Calidad del software y seguridad de aplicaciones a partir del proceso de desarrollo de software AGILISO y el estándar OWASP

Software quality and application security base on the AGILISO software development process and the OWASP standard

Carlos Arturo Blandón-Jaramillo[1], Jhon Steven Jaramillo-Becerra[2]

Blandón-Jaramillo, C.A; Jaramillo-Becerra, J.B. Calidad del software y seguridad de aplicaciones a partir del proceso de desarrollo de software agiliso y el estándar OWASP. *Tecnología en Marcha*. Vol. 36, número especial. Octubre, 2023. V Congreso Internacional en Inteligencia Ambiental, Ingeniería de Software, Salud Electrónica y Móvil. Pág. 5-22.

https://doi.org/10.18845/tm.v36i8.6923

1 Progrezando.com, Colombia.
Correo electrónico: carlos.blandon@progrezando.com
https://orcid.org/0000-0002-6179-8795
2 Universidad de Caldas, Colombia.
Correo electrónico: jhonstevenjaramillo25@gmail.com
https://orcid.org/0000-0001-8409-0012

Palabras clave

Calidad de Software; seguridad; procesos ágiles; desarrollo de software; auditoría de sistemas.

Resumen

La globalización ha impulsado a todos los sectores industriales hacia la modernización de la obtención, almacenamiento y acceso de información en los procesos de apoyo, misionales y estratégicos, modernización que se han comenzado a hacer prácticamente obligatorios e inmediata a partir de la declaratoria mundial de pandemia, que obligó a que dichos procesos se realicen en la virtualidad toda vez que por parte de los gobiernos se decretaron confinamientos a toda la población; esta circunstancia inesperada decanta en la necesidad imperativa de mejorar tanto las prácticas de desarrollo de software como las pruebas a la seguridad de las aplicaciones que soportan la operación del negocio. En este contexto los responsables de los departamentos de control interno y auditoría de sistemas de información deben generar evaluaciones permanentes tanto a los procesos de desarrollo de software como a la seguridad de las aplicaciones, asegurando el cumplimiento de los estándares internacionales ISO/IEC 27001 e ISO/IEC 29110, verificando que la lógica del negocio este soportada de manera adecuada a través de los desarrollos propios o tercerizados con los que cuentan las organizaciones.

Esta es una propuesta para evaluar la calidad del software a partir del proceso de desarrollo de software AGILISO y la seguridad en las aplicaciones en base al estándar de verificación de seguridad en aplicaciones OWASP, fortaleciendo y optimizando la actividad de auditoría por parte de control interno, auditores y consultores de sistemas de información, permitiendo la propuesta oportuna de planes de acción que procuren la corrección de las desviaciones detectadas.

Keywords

Software quality; security; agile processes; software development; system audit.

Abstract

Globalization has driven all industrial sectors towards the modernization of obtaining, storing and accessing information in the support, mission and strategic processes, modernization that have started to become practically mandatory and immediate after the world pandemic declaration, which forced these processes to be carried out virtually since governments decreed confinements to the entire population; this unexpected circumstance leads to the imperative need to improve both software development practices and security testing of the applications that support the business operation. In this context, those responsible for internal control and information systems auditing departments must generate permanent evaluations of both software development processes and application security, ensuring compliance with international standards ISO/IEC 27001 and ISO/IEC 29110, verifying that the business logic is adequately supported by the organizations' own or outsourced developments.

This is a proposal to evaluate software quality based on the AGILISO software development process and application security based on the OWASP application security verification standard, strengthening and optimizing the auditing activity by internal control, auditors and information systems consultants, allowing the timely proposal of action plans that seek to correct the deviations detected.

Introducción

En los últimos años las tecnologías de información han impulsado un proceso de globalización de gran escala que ha impactado directamente a todos los sectores industriales; como consecuencia directa de dicho proceso, la tecnología se ve avocada a ser más eficiente, segura, confiable y "a brindar la sensación de ubicuidad, requiriendo desarrollo de software de calidad que cumpla con los requisitos del mercado, por tanto, los desarrolladores requieren más control sobre recursos, tiempo, costo y calidad del producto" [5]. La falta de adopción de estándares internacionales para el desarrollo adecuado de proyectos de software en conjunción con la formación en procesos ágiles representa una gran debilidad en esta industria [9].

La declaratoria mundial de pandemia por la COVID - 19 ha acelerado los procesos de transformación digital gubernamental y empresarial [3], a partir de la necesidad de adopción de estrategias para atención al cliente y/o usuarios que involucran desarrollo acelerado de aplicaciones y protocolos de seguridad que garanticen tanto la estabilidad del servicio como la protección de la información, generando "una revolución en los procesos de producción y mejoran los estándares de vida, sobre todo en los países en desarrollo" [12].

Se espera que la evaluación sistemática y oportuna de las fases que componen el desarrollo de software tanto In House como a través de Outsourcing así como de los estándares de seguridad de las aplicaciones permita la toma de decisiones oportunas y prevengan el incremento de costos asociados a reprocesos y/o incumplimientos con los acuerdos de nivel del servicio - ANS del negocio.

Proceso de desarrollo de software

Es un conjunto de actividades y resultados asociados que producen un producto de software ver Figura 1. Existen cuatro actividades fundamentales de procesos que son comunes para todos los procesos del software: especificación del software, diseño e implementación del software, validación del software y evolución del software [17].

Así mismo, la IEEE lo define como "un conjunto de actividades y tareas interrelacionadas que transforman entradas de trabajo en productos de salida. Como mínimo, la descripción de un proceso de software incluye entradas, transformación de actividades y resultados generados" [8].

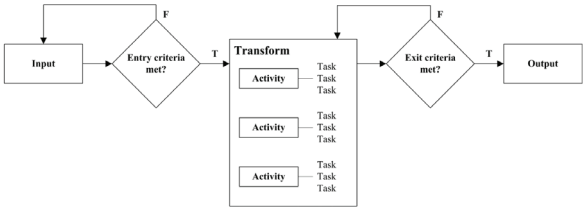


Figura. 1. Elementos de un proceso de software [8].

Constituye entonces un marco de trabajo que se usa para planificar y controlar el proceso de desarrollo de sistemas de información [10].

Procesos ágiles

Los procesos ágiles se basan en cinco fases, que definen el ciclo de desarrollo ágil, ver figura 2

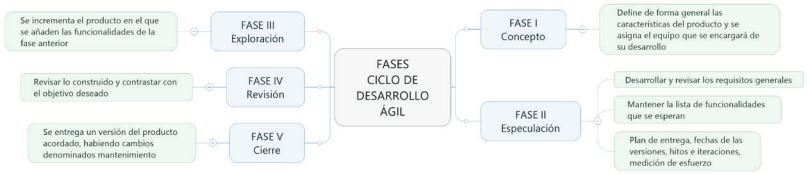


Figura. 2. Fases que definen el ciclo de desarrollo ágil [5].

Desarrollo ágil es un término derivado del Manifiesto Ágil, escrito en 2001, por un grupo que incluía a los creadores de Scrum, Extreme Programming (XP), Dynamic Development Method (DSDM) y Crystal [11].

Extreme Programming XP

Es un proceso de desarrollo ágil propuesto por Kent Beck en 1999, ver figura 3, cuyo énfasis está puesto más en la adaptabilidad que en la previsibilidad [1][21].

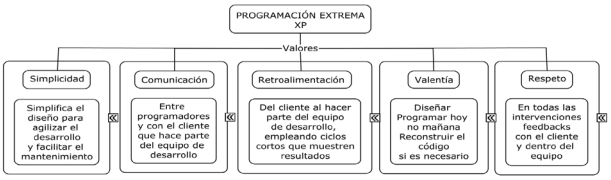


Figura. 3. Valores programación extrema XP [5].

La programación extrema se desarrolla por fases definidas a lo largo del ciclo de vida del proyecto de software, ver figura 4.

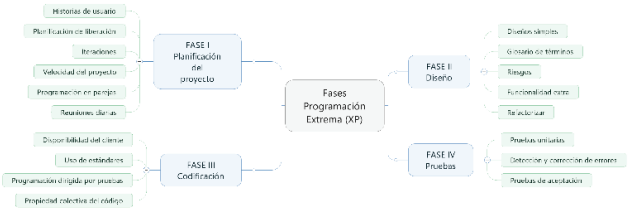


Figura. 4. Fases de XP [5].

Scrum

Scrum se basa en la teoría del control de procesos empírica o empirismo, asegurando que el conocimiento procede de la experiencia y de tomar decisiones basándose en lo que se conoce [16]. Dicha implementación del control de procesos empírica se basa en tres pilares:

- Transparencia: los aspectos significativos del proceso deben ser visibles para aquellos que son responsables del resultado [16].
- Inspección: los usuarios Scrum deben inspeccionar frecuentemente los artefactos y el progreso hacia un objetivo, para detectar variaciones. Su inspección no debe ser tan frecuente para que interfiera en el trabajo [16].
- Adaptación: si se determina que uno o más aspectos de un proceso se desvían de límites aceptables, y que el producto resultante no será aceptable, el proceso o el material que está siendo procesado deberá ser ajustado [16].

Scrum organiza el ciclo de desarrollo en tres fases, las cuales constituyen artefactos metodológicos comprendidos como reuniones, ver figura 5.

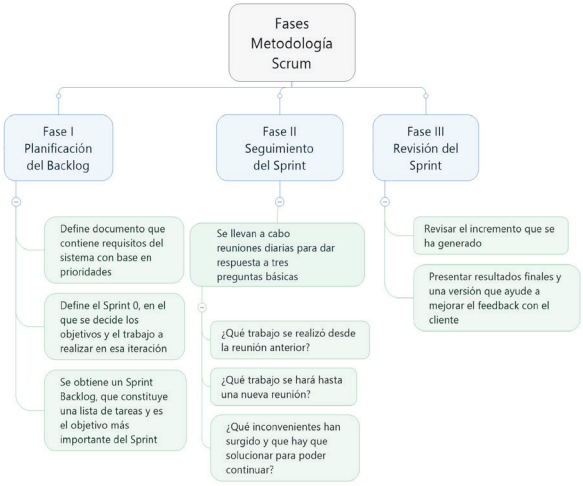


Figura. 5. Fases de Scrum [5].

OpenUp

Hace parte del marco de trabajo de proceso de Eclipse (EPF), manteniendo las características esenciales de RUP – Rational Unified Process. “Es un proceso ágil y liviano, que aplica enfoque iterativo e incremental dentro de un ciclo de vida estructurado y contiene un conjunto mínimo de prácticas que ayuda al equipo a ser más efectivo desarrollando software [6]. OpenUp se encuentra gobernada por cuatro principios fundamentales, ver figura 6:

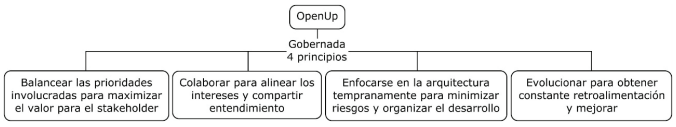


Figura. 6. Principios de OpenUp [5].

En cada una de las fases de OpenUp, pueden existir varias iteraciones, desde las primeras OpenUp se enfoca en el tratamiento de riesgos, generando reuniones que permiten definir los controles que permitan su mitigación, ver figura 7.

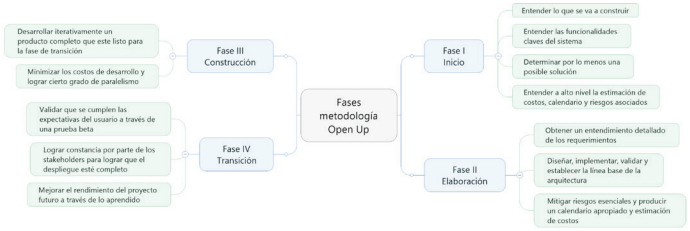


Figura. 7. Fases de OpenUp [5].

Kanban

Kanban se basa en la idea de que el trabajo en curso debería limitarse, y sólo deberíamos empezar con algo nuevo cuando un bloque de trabajo anterior haya sido entregado o ha pasado a otra función posterior de la cadena [10]. Kanban tiene los siguientes principios, ver figura 8:

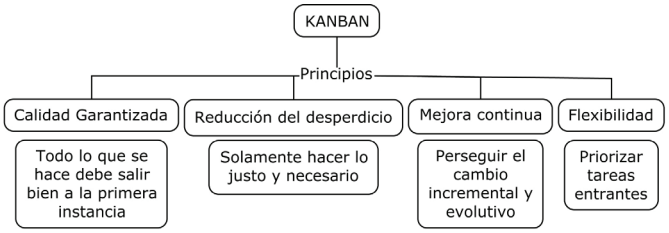


Figura. 8. Principio de Kanban [5].

La aplicación del proceso Kanban, tiene en cuenta los siguientes aspectos, ver figura 9:

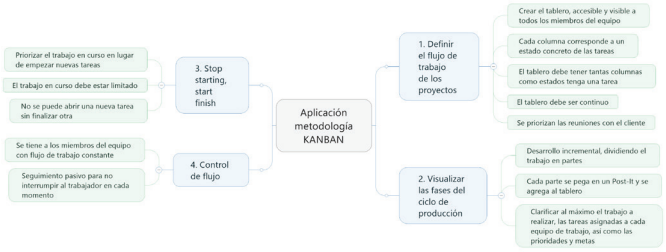


Figura. 9. Aspectos para la aplicación de Kanban [5].

Proceso de software personal PSP

De acuerdo con [20] PSP “provee a los ingenieros un marco de trabajo para desarrollar las actividades de software de manera disciplinada”, se encuentra conformado por un conjunto de métodos, formularios y scripts que les permiten a los profesionales planear, medir y administrar su trabajo, en búsqueda de generar productos con cero defectos [18]. Los principios de planificación y calidad en los cuales se basa PSP, ver figura 10:

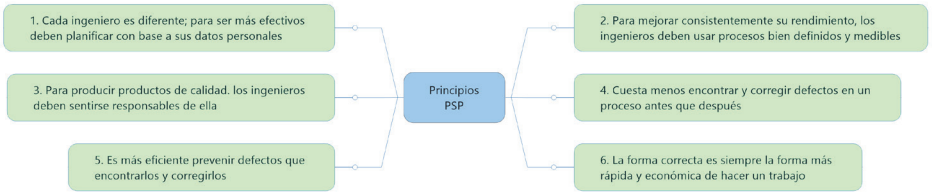


Figura. 10. Principios de PSP [5].

Cuadro 1. Niveles de mejoramiento de psp.

Nivel	Nombre	Actividad
PSP0	Medición personal	Registro de tiempo Registro de defectos
		Patrón de tipos de defectos Patrón de codificación Medida de tamaño Propuesta de mejoramiento de procesos
PSP1	Planeamiento personal	Estimación de tamaño Informe de pruebas Planeamiento de tareas Cronogramas
PSP2	Gerenciamiento de la calidad personal	Revisiones del código Revisiones del proyecto Patrones del proyecto
PSP3	Proceso personal cíclico	Desarrollo cíclico

ISO 29110:2014

Contiene los perfiles del ciclo de vida para las pequeñas entidades (PEs) dedicadas al desarrollo de software. La serie ISO/IEC 29110, ha sido desarrollada para mejorar la calidad de los productos y servicios, y el desempeño de los procesos. No es su intención evitar el uso de diferentes ciclos de vida como: cascada, iterativo, incremental, evolutivo o ágil [7]. Esta norma está compuesta por cinco partes de acuerdo con el público objetivo:

Cuadro 2. Público objetivo de la norma iso/iec 29110:2014.

ISO/IEC 29110:2014	Título	Público objetivo
Parte 1	Visión general	Pequeñas entidades (PEs), productores de normas, proveedores de herramientas y proveedores de metodologías
Parte 2	Marco de trabajo y taxonomía	Productores de normas, proveedores de herramientas y metodologías
Parte 3	Guía de evaluación	Asesores y Pequeñas entidades (PEs)
Parte 4	Especificaciones de perfil	Productores de normas, proveedores de herramientas y metodologías
Parte 5	Guía de gestión e ingeniería	Pequeñas entidades (PEs)

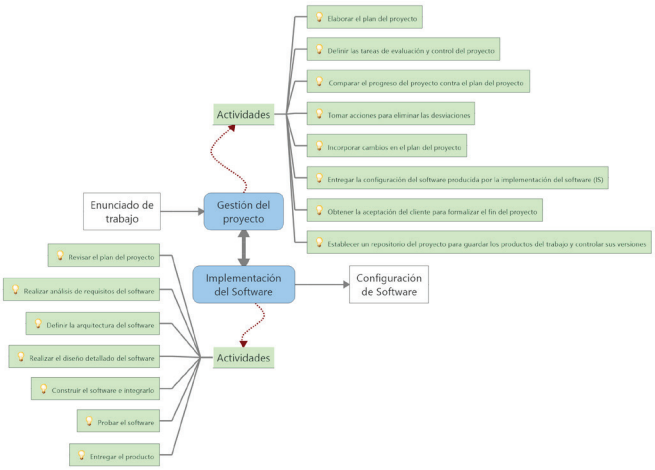


Figura 11. Procesos de la guía del perfil básico ISO/IEC 29110:2014 [5].

La norma proporciona los procesos de gestión de proyecto e implementación de software, las cuales integran prácticas basadas en la selección de elementos de la ISO/IEC 12207:2008 y la ISO/IEC 15289:2006 [2]. Las pequeñas entidades pueden establecer los mecanismos necesarios para implementar cualquier proceso de desarrollo incluyendo los ágiles. Se encuentra conformada por los procesos mostrados en la figura 11.

AGILISO

Identificar y lograr apropiar las mejores prácticas es de vital importancia para obtener una ventaja competitiva; tal como decía Szulanski “Una práctica es un método o técnica que se emplea para realizar una parte de un proceso y describe como se realiza. Las mejores prácticas permiten incrementar la satisfacción del cliente al incorporar su uso en nuestro proceso” [19]. El Proceso de desarrollo de software AGILISO fue propuesto por [5] luego del análisis de las mejores prácticas de diversas metodologías ágiles, procurando establecer aquellas que permitirían el desarrollo unipersonal de productos de software, los procesos analizados fueron: Extreme Programming, Scrum, OpenUp, Kanban, PSP, ISO/IEC 29110:2014. El proceso de desarrollo de software AGILISO se resume en la figura 12

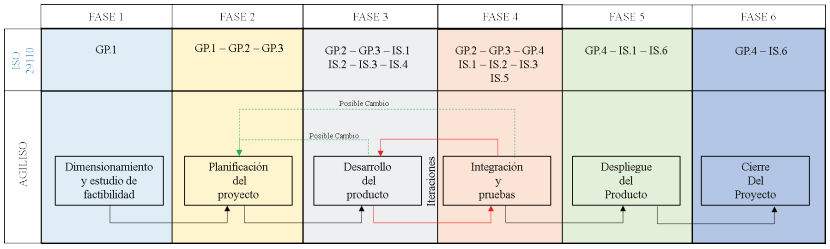


Figura. 12. Proceso de desarrollo de software unipersonal propuesto [5].

AGILISO consta de seis fases, las cuales a su vez se acoplan con los objetivos de la norma ISO 29110:2014 [5]

Fase I – dimensionamiento y estudio de factibilidad

Documenta y permite el análisis de las siguientes condiciones del proceso de desarrollo de software: tiempo de entrega, alcance, estimación del esfuerzo. Una vez acordados los puntos anteriormente mencionados se constituye el acta de inicio del proyecto y se formaliza el contrato. [5]

Fase II – planificación del proyecto

El desarrollador deberá iniciar el proceso de documentación de los requisitos, procederá a generar un plan de entregas que será acordado con el cliente y la planificación para las iteraciones del proceso.

Fase III – desarrollo del producto

Da cumplimiento al plan de iteraciones, teniendo en cuenta antecedentes técnicos, normativos, sistemas de información relacionados, interesados en la iteración, requerimientos funcionales, no funcionales, de interoperabilidad, de infraestructura, de seguridad, arquitectura general de la iteración, contrataciones requeridas, restricciones, oportunidades, riesgos. Una vez finalizada la etapa de codificación, se implementan pruebas unitarias [5].

Fase IV – integración y pruebas

El desarrollador debe integrar el producto, a fin de someterlo a pruebas finales que permitan evaluar su comportamiento una vez integrados todos los componentes [5].

Fase V – despliegue del producto

El desarrollador inicia las actividades pertinentes para poner en funcionamiento el producto terminado en el escenario real en el cual debe emplearse, teniendo presentes todas las actividades de origen técnico a que hubiere lugar. [5]

Fase VI – Cierre del proyecto

Se realizan actividades de capacitación en el manejo del producto, así mismos debe dejarse constancia de las personas que en dicha actividad intervienen; generando finalmente un acta de entrega que da fe del recibido a satisfacción por parte del cliente. [5]

Mejores prácticas del proceso unipersonal propuesto

El proceso de desarrollo unipersonal propuesto adopta las siguientes mejores prácticas

Esquema documental base del proceso de desarrollo AGILISO

La documentación está conformada por 6 grupos de acuerdo con las fases del proceso, ver figura 13.

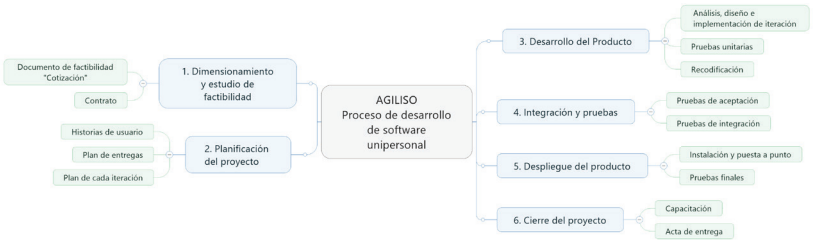


Figura. 13. Esquema documental base [5].

OWASP – Estándar de seguridad en aplicaciones

El Open Web Application Security Project – OWASP es un proyecto de código abierto dedicado a determinar y combatir las causas que hacen que el software sea inseguro [13].

OWASP define tres niveles de verificación de seguridad, ver figura 14, incrementando la profundidad en cada nivel [14]:

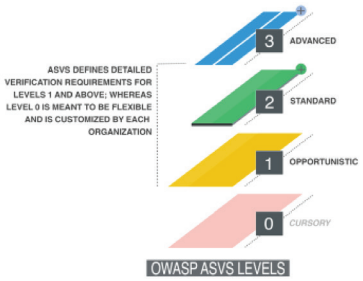


Figura. 14. Niveles de verificación de seguridad [14].

- ASVS Nivel 1: Dirigido a todo tipo de software.
- ASVS Nivel 2: Dirigido a aplicaciones que contienen datos sensibles, que requieren protección.
- ASVS Nivel 3: Dirigido a las aplicaciones más críticas que realizan transacciones de alto valor, contienen datos médicos confidenciales, o cualquier aplicación que requiera el más alto nivel de confianza.

Cuadro 3. Requisitos de verificación detallada OWASP.

Requisito	Descripción	Requisito	Descripción
V1	Arquitectura, diseño y modelado de amenazas	V11	Configuración de seguridad HTTP
V2	Autenticación	V12	Configuración de seguridad
V3	Gestión de sesiones	V13	Controles maliciosos
V4	Control de accesos	V14	Seguridad interna
V5	Manejo de entrada de datos maliciosos	V15	Lógica de negocios
V6	Codificación/Escape de salida de datos	V16	Archivos y recursos
V7	Criptografía en el almacenamiento	V17	Móvil
V8	Gestión y registro de errores	V18	Servicios WEB
V9	Protección de datos	V19	Configuración
V10	Seguridad en las comunicaciones		

De acuerdo con [22] es de vital importancia que los servicios expuestos en internet, que no solo ofrecen transaccionalidad online para las organizaciones de todo tipo, sino que, además son su imagen y buen nombre estén asegurados desde su desarrollo con metodologías claras frente a la creación y el testeo de aplicaciones antes de su salida a producción y con pruebas periódicas en busca de vulnerabilidades o fallas que puedan representar riesgo para el correcto funcionamiento de las operaciones del negocio. A partir de este estándar se desarrollan pruebas de verificación de seguridad que permiten establecer acciones oportunas a las vulnerabilidades detectadas.

Evaluación de calidad del software a partir del proceso de desarrollo agiliso y la seguridad de las aplicaciones OWASP

Como respuesta a la necesidad manifestada por los profesionales de consultoría en implementación de sistemas de gestión ISO/IEC 27001, se diseña un instrumento que apoya la evaluación de la calidad del software desde el proceso de desarrollo, para lo cual se empleó como referente el propuesto por [5], AGILISO y el estándar para verificación de requisitos de seguridad en aplicaciones OWASP. La herramienta fue construida en Microsoft Excel, conformada por ocho hojas de cálculo, las cuales se distribuye de la siguiente manera:

- Fase I a VI: Fases del proceso de desarrollo de software AGILISO
- Fase VII: Verificación de requisitos de seguridad en aplicaciones
- DashBoard: Resultados de la verificación de requisitos para las Fases I – VII

El instrumento se estructuró de tal manera que fuese posible registrar allí la información recolectada por el responsable de adelantar el proceso de revisión o auditoría a los sistemas de información, de tal manera que se catalogaron los requisitos y/o artefactos en dos componentes principales:

- Generalidades del artefacto: contiene especificaciones generales de existencia del artefacto.
- Componentes del artefacto: contiene requisitos específicos del artefacto.

ARTEFACTO: DOCUMENTO DE FACTIBILIDAD - COTIZACIÓN								
GENERALIDADES DEL ARTEFACTO								
PREGUNTAS	Si	No	N/A	Fecha	Entrevistado(s)	Requisito(s)	Observaciones y/o evidencia	PUNTAJE
¿Existe un documento que reune las características y condiciones generales del proyecto de desarrollo de software y que ha sido la base para la negociación con el proveedor?								
¿Fue socializado apropiadamente el documento y se generó acta de entendimiento de los alcances del proyecto por parte del proveedor?								
TOTAL	0%							

Figura. 15. Distribución de los campos del instrumento de evaluación.

Como se observa en la figura 15, el instrumento contiene los siguientes ítems:

- Preguntas: columna que contiene los cuestionamientos específicos del artefacto.
- Si/No/NA: Columnas para consignar la respuesta de las personas entrevistadas o en su defecto la no aplicación de la pregunta al contexto en el cual se aplica.
- Fecha: Columna para el registro de la fecha en la cual se realiza la indagación.
- Entrevistado: Columna que registra el nombre del responsable por la organización de suministrar la información para la evaluación
- Requisito: Columna que identifica los requisitos normativos y/o reglamentarios al cual se orienta el cuestionamiento.
- Observaciones y/o evidencia: Columna que permite registrar los hallazgos de la persona responsable de adelantar el proceso de evaluación.
- Puntaje: Columna que refleja de manera automática los resultados arrojados de conformidad con la información recolectada por el encargado de la evaluación.

Se obtiene un porcentaje de cumplimiento por componente de artefacto y por artefacto, de igual manera permite relacionar el porcentaje de cumplimiento total por Fase. En virtud de dichos resultados se estableció una interpretación sugerida, que siempre puede ser modificada por la organización de acuerdo con sus políticas internas.

Cuadro 4. Interpretación sugerida instrumento de evaluación fases I a VI.

% Obtenido	Interpretación sugerida
>98	Se llevan a cabo actividades adecuadas para garantizar la calidad del software en la Fase X
Entre 80 y 97	Se hace necesario ajustar la documentación con el proveedor, la información de la Fase X se debe complementar a fin de poder establecer nuevos contratos
Entre 70 y 79	Es necesario que el Proveedor presente un plan de acción tendiente a garantizar la calidad del software desde la Fase X
<70	Se debe replantar la continuidad de contratación con el proveedor

La fase VII de verificación de requisitos en la seguridad de aplicaciones, presenta los resultados en gráfico de barras identificando los hallazgos a la seguridad, no se presenta interpretación sugerida dado que se pretende la identificación de vulnerabilidades que permitan la propuesta de un plan de acción que corrija la situación presentada, ver figura 16.

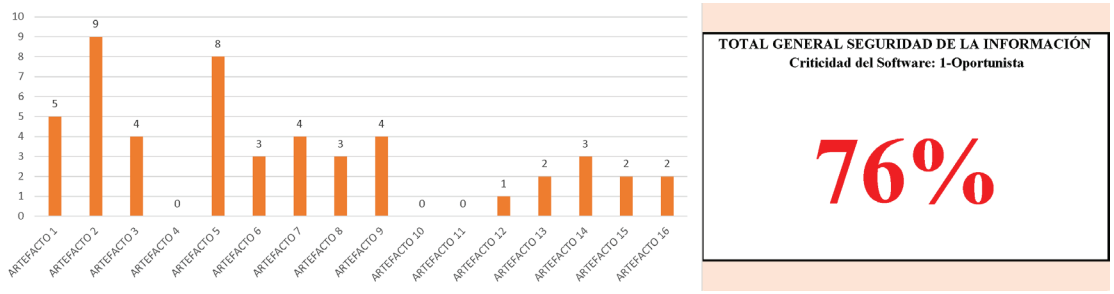


Figura. 16. Resultados DashBoard Fase VII.

Pruebas al instrumento de evaluación

Se empleó para las pruebas la metodología Design Science Research (DSR) "Ciencia del diseño" [15], de acuerdo con el modelo sugerido por [4], para lo cual se aplicaron pruebas de concepto, valor y uso.

Concepto

Se socializó el instrumento con profesionales de una Cooperativa de servicios financieros de la Ciudad de Cuenca Ecuador, se obtuvo a través de proceso de grupo focal retroalimentación que permitió mejorar el diseño inicial del instrumento de evaluación, incorporando elementos importantes tanto en la recolección de información como en la interpretación y muestra de los resultados obtenidos, alineando dichos informes de resultados a un uso estratégico por parte de los responsables de las áreas de Tecnologías de Información – TI y la Alta Dirección.

Valor

Se definió como piloto un caso de uso consistente en la evaluación de un sistema de información en proceso de desarrollo por parte del gerente de una empresa de desarrollo de software, se estableció como actividades específicas la aplicación del instrumento en desarrollos de software nuevos de pequeña envergadura.

Esta prueba se aplicó a dos desarrolladores con el mismo piloto de desarrollo de software, uno de los cuales empleó su proceso habitual y el otro incorporó procesos de evaluación continua por fases a los productos bajo su responsabilidad, una vez finalizadas las pruebas se aplicaron las métricas de: esfuerzo, cubrimiento de requisitos, porcentaje de reprocesos y defectos, arrojando los siguientes resultados:

Cuadro 5 Pruebas de valor.

Métricas	Con evaluación	Sin evaluación
Esfuerzo	35 h/h	30 h/h
Cubrimiento de requisitos	100 %	100 %
Porcentaje de reprocesos	5 %	15 %
Defectos encontrados	2	10

Uso

Con el objetivo de determinar si los resultados de la aplicación del instrumento de evaluación propuesto eran los esperados se emplearon las siguientes métricas: cantidad de hallazgos, costos de desarrollo, fallas de requerimientos descubiertas, vulnerabilidades descubiertas, planes de acción requeridos

Cuadro 6. Pruebas de uso.

Métrica	Con evaluación	Sin evaluación
Cantidad de hallazgos	21	2
Costo de desarrollo	$550.000	$850.000
Fallas de requerimientos descubiertos	0	0
Vulnerabilidades descubiertas	12	0
Planes de acción requeridos	33	2

La implementación de los requisitos propuestos en el caso de uso se llevó a cabo aplicando el instrumento de evaluación de la calidad del software a partir del proceso de desarrollo aplicado por parte del equipo de desarrollo de software (Fases I a VI) y el estándar de verificación de requisitos de seguridad en aplicaciones OWASP (Fase VII), la persona encargada de coordinar la aplicación del instrumento de evaluación en la organización fue el encargado de la Coordinación del Departamento de Tecnologías de Información en Apoyo del Oficial de Seguridad ISO/IEC 27001, quienes manifestaron la importancia de contar con un manual de usuario que permita guiar a los responsables de la evaluación en su aplicación, así mismo, se realizó un ejercicio de capacitación acerca de los conceptos específicos que deberían tenerse en cuenta en la aplicación del instrumento.

Se hizo énfasis particularmente en la criticidad de las aplicaciones para poder ejecutar de manera adecuada la Fase VII, para evaluar dicha criticidad se recomienda que se tengan presentes las directrices contenidas en el estándar internacional ISO/IEC 27001, de tal forma que sea posible aplicar la verificación de requisitos de manera coherente con la lógica del negocio.

El proceso de evaluación requiere que se tengan presentes los principios del código deontológico del auditor principalmente aquellos relacionados con la objetividad, confidencialidad y la información suficiente, toda vez que lo que busca el instrumento es el beneficio de la organización y la implementación de controles de calidad y seguridad que permitan garantizar adecuadamente la alineación de los productos de software con la estrategia de negocio y en consecuencia aportar a la continuidad del mismo.

Se presentó el Dashboard como instrumento para la presentación de información a la Alta Dirección, así como a entidades de cumplimiento, dicho reporte contiene elementos fundamentales que permiten determinar grados de cumplimiento, se condensa allí información sobre: Cantidad de artefactos, Calificación obtenida en cada artefacto, Total obtenido en cada fase, Cantidad de preguntas realizadas, Cantidad de cumplimientos, Cantidad de incumplimientos, Cantidad de preguntas que no aplican, Gráfico distribución porcentual de los artefactos, ver figura 17.

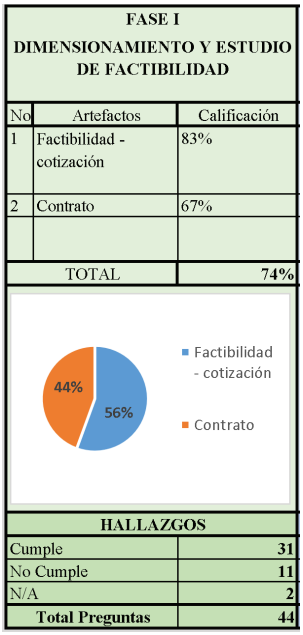


Figura. 17. Resultados DashBoard Fase I a VI.

Se hace un proceso de valoración sobre los resultados de incumplimiento de tal manera que es posible visualizar aquellas fases del proceso de desarrollo que presentan mayores debilidades y que requieren planes de acción para poder atender oportunamente las falencias encontradas, permitiendo priorizar los aspectos del proceso con el equipo de profesionales que hacen parte del proyecto de desarrollo, ver figura 18.

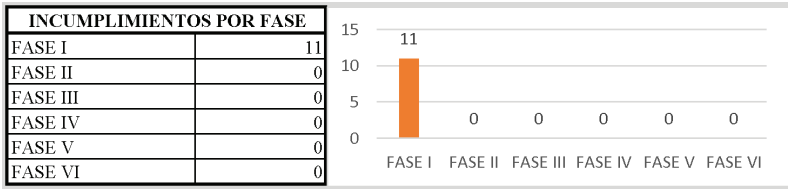


Figura. 18. Representación de incumplimientos por fase.

La aplicación del instrumento requiere que los responsables de su aplicación tengan conocimientos en procesos de desarrollo de software y en análisis de vulnerabilidades, así como en pruebas de penetración y testeo. Durante su aplicación fue posible evidenciar la necesidad de que en dicho proceso participara un equipo de profesionales que complementarán el conocimiento técnico requerido para el diligenciamiento adecuado de la evaluación.

Uno de los principios en términos de auditoría y que se sugiere sea aplicado por parte del responsable de evaluar la calidad del software y la seguridad en aplicaciones es la objetividad, es importante notar que fue posible evidenciar procesos de desarrollo organizacionales con deficiencias documentales en particular en las relaciones contractuales que no se encontraban diseñadas en beneficio de la empresa y que requerían un proceso de análisis profundo por parte de la alta gerencia y el área jurídica, de igual manera se pudo constatar que las empresas encargadas de desarrollos para la organización en modalidad Outsoursing no disponían en su totalidad con la documentación requerida por el instrumento de evaluación generando hallazgos importantes, dentro de los cuales se evidencio la ausencia de modelos E-R y códigos fuente de acceso a la organización contratante.

En cuanto a la socialización de resultados es importante resaltar el compromiso por parte de los integrantes de los equipos de desarrollo, sin embargo, ha llamado la atención la necesidad de establecer controles que permitan garantizar el apoyo imparcial de la alta gerencia en los procesos de contratación de empresas de desarrollo de software, de tal manera que se logré definir planes de acción que conlleven a una mejora substancial de las aplicaciones requeridas y en consecuencia mejor soporte para los equipos de Tecnologías de Información In House, al solicitar evidencia sobre documentación específica del sistema en evaluación, se pudo constatar que existe gran ausencia de formalización básica que de soporte a los desarrollos contratados.

Se llevó a cabo posteriormente un proceso de entrevista con todo el personal involucrado en la implementación del instrumento de evaluación quienes manifestaron haberse sentido cómodos con su uso, así mismo indicaron haber descubierto cosas que no habían identificado en ejercicios de auditoría de sistemas previos, lo que los conllevó a solicitar apoyo en temáticas que requerían la injerencia de la Alta Dirección.

Resalta el hecho de que todos los profesionales que intervinieron en las diferentes pruebas del instrumento de evaluación recalcaron la importancia de desarrollar software que cumpla dicha función del tipo SAAS – Software As A Service, de tal manera que se conserve la trazabilidad de las evaluaciones realizadas en periodos pasados.

Conclusiones

- La implementación de la evaluación incrementa la conciencia del equipo de desarrollo de software en hacer las cosas bien desde un principio, documentando y revisando oportunamente su quehacer diario.
- Impulsar el uso de instrumentos de evaluación no solo del proceso de desarrollo de software sino de la verificación de la seguridad en las aplicaciones permite que la alta gerencia tome decisiones estratégicas en relación con los procesos de desarrollo de software que se vuelven trascendentales para la continuidad del negocio y el soporte de las transacciones informacionales.
- Las empresas, consultores, auditores de sistemas de información y/o auditores de sistemas de gestión ISO 27001 que empleen el instrumento de evaluación de calidad del software desde el proceso de desarrollo de software y la verificación de requisitos

de seguridad en aplicaciones deben poseer no solo conocimientos generales del proceso sino también conocimientos técnicos relacionados a pruebas de penetración y vulnerabilidad de sistemas de información.

- La evaluación de la calidad del software a partir del proceso de desarrollo y la verificación de requisitos de seguridad en aplicaciones es un proceso que debe realizarse de manera continua en todo el proceso de conformidad como avanza el proceso de desarrollo.
- La evaluación de la calidad del software desde el proceso de desarrollo y la verificación de requisitos de la seguridad en las aplicaciones permite la protección de las organizaciones no solo a través del software que emplean como parte de su operación interna, sino también, todo el software que se emplea para brindar los servicios ofertados por la organización a los clientes.
- La pandemia de la COVID – 19 trajo consigo la imperiosa necesidad de ejecutar de manera rutinaria evaluaciones a la calidad del software y la verificación de los requisitos de seguridad en aplicaciones, toda vez que incrementaron los ciberataques y la puesta en escena de servicios virtuales.

Referencias

[1] Agarwal, R., & Umphress, D. (2008). Extreme programming for a single person team. Proceedings of the 46th Annual Southeast Regional Conference, (págs. 82-87). Auburn, Alabama. doi:10.1145/1593105.1593127

[2] Celis Mendoza, O. B. (03 de 07 de 2014). Diferencias, ventajas y desventajas estre Scrum, XP, OpenUp e ISO 12207. Recuperado el 21 de 12 de 2017, de https://prezi.com/vugjhc65whet/diferenecias-ventajas-y-desventajas-entre-scrumxpopenup-y-iso-12207/

[3] Ciudades y Gobiernos Locales Unidos CGLU. (2020). Informe Tecnologías digitales y la pandemia de COVID-19. Recuperado el 18 de mayo de 2021, de eng_briefing_technology_es.pdf (uclg.org)

[4] Eclipse. (30 de 05 de 2012). epf.eclipse.org. Recuperado el 20 de 12 de 2017, de http://epf.eclipse.org/wikis/openup/index.htm

[5] Erazo, P. (2018). Definición de un proceso de Desarrollo de software en modalidad unipersonal combinando ISO/IEC 29110:2014 y metodologías ágiles. Recuperado el 18 de mayo de 2021, de Definición_proceso_desarrollo_proyectos_software_modalidad_unipersonal_combinando_ISO_IEC_29110_2014_procesos_ágiles.pdf (autonoma.edu.co)

[6] Gimson, L. (2012). Metodologías ágiles y desarrollo basado en conocimiento. Tesis, Universidad Nacional de la Plata, Facultad de informática, Argentina. Recuperado el 20 de 12 de 2017, de http://sedici.unlp.edu.ar/bitstream/handle/10915/24942/Documento_completo__.pdf?sequence=1

[7] ICONTEC. (2014). NTC ISO/IEC TR 29110-5-1-2. (ICONTEC, Ed.) Bogotá, Colombia.

[8] IEEE Computer Society. (2014). SWEBOK V.3 Guide to the software engineering body of knowledge (Vol. 3). Piscataway, New Jersey: IEEE. Obtenido de www.swebok.org.

[9] Laporte, C. Y. (2016). La implementación de la norma ISO/IEC 29110 guías de gestión e ingeniería para las organizaciones pequeñas. Congreso internacional de mejora de procesos de software, 5, págs. 69-70. Aguascalientes, México. Recuperado el 15 de 12 de 2017, de https://www.researchgate.net/profile/Claude_Laporte/publication/312874829_La_implementacion_de_la_norma_ISOIEC_29110_Guias_de_Gestion_e_Ingenieria_para_las_organizaciones_pequenas/links/5888aa00458515098e43a754/La-implementacion-de-la-norma-ISO-IEC-29110-

[10] Maida, E. G., & Pacienzia, J. (2015). Metodologías de desarrollo de software. Tesis de Licenciatura en Sistemas y Computación, Pontificia universidad católica de Argentina Santa María de los Buenos Aires, Faculta de química e ingeniería, Buenos Aires, Argentina. Recuperado el 01 de 11 de 2017, de http://bibliotecadigital.uca.edu.ar/repositorio/tesis/metodologias-desarrollo-software.pdf

[11] Microsoft Developer Network. (2013). MSDN. Recuperado el 12 de 12 de 2017, de https://msdn.microsoft.com/es-es/library/dd997578%28v=vs.120%29.aspx?f=255&MSPPError=-2147217396

[12] Organización de las Naciones Unidas para el Desarrollo Industrial (ONUDI). (2015). Informe sobre el desarrollo industrial 2016. El rol de la tecnología y la innovación en el desarrollo industrial inclusivo y sostenible. Viena: ONUDI.

[13] OWASP. (s.f). The open web application security project (OWASP). Recuperado el 18 de mayo de 2021, de Proyecto_OWASP.pdf

[14] OWASP. (2017). Estándar de verificación de seguridad en aplicaciones. Recuperado el 18 de mayo de 2021, de Estándar de Verificación de Seguridad en Aplicaciones 3.0.1 (owasp.org)

[15] Peffers, K., Tuunanen, T., Rothenberger, M., & Chatterjee, S. (s.f.). A Desing Science Research Methodology for Information System Research. Journal of Management Informtion Systems, 24(3), págs. 45-77.

[16] Schwaber, K., & Shuterland, J. (2013). La guía definitiva de Scrum: las reglas del juego. ScrumGuides. Scrum.org and ScrumInc. Recuperado el 19 de 12 de 2017, de http://www.scrumguides.org/docs/scrumguide/v1/scrum-guide-es.pdf

[17] Sommerville, I. (2011). Sofwtare engineering (Novena edición ed.). Boston, Masachussets: Pearson Educación.

[18] Soto Duran, D. E., & Reyes Gamboa, A. X. (2010). Introduciendo PSP (Proceso Personal de Software) en el Aula. Revista Colombiana de Tecnologías de Avanzada, 2(16), 1-5. Obtenido de http://www.unipamplona.edu.co/unipamplona/portalIG/home_40/recursos/03_v13_18/revista_16/27102011/01.pdf

[19] Szulanski, G. (1996). Exploring Internal Stickiness: Impediments to the Transfer of Best Practice Within the Firm. Strategic Management Journal, 17, 27-43. Recuperado el 23 de septiembre de 2018, de http://www.jstor.org/stable/2486989

[20] Watts, H. (2000). The Personal Software Process (PSP) (CMU/SEI-2000-TR-022). Carnie Mellon University: Software Engineering Institute. Recuperado el 31 de Agosto de 2018, de https://resources.sei.cmu.edu/library/asset-view.cfm?assetid=5283

[21] Wells, D. (1999). The Rules of Extreme Programming. Recuperado el 23 de septiembre de 2018, de http://www.extremeprogramming.org/rules.html

[22] Zapata, J, (2018). Udo de tecnologías de pruebas de penetración para validación de seguridad de aplicaciones web basado en el top 10 de vulnerabilidades de OWASP. Recuperado el 18 de mayo de 2021, de Microsoft Word - Proyecto de Grado - Juliana Zapata V16.docx (unad.edu.co)

Software Quality and Application Security Base on the AGILISO Software Development Process and the OWASP Standard

Progrezando.com

Carlos Arturo Blandón Jaramillo, Jhon Steven Jaramillo Becerra
Progrezando.com, Colombia
Carlos.blandon@progrezando.com, jhonstevenjaramillo25@gmail.com

INTRODUCCIÓN

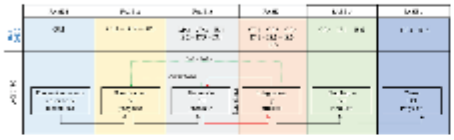
En los últimos años las tecnologías de información han impulsado un proceso de globalización de gran escala que ha impactado directamente a todos los sectores industriales; como consecuencia directa de dicho proceso, la tecnología se ve avocada a ser más eficiente, segura, confiable y "a brindar la sensación de ubicuidad, requiriendo desarrollo de software de calidad que cumpla con los requisitos del mercado, por tanto, los desarrolladores requieren más control sobre recursos, tiempo, costo y calidad del producto". La falta de adopción de estándares internacionales para el desarrollo adecuado de proyectos de software en conjunción con la formación en procesos ágiles representa una gran debilidad en esta industria.

Esta es una propuesta para evaluar la calidad del software a partir del proceso de desarrollo de software AGILISO y la seguridad en las aplicaciones en base al estándar de verificación de seguridad en aplicaciones OWASP, fortaleciendo y optimizando la actividad de auditoría por parte de control interno, auditores y consultores de sistemas de información, permitiendo la propuesta oportuna de planes de acción que procuren la corrección de las desviaciones detectadas.

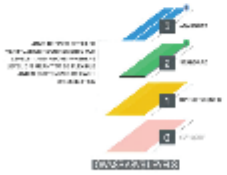
MATERIALES Y MÉTODOS

Identificar y lograr apropiar las mejores prácticas es de vital importancia para obtener una ventaja competitiva, para ello se analizan las mejores prácticas de diversas metodologías ágiles y se cruzan con las recomendaciones contenidas en el estándar internacional ISO/IEC 29110:2014.

Se propone en consecuencia el proceso de desarrollo unipersonal AGILISO.



Las condiciones del contexto informático durante la pandemia, trajeron consigo la necesidad imperativa de diversificar la prestación de servicios virtualizando los canales de atención lo que a su vez intriduce la necesidad de seguridad en las aplicaciones.



RESULTADOS Y DISCUSIÓN

Se diseña un instrumento que apoya la evaluación de la calidad del software desde el proceso de desarrollo. La herramienta fue construida en Microsoft Excel, conformada por ocho hojas de cálculo, las cuales se distribuye de la siguiente manera:

- Fase I a VI: Fases del proceso de desarrollo de software AGILISO
- Fase VII: Verificación de requisitos de seguridad en aplicaciones
- DashBoard: Resultados de la verificación de requisitos para las Fases I – VII

La fase VII de verificación de requisitos en la seguridad de aplicaciones, presenta los resultados en gráfico de barras identificando los hallazgos a la seguridad

CONCLUSIONES

- La implementación de la evaluación incrementa la conciencia del equipo de desarrollo de software en hacer las cosas bien desde un principio, documentando y revisando oportunamente su quehacer diario.
- Impulsar el uso de instrumentos de evaluación no solo del proceso de desarrollo de software sino de la verificación de la seguridad en las aplicaciones permite que la alta gerencia tome decisiones estratégicas en relación con los procesos de desarrollo de software que se vuelven trascendentales para la continuidad del negocio y el soporte de las transacciones informacionales.
- Las empresas, consultores, auditores de sistemas de información y/o auditores de sistemas de gestión ISO 27001 que empleen el instrumento de evaluación de calidad del software desde el proceso de desarrollo de software y la verificación de requisitos de seguridad en aplicaciones deben poseer no solo conocimientos generales del proceso sino también conocimientos técnicos relacionados a pruebas de penetración y vulnerabilidad de sistemas de información.
- La evaluación de la calidad del software a partir del proceso de desarrollo y la verificación de requisitos de seguridad en aplicaciones es un proceso que debe realizarse de manera continua en todo el proceso de conformidad como avanza el proceso de desarrollo.
- La evaluación de la calidad del software desde el proceso de desarrollo y la verificación de requisitos de la seguridad en las aplicaciones permite la protección de las organizaciones no solo a través del software que emplean como parte de su operación interna, sino también, todo el software que se emplea para brindar los servicios ofertados por la organización a los clientes.
- La pandemia de la COVID – 19 trajo consigo la imperiosa necesidad de ejecutar de manera rutinaria evaluaciones a la calidad del software y la verificación de los requisitos de seguridad en aplicaciones, toda vez que incrementaron los ciberataques y la puesta en escena de servicios virtuales.