

Los participantes en el juego:

Como analista de sistemas, tendremos que trabajar en proyectos de desarrollos en los cuales nos podemos topar con diferentes participantes.

Usuario: Es el participante mas importante, puesto que el sistema se crea para él. Se lo tendrá que entrevistar con gran detalle dado que este nos dará los detalles que harán de nuestro proyecto un éxito o un fracaso.

Normalmente los Usuarios no se refieren a si mismos como usuarios sino como clientes o dueño. El cliente como tal, siempre tiene la razón y además, es el que a fin de cuentas paga.

Es fácil identificar al usuario, puesto que es el que formalmente solicita un sistema.

Aunque no siempre es tan fácil identificarlo y tener contacto con el, es de suma importancia tener un contacto **directo** con el. Y de no ser posible, la documentación generada por el analista cobra mayor importancia.

Normalmente cuando uno piensa en los usuarios presupone que son “todos iguales” cuando en verdad la realidad es muy diferente. Se pueden hacer las siguientes clasificaciones.

Por Categoría de trabajo:

Operacionales: Son operadores que tendrán contacto diario con el sistema. Tiende a ponerse en contra de los cambios. Hay que tener en cuenta 3 cosas cuando se trate con estos usuarios.

- Se preocupan mucho por las funciones del sistema, se preocupan aun mas por la internas humana. Este detalle y otros que puedan surgir de la entrevista con el usuario son vitales para el proyecto.
- Solo conocen su trabajo especifico y no entienden del panorama global.
- El analista debe usar un lenguaje que le sea familiar al usuario para luego traducirlo a un modelo esencial.

Supervisores: Son empleados supervisores que manejan grupos de usuarios operacionales. A tener en cuenta:

- Muchos son usuarios operacionales promovidos, por tanto estarán de acuerdo con sus subordinados.
- El supervisor tiende a ver al nuevo proyecto como una forma de reducir el numero de usuarios operacionales. Esto no es ni bueno ni malo pero puede dar a discusiones en donde el analista quede involucrado.
- Tiende a tratar de ser un nexo entre el operacional y el analista. Esto es peligroso, porque probablemente no refleje exactamente lo que el operacional busca. Tratar de evitarlo es esencial.
- Su visión es tan local como la del operacional.
- Será con el que mas contacto se tenga durante el desarrollo del proyecto.

Ejecutivo: Son los que por lo general no se involucran directamente con el proyecto pudiendo estar dos o tres niveles arriba del supervisor. A tener en cuenta:

- Es probable que solo sirva como autoridad para financiar.
- Por lo general no fueron empleados operacionales, por tanto no tiene los conocimientos necesarios para poder definir los requerimientos del sistema para aquellos que trabajan con el sistema todos los días.
- No están interesados por asuntos operacionales sino en los detalles estratégicos y las ganancias/perdidas a largo plazo.
- Se interesan por el panorama global y no por los detalles
- Están acostumbrados a trabajar con modelos abstractos.

Por nivel de experiencia:

Amateurs: Aquel que jamás ha visto una computadora. Es todo un reto para el analista dado que presenta un problema a la hora de la comunicaciones y hay que tratar de hablar en los términos que le sean mas familiar al usuario.

Novato Presuntuoso: Es un usuario que alega saber exactamente lo que quiere haga el sistema dado sus conocimientos en informática. Tiende a hacer sugerencias sobre la tecnología a usar para el desarrollo.

Verdaderos expertos: Usuarios que realmente entiende el analisis de sistemas y tambien las tecnologías de computadora.

Administración: La reilación que se tiene con estos administradores tiene que ver con la asignación de recursos: personas, tiempo y dinero. Es nuestra tarea como analistas documentar todos estos datos. Existen diversos tipos:

Administrador usuario: Están a cargo de varias personas en el área operacional donde se va a implementar el nuevo sistema.

Administrador de informática: son las personas encargadas del proyecto en si de sistemas y los administradores de nivel superior.

Administrador general: Son los administradores de nivel superior que no están directamente involucrados con la organización de informática ni son de la organización usuaria.

Puntos a tener en cuenta:

- Cuanto mas alto el nivel que ocupen es mas probable que desconozcan sobre tecnologías de computadoras.
- Las metas entre administradores y usuarios pudieran entrar en conflicto.
- Pudiera ser que los administradores no estén dando los recursos necesarios para la implementación efectiva del nuevo sistema.

- Pudiera ser también que entre los administradores tengamos creyentes y escépticos en la implementación es un nuevo sistema.
- Pudiera también que los administradores se vieran forzados a reducirle los recursos al proyecto.

Audidores: Según sea el tamaño del proyecto pudiera haber auditores, personal de control de calidad, etc. Su objetivo es asegurar que el sistema se desarrolle de acuerdo a diversos estándares o normas externos al proyecto. A tener en cuenta:

- A menudo no se involucran hasta el final en el proyecto, y a esas alturas es dócil hacer cambios.
- Es importante que el sistema sea compatible con viejas notaciones o formatos antiguos.
- Se interesan mas por la forma y no por el contenido.

Analista de sistemas: Personaje clave puesto que cumple varios papeles:

Arqueólogo y escribano: Descubre detalles y documenta la política de un negocio.

Innovador: Con sus conocimientos de informática debe proponer soluciones novedosas y mas útiles a problemas del usuario.

Mediador: entre los otros participantes de este juego que comúnmente están en desacuerdo. Y busca un consenso.

Jefe de proyecto: suele asignársele la administración del proyecto integra

Esto significa que se requiere facilidad en el manejo de personas para poder entrevistar a los usuarios y mediar entre desacuerdos. Conocer los usos potenciales del hardware y el software, tener una mente lógica y organizada. Ser capaz de pensar en el sistema en términos abstractos y físicos.

Diseñador de sistemas: es quien recibe los datos del trabajo del analista y desarrollar un diseño arquitectónico de alto nivel que servirá de base para el trabajo de los programadores.

En muchos casos el diseñador y el analista son la misma persona. Y de no ser así es importante que se mantengan en contacto a lo largo de todo el proyecto.

Programadores: Se busca no tener contacto con los programadores y que ellos solo tengan contacto con el diseñador (puesto que ellos le dan el fruto de su trabajo a los programadores). Esto quiere decir que la labor del analista termina mucho antes de que el programador escriba nada. Si hay contacto entre analistas y programador puede ser por lo siguiente:

- En pequeños proyectos el analista, diseñador y programador son la misma persona.
- El analista a veces sirve de administrador del proyecto.
- Por lo general es el programador el que descubre ambigüedades en la propuesta de requisitos entregada por el analista.
- En el caso de que se estuviera reemplazando un sistema muy viejo del cual no existiera documentación, la fuente principal de información

seria el programador encargado del mantenimiento del sistema hasta ese momento.

Personal de operaciones: Es aquel que opera el sistema, no porque use el sistema sino porque cuida u opera a un nivel mas bajo que el administrador.

Ciclo de vida de un proyecto:

En las organizaciones pequeñas el proyecto nace de conversaciones entre el usuario y el administrador de proyecto y el proyecto va desde el análisis hasta el diseño e implantación sin mayor alboroto.

En las organizaciones grandes, la comunicaciones suele ser por escrito el proyecto pasara por la diversas fases antes de completarse.

Sirve para:

- Definir las tareas a llevarse a cabo en un proyecto de desarrollo.
- Lograr congruencia entre todos los proyectos de desarrollo de una organización.
- Para tener puntos de control y revisión administrativos de las decisiones sobre continuar o no con un proyecto.

El objetivo es:

- Para no perder de vista ni restar importancia a las diferentes etapas del desarrollo.
- Para simplificar el seguimiento del proyecto a los niveles mas altos de la administración.
- Para prever mediante los puntos de control posibles rastros o la falta de recursos. Y poder abordar los posibles problemas a tiempo.

El ciclo de vida de un proyecto no simplifica ni quita trabajo, simplemente es una herramienta que el administrador del proyecto para organizarse de mejor manera.

Ciclo de vida del proyecto clasico: Cada proyecto pasa por un tipo de análisis, diseño e implementación. Se caracteriza por tener una fuerte tendencia a la implantación ascendente del sistema y la instancia en la progresión lineal y secuencial de una fase a la siguiente.

Implantación Ascendente: Una de las grandes debilidades del ciclo de vida clásico.

Presenta un gran numero de dificultades:

- Nada esta hecho hasta que esta todo terminado: Es decir que si la fecha limite cae en la mitad del proyecto, no habrá nada que mostrar al usuario.
- Las fallas mas triviales se encuentran al comienzo y del periodo de prueba y las mas graves al final.
- La eliminación de fallas suele ser extremadamente difícil durante las ultimas etapas de prueba del sistema.

- La necesidad de prueba con la computadora aumenta exponencialmente durante las etapas finales de prueba.

Progresión secuencial: Segunda debilidad insistencia en que las fases se sucedan secuencialmente. Tendemos a tratar de desprendernos de una etapa. El problema que esto no permite el tratamiento de problemas reales como los relacionados con el personal, la política de la compañía o la economía. Otro problema que puede surgir es el cambio de padecer por parte del cliente sobre lo que quiere que haga el sistema, sobre la marcha.

Ciclo de vida del proyecto semiestructurado: Dos detalles obvio respecto al clasico:

- Implantación de arriba hacia abajo, es decir que se codifica y prueban los módulos de alto nivel primero y luego los de bajo nivel.
- El diseño clasico se reemplaza por el diseño estructurado.

Se pondran en ejecución paralelamente parte de la codificacion y de las pruebas. Ademas del hecho de que puedan darce retroalimentación entre la codificacion, la prueba y la eliminación de las fallas.

Los diseñadores tenian poco contacto con el analista que escribia la esfecificacion y definitivamente no tenian contacto con el usuario.

Ciclo de vida del proyecto estructurado: Se divide en 9 actividades:

La encuesta: conocido también como el estudio de la factibilidad. Empieza cuando el usuario solicita que una o mas partes de su sistema se automatice.

Objetivos:

- Identificar a los usuarios solicitantes del pedido y crear un ámbito de trabajo con ellos.
- Identificar las deficiencias actuales en el ambiente del usuario como pueden ser el software del sistema actual no se puede mantener o el tiempo de respuesta del sistema telefónico es malo
- Establecer metas y objetivos para un nuevos sistema.
- Determinar si es factible automatizar el sistema y de ser así sugerir escenarios aceptables.
- Preparar el esquema de guía que se usara para el resto del proyecto

Si bien la encuesta solo toma entre el 5% y 10% del tiempo total del proyecto, es una actividad fundamental y verdaderamente importante.

Análisis de sistemas: Su función es transformar sus dos entradas (las políticas del usuario y el esquema de guía del proyecto) en una especificaciones estructurada. Esto implica el desarrollo de un modelo ambiental y el desarrollo de un modelo de comportamiento. Estos dos modelos se cominan para formar el modelo esencial, que representan una descripción formal de lo que el nuevo sistema debe hacer.

El diseño: partiendo de la especificaciones estructurada, se ocupa de transformar los modelos de datos de entidad-relación en un diseño de base de datos.

Implantación: Es el paso en donde los programadores se ponen a trabajar, incluye tanto programación estructurada como implantación descendente. El analista no se vera involucrado en esta actividad.

Generación de pruebas de aceptación: Generar los datos de prueba. Puede darse al mismo tiempo que las actividades de diseño e implantación, pudiera ser que al analista le sea asignada esta labor al término del desarrollo del modelo esencial.

Garantía de calidad: También conocida como prueba final o la prueba de aceptación. Requiere como entrada los datos de la prueba de aceptación generada en la actividad anterior y el sistema integrado producido en la implantación. El analista puede estar involucrado pero por lo general no lo esta. Se busca que el sistema tenga un nivel apropiado de calidad.

Descripción del procedimiento: Generación de una descripción formal de las partes del sistema que se harán en forma manual, lo mismo que la descripción de cómo interactuaran lo usuarios con la parte automatizada del nuevo sistema. El resultado de este paso es el “manual para el usuario”. El Analista podría verse involucrado.

Conversión de Base de Datos: Por lo común, esta actividad requiere como entrada la base de datos actual del usuario.

Instalación: La entrada se basa en el manual del usuario, la base de datos convertida, y el sistema aceptado producido por la garantía de calidad.

Implantación Radical VS Implantación Descendente:

La vida del proyecto estructurado permite que más de una actividad se lleve a cabo a la vez. Es decir que ambos extremos se podrían detallar así:

Implantación radical: Las actividades de la 1 a la 9 se llevan a cabo paralelamente desde el principio del proyecto.

Implantación Descendente: es un enfoque mas conservador la actividad N completa se termina antes de comenzar con la actividad N + 1.

Obviamente son extremos, en medio existen infinidad de matices que nos dan infinitos numero de opciones y posibilidades para implementar en un proyecto.

El problema surge cuando uno tiene que decidir cual va a ser su enfoque.

Existen varios casos a tener en cuenta.

En primer lugar esta el Usuario, si constamos con un usuario altamente indeciso o poco experimentado lo mejor será optar por un enfoque mas radical y poder cumplir con los requisitos lo antes posible antes de que el usuario cambie de opinión o pretenda cambiar la función del sistema. En cambio si contamos con un usuario que sabe exactamente lo

que quiere y tiene experiencia en el ámbito, la mejor opción será trabajar con un enfoque mas conservador.

Otro factor importante es el de la presión, si el equipo desarrollador cuenta con una fecha limite en donde forzosamente debe entregar el sistema lo mejor sera un enfoque radical para poder llegar a cumplir con esa fecha en la forma mas exacta posible. En cambio si no contamos con una fecha próxima/tiránica (tener en cuenta que todos los proyectos cuentan con una fecha de entrega) se puede trabajar mas holgadamente con un enfoque mas conservador.

También a tener en cuenta esta el hecho de la presentación de estimaciones, programas y presupuestos. De tratarse es un emprendimiento menor en el cual estas presentaciones son “mera formalidades” se deberá trabajar con es de suponer con un enfoque mas radical de manera de terminar lo antes posible con el trabajo. En cambio si el nivel de detalle y presentaciones es elevado esto será directamente proporcional a la conservador que deberá ser el enfoque.

Por ultimo el administrador del proyecto debe considerar la posibilidad de cometer un error tecnico importante. Siempre existe la posibilidad de describir un error a ultimo momento.

En resumen, de presentarse un proyecto chico en el cual nadie esta muy seguro de lo que quiere o de cómo hacerlo, la solución seria realizar un proyectó cuyo ciclo de vida tenga implantación lo mas radical posible. En cambio si se trata de un proyecto grande, con mucha inversión de capitales, la opción mas acertada seria hacer un ciclo de vida del proyecto estructurado con la implementación lo mas descendente posible.

Ciclo de vida de Prototipo: variación del ciclo de vida estructurado descendente.

La principal diferencia es que el enfoque descendente construye un modelo en papel completo del sistema, en cambio el de prototipo crea un modelo de lo que seria el sistema finalizado con la intención de ser un prueba y muestra para el usuario. Que el lo vea, y decida si es lo que quiero o no y su cumple con los queriditos pedidos por el.

Es especial para aquellos casos en que:

- El usuario no puede o no esta dispuesto a examinar modelos abstractos en papel.
- El usuario no puede o no esta dispuesto a decirnos los requisitos y solo se puede determinar mediante tanteo o ensayos de prueba y error.
- Cuando se tiene la intención de que el sistema sea en línea y con operaciones total por pantalla.
- El sistema no requiere de grandes especificaciones de detalles.