



INGENIERÍA DE SOFTWARE II

PRINCIPIOS, CONCEPTOS, MÉTODOS Y HERRAMIENTAS



21 DE MAYO DE 2017

ANDREA CARTUCHE JARA

Contenido

1.	PRINCIPIOS FUNDAMENTALES	1
•	PRINCIPIOS QUE GUIAN EL PROCESO.-.....	1
•	PRINCIPIOS QUE GUIAN LA PRÁCTICA.-.....	1
•	PRINCIPIOS QUE GUIAN TODA ACTIVIDAD ESTRUCTURAL.....	2
•	PRINCIPIOS DE COMUNICACIÓN.- Un cliente tiene un problema que parece abordable mediante una solución basada en computadora. La comunicación efectiva entre colegas técnicos, con el cliente y otros participantes, y con los gerentes del proyecto se encuentran entre las actividades más difíciles que deben enfrentarse.	2
	FORO REALIZADO POR LOS ALUMNOS	3
2.	INGENIERÍA DE REQUERIMIENTOS O DE REQUISITOS	5
1.	Concepción:.....	5
2.	Indagación:.....	5
a.	Problemas de Alcance:.....	5
b.	Problemas de entendimiento:	5
c.	Problemas de volatilidad:.....	5
3.	Elaboración:.....	5
4.	Negociación:.....	5
5.	Especificación:	5
6.	Validación:.....	6
7.	Administración de los requerimientos:	6
2.1	ESTABLECER LAS BASES	6
2.1.1	IDENTIFICACIÓN DE LOS PARTICIPANTES:	6
2.1.2	RECONOCER LOS MÚLTIPLES PUNTOS DE VISTA:.....	6
2.1.3	TRABAJAR HACIA LA COLABORACIÓN:.....	6
2.1.4	HACER LAS PRIMERAS PREGUNTAS:	6
2.2	INDAGACIÓN DE LOS REQUERIMIENTOS.....	7
2.2.1	RECAPCIÓN DE LOS REQUERIMIENTOS EN FORMA COLECTIVA:	7
2.2.2	DESPLIEGUE DE LA FUNCIÓN DE CALIDAD.....	7
	Requerimientos normales.-.....	7
	Requerimientos esperados.	7
	Requerimientos emocionantes.-	8
2.2.3	ESCENARIOS DE USO	8
2.2.4	INDAGACIÓN DE LOS PRODUCTOS DEL TRABAJO	8
2.3	DESARROLLO DE CASOS DE USO	8
2.3.1	EJEMPLO DE CASOS DE USO DE UN INVENTARIO	10

INGENIERÍA DE SOFTWARE II

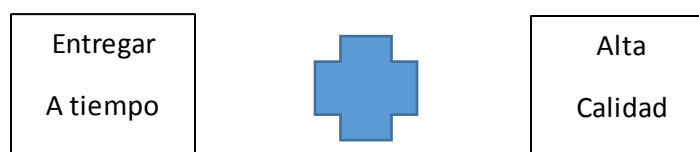
La práctica de la Ingeniería de Software es un conjunto de:

1. Principios
2. Conceptos
3. Métodos; y Herramientas que dan apoyo a la aplicación de los métodos.

1. PRINCIPIOS FUNDAMENTALES

La práctica de la Ingeniería de Software está guiada por un conjunto de principios fundamentales que ayudan en la aplicación del proceso de software significativo y en la ejecución de métodos eficaces de IS.

- **PRINCIPIOS QUE GUIAN EL PROCESO.-** Sin importar el modelo si es lineal o iterativo, prescriptivo o ágil, estos principios son aplicables a todos los modelos de proceso.
 1. **Ser Ágil.-** Poner énfasis en la economía de acción, mantener el enfoque técnico tan sencillo como sea posible y generar resultados sencillos.
 2. **En cada etapa, centrarse en la calidad.-** Calidad en el producto de trabajo (resultados) que se ha generado.
 3. **Estar listo para adaptar.-** Adapte su enfoque a las restricciones impuestas por el problema, la gente y el proyecto en sí.
 4. **Formar un equipo eficaz.-** El objetivo son las personas, forme un quipo donde haya confianza y respeto mutuo.
 5. **Establecer mecanismos para la comunicación y coordinación.-** Los proyectos suelen fallar porque los participantes no coordinan sus esfuerzos para crear un producto final exitoso.
 6. **Administrar el cambio.-** Mecanismos o formas de administrar la forma en que los cambios de solicitan, evalúan, aprueban e implementa.
 7. **Evaluar el riesgo.-** Tener un Plan de contingencia
 8. **Crear producto del trabajo (resultados) que agreguen valor para otros.-** Es una cadena de resultados que van a ser entregados para que sigan agregando su trabajo o desarrollo del mismo y a su vez pasará a otras manos que también añadirán su codificación etc.
- **PRINCIPIOS QUE GUIAN LA PRÁCTICA.-** Tiene un objetivo general: Entregar a tiempo software operativo de alta calidad.



1. **Divide y vencerás.-** Un problema grande es más fácil de resolver si se divide.
2. **Entender el uso de la abstracción.-** Abstracción (simplificación de algún elemento complejo) eliminar la necesidad de comunicar detalles; según Joel Spolsky, sin la comprensión de los detalles, no puede diagnosticarse con facilidad la causa de un problema.
3. **Buscar la coherencia.-** El principio de coherencia sugiere que un contexto familiar hace que el software sea más fácil de usar.
4. **Centrarse en la transferencia de información.-** ejm de una base de datos a un usuario final de un usuario final a una interfaz gráfica de usuario, la información fluye a través de una interfaz. Este principio implica que debe ponerse atención especial al análisis, diseño, construcción y prueba de las interfaces.
5. **Construir software que tenga modularidad eficaz.-** Separación de entidades, cualquier sistema complejo puede dividirse en módulos (componentes) y debe ser eficaz (cada módulo centrarse en un aspecto delimitado, cohesivo en su función) los módulos deben estar interconectados en forma relativamente sencilla.
6. **Buscar Patrones.-** Crear una bibliografía que ayude a los desarrolladores de software para resolver problemas recurrentes, los patrones ayudan para comunicar experiencias ya vividas y dar soluciones al caso.
7. **Cuando sea posible, representar el problema y su solución desde varias perspectivas diferentes.-** Un problema y su solución se estudian desde varias perspectivas.
8. **Tener en mente que alguien dará mantenimiento al software.-** El software será corregido en el largo del plazo, cuando se descubran sus defectos, se adapte a los cambios de su ambiente y para esto debe aplicarse una práctica sólida de IS (ingeniería de software).

- **PRINCIPIOS QUE GUIAN TODA ACTIVIDAD ESTRUCTURAL**

- **PRINCIPIOS DE COMUNICACIÓN.-** Un cliente tiene un problema que parece abordable mediante una solución basada en computadora. La comunicación efectiva entre colegas técnicos, con el cliente y otros participantes, y con los gerentes del proyecto se encuentran entre las actividades más difíciles que deben enfrentarse.

1. **Escuchar.-** Si algo no está claro pregunte para aclararlo, pero evite las interrupciones constantes, si una persona habla, nunca parezca beligerante (giros de los ojos, movimientos de la cabeza) en sus palabras o actos.
2. **Antes de comunicarse prepararse.-** Haga algunas investigaciones para entender el vocabulario propio del negocio, prepare una agenda antes de la reunión.
3. **Alguien debe facilitar la actividad.-** Toda reunión de comunicación debe tener un líder (facilitador) que:
 - a. **Mantenga la conversación en movimiento hacia una dirección positiva.**
 - b. **Sea un mediador en cualquier conflicto que ocurra**
 - c. **Garantice que se sigan otros principios.**

4. **Es mejor la comunicación cara a cara.-** Es mejor cuando está presente alguna otra representación de la información relevante, ejm. Dibujos, bosquejos, documentos en que se centre la discusión.
5. **Tomar notas y documentar las decisiones.-** Alguien que participe en la reunión debe de servir como secretario y escribir los temas y decisiones importantes.
6. **Perseguir la colaboración.-** Aceptar la colaboración y llegar a un consenso en equipo para describir las características del producto o sistema, cada colaboración sirve para generar confianza entre los miembros del equipo.
7. **Permanecer centrado; hacer módulos con la discusión.-** De haber un facilitador para formar módulos de conversación y abandonar el tema solo cuando se haya concluido con el mismo.
8. **Si algo no está claro hacer un dibujo.-** Con frecuencia un dibujo o bosquejo da la claridad cuando las palabras no alcanzan.
9. **a) Una vez que se acuerde algo avanzar. b) Si no es posible ponerse de acuerdo en algo, avanzar c) Si una característica o función no está clara o no puede aclararse en el momento, avanzar.-** La comunicación como cualquier otra actividad requiere tiempo, en vez de hacer iteraciones sin fin, las personas que participan deben reconocer que hay muchos temas que requieren análisis (véase el principio 2) a veces “avanzar” es la mejor forma de tener agilidad en la comunicación.
10. **La negociación no es un concurso o un juego. Funciona mejor cuando las dos partes ganan.-** Negociar características, funciones, tiempo de entrega, etc. Si el equipo ha colaborado bien, todo tendrán un objetivo en común, aún la negociación demandará del compromiso de todas las partes.

FORO REALIZADO POR LOS ALUMNOS

DE ESTOS PRINCIPIOS FUNDAMENTALES QUE GUÍAN EL PROCESO Y QUE GUÍAN LA PRÁCTICA, DESCRIBIR CUÁL A SU CRITERIO ES EL MÁS IMPORTANTE DE LOS PRINCIPIOS QUE GUÍAN EL PROCESO Y QUE GUÍAN LA PRÁCTICA, ADEMÁS ENCUENTRE LA SIMILITUD ENTRE LOS PRINCIPIOS DE CADA ETAPA (PROCESO Y PRÁCTICA).

Postado por Roger Blacio mar, junio 13, 2017 16:02:39

Según mi criterio personal el principio que guían el proceso que más me parece importante llevar a cabo es el Formar un equipo Eficaz porque del personal que este en el proyecto a cargo se llevaran buenos resultados en el desarrollo del software. Y El principio que guían la práctica me parece importante tener en mente que alguien dará mantenimiento al software ya que si lo descuidamos y no le damos mantenimiento nuestro trabajo será incompleto y traería problemas con el cliente. La similitud de estos principios es que cada uno nos enseña a desarrollar, ser ágil, buscar soluciones, simplificar los problemas, en el desarrollo del software.

Postado por Joao Alexander Sanchez mar, junio 13, 2017 11:44:29

El principio que más importancia tiene en la etapa que guían el proceso ,es Establecer mecanismos para la comunicación y coordinación, porque la comunicación fluida y rápida es vital para una buena organización entre superiores y empleados , también coordinar nuestros esfuerzos de manera espontánea para evitar fracasos en nuestros proyectos para tener éxito en el producto final. En mi opinión el Principio más importante en la etapa que guían la práctica, es Divide y vencerás, porque nos ahorra tiempo en resolver problemas grandes y complejos dividiéndolo en un conjunto de elementos (o entidades), así se puede desarrollar más claramente y rápido el producto final. La similitud de estos Principios de cada etapa (Proceso y Practica), es que son una guía que todos debemos seguir para crear un sistema o producto basado en computadora para llegar con éxito al destino y no tener ningún tipo de problemas o fracasos.

Postado por Keyla Za lun, junio 12, 2017 23:50:22

Dentro de los principios que guían el proceso, 'Formar un equipo eficaz', se lo podría considerar como importante ya que de nuestro grupo de trabajo dependerán los resultados del Software. Mientras que en los

principios que guían la práctica, 'Tener en mente que alguien dará mantenimiento al software' cuidar del software, darle un mantenimiento cada determinado tiempo podrá mejorar la utilización del mismo. La similitud que presentan estos principios, tanto como los que guían la práctica y como los que guían el proceso es, que ambos buscan entregar a tiempo considerado un software operativo de alta calidad, es decir, ambos buscan la calidad ante todo en sus modelos de software.

Postado por Lenín Blacio Díaz lun, junio 12, 2017 23:22:27

De acuerdo a mi consideraciones el principio que juega un papel importante es el de forma un equipo eficaz porque nos permite tomar medidas objetivas, evaluar el desempeño de los administradores con el fin de determinar componentes claves y eficaz para el desarrollo aplicaciones. Por otra parte, en los principios que guían el proceso, mi perspectiva a punta la construir software que tenga modularidad eficaz, porque para resolver un problema conviene separarlo en partes, que se pueden diseñar, desarrollar, probar y modificar, de manera sencilla y lo más independientemente de resto de la aplicación. La similitud radica que cada uno de ellos viene interrelacionado porque ambos buscan un mismo fin que sería la calidad y eficaz del sistema operativo, por ejemplo para el evaluar el riesgo se debe buscar patrones con el fin de reducir fallos técnicos y decepciones de clientes.

Postado por Leonardo Zhunaula lun, junio 12, 2017 22:46:53

En lo personal el principal Principio para el proceso es formar un equipo eficaz, hay que contar con personas indicadas para obtener los resultados esperados. Y por último el principal principio que guían la práctica es Divide y vencerás ya que si contamos con el personal adecuado podremos dividimos el trabajo así podríamos terminar más rápido y tendríamos un Software de calidad

Postado por Angie Sarmiento lun, junio 12, 2017 20:08:51

"Estar listo para adaptar", ya que todo software de adaptarse a cualquier ambiente de trabajo, tanto para los usuarios como administradores de sistema. La similitud que existe entre los principios del proceso y la práctica es centrarse en la calidad y construir un software que tenga modularidad eficaz ya que ambos se centran en obtener como resultado un producto rápido, eficaz y sin error alguno, satisfaciendo los requerimientos del cliente.

Postado por Jose Xavier Serrano lun, junio 12, 2017 00:25:22

Brindar un producto de muy buena calidad y que tenga coherencia para que sea más fácil al momento de usar..!

La similitud entre ambos es de entregar a tiempo Software Operativo de alta calidad que contengan características que satisfagan las necesidades de todos los participantes

Postado por Soraya Duran dom, junio 11, 2017 20:11:41

Para que el proceso sea eficaz se debería de tomar en consideración que se debe formar un equipo, dado que con este se trabajara de manera ordenada y plena para que el mismo tenga un buen resultado y en lo que respecta a principios que guían la práctica es buscar la coherencia dentro de la practica ya que si el programador utiliza palabras que sean fáciles de entender será más fácil para el usuario entender y manejar el programa.

La similitud que hay entre los principios que guían el proceso y los principios que guían la práctica es centrarse en la calidad y buscar la coherencia por lo que los dos hacen referencia a tener un buen programa con un buen resultado y que el usuario este satisfecho del servicio que le prestamos

Postado por Jonathan Bonilla Grijalva sáb, junio 10, 2017 23:55:39

"En cada etapa, centrarse en la calidad", es un principio que fundamenta o guía tanto al proceso y practica en la realización de Ingeniería de Software esto se debe a que en los quince principios restantes hay normas que se enfocan en ayudar a obtener la calidad deseada, por lo tanto no dejan de ser importantes.

2. INGENIERÍA DE REQUERIMIENTOS O DE REQUISITOS

Muchos desarrolladores quieren ir directo al desarrollo del sistema sin entender que es lo que se necesita, esto puede llevar el proyecto de software al fracaso. La ingeniería de requisitos proporciona el mecanismo apropiado para:

- Entender lo que desea el cliente
- Analizar las necesidades
- Evaluar la factibilidad
- Negociar una solución razonable
- Especificar la solución sin ambigüedades
- Validar la especificación
- Administrar los requerimientos a medida que se transformen en un sistema funcional.

Incluyen 7 tareas diferentes:

1. **Concepción:** La mayor parte de proyectos inicia cuando se identifica una necesidad del negocio o se descubre un nuevo mercado o servicio potencial. En la concepción del proyecto se establece el entendimiento básico del problema, las personas que quieren una solución, la naturaleza de la solución, la eficacia de la comunicación y colaboración preliminares entre los otros participantes y el equipo de software.
2. **Indagación:** Problemas que se encuentran cuando ocurre una indagación:
 - a. **Problemas de Alcance:** La frontera de los sistemas está mal definidas o los clientes o usuarios finales especifican detalles técnicos innecesarios que confunden.
 - b. **Problemas de entendimiento:** Los clientes o usuarios no están seguros de lo que necesitan, comprenden mal las capacidades y limitaciones de su ambiente de computación, omiten información que creen que es obvia, especifican requerimientos que están en conflicto con las necesidades de otros clientes o usuarios, solicitan requerimientos ambiguos.
 - c. **Problemas de volatilidad:** Los requerimientos cambian con el tiempo.
3. **Elaboración:** La información se refina durante la elaboración, creación y mejora de escenarios de usuario que describan como interactuará el usuario final y otros actores con el sistema. Análisis, clases, atributo.
4. **Negociación:** Que los clientes, usuarios y otros participantes ordenen sus requerimientos por orden de prioridad y que después analicen sus conflictos.
5. **Especificación:** Para la especificación desarrollarse una plantilla estándar, sin embargo algunas veces se debe ser flexible, para sistemas grandes se usa lenguaje natural con modelos gráficos, para sistemas pequeños se requiere escenarios de uso.

6. **Validación:** la validación de los requerimientos analiza la especificación a fin de garantizar que todos ellos se han enunciado sin ambigüedad, que se detectaron y se corrigieron las inconsistencias, las omisiones y los errores y que los productos del trabajo se presentan conforme a los estándares establecidos para el proceso, proyecto y producto. “EL MECANISMO PRINCIPAL DE VALIDACIÓN DE LOS REQUERIMIENTOS ES LA REVISIÓN TÉCNICA”.
7. **Administración de los requerimientos:** Es el conjunto de actividades que ayudan al equipo del proyecto a identificar, controlar y dar seguimiento a los requerimientos y a sus cambios en cualquier momento del desarrollo del proyecto.

2.1 ESTABLECER LAS BASES

Las bases que permiten entender los requerimientos de software a fin de que el proyecto comience en forma tal que se mantengan avanzando hacia una solución exitosa.

- 2.1.1 **IDENTIFICACIÓN DE LOS PARTICIPANTES:** Cualquier persona que se beneficie en forma directa o indirecta del sistema en desarrollo. Ejm. Gerente de operaciones de negocio, gerente de productos, personal de mercadotecnia, clientes internos y externos, usuarios finales, consultores, ingenieros de producto, ingenieros de software e ingenieros de apoyo y mantenimiento. Se debe hacer una lista a quien se le debe consultar y esta lista inicial crecerá cuando se haga contacto con los participantes porque surgirá la pregunta ¿A quién más piensa que se le debería consultar?
- 2.1.2 **RECONOCER LOS MÚLTIPLES PUNTOS DE VISTA:** Los requerimientos del sistema se explorarán desde distintos puntos de vista recolectar cada información de cada participante puede ser inconsistente entre sí o conflictiva, pero se debe clasificar y elegir la que tenga coherencia interna.
- 2.1.3 **TRABAJAR HACIA LA COLABORACIÓN:** El trabajo del ingeniero de requerimientos es identificar todas las áreas de interés común, en las que todos estén de acuerdo, y las de conflicto o incongruencia, requerimientos que desea un participante pero que estén en conflictos con las necesidades de otro, pero el encargado del proyecto toma la decisión final.
- 2.1.4 **HACER LAS PRIMERAS PREGUNTAS:** Las primeras preguntas deben estar “libres del contexto” deben centrarse en el cliente y en otros participantes, en las metas y beneficios generales. Por ejemplo tal vez se pregunte:
 - ¿Quién está detrás de la solicitud de este trabajo?
 - ¿Quién usará la solución?
 - ¿Cuál será el beneficio económico de una solución exitosa?
 - ¿Hay otro origen para la solución que se necesita?

Las preguntas siguientes permiten entender mejor el problema y hacen que el cliente exprese sus percepciones respecto de la solución:

- ¿Cuál sería una buena salida generada para una solución exitosa?
- ¿Qué problemas resolvería esta solución?
- ¿Puede mostrar o describir el ambiente de negocios en el que se usaría la solución?
- ¿Hay aspectos especiales del desempeño o restricciones que afecten el modo en el que se enfoque la solución?

Las preguntas finales se centran en la eficacia de la actividad de comunicación en sí. Gause y Weinberg las llaman “metapreguntas” y proponen la siguiente lista:

- ¿Es usted la persona indicada para resolver estas preguntas? ¿Sus respuestas son oficiales?
- ¿Mis preguntas son relevantes para el problema que se tiene?
- ¿Estoy haciendo demasiadas preguntas?
- ¿Puede otra persona dar información adicional?
- ¿Debería yo preguntarle algo más?

La sesión de preguntas y respuestas sólo se deben usar para el primer encuentro y luego ser reemplazadas por un formato de indagación de requerimientos que combine elementos de solución de problemas, negociación y especificación.

2.2 INDAGACIÓN DE LOS REQUERIMIENTOS

Combina los elementos de solución de problemas, elaboración, negociación y especificación.

2.2.1 RECAACIÓN DE LOS REQUERIMIENTOS EN FORMA COLECTIVA:

Lineamientos básicos:

- Tanto ingenieros de software como otros participantes dirigen o intervienen en las reuniones
- Se establecen reglas para la preparación y participación
- Se sugiere una agenda con suficiente formalidad para cubrir todos los puntos importantes, pero con la suficiente informalidad para que estimule el libre flujo de ideas.
- Un “facilitador” (cliente, desarrollador o participante externo) controla la reunión.
- Se utiliza un “mecanismo de definición” (que pueden ser hojas de trabajo, tablas sueltas, etiquetas adhesivas, pizarrón electrónico, grupos de conversación o foro virtual)

La meta es identificar el problema, proponer elementos de la solución, negociar distintos enfoques y especificar un conjunto preliminar de requerimientos de la solución en una atmosfera que favorezca el logro de la meta.

2.2.2 DESPLIEGUE DE LA FUNCIÓN DE CALIDAD

El despliegue de la función de calidad (DFC) es una técnica de administración de la calidad que traduce las necesidades del cliente en requerimientos técnicos para el software. Se concentra en MAXIMIZAR la satisfacción del cliente a partir del proceso de ingeniería de software, pone el énfasis en entender lo que resulta VALIOSO para el cliente y luego despliega dichos valores en todo el proceso de ingeniería. El DFC identifica 3 tipos de requerimientos:

Requerimientos normales.- Objetivos y metas que se establecen para un producto o sistema durante las reuniones con el cliente. Si estos requerimientos están presentes, el cliente queda satisfecho.

Requerimientos esperados.- Están implícitos en el producto o sistema y quizá sean tan importantes que el cliente no los mencione de manera explícita. Su ausencia causará mucha insatisfacción.

Requerimientos emocionales.- Estas características van más allá de las expectativas del cliente y son muy satisfactorias si están presente.

El DFC utiliza entrevistas con los clientes, observación, encuestas y estudios de datos históricos, después estos datos se llevan a una tabla de requerimientos, llamada TABLA DE LA VOZ DEL CLIENTE, que se revisa con el cliente y con otros participantes, luego se emplean varios diagramas, matrices y métodos de evaluación para extraer los requerimientos esperados y tratar de percibir los requerimientos emocionales.

2.2.3 ESCENARIOS DE USO

Se necesita entender cómo emplearán los usuarios finales las funciones y características del sistema, para lograr esto los desarrolladores y usuarios crean un conjunto de escenarios que identifican la naturaleza de los usos para el sistema que se va a construir, estos se denominan CASOS DE USO, proporcionan la descripción de la manera en la que se utilizará el sistema.

2.2.4 INDAGACIÓN DE LOS PRODUCTOS DEL TRABAJO

Para la mayoría de sistemas los productos del trabajo incluyen lo siguientes:

- Un enunciado de la necesidad y su factibilidad
- Un enunciado acotado del alcance del sistema o producto.
- Una lista de clientes, usuarios y otros participantes que intervienen en la indagación de los requerimientos.
- Una descripción del ambiente técnico del sistema.
- Una lista de requerimientos (de preferencia organizados por función) y las restricciones del dominio que se aplican a cada uno.
- Un conjunto de escenarios de uso que dan perspectiva al uso del sistema o producto en diferentes condiciones de operación.
- Cualquier prototipo desarrollado para definir requerimientos.

Cada uno de estos productos del trabajo es revisado por todas las personas que participan en la indagación de los requerimientos.

2.3 DESARROLLO DE CASOS DE USO

Un caso de uso capta un contrato que describe el comportamiento del sistema en distintas condiciones en las que el sistema responde a una petición de alguno de sus participantes.

En esencia, un caso de uso narra una historia estilizada sobre cómo interactúa un usuario final (que tiene cierto número de roles posibles) con el sistema en circunstancias específicas.

El primer paso para escribir un caso de uso es definir un conjunto de “actores” que estarán involucrados en la historia.

Los actores son las distintas personas (o dispositivos) que usan el sistema o producto en el contexto de la función y comportamiento que va a describirse. Los actores representan los papeles que desempeñan las personas (o dispositivos) cuando opera el sistema. Con una definición más formal, **un actor es cualquier cosa que se comunique con el sistema o producto y que sea externo a éste**. Todo actor tiene uno o más objetivos cuando utiliza el sistema.

Es importante notar **que un actor y un usuario final no necesariamente son lo mismo**. Un usuario normal puede tener varios papeles diferentes cuando usa el sistema, mientras que un actor representa una clase de entidades externas (gente, con frecuencia pero no siempre) que sólo tiene un papel en el contexto del caso de uso.

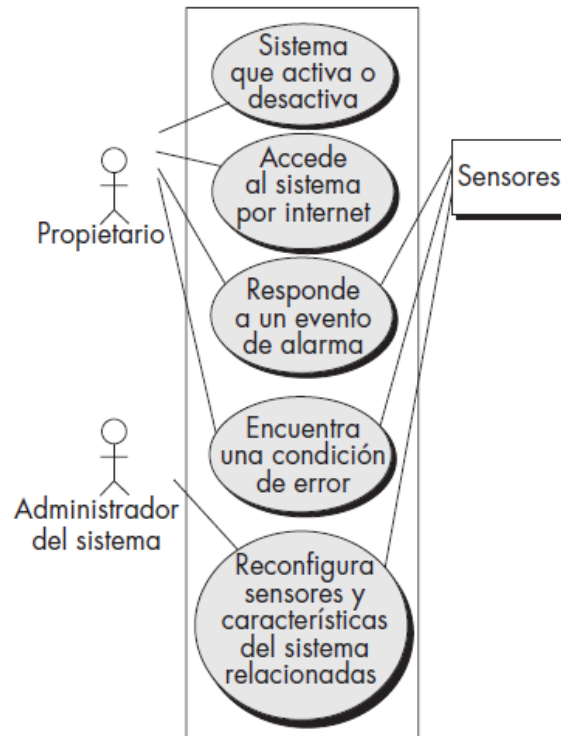
Una vez identificados los actores, es posible desarrollar casos de uso. Que debe responder un caso de uso:

- ¿Quién es el actor principal y quién(es) el(los) secundario(s)?
- ¿Cuáles son los objetivos de los actores?
- ¿Qué precondiciones deben existir antes de comenzar la historia?
- ¿Qué tareas o funciones principales son realizadas por el actor?
- ¿Qué excepciones deben considerarse al describir la historia?
- ¿Cuáles variaciones son posibles en la interacción del actor?
- ¿Qué información del sistema adquiere, produce o cambia el actor?
- ¿Tendrá que informar el actor al sistema acerca de cambios en el ambiente externo?
- ¿Qué información desea obtener el actor del sistema?
- ¿Quiere el actor ser informado sobre cambios inesperados?

Considerando la situación en la que el propietario de la casa usa el panel de control, a continuación se plantea el caso de uso básico para la activación del sistema

1. El propietario observa el panel de control de CasaSegura para determinar si el sistema está listo para recibir una entrada. Si el sistema no está listo, se muestra el mensaje no está listo en la pantalla de cristal líquido y el propietario debe cerrar físicamente ventanas o puertas de modo que desaparezca dicho mensaje [el mensaje no está listo implica que un sensor está abierto; por ejemplo, que una puerta o ventana está abierta].
2. El propietario usa el teclado para introducir una clave de cuatro dígitos. La clave se compara con la que guarda el sistema como válida. Si la clave es incorrecta, el panel de control emitirá un sonido una vez y se reiniciará para recibir una entrada adicional. Si la clave es correcta, el panel de control espera otras acciones.
3. El propietario selecciona y teclea permanecer o fuera para activar el sistema. La entrada permanecer activa sólo sensores perimetrales (se desactivan los sensores de detección de movimiento interior). La entrada fuera activa todos los sensores.
4. Cuando ocurre una activación, el propietario observa una luz roja de alarma.

Diagrama de caso de uso de una Casa Segura



CONCLUSIÓN.

El caso de uso es la interacción entre el actor y el sistema para así dar a conocer de como el usuarios o actores van interactuar en el sistema

2.3.1 EJEMPLO DE CASOS DE USO DE UN INVENTARIO

1. Identificación de los Actores

Nuestro sistema consta de los siguientes Actores:

Usuario SuperAdministrador
Usuarios Administradores
Usuarios Encargados o Auxiliares
Base de Datos

2. Identificación de Metas por Actor

Usuario SuperAdministrador:

- Administrar y velar por el buen manejo del sistema.
- Crea, Cerrar, Editar Cuentas de Inventario.
- Manejo directo de la Base de Datos General.
- Impresión de informe de movimientos.

Usuario Administrador

- Registrar, Ver, Editar, Borrar elementos de su Inventario.
- Crear, Editar, y Eliminar cuentas de Encargado.
- Autorizar préstamo de elementos.
- Visualización e impresión de informe de movimientos máximo 6 meses.

Usuario Encargado

- Ver Inventario autorizado para acceso.
- Ingresar el movimiento del inventario con previa autorización.
- Registrar y firmar entrega del elemento prestado del inventario.

3. Identificación de los Casos de Uso

SuperAdministrador

- Crear Cuenta de Administrador.
- Administrar Soporte Web.

Teniendo en cuenta lo anterior este actor está relacionado con el Sistema Web de la E3T.

Administrador

- Edición del Perfil de Usuario.
- Crea cuentas de Encargado.
- Registrar nuevos elementos de Inventario.
- Búsqueda de elementos de inventario en base a criterios específicos.
- Elimina elementos de Inventario.
- Editar Elementos de Inventario.
- Autoriza préstamos de elementos de Inventario.

Encargado

- Visualizar el Inventario en Búsqueda de un elemento en Particular.
- Registrar movimientos de elementos que han sido autorizados para préstamos.
- Firma su responsabilidad bajo el elemento que ha sido prestado.
- Cierra la entrega de un préstamo.

