando integrar a los usuarios en el proceso para que se sientan parte del cambio y no víctimas del mismo.

En definitiva, aunque el analista no tenga control sobre decisiones de despido (que son responsabilidad de la empresa), sí puede asumir una posición humana y ética, entendiendo el impacto psicológico que el sistema puede tener en quienes lo usarán.

14) Escenario en el que los usuarios operacionales pudieran ocasionar que un nuevo sistema no funcione:

Un escenario en el que los usuarios operacionales pueden ocasionar que un nuevo sistema no funcione ocurre cuando estos resisten activamente o pasivamente su implementación. Esto puede manifestarse de varias formas:

Negarse a usar el sistema, continuando con los métodos anteriores. Usarlo de forma incorrecta o mínima, afectando su eficacia.

Proporcionar datos incompletos o erróneos, ya sea por falta de capacitación, miedo, desconfianza o simplemente desinterés.

Boicotear sutilmente el proceso, ralentizando su adopción o destacando constantemente sus fallas.

Este escenario es totalmente realista, y ha ocurrido en múltiples implementaciones tecnológicas. Los sistemas, por más bien diseñados que estén, dependen de la aceptación y cooperación del usuario final para funcionar correctamente. Si no hay compromiso, el sistema fracasa en la práctica, aunque técnicamente esté bien construido.

En cuanto a la idea de que el usuario supervisor podría simplemente ordenar que se use el sistema, en teoría es cierto: puede emitir la orden. Sin embargo, en la práctica, la imposición no garantiza el uso efectivo. Si no se trabaja previamente en la aceptación, participación, capacitación y motivación de los usuarios operacionales, el sistema puede convertirse en una carga más que en una herramienta útil.

Además, obligar sin considerar las inquietudes y necesidades del personal puede generar conflictos, baja moral y pérdida de productividad.

15) ¿Cuándo cree que deban discutirse con los usuarios los asuntos relacionados con la interfaz humana? ¿Al comienzo del proyecto? ¿A finales de éste?

Consideramos que las discusiones relacionadas con el desarrollo de la interfaz humana deben llevarse a cabo a lo largo de todo el proceso de desarrollo del sistema. Esto permite obtener una mejor retroalimentación por parte del usuario, quien participa de forma activa en el proyecto, y además contribuye a lograr una mayor satisfacción con el producto final.

¿Cuál es la interacción indicada?

Este proceso puede abordarse en las siguientes etapas:

1. Inicio del desarrollo del sistema:

En esta fase se realizan las primeras charlas con el usuario para definir la funcionalidad principal del sistema y establecer un croquis inicial de la interfaz humana.

2. Desarrollo del sistema:

A lo largo de esta etapa se mantienen reuniones constantes con el usuario con el objetivo de alcanzar el sistema deseado. En cada encuentro se discuten las funcionalidades, y mediante prototipos se busca definir y mejorar la interfaz humana. Además, se orienta al usuario desde una perspectiva técnica para tomar decisiones informadas.

3. Revisión final del sistema:

Tras múltiples iteraciones y revisiones, se lleva a cabo una reunión final en la que se presenta el sistema completo, verificando si cumple con todos los requerimientos planteados por el usuario.

16) ¿Cree que sea poco realista que los usuarios operativos tengan solo un panorama local del sistema en el que participan?

No creemos que sea poco realista asumir que los usuarios operativos tienen solo un panorama local del sistema en el que participan. De hecho, como se menciona en el capítulo del libro, esto es bastante común: los usuarios que están en contacto directo con la operación del sistema suelen centrarse únicamente en su propio rol o en el pequeño fragmento del sistema que manejan a diario. No tienen por qué conocer todos los detalles del funcionamiento general, ni tampoco se les exige eso.

¿Cree que sea seguro para el analista dar por hecho esto?

dar por hecho esto como analista podría ser riesgoso. Porque puede llevar a subestimar el conocimiento o experiencia que sí tienen algunos usuarios sobre otros procesos, o incluso a dejar de lado información valiosa que podrían aportar. A veces, los usuarios operativos tienen observaciones clave sobre cómo su parte del sistema interactúa (o choca) con otras áreas, aunque no lo vean desde una perspectiva global.

¿Cree que esto sea positivo?

Si... y no, Tener una visión local puede ayudar a los usuarios a especializarse y volverse más eficientes en sus tareas. Pero si queremos lograr una buena integración del sistema o mejorar procesos, darles al menos una idea del panorama general puede ser muy útil. Así entienden mejor por qué se les pide cierto dato, por qué un cambio afecta a otra parte del sistema, o incluso cómo su trabajo impacta en el rendimiento general.

¿Debería tratar el analista de proporcionar un panorama global a los usuarios operativos?

El analista sí debería tratar de proporcionar un panorama más global a los usuarios operativos. No hace falta abrumarlos con toda la teoría de sistemas, pero mostrarles cómo encajan dentro del sistema completo puede motivarlos, hacerlos sentir parte del proceso y facilitar la colaboración con otras áreas.

17) Un ejemplo del panorama físico de un sistema que podría tener el usuario operativo:

Por ejemplo el de una persona que trabaja en el área de facturación de una empresa. Esa persona puede ver su computadora, el software de facturación, la impresora con la que imprime las boletas, y tal vez alguna máquina registradora. Para él o ella, el sistema es eso y nada más, porque es lo que usa todos los días.

¿Le encuentra algún problema a esto?

El problema es que no contempla todo lo que hay detrás: como los servidores donde se guardan los datos, la red interna, el soporte técnico, las bases de datos compartidas con otras áreas, o incluso los procesos automatizados que se ejecutan sin que los vea. Si llega a fallar algo (como el software), el usuario puede frustrarse o no saber a quién acudir.

Entonces, aunque es normal que tengan una visión local, puede ser un problema si esa visión les impide comprender el impacto de su trabajo en el resto del sistema, o si no saben cómo reaccionar ante ciertos errores o cambios.

18) ¿Qué debe hacer el analista si el usuario supervisor no le permite hablar directamente con los usuarios operativos? ¿Cómo puede el analista manejar esta situación?

En caso de que el analista no pueda entrevistar a los usuarios operacionales, consideramos que debería manejar este inconveniente siguiendo ciertos pasos.

Primero, conocer el/los motivos por los que el usuario supervisor no permite que los usuarios operacionales participen de forma directa.

Segundo, hacerle saber al usuario supervisor que la opinión y la participación de los usuarios operacionales es clave para desarrollar un sistema que, además de cumplir con los requisitos técnicos, logre satisfacer las necesidades de los usuarios.

Tercero, el analista debería optar por realizar reuniones grupales en las que estén presentes tanto el usuario supervisor como los usuarios operacionales, de forma que estos últimos puedan opinar y formar parte del proyecto, facilitando el levantamiento de requerimientos al analista.

En resumen, ante esta situación el analista debe comprender el porqué de las restricciones, y buscar alternativas para que los usuarios operacionales puedan opinar y formar parte del proyecto.

19) ¿Qué responsabilidad ética o moral tiene el analista con el usuario supervisor si los usuarios operativos le expresan su preocupación acerca de posibles despidos ocasionados por el nuevo sistema?

En esta delicada situación, el analista de sistemas tiene una responsabilidad ética y moral tanto con el usuario supervisor y los usuarios operativos con sus preocupaciones .

Estas responsabilidades pueden incluso entrar en tensión y requerir una cuidadosa consideración.

El analista tiene la responsabilidad de escuchar y tomar en serio las preocupaciones de los usuarios operativos. Estas preocupaciones son válidas y reflejan un temor real por su futuro laboral. Ignorarlas o minimizarlas sería una falta de respeto y empatía.

Si bien el analista puede no tener información completa o la autoridad para confirmar o negar los despidos, tiene la responsabilidad de ser honesto sobre lo que sabe y lo que no sabe.

20) Un ejemplo de un sistema en el que el usuario supervisor podría no estar familiarizado con la política detallada de negocios:

Un sistema de gestión de atención al cliente, dentro de una empresa grande.

Supongamos que el usuario supervisor ha sido recientemente promovido a ese rol desde un área técnica o administrativa, y es su primera experiencia supervisando usuarios operativos. En esta situación, es muy posible que aún no conozca en detalle todas las políticas de interacción con el cliente, los protocolos internos, o las excepciones que los operadores manejan a diario, ya que su enfoque ha cambiado de ejecutar tareas a supervisar procesos.

Este cambio puede generarle un gran nivel de estrés y presión, ya que no solo debe aprender nuevas responsabilidades, sino también actuar como un puente entre los usuarios operacionales y los ejecutivos, muchas veces sin tener aún una visión clara de las políticas que se aplican en el día a día.

21) ¿Por qué los usuarios ejecutivos típicos normalmente no se interesan por el posible ahorro que representaría la reducción de personal, lo cual se hará posible con la puesta en práctica o la implantación del nuevo sistema?

Creemos que, aunque reducir personal podría generar un ahorro, no siempre es lo primero que buscan los usuarios ejecutivos. En general, ellos están más enfocados en mejorar el rendimiento general de la empresa, hacer que todo funcione mejor o crecer a largo plazo. Además, despedir gente también trae consecuencias: puede afectar el ambiente laboral, la motivación del resto del equipo o incluso la imagen de la empresa.

También puede pasar que, por una cuestión ética o moral, no les parezca correcto reducir personal solo por ahorrar. Entonces, aunque es un dato que tienen en cuenta, no suele ser lo que más les interesa cuando piensan en un nuevo sistema.

22) ¿Qué tanto se debe involucrar a los usuarios ejecutivos en el desarrollo de un nuevo sistema de información?

Creemos que los usuarios ejecutivos deberían involucrarse lo suficiente como para dar el empujón inicial al proyecto, aportar su visión general y estar al tanto del progreso. Es importante que acompañen el proceso, no solo para que vean cómo se va formando el sistema, sino también para que sigan alineados con los objetivos del negocio.

Ahora, tampoco deberían meterse demasiado en lo técnico o en las decisiones del día a día, porque para eso están los analistas y los usuarios que operan el sistema. Si se involucran demasiado, podrían terminar frenando el trabajo o desviándolo hacia sus propias ideas, que tal vez no son lo más útil para quienes realmente van a usar el sistema. Lo ideal es que estén cerca, pero sin controlar todo. Como quien guía, apoya y da feedback cuando es necesario.

23) ¿Qué opciones tiene el analista si el usuario no entiende los modelos abstractos en el documento?

Si el usuario no entiende los modelos abstractos del documento (como los diagramas de flujo, modelos de datos o estructuras que usamos para representar el sistema), el analista tiene que buscar la forma de traducir esa información a algo que el usuario pueda entender sin problemas:

- **Una opción es usar ejemplos prácticos del trabajo diario del usuario**, para que pueda relacionar lo que ve en el modelo con lo que hace todos los días.
- **También puede simplificar los diagramas**, hacer esquemas más visuales o incluso usar palabras más comunes en vez de tecnicismos.
- **Otra posibilidad es sentarse a explicar cara a cara lo que representan esos modelos**, o incluso armar una presentación o demostración con casos reales.

Si el usuario aún no comprende, puede correr el riesgo de que el sistema no funcione para lo que realmente se necesita si se pasa este problema por alto. En todo caso, el analista debe tener buena comunicación así como un buen diseño de sus trabajos para poder guiar a aquellos que no son capaces de comprenderlo

24) ¿Cómo debe hacerse cargo el analista del "nuevo presuntuoso" descrito en este capítulo?

Según Yourdon, el "nuevo presuntuoso" es el tipo de usuario que, tras tener un contacto reciente con la tecnología, se siente experto cuando en realidad no lo es. Por lo general, ha tomado un curso, ha leído sobre informática o ha usado un software, y con eso intenta influir en decisiones técnicas del desarrollo del sistema, sin tener la capacitación necesaria.

El analista debe tratar esta situación de manera respetuosa y profesional. No se trata de despreciar al usuario, sino de escuchar sus ideas y canalizar su entusiasmo de manera positiva. Lo primero

es dejar claro quién hace qué: el usuario puede compartir su experiencia, pero las decisiones técnicas son del analista, quien debe evaluarlas en función de viabilidad, rendimiento y costos.

¿Qué hacer si el usuario insiste en un hardware o software específico para el nuevo sistema?

Si el usuario insiste en usar un hardware o software específico, el analista tiene que pedir razones concretas y hacer un análisis técnico que compare esa opción con otras. Si la opción no es adecuada, debe explicar por qué y ofrecer alternativas. Si el usuario o cliente decide seguir adelante con una opción no recomendada, el analista debería documentar su opinión técnica para dejar constancia de los riesgos.

25) ¿Cuánta responsabilidad debe asumir el analista por la obtención del consentimiento de los usuarios?

El analista de sistemas juega un papel clave en conseguir el visto bueno de los usuarios. No solo se trata de crear un sistema técnico, sino también de asegurarse de que la gente esté lista y dispuesta a usarlo. Para obtener ese consentimiento, los usuarios deben entender por qué se está implementando el sistema, qué beneficios traerá y cómo afectará su día a día.

Desde un punto de vista ético, el analista tiene que comunicar de manera clara cómo el nuevo sistema impactará la organización y los trabajos de todos. También es importante incluir a los usuarios en el proceso, escuchar sus inquietudes y considerar sus comentarios mientras se analiza y diseña el sistema. Esto no solo mejora el sistema, sino que también ayuda a construir confianza y a que la gente acepte el cambio más fácilmente.

¿Qué tal si el analista no logra hacerlo?

Si el analista no logra que los usuarios den su consentimiento, pueden surgir problemas como resistencia, uso ineficiente, desinterés, o incluso sabotaje, como no usar el sistema o manipular los datos. En estos casos, el sistema puede no funcionar bien, aunque técnicamente esté en orden.

Si las cosas van mal y no se consigue el consentimiento, el analista debe avisar al cliente o al equipo de dirección del proyecto, anotar los intentos hechos para involucrar a los usuarios y sugerir alternativas, como capacitaciones, demostraciones o apoyo de personas influyentes en la empresa.

26) ¿Qué riesgos cree que enfrenta el analista provenientes de la administración, según se expuso en la sección 3.2? ¿Qué puede hacer el analista para minimizar estos riesgos?

El analista enfrenta una serie de riesgos y desafíos particulares debido a su trayectoria y las posibles percepciones asociadas a ella. Estos riesgos pueden manifestarse en diferentes aspectos de su trabajo:

Percepción de Falta de Objetividad e Independencia, dificultad para ganar credibilidad, desafíos en la comunicación y la colaboración.

Para mitigar estos riesgos, el analista proveniente de la administración debe:

- **Demostrar activamente su objetividad e independencia a través de análisis rigurosos y basados en datos.**
- **Ser transparente sobre su trayectoria y reconocer posibles sesgos, mostrando disposición a considerar diferentes perspectivas.**
- **Invertir en el desarrollo de sus habilidades analíticas y metodológicas.**
- **Comunicarse de manera clara y efectiva con diferentes audiencias, adaptando su lenguaje y enfoque.**

27) ¿Qué debe hacer el analista si las metas y prioridades de la administración entran en conflicto con las de los usuarios?

Cuando los objetivos y prioridades de la administración entran en conflicto con los de los usuarios, el analista de sistemas está en una posición delicada, ya que debe actuar como mediador entre las dos partes. Eso no significa que tenga que elegir un lado; en cambio, es necesario encontrar un equilibrio que permita crear un sistema que sea útil, funcional y aceptado por ambos. En primer lugar, el analista debe comprender los intereses de ambas partes.

A menudo, la administración respalda la eficiencia, la reducción de costos, el control y los resultados estratégicos, mientras que los usuarios "realizan" sistemas se centran en la usabilidad, la flexibilidad, su capacidad para adaptarse a sus procesos diarios de trabajo y sus empleos. Por lo tanto, el analista necesita escuchar a cada uno, documentar los requisitos y luego negociar un compromiso que beneficie a todos.

Esto puede significar ofrecer una alternativa técnica que cumpla con los requisitos de la dirección sin dañar los sistemas usuarios, o proponer una implementación en etapas que faciliten la adopción. También es esencial ser honesto y transparente: el analista debe explicar a la administración las consecuencias de no seguir considerando a los usuarios. A veces, hay resistencia de los usuarios, errores o una disminución en la productividad que la administración no previó. Al mismo tiempo, los usuarios deben entender las prioridades de la administración y los resultados que buscan. Si el conflicto no se resuelve, el informe debe llevarse al comité del proyecto o la alta gerencia, y todos los argumentos deben ser respaldados por evidencia técnica, funcional y humana.

28) ¿Cuándo cree que debería participar en el proyecto la gente de operaciones?:

Desde nuestro punto de vista, deberían participar desde el principio. La gente de operaciones muchas veces es el usuario final del sistema, o al menos quienes lo van a estar utilizando todos los días. Entonces, ellos podrían decírnos qué funciona, qué no, o qué cosas podrían mejorarse.

Incluirlos desde las primeras etapas permite que el sistema realmente se adapte a sus necesidades. Además, si no se sienten escuchados, es más probable que después no se adapten bien al cambio o incluso rechacen el sistema por completo como dijo Yourdon.

Incluso si no quieren o no pueden participar de forma directa todo el tiempo, podrían hacerlo en momentos clave: pruebas, prototipos, sesiones de feedback o hasta simplemente reportando problemas.

La participación de los usuarios operativos debería ser constante o al menos estratégica, porque el éxito del sistema depende mucho de que ellos lo acepten y lo entiendan.

29) ¿Debe la misma persona (o el mismo grupo de personas) llevar a cabo tanto el análisis como el diseño (y la programación) del sistema? ¿Cuáles son las ventajas y desventajas?

La decisión de que una persona o un equipo se encarguen del análisis, diseño y de la programación de un sistema dependen de ciertos factores, tales como la complejidad del proyecto y las habilidades que pueda tener el equipo.

Ventajas:

- **Coherencia en la visión del proyecto:** Un equipo que trabaja unido puede mantener una comprensión clara de los objetivos y requisitos del sistema, lo que puede ayudar a reducir malentendidos entre las diferentes etapas.
- **Comunicación más eficiente:** No se necesita transferir información entre distintos equipos, lo que minimiza las pérdidas de datos y acelera el proceso de desarrollo.
- **Mayor flexibilidad en la toma de decisiones:** Un equipo que maneja todas las fases se puede adaptar más rápido a los cambios o a nuevos requerimientos, ajustando el diseño y la implementación mientras sea necesario.

Desventajas:

- **Limitaciones en la especialización:** Es posible que el equipo no tenga el nivel de experiencia necesario en todas estas áreas como en análisis, diseño y programación, lo que podría afectar la calidad del producto final.
- **Sobrecarga de responsabilidades:** Asumir múltiples roles puede provocar una carga excesiva de trabajo, aumentando el riesgo de cometer errores y disminuyendo la eficiencia.
- **Falta de perspectivas externas:** La ausencia de revisiones por parte de terceros puede llevar a omitir ciertos errores o a no considerar las alternativas más eficientes e innovadoras.