Análisis y gestión de riesgos en la simulación de vuelo

Instituto Universitario Aeronáutico

Abstract

El análisis y detección de riesgos en un proyecto de Software es una de los pasos iniciales en el proceso e incluso, en algunos sistemas, lo más importante a tratar, sobre todo si hablamos de sistemas críticos. En el siguiente trabajo, se presentaron y justificaron las causas que clasificarían a los simuladores de vuelo como sistemas críticos, haciendo un prioritario hincapié en los que se denominaron "riesgos cognitivos". A su vez se presentó una metodología de identificación y validación de dichos riesgos para evitar, eliminar o minimizar el impacto que podrían causar en el aprendizaje de manipulación de aeronaves.

Palabras claves
Riesgos, simulador de vuelo, riesgo cognitivo, software, sistema crítico.

Introducción

La gestión de riesgos es una actividad complementaria o de soporte que viene acompañando a la Ingeniería de Software desde su nacimiento. En efecto, la importancia de anticipar, prevenir y/o reducir riesgos fue advertida en épocas muy tempranas de la historia de la computación y no se discute.

Sin embargo, hay numerosos aspectos que deben ser aclarados y para ello lo más conveniente es comenzar por clasificar lo que se entiende por riesgo. Desde una mirada global, hay dos tipos de riesgos:

Riesgos del proyecto: incluye a toda circunstancia que pueda afectar el cumplimiento de los objetivos de un proyecto: impactar la calidad del producto, incrementar su costo o extender su plazo de ejecución. [1]

Riesgos técnicos: se refiere a las consecuencias de que un producto software pueda no haber alcanzado en su elaboración la calidad esperada o prevista. [2]

También pueden agruparse los riesgos según su manifestación o sus consecuencias. En este último caso puede tratarse de resultados erróneos que pueden conducir a importantes pérdidas económicas, daños ambientales o vidas humanas, y aquí el producto software es reconocido como un "*sistema crítico*".[3]

En la mayoría de los casos la asociación de un sistema con su condición *"crítica"* es inmediata: por ejemplo, nadie duda que lo es el software que gobierna el piloto automático de un gran avión de pasajeros. Sin embargo, hay otros casos en que no es evidente que un sistema sea crítico ni que esté asociado a riesgos importantes.

Uno de estos casos es el de los Sistemas de Simulación de Vuelo. La presencia de un defecto en estos sistemas para entrenamiento de pilotos llevará al sujeto entrenado a tener una percepción equivocada del desempeño que puede esperar de su aeronave, y de esta manera, conducirlo a cometer errores de consecuencias fatales cuando se encuentre al comando de un avión real.
Esta condición podría definirse como un ***riesgo cognitivo*** y esto califica como crítico a un sistema que opera sobre un escenario simulado, lo que parece una paradoja ya que en tiempo de simulación el piloto no corre riesgo alguno. Los defectos que conducen al ***riesgo cognitivo*** no provocan fallas de consecuencias fatales, no hay ninguna posibilidad que lo hagan, por el contrario predisponen al piloto a que cometa los errores fatales en una circunstancia

futura. Es decir hay un desfasaje en tiempo respecto al proceso de simulación. Debe observarse que el riesgo cognitivo queda así encuadrado como un caso muy especial de riesgo técnico, y aquí podría rescatarse la idea de que éstos son consecuencia de que el problema es más difícil de resolver de lo que se pensaba o de lo que aparentaba ser.

La originalidad de esta circunstancia estimuló la presentación de este trabajo, cuya organización se lleva a cabo de la siguiente manera:

- Introducción a un marco teórico de riesgos.
- Descripción de la forma en que los simuladores de vuelo son adaptados para reproducir ciertas aeronaves.
- Caracterización de los riesgos cognitivos en comparación con los riesgos operativos a través de una selección de escenarios y condiciones.
- Propuesta de un proceso de identificación de riesgos en estos sistemas y su estrategia de mitigación.
- Resultados obtenidos.
- Presentación de las conclusiones y trabajo futuro previsto.

Por último, está clara la necesidad de trabajar al amparo de procedimientos que liberen en lo posible al desarrollador de cometer errores, y si lo hace, que no pasen inadvertidos.

Marco teórico

A continuación se detallaran algunas de las definiciones de riesgos que ya se mencionaron a modo introductorio.

Riesgos del proyecto: Como ya se mencionó, esto tienen que ver con los riesgos que amenazan el plan del proyecto. Identifican potenciales problemas de presupuestos, calendario, personal, recursos, participantes y requisitos, así como su impacto sobre un proyecto de software [4]. Aquí la Gestión de Riesgos está íntimamente ligada a la planificación de un proyecto, incluyendo de definición y precedencia de actividades, identificación de caminos críticos, cumplimiento de plazos y objetivos parciales y un sinnúmero de circunstancias, muchas de ellas ajenas al proceso y al propio proyecto.

Los riesgos empresariales amenazan la viabilidad del software que se va a construir y con frecuencia ponen en peligro el proyecto o el producto. Por ej. construir un producto o sistema excelente que realmente no se quiere (riesgo de mercado), o construir un producto que ya no encaje en la estrategia empresarial global de la compañía (riesgo estratégico), o tal vez construir un producto que el equipo de ventas no sabe cómo vender (riesgo de ventas).

Riesgos técnicos: Amenazan la calidad y temporalidad del software que se va a producir. Si un riesgo técnico se vuelve una realidad, la implementación puede volverse difícil o impredecible.[5]
A diferencia de los riesgos del proyecto, asociados a circunstancias claramente comprobables y consecuencias inevitablemente visibles, el hecho de que un producto software no haya alcanzado la calidad requerida no es siempre evidente, muchas veces pasa inadvertido, se manifiesta muy tardíamente o no lo hace nunca. Procurando esclarecer el problema aquí caben múltiples agrupamientos o clasificaciones en la referida "falta de calidad". En primer lugar, puede tratarse de: *i*) un defecto introducido involuntariamente, accidentalmente, o *ii*) ser consecuencia de una decisión consciente apoyada en una elección equivocada. Peor aún,

cualquiera sea el caso, puede estar en las diversas etapas del ciclo de vida del producto software: requerimientos, diseño, desarrollo, gestión de configuración o mantenimiento.

Otras clasificaciones de riesgos:

Es extremadamente importante observar que la categorización simple de riesgos no siempre funciona. Algunos de ellos son simplemente impredecibles por adelantado. Otra categorización general de los riesgos es la propuesta por Charette [6]. Los riesgos conocidos son aquellos que pueden descubrirse después de una evaluación cuidadosa del plan del proyecto, del entorno empresarial o técnico donde se desarrolla el proyecto y de otras fuentes de información confiables (por ejemplo, fecha de entrega irreal, falta de requisitos documentados o ámbito de software, pobre entorno de desarrollo). Los riesgos predecibles se extrapolan de la experiencia en proyectos anteriores (por ejemplo, rotación de personal, pobre comunicación con el cliente, disolución del esfuerzo del personal conforme se atienden las solicitudes de mantenimiento). Los riesgos impredecibles son el comodín en la baraja. Pueden ocurrir y lo hacen, pero son extremadamente difíciles de identificar por adelantado. [7]

A su vez, los riesgos pueden clasificarse según su manifestación: *i*) comportamiento crónico no deseado y *ii*) manifestación súbita, o según sus consecuencias: *i*) no funcionamiento, *ii*) bajo rendimiento y *iii*) mal funcionamiento. En este último caso puede tratarse de resultados erróneos que pueden conducir a importantes pérdidas económicas, daños ambientales o vidas humanas, y aquí el producto software es reconocido como un "*sistema crítico*". [8]

Simuladores de vuelo:

Un simulador de vuelo es un sistema que intenta replicar, o simular, la experiencia de volar una aeronave de la forma más precisa y realista posible. Los diferentes tipos de simuladores de vuelo van desde videojuegos hasta réplicas de cabinas en tamaño real montadas en accionadores de movimiento, controlados por sistemas modernos computarizados.

Emular vs Simular: Las palabras simular y emular comparten un objetivo común: la imitación de un sistema complejo con otro sistema.

Definición de simulador: Un simulador es un sistema de software que imita otro sistema complejo, con un nivel variable de realidad. Los simuladores reproducen sensaciones y experiencias que en la realidad pueden llegar a suceder.

Constituyen básicamente un proceso matemático iterativo que se retroalimenta a sí mismo.

Definición de emulador: Los emuladores se limitan a imitar a los sistemas de hardware. En resumen, los emuladores imitan el funcionamiento de un sistema por otro.

Marco metodológico

Todo lo anterior condujo a definir y establecer una metodología de trabajo destinada a evitar defectos en los sistemas de simulación de vuelo, que está centrada en responder a la pregunta clave:"¿Qué características especiales de este producto pueden estar amenazadas por la forma en que se ha planificado el proyecto y por la modalidad con la que se lo concreta?"

En caso de presentarse estos riesgos, una vez identificados y clasificados según su importancia, serán incluidos en la gestión del proyecto para ser tratados como un riesgo más. En efecto, es necesario destacar que la predisposición de los sistemas de simulación a incluir riesgos cognitivos no los exime de

contener riesgos clásicos, igualmente nocivos para el desempeño del producto.

Planteado el problema, el objetivo principal y los objetivos secundarios son:

Objetivo principal:

Reconocer, a través de un proceso sistemático, los posibles ***riesgos cognitivos*** en los sistemas de simulación que se utilizan para realizar entrenamiento y la importancia de identificarlo e incluirlo en la gestión del proyecto para considerarlo como un riesgo relevante.

Objetivos secundarios:

1) Proponer un procedimiento que conduzca a identificar los posibles riesgos que impacten en la calidad del software desarrollado.
2) Asociar a estos riesgos una valoración de las probabilidades de ocurrencia y la importancia de sus consecuencias.
3) Analizar las formas de evitarlos, eliminarlos y/o minimizar sus efectos según sea el caso.
4) Documentar los riesgos identificados, acciones cumplidas y resultados obtenidos.

Para alcanzar los objetivos enunciados se ha previsto la identificación y análisis de los riesgos a través de un proceso sistemático, consistente y estructurado que acompañará al proyecto principal en el desarrollo del software previsto.

Está claro que a los riesgos técnicos que son específicos de un producto en su conjunto, en este caso un dispositivo simulador de vuelo, solo se los puede identificar con un profundo conocimiento del software objeto del desarrollo y del entorno en que opera, en este caso las herramientas de simulación X-plane [9].

Esto hace indispensable la adecuada capacitación y suficiente experiencia del personal técnico participante en el proyecto. Aquí cabe acotar que CMMI presenta la Gestión de Riesgos como una de sus áreas de procesos fundamentales para alcanzar el nivel de madurez 3.

El marco metodológico propuesto, que está ilustrado en la Figura 1, prevé los siguientes seis pasos:

1) Análisis de los planes de vuelo incluidos en el programa de entrenamiento de pilotos e identificación de las condiciones de vuelo que puedan considerarse críticas.
2) Confirmación de las calificaciones críticas asignadas en los escenarios "reales" (avión en vuelo)
3) Implementación y reproducción de las condiciones de vuelo críticas en el "mundo simulado" (simulador de vuelo).
4)

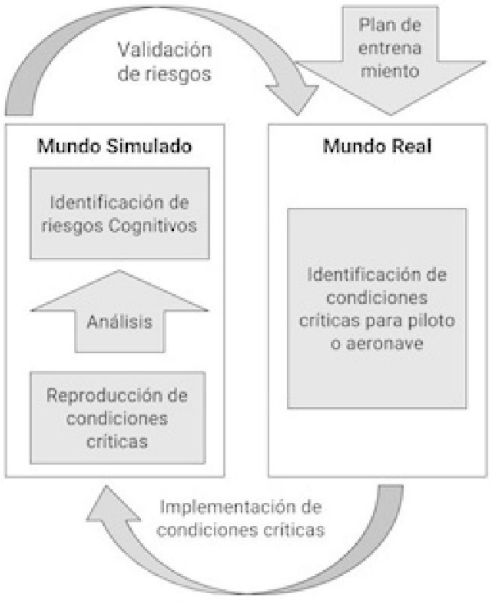

Figura 1: Modelo Metodológico Propuesto

Análisis de estas experiencias de vuelo simuladas para identificar aquellas condiciones críticas que puedan dar lugar a riesgos cognitivos.

5) Confección de una tabla de contingencias que incluirá un listado de todos los riesgos cognitivos que

hayan sido progresivamente identificados. Esta tabla contendrá:

a) Descripción de cada riesgo
b) Categorización,
c) Calificación de la probabilidad de ocurrencia,
d) Entorno en que es más probable que se presente,
e) Condiciones en que se manifestará,
f) Descripción del impacto directo en la funcionalidad del sistema
g) Eventuales efectos indirectos o daño colateral en otras funcionalidades.
h) Condición actual: resuelto, en proceso de resolución y no resuelto,
i) Fecha de esta condición.

6) Validación final de cada riesgo cognitivo identificado en el "mundo simulado" por confrontación con la condición crítica identificada en el escenario de vuelo (mundo real).

Para alcanzar los objetivos enunciados y en el marco del plan de trabajo previsto, la citada tabla de contingencias está destinada a constituirse en el eje del proceso de gestión de riesgos técnicos, permitiendo tanto la actividad específicamente técnica relacionada con cada potencial defecto identificado como también su planificación, seguimiento y control.

Caso de estudio

El ámbito en que se inscribe el caso de estudio es el Centro de Entrenadores y Simuladores de Vuelo (CES), cuya actividad consiste en desarrollar Dispositivos Entrenadores de Vuelo (DEV).

Los DEV son sistemas que tienen por finalidad replicar la experiencia de pilotear una aeronave, de la manera más realista y precisa posible, para lo cual existen diferentes tipos de arquitecturas, todas ellas implementadas a través de complejos sistemas computarizados y plataformas de simulación, sobre las cuales se montan los diferentes desarrollos de componentes de software para estimular y ocasionar las situaciones de vuelo.

Una de las actividades realizadas en CES es el desarrollo del software utilizado para implementar las distintas rutinas de simulación que ejecutan cálculos y conversiones de valores lógicos a unidades de ingeniería, que no están previstas en la plataforma de simulación, como así también la adquisición de datos en tiempo real que son necesarios para la retroalimentación del sistema. Además el software incita, dirige y coordina la plataforma de simulación desde el exterior por el usuario piloto e instructor.

Para poder desempeñar sus funciones, estos dispositivos deben ser configurados y ajustados. El objetivo es reproducir fielmente las características de la aeronave que se está representando en las diversas condiciones de operación, que se especifican según el manual de vuelo de la aeronave provisto por el fabricante de la misma.

Una de las principales exigencias para estos dispositivos es alcanzar una performance en las cualidades de vuelo que se aproximen a las características de la aeronave que es simulada, además de representar también por ejemplo: condiciones de falla, secuencias de maniobras, condiciones meteorológicas, etc.

Distinciones de riesgos dentro del caso planteado

Aunque hay un considerable debate acerca de la definición adecuada de riesgo de software, existe un acuerdo

general en que los riesgos siempre involucran dos características: incertidumbre (el riesgo puede o no ocurrir; es decir, no hay riesgos 100 por ciento probables1) y pérdida (si el riesgo se vuelve una realidad, ocurrirán consecuencias o pérdidas no deseadas [10]).

A diferencia de las categorías presentadas en el marco teórico, el riesgo cognitivo se percibe en que el usuario puede llegar a tomar decisiones incorrectas basándose en el comportamiento de un sistema que no está siendo adecuadamente representado. Este es el caso de un simulador de entrenamiento de vuelo, donde no existe ningún defecto en el funcionamiento del software, pero el mismo evidencia un comportamiento que lleva a que el piloto reciba una percepción equivocada del desempeño de su aeronave, poniendo en riesgo su vida.

Además en el marco de este trabajo, para la adecuada identificación de los riesgos, se considera importante tener presente la naturaleza del software que se está por construir. En este caso se destaca que al tratarse de un sistema crítico en donde entrenar incorrectamente a un piloto pone en riesgo su vida, debería ser importante incluir la categoría planteada en esta investigación: ***el riesgo cognitivo***.

A continuación se muestran algunos ejemplos que amplían la explicación de este concepto:

Caso 1: Vuelo invertido

- Descripción de la condición: El simulador representa la condición de vuelo y todos los recursos donde el piloto puede hacer el vuelo invertido, de acuerdo a la interfaz mostrada, al panel de instrumentos, etc.
- Defecto inadvertido en el entorno de simulación: El simulador no contempla un registro del tiempo límite del vuelo invertido, por lo tanto no activa las alarmas correspondientes.
- Consecuencia: el piloto en un vuelo real cuando en condición de vuelo invertido se ve sorprendido por alarmas es incapaz de interpretarlas ya que nunca se enfrentó con ellas en el proceso de entrenamiento.

Caso 2: Bastón de mandos

- Descripción de la condición:El simulador contiene una réplica del bastón de mandos para realizar las maniobras del avión.
- Defecto inadvertido en el entorno de simulación:La réplica del bastón puede estar mal calibrada, por lo que la fuerza que hay que aplicarle o la sensibilidad del mismo no es la igual que en la realidad.
- Consecuencia: El piloto en un vuelo real puede accionar el bastón de mando con mayor o menor fuerza según la percepción aprendida y esto lo puede llevar a realizar movimientos bruscos en la aeronave que compliquen su estabilidad y generen consecuencias irreversibles.

Algunas situaciones donde se plantea el riesgo cognitivo

Las situaciones que pueden presentar riesgos cognitivos pueden ser:

- Cuando no se encuentra el hardware adecuado para emular el componente, por tanto se debe condicionar el funcionamiento operativo del mismo mediante rutinas de software para recrear la salida requerida. (ejemplo solo se consiguen potenciómetros que giran sin tope es decir dan toda la vuelta y siguen girando, en los

instrumentos reales esto normalmente no sucede ya que poseen un tope, dan media vuelta o vuelta entera hasta el tope, entonces esta condición se debe condicionar por software para establecer hasta donde se establece el tope para que a partir de ese momento no genere más valores de input.)

- Cuando el motor de simulación comercial que se utiliza no contempla parámetros, componentes o sistemas particulares de la aeronave a representar. Entonces se debe generar una rutina de software que simule las condiciones necesarias para representar y retroalimentar los parámetros haciendo funcionar el sistema de manera completamente simulado por fuera del motor de simulación, aunque al momento de ejecución se ejecute como parte de la misma. (ejemplo si necesitamos tener 2 tanques de combustibles en la aeronave y el motor de simulación solo provee uno, se debe hacer todo un componente de software que simule el consumo de combustible de ese tanque y además la potencia extra que tendrán los motores al poseer mayor flujo de combustible)

Estas situaciones presentadas muestran la presencia de posibles riesgos cognitivos. El aporte de este trabajo es destacarlos como tal y aplicar un proceso para su incorporación a la gestión del proyecto. A continuación se describe el proceso propuesto:

Resultados

El trabajo propone un proceso que los autores sugieren aplicar en software de simuladores en los que se identifican riesgos cognitivos, y en las cuales las diferencias entre la realidad y el modelo implementado podrían generar consecuencias en el entrenamiento de personal a cargo de sistemas críticos. De esta manera se pretende manejar esta clase de riesgos con la finalidad de que el diseño que el desarrollo del software de simulación sea lo más cercano posible a la realidad que simula o representa. El proceso descripto en este trabajo se fundamenta en la importancia y la necesidad del tratamiento de los riesgos, específicamente los que se denominaron cognitivos.

Conclusiones

En los sistemas de simulación, y en particular en los simuladores de vuelo, se identificó una forma muy particular de riesgo técnico al que se denominó "riesgo cognitivo". Su principal característica es el desfasaje en el tiempo entre la instancia en que la falla se manifiesta, haciéndolo en el "mundo simulado", y la situación en que se pone en riesgo a la aeronave y/o al propio piloto, lo que ocurrirá en el "mundo real" en un tiempo futuro. La severidad de las consecuencias de estas fallas encuadra a estos sistemas como críticos, a pesar que al manifestarse la falla del sistema el proceso de simulación no permite imaginar riesgo alguno. La necesidad de identificar estas condiciones potenciales de falla de manera objetiva y sistemática llevó a proponer una metodología de trabajo, que se desarrolla necesariamente en ambos mundos: el real y el simulado. De esta manera se evita la búsqueda de riesgos potenciales a través del análisis exhaustivo del sistema de simulación, lo que sería extremadamente laborioso y no garantizaría la ausencia de fallas, que muy probablemente pasarían inadvertidas.

Como próxima etapa se prevé la aplicación de la metodología propuesta a la adecuación de un sistema de simulación de un avión específico, lo

que permitirá validar las etapas del proceso, analizar los resultados y realizar los ajustes necesarios.

Referencias

[1][2][3] Pressman S.Roger ; Ingeniería del software : Un enfoque práctico, 7 edición, University of Connecticut , Pgs 64-68
[4] Pressman S.Roger ; Ingeniería del software : Un enfoque práctico, 7 edición, University of Connecticut , Cap 28.
[5] http://www.actuarios.org/espa/revista21/riesgos.htm- May-2003
[6] Charette, Robert ; Análisis de Riesgos.
[7][8] Pressman S.Roger ; Ingenieria del software : Un enfoque práctico, 7 edición, University of Connecticut , Cap 28.3.2.
[9] http://www.x-plane.com/desktop/home/, 201 Booch, G.; Object Oriented Design with .
[10]Incertidumbre y riesgo: http://www.fao.org/docrep/v8400s/v8400s05.htm