

Universidad de las Ciencias Informáticas



Facultad 5 “Entornos Virtuales”

Título

Estrategia para realizar los procesos de validación y verificación en los proyectos de realidad virtual.

Trabajo de Diploma para optar por el título de
Ingeniero en Ciencias Informáticas

Autores

Leisy Reyes Martínez
Lenier Igarza Del Toro

Tutores

Ing. Jandrich Domínguez Fortún
Ing. Gerandys Hernández Casanova

Ciudad de la Habana, junio del 2008
“Año 50 de la Revolución”

El futuro tiene muchos nombres. Para los débiles es lo inalcanzable. Para los temerosos lo desconocido. Para los valientes es la oportunidad.

Victor Hugo

Declaramos que somos los únicos autores de este trabajo y autorizamos a la facultad 5 de la Universidad de las Ciencias Informáticas; así como a dicho centro para que hagan uso del mismo en su beneficio.

Para que así conste firmamos la presente a los ____ días del mes de _____ del año 2008.

Leisy Reyes Martínez

Autor

Lenier Igarza Del Toro

Autor

Ing. Jandrich Domínguez Fortún

Tutor

Ing. Gerandys Hernández Casanova

Tutor

Autores

Nombre: Leisy Reyes Martínez

Correo Electrónico: lrmartinez@estudiantes.uci.cu

Nombre: Lenier Igarza Del Toro

Correo Electrónico: ligarza@estudiantes.uci.cu

Tutores

Nombre: Ing. Jandrich Domínguez Fortún

Correo Electrónico: jandrich@uci.cu

Nombre: Ing. Gerandys Hernández Casanova

Correo Electrónico: ghermandez@uci.cu

De Leisy:

Resulta difícil ser breve a la hora de agradecer a todas las personas que han estado a mi lado durante estos 5 años de estudios. Es mi deseo que cada una de ellas encuentre en estas líneas mi más sincero agradecimiento y se sienta identificado.

No encuentro la palabra exacta para describir el agradecimiento hacia mis padres, a los que quiero y admiro con creces. Mami y Papi, gracias por su apoyo y sabios consejos. Mil gracias por darme la oportunidad de estar aquí, por animarme y hacerme ver el lado positivo cuando no lo veía y por guiarme hacia el camino correcto sin importar lo difícil que resulte. Gracias por ser ejemplo de padres, hijos y amigos.

A Mima, mi otra figura materna, gracias por el aroma de cariño y ternura que desprende. Y a ti mi hermanito, por ser un motivo más para seguir adelante y por los momentos de diabluras que disfrutamos juntos. A toda mi familia, mis abuelos, tías y primos, mil gracias por su constante preocupación y apoyo.

Quiero agradecer a todas las personas que siempre han estado al tanto de mis estudios, a los amigos que dejaban saludos a pesar de la distancia, a los que estuvieron a mi lado durante estos 5 años, con los cuales compartí días de mucho de estrés (sobre todo en los días de la PN) y felicidad, de risas y llanto, momentos preciados que serán inolvidables y dignos de recordar, pero en especial quiero agradecer a Lenier, mi compañero de tesis, por su constante esfuerzo y paciencia que incluso en los momentos en los que las ideas brillaban por su ausencia mantuvo vivo el sentido del humor.

Quiero agradecer a nuestros tutores y demás personas que de una forma u otra gracias a su colaboración y paciencia contribuyeron a la materialización de este sueño.

Gracias a todos!!! Con mucho afecto Leisy.

De Lenier:

Durante el desarrollo de este trabajo muchas han sido las personas que me han brindado su ayuda, experiencia, conocimientos y hasta parte de su tiempo, a todas ellas quisiera agradecer pues con su colaboración logramos que este trabajo pudiera salir adelante.

Muy en especial agradecer a mis padres que han sido mi fuente de inspiración durante los últimos 5 años en la universidad. A mi segunda mamá, Elsis, que con sus consejos oportunos me ha guiado y apoyado durante toda mi vida para que lograra concretar este sueño de convertirme en ingeniero. Agradecer a toda mi familia por confiar en mí y ayudarme cada vez que lo necesité en este largo camino de estudios.

No puedo continuar sin antes agradecer a Yisel por toda la ayuda que me dio durante todo este tiempo y por tenderme su mano cada vez que fue preciso. Gracias flaca.

Agradecer a mi compañera de tesis, Leisy, por su locura y sensatez al mismo tiempo. Por estar siempre presente a lo largo de toda la carrera, pero en especial por ser como es, sencilla y con un corazón enorme. Muchas gracias por todo.

Quisiera agradecer también a mis tutores por todos sus consejos, críticas y recomendaciones durante todo el período que duró el desarrollo de este trabajo.

Gracias a todos!!! Lenier

*A los mayores tesoros que puedo tener,
mi familia y amigos.*

Leisy

A toda mi familia y amigos.

A la memoria de Birilo.

Lenier

RESUMEN

La industria cubana del software está llamada a convertirse una de las principales fuentes de ingreso económico para el país, con este objetivo se han realizado una serie de acciones entre las cuales resalta la creación de la Universidad de las Ciencias Informáticas, dicha institución se encuentra vinculada activamente a la producción de software tanto para el consumo nacional como extranjero. Con el propósito de estructurar este proceso productivo en la universidad se tienen definidos un conjunto de polos productivos los cuales están asociados a cada una de sus diez facultades según los perfiles de estas.

La facultad 5 de la UCI tiene como segundos perfiles la automatización y la realidad virtual, actualmente el mayor peso de la producción recae sobre este último, pues 9 de los 15 proyectos con que cuenta la facultad pertenecen a este polo. Por estudios realizados durante este trabajo se demostró que de manera general en los proyectos de realidad virtual de esta facultad no se realizan de manera correcta los procesos de validación y verificación, lo que provoca una disminución de la calidad final de los productos de trabajo.

El presente trabajo provee a la facultad 5 de una estrategia basada en el modelo CMMI para realizar los procesos de validación y verificación en sus proyectos de realidad virtual.

La estrategia describe como desarrollar las actividades claves dentro de estos, entre las que podemos encontrar: preparar los procesos, para esto se define como seleccionar los productos o componentes de estos, como establecer un entorno para los procesos y por último propone un grupo de criterios y procedimientos a seguir; la otra de las actividades es realizar los procesos, para la cual se relacionan los pasos a seguir y se propone cuando y como analizar los resultados arrojados. Además de esto se adjuntan algunas planillas consideradas importantes para documentar la validación y la verificación.

Palabras claves: Validación, Verificación, CMMI.

ÍNDICE

INTRODUCCIÓN.....	1
CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA	5
INTRODUCCIÓN.....	5
1.1 PRINCIPALES CONCEPTOS	5
1.1.1 CALIDAD DE SOFTWARE.....	5
1.1.2 VERIFICACIÓN	6
1.1.3 VALIDACIÓN	6
1.1.4 PROCESO	7
1.1.5 ÁREA DE PROCESO.....	7
1.1.6 REALIDAD VIRTUAL.....	8
1.1.6.1 CONCEPTOS DE REALIDAD VIRTUAL.....	8
1.1.6.2 MODELO GENÉRICO DE UN SISTEMA DE REALIDAD VIRTUAL.	8
1.1.6.3 TIPOS DE REALIDAD VIRTUAL.....	9
1.2 CMMI.....	9
1.2.1 INTRODUCCIÓN A CMMI	9
1.3 ÁREA DE PROCESO VERIFICACIÓN.....	11
1.3.1 METAS PROPUESTAS POR CMMI PARA ESTA ÁREA DE PROCESO	11
1.4 ÁREA DE PROCESO VALIDACIÓN.....	14
1.4.1 METAS PROPUESTAS POR CMMI PARA ESTA ÁREA DE PROCESO	14
1.5 TÉCNICAS DE VALIDACIÓN Y VERIFICACIÓN.....	16
1.5.1 TÉCNICAS ESTÁTICAS.	16
1.5.1.1 REVISIONES TÉCNICAS FORMALES.	16
1.5.1.2 REVISIONES POR PARES.	18
1.5.1.3 AUDITORÍAS DE CALIDAD.....	19
1.5.2 TÉCNICAS DINÁMICAS.	19
1.5.2.1 NIVELES DE PRUEBAS.....	20
1.5.2.1.1 PRUEBAS DE UNIDAD	20
1.5.2.1.2 PRUEBAS DE INTEGRACIÓN	21
1.5.2.1.3 PRUEBAS DE SISTEMA	22
1.5.2.1.4 PRUEBAS DE ACEPTACIÓN.....	22

1.5.2.2 MÉTODOS DE PRUEBA.....	23
1.5.2.2.1 MÉTODO DE CAJA NEGRA.....	23
1.5.2.2.1.1 PARTICIÓN EQUIVALENTE.....	23
1.5.2.2.1.2 ANÁLISIS DE VALORES LÍMITES.....	24
1.5.2.2.2 MÉTODO DE CAJA BLANCA.....	24
1.5.2.2.2.1 PRUEBA DEL CAMINO BÁSICO.....	25
1.5.2.2.2.2 PRUEBA DE CONDICIONES.....	25
1.6 ESTUDIOS PREVIOS SOBRE VALIDACIÓN Y VERIFICACIÓN DE SOFTWARE.....	25
1.6.1 LA EVALUACIÓN DEL SOFTWARE. PRODUCTO DE SOFTWARE MÉDICO. GUÍA PARA LOS FABRICANTES EN CUANTO A LA EVALUACIÓN.....	26
1.6.2 TOP 10 DE FACTORES QUE OBSTACULIZAN LA MEJORA DE LOS PROCESOS DE VERIFICACIÓN Y VALIDACIÓN EN ORGANIZACIONES INTENSIVAS EN SOFTWARE.....	26
1.6.3 OPTIMIZACIÓN DE LOS ENSAYOS CLÍNICOS DE FÁRMACOS MEDIANTE SIMULACIÓN DE EVENTOS DISCRETOS, SU MODELIZACIÓN, VALIDACIÓN, VERIFICACIÓN Y LA MEJORA DE LA CALIDAD DE SUS DATOS.....	26
1.6.4 IMPORTANCIA DE LAS HERRAMIENTAS DE VALIDACIÓN Y VERIFICACIÓN.....	27
1.7 VALIDACIÓN Y VERIFICACIÓN EN MODELOS DE SIMULACIÓN.....	27
1.7.1 DIFICULTADES DE LA VALIDACIÓN Y VERIFICACIÓN.....	27
1.7.2 PRINCIPIOS BÁSICOS DE VALIDACIÓN Y VERIFICACIÓN.....	28
1.8 SITUACIÓN EN LATINOAMÉRICA RESPECTO A CMMI.....	29
CAPÍTULO 2: DIAGNÓSTICO DE LA SITUACIÓN ACTUAL.....	31
INTRODUCCIÓN.....	31
2.1 RESULTADO DE LAS ENTREVISTAS A LOS PROYECTOS.....	33
2.1.1 HERRAMIENTAS DE DESARROLLO PARA SISTEMAS DE REALIDAD VIRTUAL.....	33
2.1.2 SIMULADOR QUIRÚRGICO.....	36
2.1.3 JUEGOS DE CONSOLA.....	39
2.1.4 DISEÑO 3D.....	41
2.1.5 JUEGOS CNEURO.....	41
2.1.6 LABORATORIOS VIRTUALES.....	44
2.2 RESULTADOS GENERALES DE LOS PROYECTOS DE RV.....	44
CAPÍTULO 3: PROPUESTA DE SOLUCIÓN.....	48
INTRODUCCIÓN.....	48
3.1 PREPARAR LA V&V.....	48
3.1.1 SELECCIONAR LOS PRODUCTOS.....	48

3.1.2 PREPARAR EL ENTORNO DE V&V.....	51
3.1.2.1 RECURSOS HUMANOS.....	52
3.1.2.2 RECURSOS MATERIALES.....	54
3.1.3 ESTABLECER LOS PROCEDIMIENTOS Y CRITERIOS DE V&V.....	56
PROCEDIMIENTOS DE V&V.....	56
3.1.3.1 TÉCNICAS ESTÁTICAS.....	56
3.1.3.1.1 REVISIONES.....	56
3.1.3.1.2 AUDITORÍAS DE CALIDAD.....	60
3.1.3.2 TÉCNICAS DINÁMICAS.....	60
CRITERIOS DE V&V.....	63
CRITERIOS PARA LA VERIFICACIÓN.....	63
CRITERIOS PARA LA VALIDACIÓN.....	66
3.2 VALIDAR Y VERIFICAR LOS PRODUCTOS SELECCIONADOS.....	73
3.2.1 REALIZAR LA VALIDACIÓN Y LA VERIFICACIÓN.....	73
3.2.2 ANALIZAR LOS RESULTADOS DE LA V&V.....	75
3.3 VALIDACIÓN DE LA SOLUCIÓN PROPUESTA.....	76
CONCLUSIONES.....	79
RECOMENDACIONES.....	80
BIBLIOGRAFÍA.....	81
ANEXOS.....	84
ANEXO 1: MODELO GENÉRICO DE UN AMBIENTE VIRTUAL.....	84
ANEXO 2: ESTRUCTURA DE LA REPRESENTACIÓN CONTINUA DE CMMI.....	85
ANEXO 3: ESTRUCTURA DE LA REPRESENTACIÓN ESCALONADA DE CMMI.....	86
ANEXO 4: NIVELES DE CAPACIDAD Y MADUREZ PRESENTES EN CADA REPRESENTACIÓN DE CMMI.....	87
ANEXO 5: ESTRUCTURA DEL MODELO CMMI.....	88
ANEXO 6: TÉCNICAS DE V&V EN EL CICLO DE VIDA DEL SOFTWARE.....	89
ANEXO 7: LISTA DE CHEQUEO SOBRE EL PROCESO DE VALIDACIÓN Y VERIFICACIÓN.....	90
ANEXO 8: INFORME FINAL DE REVISIÓN.....	93
ANEXO 9: PLAN DE REVISIÓN DE SOFTWARE.....	94
ANEXO 10: CUESTIONARIO UTILIZADO PARA LA VALIDACIÓN DE LA PROPUESTA.....	96
ANEXO 11: RESULTADOS DE LA VALIDACIÓN DE LA PROPUESTA.....	97
GLOSARIO DE TÉRMINOS.....	98

INTRODUCCIÓN

La industria de software se desarrolla a un ritmo vertiginoso a nivel mundial, pues el número de personas, empresas e instituciones que se dedican a la producción de software aumenta considerablemente con el transcurso del tiempo. Todo este desarrollo lleva consigo un incremento de las competencias en el mercado de software internacional lo que ha despertado en los desarrolladores una gran preocupación por la calidad del producto final.

Independientemente de este desarrollo continuo de la industria de software, la situación actual de esta muestra un pobre desempeño de las organizaciones de software, un incremento de la presión por mejorar su desempeño y dificultades con las mejoras de sus procesos. Todo esto provoca una crisis de que trae como resultado una insuficiente calidad del producto final, estimaciones de duración de proyectos y asignación de recursos inexactas, retrasos en la entrega de productos terminados, están fuera de control los costos de desarrollo y mantenimiento de productos, escasez de personal calificado en un mercado laboral de alta demanda y tendencia al crecimiento del volumen y complejidad de los productos. (1)

Para hacer frente a esta situación la comunidad involucrada en el desarrollo del software ha reaccionado con diversas iniciativas metodológicas, tales como:

- Definición de modelos de referencia. Así nació el Modelo de Madurez de las Capacidades Integrado (CMMI/SW), diseñado por el Instituto de Ingeniería del Software (SEI), que establece cinco niveles de madurez, especificando para cada uno de ellos los objetivos que deben ser cubiertos para que una organización pueda ser calificada con el nivel de madurez correspondiente.
- Establecimiento de normas y guías para el desarrollo de software. Éstas han sido promovidas o generadas por entidades de reconocido prestigio (IEEE, SEI, BSI, AENOR) y se han orientado a definir el ciclo de vida del software, a normalizar la terminología o a desarrollar aspectos específicos del ciclo de vida, tales como la documentación, la verificación y validación o las pruebas de componentes (2).

Cuba no se encuentra exenta de esta situación pues la Industria Cubana del Software (ICSW) está llamada a convertirse en una significativa fuente de ingresos nacional, como resultado del correcto aprovechamiento de las ventajas del considerable capital humano disponible. La Universidad de las

Ciencias Informáticas juega un papel importante en el desarrollo de la Industria Cubana del Software, y en la materialización de los proyectos asociados al programa cubano de informatización. (3)

Pero la realidad cubana según el criterio de los especialistas es que el nivel de resultados obtenidos en la exportación de software no se corresponde con el elevado nivel de dedicación, promoción, recursos económicos, participación en misiones, ferias y eventos que se han puesto en función de esta actividad (4).

Entre las causas se consideran, en primer orden:

- La escasa imagen que se tiene de Cuba en el exterior como un país productor de software. Sólo a partir de la Feria Informática 2004 se ha comenzado a crear cierta imagen para el mercado externo, mediante la Industria Cubana del Software.
- La falta de certificación de las empresas cubanas, a partir de los estándares internacionales de calidad, constituye un factor importante.

En la actualidad en Cuba se realizan grandes esfuerzos por cambiar esta situación y como resultado de estos esfuerzos se puede citar el caso de la Empresa de Tecnologías de la Información y Servicios Telemáticos Avanzados (CITMATEL) que se ha convertido en una de las primeras empresas dentro del campo de las tecnologías de la información que certificó su sistema de gestión de calidad con la norma cubana NC ISO 9001 del 2001, y con ello se ha sumado al grupo de entidades que en Cuba cuentan con ese aval. (5)

En la UCI también se llevan a cabo disímiles tareas en busca de una mayor calidad de los productos finales entre las que se puede encontrar, la creación por parte de la dirección de calidad de software de la universidad de los grupos de calidad en cada una de las diez facultades por la que esta compuesta dicha institución. Estos grupos fueron creados con el objetivo de elevar la calidad en todos los proyectos productivos asociados a cada facultad.

Para que los proyectos de la Universidad de las Ciencias Informáticas (UCI) trabajen con la calidad esperada por los clientes, es necesario algo más que unas pruebas finales en las que se comprueba que el software coincide con las especificaciones. Una correcta estrategia de validación y verificación ayudará a mejorar y asegurar la calidad de los proyectos y permitirá reducir los costes de corrección de errores.

La facultad 5 de la UCI no realiza en sus proyectos de realidad virtual una correcta aplicación de los procesos de validación y verificación de software. Esta situación problemática trae consigo atrasos de los productos con respecto a los plazos fijados con los clientes, mala o escasa documentación de los productos de trabajo y todo esto atenta fuertemente contra la calidad final del software. Para el desarrollo de esta investigación se ha definido el siguiente **problema científico**:

¿Cómo realizar los procesos de validación y verificación en proyectos de realidad virtual?

El **objeto de investigación** establecido para el desarrollo de esta estrategia es el proceso de desarrollo de software de realidad virtual y el **campo de acción** en el cual estará enmarcada son los procesos de validación y verificación.

Desde los inicios de la investigación surge la siguiente idea a defender:

Con la aplicación de una estrategia para los procesos de validación y verificación en los proyectos de realidad virtual de la facultad 5 se logrará una mayor calidad en los productos finales.

Para dar solución al problema planteado se ha concebido como **objetivo general**, desarrollar una estrategia para realizar los procesos de validación y verificación en los proyectos de realidad virtual de la facultad 5 de la UCI basada en CMMI. Para poder dar cumplimiento a este objetivo se trazaron los siguientes **objetivos específicos**:

- Realizar un estudio del arte profundizando en las metas que propone el modelo CMMI para las áreas de proceso validación y verificación.
- Definir criterios y procedimientos para la validación y verificación.
- Proponer guías para preparar el entorno de validación y verificación.
- Proponer plantillas para la documentación de las actividades de validación y verificación.

Para dar respuesta a estos objetivos se utilizó como método teórico el analítico-sintético, lo que permite buscar la esencia de los fenómenos que se encuentran alrededor del objeto de investigación y con ello lograr encontrar los rasgos que lo caracterizan y distinguen. Además se aplicó el método de la entrevista como método empírico, para lo cual fue preparada una lista de chequeo que permitió obtener información sobre el estado de los proyectos.

El contenido del presente trabajo está estructurado en tres capítulos:

Capítulo 1: Fundamentación Teórica.

En este capítulo se recogen un conjunto de conceptos y definiciones necesarios para la comprensión del desarrollo de este trabajo; se describen las metas propuestas por el modelo CMMI para realizar los procesos de validación y verificación, así como un conjunto de técnicas estáticas y dinámicas que serán usadas durante la solución.

Capítulo 2: Diagnóstico de la Situación Actual.

En este capítulo se presentan los resultados arrojados de las entrevistas realizadas a los líderes de 6 de los proyectos de realidad virtual de la facultad 5. Estas entrevistas fueron guiadas en su totalidad por una lista de chequeo basada en las metas específicas y sus respectivas prácticas propuestas por el modelo CMMI en las áreas de procesos validación y verificación.

Capítulo 3: Propuesta de Solución.

Este capítulo está centrado en su totalidad en la propuesta de una solución para resolver los problemas detectados al momento de realizar los procesos de validación y verificación en los proyectos de realidad virtual de la facultad 5. Además recoge la validación de la solución propuesta según el criterio de especialistas en calidad y desarrollo de software de realidad virtual.

CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA

INTRODUCCIÓN

En este capítulo se describen los fundamentos teóricos que sustentan toda la investigación, en el mismo se definen conceptualmente una serie de términos que permitirán una mayor comprensión del trabajo. Se brinda una introducción a CMMI en la que se referencia algunas de las características fundamentales del modelo. También se hace alusión a las metas generales y específicas que propone CMMI para las áreas de proceso validación y verificación (V&V) del nivel 3 de madurez o definido. Como parte de este capítulo se identifican algunos principios y dificultades del proceso que se deben tomar en consideración para definir la estrategia de V&V,

1.1 PRINCIPALES CONCEPTOS

1.1.1 CALIDAD DE SOFTWARE

La creciente demanda de software de calidad ha provocado que la calidad sea un factor de discriminación entre productos de software, contribuyendo en gran medida a su éxito. Si bien existe un acuerdo común de que se necesita software de calidad, no existe un consenso de cómo, cuándo y dónde evaluar calidad. El concepto de "calidad" es un tanto ambiguo ya que diferentes personas pueden darle significados e interpretaciones distintas.

- Según CMMI calidad es la habilidad inherente de un producto, componente del producto o proceso para cumplir los requisitos de los clientes. (6)
- La definición propuesta por Watts S. Humphrey en su libro *Introducción al Proceso De Software Personal (PSP)* plantea que la calidad de software consiste en satisfacer las necesidades de los usuarios haciendo el trabajo de los mismos de una forma fiable y consistente. Esto requiere que el software que hagas tenga pocos defectos. (7)
- Roger S. Pressman define calidad como concordancia con los requisitos funcionales y de rendimiento explícitamente establecidos, con los estándares de desarrollo explícitamente documentados, y con las características implícitas que se espera de todo software desarrollado profesionalmente (8).

Como se puede observar claramente todos estos autores dirigen su concepto de calidad de software hacia las necesidades del cliente, es lógico, pues estos últimos son los más apropiados para determinar si el software puesto en sus manos cumple con sus expectativas y satisface todas sus necesidades.

1.1.2 VERIFICACIÓN

CMMI define como verificación la confirmación de que los productos de trabajo reflejan los requisitos especificados para ellos. En otros términos, la verificación asegura que “usted construyó el producto correctamente.” (6)

Existen muchos otros conceptos de verificación según el entender de muchos autores entre los que se pueden citar:

- Roger S. Pressman plantea que la verificación se refiere al conjunto de actividades que aseguran que el software implementa correctamente una función específica (8).
- Determina si los requisitos de un sistema o software son completos y correctos y si los productos de software en cada fase de desarrollo cumplen con los requisitos o condiciones impuestas sobre ellos en las fases previas. (9)
- Es la operación de establecer las especificaciones y entradas adecuadas a una actividad y la de establecer que las salidas de las actividades son correctas y consistentes con las especificaciones y las entradas. (10)
- Es un proceso para determinar si los productos de software de una actividad cumplen las condiciones o los requisitos impuestos sobre éstos en las actividades precedentes. (11)

Es de simple apreciación ver como estos conceptos centran su atención en que, la verificación es el proceso de corroborar el cumplimiento correcto de cada uno de los requisitos establecidos para los productos; como bien define CMMI es el aseguramiento de que usted construyó el producto correctamente.

1.1.3 VALIDACIÓN

CMMI define como validación el proceso de confirmación que el producto proporcionado cumplirá su uso esperado en un ambiente operativo planeado. En otros términos, la validación asegura que “usted construyó el producto correcto.” (6)

Muchos otros autores tienen conceptos diferentes sobre validación entre los que se pueden encontrar:

- Roger S Pressman, según su criterio, la validación es el conjunto diferente de actividades que aseguran que el software construido se ajusta a los requisitos del cliente (8).
- Determina si los requisitos y el sistema o software construidos al final cumplen con su uso proyectado. (9)
- Es un proceso para determinar si los requisitos y el sistema o el producto de software, tal como se han construido, cumplen con su uso específico previsto. (11)

Atendiendo a los conceptos propuestos por diferentes autores se puede llegar a la conclusión que el proceso de validación tiene como objetivo confirmar que el uso del software cumplirá las expectativas esperadas por el cliente; como bien plantea CMMI, asegura que usted construyó el producto correcto.

1.1.4 PROCESO

Existen diferentes conceptos relacionados con el término proceso entre los que se pueden citar:

- Del latín procesus o procedere, proceso es el conjunto de fases sucesivas de un fenómeno en un lapso de tiempo. Es la marcha hacia un fin determinado. (12)
- Sucesión de actos, vinculados entre sí, respecto de una cosa común. (13)

Otros autores tienen sus propias definiciones de proceso que son más cercanas a términos informáticos, entre las que se encuentran:

- En CMMI se define como las actividades que pueden ser reconocidas, como las implementaciones de prácticas del modelo. Estas actividades pueden ser dirigidas a una o más áreas de proceso del CMMI para permitir un modelo que sea útil para la mejora de procesos. (6)
- En Ingeniería de Software. Un enfoque práctico, se define como un marco de trabajo de las tareas que se requieren para construir software de alta calidad (8).
- En Introducción al Proceso Software Personal se establece como proceso al conjunto definido de pasos para hacer un trabajo. (7)

1.1.5 ÁREA DE PROCESO

- Es un conjunto de prácticas relacionadas en un área que, al realizarse, satisfacen un conjunto de metas que se consideran importantes para lograr mejoras significativas en el área. (6)

1.1.6 REALIDAD VIRTUAL

1.1.6.1 CONCEPTOS DE REALIDAD VIRTUAL

- La realidad virtual (RV) es una simulación tridimensional generada o asistida comúnmente por computadora de algún aspecto del mundo real o ficticio, en el cual el usuario tiene la sensación de pertenecer a ese ambiente sintético o interactuar con él. La RV permite interactuar con mundos tridimensionales de una manera más natural, por ejemplo, un usuario puede realizar acciones dentro de un modelo virtual, desplazarse, moverse, caminar a través de él o levantar cosas, y de esta forma experimentar situaciones que se asemejan al mundo real (14).
- Es un paso más allá de lo que sería la simulación por computadores, tratándose más bien de una simulación interactiva, dinámica y en tiempo real. (15)
- Es el medio que proporciona una visualización participativa en tres dimensiones y la simulación de mundos virtuales. En los sistemas más avanzados, los participantes pueden incluso experimentar distintas sensaciones provocando la percepción de inmersión dentro de dicho mundo virtual. (15)

1.1.6.2 MODELO GENÉRICO DE UN SISTEMA DE REALIDAD VIRTUAL.

Las partes básicas de un sistema de realidad virtual son: el modelo de simulación, la representación del ambiente virtual, la entrada/salida y por supuesto el usuario (Ver anexo 1).

Modelo de simulación: El modelo es la representación matemática del sistema que se está presentando. Un modelo virtual necesita responder dinámicamente en respuesta de la entrada del usuario. Se pueden crear modelos matemáticos sofisticados de realidad virtual, pero lo importante es la manera en que estos modelos están asociados con un sistema visual y auditivo.

Entrada: Los dispositivos de entrada se utilizan para interactuar con el ambiente virtual como puede ser el teclado o el mouse, un joystick o guantes, pedales, etc.

Salida: La salida se refiere a la tecnología mediante la cual el usuario percibe estímulos, esta tecnología abarca un amplio rango y va desde aparatos de despliegue visual, sistemas sonoros, sistemas sensoriales, entre otros.

Usuario: El usuario es la razón de existir de un sistema de realidad virtual pues es él quien recibe los estímulos de parte del sistema y a su vez se encarga de retroalimentarlo y definir su comportamiento.

1.1.6.3 TIPOS DE REALIDAD VIRTUAL

Existen dos tipos de realidad virtual, los cuales se describen brevemente a continuación:

Sistemas inmersivos: Los sistemas inmersivos son aquellos sistemas donde el usuario se siente dentro del mundo virtual que está explorando. Este tipo de sistemas utiliza diferentes dispositivos denominados accesorios, como pueden ser guantes, trajes especiales, visores o cascos, estos últimos le permiten al usuario visualizar los mundos a través de ellos, y precisamente estos son el principal elemento que lo hacen sentirse inmerso dentro de estos mundos. Este tipo de sistemas son ideales para aplicaciones de entrenamiento o capacitación.

Sistemas no inmersivos: Los sistemas no inmersivos o de escritorio, son aquellos donde el monitor es la ventana hacia el mundo virtual y la interacción es por medio del teclado, micrófono, mouse o joystick, este tipo de sistemas son idóneos para visualizaciones científicas, también son usadas como medio de entretenimiento y aunque no ofrecen una total inmersión son una buena alternativa de bajo costo.

1.2 CMMI

1.2.1 INTRODUCCIÓN A CMMI

CMMI (Modelo de madurez y capacidad integrado) es un modelo que brinda una guía con los elementos esenciales para que las organizaciones puedan desarrollar software con una alta calidad. Este modelo fue creado por el SEI, centro de investigación y desarrollo adscrito a la Carnegie Mellon University of Philadelphia con el objetivo de:

- Proporcionar un marco que ayude a la organización a mejorar sus procesos.
- Proporcionar una guía para mejorar la capacidad de desarrollar, adquirir y mantener los productos o servicios que proporciona una organización.
- Describir un conjunto de buenas prácticas, tanto de gestión como de ingeniería.

CMMI permite a las organizaciones ver cual es su nivel de madurez y su capacidad en cuanto a los procesos que se llevan a cabo en ella. Durante los últimos años CMMI se ha convertido en uno de los estándares de calidad de software más importantes por lo que puede considerarse como el estándar de mayor aceptación de esta disciplina.

La versión más reciente de CMMI es el documento "CMMI for Development", versión 1.2, disponible desde agosto de 2006. En él se cubren los modelos CMMI for Development (CMMI-DEV, CMMI para

Desarrollo) y CMMI for Development + IPPD (CMMI-DEV+IPPD, CMMI para Desarrollo y para Desarrollo Integrado de Productos y Procesos). Estos modelos se aplican al desarrollo y mantenimiento de productos y servicios, independientemente de la disciplina o área de interés.

Para poder implementar este modelo lo primero que debe hacerse es escoger el tipo de representación que se desea por parte de la institución; los tipos de representación que pueden ser escogidos son:

- Representación Continua.
- Representación Escalonada.

En cualquiera de los dos casos las áreas de proceso, objetivos y prácticas son los mismos. La diferencia entre ambas representaciones está en el orden en el que se lleva a la práctica la mejora de procesos:

- En el caso de la representación continua, se seleccionan los procesos a mejorar en el orden que la organización considere oportuno en base a sus objetivos de negocio. Para cada área de proceso existen unos niveles de capacidad (de 0 a 5) que proporcionan una indicación acerca del orden en el que se debe abordar la mejora dentro de cada una de ellas.
- En el caso de la representación escalonada las áreas de proceso a mejorar se eligen en un orden predefinido que viene dado por su asignación a niveles de madurez de la organización.

En los anexos 2 y 3 se pueden apreciar claramente las estructuras de las representaciones continua y escalonada de CMMI respectivamente.

En las representaciones continua y escalonada están presentes diferentes niveles de capacidad y madurez respectivamente. (Ver anexo 4).

En la representación escalonada, los niveles de madurez son acumulativos lo que significa que para alcanzar cada uno de ellos deben implementarse tanto las áreas de proceso específicas de ese nivel como las de todos los niveles inferiores.

En la representación continua las áreas de procesos de este modelo son agrupadas en cuatro grupos:

- **Gestión de proceso:** Incluye las áreas de proceso que tienen que ver con la definición, planificación e implementación de los procesos en la organización.
- **Gestión de proyecto:** Incluye las áreas de proceso que cubren las actividades de planificación y seguimiento de los proyectos.
- **Ingeniería:** Incluye las áreas de proceso relacionadas con las actividades de desarrollo y mantenimiento de los productos fabricados por la organización.

- **Soporte:** Incluye las áreas de proceso que describen actividades realizadas en el contexto de otros procesos y que sirven como apoyo en el desarrollo de productos.

Cada área de proceso esta compuesta por los siguientes elementos:

- **Objetivos específicos (SG) y objetivos genéricos (GG):** Estos permiten conocer cuales son las actividades que la organización debe realizar para poder asegurar que el área de proceso está implementado. Los objetivos específicos solo se aplican a un área de proceso, mientras que los objetivos genéricos aparecen en varias.
- **Prácticas específicas (SP) y prácticas genéricas (GP):** Son el conjunto de actividades que una organización suele llevar a cabo para cumplir los objetivos de cada área de proceso. No obstante, no son obligatorias para el correcto cumplimiento del área de proceso. Las prácticas genéricas tienen como objetivo asegurar que los procesos se llevarán a cabo de forma eficaz, repetible y duradera.
- **Productos típicos, subprácticas, notas, referencias, etc.** son elementos que pueden servir para ayudar a entender los objetivos y prácticas y la mejor manera de abordarlas. Como se ha mencionado anteriormente el desarrollo de esta investigación estará centrado en las áreas de proceso verificación y validación, las cuales pertenecen al nivel de madurez número 3 o definido. Aunque ambos procesos se encuentran estrechamente relacionados y tienen actividades similares cada uno persigue objetivos diferentes, la verificación tiene como objetivo demostrar que el producto se ajusta a los requisitos especificados, mientras que el objetivo de la validación es demostrar que el producto se ajusta al uso que de él se va a hacer en el entorno de producción. Esta idea se puede resumir en la expresión: “la verificación demuestra que el producto se ha construido correctamente, la validación demuestra que se ha construido el producto correcto”.

1.3 ÁREA DE PROCESO VERIFICACIÓN

1.3.1 METAS PROPUESTAS POR CMMI PARA ESTA ÁREA DE PROCESO

El propósito de esta área de proceso es asegurar que los productos de trabajo seleccionados responden a los requerimientos especificados. A continuación se desglosan las metas a conseguir con este proceso, y las prácticas que se requieren para conseguir estas metas:

Prácticas específicas

SG 1. Preparar la verificación

SP 1.1. Seleccionar los productos de trabajo para la verificación

SP 1.2. Establecer el entorno de verificación.

SP 1.3. Establecer los procedimientos y criterios de verificación.

SG 2. Realizar revisiones por pares

SP 2.1. Preparar revisiones por pares.

SP 2.2. Realizar revisiones por pares.

SP 2.3. Analizar resultados de revisiones por pares.

SG 3 Verificar los productos de trabajo seleccionados

SP 3.1. Realizar la verificación.

SP 3.2. Analizar los resultados de la verificación.

SG 1. Preparar la verificación.

Es necesario preparar la verificación para garantizar que se creen todas las condiciones necesarias para llevar este proceso. La verificación incluye la selección, inspección, ensayo, análisis y demostración de los productos del trabajo. La preparación también implica la definición de las herramientas de apoyo, equipos de prueba, software, simulaciones, prototipos y demás condiciones necesarias para poder acometer esta tarea.

SP 1.1. Seleccionar los productos de trabajo para la verificación.

Se seleccionan los productos de trabajo que tienen que ser verificados y los métodos de verificación que se utilizarán para cada uno. Los productos de trabajo son seleccionados sobre la base de su contribución al cumplimiento de los objetivos del proyecto y a las necesidades de hacer frente a los riesgos del proyecto.

SP 1.2. Establecer el entorno de verificación.

El objetivo de esta práctica es establecer y mantener el ambiente de verificación, este se puede adquirir, desarrollar, reutilizar, modificar, o una combinación de éstas, dependiendo de las necesidades del proyecto. El tipo de entorno utilizado dependerá de los productos seleccionados para la verificación

y los métodos de verificación aplicados. El producto de trabajo de esta práctica es precisamente el ambiente de verificación.

SP 1.3. Establecer los procedimientos y criterios de verificación.

Se deben establecer y mantener los procedimientos y criterios de verificación que serán aplicados a los productos seleccionados. Los procedimientos y criterios de verificación deben elaborarse conjuntamente e iterativamente con el producto y/o los componentes del producto. Estos criterios se establecen para garantizar que los productos cumplen con las especificaciones necesarias.

SG 2. Realizar revisiones por pares

Las revisiones se realizan sobre una selección de los productos de trabajo. Este tipo de revisiones, suponen un examen metódico de los productos para detectar defectos y recomendar los cambios que se necesiten.

La revisión en equipo es un método importante y eficaz de verificación aplicado a través de inspecciones, tutoriales, o una serie de métodos con estos fines. Este tipo de revisiones, son principalmente aplicados a los productos de trabajo elaborados por el proyecto, pero también se pueden aplicar a otros productos como la documentación de los productos de trabajo.

SP 2.1. Preparar revisiones por pares.

Las actividades de preparación para revisiones por pares suelen incluir la identificación del personal de los que se invitará a participar en la evaluación de cada producto de trabajo, la preparación y actualización de todos los materiales que se utilizarán en las revisiones, tales como listas de chequeo, criterios de revisión y la programación.

SP 2.2. Realizar revisiones por pares.

Realizar evaluaciones por pares sobre una selección de los productos de trabajo e identificar cuestiones derivadas de la revisión por equipos. Uno de los objetivos de una revisión por pares es encontrar y eliminar defectos antes de tiempo.

SP 2.3. Analizar resultados de revisiones por pares.

Se analizan los datos acerca de la preparación, la realización y los resultados de las verificaciones por pares.

SG 3 Verificar los productos de trabajo seleccionados

Se verifican los productos de trabajo seleccionados contra los requerimientos especificados. Las actividades de verificación se deben realizar en todo el ciclo de vida del producto.

SP 3.1. Realizar la verificación.

Se realiza la verificación de los productos de trabajo seleccionados. Llevar a cabo la verificación de los productos de trabajo desde temprano en el ciclo de vida del software promueve la detección de problemas y puede dar lugar a la pronta eliminación de los defectos.

SP 3.2. Analizar los resultados de la verificación.

Se analizan los resultados de todas las actividades de verificación y los reales se comparan con los criterios de verificación establecidos para determinar la aceptabilidad. Los resultados del análisis se registran como prueba de que la verificación se llevó a cabo. Dado que una revisión por pares es uno de los métodos de verificación, los resultados de esta revisión deben ser incluidos también en este análisis para lograr un análisis más profundo de la situación real del producto.

1.4 ÁREA DE PROCESO VALIDACIÓN

1.4.1 METAS PROPUESTAS POR CMMI PARA ESTA ÁREA DE PROCESO

El propósito del área de proceso Validación es demostrar que un producto o componente satisface su uso esperado, en el ambiente operativo planeado. A continuación se desglosan cuales son las metas a implementar en este proceso y las prácticas que se deben desarrollar para cumplir estas metas, además se brinda una explicación general sobre cada una de ellas.

SG 1. Preparar la validación.

- SP 1.1. Seleccionar los productos a validar.
- SP 1.2. Establecer el entorno de validación.
- SP 1.3. Establecer los procedimientos y criterios de validación.

SG 2. Validar los productos o componentes de los productos.

- SP 2.1. Realizar la validación.
- SP 2.2. Analizar los resultados de la validación.

SG 1. Preparar la validación.

Su preparación incluye seleccionar los productos y componentes para validar, estableciendo y manteniendo el ambiente, además de definir criterios y procedimientos. Los elementos seleccionados para la validación pueden ser el producto o solo los componentes del producto. El ambiente requiere ser preparado para validar el producto o sus componentes, debido a que el mismo puede comprarse, especificarse, diseñarse o construirse.

SP 1.1. Seleccionar los productos a validar.

Se seleccionan productos y componentes del producto para la validación en base a las necesidades del usuario. Para cada componente del producto, debe determinarse el alcance de la validación. Además se definen cuales son los requisitos para realizar la validación y poder seleccionar cuales serán los métodos que se usarán.

SP 1.2. Establecer el entorno de validación.

Esta práctica tiene como objetivo establecer y mantener el ambiente necesario para apoyar la validación. Los requisitos para el ambiente de validación dependen del producto o los componentes del producto seleccionados, del tipo de productos de trabajo y de los métodos de validación. Estos requisitos pueden ser la compra o desarrollo de equipos, software u otros recursos. El producto de esta práctica es precisamente el ambiente de validación.

SP 1.3. Establecer los procedimientos y criterios de validación.

Se definen procedimientos y criterios de validación para asegurar que el producto o componente del producto cumplirán su uso esperado cuando se pongan en su ambiente real. Los casos de prueba de aceptación y procedimientos pueden satisfacer las necesidades de los procedimientos de validación.

SG 2. Validar los productos o componentes de los productos.

Se valida el producto o componentes del producto para asegurar que ellos, una vez ubicados en su ambiente de trabajo real, podrán cumplir con las expectativas esperadas. Estas actividades de validación se realizan a lo largo del ciclo de vida del producto.

SP 2.1. Realizar la validación.

El objetivo de esta práctica es realizar la validación de los productos o componentes del producto seleccionados. Para que los productos puedan ser aceptados por los usuarios la validación debe realizarse en un ambiente real. Una vez que se realizan las actividades de validación, los datos

resultantes son reunidos según los métodos establecidos, procedimientos, y criterios para luego documentarse y anotar las desviaciones que ocurren durante la ejecución.

SP 2.2. Analizar los resultados de la validación.

En esta práctica se analizan los datos que son el resultado de las pruebas de validación, inspecciones, las demostraciones, o análisis de las evaluaciones contra el criterio de validación definido. El análisis de los informes indica si las necesidades fueron satisfechas; en el caso de las deficiencias, estos informes documentan el grado de éxito o fracaso y categorizan la causa probable del fracaso.

Se analiza el informe o la documentación de como ejecutar la validación, también puede indicarse que los malos resultados de las pruebas son debidos a un problema de los procedimientos de validación o a un problema de ambiente de validación.

1.5 TÉCNICAS DE VALIDACIÓN Y VERIFICACIÓN.

Los procesos de validación y verificación (V&V) deben aplicarse en todas las etapas del ciclo de vida del software y tienen como principal objetivo asegurarse de que el software cumple con su especificación y satisface las necesidades del usuario. Para satisfacer estos objetivos de V&V es necesario aplicar algunas técnicas que en su mayoría se pueden clasificar como estáticas y dinámicas.

Como se ha planteado anteriormente el proceso de V&V se debe aplicar durante todo el ciclo de vida del software pero se debe aclarar que las técnicas estáticas de verificación pueden ser usadas durante todo el proceso de software; las dinámicas, sin embargo, sólo pueden ser usadas cuando disponemos de un prototipo o una versión ejecutable del producto (Ver anexo 6).

1.5.1 TÉCNICAS ESTÁTICAS.

Las técnicas estáticas son utilizadas fundamentalmente en el proceso de verificación y tienen que ver con el análisis y control de las representaciones del sistema, es decir de los diferentes modelos construidos durante el proceso de desarrollo de software tales como: documentos de requerimientos, diagramas de análisis y diseño y código fuente. En esta categoría caen, por ejemplo, las revisiones técnicas formales, revisiones por pares y auditorias.

1.5.1.1 REVISIONES TÉCNICAS FORMALES.

Es una actividad de garantía de calidad del software y es una reunión específica de revisión que se ha mostrado efectiva en el descubrimiento de defectos. Las Revisiones Técnicas Formales (RTF) son una

actividad colectiva lo que permite ampliar la mirada sobre lo que se revisa, situación que se profundiza al ser aplicada por distintos niveles y especialidades de profesionales a distintos elementos que componen el software.

OBJETIVOS DE LAS RTF

- Descubrir errores en la función, la lógica o la implementación de cualquier representación del software.
- Garantizar que se siguen los estándares definidos.
- Conseguir un desarrollo uniforme.
- Facilitar la gestión de los proyectos.
- Verificar el cumplimiento de los requisitos
- Obtener proyectos que hagan más sencillo los trabajos técnicos (análisis que permitan buenos diseños, diseños que permitan implementaciones sencillas, estrategias de pruebas que faciliten éstas)

DIRECTRICES DE LA REVISIÓN:

Para realizar las RTF es necesario establecer un conjunto de directrices para guiar este tipo de revisión. A continuación se presentan un grupo de ellas:

- Se debe hacer la revisión del producto, no del productor.
- Fijar una agenda y mantenerla.
- Limitación del debate e impugnaciones. Algunos asuntos pueden dejarse para discusión posterior
- No se resuelve el problema, sólo se identifica.
- Limitar el número de participantes.
- Destinar recursos y agenda para las RTF en la planificación.
- Entrenamiento de los revisores.
- Repaso de revisiones anteriores
- Usar una lista de chequeo para el producto a revisar.

VENTAJAS

- Más efectivo que las pruebas para encontrar errores. Encuentra la causa del error en lugar del síntoma.

- Los programadores saben que sus programas serán revisados por lo que los programas son más fáciles de entender y los programadores son más cuidadosos.
- Anula el efecto de "puntos ciegos" (programador pasa reiteradamente sobre el error sin verlo).
- Es posible imponer estándares de codificación con facilidad.
- Reduce considerablemente el tiempo dedicado a las pruebas.

DIFICULTADES

- Los problemas de personalidad:
 - ✓ Personas con buenas ideas no se expresan.
 - ✓ Personas con malas ideas se expresan mucho.
 - ✓ Algunas personas odian discutir o estar en desacuerdo.
 - ✓ Fácil de perderse en cosas triviales.
- Es agotador (pierde efectividad después de un par de horas o menos)
- Dificultades para convencer a gerentes de beneficios económicos.

1.5.1.2 REVISIONES POR PARES.

Estas revisiones son realizadas por personas externas al proyecto y para cada producto, componente o artefacto a revisar se convocan a integrantes de otros proyectos que han cumplido los mismos roles que los desarrolladores de los elementos a revisar.

En el mundo del software una revisión por pares no es solo una oportunidad para mejorar la calidad del producto a través de la detención temprana de defectos, como errores de programación, o desviaciones de la especificación, de la arquitectura, o de las buenas prácticas. Es también una importante oportunidad de aprendizaje para los participantes, y en ello una oportunidad de mejorar el entorno.

Una revisión por pares enfocada solo en encontrar anomalías, o "el qué", pierde la valiosa oportunidad de analizar también el "cómo" y el "por qué":

- Falta de conocimiento.
- Herramientas inadecuadas.
- Tareas repetitivas y posibles a equivocaciones.

Durante una revisión por pares buscamos que el código sea "bueno". Un código que no es de calidad puede pasar una primera inspección y ser todavía vulnerable a la ocurrencia de cientos de defectos en

una siguiente iteración. Esto puede no ocurrir solamente con el código, pues cualquier artefacto, producto o componente del producto puede estar sujeto a errores.

Por esta razón se trata de que los participantes (desarrolladores y revisores) en estas revisiones logren retroalimentarse cada uno de ellos con los conocimientos de los demás para que en futuros programas logren hacer buen uso de lo aprendido y obtener productos con una mayor calidad.

De forma general se puede decir que una revisión por pares no es solo una oportunidad para detectar y corregir los errores de hoy, sino para aprender, y así poder hacer del trabajo de manera más fácil y mejor en el futuro.

1.5.1.3 AUDITORÍAS DE CALIDAD

Una auditoría es una actividad dirigida por una persona autorizada con el propósito de proporcionar una evaluación independiente de los productos y procesos de software con el fin de evaluar la conformidad con los requisitos. (11).

La auditoría es una herramienta que nos permite evaluar la eficacia del sistema establecido, sus defectos y los medios de mejora. Esta técnica compara un producto o componente de este contra los criterios específicos que el mismo exige (por ejemplo, requerimientos). Además, una auditoría involucra una comparación entre la calidad observada y la calidad deseada de acuerdo con unas normas o criterios previamente establecidos.

VENTAJAS DE LAS AUDITORÍAS DE CALIDAD

Esta técnica concentra el esfuerzo en ámbitos organizativos y de procedimientos puntuales, esto tiene entre sus principales ventajas:

- Consiguen mejoras en un corto plazo y resultados visibles.
- Constituye un elemento importante para la reducción de los costos de los productos de trabajo.
- Incrementa la productividad y dirige a la organización hacia la competitividad, lo cual es de vital importancia para las actuales organizaciones.
- Contribuye a la adaptación de los procesos a los avances tecnológicos.

1.5.2 TÉCNICAS DINÁMICAS.

Las técnicas dinámicas, también conocidas como prueba (o testing, en inglés) se basan en ejercitar una implementación. Por lo tanto, sólo pueden ser aplicadas si existe una versión operativa o ejecutable del

producto. Independientemente del tipo de software que se desee probar es necesario seguir varias normas establecidas por Glem Myers las cuales pueden servir como objetivos de las pruebas, las mismas se enumeran a continuación:

1. La prueba es un proceso de ejecución de un programa con la intención de descubrir un error.
2. Un buen caso de prueba es aquel que tiene una alta probabilidad de mostrar un error no descubierto hasta entonces.
3. Una prueba tiene éxito si descubre un error no detectado hasta entonces.

1.5.2.1 NIVELES DE PRUEBAS.

Para realizar un buen proceso de pruebas es indispensable establecer el nivel al cual se desarrollarán las mismas, puesto que esto ayuda a acotar el número de casos de pruebas necesarios además de que permite al equipo de pruebas centrar su atención en lo que verdaderamente se necesita probar en ese momento. A continuación se describen los diferentes niveles existentes para desarrollar pruebas de software.

1.5.2.1.1 PRUEBAS DE UNIDAD

Está enfocada a probar los elementos más pequeños de un software y se aplica a los componentes representados en el modelo de implementación para verificar que los flujos de control y datos están cubiertos y que a su vez estos funcionan de manera esperada.

En otras palabras se puede ver una prueba unitaria como una forma de probar el correcto funcionamiento de un módulo de código. Esta prueba permite asegurar que cada uno de los módulos funcione correctamente por separado. La idea es escribir casos de prueba para cada función no trivial o método en el módulo de forma que cada caso sea independiente del resto.

PRUEBA DE UNIDAD EN EL CONTEXTO ORIENTADO A OBJETOS.

En vez de realizar pruebas de módulos individuales, la menor unidad a probar en este contexto es una clase u objetos de estas. Esta prueba de clases para el software orientado a objetos es equivalente a la prueba de unidad para el software convencional.

La prueba de clases para este tipo de software está dirigida por las operaciones encapsuladas por la clase y el estado del comportamiento de esta. Las operaciones no se pueden probar de forma independiente sino como parte de una clase.

VENTAJAS

El correcto desarrollo de las pruebas unitarias proporciona al equipo de trabajo ciertas ventajas, entre las que se pueden encontrar:

- **Fomentan el cambio:** Facilitan que el programador cambie el código para mejorar su estructura puesto que permiten hacer pruebas sobre los cambios y así asegurarse de que los nuevos cambios no han introducido errores.
- **Simplifica la integración:** Provee de un menor grado de dificultad a las pruebas de integración debido a que existe un alto nivel de seguridad de que el código está funcionando correctamente.
- **Los errores están más acotados y son más fáciles de localizar:** Dado que tenemos pruebas unitarias que pueden desenmascararlos.

LIMITACIONES

Las pruebas unitarias no descubren todos los errores del código debido a que sólo prueban las unidades por sí solas. Por lo tanto, no descubrirán errores de integración, problemas de rendimiento y otros problemas que afectan a todo el sistema en su conjunto. Las pruebas unitarias sólo son efectivas si se usan en conjunto con otras pruebas de software.

1.5.2.1.2 PRUEBAS DE INTEGRACIÓN

Por motivo de que cada clase probada unitariamente es agregada progresivamente dentro de la estructura del programa, las pruebas de integración son el mecanismo para asegurarse del correcto ensamblaje del sistema completo. Las pruebas de integración se centran en probar la coherencia semántica entre los diferentes módulos, tanto de semántica estática (se importan los módulos adecuados; se llama correctamente a los procedimientos proporcionados por cada módulo), como de semántica dinámica (un módulo recibe de otro lo que esperaba). Normalmente estas pruebas se van realizando por etapas, englobando progresivamente más módulos en cada prueba.

Las pruebas de integración se pueden comenzar a realizar desde el momento en que contamos con algunos módulos aunque no terminan hasta contar con la integración de todos los módulos que componen el sistema. Existen dos tipos de diseños fundamentales para pruebas de integración que son la integración descendente (Top-Down) en el cual se comienza a probar a partir de módulos más generales y la integración ascendente (Top-Up) en la que se prueba a partir de módulos de la base.

Sin lugar a dudas se recomienda como una buena alternativa la combinación de las dos estrategias (ascendente y descendente) donde se use la descendente para los niveles superiores de la estructura y la ascendente para los niveles inferiores o subordinados.

Conjuntamente con el proceso de integración se deben aplicar las llamadas pruebas de regresión, para asegurar que los módulos causantes de fallas fueron efectivamente corregidos y que no se introdujeron nuevos errores al tratar de solucionar alguno. Estas pruebas de regresión consisten simplemente en aplicar el mismo proceso de pruebas planificado originalmente, sólo que se repiten cada vez que se identifica un resultado erróneo y se le da solución.

1.5.2.1.3 PRUEBAS DE SISTEMA

Estas pruebas se realizan con el objetivo de asegurarse de que el sistema funciona como se espera o como fue diseñado. Estas pruebas pueden incluir la validación de campos, reglas de negocio, integración de procesos y la entrada de datos. Con estas pruebas se puede asegurar que el sistema es capaz de manejar el volumen de datos y que el tiempo de respuesta al usuario es el esperado.

1.5.2.1.4 PRUEBAS DE ACEPTACIÓN.

Estas pruebas son realizadas totalmente por el cliente y resulta satisfactoria cuando se logre asegurar que el software funciona de acuerdo con las expectativas de este, para esto se usan métodos de caja negra que demuestran la conformidad con los requisitos establecidos para el software. Existen dos tipos de pruebas fundamentales; dentro de las pruebas de aceptación, que son las pruebas alfa y las pruebas beta.

PRUEBAS ALFA

Estas pruebas son llevadas a cabo solamente por los clientes en el lugar de desarrollo. En este tipo de pruebas los usuarios usan el software de forma natural mientras que los desarrolladores se mantienen observando y registrando todos los errores que valla arrojando la prueba. Es necesario precisar que estas se realizan en un ambiente controlado.

PRUEBAS BETA

Las pruebas beta son también realizadas por los clientes finales pero a diferencia de las pruebas alfa estas se desarrollan en un entorno real de trabajo sin la presencia de los desarrolladores (normalmente). De esta manera estas pruebas son una aplicación en vivo del software en un entorno

que no puede ser controlado por el desarrollador y son los propios clientes quienes registran los defectos que encuentran y se los envían a los desarrolladores para que estos los corrijan.

1.5.2.2 MÉTODOS DE PRUEBA.

Para desarrollar un correcto proceso de pruebas de software es necesario establecer y utilizar los métodos necesarios, existen dos fundamentales:

- **Método de caja negra:** dirige su atención a las pruebas que se llevan a cabo sobre la interfaz del software. Se pretende demostrar que el software cumple correctamente con todas las funcionalidades operativas que requiere.
- **Método de caja blanca:** se comprueban los caminos lógicos del software proponiendo casos de prueba que se ejerciten conjuntos específicos de condiciones y/o bucles.

1.5.2.2.1 MÉTODO DE CAJA NEGRA

Con este método los casos de prueba y los resultados se determinan a partir de la especificación funcional del método de una clase. Es decir, la prueba de caja negra se refiere a las pruebas que se llevan a cabo sobre la interfaz del software. Una prueba de caja negra examina algunos aspectos del modelo fundamental del sistema sin tener mucho en cuenta la estructura lógica interna del software. Para aplicar este método existen varias técnicas, a continuación se describen brevemente algunas de ellas.

1.5.2.2.1.1 PARTICIÓN EQUIVALENTE.

Esta técnica divide los datos de entrada de un software en diferentes clases de datos de las que se pueden derivar los casos de prueba. Cada caso de prueba correcto puede descubrir una clase de error evitando así tener que hacer muchos casos de pruebas antes de poder encontrar un error genérico. El diseño de los casos de prueba para la partición equivalente consiste en evaluar cada clase equivalente en una condición de entrada, donde cada clase de equivalencia representa un conjunto de estados válidos o inválidos para estas condiciones de entrada.

1. Si una condición de entrada es la especificación de un rango, se debe definir una clase de equivalencia válida y dos inválidas. O sea una clase dentro del rango, una por debajo y una por encima del rango respectivamente.

2. Si la condición de entrada consiste en un valor específico, se debe definir una clase de equivalencia válida y dos inválidas. Es decir, una clase con el valor esperado, una con un valor inferior y otra con un valor superior al esperado respectivamente.
3. Si la condición de entrada es booleana, se define una clase de equivalencia de cada tipo. Es decir, una clase correcta y una incorrecta.
4. Si la condición de entrada consiste en un valor dentro de los de un conjunto, se define una clase de equivalencia de cada tipo. Es decir, se define una clase con un valor dentro del conjunto y otra fuera.

1.5.2.2.1.2 ANÁLISIS DE VALORES LÍMITES

Esta técnica es el complemento de la técnica de partición equivalente, pues en lugar de seleccionar cualquier elemento de la clase de equivalencia la elección de los casos de prueba se lleva en los bordes o límites de las clases. A diferencia de la partición equivalente que se centra únicamente en las condiciones de entrada esta técnica también deriva casos de prueba para el campo de salida.

1. Si una condición de entrada consiste en valores comprendidos dentro de un rango a y b , los casos de prueba se deben diseñar para los valores a y b , además de que deben diseñarse para valores que se encuentren por debajo de a y por encima de b .
2. Si la condición de entrada consiste en un valor numérico, los casos de prueba que se diseñen deben probar valores que estén por debajo y por encima del mínimo y máximo respectivamente.
3. Al aplicar las directrices 1 y 2 a las condiciones de salida, se deben diseñar casos de prueba que muestren las salidas correspondientes al probar los valores máximo y mínimo permitidos en las condiciones de entradas.

1.5.2.2.2 MÉTODO DE CAJA BLANCA

Para aplicar este método se necesita conocer la estructura interna del programa debido a que estas pruebas son derivadas a partir de las especificaciones internas del diseño o el código. Con la utilización de este método los desarrolladores pueden realizar casos de pruebas que ejerciten por lo menos una vez cada uno de los caminos independiente de los módulos probados, ejerciten todas las decisiones lógicas en sus vertientes verdadera y falsa; ejecuten todos los ciclos en sus límites y con sus límites operacionales, y ejerciten las estructuras internas de datos para asegurar su validez. Los casos de prueba intentan demostrar que las funciones del software son operativas y que una entrada adecuada produce una salida correcta.

Pudiera parecer que realizar una prueba de caja blanca muy profunda nos puede llevar a obtener programas totalmente libre de errores pero por desgracia resulta imposible realizar una prueba exhaustiva por problemas logísticos puesto que incluso en pequeños programas el número de caminos independientes resultaría demasiado grande como para poder probarlos todos en un tiempo razonable.

A continuación se describen algunas técnicas muy usadas al realizar pruebas de caja blanca:

1.5.2.2.1 PRUEBA DEL CAMINO BÁSICO

Este tipo de prueba le brinda la posibilidad al equipo de obtener una medida de la complejidad lógica de los algoritmos del software y les permite usar esa medida para guiarse en la definición de un conjunto básico de caminos de ejecución. Es decir, la idea es derivar casos de prueba a partir de un conjunto dado de caminos independientes donde un camino independiente no es más que aquel que introduce por lo menos una sentencia de procesamiento (o valor de condición) que no estaba considerada hasta el momento. Para poder seleccionar los caminos independientes se debe construir el grafo de flujo asociado al procedimiento probado y calcularle la complejidad ciclomática, donde la cantidad de caminos independientes a probar es directamente proporcional a la complejidad ciclomática.

1.5.2.2.2 PRUEBA DE CONDICIONES

La prueba de condiciones es un método de diseño de casos de prueba que ejercita las condiciones lógicas contenidas en el módulo de un programa. Es decir, se desarrollan casos de pruebas para comprobar el cumplimiento o no de cada parte de las condiciones de los procedimientos implementados. Para estas pruebas se necesitan varios casos de prueba para determinar expresiones simples en las condiciones, uno por cada operando lógico o comparación y cada expresión simple debe cumplirse en un caso y en otro no, siendo decisiva en el resultado.

1.6 ESTUDIOS PREVIOS SOBRE VALIDACIÓN Y VERIFICACIÓN DE SOFTWARE.

Mediante la investigación desarrollada para realizar este trabajo, se ha podido constatar la existencia de otras investigaciones dirigidas hacia este sentido llevadas a cabo por profesionales. Entre estos trabajos podemos citar:

1.6.1 LA EVALUACIÓN DEL SOFTWARE. PRODUCTO DE SOFTWARE MÉDICO. GUÍA PARA LOS FABRICANTES EN CUANTO A LA EVALUACIÓN.

Este es un trabajo presentado por J. M. Santos en el II Congreso Latinoamericano de Ingeniería Biomédica, en mayo del 2001 en La Habana. Esta investigación es el resultado de un trabajo multicéntrico y multidisciplinario de un conjunto de especialistas, documentos normativos y regulatorios que abordan la mejor forma de obtener evidencias del cumplimiento de los requisitos establecidos sobre la eficacia, seguridad y por extensión de la calidad del software en general y en particular del software médico. Además en este trabajo se dedican algunos epígrafes a los principales principios de la validación de software y a las actividades de validación en cada una de las fases del ciclo de vida del software. (16)

1.6.2 TOP 10 DE FACTORES QUE OBSTACULIZAN LA MEJORA DE LOS PROCESOS DE VERIFICACIÓN Y VALIDACIÓN EN ORGANIZACIONES INTENSIVAS EN SOFTWARE.

Aquí se hace referencia al trabajo de Javier García, Antonio de Amescua, Manuel Velasco, todos pertenecientes a la Universidad Carlos III de Madrid. A pesar de los beneficios que se pueden obtener de la correcta aplicación de los procesos, técnicas y herramientas de verificación y validación de software, la utilización institucionalizada de los mismos en la industria, sobre todo en las pequeñas y medianas organizaciones de desarrollo de software, no alcanzan los niveles mínimos para el logro de los beneficios potenciales.

En este artículo se presentan los 10 factores más importantes que impiden la correcta aplicación de las prácticas eficientes de verificación y validación de software, a partir de la experiencia recogida en los numerosos programas de mejora de procesos software en los que han participado los autores. (17)

1.6.3 OPTIMIZACIÓN DE LOS ENSAYOS CLÍNICOS DE FÁRMACOS MEDIANTE SIMULACIÓN DE EVENTOS DISCRETOS, SU MODELIZACIÓN, VALIDACIÓN, VERIFICACIÓN Y LA MEJORA DE LA CALIDAD DE SUS DATOS.

Este trabajo es la tesis presentada por Antonio Monleón Getino para optar por el grado de Doctor en Ciencias Biológicas, el cual pertenece al Departamento de Estadística e Investigación Operativa de la Universidad de Barcelona. En dicha investigación se dedica el epígrafe 3.4 a la **VALIDACIÓN, VERIFICACIÓN Y ACREDITACIÓN (VV&A) DE LOS MODELOS DE SIMULACIÓN** en el cual se describen paradigmas de Verificación y Validación de los modelos de simulación, validación de datos y

del modelo conceptual, pruebas de simulación, validación y verificación del modelo computacional y otro grupo de actividades de validación y verificación en modelos de simulación. (18)

1.6.4 IMPORTANCIA DE LAS HERRAMIENTAS DE VALIDACIÓN Y VERIFICACIÓN.

Este es un trabajo de Cristian A. Martínez quien pertenece al Instituto Tecnológico y de Estudios Superiores de Monterrey Campus Puebla. Este artículo describe brevemente técnicas de verificación y validación que se utilizan en el proceso de V&V durante el ciclo de vida de desarrollo de software. Entre las técnicas que se hace alusión en el trabajo están Technical Reviews (Inspecciones Simples), inspecciones técnicas formales y peer reviews aplicadas al peer programming. (19)

1.7 VALIDACIÓN Y VERIFICACIÓN EN MODELOS DE SIMULACIÓN.

Uno de los procesos más complejos e importantes que deben llevar a cabo los desarrolladores de modelos de simulación, es el proceso de validación y verificación. Estos deben dirigir los esfuerzos a satisfacer las expectativas de los usuarios y es por eso que durante todo el desarrollo del producto, el desarrollador debe trabajar lo más cerca posible a los clientes finales en aras de disminuir las inconformidades de estos para con el producto. El objetivo fundamental de la validación en modelos de simulación es:

- Producir un modelo que represente el comportamiento de un sistema real de la forma que brinde un nivel aceptable de credibilidad del modelo para que sea aceptado por los usuarios finales a nivel de toma de decisiones sobre los resultados proporcionados por dicho modelo.

En general, la verificación enfoca el tema de la consistencia interna de un modelo, mientras que la validación está relacionada con la correspondencia entre el modelo y la realidad.

1.7.1 DIFICULTADES DE LA VALIDACIÓN Y VERIFICACIÓN

A la hora de llevar a cabo el proceso de validación, surgen ciertas dificultades entre las que se encuentran:

- La inexistencia de la validación y verificación general: Un modelo determinado se valida con respecto a sus objetivos específicos, por lo que no se puede afirmar que un modelo válido para un propósito lo sea para otros. La realidad es el único “modelo” válido de forma general.

- Inexistencia de elementos del mundo real con los cuales comparar: A partir de sistemas del mundo real se desarrollan nuevas funcionalidades o servicios, donde en algunos casos puede no existir referencia del mundo real con la cual establecer una comparación.
- ¿Cuál es el mundo real?: Cada persona puede tener una visión diferente sobre el mundo real por ejemplo, un empleado del banco puede verlo como un medio para ganar dinero, mientras que un usuario de dicho banco puede considerarlo como un medio de depositar o pedir prestado dinero; esto es un problema cuando se intenta validar modelos pues si la gente tiene diferentes interpretaciones del mundo real, ¿qué interpretación es la adecuada para desarrollar y validar el modelo? Un modelo para una persona puede no serlo para otra.
- Imprecisión de los datos obtenidos del mundo real: A menudo para realizar la validación de un modelo de simulación se hace una comparación del mismo con datos del mundo real, pero estos datos no tienen porque ser siempre precisos; además aunque estos sean fiables se debe recordar que solo son una muestra de la realidad lo que crea cierta imprecisión.
- Insuficiente tiempo para verificar y validar cualquier cosa: En el desarrollo de un modelo de simulación, existen plazos que cumplir, por lo que no se dispondrá del tiempo suficiente para verificar y validar cada aspecto del modelo.

1.7.2 PRINCIPIOS BÁSICOS DE VALIDACIÓN Y VERIFICACIÓN.

Para una correcta aplicación del proceso de validación y verificación en modelos de simulación se deben tener en cuenta los siguientes principios:

- La verificación y validación se deben realizar a través del ciclo completo de un proyecto de simulación.
- El resultado de la V&V de un modelo de simulación no debiera ser considerado como una variable binaria en la que el modelo es completamente correcto o completamente incorrecto.
- Un modelo de simulación se construye con respecto a unos objetivos de estudio y su credibilidad se juzga respecto a esos objetivos.
- El proceso de V&V del modelo de simulación es difícil y requiere creatividad y perspicacia.
- La credibilidad solo se puede afirmar para las condiciones en las que se ha probado el modelo.
- No es posible la prueba completa de un modelo de simulación.
- La V&V deben ser planificadas y documentadas.
- Los errores se deben detectar tan pronto como sea posible en el ciclo de vida.
- Pasar correctamente las pruebas en cada módulo no implica la credibilidad total del modelo.

- La validez de un modelo de simulación no garantiza la credibilidad y aceptación de los resultados de la simulación.

1.8 SITUACIÓN EN LATINOAMÉRICA RESPECTO A CMMI

Los países latinoamericanos no muestran grandes resultados en el mercado internacional del software y una de las causas fundamentales de este fenómeno se debe al escaso número de empresas desarrolladoras de software que han logrado certificarse con los principales estándares y normas de calidad existentes, con los cuales aumentaría el nivel de seguridad y confianza por parte de los clientes con relación a sus productos. En el caso de CMMI, convertido en el estándar más utilizado a nivel mundial para la mejora de los procesos de software y a su vez el estándar al cual están dirigidos los mayores esfuerzos de las instituciones de software, en aras de lograr una certificación de este modelo; los países de esta área presentan un bajo índice de empresas certificadas con dicho estándar y para tener una mayor comprensión de esta situación se referencia al trabajo **Comparación de las Iniciativas Latinoamericanas para Mejorar la Industria del Software** (20) presentado por el Dr. Marcelo Jenkins en el evento Informática 2007 en La Habana, en el cual se recogen los siguientes datos:

- En **Brasil** hasta septiembre del 2006 se tenían 39 empresas evaluadas de CMMI.
- En **México** para la misma fecha, septiembre del 2006, se contaba con menos de 10 empresas evaluadas de CMMI.
- El caso de **Chile** es similar al caso de México, para septiembre del 2006 también contaba con menos de 10 empresas evaluadas de CMMI.
- **Perú** presentaba para la misma fecha una situación mucho más crítica al no tener ninguna empresa certificada con este modelo.
- **Costa Rica** se encontraba en las mismas condiciones de Perú.
- **Ecuador** al igual que Costa Rica y Perú no presentaba ninguna empresa evaluada por CMMI pero este dato consta desde marzo del 2006.
- **Argentina** presentaba para septiembre del 2006 unas 15 empresas certificadas por CMMI.

Las empresas latinoamericanas desarrolladoras de software enfrascadas en lograr una mayor inserción en el mercado de software, continúan esforzándose para acreditarse con el máximo estándar internacional que se otorga a los desarrolladores de software, CMMI; estos esfuerzos quedan reflejados en los resultados alcanzados en algunos de estos países como es el caso de:

- Argentina, donde la compañía Snoop Consulting consiguió la acreditación de Nivel 2 de CMMI, esta empresa especializada en servicios sobre nuevas tecnologías de información logró con esto un nuevo posicionamiento y una nueva consolidación en el mercado. Además, la compañía logró cumplimentar casi el 50% de las prácticas necesarias para obtener el nivel 3 de certificación. “Esto demuestra que Snoop sigue fiel a su proceso de mejora continua”, señaló Sergio Candelo, Director de Operaciones de Snoop Consulting (21).
- En México, gracias al compromiso que la empresa TI-M ha hecho con sus clientes y con la calidad, se posiciona como la primera empresa en aprobar exitosamente la evaluación CMMI® nivel 3 en el norte de México, demostrando una vez más la alta capacidad técnica y humana de todo su personal (22). Otro de los resultados obtenidos en este país es la certificación en CMMI nivel 2 de la Empresa Innox (23).
- En Perú, la empresa GMD, especializada en servicios de tecnologías de la información, obtuvo la certificación CMMI, convirtiéndose en la primera compañía nacional en obtener un reconocimiento de esta categoría que la coloca además en el nivel de las más importantes transnacionales del sector. (24)
- En Venezuela, DBAccess superó con éxito la evaluación a sus procesos de desarrollo de software, según el reconocido modelo de calidad internacional CMMI, lo que demuestra el alto grado de elaboración que tienen sus soluciones, y constituye el primer paso para que obtenga la certificación CMMI Nivel 3. Cabe resaltar que es la primera vez que se realiza este diagnóstico en este país, y que DBAccess es la única organización venezolana en vías de alcanzar la certificación CMMI Nivel 3 (25).

CAPÍTULO 2: DIAGNÓSTICO DE LA SITUACIÓN ACTUAL

INTRODUCCIÓN

La facultad 5 de la UCI se encuentra actualmente vinculada a la producción de software, en dicha facultad se pueden identificar dos polos productivos los cuales son automatización y realidad virtual, pero para este trabajo solo se tendrá en cuenta este último para el cual se tienen definidas las siguientes líneas de investigación:

1. Simulación Basada en Física
2. Detección Colisiones
3. Animación 3D
4. Inteligencia Artificial aplicada a RV
5. Iluminación y Sombras
6. Modelación Geométrica
7. Reconstrucción tridimensional
8. Gráficos basados en Puntos
9. Diseño 3D
10. IGSW aplicada al desarrollo de juegos y aplicaciones de RV
11. Programación del Hardware Gráfico (shaders)
12. Optimización de visualización en ambientes 3D
13. Aplicaciones Gráficas para la WEB
14. Sonido 3D

Como complemento de estas líneas de investigación del polo de realidad virtual se trabaja sobre las siguientes líneas de desarrollo:

- Video-Juegos de entretenimiento.
- Simuladores para Entrenamiento – Evaluación.
 - Conducción de vehículos (autos, camiones, bus, tren, embarcaciones marítimas, maquinaria industrial).
- Maquetas virtuales.
- Laboratorios Virtuales.

- Diseño y realización de videos y Animaciones 3D.
- Aplicaciones de realidad Virtual para la salud.
 - Ejercicios de tratamiento de fobias.
 - Software para la Rehabilitación.
 - Simuladores de Cirugía de mínimo acceso.
 - Video-Juegos para el tratamiento de discapacidades.
- Evaluadores teóricos basados en situaciones 3D.
- Herramientas para el desarrollo de SRV.

Esta facultad cuenta con 15 proyectos productivos de los cuales 9 proyectos pertenecen a la rama de la realidad virtual, estos últimos representan el 60 % del total de los proyectos de la facultad, los mismos se encuentran a continuación:

- Laboratorios Virtuales
- Compilación de juegos
- Diseño 3D
- Simulador Quirúrgico
- Juegos CNEURO
- Juegos de Consola
- Rehabilitación
- Interacción de Elementos Virtuales
- Herramientas de Desarrollo para Sistemas de Realidad Virtual

Como resultado del trabajo de estos proyectos productivos en estos momentos se trabaja sobre los siguientes productos:

- Juegos de MENPET.
- Juego Rápido y Curioso v.1.0.
- Juego CNEURO
- Herramienta SceneToolKit versión 4.0.
- Producto de Rehabilitación.
- Software del Simulador Quirúrgico

Para el desarrollo de este trabajo se realizó un estudio con el objetivo de obtener la situación real que presentan cada uno de los proyectos de realidad virtual en cuanto a lo que validación y verificación se refiere, para esto se tomó como población a estudiar precisamente a los proyectos asociados a esta rama y la muestra de la población establecida para la investigación fue de 6 de los proyectos de este tipo existentes en la facultad. A continuación se muestran los nombres de todos los proyectos vinculados a este estudio:

- Herramientas de desarrollo para sistemas de realidad virtual.
- Simulador Quirúrgico.
- Laboratorios Virtuales.
- Diseño 3D.
- Juegos CNEURO.
- Juegos de Consola.

Como parte del trabajo realizado para recoger toda la información referente a la validación y verificación en estos proyectos fue elaborada una lista de chequeo (Ver Anexo 8), en la misma se abordan fundamentalmente las metas y prácticas propuestas por el modelo CMMI para las áreas de procesos validación y verificación respectivamente. El resultado arrojado por la aplicación de esta lista de chequeo se presenta en los próximos epígrafes.

2.1 RESULTADO DE LAS ENTREVISTAS A LOS PROYECTOS.

2.1.1 HERRAMIENTAS DE DESARROLLO PARA SISTEMAS DE REALIDAD VIRTUAL

El líder de este proyecto es el ingeniero Yanoski Rogelio Camacho Román y el objetivo para el cual fue concebido el mismo es el desarrollo de herramientas gráficas que serán utilizadas para producir sistemas de realidad virtual, se define inicialmente la posibilidad de que estas herramientas sean utilizadas por todos los proyectos de Realidad Virtual de la Facultad 5, existe además la posibilidad futura de comercializar las herramientas obtenidas. El producto en el cual se trabaja en estos momentos es SceneToolKit: Alias de la herramienta desarrollada por el proyecto (Herramienta Básica para Desarrollo de Sistemas de Realidad Virtual).

Al momento de la entrevista este proyecto no estaba regido por ninguno de los modelos o normas de calidad existentes. En lo que respecta al proceso de validación y verificación aunque tienen bien claro la importancia de desarrollar este proceso, no tienen establecido los objetivo para el mismo, además no cuentan con ningún miembro del equipo de trabajo que sea responsable de dirigir este proceso en

su totalidad aunque tienen definidos roles que se encargan de varias de las actividades de V&V que se llevan a cabo en el proyecto, estos roles son probador, diseñador de casos de prueba, auditor de calidad, revisor técnico formal y un planificador de calidad. Entre las actividades de V&V realizadas se encuentran:

- Identificar técnicas apropiadas, herramientas e instrucciones para la implementación de las pruebas.
- Diseñar los casos de pruebas.
- Definir listas de chequeo para las pruebas.
- Ejecutar las pruebas diseñadas.
- Registrar los resultados.
- Elaborar los criterios de la auditoría.
- Ejecutar las auditorías planificadas.
- Registrar los resultados de las auditorías.
- Elaborar listas de chequeo para las revisiones.
- Registrar los resultados de las revisiones.
- Planificar el proceso de Aseguramiento de la Calidad en el proyecto (Plan de Calidad).
- Realizar el Plan de Prueba, de Revisión y Auditoría de cada iteración.
- Realizar el Resumen de Evaluación de Pruebas.

En la Plantilla DCS - Roles y responsabilidades v1.0 del expediente del proyecto se puede ver con mayor profundidad cual es el rol responsable de cada una de las actividades antes mencionadas.

Además de esto, se debe señalar que con el objetivo de gestionar el proyecto se usa la herramienta Trac para la asignación de tareas y responsabilidades a los miembros del equipo de trabajo.

Este equipo de desarrollo de software no tiene establecido ningún plan para realizar la V&V, aunque asegura realizar pruebas de caja blanca y caja negra cada vez que terminan algún producto o

componentes del mismo, pero no cuentan con un plan aseguramiento de la calidad, tampoco con un plan de pruebas y por último no tienen diseñado los casos de pruebas para los casos de uso (CU).

Al proyecto se le han realizado revisiones del expediente del proyecto por parte del grupo de calidad de la facultad 5. Estas revisiones han arrojado no conformidades a lo largo del proceso de desarrollo del software de las cuales se pueden ver como las mas significativas para el proceso de V&V el déficit de la documentación del proyecto, inexistencia de un plan de calidad, etc. Se debe apuntar que luego de estas revisiones en el proyecto se ha dado respuesta a la mayor parte de las no conformidades por lo que ha disminuido considerablemente el problema de la documentación, aunque todavía persisten los problemas con el plan de aseguramiento de la calidad y demás artefactos relacionados con este. Las no conformidades y respuestas a las mismas dentro del proyecto se pueden comprobar en las planillas No_Conformidades Detectadas y Respuestas a No Conformidades Detectadas que se encuentran dentro del expediente del proyecto.

El proyecto cuenta con todos los recursos adecuados para realizar el proceso, lo cual consta en las plantillas pertenecientes a la parte de recursos que a su vez tributan a la gestión del proyecto. En las plantillas antes mencionadas se puede comprobar como el proyecto cuenta con los recursos humanos necesarios, a los cuales se les tienen asignados sus roles y responsabilidades aunque se debe precisar que aún no se tienen identificados todos los roles necesarios para el proceso de V&V. Por otra parte se debe tener presente que el proyecto tiene establecido un plan de capacitación para el equipo de trabajo para el cual cuenta actualmente con todos los recursos materiales de los que se precisa para poder impartirlos, en la mayoría de los cursos son:

- Un aula con computadora y pizarra. Plumones y borrador.
- Posible documentación digital sobre el contenido a impartir, para estudio independiente.
- Computadoras del laboratorio con herramientas instaladas y SO Linux.

Como se ha mencionado anteriormente aunque en el proyecto se tiene establecido un plan de capacitación al personal encargado del proceso, se debe indicar que en el mismo solo se ha realizado hasta el momento un curso sobre la herramienta Trac, usada para la gestión de asignación de tareas a las personas involucradas en el proceso de V&V, esto se puede comprobar en la Plantilla DCS - Plan de capacitación v1.0 del proyecto.

2.1.2 SIMULADOR QUIRÚRGICO.

El líder de este proyecto es el licenciado Juan Manuel Medero. El propósito de este proyecto es la creación de un prototipo de Simulador Cubano para la enseñanza de la cirugía laparoscópica y la endoscopia diagnóstica y terapéutica del tracto gastrointestinal. El proyecto genera una Maqueta de Habilidades Básicas en sus versiones 1.0 y 2.0. El producto principal es un Simulador Quirúrgico (KHEIPROS) también en sus versiones 1.0 y 2.0.

No se rige por ninguno de los modelos de calidad existentes, aunque en las plantillas del expediente del proyecto se hace referencia a la ISO 9126, ISO 12207, ISO 12119, IEEE 1008, IEEE 1028, el líder del proyecto asegura que realmente no se hacía uso de ninguna de las normas antes mencionados. Al realizar esta entrevista en dicho proyecto no se tenían establecidos objetivos claves para el proceso de validación y verificación. En lo que concierne a responsables del proceso, se cuenta con la definición de algunos roles como, el diseñador de casos de prueba, auditor de la calidad, revisor técnico formal y planificador de la calidad los cuales tienen entre sus responsabilidades

Diseñador de casos de prueba

- Identificar técnicas apropiadas, herramientas e instrucciones para la implementación de las pruebas.
- Diseñar los casos de pruebas.
- Definir listas de chequeo para las pruebas.

Auditor de la calidad

- Elaborar los criterios de la auditoría.
- Ejecutar las auditorías planificadas.
- Registrar los resultados de las auditorías.

Revisor técnico formal

- Elabora lista de chequeo para las revisiones.
- Revisa todos los artefactos que se generan en el proyecto.
- Registrar los resultados de las revisiones.

Planificador de la calidad

- Planificar el proceso de Aseguramiento de la Calidad en el proyecto (Plan de Calidad).
- Realizar el Plan de Prueba, de Revisión y Auditoría de cada iteración.
- Coordinar el proceso de recopilación, análisis y reporte de las estadísticas de calidad (Plan de mediciones).
- Hacerle seguimiento a estos planes.
- Guiar las revisiones técnicas formales.
- Guiar las pruebas que se realicen.
- Guiar las auditorías que se realice.
- Manejar todo lo relacionado con los riesgos de calidad (Lista de Riesgos).
- Realizar el Resumen de Evaluación de Pruebas.

Como parte de la planificación del proceso se tienen definido un plan de pruebas que abarca fundamentalmente las pruebas de caja negra donde los CP se diseñaron considerando exclusivamente las entradas y salidas del sistema, sin preocuparse por la estructura interna del mismo. Dentro de los elementos indispensables para una correcta realización de las pruebas tienen definido un entorno de pruebas, el mismo se describe a continuación:

Las computadoras solicitadas deben satisfacer como mínimo las siguientes características técnicas:

- Microprocesador: Pentium 4 o superior.
- Memoria: 256 MB (En el caso de la empleada para corrección de no conformidades debe ser de 1GB).
- Monitor: SVGA de 15 pulgadas.
- Disco Duro: 10 GB o superior.
- Tarjeta gráfica 256 MB (NVidia o una de iguales prestaciones).

El proceso de pruebas que se usa en el proyecto consta de tres actividades fundamentales:

1. Preparar la ejecución de las pruebas, con esta actividad se trata de asegurarse de la existencia de todos los recursos tanto humanos como materiales previstos en el plan de pruebas, asignación de responsabilidades y preparación de la documentación etc.
2. Medición, proceso mediante el cual se registran los resultados de las pruebas y se evalúa el progreso de estas.
3. Evaluación, se valoran los resultados de las pruebas en cada iteración, se analizan las tendencias de las no conformidades encontradas, se elabora el informe de pruebas y se realiza la reunión de cierre.

Como complemento del plan de pruebas se tiene definida la plantilla Descripción de los Casos de Prueba en la cual se recogen los CP que se le aplicaran a cada funcionalidad a probar y para cada uno de estos se tiene una descripción general, pruebas que se realizarán, descripción de la funcionalidad, flujo central, condiciones de ejecución e iteraciones.

Al proyecto se le han realizado revisiones, ejemplo de esto es la revisión técnica formal por parte el grupo de calidad de la facultad, dicha revisión fue dirigida en su totalidad a las planillas que conforman el expediente del proyecto, otra de las revisiones recibidas en el proyecto fue una realizada por la dirección central de calidad de la UCI.

Por motivo de que el proyecto se encuentra durante la fase de desarrollo de la primera de sus cuatro etapas planificadas no se ha realizado el proceso de validación y verificación con la participación de todos los stakeholders, para ser más específicos no se ha realizado en presencia de los clientes.

Se puede afirmar que el proceso es institucionalizado como un proceso gestionado de manera parcial, pues aunque no en su totalidad, se realizan y documentan algunas de las prácticas requeridas para la gestión del proceso de validación y verificación. Entre las prácticas antes mencionadas se pueden encontrar la impartición de cursos de capacitación a los involucrados en el proceso de V&V como el de la herramienta Trac usada para la asignación de responsabilidades a los miembros del proyecto, además del curso sobre la herramienta Subversión usada para el control de versiones del proyecto. Formando parte de la gestión del proceso se tiene definido un plan de aseguramiento de la calidad donde se describen un conjunto de actividades de V&V entre las que encontramos fundamentalmente:

- Las auditorías de la arquitectura, gestión de la configuración, sistema de la calidad.
- Revisiones de la documentación (RTF), el diseño, etc.

- Realización de pruebas de caja blanca y caja negra.

Se ha de señalar que las prácticas de las áreas de procesos validación y verificación que han sido realizados en este proyecto se encuentran en su mayoría documentadas.

2.1.3 JUEGOS DE CONSOLA

El líder de este proyecto es el ingeniero Yoander Cabrera Díaz y el propósito del mismo es como su nombre indica el desarrollo de juegos, en estos momentos se trabaja en Rápido y Curioso el cual es un juego instructivo 3D de conducción de automóviles para niños en el rango de edad de 8 a 12 años que se va a desplegar en parque de diversiones en la capital a solicitud del consejo de estado.

Con una entrevista realizada al jefe de este proyecto se pudo constatar que este no esta regido por ningún modelo o estándar de calidad existente en industria de software, pero a pesar de esto en su expediente de proyecto referencian algunos modelos como CMMI, la ISO 9000:2000, etc. Además se pudo comprobar la inexistencia de objetivos para el proceso de validación y verificación, para llevar a cabo parte del proceso dentro del proyecto se tienen definidos los roles de analista de pruebas y auditor de calidad, estos roles se encuentran descritos en la Plantilla DCS - Roles y responsabilidades v1.0 del expediente del proyecto.

Para realizar la validación y verificación no cuentan con un plan para la correcta gestión de los procesos según el modelo CMMI, aunque se debe destacar la existencia de un plan para el aseguramiento de la calidad (Plan de Aseguramiento de la Calidad v1.0) en el cual se recogen varias actividades incluidas en los procesos entre las que podemos encontrar:

- Auditoría de la arquitectura (Revisión preliminar del diseño).
- Revisión del diseño (Revisión de diseño crítico)
- Pruebas de caja blanca.
- Pruebas de caja negra.
- Revisión de la documentación (Revisión Técnica formal)
- Auditoría al Sistema de Gestión de la Calidad.
- Revisiones Post-Morten

Para dar cumplimiento a estas actividades se tienen asignadas los responsables de cada una de ellas entre los que encontramos fundamentalmente al planificador o responsable de la calidad, jefes de equipos, líder del proyecto y el asesor de calidad de la Facultad 5. Según se registra en esta plantilla

no se tiene definido un plan de capacitación o entrenamiento para las personas involucradas en este proceso, esto también puede comprobarse en la Plantilla DCS - Plan de capacitación v1.0 en la cual se recogen todas las actividades de capacitación impartidas a los miembros del proyecto.

Como parte de la V&V también se cuenta con un plan de pruebas (Ver Plantilla DCS - Plan de Pruebas v1.0) en el cual se especifican los integrantes del equipo de pruebas que en este caso esta conformado por un diseñador de pruebas y dos probadores los cuales están encargados de realizar las pruebas de caja negra donde los casos de prueba se diseñan considerando exclusivamente las entradas y salidas del sistema, sin preocuparse por la estructura interna del mismo. Como eslabón indispensable para las pruebas se tiene establecido un entorno en el cual se recogen las principales especificaciones que requiere el proceso, entre estas se pueden encontrar; que las computadoras solicitadas deben satisfacer como mínimo las siguientes características técnicas:

- Microprocesador: Pentium 4 o superior.
- Memoria: 256 MB.
- Monitor: SVGA de 15 pulgadas.
- Disco Duro: 10 GB o superior.
- Tarjeta gráfica 256 MB (NVidia, ATI o una de iguales prestaciones).

Como complemento de las pruebas se tiene definido un procedimiento para realizarlas, el consta de tres pasos fundamentales:

1. Preparar la ejecución de las pruebas
2. Medición
3. Evaluación

Además en este plan se especifica el criterio de término de las pruebas, recursos humanos y materiales necesarios y se hace referencia a la Plantilla DCS - Diseño Casos de Prueba v1.0 en la cual se encuentran diseñados los casos de pruebas de caja negra que como se ha planteado anteriormente enfatizarán en los requisitos funcionales del juego y para ellos se definieron las clases válidas e inválidas para cada uno de ellos; se debe señalar que a pesar de esto no se cuenta con procedimientos para llevar a cabo cada uno de estos casos de prueba.

Para seleccionar los productos o componentes del proyecto a validar y verificar no se tiene establecido ningún criterio en específico, pues cada vez que se termina alguno de estos se le aplica una revisión en aras de asegurarse de la calidad con que cuenta este.

El proyecto hasta el momento ha recibido revisiones solo por parte del proyecto de calidad de la facultad, estas revisiones han centrado su atención fundamentalmente a los aspectos concernientes al expediente de proyecto.

2.1.4 DISEÑO 3D

El líder del proyecto es el ingeniero Gadied Alejandro Carrero Sotolongo y el objetivo del mismo es la modelación objetos en tres dimensiones a solicitud de los restantes proyectos de realidad virtual existentes en la facultad 5.

El proyecto no se acata a ningún modelo o norma de calidad existentes. En cuanto al proceso de validación y verificación se pudo comprobar que no tienen establecido objetivos para este. Los responsables identificados para realizar con efectividad las actividades del proceso que se llevan a cabo en el proyecto son el propio jefe de proyecto y los jefes de áreas de Modelación y Animación en las cuales está estructurado el proyecto.

Aunque en el proyecto existe una costumbre por llamarlo de alguna manera de validar y verificar los productos de trabajo, no cuentan con un plan para guiar el mismo. Es necesario precisar que a pesar de esto el proyecto cuenta con los recursos necesarios para desarrollar el proceso.

Los procedimientos usados como parte del proceso son la prueba directa en las aplicaciones para las cuales a sido modelado los diferentes objetos y la entrega de los productos a los proyectos que los hayan solicitado de forma tal que estos prueben y emitan criterios sobre el producto, a parte de esto último no existe otro tipo de participación de los stakeholders durante el proceso.

De las prácticas propuestas por CMMI para validación y verificación que se les ha dado cumplimiento en este proyecto, no se cuenta con la documentación que debiera existir para respaldar este proceso.

2.1.5 JUEGOS CNEURO

El líder del proyecto es el ingeniero Yulier Casas Estrada y el mismo va orientado directamente a los niños que padecen la enfermedad de Discalculia y a los especialistas que atienden este padecimiento. El producto final que se obtenga será utilizado primeramente por todos los niños de 6to grado afectados con la Discalculia en el país, aunque no se descarta una futura comercialización del mismo.

Actualmente se está trabajando sobre el producto Venganza en su versión 1.0, para el se espera tener una primera entrega con 3 niveles del juego terminado en septiembre del 2008.

Este proyecto no se guía por ningún modelo o estándar de calidad de software existente. En cuanto al proceso de validación y verificación se puede señalar que no están definidos los objetivos concretos y documentados para el mismo. Como parte de los responsables para el proceso cuentan en el proyecto dos estudiantes encargados del proceso de aseguramiento de la calidad los cuales cubren algunas de las actividades necesarias para validar y verificar, además se piensa incorporar un nuevo rol en el proyecto que se encargue de las actividades de calidad de software, estos estudiantes tienen entre sus responsabilidades las siguientes tareas:

- Planificar el proceso de Aseguramiento de la Calidad en el proyecto (Plan de Calidad).
- Realizar el Plan de Prueba, de Revisión y Auditoría de cada iteración.
- Coordinar el proceso de recopilación, análisis y reporte de las estadísticas de calidad (Plan de mediciones).
- Hacerle seguimiento a estos planes.
- Guiar las revisiones técnicas formales.
- Guiar las pruebas que se realicen.
- Guiar las auditorías que se realice.
- Manejar todo lo relacionado con los riesgos de calidad (Lista de Riesgos).
- Realizar el Resumen de Evaluación
- Identificar técnicas apropiadas, herramientas e instrucciones para la implementación de las pruebas.
- Diseñar los casos de pruebas.
- Definir listas de chequeo para las pruebas.

Lo antes mencionado consta así en la Plantilla Roles y responsabilidades del expediente del proyecto.

El proyecto ha sido revisado por parte del grupo de calidad de software de la facultad, centrando su atención en la parte de la documentación, específicamente en lo referente al Expediente de Proyecto.

Además el proyecto recibió una auditoría de calidad realizada por la dirección central de calidad de software de la universidad.

En el proyecto se han impartido varios cursos de capacitación, de los cuales se consideran importantes para el proceso de V&V, el curso acerca de la herramienta Trac usada en el proyecto para gestionar la asignación de tareas a los miembros del equipo y el curso sobre la herramienta Subversión usada para el control de versiones del proyecto. Independientemente de que se hayan impartido estos cursos, se considera que el plan de capacitación es insuficiente con respecto a las necesidades reales de las personas involucradas en el proceso de V&V. Para una mayor comprensión de lo planteado en este sentido se debe revisar el Plan de Capacitación del proyecto.

Se cuenta con un Plan de Aseguramiento de la Calidad en el cual se recogen varias actividades de V&V entre las que se encuentran:

- Auditoría de la arquitectura (Revisión preliminar del diseño).
- Revisión del diseño (Revisión de diseño crítico).
- Pruebas de caja blanca.
- Pruebas de caja negra.
- Auditoría a la Gestión de la configuración.
- Auditoría de la configuración física.
- Revisión de la documentación (Revisión Técnica formal).
- Auditoría al proyecto.
- Auditoría al Sistema de Gestión de la Calidad.

Es preciso añadir a esto que aunque se tienen identificadas cuales son las actividades a realizar aun no tienen establecido un cronograma donde se recoja cual será la fecha de cada una de las auditorías, revisiones y demás actividades; tampoco se cuenta con una organización y asignación de responsabilidades donde explique cuales van a ser las personas involucradas en cada una de las actividades además de cuales van ser las tareas a cumplir por cada una de ellas.

La actividad de V&V que cuenta con una mayor preparación en el proyecto son las pruebas de caja negra para lo cual se cuenta con la plantilla Diseño de Casos de Prueba en la cual se describen los caso de prueba (CP) que se realizaran con el fin de probar el mismo, en esta planilla se indica para cada CP una descripción general, descripción de la funcionalidad, flujo central, condiciones de ejecución entre otras características de cada uno de ellos.

2.1.6 LABORATORIOS VIRTUALES

El líder del proyecto es el licenciado Luis Gabriel Viciado Carabaloso en el mismo se tienen establecidos dos objetivos fundamentales consistentes en la creación de un Entorno Virtual de Aprendizaje para laboratorios y la generación de simulaciones de entornos para laboratorios. En el momento de esta entrevista se encuentra en la fase de inicio en la cual se trabaja en la propuesta de guiones para el proyecto, teniendo como centro de atención los aspectos comunes existentes entre laboratorios de Química, Física y Biología desarrollando los módulos pertinentes.

En el proyecto no se tienen establecidos objetivos concretos para el proceso de validación y verificación y el rol encargado de todo lo concerniente a la calidad de software es el asegurador de calidad. Se conoce además que no se rige por ningún modelo o norma de calidad existente, aunque siguen las pautas fijadas para software de realidad virtual en la facultad 5.

No se cuenta con ningún plan para validar y verificar y por consiguiente no se ha realizado de manera correcta este proceso. Aunque en la actualidad no se realiza una gestión del proceso, es preciso señalar que se trabaja por institucionalizar la gestión del proyecto de forma general mediante la utilización de la herramienta RedMine la cual permite mediante la creación de tickets la asignación de tareas a todos los miembros del proyecto.

Los lenguajes de programación utilizados en el desarrollo del software del proyecto son C++ para las aplicaciones de escritorio y PHP con Actionscript para las aplicaciones Web.

2.2 RESULTADOS GENERALES DE LOS PROYECTOS DE RV

Después de la revisión de los elementos arrojados por las entrevistas a los líderes de los proyectos de realidad virtual de la facultad 5 se puede resumir de forma general cual es la situación actual existente en estos.

El primer elemento significativo encontrado en todos los proyectos entrevistados fue la inexistencia de modelos o estándares de calidad de software como guía del proceso de desarrollo de software. Se debe aclarar en este punto que en la documentación del proyecto aparecen referenciados varios estándares y modelos como:

Modelo o estándar	Descripción
IEEE 730	Estándar para el plan de aseguramiento de calidad de Software
ISO 12207	Ciclo de vida del Software
CMMI	Modelo de Madures de las Capacidades Integrado

Pero a pesar de contar con estas referencias a los modelos, no se puede dejar de destacar el desconocimiento de estos por parte de los desarrolladores, incluyendo a los líderes de estos, los cuales en el 100% de los casos de los proyectos entrevistados, expresaron que sus equipos de trabajo no estaban regidos por ningún modelo o estándar de calidad.

Relacionado con el proceso de V&V se debe señalar el desconocimiento del término existente por parte de los integrantes del equipo. A pesar de esto en la mayoría de los proyectos tienen definidos roles que aunque no son suficientes para llevar a cabo todas las actividades de V&V se encargan de realizar algunas indispensables para el proceso. Los roles establecidos en los proyectos son:

- Auditor de la calidad.
- Diseñador de casos de prueba.
- Probador.
- Revisor Técnico Formal.
- Planificador de calidad.
- Analista de pruebas.
- Asegurador de la calidad.

Cabe destacar que no todos los proyectos cuentan con todos estos roles, por lo que para realizar la V&V no existen los responsables suficientes para llevar a cabo este proceso.

El 50 % de los proyectos entrevistados cuentan con un plan de aseguramiento de la calidad, en el cual se listan una serie de actividades a realizar, entre las que encontramos fundamentalmente:

- Las auditorías de la arquitectura, gestión de la configuración, sistema de la calidad.
- Revisiones de la documentación (RTF), el diseño, etc.
- Realización de pruebas de caja blanca y caja negra.

A pesar de tener estas actividades definidas, solo se tiene constancia de que se lleven a cabo las revisiones de la documentación, las auditorías al sistema de la calidad y la realización de las pruebas de caja negra, para estas últimas se tienen definidos planes de prueba en los cuales están registrados el equipo responsable de llevar a cabo las pruebas, el entorno de pruebas, la descripción de los requisitos y casos de uso a probar y las estrategias de prueba que se utilizan; adjunto a esto se encuentra la plantilla Diseño de Casos de Prueba donde para cada CU se describe las pruebas a realizar, el flujo central y las condiciones de ejecución entre otros elementos.

A cuatro de los proyectos entrevistados se les ha hecho revisiones, principalmente por parte del grupo de Calidad de Software de la Facultad, el cual ha hecho mayor énfasis en la documentación del Expediente del Proyecto.

Dos de los proyectos estudiados fueron objetos de revisiones llevadas a cabo por la dirección central de calidad de software de la UCI, en la cual se encontraron fundamentalmente problemas con la documentación y migración de esta al expediente del proyecto.

Como parte del plan de capacitación del personal involucrado con el proceso de V&V, se han impartido en tres de estos proyectos cursos sobre la herramienta Trac, usada para la gestión y asignación de tareas a los integrantes del proyecto. Además se usa en 2 de estos la herramienta Subversión, la cual es para llevar el control de versiones de los productos. En uno de los proyectos de realidad virtual de la facultad se hace uso de la herramienta RedMine, la cual se utiliza para la asignación de tareas a sus integrantes, pero en este caso no se han realizado cursos de capacitación.

Los proyectos cuentan tanto con los recursos humanos como materiales para llevar a cabo el proceso, puesto que los recursos humanos constituyen un grupo de profesores y estudiantes de la Facultad y los recursos materiales son fundamentalmente computadoras con las siguientes características:

- Microprocesador: Pentium 4 o superior.
- Memoria: 256 MB

- Monitor: SVGA de 15 pulgadas.
- Disco Duro: 10 GB o superior.
- Tarjeta gráfica 256 MB (NVidia, ATI o una de iguales prestaciones).

De manera general se puede afirmar que aunque actualmente en los proyectos de RV de la facultad 5 se trabaja en algunas de las metas y prácticas propuestas por el modelo CCMI para el proceso de V&V, todavía se debe trabajar mucho para lograr que este proceso se realice con calidad, pues aún existen problemas en una serie de aspectos que resultan de gran importancia para conseguir el correcto cumplimiento de los objetivos del proceso.

CAPÍTULO 3: PROPUESTA DE SOLUCIÓN.

INTRODUCCIÓN

Independientemente de que el modelo CMMI propone dos áreas de procesos diferentes para la validación y la verificación, y que los objetivos de ambas no son los mismos; para dar solución a los problemas presentados en los proyectos de realidad virtual de la facultad 5 de la UCI a la hora de realizar estas actividades, se trabajará como un único proceso denominado “Proceso de Validación y Verificación”. Las dos causas fundamentales que conllevaron a tomar esta decisión son, primeramente la concordancia con muchos autores de que estas actividades se encuentran estrechamente relacionadas por lo que la mejor opción es darle tratamiento como dos procesos concurrentes y la segunda de las razones es la similitud existente entre las metas generales y específicas de estas áreas de procesos descritas en el modelo.

3.1 PREPARAR LA V&V.

3.1.1 SELECCIONAR LOS PRODUCTOS.

Para seleccionar los productos de trabajo se debe tener presente en cada proyecto el momento actual dentro del ciclo de vida del software (CVS) en el cual se encuentra el producto. Esto se vuelve un factor determinante en este sentido debido a que si el producto en general se encuentra en las primeras fases de CVS definidas en el Proceso Unificado de Software (RUP), al mencionar primeras fases se refiere a la fase de inicio y la fase de elaboración en sus primeras iteraciones; como no se cuenta con versiones operativas del producto, el centro de atención para validar y verificar recae principalmente sobre la documentación de los artefactos generados hasta el momento. Luego de la aparición de los primeros prototipos y versiones ejecutables del producto, sin descuidar la parte de la documentación, se debe comenzar la aplicación del proceso sobre estas versiones en aras de encontrar los posibles defectos desde un tiempo temprano en el CVS del software.

En el caso de la documentación se debe profundizar fundamentalmente en las planillas estandarizadas en la universidad para el expediente de proyecto, las cuales se encuentran divididas en cuatro grupos los cuales se presentan a continuación:

I. Ingeniería

1.1 Requisitos

1.1.1 Plantilla DCS - Diagrama de Proceso - Nombre del Proceso

1.1.2 Plantilla DCS - Evaluación de Áreas de la Organización

1.1.3 Plantilla DCS - Plan de gestión de requisitos

1.1.4 Plantilla DCS - Modelo de Casos de uso del sistema

1.1.5 Plantilla DCS - Modelo del Negocio

1.1.6 Plantilla DCS - Modelo del Dominio

1.2 Arquitectura y diseño

1.2.1 Plantilla DCS - Documento de Arquitectura de Software

1.2.2 Plantilla DCS - Modelo de Diseño

1.3 Implementación y pruebas

1.3.1 Código fuente

1.3.2 Manual de usuario

1.3.3 Plantilla DCS - Diseño casos de prueba

1.3.4 Plantilla DCS - Plan de pruebas

1.4 Despliegue e instalación

1.4.1 Plantilla DCS - Modelo de Despliegue

II. Gestión de proyecto

2.1 Plan del proyecto

2.1.1 Plantilla DCS - Plan Desarrollo de Software

2.2 Riesgos

2.2.1 Plantilla DCS - Lista de riesgos

2.2.2 Plantilla DCS - Plan Mitigación de Riesgos

2.3 Recursos

2.3.1 Plantilla DCS - Ambiente de desarrollo

2.3.2 Plantilla DCS - Plan de capacitación

2.3.3 Plantilla DCS - Roles y responsabilidad

2.3.4 Asistencia y evaluación a cursos de capacitación

2.3.5 Cuestionario diagnóstico para nuevos miembros del proyecto

2.3.6 Historial del proyecto

2.4 Acuerdos de trabajo

2.4.1 Plantilla DCS - Documento Visión

2.5 Informes

2.5.1 Plantilla DCS - Diagnóstico

2.6 Reuniones

2.6.1 Plantilla DCS - Minuta de reunión

III. Soporte

3.1 Aseguramiento de la calidad

3.1.1 Plantilla DCS - Glosario de términos

3.1.2 Plantilla DCS - Listas de chequeo

3.1.3 Plantilla DCS - No Conformidades

3.1.5 Plantilla DCS - Plan aseguramiento de la calidad

3.1.6 Plantilla DCS - Plan de mediciones

3.1.7 Plantilla DCS - Respuestas a No Conformidades

3.1.8 Plantilla DCS - Estándares de código y documentación

3.2 Gestión de configuración

3.2.1 Plantilla DCS - Pedido de cambio

3.2.2 Registro de pedidos de cambio

3.2.2 Plantilla DCS - Plan Gestión de Configuración

IV. Legal

4.1 Plantilla ALBET - Acta de Aceptación

- 4.2 Plantilla ALBET - Acta de Entrega
- 4.3 Plantilla ALBET - Acta de inicio de proyecto
- 4.4 Plantilla ALBET - Acta de Terminación de Proyecto
- 4.5 Plantilla ALBET – Carta
- 4.6 Plantilla ALBET – Indefiniciones
- 4.7 Plantilla ALBET - Informe Técnico
- 4.8 Plantilla ALBET - Minuta de reuniones
- 4.9 Plantilla ALBET - Proyectos Técnicos

En el caso de las versiones operativas del software, el factor determinante al momento de seleccionar los productos a validar y verificar que se debe seguir es el nivel de prioridad que le ha otorgado el arquitecto a los casos de usos definidos para dar solución a las necesidades del cliente. El orden que se debe tomar según la prioridad de los CU debe ser el siguiente:

- Primero, se deben tratar los CU **críticos** pues estos son considerados los mas importantes para los usuarios porque cubren las principales tareas o funcionalidades que el sistema debe realizar y son los que definen la arquitectura básica de este.
- En segundo orden, se deben considerar los CU **secundarios** que sirven de apoyo a los CU críticos, involucran funciones secundarias y tienen un impacto más modesto sobre la arquitectura, pero deben implementarse pronto porque responden a requerimientos de interés para los usuarios.
- En tercer lugar se consideran los CU **auxiliares** debido a que no son claves para la arquitectura ya que solo constituyen un complemento para los CU críticos o secundarios.
- Finalmente los CU **opcionales**, pues estos responden a funcionalidades que pueden o no estar en la aplicación y que no son imprescindibles en las primeras versiones.

3.1.2 PREPARAR EL ENTORNO DE V&V.

El entorno de V&V constituye un elemento indispensable para la correcta ejecución del proceso, pues en este aparato se describen los requerimientos mínimos desde el punto de vista de los recursos que se necesitan para desarrollar el proceso. No se puede olvidar que los recursos necesarios en cada caso están en dependencia de las técnicas, métodos y procedimientos que se emplearan en cada momento del proceso. El entorno a establecer para los proyectos de realidad virtual de la facultad 5

resulta generalmente el mismo. A continuación se proponen un conjunto de elementos que deben estar presentes al momento de establecer el entorno, aunque cada uno de ellos debe adaptarse a las características propias de cada proyecto y/o producto seleccionado.

3.1.2.1 RECURSOS HUMANOS

En la totalidad de los proyectos de la facultad 5 se cuenta con el personal necesario para la realización del proceso, pues cerca del 100% de estas personas las constituyen estudiantes y profesores de la facultad; el pequeño por ciento que no se ha incluido dentro de personas de la facultad suelen ser especialistas de la dirección central de calidad de la universidad y en algunos casos los especialistas funcionales que juegan un papel importante en el proyecto como por ejemplo; en el proyecto Juegos CNeuro se necesitan especialistas en el área de la discalculia los cuales pertenecen al Centro de Neurociencias de Cuba, para el proyecto Simulador Quirúrgico se precisa de especialistas en cirugías de mínimo acceso y endoscopia, por otro lado y de forma general se puede necesitar la ayuda de especialistas de SIMPRO la cual constituye una empresa dedicada al desarrollo de Simuladores Profesionales utilizando la Realidad Virtual que ha estado muy ligada a la producción en la facultad 5 de la UCI.

Las personas involucradas en el proceso necesitan desempeñar varios roles dentro del mismo, y entonces según el papel que juegue cada una de ellas, serán las actividades por las que han de responsabilizarse. Para satisfacer todas las necesidades requeridas por el proceso, a continuación se muestra una relación con los roles recomendados para cada uno de los proyectos.

Líder del proyecto: Jefe o persona más influyente dentro del proyecto. Persona capaz de conducir a los miembros del equipo de desarrollo hacia sus objetivos.

Planificador: El planificador del proyecto tiene la responsabilidad de programar las actividades de V&V conjuntamente con el líder del proyecto.

Stakeholders: personas u organización que están activamente implicadas en el negocio, ya sea porque participan en él o porque sus intereses se ven afectados con los resultados del proyecto. Pueden ser los propietarios, la dirección, quienes financian, los clientes, los trabajadores, los proveedores etc.

Para las revisiones técnicas formales o revisiones por pares los roles propuestos son:

Líder de Revisión: El líder de revisión es el responsable de asegurar una conducta eficiente y efectiva de la revisión dentro de los alcances de la misma.

Revisor Técnico: Es el responsable de proporcionar la información necesaria para la revisión. Se ocupa de la revisión técnica de los artefactos del proyecto y proporciona información adecuada sobre el proyecto que se está revisando.

Para realizar las pruebas se proponen los siguientes roles:

Administrador de pruebas: Es responsable del éxito del proceso de pruebas, sus principales tareas dentro de este ámbito son la planificación, gestión de recursos y la solución de problemas que impiden la correcta realización de las pruebas, en fin asegura que el proceso se realice con el máximo de calidad posible.

Analista de pruebas: Es responsable inicialmente de identificar y posteriormente de definir las pruebas requeridas, de supervisar la cobertura de la prueba y de evaluar la calidad total experimentada al probar los elementos. Este rol lleva la responsabilidad para representar apropiadamente las necesidades de los stakeholders que no tienen representación regular y directa en el proyecto.

Diseñador de pruebas: Es responsable de definir el método de prueba y asegurar su implementación exitosa. El rol incluye la identificación de técnicas apropiadas, herramientas e instrucciones para implementar las pruebas necesarias y encausar los recursos correspondientes para las pruebas. Se encarga del diseño de los casos de prueba que se aplicarán al software durante las pruebas.

Probador: Su principal labor es ejecutar los casos de prueba establecidos por el diseñador de pruebas, además de documentar los resultados obtenidos con la ejecución de cada uno de estos.

Hasta aquí se han definido los principales roles que intervienen en los procesos, pero cabe destacar que todos los miembros del proyecto pueden intervenir de alguna manera dentro de las actividades de V&V.

A todos los roles antes mencionados se deben agregar los correspondientes a las auditorías, aunque estos no deben definirse dentro del proyecto porque las personas que los desempeñan pertenecen a la dirección central de calidad de software de la UCI.

Líder de Auditoría: El auditor líder es el responsable de asegurar una conducta eficiente y efectiva de la auditoría dentro de los alcances de la misma.

Auditor de calidad: Son las personas que llevan a cabo el conjunto de tareas asignadas por el líder de auditoría dentro del marco de esta.

Todos estos roles con sus responsabilidades han de estar correctamente descritos en la Plantilla DCS - Roles y responsabilidad del expediente del proyecto.

Una vez que se han identificado los roles que intervendrán en el proceso de V&V, se debe realizar un proceso de capacitación al personal que desempeñará estos roles. Algunos cursos de importancia que deben recibir estas personas son:

- Introducción a la calidad de software.
- Diseños de casos de pruebas.
- Auditorías y Revisiones de software.
- Introducción a CMMI, profundizando en V&V.

A todo esto se puede adjuntar también algún tipo de material de capacitación para que las personas ajenas al proyecto logren la concepción de una idea clara de cuales son los objetivos del producto y las expectativas que este debe cumplir.

3.1.2.2 RECURSOS MATERIALES

Los recursos materiales representan otro de los factores claves para realizar el proceso, en esta categoría se destacan fundamentalmente las computadoras con sus necesidades de software y hardware.

Servidores

Los servidores en los proyectos juegan un papel importante pues en ellos se almacena y/o comparte la documentación, entregables, elementos de configuración y todo lo que se necesite para el desarrollo del proceso. Las características mínimas de hardware que deben tener estos son: 512 MB o 1 GB de RAM y una capacidad de almacenamiento entre 80 y 160 GB. El software base de este está en dependencia de las tecnologías usadas en cada proyecto, luego de un estudio realizado en cada uno de estos se pueden establecer dentro de la facultad dos fundamentales, Windows y Linux. Siguiendo el estilo de varios de estos proyectos los principales servicios que deben brindar estos servidores son:

Herramientas: Descripción

Trac: Gestión del Personal y Tareas.

Subversión (SVN): Gestión de versiones y salvvas.

Servidor de Ficheros: Compartición de información y bibliografía del proyecto.

PCs Clientes

Hardware:

- Microprocesador: Pentium 4 o superior.
- Memoria: 512 MB o 1 GB.
- Monitor: SVGA de 15 pulgadas.
- Disco Duro: 40 GB o superior.
- Tarjeta gráfica 256 MB (NVidia, ATI o una de iguales prestaciones).

Software:

- Microsoft Visual Studio .Net 2003 / IDE desarrollado por Microsoft.
- Rational Rose 2003 Enterprise Edition / Herramienta de modelado.
- Radiant / Compilador mapas 3D.
- 3DStudioMax 5.0 / Herramienta diseño gráfico 3D sobre Windows.
- Photoshop / Herramienta de diseño en 2D.
- DoxyGen. / Generador de ayuda automático.
- Microsoft Project / Herramienta para la gestión de proyecto.
- C++ / Lenguaje de programación Orientado a Objetos.
- OpenAL / Librería para el trabajo con sonido 3D.
- Microsoft Office / Herramienta para editar Documentos.
- Visual Paradigm / Herramienta de Modelado.

LISTAS DE CHEQUEO.

Las listas de chequeo constituyen un factor importante dentro del proceso de V&V, pues estas proveen a los responsables del proceso de una guía de preguntas, en forma de cuestionario que sirven para verificar el grado de cumplimiento de determinadas reglas establecidas a priori con un fin determinado. Para obtener unas listas de chequeo adaptadas a este proceso se deben tener en cuenta los criterios de V&V definidos en el acápite 3.1.3.

3.1.3 ESTABLECER LOS PROCEDIMIENTOS Y CRITERIOS DE V&V.

Este punto constituye el elemento más importante dentro de la preparación del proceso de V&V pues es aquí donde se describen cuales serán las técnicas que se aplicarán a los productos o componentes de estos. Además para cada una de las técnicas seleccionadas se describe cuales son los criterios y procedimiento a seguir para ejecutarlas.

PROCEDIMIENTOS DE V&V.

Teniendo en cuenta que las técnicas han sido clasificadas como estáticas y dinámicas, a modo de organización se abordará primeramente los aspectos concernientes a las estáticas y luego las dinámicas.

3.1.3.1 TÉCNICAS ESTÁTICAS

3.1.3.1.1 REVISIONES

Es una actividad de V&V con resultados muy efectivos para la detección de defectos, que entre otras ventajas se puede señalar que sirven como campo de entrenamiento para que el personal más joven puedan observar los diferentes enfoques al análisis, diseño e implementación de los sistemas.

El procedimiento que se propone a continuación está enfocado a revisiones externas del proyecto, pues los revisores son miembros del proyecto calidad de la facultad o integrantes de otros proyectos de realidad virtual. Sin embargo se considera fundamental que antes de pasar por estas revisiones externas, que el proyecto incluya en su cronograma de trabajo las revisiones internas donde los revisores deben ser miembros del proyecto que desempeñen estos roles, para ello se puede utilizar el mismo procedimiento de revisión bajo algunos ajustes.

Para realizar una revisión con calidad se deben seguir tres pasos fundamentales:

1. Preparar las revisiones.
2. Realizar las revisiones.
3. Analizar los resultados de las revisiones.

A continuación se establecen las especificaciones a seguir en cada punto:

Preparar las revisiones.

Con el objetivo de dar cumplimiento al primer punto se debe primeramente decidir que clases, módulos, documentos, etc. se desean revisar, esto debe de realizarse según estime conveniente el líder del proyecto calidad conjuntamente con los desarrolladores del mismo, partiendo de los criterios descritos en el acápite 3.1.1 referente a la selección de los productos de trabajo.

Determinar el tipo de revisión que se desea realizar:

- Revisión por Pares.
- Revisiones Técnicas Formales.

Luego el líder del proyecto calidad debe definir una agenda para realizar la revisión en la cual se recojan las fechas, tareas y responsables de cada actividad.

Además como parte de esta preparación es necesario seleccionar el personal que se involucrará en la revisión de lo cual es responsable el líder del proyecto. Existen tres tipos de personas involucradas en las revisiones:

- El líder es quien se encarga de realizar la planificación de la actividad, además de asegurarse que dicha planificación sea cumplida correctamente y como características más importantes debe tener la facilidad de dirigir, decidir y guiar al grupo de involucrados.
- Los desarrolladores, que como su nombre indica son los autores del producto o componente de este que va a revisar.
- Los revisores: para la asignación de este rol se debe tener en cuenta el tipo de revisión que se desea realizar:
 - a) Revisiones por pares, las personas seleccionadas deben poseer habilidades similares o superiores comparadas con las habilidades de los desarrolladores, en otras palabras son

personas que desempeñan los mismos roles que los desarrolladores del producto a revisar pero en otros proyectos.

- b) Otras revisiones, deben ser seleccionadas personas pertenecientes al proyecto calidad de la facultad o la universidad. En este caso no es necesario que los revisores desempeñen los mismos roles que los desarrolladores.

A continuación se deben preparar cuales son los recursos que se usaran durante la revisión, tomando como guía el acápite 3.1.2 referente a la preparación del entorno de V&V.

Otro elemento importante a la hora de definir el proceso de revisión es la selección de los criterios de verificación del producto de trabajo, los cuales están descritos en el acápite 3.1.3 relacionado con los criterios y procedimientos de V&V.

Realizar las revisiones.

Con el fin de llevar a cabo correctamente la revisión se presenta a continuación un procedimiento que puede servir de gran utilidad en este sentido.

1. El jefe de un proyecto de RV de la facultad solicita una revisión al jefe del proyecto calidad de software de la facultad en dependencia del cronograma de trabajo del proyecto y de los hitos establecidos en este. Esta solicitud debe realizarse por escrito, especificando claramente el nombre del proyecto, líder del proyecto solicitante y el producto o componente del producto que se desea revisar.
2. El jefe del proyecto calidad de software contacta con el líder del proyecto a revisar que evalúa la disponibilidad del producto, genera copias del material del producto y las entrega al primero para que este las distribuya a cuatro o cinco revisores como máximo.
3. Preparar la revisión teniendo en cuenta los elementos expuestos el punto de preparación de las revisiones.
4. Se reúnen el jefe de revisión, los revisores (uno actuará como documentador) y el líder del proyecto a revisar. En esta reunión el líder del proyecto a revisar describe los principales aspectos y características del elemento a revisar y luego el jefe de la revisión brinda una explicación de la agenda de trabajo que se ha establecido para dar cumplimiento a esta actividad.

5. Los revisores y el jefe de revisión revisan el producto (una o dos horas por día) según la planificación. En este punto se registran las no conformidades encontradas durante la actividad, para esto se cuenta en la universidad con la Plantilla DCS – No conformidades.
6. Una vez terminado el paso 5 se realiza la reunión de cierre donde se plantean y discuten las no conformidades encontradas por cada revisor, durante el debate el documentador debe ir tomando las notas oportunas. En búsqueda de una mejor calidad de la reunión se recomienda que la duración de esta sea inferior a las dos horas.
 - a) Al final decidirán si:
 - Aceptan el producto (no se hacen modificaciones).
 - Rechazan el producto (existen errores graves).
 - Aceptan el producto modificándolo (corregir errores).
7. El material obtenido de la revisión es:
 - Informe Final de revisión: Recoge el producto que se ha revisado, los revisores y los problemas detectados, así como las conclusiones finales. (Ver Anexo 8)
 - Plantilla de No Conformidades: Identifica las áreas problemáticas dentro del producto y sirve como guía para la corrección de errores por parte del productor.
8. Todos los revisores firman la conformidad de los resultados de la revisión y se da por concluida la revisión.
9. Se entregan los resultados de la revisión al líder del proyecto revisado para que las no conformidades encontradas sean corregidas. Una vez terminada la corrección se debe volver a comenzar otra vez por el paso uno y así sucesivamente hasta que desaparezcan todas las no conformidades o hasta que la dirección del proyecto calidad lo estime conveniente.

Analizar el resultado de las revisiones.

Para lograr un buen análisis de los resultados obtenidos durante la revisión es preciso que durante la reunión de cierre se dedique un punto a analizar los defectos encontrados, identificar las causas por las cuales se llegó a incurrir en errores y proponer las posibles acciones a tomar ante cada defecto encontrado para darle solución.

3.1.3.1.2 AUDITORÍAS DE CALIDAD.

Como es conocido las auditorías de calidad son consideradas una potente técnica para la detección de no conformidades que pueden atentar contra la calidad final del producto, por esta razón es de vital importancia su presencia dentro del proceso de V&V. Pero en nuestra universidad este tipo de actividad es realizada por la dirección central de calidad de software por lo que no corresponde a este trabajo la definición de procedimientos para su realización.

3.1.3.2 TÉCNICAS DINÁMICAS

Procedimiento de pruebas

El tema de las pruebas en los ambientes virtuales representa un factor de suma importancia pues con estas se busca validar hasta que punto se satisfacen las expectativas del cliente, para ello se deben diseñar casos de pruebas a diferentes niveles aplicando varios de los métodos y técnicas existentes con este fin; para obtener resultados satisfactorios en este sentido es necesario establecer con claridad cual es el objetivo que se persigue con la aplicación de cada CP.

Partiendo de lo anteriormente mencionado se pasa a definir un procedimiento general basado fundamentalmente en los métodos de caja blanca y caja negra. Pero debido a que por lo general los clientes solo muestran interés por la apariencia externa del producto y los resultados que ellos obtengan con su uso, y no se detienen a revisar la estructura interna de este; el método que tendrá mayor peso durante las actividades de pruebas será el método de caja negra.

El procedimiento de pruebas consiste en la aplicación de los métodos seleccionados dentro la sucesión de cuatro niveles fundamentalmente.

El primero de los niveles de prueba es el de **unidad**, como es conocido, a este nivel se tratan de encontrar los defectos de cada módulo por separado. El método que requiere mayor esfuerzo en este momento es el de caja blanca. Uno de problemas fundamentales que podemos encontrar, es la existencia de proyectos con gran cantidad de código, por lo que resultaría en ocasiones imposible probarlo todo debido a que por lo general, los proyectos no cuentan con el tiempo suficiente para hacerlo como consecuencia de los plazos fijados con los clientes. Como solución inmediata para este problema se deben seleccionar los módulos que tengan mayor peso dentro de la arquitectura base del producto, además se deben probar la mayor cantidad de algoritmos que contemplen operaciones importantes para el entorno virtual tales como, tratamiento de colisiones, simulación de leyes físicas, inteligencia artificial, control de sonido, etc.

Aunque en menor grado, las pruebas de caja negra también juegan un papel decisivo en este nivel, pues en un ambiente virtual existen elementos que deben ser validados antes de integrarse al sistema tales como, los objetos o cuerpos del entorno que pueden estar sujetos a características de apariencia externa previamente descritas, entre las que podemos encontrar el tamaño, color, textura, forma, etc. Las interfaces de usuario constituyen también factores claves que deben probarse en este nivel con el objetivo de asegurarse del correcto flujo de datos desde estas hasta la estructura interna de software y viceversa.

Luego de haber probado el software en unidades se debe pasar a ensamblar cada una de las partes como actividad de la **integración** del producto. En este nivel prevalece el uso de pruebas de caja negra con el objetivo de asegurar que la unión de los módulos satisface el cumplimiento de los requerimientos del software desde el punto de vista funcional. Aquí trataremos de encontrar fallos en la respuesta de un módulo cuando su operación depende de los servicios prestados por otro(s) módulo(s). Según nos vamos acercando al sistema total, estas pruebas se van basando cada vez más en la especificación de requisitos del usuario.

En este nivel pueden usarse además las pruebas de caja blanca de forma similar a las usadas a nivel de unidad pero esta vez trabajan con un nivel conceptual superior, pues aquí prestarán mayor atención a la llamada entre módulos ya que se trata de identificar todos los posibles esquemas de llamadas y ejercitarlos para asegurar una buena interacción entre módulos.

Terminado el proceso de integración se pasa inmediatamente a desarrollar el proceso de pruebas de **sistema**, en esta etapa se realizan un conjunto de pruebas diferentes con el objetivo de probar el sistema de manera global. Aunque cada prueba tiene un fin distinto al de las demás, todas tienen como punto en común la comprobación de que se hayan integrado correctamente todos los elementos del sistema permitiendo que este realice apropiadamente todas las funciones para las cuales fue concebido. En este nivel se usa el método de caja negra. Las pruebas del sistema que se deben considerar son:

- Función
- Seguridad
- Volumen
- Usabilidad
- Recuperación ante fallos

1. Pruebas de Función

Aseguran el comportamiento apropiado de los requisitos funcionales, incluyendo la navegación, entrada de datos, implementación adecuada de las reglas del negocio, procesamiento y obtención de resultados. Para realizar este tipo de prueba se ejecuta cada caso de uso, flujo de caso de uso, o función, usando datos válidos e inválidos, para verificar lo siguiente:

- Que se valide correctamente las reglas de negocio.
- Que el uso de datos válidos produzcan los resultados esperados.
- Que el uso de datos inválidos generen los mensajes de error y precaución correspondientes.

2. Pruebas de Seguridad

Comprueban que solo aquellos usuarios autorizados a acceder al sistema son capaces de ejecutar las funciones de este, pero estos usuarios deben estar restringidos a funciones específicas donde su acceso esté limitado solamente a los datos autorizados para él. Para realizar estas pruebas se deben diseñar CP teniendo en cuenta cada tipo de usuario, sus permisos y las operaciones que estos pueden realizar. Se deben considerar terminadas cuando para cada tipo de usuario sus operaciones funcionan como se esperaba.

3. Pruebas de Volumen

Estas pruebas se basan en asegurar que la aplicación funciona correctamente cuando se encuentran accediendo al sistema simultáneamente la cantidad máxima de clientes para los que fue concebido el mismo, todos ejecutando la misma funcionalidad en un período de tiempo relativamente largo. Estas pruebas terminan cuando han realizadas todos los CP planeados y se ha conseguido que el sistema no falle.

4. Pruebas de Usabilidad

La usabilidad es la característica de los productos de trabajo que permite que los usuarios puedan interactuar con ellos de la forma más fácil, cómoda e intuitiva posible. La mejor forma de obtener un producto usable es realizando un diseño pensado en el usuario, diseñando para y por el usuario. Los tipos de pruebas recomendados para verificar la usabilidad del software son:

Observación

Para realizar este método se recomienda que durante las pruebas alfa – beta (pruebas de aceptación) se observe a un pequeño grupo de usuarios y se anoten las principales dificultades que puedan presentar a la hora de encontrar algunos elementos de la interfaz necesarios para poder completar las funciones deseadas por estos. Este método es muy sencillo de usar pero a su vez es muy importante. El hecho de realizarse conjuntamente con las pruebas alfa-beta contribuye a la disminución del costo de las pruebas.

Entrevistas

En las entrevistas es necesario contar con una persona que lea las preguntas y anote las respuestas del entrevistado, esto permite formular preguntas de profundidad respecto a la interfaz del producto. Este método proporciona resultados inmediatos, contando con la ventaja de la profundidad a la que se puede llegar con el usuario.

5. Prueba de recuperación

Tras la ocurrencia de fallos, muchos sistemas deben recuperarse en un período de tiempo establecido y continuar brindando servicios, y en otros casos los sistemas deben ser tolerante a los fallos, o sea, aunque exista algún tipo de fallo estos no deben cesar su funcionamiento. Este tipo de prueba debe forzar al fallo del sistema para comprobar que la recuperación se lleva a cabo correctamente y evaluar la corrección de la inicialización ya sea de manera automática o manual.

Llegamos al último de los niveles de prueba, **Aceptación** del software. La aceptación del software se consigue mediante una serie de pruebas que demuestran la conformidad con los requisitos. En este nivel se usan exclusivamente pruebas de caja negra; las cuales son realizadas fundamentalmente mediante dos técnicas que son las pruebas alfa y las pruebas beta, con las cuales los clientes pueden validar hasta que punto el software satisface o no sus expectativas e informan a los desarrolladores los resultados de estas.

CRITERIOS DE V&V.

CRITERIOS PARA LA VERIFICACIÓN.

Verificación de los requisitos.

Se deberán verificar los requisitos, teniendo en cuenta los criterios siguientes:

- a) Se verificará la correcta identificación, elaboración y descripción de todos los artefactos generados en este flujo de trabajo.
- b) Los requisitos funcionales son consistentes, factibles y se pueden probar.
- c) Los requerimientos no funcionales son consistentes, factibles, se pueden probar y reflejan con precisión los requisitos del sistema.
- d) Los requerimientos funcionales han sido asignados adecuadamente a los requerimientos no funcionales de acuerdo con los criterios de diseño.
- e) Los documentos de requerimientos serán revisados en detalle para asegurarse de que estén realizados completamente y así mismo asegurar que proveen una base suficiente para el diseño del software.
- f) Los requerimientos funcionales están reflejados en al menos un caso de uso.
- g) Los requerimientos deben ser:
 - Especificados por escrito.
 - Descritos como una característica del sistema a entregar.
 - Los más abstracto y conciso posible.

Verificación del análisis y diseño.

Se deberá verificar el diseño, teniendo en cuenta los criterios siguientes:

- a) Se verificará la correcta identificación, elaboración y descripción de todos los artefactos generados en este flujo de trabajo.
- b) El diseño está correcto, es consistente con los requisitos y se puede rastrear, trazar, hasta éstos.
- c) El diseño implementa: una secuencia correcta de eventos, entradas, salidas, interfaces, flujo lógico, así como la definición, aislamiento y recuperación de errores.
- d) El diseño seleccionado se puede obtener de los requisitos.
- e) Se verificará el diseño total del sistema incluyendo todos los requerimientos de los componentes del software.

Verificación del código.

Se deberá verificar el código, teniendo en cuenta los criterios siguientes:

- a) El código se puede rastrear, trazar, hasta el diseño y los requisitos, se puede probar, está correcto y conforme con los requisitos y estándares de implementación del proyecto.
- b) El código implementa: una secuencia correcta de eventos, interfaces consistentes, un flujo de datos y de control adecuado, así como la definición, aislamiento y recuperación de errores.

- c) El código seleccionado se puede obtener del diseño o de los requisitos.
- d) Se verificará que se implemente el diseño planeado y buenas prácticas de implementación.
- e) Se verificará la correcta identificación, elaboración y descripción de todos los artefactos generados en este flujo de trabajo.

Verificación de la documentación.

Se deberá verificar la documentación, teniendo en cuenta los criterios siguientes:

- a) La documentación es adecuada, completa y consistente.
- b) La preparación de la documentación es oportuna.
- c) La gestión de la configuración de los documentos obedece a lo establecido en los procedimientos especificados.

Verificación de productos y procesos

Asegurar que el producto.

- a) Está de acuerdo con las especificaciones.
- b) Es correcto.
- c) Es íntegro, claro y consistente.
- d) Se han considerado alternativas apropiadas.
- e) Cumple con todos los estándares.
- f) Reúne los atributos de calidad especificados.

Hacer recomendaciones.

- a) Continuar en la etapa siguiente (si todo está bien).
- b) Corregir elementos determinados, luego continuar.
- c) Retroceder algunos pasos y corregir problemas.
- d) Monitorizar el progreso de un elemento de dudosa calidad.
- e) Hacer cambios en la metodología o herramientas.
- f) Modificar la programación del tiempo.
- g) Tomar decisiones de gestión (más recursos, formación, cambio de asignaciones, etc.).

CRITERIOS PARA LA VALIDACIÓN.

1. Validar el producto según el cumplimiento de los requerimientos establecidos para este.

Teniendo en cuenta el objetivo principal de la validación, es preciso asegurarse de que el producto satisface las expectativas del cliente y para ello deben diseñar CP que permitan demostrar que todas las funcionalidades previstas para este fueron desarrolladas y funcionan correctamente.

2. Validar acceso al sistema según el número de usuarios.

Necesidad: En no pocas ocasiones los ambientes virtuales son construidos con el objetivo de ser utilizados por uno o más usuarios simultáneamente.

Objetivo: Asegurarse de que el producto funciona correctamente una vez que los usuarios para los cuales fue concebido accedan a este.

Partición Equivalente

Tipo de Ambiente	Clases	Descripción
Mono-Usuario		En este tipo de ambiente solo puede tener acceso al sistema un único usuario.
	Válidas	<ul style="list-style-type: none"> • Accede al sistema un solo usuario.
	Inválidas	<ul style="list-style-type: none"> • Acceden al sistema más de un usuario.
Multi-Usuario		Este tipo de ambiente permite el acceso al sistema de uno o varios usuarios, presenta tres variantes fundamentales.
Variante 1	Permite acceder a un número X de usuarios como máximo.	
	Válidas	<ul style="list-style-type: none"> • Permitir el acceso de un número de usuarios

		menor o igual que X.
	Inválidas	<ul style="list-style-type: none"> Permitir el acceso de un número de usuarios mayor que X.
Variante 2	Permite acceder a un número X de usuarios como mínimo.	
	Válidas	<ul style="list-style-type: none"> Permitir el acceso de un número de usuarios mayor o igual que X.
	Inválidas	<ul style="list-style-type: none"> Permitir el acceso de un número de usuarios menor que X.
Variante 3	Permite acceder una cantidad de usuarios entre X y Z obligatoriamente.	
	Válidas	<ul style="list-style-type: none"> Permitir el acceso de un número de usuarios mayor o igual que X y menor o igual que Z.
	Inválidas	<ul style="list-style-type: none"> Permitir el acceso de un número de usuarios menor que X. Permitir el acceso de un número de usuarios mayor que Z.

Análisis de Valores Límites

Tipo de Ambiente	Clases	Descripción
Mono-Usuario		En este tipo de ambiente solo puede tener acceso al sistema un único usuario.
	Válidas	<ul style="list-style-type: none"> Accede al sistema un solo usuario.
	Inválidas	<ul style="list-style-type: none"> Acceden al sistema más de un usuario.

Multi-Usuario		Este tipo de ambiente permite el acceso al sistema de uno o varios usuarios, presenta tres variantes fundamentales.
Variante 1	Permite acceder a un número X de usuarios como máximo.	
	Válidas	<ul style="list-style-type: none"> • Permitir el acceso de un número X de usuarios.
	Inválidas	<ul style="list-style-type: none"> • Permitir el acceso de un número de usuarios menor que X. • Permitir el acceso de un número de usuarios mayor que X.
Variante 2	Permite acceder a un número X de usuarios como mínimo.	
	Válidas	<ul style="list-style-type: none"> • Permitir el acceso de un número X de usuarios.
	Inválidas	<ul style="list-style-type: none"> • Permitir el acceso de un número de usuarios menor que X. • Permitir el acceso de un número de usuarios mayor que X.
Variante 3	Permite acceder una cantidad de usuarios entre X y Z obligatoriamente.	
	Válidas	<ul style="list-style-type: none"> • Permitir el acceso de un número X de usuarios. • Permitir el acceso de un número Z de usuarios.

	Inválidas	<ul style="list-style-type: none"> • Permitir el acceso de un número de usuarios menor que X. • Permitir el acceso de un número de usuarios mayor que Z.
--	-----------	--

3. Probar ambiente virtual según el grado de inmersión.

Como consecuencia del avance en el campo de la realidad virtual aplicada, se hace necesario clasificar los entornos virtuales según el grado de inmersión de estos, las clasificaciones existentes son sistemas inmersivos y no inmersivos. Teniendo presente que cada uno de estas cuentan con características distintas, no se recomienda trazar las mismas pautas para determinar si el sistema contiene o no los requerimientos inmersivos para las cuales fue concebido. Para asegurarnos que nuestro entorno cumple con el grado de inmersión requerido por el usuario; se debe tener en cuenta durante el diseño de los CP el ajuste al documento de requerimientos, en el cual deben estar reflejados con claridad que tipo de inmersión se quiere lograr en el EV.

4. Probar ambiente virtual según el grado de realismo

En esta categoría se encuentran fundamentalmente tres tipos de ambientes virtuales que son los ambientes realistas, adaptados a la realidad y los fantásticos. Para cada uno de ellos se definen características diferentes por lo que aquí es imprescindible adaptarse al principio de V&V que plantea que “un modelo de simulación se construye con respecto a unos objetivos de estudio y su credibilidad se juzga respecto a esos objetivos”. En este sentido se debe tener bien claro cual es el objetivo para el que es concebido cada ambiente virtual puesto que no existe ninguna forma de V&V general y el hecho de que un ambiente virtual sea válido para un propósito no significa que lo sea para otros, pues la realidad es el único modelo válido de manera general. Independientemente del tipo de ambiente se hace indispensable tener en cuenta aspectos tales como la detección de colisiones, ajustes de sonido, tamaño, texturas, colores, etc., para obtener buenos resultados después de aplicar las pruebas.

Los ambientes realistas tienen a su favor la existencia de un mundo real con el que comparar y establecer criterios de similitud para valorar la credibilidad. El problema fundamental a surgir puede ser a la hora de establecer ¿Cuál es el mundo real? teniendo en cuenta que cada individuo puede tener una visión distinta sobre el mundo real, por lo que es de suma importancia haber llegado a un

consenso con los clientes sobre que es lo que realmente ellos necesitan para entonces trazar las pautas necesarias para satisfacer las expectativas de este. Además se puede presentar otro problema durante el proceso por motivo de que no siempre los datos obtenidos del mundo real son precisos y aun siéndolos no se debe olvidar que solo representan una muestra de la realidad lo que crea cierta imprecisión.

En los ambientes adaptados a la realidad se debe tener en consideración los mismos elementos que en los ambientes realistas pero se han de incorporar otros como el ajuste a factores tales como escalas de tamaño, escalas de sonido etc.

En los ambientes fantásticos el grado de dificultad para realizar las pruebas aumenta en complejidad debido que en este tipo de sistema cada elemento puede ser producto de la imaginación de los desarrolladores y por lo general nos enfrentamos al problema de inexistencia de un mundo real con el que comparar, por lo que en este sentido la correcta aplicación de las pruebas requiere de gran creatividad y perspicacia por parte del equipo de pruebas.

5. Probar ambiente virtual según el grado de interactividad

Al momento de probar cuan interactivo resulta ser el ambiente virtual se tienen establecidas tres categorías con características propias de cada una, estas son ambientes pasivos, exploratorios e interactivos.

Los ambientes pasivos son entornos inmersivos no interactivos. Entornos en los que podemos ver, oír, y quizás sentir lo que sucede, pero no es posible controlar lo que ocurre. Los CP en este sentido se deben centrar en tratar de modificar o controlar los elementos y eventos que ocurren dentro del ambiente. Las únicas clases válidas para este tipo de prueba consisten en la negatividad del ambiente a ser controlado o modificado por el usuario.

Los ambientes exploratorios permiten al usuario desplazarse por el entorno virtual para explorarlo; como ejemplo de estos encontramos los paseos arquitectónicos y las obras de arte virtuales. Los CP para este caso deben consistir en que el usuario se mueva por las diferentes secciones del ambiente y traten de modificar los elementos y eventos de este, por supuesto estas resultarían ser clases inválidas en todo momento.

Los ambientes interactivos permiten al usuario explorar y experimentar con el entorno, modificándolo; este tipo de ambiente puede ser una mezcla de los anteriores con la particularidad de que en este el

usuario puede modificar y controlar algunos de los elementos del sistema. Los CP deben consistir en permitir a los usuarios desplazarse, oír y/o ver etc, por todo el ambiente pero además debe permitir que este pueda realizar en el ambiente las operaciones o modificaciones previamente establecidas por los desarrolladores.

6. Probar detección de colisiones

Para validar si en nuestro ambiente virtual se realiza correctamente el proceso de detección de colisiones es necesario ubicarse en que categoría se encuentra este según el grado de realismo. Pues para los ambientes realistas y adaptados a la realidad se hace necesario establecer CP en los cuales los objetos dinámicos (objetos en movimiento) del entorno interactúen con sí mismos y con los objetos estáticos (objetos fijo en tiempo y espacio) del entorno con el objetivo de asegurar que el comportamiento frente a las colisiones de los cuerpos rígidos o deformables se manifieste lo más real posible. En el caso de los ambientes fantásticos el comportamiento ante las colisiones de cada objeto debe estar especificado en los requerimientos del sistema.

7. Probar ajuste de sonido

Las escenas pueden contener sonidos que tienen posiciones específicas en el espacio 3D, de manera que el usuario que navegue en ella tenga la sensación auditivamente de que esta en un ambiente real. Otra forma de encontrar sonidos en el EV puede ser producto de acciones de los usuarios. Por esta razón los CP deben estar dirigidos a moverse por todo el entorno posible y tratar de realizar todas las operaciones y acciones permitidas a los usuarios con el fin de identificar la inexistencia o desajustes de sonido dentro del ambiente.

8. Probar Animación de objetos

Las escenas 3D pueden contener elementos no estáticos o dinámicos, como objetos animados dentro de la aplicación o animaciones creadas con algún software de animación. Para asegurar que la animación de los objetos del ambiente virtual se comporta de forma esperada, en el caso de animaciones dependientes de acciones del usuario es preciso que los CP traten probar que ante cada entrada correcta de los usuarios el ambiente responde correctamente y en caso de animaciones propias del sistema se debe tratar de ejercitar todo el ambiente con el objetivo de ver que estas se realicen de manera correcta.

9. Probar Simulación de física

Para incrementar el realismo en los EV es necesario implementar una simulación de física que permita a los objetos ser afectados por la gravedad, que haya fricción entre ellos, que puedan aplicarse fuerzas a mecanismos, etc. Este elemento es importante para ambientes realistas y adaptados a la realidad pues si se desea mostrar un ambiente lo suficientemente creíble, los CP deben enfocar su atención a probar el cumplimiento de principales leyes físicas que son de fácil identificación por los clientes. En los ambientes fantásticos el cumplimiento de cada elemento de estos debe estar claramente definido en los requerimientos del ambiente para poder emplear estos como forma de referencia a la hora de probar.

10. Probar la Inteligencia Artificial

Para la simulación de comportamientos complejos dentro de los ambientes virtuales, se requiere la implementación de algoritmos de inteligencia artificial. Por ejemplo: para determinar que metas requiere cumplir un usuario en una aplicación de entrenamiento etc, en este sentido se deben identificar los principales momentos en los cuales el sistema debe tomar ciertas decisiones en dependencias de metas cumplidas por los usuarios o acciones realizadas por estos para entonces diseñar los CP capaces de ejercitar cada uno de estos momentos y poder validar el estado en el que se encuentra la inteligencia artificial.

11. Probar comportamientos de los objetos.

El comportamiento de los objetos es un aspecto importante en la construcción de cualquier SRV. En un sistema, a un objeto le corresponderá un conjunto de comportamientos, los cuales pueden ser asignados por el usuario, a partir de una lista predefinida de comportamientos. Algunos de los posibles comportamientos son los siguientes: modificar el color de un objeto, cambiar el sonido asociado a un objeto, modificar la transparencia de un objeto, cambiar la intensidad de una luz, modificar el color de una luz, activar o desactivar un sonido, encender o apagar una luz, cambiar la posición de un objeto, modificar la orientación de un objeto, cambiar el tamaño de un objeto, y modificar las siguientes propiedades a los objetos: si está afectado o no afectado por la gravedad, si el objeto es móvil o fijo. Los CP en este sentido deben prestar mayor atención al hecho de ejercitar cada comportamiento posible de los objetos, teniendo presente que se deben verificar tanto los comportamientos dependientes de acciones del usuarios como los comportamientos automáticos que ha de generar el sistema.

3.2 VALIDAR Y VERIFICAR LOS PRODUCTOS SELECCIONADOS

3.2.1 REALIZAR LA VALIDACIÓN Y LA VERIFICACIÓN

Para realizar el proceso de V&V ajustado al modelo CMMI en los proyectos de RV de la facultad 5 se deben seguir un conjunto de pasos que serán explicados a continuación. Para la documentación de todo lo referente al proceso se usará fundamentalmente el Plan de aseguramiento de calidad propuesto por la dirección de calidad de software de la UCI, aunque se propondrán nuevos acápites dentro de este para lograr que en el mismo quede adaptado a las necesidades de documentación del proceso.

El primer paso para llevar a cabo el proceso consiste en introducir el término validación y verificación dentro del vocabulario de todos los miembros del proyecto, esto debe quedar reflejado en el acápite referente a definiciones, acrónimos y abreviaturas. En este punto han de plasmarse los conceptos de estos términos por separado, así como la unión de ambos en un único proceso llamado, proceso de validación y verificación, en este momento se debe describir también el uso de V&V como acrónimo de estos términos.

Dentro de los objetivos de calidad que se proponga cada proyecto con su plan de aseguramiento de la calidad, debe quedar registrado cual es el propósito que se persigue al realizar el proceso de V&V. En términos generales se puede decir que dicho propósito consiste entre otras cosas en asegurar que los productos cumplen los requisitos especificados y que el producto y sus componentes cumplen con su uso especificado cuando son puestos en su entorno real.

Entrando en materia de gestión se recomienda agregar un acápite donde se describan los principales recursos humanos, materiales y otros, descritos en el acápite 3.1.2 de este trabajo el cual se refiere a la preparación del entorno de V&V que se utilizarán durante este proceso; estos recursos son previamente definidos durante la preparación de la actividad; además deben quedar documentados en la Plantilla DCS - Ambiente de desarrollo, Plantilla DCS - Plan de capacitación, Plantilla DCS - Roles y responsabilidades, Plantilla DCS - Listas de chequeo, etc.

Luego, en los estándares y guías, no se puede dejar de incluir el modelo CMMI; pues este ha sido tomado como guía para diseñar procesos efectivos dentro del ámbito de la V&V. Otro elemento a utilizar como guía es el documento de lineamientos mínimos de calidad de los proyectos productivos de la UCI en su versión 5.3, debido fundamentalmente a que en este se recogen una serie de elementos que ajustan el proceso a las políticas de la universidad.

Antes de aterrizar sobre las actividades concretas que se han de realizar durante la V&V, se deben seleccionar cuales serán los productos o componentes de estos que participarán en el proceso. Para esto se deben definir los criterios discriminantes al momento de hacer la selección; estos criterios deben quedar reflejados también en el plan de aseguramiento de la calidad.

Al llegar a este punto se está en condiciones de hablar sobre las técnicas a usar, que como es conocido se encuentran divididas en estáticas y dinámicas. Para ambas técnicas deben quedar identificadas en el plan cuales son las actividades, responsables, recursos, así como el cronograma de trabajo para cada una de ellas.

Dando continuidad al estilo con el cual se ha trabajado en esta investigación se abordarán primeramente las estáticas, o sea, las revisiones y las auditorías de calidad.

Para realizar las revisiones se debe establecer una planificación de estas, para ello se propone una Plantilla – Plan de revisión de software (Ver Anexo 9), pues se considera que la forma de manejar en este tema en el Plan de Aseguramiento de la Calidad no es la más óptima debido a que todas las revisiones no tienen el mismo propósito, alcance, cronograma, responsables, etc. Esta plantilla contribuye a una mejor planificación de las revisiones. Además se deben describir los criterios y procedimientos a utilizar para estas. Otro factor importante al realizar estas actividades consiste en elaborar listas chequeo ajustadas a las características de los productos seleccionados.

A continuación se muestran un conjunto de revisiones y auditorías que pueden aplicarse:

- Revisión de los Requerimientos (se corresponde con la tradicional Revisión de las Especificaciones del Software)
- Revisión de la Arquitectura (se corresponde con la tradicional Revisión del Diseño Preliminar)
- Revisión del Diseño (se corresponde con la tradicional Revisión del Diseño Crítico)
- Auditoría de la configuración funcional (para verificar que todos los requerimientos han sido cumplidos)
- Auditoría de la configuración física (para verificar que el software y su documentación están completos y listos para entregar)
- Auditoría del Proceso
- Revisión del Proceso

Para aplicar las llamadas técnicas dinámicas se ha de establecer un plan de pruebas con el objetivo de organizar el trabajo durante esta fase. En este plan se recogen varios elementos generales que concuerdan con algunos de las técnicas dinámicas (asignación de responsabilidades, definición de roles, cronograma, etc.), pero el elemento más significativo que caracteriza a este, es la presencia de una estrategia de pruebas que describe como serán alcanzado los objetivos para cada tipo de prueba propuesto; como parte de esta estrategia se definen objetivo, casos de pruebas, técnicas, etc.

Una vez definidas las técnicas, procedimientos y criterios a usar para realizar la V&V solo nos queda ejecutar cada una de estas con la mayor calidad posible para poder obtener resultados satisfactorios. Además se deben establecer los criterios de finalización de las pruebas, que pueden ser:

Criterios erróneos

- Se acabo el tiempo.
- Hay mucho trabajo que hacer.
- Se ha detectado un fallo no corregible.
- No ha habido fallo durante las pruebas realizadas.

Criterios Adecuados

- El análisis de valores límites en las interfaces no ha detectado errores.
- Los casos de pruebas restantes no han encontrado defectos.
- Los clientes están satisfechos con el producto final.
- Otros.

3.2.2 ANALIZAR LOS RESULTADOS DE LA V&V.

Para analizar los resultados de la V&V se propone que durante las reuniones de cierre de cada revisión, auditoría o prueba se dedique un punto a este fin. Durante este momento se deben mencionar y describir las áreas problemáticas encontradas durante cada actividad; luego se debe pasar a identificar las principales causas que pudieron motivar la ocurrencia de errores en el proyecto y como aspecto complementario a cada causa de defecto involucrada se deben proponer las posibles vías de solución en aras de no volver a repetir las mismas fallas. Todo este proceso debe quedar por escrito como constancia del análisis de los resultados del proceso, además esta documentación puede ser utilizada por los desarrolladores como parte de su capacitación personal lo que tiene como fundamental ventaja que en próximos productos del proyecto el número de defectos cometidos disminuya.

3.3 VALIDACIÓN DE LA SOLUCIÓN PROPUESTA

Después de proponer la estrategia para dar solución a los problemas detectados durante la validación y verificación de los productos en los proyectos de realidad virtual de la facultad 5 fue necesario conocer el grado de aceptación de esta por expertos en temas de calidad de software y desarrollo de software de realidad virtual.

La validación se realizó mediante un cuestionario (Ver Anexo 10), este fue aplicado a 7 expertos en los temas antes mencionados. A continuación se muestran los expertos seleccionados:

Experto 1: Ing. José Manuel Pardo Matos

Ingeniero en Ciencias Informáticas. Graduado en Julio de 2007. Profesor del Departamento de Ciencias Básicas desde septiembre de 2007; Profesor Adiestrado. Asesor de Calidad de la Facultad 5. Actualmente cursando la maestría en Calidad de Software. Ha cursado varios postgrados durante su etapa de adiestramiento entre los que están Docencia e Innovación Universitaria, Auditoría TIC, Técnicas Avanzadas de desarrollo de SW, Métricas de SW, Validación y Verificación, Monitoreo y Control de Proyectos. Participó en UCIENCIA 2007.

Experto 2: Ing. Yanoski Rogelio Camacho Román

Ingeniero en Informática. Graduado de la CUJAE (Instituto Superior Politécnico José Antonio Echeverría) en el 2004. Cuatro años de experiencia en los gráficos y la Realidad Virtual, Líder del proyecto Herramientas de desarrollo para sistemas de realidad virtual desde el 2004. Diplomado en Diseño y Programación de Videojuegos de la Universidad Abierta de Catalunya Premio del rector 2008 en el proyecto

Experto 3: Ing. Yoander Cabrera Díaz.

Ingeniero en Ciencias Informáticas, profesor adiestrado que imparte la asignatura de Ingeniería de Software. Graduado en Julio de 2007. Actual líder del proyecto Juegos de Consola, el cual pertenece al polo de Realidad virtual. Ha presentado publicaciones en UCIENCIA en las que ha obtenido la distinción de destacado. Actualmente se encuentra cursando un diplomado de Realidad Virtual.

Experto 4: Ing. Yulier Casas Estrada.

Ingeniero en Ciencias Informáticas, Graduado en la UCI, profesor adiestrado que imparte asignaturas de Programación, con dos años de graduado y más de 1 año como líder del proyecto Juego CNeuro.

Ha participado como ponente en un evento nacional y dos internacionales, y en todos los casos los trabajos han sido publicados.

Experto 5: Lic. Luis Gabriel Viciado Caraballoso

Licenciado en educación. Graduado en 1980 en el ISP “José Martí” de Camagüey. Veintiocho años de experiencia en educación universitaria y preuniversitaria. Líder del proyecto Laboratorios Virtuales (PROLAVI) desde 2004. Participación en todas las ediciones del evento UCIENCIA. Miembro de la sociedad cubana de Matemática y Computación.

Experto 6: Ing. Gadiel Carrero Sotolongo

Ingeniero en telecomunicaciones y electrónica, graduado en la Universidad Central de las Villas. Con experiencia de 4 años como jefe de proyecto de Diseño 3D. Participó en el desarrollo del primer producto de exportación de la UCI, un simulador de Autos.

Experto 7: Lic. Juan Manuel Medero Martínez

Licenciado en Ciencias de la Computación, tres años de graduado y uno de experiencia en la rama de realidad virtual. Actualmente se desempeña como líder del proyecto Simulador Quirúrgico.

Los resultados arrojados por la entrevistas son de forma general los siguientes

1. ¿Considera usted que la estrategia propuesta para realizar los procesos de validación y verificación se ajusta a las necesidades de los proyectos de realidad virtual?

Todos los expertos coinciden en que la propuesta presentada se ajusta a las características y necesidades de los proyectos de realidad virtual de la facultad 5. Además el 100% de los entrevistados le atribuyen gran importancia al trabajo pues en ninguno de los proyectos del polo de RV se tiene establecido como realizar estos procesos.

2. ¿Cree usted que con la aplicación de las técnicas y procedimientos propuestos se logre incrementar la calidad final de los productos de trabajo?

Plantean que la calidad del producto final depende de muchos otros factores pero con la definición de estas técnicas y procedimiento se incrementará en alguna medida la calidad final de los productos porque calidad en los procesos de producción tributa a la calidad final de los productos. Además

consideran que la técnica de revisiones por pares puede traer grandes beneficios pues además de detectar errores puede retroalimentar a los miembros de los proyectos con los conocimientos de sus homólogos en otros proyectos.

3. ¿Considera usted que son suficientes los elementos propuestos para preparar el entorno de validación y verificación? ¿Agregaría o eliminaría algún otro elemento?

Los entrevistados respondieron que de manera general que en la propuesta se definen los elementos necesarios para preparar el entorno de validación y verificación, aunque señalan que en dependencia de las características de cada proyecto se deben agregar algunos elementos propios de los productos, ejemplo de estos pueden ser, pinzas, sensores, joystick, etc.

4. ¿Considera importante el hecho de que la propuesta esté basada en el modelo CMMI?

Los expertos en RV consideran que el hecho de que esta propuesta esté basada en CMMI le da importancia y nivel científico a esta pues este es un modelo reconocido internacionalmente por su calidad y nivel. Además uno de los expertos en calidad de software precisa que CMMI propone metas y prácticas para los procesos de validación y verificación de forma clara.

5. ¿Qué evaluación le otorgaría a la solución propuesta?

El total de los expertos entrevistados consideran que con la puesta en práctica del presente trabajo se incrementaría la calidad final de los productos de trabajo, por tal razón le otorgan la evaluación de “Bien”.

CONCLUSIONES

Al finalizar el presente trabajo se dio cumplimiento al objetivo general concebido, con esto se provee a la facultad 5 de la Universidad de las Ciencias Informáticas de una estrategia basada en CMMI para realizar los procesos de validación y verificación en los proyectos de realidad virtual con la cual se logrará el incremento de la calidad final de los productos de trabajo en estos proyectos.

En la estrategia propuesta se definieron dos actividades fundamentales que son preparar la V&V, y validar y verificar los productos seleccionados.

Se propuso un procedimiento para las revisiones el cual consta de tres actividades principales, preparar la revisión, realizar la revisión y analizar los resultados de la revisión. Como complemento de estas actividades se definieron las plantillas, Plan de revisión de software e Informe final de revisión las cuales facilitarán el control y organización de las revisiones de software.

Además se definió un procedimiento general para realizar las pruebas de software mediante el uso de los métodos de caja blanca y caja negra a los niveles de unidad, integración, sistema y aceptación.

RECOMENDACIONES

Partiendo de que con el desarrollo del presente trabajo se dio cumplimiento a su objetivo general se recomienda que se ponga en práctica la estrategia propuesta en los proyectos de realidad virtual de la facultad 5 de la UCI.

Además se propone que en la facultad se continúe trabajando en aras de adaptar los fundamentos del modelo CMMI en relación a las necesidades de los proyectos de realidad virtual con el fin de optar en un futuro por una certificación del mismo, esto incrementaría considerablemente el prestigio del polo de realidad virtual de la UCI.

BIBLIOGRAFÍA

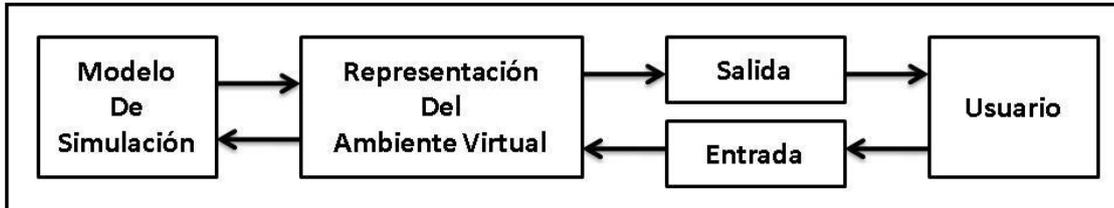
1. Introducción a la Gestión de Software. Principios de calidad establecidos en la UCI. **Departamento de Ingeniería y Gestión de Software de la UCI**. Ciudad de la Habana : s.n., 2007.
2. **Morales, Luisa**. BAQUIA, Knowledge Center. [En línea] 13 de Junio de 2005. [Citado el: 16 de Octubre de 2007.] <http://www.baquia.com/noticias.php?id=9778>.
3. **MINREX**. www.cubaminrex.cu. [En línea] 2004. [Citado el: 2 de Octubre de 2007.] www.cubaminrex.cu/Sociedad_Informacion/Cuba_SI/Informatizacion.htm .
4. **IPS Cuba**. Cuba a la mano. [En línea] 2004. [Citado el: 25 de septiembre de 2007.] <http://cubaalamano.net/sitio/client/article.php?id=5780>.
5. **CITMATEL**. CITMATEL (Tecnologías de la Información y Servicios Telemáticos). [En línea] CITMATEL, 1 de Noviembre de 2007. [Citado el: 1 de Noviembre de 2007.] <http://www.citmatel.cu/noticia.php?id=69>.
6. **Software Engineering Institute**. CMMI® for Development, Version 1.2. 2006.
7. **Humphrey, Watts S**. Introducción al Proceso Software Personal. La Habana : Felix Varela, 2005.
8. **Pressman, Roger S**. Ingeniería de Software. Un enfoque práctico. La Habana : Félix Varela, 2005.
9. **Laguna, Jesus Rivero**. Ingeniería del Software. Madrid : Fundacion para la difusión de las ingenierías Informática y de Telecomunicaciones, 2000.
10. **Software Quality Systems S.A.** . Software Quality Systems S. A. [En línea] 2007. [Citado el: 05 de Noviembre de 2007.] <http://www.sqs.es/es/services/validation.php>.
11. **Dirección de Calidad. Desoft**. Tecnología de la información. Procesos del ciclo de vida del software (ISO/IEC 12207 - 2006) Versión 2 en Español. 2006.
12. **Mora, Fernando Guzmán**. [En línea] [Citado el: 04 de Febrero de 2008.] <http://encolombia.com/medicina/fmc/Gaceta7N2-editorial.htm>.
13. [En línea] [Citado el: 04 de Febrero de 2008.] <http://64.233.169.104/search?q=cache:2rM4DnEwtuwJ:cursweb.educadis.uson.mx/raragon/TGP1.doc+concepto+de+proceso&hl=es&ct=clnk&cd=4>.
14. **Ericka Corrado Padilla, Julián J. Delgado y Salvador Castañeda**. Supercomputadora. CICESE2000. [En línea] Febrero de 2000. [Citado el: 07 de Noviembre de 2007.] <http://telematica.cicese.mx/computo/super/cicese2000/realvirtual>.
15. **Carbayo, Jesús Ledesma**. Departamento de Ingeniería Electronica. Universidad Politécnica de Madrid. [En línea] [Citado el: 08 de Noviembre de 2007.] <http://www.die.upm.es/cursos/insn/VR.pdf>.

16. **Santos, J. M.** [En línea] Mayo de 2001. [Citado el: 22 de Noviembre de 2007.]
<http://www.hab2001.sld.cu/arrepdf/00275.pdf>.
17. **Javier García, Antonio de Amescua, Manuel Velasco.** Asociación de Técnicos de Informática. [En línea] 2006. [Citado el: 22 de Noviembre de 2007.]
<http://www.ati.es/IMG/pdf/GarciaGuzmanNum2Vol2.pdf>.
18. **Getino, Antonio Monleón.** Tesis Doctorals en Xarxa. [En línea] 21 de Octubre de 2005. [Citado el: 26 de Noviembre de 2007.] <http://www.tdx.cesca.es/TDX-0112106-093218/>.
19. **Martínez, Cristian A.** [En línea] [Citado el: 28 de Noviembre de 2007.]
20. **Jenkins, Dr. Marcelo.** Informática 2007. Evento Virtual. [En línea] 2007. [Citado el: 20 de Noviembre de 2007.]
http://www.informaticahabana.com/evento_virtual/?q=node/624&ev=III%20Taller%20Internacional%20de%20Calidad%20en%20las%20TICs.
21. **Candelo, Sergio.** Snoop Consulting. [En línea] 11 de Septiembre de 2007. [Citado el: 26 de Noviembre de 2007.]
http://www.snoopconsulting.com/snoop_es/index.php?option=com_content&task=view&id=380&Itemid=1.
22. **Itera.** Itera. [En línea] 27 de Junio de 2007. [Citado el: 26 de Noviembre de 2007.]
http://www.iteraprocess.com/index.php?option=com_content&task=view&id=2&Itemid=59.
23. **Martínez, Tansania León.** [En línea] Noviembre de 2006. [Citado el: 26 de Noviembre de 2007.]
http://www.itesm.mx/cronicaintercampus/no_39/academica_sec_5.html.
24. **H@nz.** H@nz. [En línea] 18 de Junio de 2007. [Citado el: 26 de Noviembre de 2007.]
<http://hancocchi.net/empresa-peruana-obtiene-certificacin-cmmi/>.
25. **DBAccess.** DBAccess. Soluciones Técnicas para el mercado Global. [En línea] Noviembre de 2007. [Citado el: 02 de Diciembre de 2007.] <http://www.dbaccess.com/spanish/noticias/cmmi.html>.
26. **Alba, Rafael.** www.americaeconomica.com. [En línea] 11 de Octubre de 2002. [Citado el: 18 de Octubre de 2007.] <http://www.americaeconomica.com/numeros3/182/reportajes/rafa182.htm>.
27. **Méndez, Carlos.** [En línea] [Citado el: 12 de Septiembre de 2007.]
http://www.acis.org.co/fileadmin/Conferencias/IntroduccionCMMI_CarlosMendez.pdf.
28. **Fuente (a), Pablo de la.** ETSII. Laboratorio docente de la ETSI Informática. [En línea] 2007. [Citado el: 03 de Diciembre de 2007.]
http://jair.lab.fi.uva.es/~pablfue/leng_simulacion/slides/0607/vv_0607_t.pdf.
29. **Fuente (b), Pablo de la.** ETSII. Laboratorio docente de la ETSI Informática. [En línea] 2007. [Citado el: 3 de 12 de 2007.]
http://jair.lab.fi.uva.es/~pablfue/leng_simulacion/materiales/v_v_0405.pdf.

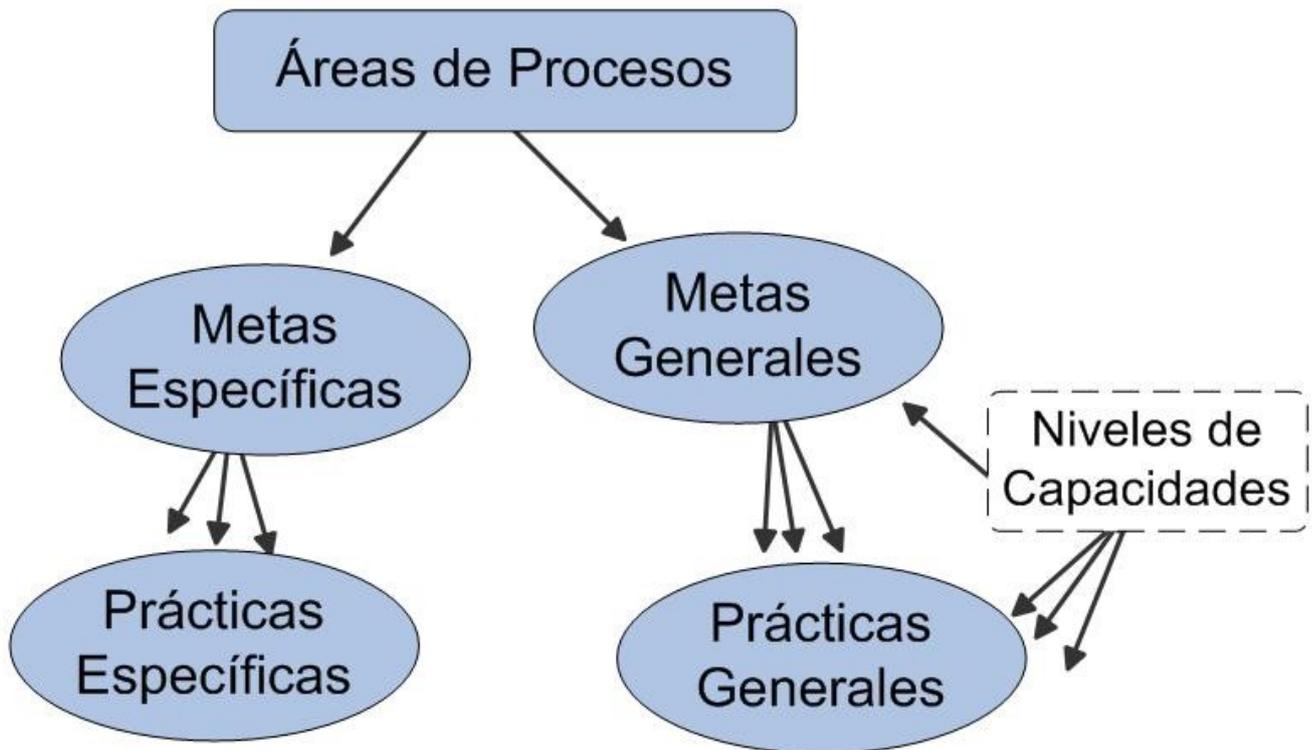
30. **Fernández Pons, Yanet.** Monografias.com. Pruebas de Aceptación del Cliente . [En línea] [Citado el: 15 de Febrero de 2008.] <http://www.monografias.com/trabajos36/pruebas-de-aceptacion/pruebas-de-aceptacion2.shtml>.
31. Wikipedia. Prueba unitaria. [En línea] 12 de Febrero de 2008. [Citado el: 10 de Marzo de 2008.] http://es.wikipedia.org/wiki/Prueba_unitaria.
32. Sistema de Preparación del Plan Diario de Producción. Pruebas al Sistema. [En línea] [Citado el: 18 de Marzo de 2008.] <http://sisab.lce.org/~roconner/pruebas.html>.
33. Sabueso Web. Métodos de caja blanca y caja negra . [En línea] 18 de Julio de 2007. [Citado el: 18 de Marzo de 2008.] <http://sabuesoweb.wordpress.com/2007/07/18/metodos-de-caja-blanca-y-caja-negra/>.
34. Angelfire. Clase 12: Diseño de casos de prueba . [En línea] [Citado el: 18 de Marzo de 2008.] <http://www.angelfire.com/empire2/ivansanes/bywbox.htm>.
35. **Collado, Manuel.** Pruebas de Software. [En línea] Marzo de 2003. [Citado el: 19 de Marzo de 2008.] <http://lml.ls.fi.upm.es/ftp/ed2/0203/Apuntes/pruebas.ppt>.
36. **Barreiro Alonso, Enrique.** Control de calidad: revisiones técnicas formales. [En línea] [Citado el: 04 de Diciembre de 2007.] <http://trevinca.ei.uvigo.es/~ebalonso/asignaturas/esx/guiones/esxClase29.pdf>.
37. **Oviedo, Ana Isabel.** Revisiones Técnicas Formales. [En línea] [Citado el: 04 de Diciembre de 2007.] <http://siona.udea.edu.co/~aoviedo/Arquitectura%20de%20Software/RTF.htm>.
38. INEI, Instituto Nacional de Estadística e Informática. GARANTIA Y CONTROL DE CALIDAD DEL SOFTWARE. [En línea] [Citado el: 04 de Diciembre de 2007.] <http://www.inei.gob.pe/biblioineipub/bancopub/inf/Lib5042/cap20.htm>.
39. **Hernández Parro, Casimiro.** Uso de la representación continua de CMMI® para la Mejora de Negocio. [En línea] 01 de Marzo de 2007. [Citado el: 20 de Marzo de 2008.] http://www.calidaddelsoftware.com/documentos/III%20Semana%20CMMI/Ponencias%20III%20Semana%20CMMI/representacion_continua_cmml_para_mejora_de_negocio.pdf.
40. Wikiprod. Wikipedia de producción. [En línea] 05 de Abril de 2008. [Citado el: 06 de Abril de 2008.] http://wiki.prod.uci.cu/index.php/Facultad_5.
41. SQA5 Realizar Revisión Técnica Formal (RTF). [En línea] [Citado el: 22 de Abril de 2008.] http://www.fing.edu.uy/inco/cursos/ingsoft/pis/memoria/experiencia2000/modelo_de_proceso1/lineas_de_trabajo/gestion_de_calidad/SQA5.htm.
42. **Añez, Juan Carlos.** NEOGENY. [En línea] 6 de Junio de 2007. [Citado el: 24 de Abril de 2008.] <http://blog.neogeny.org/2007/06/sobre-la-revisin-por-pares.html>.

ANEXOS

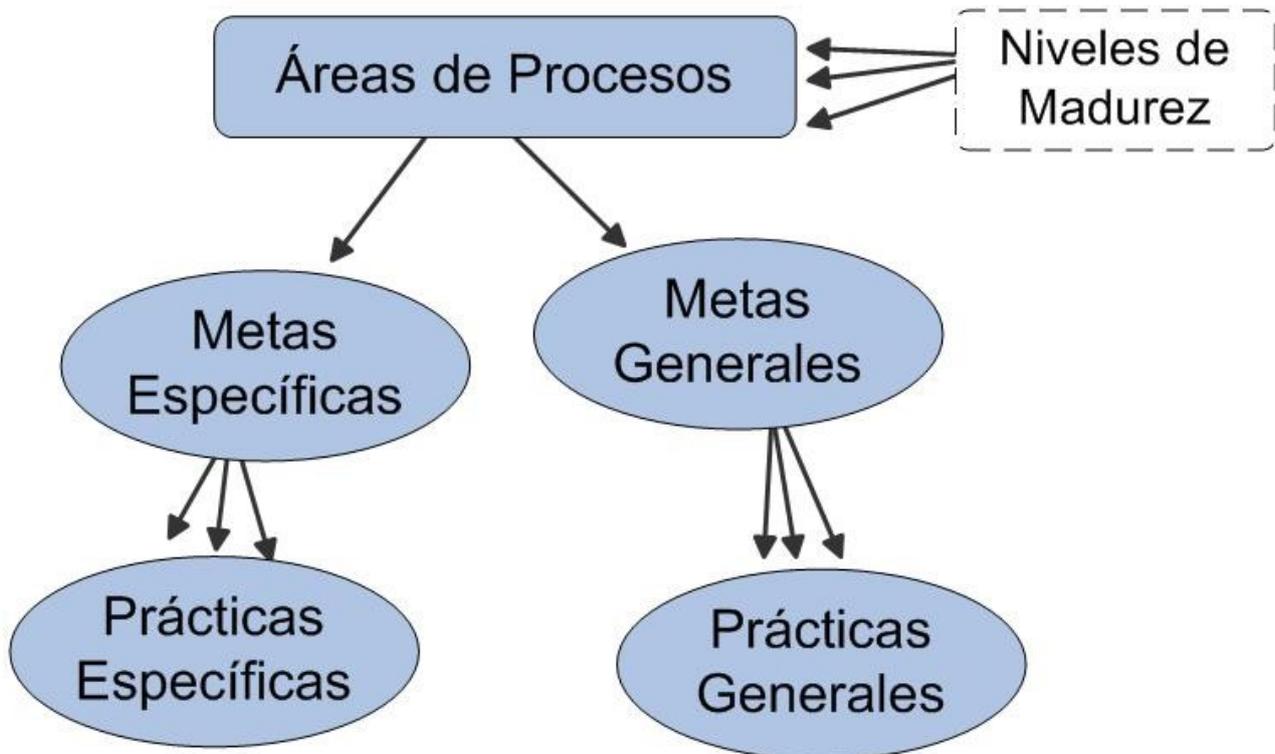
ANEXO 1: MODELO GENÉRICO DE UN AMBIENTE VIRTUAL



ANEXO 2: ESTRUCTURA DE LA REPRESENTACIÓN CONTINUA DE CMMI.



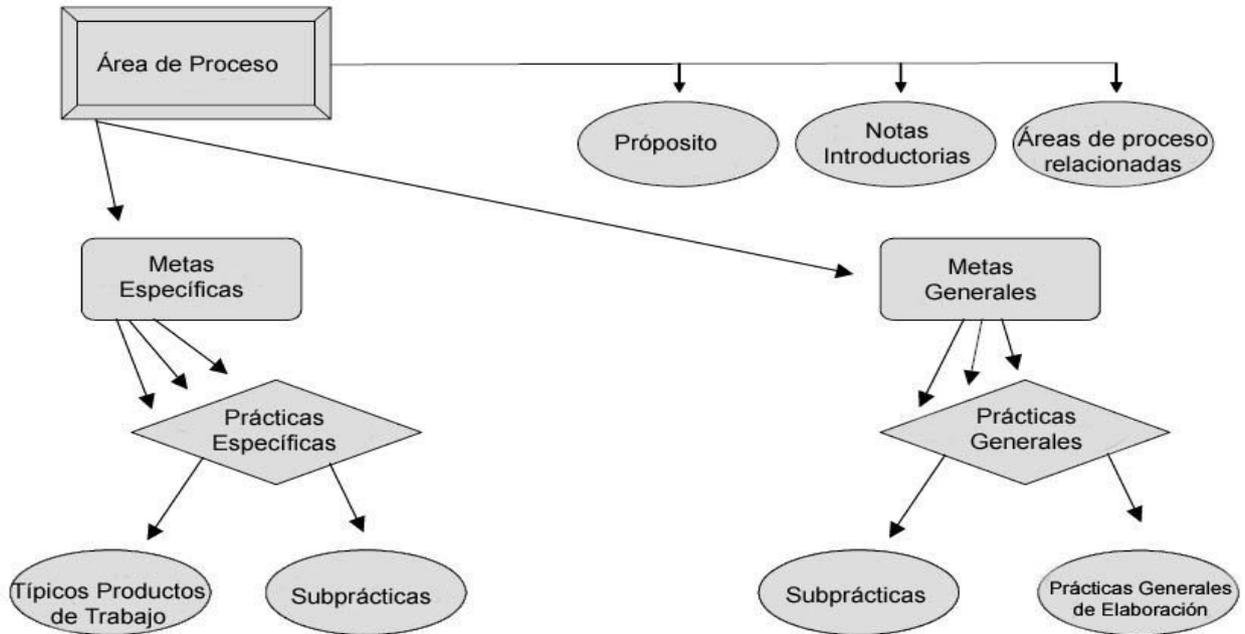
ANEXO 3: ESTRUCTURA DE LA REPRESENTACIÓN ESCALONADA DE CMMI



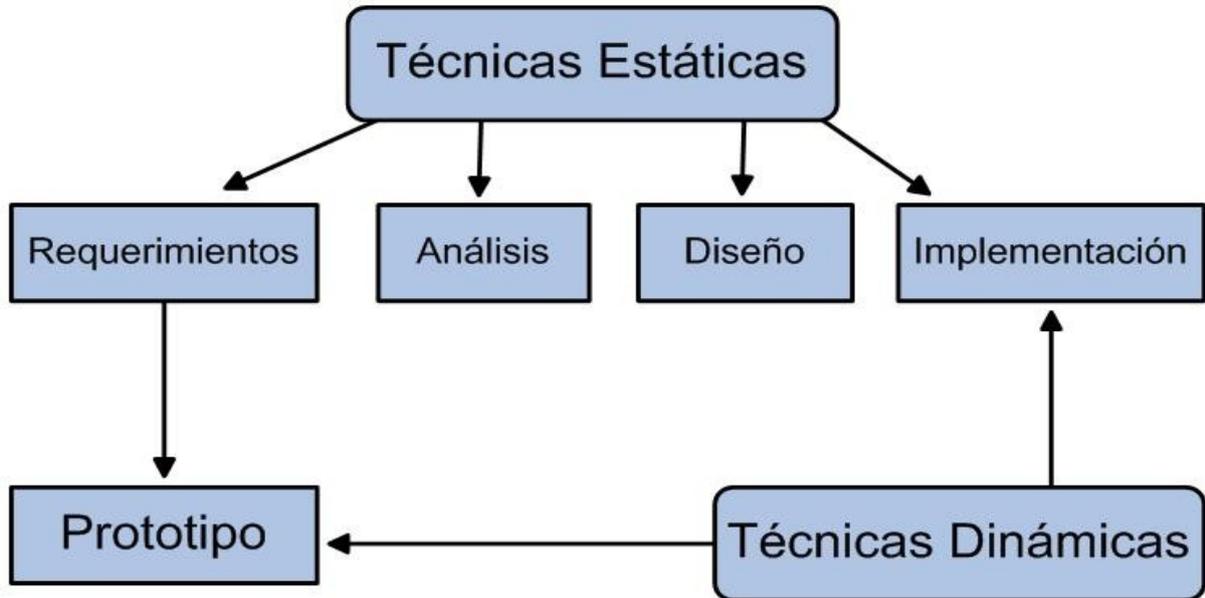
ANEXO 4: NIVELES DE CAPACIDAD Y MADUREZ PRESENTES EN CADA REPRESENTACIÓN DE CMMI

Nivel	Representación Continua Niveles de Capacidad	Representación Escalonada Niveles de Madurez
Nivel 0	Incompleto	
Nivel 1	Ejecutado	Inicial
Nivel 2	Gestionado	Gestionado
Nivel 3	Definido	Definido
Nivel 4	Gestionado cuantitativamente	Gestionado cuantitativamente
Nivel 5	Optimizado	Optimizado

ANEXO 5: ESTRUCTURA DEL MODELO CMMI.



ANEXO 6: TÉCNICAS DE V&V EN EL CICLO DE VIDA DEL SOFTWARE.



ANEXO 7: LISTA DE CHEQUEO SOBRE EL PROCESO DE VALIDACIÓN Y VERIFICACIÓN.

Lista de Chequeo. (Las letras S (si), N (no), P (parcialmente) NP (No procede) R (Referencia con CMMI) para indicar el estado de implementación.)

Proyecto: _____

Líder de Proyecto: _____

Fecha: ____/____/____

Validación y Verificación en proyectos de realidad virtual de la facultad 5 de la UCI						
Sección 1						
Preguntas Generales						
	R	S	N	P	NP	OBSERVACIONES
1. ¿Tienen establecido los objetivos de validación y verificación?						
2. ¿Existe alguien en su proyecto responsable del proceso de validación y verificación? ¿Quién?						
3. ¿El proyecto se rige por alguno de los modelos o normas de calidad existentes (ISO 9000, CMMI, SPICE, IEEE)? ¿Cuáles?						
4. ¿Estos estándares y normas son del conocimiento de los integrantes de los proyectos?						
5. ¿Se proporcionan los recursos adecuados para realizar el proceso?						
6. ¿Se tienen definidas las responsabilidades para llevar a cabo el proceso?						
7. ¿Se realiza una capacitación a las personas involucradas en el proceso?						

Validación y Verificación en proyectos de realidad virtual de la facultad 5 de la UCI						
Sección 1.1						
Prácticas específicas de CMMI v1.2 para Área de Proceso Validación						
	R	S	N	P	NP	OBSERVACIONES
1. ¿Se tiene establecido un plan para realizar la validación? ¿Cuál?	SG1					
2. ¿Se seleccionan los productos de trabajo para la validación? ¿Cómo?	SP 1.1					
3. ¿Se tiene establecido un entorno de validación? ¿Cuál?	SP 1.2					
4. ¿Están establecidos los procedimientos y criterios de validación? ¿Cuáles?	SP 1.3					
5. ¿Se realiza el proceso de validación?	SP 2.1					
6. ¿Se analizan los resultados de la validación? ¿Cómo?	SP 2.2					

Validación y Verificación en proyectos de realidad virtual de la facultad 5 de la UCI						
Sección 1.2						
Prácticas específicas de CMMI v1.2 para Área de Proceso Verificación						
	R	S	N	P	NP	OBSERVACIONES
1. ¿Se tiene establecido un plan para realizar la verificación? ¿Cuál?	SG1					
2. ¿Se seleccionan los productos de trabajo para la verificación? ¿Cómo?	SP 1.1					
3. ¿Se tiene establecido un entorno de verificación? ¿Cuál?	SP 1.2					
4. ¿Están establecidos los procedimientos y criterios de verificación? ¿Cuáles?	SP 1.3					

5. ¿Existe un plan para las revisiones por pares? ¿Cuál?	SP 2.1					
6. ¿Se Realizan revisiones por pares?	SP 2.2					
7. ¿Se analizan los resultados de la revisión? ¿Cómo?	SP 2.3					
8. ¿Se realiza el proceso de Verificación?	SP 3.1					
9. ¿Se analizan los resultados de la verificación? ¿Cómo?	SP 3.2					

ANEXO 8: INFORME FINAL DE REVISIÓN

Identificación de la revisión

Proyecto: [Nombre del proyecto revisado]

Número revisión: [Número de revisión]

Lugar: [Lugar de la revisión]

Fecha: [Fecha de revisión]

Hora: [Hora de la revisión]

Identificación del producto

Material revisado: [Producto o componente del producto revisado]

Desarrollador: [Nombre del desarrollador del material revisado]

Breve descripción: [Descripción del material revisado]

Equipo de revisión

Nombre y apellidos

Firma

1. [Integrante 1]

2. [Integrante 2]

3. [Integrante 3]

Aprobación del producto

- Aceptan el producto (no se hacen modificaciones).
- Rechazan el producto (existen errores graves).
- Aceptan el producto modificándolo (corregir errores).

Material adicional adjuntado

- Plantilla DCS - No conformidades.
- Plantilla DCS - Minuta de reunión.

ANEXO 9: PLAN DE REVISIÓN DE SOFTWARE

Plan de Revisión de Software

<Nombre del Proyecto>

<Nombre del producto>

<Versión>

Control de versiones

Fecha	Versión	Descripción	Autor
<dd/mmm/yy>	<x.x>	<detalles>	<nombre>

Introducción

Propósito

[Resumen del propósito del Plan de Revisión]

Tipo de Revisión

[Especificar tipo de revisión a realizar, ejemplo: Revisión por pares, revisión conjunta entre desarrolladores y clientes, etc.]

Alcance

[Proyectos, productos o componentes del producto con los que se involucra el Plan]

Definiciones, Acrónimos y Abreviaturas

[Definiciones, acrónimos y abreviaturas utilizadas en el documento]

Referencias

[Lista de documentos a los que se hace referencia en el Plan de Revisión]

Código	Título
[1]	Documento 1
[2]	Documento 2

Estándares y Guías

[Lista de los estándares y guías utilizados por el Plan de Revisión]

Resumen

[Resumen de los aspectos del plan]

Gestión**Organización**

[Se describe la estructura de la organización. Especificar cada uno de los roles y responsables de las actividades de la revisión]

Tareas y responsabilidades

Tarea de la Revisión	Responsable	Comentarios
Actividad 1		

Cronograma

[Detalle aquí el cronograma para las revisiones programadas en las fechas principales del proyecto, así como revisiones que son provocadas por la entrega de artefactos del proyecto.]

Tarea de la Revisión	Fecha
Actividad 1	

Entorno de Revisión

[Detalle aquí el entorno de revisión. Deben quedar reflejadas las necesidades de recursos humanos, materiales y demás. Aquí se debe hacer referencia a los cursos de capacitación para el personal involucrado y las listas de chequeo utilizadas]

Procedimiento de Revisión

[Detalle aquí el procedimiento a utilizar]

Sugerencia: *Crear un documento denominado “Procedimiento de Revisiones” donde quede plasmado el procedimiento de revisiones propuesto en ese trabajo y luego en este punto se debe hacer referencia al mismo.*

Resultados de la revisión

[Detalle aquí los resultados finales de revisión, use para esto la Plantilla Informe Final de Revisión]

ANEXO 10: CUESTIONARIO UTILIZADO PARA LA VALIDACIÓN DE LA PROPUESTA

6. ¿Considera usted que la estrategia propuesta para realizar los procesos de validación y verificación se ajusta a las necesidades de los proyectos de realidad virtual?

Si ____ No ____

7. ¿Cree usted que con la aplicación de las técnicas y procedimientos propuestos se logre incrementar la calidad final de los productos de trabajo?

Si ____ No ____

8. ¿Considera usted que son suficientes los elementos propuestos para preparar el entorno de validación y verificación? ¿Agregaría o eliminaría algún otro elemento?

Si ____ No ____

Nota: En caso necesario especifique que elemento agregaría o eliminaría.

9. ¿Considera importante el hecho de que la propuesta esté basada en el modelo CMMI?

Si ____ No ____

10. ¿Qué evaluación le otorgaría a la solución propuesta?

Bien ____ Regular ____ Mal ____

ANEXO 11: RESULTADOS DE LA VALIDACIÓN DE LA PROPUESTA

Para registrar los resultados de la validación de la propuesta se colocará un cero en las celdas donde la respuesta del experto sea negativa y un uno en caso contrario.

Expertos\Preguntas	Pregunta 1	Pregunta 2	Pregunta 3	Pregunta 4	Pregunta 5
Experto 1	1	1	1	1	Bien
Experto 2	1	1	1	1	Bien
Experto 3	1	1	1	1	Bien
Experto 4	1	1	1	1	Bien
Experto 5	1	1	1	1	Bien
Experto 6	1	1	1	1	Bien
Experto 7	1	1	1	1	Bien

GLOSARIO DE TÉRMINOS

AENOR: Asociación Española de Normalización y Certificación.

BSI: Instituto Británico de Normalización (British Standards Institution).

CITMATEL: Empresa de Tecnologías de la Información y Servicios Telemáticos Avanzados.

CMMI/SW: CMMI para Software.

CMMI: Modelo de Madurez de las Capacidades Integrado

CMMI-DEV: CMMI para Desarrollo.

CMMI-DEV+IPPD: CMMI para Desarrollo y para Desarrollo Integrado de Productos y Procesos.

CP: Caso de Prueba.

CU: Caso de Uso.

CVS: Ciclo de Vida del Software.

GG: Objetivos Genéricos.

GP: Prácticas Genéricas.

ICSW: Industria Cubana Del Software.

IEEE: Instituto de Ingenieros Eléctricos y Electrónicos.

ISO: Organización Internacional de Normalización.

NC: Norma Cubana.

OO: Orientado a Objetos.

PSP: Proceso de Software Personal.

RTF: Revisión Técnica Formal.

RUP: Proceso Unificado de Software.

RV: Realidad Virtual.

SEI: Instituto de Ingeniería del Software de la Universidad Carnegie Mellon.

SG: Objetivos Específicos.

SO: Sistema Operativo

SP: Prácticas específicas.

SRV: Sistema de Realidad Virtual.

SVN: Subversión.

UCI: Universidad de las Ciencias Informáticas.

V&V: Validación y Verificación.

VV&A: Validación, Verificación y Acreditación.