



LOS LIBERTADORES
FUNDACIÓN UNIVERSITARIA

Ciclo de vida de desarrollo ágil de software seguro

Miguel Hernández Bejarano*
mhernandezb@libertadores.edu.co

Luis Eduardo Baquero Rey*
lebaqueror@libertadores.edu.co

Colección INVESTIGACIÓN

* Grupo de Investigación en Ingeniería Aplicada (GUIAS).

Catalogación en la Publicación Fundación Universitaria Los Libertadores

Hernández Bejarno, Miguel

Ciclo de vida de desarrollo ágil de software seguro Miguel Hernández Bejarano / Luis Eduardo Baquero Rey, – Bogotá: Fundación Universitaria Los Libertadores, 2020.

108 páginas, ilustraciones, gráficas; 26 cm (Colección Investigación)

ISBN: 978-958-5478-41-1 (impreso) | ISBN: 978-958-5478-44-2 (digital)

1. Seguridad en el software – 2. Metodologías ágiles – 3. Desarrollo de software seguro – 4. Seguridad informática. I Hernández Bejarano, Miguel, Autor II. Fundación Universitaria Los Libertadores.

SCDD 005.8 B222c –dc23

FULL BIBLIOTECA

Primera Edición: Bogotá, diciembre de 2020
©Fundación Universitaria Los Libertadores.

Cra. 16 No. 63A-68/ Tel. 2544750.
www.ulibertadores.edu.co

Juan Manuel Linares Venegas
Presidente del Claustro

Ángela María Merchán Basabe
Rectora

©*Miguel Hernández Bejarano*
©*Luis Eduardo Baquero Rey*
Autores

Jefferson Miguel Hernández Cáceres
Diagramación

Diego A. Martínez Cárdenas
Coordinador Editorial

Los autores declaran que esta investigación fue financiada por la Fundación Universitaria Los Libertadores en el marco de la Convocatoria de Investigaciones internas de la institución. Los conceptos emitidos en esta publicación son responsabilidad expresa de los autores y no comprometen de ninguna forma a la institución. Se autoriza la reproducción del texto citando autor y fuente, únicamente con fines académicos. En caso distinto, se requiere solicitar autorización por escrito al editor. Escrito en LaTeX.

Agradecimientos

Los autores expresan especial agradecimiento a todas aquellas personas que de una u otra manera han contribuido para que esta obra haya sido posible, empezando por los estudiantes del semillero de investigación en seguridad informática donde nació la idea del proyecto, a los colegas del programa de Ingeniería de Sistemas, a los colegas del Grupo de Investigación en Ingeniería Aplicada (GUIAS), a los directivos de la Facultad de Ingeniería y Ciencias Básicas, y en general a la Fundación Universitaria Los Libertadores por brindar los espacios y el apoyo financiero necesario.

A Dios, a mi esposa Flor Ángela e hijos (Jefferson, Julieth y Oscar).
Miguel

A Dios, mi familia, colegas y estudiantes
Eduardo

Prólogo

Nos encontramos inmersos en una sociedad donde el consumo de datos sobre diversas plataformas conectados a través de Internet requiere de un alto nivel prestacional por parte de diversos sectores y actores tecnológicos, como proveedores de servicio, fabricantes, así como el sector de la industria del software. Soluciones basadas en aplicaciones que soportan solicitudes para el ingreso o consumo de datos en tiempo real sobre cualquier área: lúdica, consumo, ocio, trabajo, educación, entre otros. Esta variedad de soluciones demanda grandes desafíos, orientados a cada vez más altos niveles de prestaciones que puedan ofrecer, tales como la disponibilidad, robustez, interoperabilidad, integridad, etc., de cada uno de sus componentes, incluyendo aspectos trascendentales propios para lograr mitigar riesgos y vulnerabilidades de seguridad.

La realidad de la cantidad de procesos que se desprenden de soluciones informáticas es tan variada, que, para llevar a cabo su correcto desarrollo bajo especificaciones y estándares, requieren de un alto nivel de madurez en todos y cada uno de los procesos establecidos en las etapas de desarrollo de software. Es así como la presente obra, presenta una de esas realidades inmersas sobre un variado ecosistema de soluciones digitales basadas en software; enfocado puntualmente sobre estrategias y métodos para mitigar riesgos de seguridad dentro de procesos enmarcados en el ciclo de vida del desarrollo de software. Lo anterior, bajo diversas miradas que presenta el mercado informático a partir de metodologías ágiles. Indudablemente una apuesta que permite acercarnos a una serie de consideraciones y criterios que todo equipo, empresa o casa matriz de desarrollo necesitan conocer e identificar, con el propósito de poder cubrir aspectos esenciales en el despliegue de aplicaciones y soluciones informáticas que logren ser distribuidas sobre el mercado.

Dentro de la presente obra, el lector podrá encontrar apartados orientados a destacar el panorama que presentan las metodologías de desarrollo ágil dentro del contexto de software seguro. Así mismo, la relevancia que presenta el desarrollo ágil, de la mano de estrategias y buenas prácticas para el desarrollo seguro del software. Finalmente, los autores exponen de manera detallada y validando mediante caso de estudio aplicado, una propuesta metodológica partiendo de los principios de ciclo de vida de desarrollo de software, y el uso de estrategias de seguridad a considerar en cada una de las fases planteadas.

Me place encontrar dentro de la literatura nacional, este tipo de apuestas orientadas a ofrecer espacios de análisis, reflexión y propuestas concretas asociadas a métodos de desarrollo de software ágil a partir de estrategias de seguridad, sobre todo, en el marco de una variada industria de software apalancada por emprendimientos de base tecnológica, donde el software como actor del ecosistema digital, es uno de los insumos valiosos para una sociedad que demanda cada vez más servicios en línea.

Paulo Alonso Gaona García, Ph.D

Introducción

El mercado es cada día más exigente en cuanto a desarrollo de software se refiere. Ya no es suficiente con que las casas desarrolladoras de programas informáticos cumplan a cabalidad con los requerimientos de funcionalidad que satisfacen las necesidades del cliente, sino que además se deben involucrar aspectos relevantes de los requisitos no funcionales que de alguna manera mitiguen los posibles riesgos a los que pueda estar expuesto el producto cuando este se encuentre en la etapa productiva, pues el número de vulnerabilidades es cada vez mayor, lo que pone en riesgo la información y, por ende, la continuidad del negocio de las organizaciones.

Las herramientas ágiles buscan dar soluciones informáticas oportunas con la participación activa y directa del cliente, y pueden ofrecer mayor seguridad a las aplicaciones desarrolladas al involucrarlas con un ciclo de vida de desarrollo de software seguro. Esta obra da cuenta de los resultados obtenidos durante una investigación aplicada a la integración de un ciclo de vida de desarrollo de software seguro (S-SDLC), de fundamentos teóricos y herramientas existentes para el desarrollo ágil de este tipo de proyectos y de buenas prácticas de seguridad de la información en los desarrollos de productos software. Para ello, se propuso una metodología implementada bajo las circunstancias del planteamiento de un caso de estudio hipotético, que podría ser real.

Es conveniente reseñar la concepción del proyecto desde el semillero de investigación en seguridad informática, que posteriormente se planteó en el marco de la convocatoria interna de proyectos de investigación institucional en la línea de investigación de Innovación y Emprendimiento, como parte integrante del Grupo de Investigación en Ingeniería Aplicada (GUIAS).

El objetivo fundamental en la seguridad del software es la construcción de aplicaciones informáticas de mejor calidad, más robustas y libres de fallos, que implementen el principio de resiliencia, es decir, que sigan funcionando correctamente ante ataque malicioso (McGraw, 2006).

En respuesta a la creciente tasa de daños y perjuicios causados a las empresas por explotación de las vulnerabilidades de seguridad en los productos de software, se requiere de mayor esfuerzo para desarrollar software más seguro (Keramati y Mirian-Hosseiniabadi, 2008).

Para tal efecto, este documento se presenta en seis capítulos, de la siguiente manera:

En el capítulo 1 se realiza un estudio comparativo de diferentes ciclos de vida de desarrollo de software seguro y se determinan sus ventajas y desventajas, para luego establecer la conveniencia de su aplicabilidad con metodologías ágiles de desarrollo de software, partiendo de la flexibilidad y las características que ofrecen, lo que producirá un análisis de resultados y su posterior discusión. Ello servirá de insumo para la industria en este campo.

En el capítulo 2 se estudian los principales elementos y técnicas ágiles implicadas en el desarrollo de un producto software, a fin de determinar los principales elementos aportantes para el desarrollo de la investigación.

El capítulo 3 trata sobre las buenas prácticas y los estándares de seguridad de la información, como la norma ISO 27001 y los Common Criteria (CC); estudia organizaciones como Open Web Application Security Project (OWASP), que aportan esfuerzos importantes para el desarrollo de aplicaciones web seguras; compara herramientas para el desarrollo de software seguro y para el análisis de código, como UMLsec, e identifica las propiedades que debe cumplir un software seguro y las amenazas a dicha seguridad.

En el capítulo 4 se presentan las fases del ciclo de vida de desarrollo de software propuesto, la metodología adoptada, el caso de estudio planteado y su desarrollo en los términos ya enunciados, adoptando el marco de trabajo SCRUM.

Finalmente, en los capítulos 5 y 6 se hace un análisis de los resultados obtenidos y se relacionan las principales conclusiones producto del desarrollo del proyecto, respectivamente.

Con esta investigación se busca adoptar una buena práctica que dé respuesta a requisitos de seguridad que deben contemplar los equipos de desarrollo ágil de software, con el fin de construir productos más estables y robustos, fundamentales en materia de seguridad informática, en pro de garantizar la confidencialidad, disponibilidad e integridad de un producto software. Entretanto, también pretende servir de apoyo a la academia en la formación de nuevos profesionales, como un insumo a tener en cuenta en las líneas de Ingeniería de Software y Seguridad de la Información.

Índice general

1. Ciclos de vida de desarrollo de software seguro	6
1.1. Introducción	6
1.2. Tipos de ciclos de vida de desarrollo de <i>software</i> seguro	6
1.2.1. McGraw's Seven Touchpoints	7
1.2.2. Microsoft Security Development Lifecycle (SDL)	8
1.2.3. Correctness by Construction (CbyC)	9
1.2.4. Comprehensive, Lightweight Application Security Process (CLASP)	10
1.2.5. Software Assurance Maturity Model	11
1.2.6. Team Software Process (TSP)	12
1.2.7. Otras metodologías	16
1.3. Características de los <i>S-SDLC</i>	16
1.4. Modelado de amenazas	19
2. El desarrollo ágil en el software	21
2.1. Introducción	21
2.1.1. El manifiesto ágil	21
2.1.2. Las técnicas ágiles y la madurez de la industria del software	22
2.1.3. Scrum	23
2.1.4. Metodología XP	24
2.1.5. Kanban	24
2.1.6. OpenUP/Basic	27
2.1.7. SD3+C	28
3. Buenas prácticas en seguridad de la información	29
3.1. Introducción	29
3.2. Estándares	29
3.2.1. Norma ISO 27001	29
3.2.2. Common Criteria (CC)	30
3.3. Organizaciones	30
3.3.1. Common Weakness Enumeration (CWE)	30
3.3.2. Web Application Security Consortium (WASC)	30
3.3.3. The Open Web Application Security Project (OWASP)	30
3.3.4. Secure Software Programmer	33
3.4. Desarrollo de software seguro	34
3.4.1. Security Requirements Engineering Process (SREP)	34
3.5. Buenas prácticas de seguridad del <i>software</i>	34
3.6. Herramientas de desarrollo de <i>software</i> seguro	34
3.6.1. Casos de mal uso (casos de abusos)	34
3.6.2. UMLsec	34
3.6.3. Herramientas para análisis de código	35
3.7. Seguridad en el <i>software</i>	35

3.7.1.	Propiedades de un <i>software</i> seguro	35
3.7.2.	Amenazas a la seguridad del <i>software</i>	36
4.	Metodología y caso de estudio	37
4.1.	Introducción	37
4.2.	S-SDLC propuesto	37
4.2.1.	Fase de Capacitación	37
4.2.2.	Fase de Construcción de los requerimientos y Casos de abuso	38
4.2.3.	Fase de Análisis de riesgos	38
4.2.4.	Fase de Diseño	38
4.3.	Fase de Codificación y pruebas	38
4.3.1.	Fase de Implementación	38
4.3.2.	Fase de Seguimiento	38
4.4.	Fase de Auditoría de seguridad	38
4.5.	Metodología adoptada	39
4.6.	Caso de estudio	42
4.7.	Desarrollo del caso de estudio	42
4.7.1.	Fase de Capacitación	42
4.7.2.	Fase de Construcción de los requerimientos y Casos de abuso	43
4.7.3.	Ejecución del <i>sprint</i>	50
4.7.4.	Fase de Análisis de riesgos	55
4.7.5.	Fase de Diseño	57
4.7.6.	La seguridad en las bases de datos	59
4.7.7.	Seguridad en motores de bases de datos	61
4.7.8.	Fase de Codificación y pruebas	69
4.7.9.	Algunas pruebas	91
4.8.	La seguridad <i>framework</i>	91
4.8.1.	<i>Framework</i> Java	91
4.8.2.	<i>Framework</i> PHP	97
4.9.	Fase de implementación	99
4.9.1.	Salvaguardas de los sistemas operativos	99
4.9.2.	Arquitectura de la aplicación	100
4.10.	Fase de Seguimiento	103
4.11.	Fase de Auditoría de seguridad	104
5.	Análisis de resultados	107
6.	Conclusiones	108
	Bibliografía	109

Índice de figuras

1.1. Touchpoints de Gary McGraw	8
1.2. Ciclo de vida de Microsoft SDL	8
1.3. Ciclo de vida de Microsoft SDL	9
1.4. Ciclo de vida de Microsoft SDL	10
1.5. Relación de los niveles de vistas CLASP	11
1.6. Organización SAMM	15
1.7. <i>STRIDE</i>	20
2.1. <i>Valores del manifiesto ágil</i>	21
2.2. <i>Principios del manifiesto ágil</i>	22
2.3. <i>Enfoque SD3+C</i>	28
3.1. <i>Top 10 de OWASP</i>	31
3.2. <i>Vulnerabilidades asociadas al Top 10</i>	32
3.3. <i>OWASP Cloud-10</i>	33
4.1. <i>Fases del S-SDLC propuesto</i>	37
4.2. <i>Scrum</i>	39
4.3. <i>Ciclo Scrum</i>	39
4.4. <i>Scrum adoptado para el proyecto</i>	40
4.5. <i>Product backlog</i>	44
4.6. <i>Historia de usuario Ingreso al sistema</i>	44
4.7. <i>Historia de usuario Registro</i>	44
4.8. <i>Historia de usuario Renovación de contrato</i>	45
4.9. <i>Historia de usuario Realizar pago</i>	45
4.10. <i>Historia de usuario Consultar temas</i>	45
4.11. <i>Historia de usuario Buscar revistas</i>	45
4.12. <i>Caso de abuso Ingreso al sistema</i>	45
4.13. <i>Historia de usuario HU01 - Iteración 3</i>	47
4.14. <i>Ajuste historia de usuario HU02</i>	48
4.15. <i>Planeación de historia de usuario</i>	49
4.16. <i>Historia de usuario HU03 - Iteración 3</i>	50
4.17. <i>Sprint propuestos</i>	51
4.18. <i>Tablero Kanban para historias de usuario</i>	52
4.19. <i>Tablero Kanban incluyendo seguridad</i>	52
4.20. <i>Diagrama de clases del usuario</i>	58
4.21. <i>Diagrama de clases de la revista</i>	59
4.22. <i>Seguridad en Oracle</i>	61
4.23. <i>Tabla de Usuario</i>	62
4.24. <i>Modelo E-R para Suscriptor</i>	64
4.25. <i>Modelo E-R para la Revista</i>	65
4.26. <i>Capas de la aplicación</i>	69

4.27. Estadísticas de los lenguajes de programación	70
4.28. To de los leguajes	71
4.29. Seguridad en .Net	72
4.30. Seguridad en PHP	73
4.31. Seguridad en Java	74
4.32. Pruebas de ingreso al sistema	76
4.33. Captcha	82
4.34. Apache Shiro	92
4.35. Java de cifrado simplificado	94
4.36. Spring Security	95
4.37. jGuard	96
4.38. Codelnigter	97
4.39. Laravel	98
4.40. Autenticación y autorización en Symfony	98
4.41. Aspectos de seguridad en profundidad	99
4.42. Elementos de una arquitectura segura	100
4.43. Internet Information Server (IIS)	101
4.44. Servidor Apache	101
4.45. Servidor NGINX	102
4.46. Servidores web más utilizados	102