



UNIVERSIDAD NACIONAL
Facultad de Ciencias Exactas y Naturales
Escuela de Informática



Implementación de un marco de trabajo para el desarrollo ágil de software, basado en prácticas de integración y entrega continua

Para optar al grado de de Licenciado en Informatica con énfasis en Sistemas Web bajo la modalidad de Proyecto de Graduación.

Estudiantes

Kristopher Barrot Díaz
Cédula No.: 207490502
kristopherbd@hotmail.com

Marco Rojas Maroto
Cédula.: 116380444
romarco1496@gmail.com

Heredia, Costa Rica
Octubre, 2018

1. TEMA.

En Costa Rica, al igual que en el resto del mundo, las organizaciones públicas y privadas poseen departamentos de información, los cuales deben seguir una serie de estándares y procesos a la hora de producir software, ya sea por la necesidad de llevar un buen control y manejo de los diferentes proyectos o por el hecho de que el estado defina una serie de normas que estos departamentos deban seguir, como es el caso en nuestro país con el documento “Normas técnicas para la gestión y control de las tecnologías de información”.

Debido a todo esto han surgido diferentes metodologías y herramientas de software que facilitan el proceso de desarrollo de software en todos sus ámbitos de manera que este proceso se torne mucho más organizado y estandarizado.

Dicho esto, en el año 2007 se establece la Sede Interuniversitaria de Alajuela, la cual surge en el contexto de la creación del “Régimen de Sedes Interuniversitarias de la Educación Superior Universitaria Estatal de Costa Rica”, el cual fue aprobado por CONARE el 6 de marzo de ese mismo año. Al mismo tiempo nace la Unidad de Gestión e Innovación Tecnológica (UGIT) con el objetivo de brindar soporte a todo lo que envuelve a las tecnologías de información y comunicación en la Sede y de esta manera lograr cubrir las necesidades tecnológicas de la Universidad Nacional (UNA), Tecnológico de Costa Rica (TEC) y la Universidad Estatal a Distancia (UNED).

A pesar de un incremento en la necesidad de implementar proyectos de software dentro de la institución, nunca se ha visto a la UGIT como un departamento consolidado, sino, se ha visto como una unidad para brindar soporte técnico y uno que otro proyecto de desarrollo, por lo cual nunca se ha visto la necesidad de contratar más personal para el departamento. Esto se debe a la falta de comunicación y coordinación entre las diferentes universidades que conforman la SIUA, Cuando la realidad es que esta unidad, es la encargada de solventar todas las necesidades tecnológicas de la Sede.

Debido al hecho de que el desarrollo de software de la UGIT se concentra en un único empleado, nunca ha existido la iniciativa dentro del departamento por implementar algún tipo de metodología o infraestructura de desarrollo de software que se adapte a las necesidades y a los estándares de las organizaciones nacionales de acuerdo a lo estipulado por la Contraloría General de la República.

2. JUSTIFICACIÓN.

De acuerdo al documento “Normas técnicas para la gestión y control de las tecnologías de información”, publicado en la gaceta el día 21 de junio de 2007 por la Contraloría Reneral de la República, todas las instituciones y órganos sujetos a su fiscalización deben cumplir con una serie de normas que regulan el uso y la implementacion de Tecnologias de informacion dentro de sus respectivos departamentos de TI.

Capítulo I

- 1.2. Gestión de la Calidad
- 1.4. Gestión de la Seguridad de la Información.
- 1.5. Gestión de Proyectos

Capítulo II

- 2.2. Modelo de Arquitectura de Información
- 2.3. Infraestructura Tecnológica.

Capítulo III

- 3.2. Implementación De Software
- 3.3. Implementación de Infraestructura Tecnológica.

Capítulo IV

- 4.2. Administracion y operacion de la infraestructura Tecnológica.

Capítulo V

- 5.2. Seguimiento y evaluación del control interno en TI.

Considerando este documento, la UGIT tiene una problemática particular en cuanto al desarrollo ágil de software, misma que se describe en la siguiente tabla.

Descripción del Problema	<ul style="list-style-type: none">• No existe ninguna metodología de desarrollo implementada dentro de la UGIT• No existen herramientas de que garanticen la calidad del software que se genera dentro de la UGIT• No existen herramientas de software para la coordinación de proyectos
--------------------------	--

	<ul style="list-style-type: none"> • No existen herramientas de software para la automatización de tareas • No existen los ambientes de desarrollo, pruebas y producción.
Afecta a	Desarrolladores / Usuarios
El impacto asociado es	<ul style="list-style-type: none"> • Desorganización, mala gestión del proyecto, fracaso del proyecto, pérdida del código fuente. • Software final en producción con muchos errores lo cual genera insatisfacción con el usuario y este tenga que estar fuera de servicio por constantes mantenimientos. • Mala distribución de tareas. • No hay fechas establecidas para la entrega de objetivos. • No hay objetivos claros. • Pérdida de tiempo ya que se repiten y generan pruebas. • Migración del proyecto entre ambientes es manual. • Que los desarrolladores tengan la obligación de realizar cambios sobre la versión de producción lo cual genera el descontento de usuarios • Que las herramientas de software estén fuera de servicio por mantenimiento

3. MARCO TEÓRICO

Desarrollo Ágil

El desarrollo de software ágil es un concepto usado en el desarrollo de software para describir las metodologías de desarrollo incrementales (Cohen, Lindvall & Costa, 2003).

El desarrollo ágil de software envuelve un enfoque para la toma de decisiones en los proyectos de software, que se refiere a métodos de ingeniería del software basados en el desarrollo iterativo e incremental, donde los requisitos y soluciones evolucionan con el tiempo

según la necesidad del proyecto. Así el trabajo es realizado mediante la colaboración de equipos auto-organizados y multidisciplinarios, inmersos en un proceso compartido de toma de decisiones a corto plazo. (Morfin, 2017)

Metodologías Ágiles

Debido al gran avance tecnológico de las últimas décadas existe una necesidad de software que complementa estos avances. Producto de esta necesidad el proceso de desarrollo de software ha tenido que evolucionar de manera tal que se ha optado por utilizar metodologías modernas para la gestión del desarrollo de proyectos de TI, dejando de lado algunos de los principios del desarrollo en cascada. (Morfin, 2017)

El desarrollo en cascada podemos definirlo como un modelo secuencial de etapas que el equipo de trabajo deberá ir completando, una a la vez, hasta llegar a su etapa final. Producto de cada fase se genera un entregable, el cual es necesario para iniciar la fase siguiente, lo que imposibilita que se puedan trabajar aspectos del proyecto de manera simultánea. Por lo tanto este proceso es lento y no es muy tolerante a cambios, ya que si se produce un cambio es necesario empezar de nuevo todo el ciclo con el fin de incorporar este al proyecto. (Riquelme, s.f).

El modelo de Gestión de Proyectos Ágil es una metodología de trabajo moderna la cual se creó bajo el concepto de flexibilidad y adaptabilidad a los cambios, el cual busca una manera distinta de desarrollar proyectos en los cuales el foco de atención es el resultado final. (Riquelme, s.f).

A diferencia del modelo tradicional en cascada, la metodología ágil trabaja con ciclos iterativos. Lo cual promueve el trabajo simultáneo tanto en el desarrollo como en las pruebas y la interacción con el cliente. (Alvis, 2016)

Si bien, las metodologías ágiles buscan solucionar los diversos problemas de la metodología tradicional, esta no deja de lado los procesos de documentación y normativas. Se busca garantizar un producto final de calidad el cual satisfaga las necesidades del clientes, por lo tanto estos son enfocados con otra prioridad en la cual son necesarios pero no indispensables. (Morfin, 2017)

Scrum

Según lo definen los autores en el artículo Scrum Distribuido “Scrum es una Metodología Ágil de Gestión de Proyectos que se basa en la adaptación continua a las circunstancias evolutivas del Proyecto apoyándose en iteraciones cortas conocidas como Sprints a través del siguiente ciclo. Es una metodología simple en la que interactúan diferentes roles (Equipo de Desarrollo, Scrum Manager, Propietario del Producto y Stakeholders)”. (Herranz et al, 2011)

DevOps

Según IBM DevOps es un enfoque que promueve la colaboración entre líneas de negocio, desarrollo y operaciones de TI. Es una funcionalidad empresarial que habilita la entrega continua, el despliegue continuo y la supervisión continua de aplicaciones. Reduce el tiempo necesario para tratar el feedback de los clientes. El desarrollo y las operaciones, e incluso las pruebas, antes se organizaban en silos. DevOps las reúne para mejorar la agilidad. (IBM, s.f)

“DevOps es un acrónimo inglés de development (desarrollo) y operations (operaciones), que se refiere a una metodología de desarrollo de software que se centra en la comunicación, colaboración e integración entre desarrolladores de software y los profesionales de sistemas en las tecnologías de la información (IT)”. (Ruiz, 2015)

Es una respuesta a la interdependencia del desarrollo de software y las operaciones IT. Su objetivo es ayudar a una organización a producir productos y servicios software más rápidamente, de mejor calidad y a un coste menor.” (Ruiz, 2015)

DevOps es una metodología para creación de software que se basa en la integración entre desarrolladores software y administradores de sistemas, permite fabricar software más rápidamente, con mayor calidad, menor coste y una altísima frecuencia de releases. (Ruiz, 2015)

Integración Continua

La integración continua es una práctica desarrollo en donde los miembros de un equipo integran su trabajo con frecuencia, usualmente cada integrante del equipo integra su trabajo diariamente, lo que conduce a que cada día se hagan varias integraciones. Cada integración se

verifica mediante una compilación automatizada (que incluye pruebas automáticas) para detectar errores de integración lo más rápido posible.(Fowler, 2006)..

Entrega Continua

De Acuerdo a Martin Fowler (2013) “la entrega continua es una disciplina en la cual el software es compilado de manera que este pueda ser puesto en producción en cualquier momento”.(Fowler, 2006)..

4. OBJETIVOS

Objetivo General

Implementar un marco de trabajo para el desarrollo ágil de software, basado en las prácticas de integración y entrega continua en la Unidad de Gestión e Innovación Tecnológica de la Sede INteruniversitaria de Alajuela.

Objetivos Específicos

- Investigar el estado del arte de las metodologías de desarrollo de software ágil y la práctica de Entrega e Integración Continua.
- Analizar el proceso de desarrollo actual desde la perspectiva de funcionalidad, tiempos de desarrollo y las diferentes herramientas de software que existen en el mercado, con el fin de seleccionar el conjunto que mejor se adapte y permita aplicar las prácticas de Entrega e Integración Continua.
- Diseñar un marco de trabajo basado en las herramientas de software de integración y entrega continua previamente seleccionadas, con el fin de mejorar el proceso de producción de software dentro de la Unidad de Gestion e Innovacion Tecnologica en la Sede Interuniversitaria de Alajuela, Costa Rica.

- Configurar los ambientes propios de desarrollo, pruebas y producción con el fin de Implementar el marco de trabajo previamente diseñado dentro de la estructura conceptual y tecnológica de desarrollo ágil de software de la UGIT.
- Evaluar el framework desarrollado mediante la perspectiva de funcionalidad y tiempos de desarrollo de proyectos por medio de la opinión de expertos y funcionarios.

5. METODOLOGÍA A SEGUIR.

- Observación: Esta técnica consiste en tomar información ya sea mediante la observación directa de algún proceso fenómeno, etc. que se estudia y la misma es registrada en un informe objetivo, este es un medio muy sencillo y práctico de obtener información que puede ser utilizada posteriormente para ser comparada mediante otros datos o estudios. (Cerde, H, 1991).

Por medio de esta técnica se obtendrá información sobre el estado actual de la UGIT y cuales son sus necesidades en aspectos de desarrollo ágil, para determinar cuál conjunto de herramientas es el más apropiado para la unidad. Gracias a esta técnica podremos obtener las características más importantes a la hora de escoger las herramientas que compondrán el marco de trabajo ya que en el mercado existe gran variedad de estas por lo cual es de suma importancia determinar las que se mejor adaptan a la situación actual de la UGIT.

- Entrevistas: La entrevista es una técnica de investigación que consiste obtener información mediante conversación directa con expertos realizando preguntas y profundizando en un tema específico. Este tipo de método de investigación puede estar acompañado de otras técnicas como cuestionarios los cuales ayudan al entrevistador en la obtención de información del entrevistado experto. (Cerde, H, 1991).

Se realizará una entrevista con varios expertos sobre diferentes herramientas que se utilizan en el ámbito laboral hoy en día. Esto con el fin de recopilar mayor información acerca del uso de las herramientas, sus características, necesidades, etc.. Esta técnica se realizará mediante

un documento con preguntas sobre dudas que puedan surgir de las diferentes herramientas estudiadas y que beneficios o desventajas puede traer en el departamento.

- Grupos focales: El Grupo Focal es una de las técnicas para recopilar información de los métodos cualitativos de investigación. Un grupo focal puede definirse como una discusión cuidadosamente diseñada para obtener las percepciones sobre una particular área de interés (Krueger, 1991).

Mediante esta técnica se busca recopilar información en grupos de trabajo en el departamento que permita obtener información, esto aparte de los anteriores mencionados permitirá más que todo que varios involucrados del trabajo planteen dudas y opiniones acerca de las propuestas que se van desarrollando.

6. PLAN DE TRABAJO

Actividad	Duración (Semanas)	Fecha Inicio
Investigación sobre las metodologías de desarrollo ágil	2	23/07/2018
Análisis del proceso de desarrollo actual de software	2	
Investigación acerca de los conceptos y prácticas de Integración y Entrega Continua	4	
Búsqueda de Herramientas que permitan la Integración y Entrega Continua	8	
Diseño de la nueva infraestructura conceptual de la unidad	8	
Configuración de herramientas que permiten la Integración y Entrega Continua	16	
Evaluación del marco de trabajo	8	

7. DISEMINACIÓN DE LOS RESULTADOS

Para la diseminación de resultados se utilizarán recursos proveídos por la Escuela de Informática de la Universidad Nacional

8. BIBLIOGRAFÍA

Por su parte, en esta sección se incluye la lista de las referencias bibliográficas en formato APA, organizadas alfabéticamente.

Fowler, M. (2006, mayo 6). Continuous Integration. Recuperado 20 de septiembre de 2018, de <https://martinfowler.com/articles/continuousIntegration.html>

Fowler, M. (2013, mayo 30). ContinuousDelivery. Recuperado 20 de septiembre de 2018, de <https://martinfowler.com/bliki/ContinuousDelivery.html>

IBM – ¿Qué es DevOps? - España. (2018, junio 20). Recuperado 28 de octubre de 2018, de <https://www.ibm.com/cloud-computing/es-es/learn-more/what-is-devops/>

Manifiesto por el Desarrollo Ágil de Software. (s. f.). Recuperado 28 de octubre de 2018, de <http://recursos.evaluandosoftware.com/pdf/manifiesto-desarrollo-agil-software/>

Metodologías en Cascada vs Agile. (2017, mayo 18). Recuperado 28 de octubre de 2018, de <https://comunidad.iebschool.com/ladyintech/2017/05/18/metodologias-en-cascada-vs-agile/>

Mota, E. (2017, enero 18). ¿Qué es el desarrollo ágil y cómo está cambiando la industria del software? Recuperado 28 de octubre de 2018, de <https://www.azulweb.net/desarrollo-agil-esta-cambiando-la-industria-del-software/>

Principios del desarrollo ágil. Cómo aplicar metodologías ágiles. (2016, octubre 27). Recuperado 28 de octubre de 2018, de <https://www.renacen.com/blog/principios-del-desarrollo-agil-metodologias-agiles/>

Principios del Manifiesto Ágil. (s. f.). Recuperado 31 de agosto de 2018, de <http://agilemanifesto.org/iso/es/principles.html>

¿Qué es desarrollo de software ágil? (2018, febrero 6). Recuperado 28 de octubre de 2018, de <http://www.evaluandosoftware.com/desarrollo-software-agil/>

Qué es DevOps (y sobre todo qué no es DevOps). (2015, noviembre 26). Recuperado 28 de octubre de 2018, de <https://www.paradigmadigital.com/techbiz/que-es-devops-y-sobre-todo-que-no-es-devops/>

Riquelme, M. (2015, mayo 8). ¿Gestion de Proyectos Agil o Tradicional? Recuperado 28 de octubre de 2018, de <https://www.webyempresas.com/gestion-de-proyectos-agil-o-tradicional/>

<https://www.renacen.com/blog/principios-del-desarrollo-agil-metodologias-agiles/>

<https://comunidad.iebschool.com/ladyintech/2017/05/18/metodologias-en-cascada-vs-agile/>

<https://www.webyempresas.com/gestion-de-proyectos-agil-o-tradicional/>

<http://agilemanifesto.org/iso/es/principles.html>

<http://recursos.evaluandosoftware.com/pdf/manifiesto-desarrollo-agil-software/>

<https://www.azulweb.net/desarrollo-agil-esta-cambiando-la-industria-del-software/>

<http://www.evaluandosoftware.com/desarrollo-software-agil/>

<https://www.ibm.com/cloud-computing/es-es/learn-more/what-is-devops/>

<https://www.paradigmadigital.com/techbiz/que-es-devops-y-sobre-todo-que-no-es-devops/>