

INSTITUTO TECNOLÓGICO DE COSTA RICA

ESCUELA DE COMPUTACIÓN

PROGRAMA DE MAESTRÍA EN COMPUTACIÓN



**Crear una propuesta de aplicación del marco de trabajo “Scrum”
en Dino I.T. Technologies**

**Proyecto sometido a consideración de la Escuela de Computación
para optar al grado de Maestría Profesional con énfasis en Sistemas de
Información**

Auxiliadora Martínez Martínez

Profesor tutor: Ronald Monge

**Cartago, Costa Rica
2016**

Dedicatoria

Para mi hijo Aarón y mi esposo Mauricio, quienes son lo mejor que la vida me ha regalado. A mi madre Lourdes Martínez por su apoyo incondicional en este largo viaje. A mi padre Minor Martínez, quien me demostró con su ejemplo de lucha a no rendirme jamás.

Agradecimientos

Quiero agradecer a mi familia por su apoyo y consejos.

Agradezco a todos los profesores del TEC, gracias a ellos pude obtener los conocimientos que me permitieron culminar este trabajo.

Gracias al país en el que vivo, en el cual tenemos privilegio de estudiar.

Epígrafe

“No se ve bien sino con el corazón, lo esencial es invisible a los ojos”.

El principito

Resumen

“Scrum” es un marco de trabajo que ha tomado relevancia en los últimos años, debido a que plantea una manera diferente de organizar el trabajo en un proyecto. Se basa en una serie de valores que establece el “Agile Manifiesto”, donde se le da más importancia a la comunicación directa y a la colaboración, que a los procesos, herramientas y documentación.

El presente documento, expone un proyecto que se llevó a cabo en la empresa “Dino I.T. Technologies”, que tenía como objetivo el planteamiento de un modelo de desarrollo de software, basado en “Scrum”, con el fin de solventar una serie de necesidades que estaba afrontando el equipo de Integración de dicha compañía.

El modelo de desarrollo propuesto estuvo basado en las circunstancias de situación actual que estaba viviendo el equipo de Integración. Los elementos que lo conformaron fueron:

- ✓ Un proceso de desarrollo de software
- ✓ Una herramienta para soportar dicho proceso de desarrollo
- ✓ Una serie de recomendaciones para implantar el proceso de desarrollo planteado

La investigación realizada en este proyecto utilizó un método cualitativo, con un enfoque “Investigación-acción”, pues se pretendía que a partir de la comprensión de la situación actual, se plantearan mejoras que permitieran afinar el proceso de desarrollo de software que empleaba el equipo de Integración, así como de la herramienta utilizada para soportar dicho proceso.

Palabras claves: “Scrum”, “Agile Manifesto”, Software, “Desarrollo de software”, “Investigación”, “Herramientas de software”, “Modelo de Desarrollo”, “Investigación”, “Redmine”, Tablero Físico.

APROBACIÓN DE PROYECTO

“Crear una propuesta de aplicación del marco de trabajo “Scrum” en Dino I.T. Technologies”

TRIBUNAL EXAMINADOR



M.B.A Ronald Monge Monge.
Profesor Asesor



Dr. José Helo Guzmán
Profesor Lector



Máster. Karen Aney Padilla Valverde
Profesional Externo



Dr. Roberto Cortés Morales
Coordinador del Programa de
Maestría en Computación

Índice General

1. Introducción	16
1.1. Descripción general	16
1.2. Antecedentes	17
1.2.1. Descripción de la empresa	17
1.2.2. Descripción del problema	20
1.3. Definición del problema	22
1.4. Justificación	23
1.4.1. Impacto	23
1.5. Objetivos.....	24
1.5.1. Objetivo General.....	24
1.5.2. Objetivos específicos	24
1.6. Alcance	24
1.7. Entregables.....	25
2. Marco Teórico.....	26
2.1. ¿Que es “Scrum”?	26
2.1.1. Beneficios de “Scrum”	28
2.1.2. ¿Cuándo utilizar “Scrum”?	28
2.2. Artefactos de “Scrum”	30
2.2.1. “Sprint”	31
2.2.2. “User Stories”	32
2.2.3. “Product Backlog”	34
2.2.4. “Sprint Backlog”	36
2.3. Roles de “Scrum”	37

2.3.1. “ScrumMaster”	37
2.3.2. “Product Owner”	38
2.3.3. El Equipo de Trabajo	40
2.3.4. Estructuras de equipos de “Scrum”	40
2.4. Actividades de “Scrum”	43
2.4.1. “Daily Scrum”	43
2.4.2. “Sprint Planning”	44
2.4.3. “Sprint Review”	45
2.4.4. “Sprint Retrospective”	46
3. Desarrollo metodológico	48
3.1. Análisis de la situación actual.....	50
3.1.1. Caracterizar el proceso de desarrollo actual	50
3.1.2. Determinar los estados de un ticket	51
3.1.3. Revisar documentos.....	51
3.2. Formulación del proceso de desarrollo de software.....	51
3.2.1. Definir los artefactos del proceso de desarrollo	52
3.2.2. Definir los estados de un “User Story”	52
3.2.3. Definir los roles del proceso de desarrollo	53
3.2.4. Definir los eventos del proceso de desarrollo.....	53
3.3. Evaluación de herramientas	53
3.3.1. Evaluar la herramienta “Redmine”	53
3.3.2. Definir las características deseables en la herramienta	54
3.3.3. Seleccionar y evaluar herramientas	54
3.4. Simulación del modelo de desarrollo de software	55
3.4.1. Crear el modelo de simulación	55

	10
3.4.2. Realizar la simulación	55
3.5. Recomendaciones para implementar el modelo de desarrollo de software	56
3.6. Técnicas de recolección de información	56
3.6.1. Cuestionario	56
3.6.2. Entrevista	56
3.6.3. Revisión de documentos	57
3.7. Instrumentos utilizados en la investigación	57
3.7.1. Cuestionario para caracterizar el proceso de desarrollo actual.....	57
3.7.2. Entrevista para determinar los estados de un tiquete.....	62
3.7.3. Tabla de documentos recopilados.....	62
3.7.4. Tabla de artefactos del proceso de desarrollo.....	63
3.7.5. Tabla de estados de un “User Story”	63
3.7.6. Tabla de roles del proceso de desarrollo	64
3.7.7. Tabla de eventos del proceso de desarrollo	64
3.7.8. Entrevista para evaluar la herramienta “Redmine”	65
3.7.9. Entrevista para determinar las características deseables en la herramienta....	66
3.7.10. Tabla de herramientas seleccionadas.....	67
3.7.11. Modelo de simulación	67
3.7.12. Bitácora de la simulación del modelo de desarrollo.....	68
4. Análisis de resultados	69
4.1. Análisis de la situación actual.....	69
4.1.1. Características del proceso de desarrollo actual	69
4.1.2. Estados de un tiquete	78
4.1.3. Revisión de documentos	80
4.1.4. Situación Actual del proceso de desarrollo del equipo de Integración.....	82

4.2. Formulación el proceso de desarrollo de software	92
4.2.1. Artefactos del proceso de desarrollo	92
4.2.2. Estados de un “User Story”	96
4.2.3. Roles del proceso de desarrollo	100
4.2.4. Eventos del proceso de desarrollo	101
4.3. Evaluación de herramientas	105
4.3.1. Evaluación la herramienta “Redmine”	106
4.3.2. Características deseables en una herramienta.....	107
4.3.3. Selección y evaluación de herramientas.....	107
4.3.4. Herramienta recomendada.....	109
4.4. Simulación del Modelo de desarrollo de software	111
4.4.1. Modelo de simulación	111
4.4.2. Hallazgos	112
4.5. Recomendaciones para implementar el modelo de desarrollo de software	116
4.5.1. Formato para definir “User Stories”	116
4.5.2. Estructuras de Equipos de “Scrum”.....	118
4.5.3. Establecer con el “Product Owner” sus responsabilidades	119
4.5.4. Capacitar al equipo de Integración	119
4.5.5. “Sprints” mensuales.....	120
4.5.6. Gestión del cambio	120
4.5.7. Gestión del riesgo	121
5. Conclusiones	123
5.1. Limitaciones	126
5.2. Trabajos Pendientes	126
6. Anexos	127

6.1. Ideas para mejorar el proceso de desarrollo actual	127
6.2. Entrevista para determinar los estados de un tiquete.....	127
6.3. Documentos revisados	130
6.4. Entrevista para evaluar la herramienta “Redmine”	133
6.5. Entrevista para determinar las características deseables en la herramienta ..	135
6.6. Bitácora de la simulación del modelo de desarrollo	136
7. Bibliografía	141

Índice de Figuras

FIGURA 1: ORGANIGRAMA DE DINO IT TECHNOLOGIES	19
FIGURA 2: CARACTERÍSTICAS DEL “SPRINT”	31
FIGURA 3: PLANTILLA DE UN "USER STORY"	34
FIGURA 4: “GROOMING” ES UN ESFUERZO COLABORATIVO.....	35
FIGURA 5: EJEMPLO DE UN “SPRINT BACKLOG”	36
FIGURA 6: DISTRIBUCIÓN DEL TIEMPO DEL “SCRUMMASTER”	38
FIGURA 7: UN PRODUCTO CON MÚLTIPLES EQUIPOS “SCRUM”	42
FIGURA 8: TAREAS A REALIZAR PREVIO AL SPRINT REVIEW	46
FIGURA 9: MODELO METODOLÓGICO UTILIZADO EN LA INVESTIGACIÓN.....	50
FIGURA 10: GRÁFICO DE LA PREGUNTA #1 DEL CUESTIONARIO PARA CONOCER LA SITUACIÓN ACTUAL	70
FIGURA 11:GRÁFICO DE LA PREGUNTA #2 DEL CUESTIONARIO PARA CONOCER LA SITUACIÓN ACTUAL	70
FIGURA 12: GRÁFICO DE LA PREGUNTA #3 DEL CUESTIONARIO PARA CONOCER LA SITUACIÓN ACTUAL	71
FIGURA 13: GRÁFICO DE LA PREGUNTA #4 DEL CUESTIONARIO PARA CONOCER LA SITUACIÓN ACTUAL	71
FIGURA 14: GRÁFICO DE LA PREGUNTA #5 DEL CUESTIONARIO PARA CONOCER LA SITUACIÓN ACTUAL	72
FIGURA 15: GRÁFICO DE LA PREGUNTA #6 DEL CUESTIONARIO PARA CONOCER LA SITUACIÓN ACTUAL	72
FIGURA 16: GRÁFICO DE LA PREGUNTA #7 DEL CUESTIONARIO PARA CONOCER LA SITUACIÓN ACTUAL	73
FIGURA 17: GRÁFICO DE LA PREGUNTA #8 DEL CUESTIONARIO PARA CONOCER LA SITUACIÓN ACTUAL	73
FIGURA 18: GRÁFICO DE LA PREGUNTA #9 DEL CUESTIONARIO PARA CONOCER LA SITUACIÓN ACTUAL	74
FIGURA 19: GRÁFICO DE LA PREGUNTA #10 DEL CUESTIONARIO PARA CONOCER LA SITUACIÓN ACTUAL	74
FIGURA 20: GRÁFICO DE LA PREGUNTA #11 DEL CUESTIONARIO PARA CONOCER LA SITUACIÓN ACTUAL	75
FIGURA 21: GRÁFICO DE LA PREGUNTA #12 DEL CUESTIONARIO PARA CONOCER LA SITUACIÓN ACTUAL	75
FIGURA 22: GRÁFICO DE LA PREGUNTA #13 DEL CUESTIONARIO PARA CONOCER LA SITUACIÓN ACTUAL	76
FIGURA 23: GRÁFICO DE LA PREGUNTA #14 DEL CUESTIONARIO PARA CONOCER LA SITUACIÓN ACTUAL	76
FIGURA 24: GRÁFICO DE LA PREGUNTA #15 DEL CUESTIONARIO PARA CONOCER LA SITUACIÓN ACTUAL	77
FIGURA 25: GRÁFICO DE LA PREGUNTA #16 DEL CUESTIONARIO PARA CONOCER LA SITUACIÓN ACTUAL	78
FIGURA 26: DIAGRAMA DE ESTADOS DE UN TIQUETE	79

FIGURA 27: FLUJO DE EJECUCIÓN DE LOS “USER STORIES”	98
FIGURA 28: EJEMPLO DE UN TABLERO FÍSICO	111
FIGURA 29: TABLERO FÍSICO AL INICIO DE LA SIMULACIÓN	114
FIGURA 30: TABLERO FÍSICO AL FINAL DE LA SIMULACIÓN	115
FIGURA 31: RESPUESTAS DE LA PREGUNTA #15 DEL CUESTIONARIO PARA CONOCER LA SITUACIÓN ACTUAL..	127
FIGURA 32: 05 06 2016 STATUS APRIL 2016.XLSX.....	130
FIGURA 33: 7.5 - PATCH MAY.....	131
FIGURA 34: PATCH APRIL AND 7.6	132
FIGURA 35: BUG #72409	133

Índice de Tablas

TABLA 1: CUESTIONARIO PARA CARACTERIZAR EL PROCESO DE DESARROLLO ACTUAL	61
TABLA 2: ENTREVISTA PARA DETERMINAR LOS POSIBLES ESTADOS DE UN TIQUETE	62
TABLA 3: DOCUMENTOS RECOPIADOS DE DINO.....	63
TABLA 4: TABLA DE ARTEFACTOS DEL PROCESO DE DESARROLLO.....	63
TABLA 5: TABLA DE ESTADOS DE UN “USER STORY”.....	64
TABLA 6: TABLA DE ROLES DEL PROCESO DE DESARROLLO	64
TABLA 7: TABLA DE EVENTOS DEL PROCESO DE DESARROLLO.....	65
TABLA 8: ENTREVISTA PARA EVALUAR LA HERRAMIENTA “REDMINE”	66
TABLA 9: ENTREVISTA PARA DETERMINAR LAS CARACTERÍSTICAS DESEABLES EN LA HERRAMIENTA	66
TABLA 10: TABLA DE HERRAMIENTAS SELECCIONADAS	67
TABLA 11: MODELO DE SIMULACIÓN	67
TABLA 12: BITÁCORA DE LA SIMULACIÓN DEL MODELO DE DESARROLLO	68
TABLA 13: DOCUMENTOS RECOPIADOS DE DINO.....	82
TABLA 14: TABLA DE ARTEFACTOS DEL PROCESO DE DESARROLLO.....	96
TABLA 15: ESTADOS DE UN “USER STORY”	97
TABLA 16: COMPARACIÓN DE ESTADOS UTILIZADOS EN “REDMINE” VERSUS “USER STORIES”	99
TABLA 17: TABLA DE ROLES DEL PROCESO DE DESARROLLO	101
TABLA 18: TABLA DE EVENTOS DEL PROCESO DE DESARROLLO.....	105
TABLA 19: TABLA DE HERRAMIENTAS SELECCIONADAS	109
TABLA 20: DATOS DEL MODELO DE SIMULACIÓN.....	112
TABLA 21: PLANTILLA PARA LA CREACIÓN DE “USER STORIES”.....	117
TABLA 22: RIESGOS IDENTIFICADOS QUE SE PODRÍAN DAR DURANTE LA FASE DE IMPLEMENTACIÓN.....	122
TABLA 23: APLICACIÓN DE LA ENTREVISTA PARA DETERMINAR LOS ESTADOS DE UN TIQUETE	130
TABLA 24: APLICACIÓN DE LA ENTREVISTA PARA EVALUAR LA HERRAMIENTA “REDMINE”	135
TABLA 25: ENTREVISTA PARA DETERMINAR LAS CARACTERÍSTICAS DESEABLES EN LA HERRAMIENTA	136
TABLA 26: BITÁCORA DE LA SIMULACIÓN DEL MODELO DE DESARROLLO	140

1. Introducción

A continuación se describe de manera general el contenido del presente documento. Seguidamente, se presentan los detalles de la empresa, el problema y el proyecto que se va a desarrollar.

1.1. Descripción general

El presente capítulo inicia con una revisión de los antecedentes, se describe, en primera instancia, la empresa para la cual se realizó el proyecto, sus orígenes, cliente, productos, estructura organizacional y estrategia. Seguidamente se profundiza en el problema que se deseaba resolver, así como el enfoque que se propone para su resolución. La justificación describe la importancia del proyecto y hace énfasis en su impacto. Posteriormente, se define el objetivo general y sus objetivos específicos. Se delimita el proyecto por medio de la definición del alcance, y finalmente, se describen los entregables que se producirán.

En el Capítulo 2 se revisan conceptos de algunos de los autores más influyentes en la actualidad en materia de “Scrum”. Se comienza por definir que es “Scrum”, sus orígenes y beneficios. Seguidamente, se inspecciona un marco de trabajo que ayuda a identificar los escenarios en los cuales es más adecuado utilizar “Scrum”. También, se inspeccionan los distintos elementos que componen “Scrum”, esto es, sus artefactos, eventos y roles.

El Capítulo 3 describe en detalle el modelo metodológico utilizado para llevar a cabo la investigación. Comienza por describir el método y el enfoque de la investigación, para después describir cada fase y actividad del modelo. Para cada actividad, se especifica el instrumento que se utilizó para obtener la información.

El Capítulo 4 presenta el análisis de los resultados obtenidos de la aplicación de los instrumentos definidos en el Modelo Metodológico. Los resultados son presentados de acuerdo con los objetivos planteados en el proyecto.

El Capítulo 5 ofrece las conclusiones finales del proyecto. En el Capítulo 6 se adjuntan los Anexos, que son documentos complementarios que permitieron la realización del proyecto.

1.2. Antecedentes

A continuación, se describen los antecedentes del proyecto por medio de la descripción de la empresa y la descripción del problema.

1.2.1. Descripción de la empresa

Dino I.T. Technologies (de aquí en adelante llamada Dino), es una empresa costarricense, dedicada a ofrecer los servicios de desarrollo de software por medio de tercerización a una única empresa localizada en Estados Unidos. Inició operaciones en agosto de 2014 con seis personas, en la actualidad laboran dieciocho empleados.

La empresa cliente se llama Spiceworks Inc., que posee sus oficinas centrales en Austin, Texas; además de una subsidiaria en Londres. Fue fundada en el año 2006 por Scott

Abel, Jay Hallberg, Greg Kattawar y Francis Sullivan. Actualmente cuenta con más de cuatrocientos cincuenta colaboradores. Spiceworks ha ganado en los últimos cinco años el premio a la mejor compañía para trabajar en Austin, otorgado por la organización “Austin American Statesman”. Adicionalmente, ha sido calificada, por el sitio web Glassdoor, como uno de los diez mejores lugares para trabajar en Estados Unidos (Schaeffer, 2014).

El producto que ofrece Spiceworks es un sistema gratuito que proporciona monitoreo e inventario de los dispositivos electrónicos de una red. Le permite al administrador de la red tomar el control de la misma, ya que escanea dispositivos y software, tales como computadoras, dispositivos de red, servicios en la nube, licencias, dispositivos móviles, otros. Cuando Spiceworks encuentra algún problema en los dispositivos o en el rendimiento de la red entonces alerta al administrador para que tome las medidas necesarias. Además, ofrece una mesa de trabajo para los administradores de la red, que les permite organizar su trabajo por medio de tiquetes, crear portales personalizados para sus usuarios e incluso, configurar su propio sitio de gestión del conocimiento.

Adicionalmente, Spiceworks ofrece un sitio web llamado la “comunidad”, el cual es utilizado por millones de profesionales y proveedores de tecnologías de la información para ponerse en contacto entre sí. Los profesionales pueden intercambiar recomendaciones, mostrar sus proyectos y compartir consejos; mientras que los proveedores están presentes para responder preguntas sobre sus productos. La “comunidad” simplifica como los profesionales de tecnologías de la información descubren, compran y administran más de \$600 mil millones en productos de tecnología y servicios cada año (Schaeffer, 2013).

En los últimos años, Spiceworks ha enfrentado dificultades para cubrir sus necesidades de personal en Austin, lo que generó que decidiera buscar profesionales externos por medio de tercerización en Costa Rica, y fue así como nació Dino. El convenio suscrito

entre Spiceworks y Dino establece que Spiceworks va a encargarse de la administración de los proyectos, y va a delegar en el equipo costarricense el desarrollo de las tareas asignadas.

Dino está compuesta por tres equipos: Integración, Desarrollo de Aplicaciones para Dispositivos Móviles y Aseguramiento de Calidad. El equipo de Integración debe su nombre a que integra las diversas unidades de negocio que tiene Spiceworks entre sí, además de integrar diferentes aplicaciones externas. La siguiente imagen (Figura 1: Organigrama de Dino IT Technologies) muestra el organigrama de Dino.

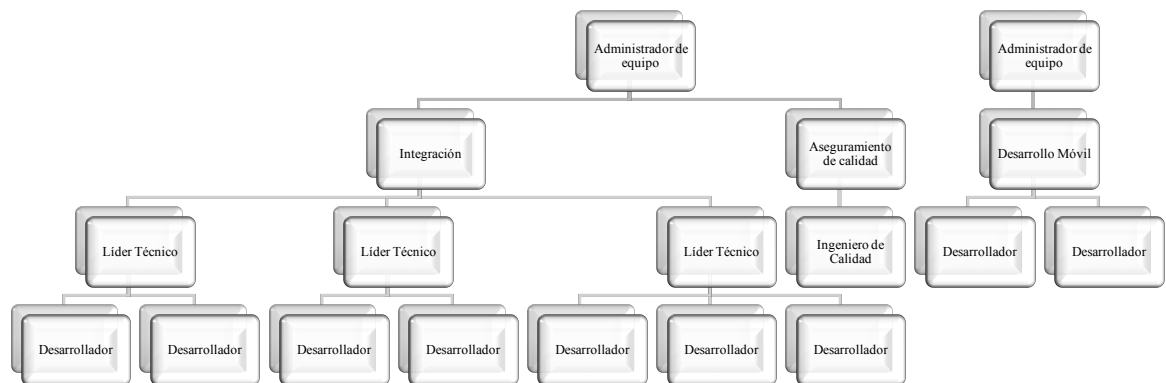


Figura 1: Organigrama de Dino IT Technologies

Fuente: Dino IT Technologies

La estructura organizacional está conformada en primera instancia por dos administradores de equipos, quienes proveen dirección y apoyo a los equipos de Integración, Aseguramiento de calidad y Desarrollo de Aplicaciones para Dispositivos Móviles. Dichos administradores son el principal punto de contacto entre Dino y Spiceworks; esto no limita

el hecho de que los colaboradores también tienen comunicación directa con el cliente por medio de correo electrónico, mensajería instantánea y videoconferencias.

Tres de sus colaboradores cuentan con más de 10 años de experiencia en desarrollo web, los líderes técnicos, quienes están a cargo de asistir a los desarrolladores con menos experiencia en sus tareas, evacuando dudas, eliminando bloqueos, proveyendo soporte técnico, funcionando como mentores, brindando capacitaciones internas, y guiando el desarrollo de las tareas asignadas.

La ventaja competitiva con la que cuenta Dino se fundamenta en la calidad del servicio ofrecido. Dino no tiene interés de ofrecer sus servicios a otros clientes, por el contrario, basa su supervivencia y crecimiento en el cumplimiento de las expectativas de su único cliente. Debido a lo anterior, la estrategia de la empresa se ha enfocado en el establecimiento de procesos y métricas que permitan mantener o aumentar la calidad de sus servicios.

1.2.2. Descripción del problema

El proceso de desarrollo de software que utiliza Spiceworks es iterativo e incremental. De manera mensual esta compañía publica una nueva versión de su producto, en el cual agrega mejoras o características nuevas, las cuales son desarrolladas tanto por las personas que se encuentran en las oficinas centrales de Spiceworks, como las que pertenecen al equipo de Integración en Dino.

El esquema de trabajo pactado entre Spiceworks y Dino consiste en que Spiceworks asigna un conjunto de tareas a Dino por medio de tiquetes y Dino por su parte, debe encargarse del desarrollo y aseguramiento de la calidad de sus asignaciones, luego son

entregadas a Spiceworks para que las acepte y las lleve a producción. Dino tiene la potestad de gestionar el trabajo que le es asignado bajo sus propios procesos de desarrollo de software.

La asignación de tiquetes por parte de Spiceworks a Dino ocurre de manera diaria, conforme van surgiendo nuevos requerimientos, defectos o mejoras. Esto significa que la etapa de asignación de tareas no está supeditada a una ventana de tiempo específica. Estas nuevas asignaciones se dan bajo el supuesto de que estarán listas para la siguiente versión del producto.

La gestión de los tiquetes asignados al equipo de Integración es desempeñada por un comité conformado por el administrador de dicho equipo y los líderes técnicos. Esta gestión involucra priorizar, asignar y dar seguimiento a los tiquetes; sin embargo, no se sigue ningún proceso formal para ello. El equipo de Integración ha detectado los siguientes problemas bajo el proceso actual:

1. Dificultad para determinar el nivel de progreso que tiene cada colaborador en sus asignaciones.
2. Se desconoce el desempeño tanto individual como a nivel de equipo.
3. No es posible presentarle al cliente informes de avance sobre el trabajo realizado.
4. De manera frecuente, las prioridades son cambiadas por el cliente, y no existe un proceso previamente definido que establezca cómo proceder ante esta situación.

El equipo de Aseguramiento de Calidad de Dino está compuesto por una persona, y su responsabilidad consiste en verificar que los tiquetes cumplan con los requerimientos establecidos, pero su aprobación no significa que el tickete está listo para salir a producción, ya que el Departamento de Aseguramiento de Calidad de Spiceworks se encarga de eso. Los tiquetes que revisa el equipo de calidad son asignados por el equipo de Integración basándose

en los criterios de prioridad, importancia y oportunidad, lo anterior significa que solo algunos de los tickets desarrollados por el equipo de Integración pasan por el proceso de aseguramiento de calidad interno. El equipo de Integración ha detectado los siguientes problemas bajo el esquema actual:

1. Tickets con errores debido a que no pasaron por el proceso de calidad interno
2. Tickets que no cumplen con los requerimientos establecidos por el cliente
3. Carga de trabajo desigual a lo largo del tiempo, en ocasiones hay tickets por verificar, y otras veces el equipo se encuentra ocioso

1.3. Definición del problema

Con el proceso de desarrollo de software actual, Dino no puede determinar el nivel de progreso sobre el trabajo asignado, por lo cual no puede presentarle informes de avance al cliente. Cuando el cliente cambia las prioridades de los tickets asignados, el trabajo que se encuentra en progreso es pausado para atender las nuevas prioridades, sin analizar las consecuencias que esto puede causar. Es desconocido el desempeño y la velocidad tanto individual como grupal.

La ventaja competitiva de Dino es la calidad del trabajo que le entrega a Spiceworks, por esta razón, las estrategias están orientadas al fortalecimiento y aumento de este hito. En la actualidad, Dino no puede asegurar por medio de su proceso interno de aseguramiento de calidad que el trabajo entregado cumple idóneamente con la calidad que exige y espera Spiceworks.

La adopción de un proceso de desarrollo de software basado en el marco de trabajo “Scrum” y una herramienta que soporte dicho proceso permitiría:

1. Medir el nivel de progreso del trabajo que se tiene asignado
2. Presentar informes de avance al cliente
3. Medir el desempeño tanto individual como grupal
4. Establecer un proceso para priorizar el trabajo
5. Incluir dentro del proceso de desarrollo, una etapa para el aseguramiento de calidad

El equipo de Integración en el pasado trató de implementar “Kanban” para soportar su proceso de desarrollo; sin embargo, no se obtuvo los resultados esperados. Debido a lo anterior, los ejecutivos de la compañía ven “Scrum” como una opción viable.

1.4. Justificación

Es de gran interés para Dino contar con procesos de desarrollo de software que le permitan ofrecerle a Spiceworks un servicio de calidad, mediante una carga equitativa y razonable de trabajo entre los colaboradores. Además de contar con métricas y reportes que le permitan determinar el desempeño de sus colaboradores y el progreso del trabajo realizado.

1.4.1. Impacto

La propuesta que plantea el proyecto vendría a modificar el proceso de desarrollo de software empleado en la actualidad, impactando la priorización, asignación, seguimiento y aseguramiento de calidad que reciben los tiquetes asignados, a fin de lograr aumentar la calidad del servicio ofrecido a Spiceworks.

1.5. Objetivos

A continuación, se especifica el objetivo principal que persigue el proyecto, así como sus objetivos específicos.

1.5.1. Objetivo General

Crear una propuesta de modelo de desarrollo de software para Dino I.T. Technologies, basado en el marco de trabajo “Scrum”.

1.5.2. Objetivos específicos

1. Analizar la situación actual del proceso de desarrollo de software
2. Diseñar un proceso de desarrollo de software
3. Identificar los roles necesarios para el proceso de desarrollo de software
4. Proponer una herramienta para soportar el proceso de desarrollo de software
5. Proponer recomendaciones para la implementación del modelo de software

1.6. Alcance

El modelo de desarrollo de software que se va a proponer se sustentará en el marco de trabajo “Scrum”, y estará adecuado para satisfacer las necesidades del equipo de Integración y Aseguramiento de Calidad específicamente. El modelo incluirá la descripción de los roles necesarios.

El modelo no abarcará el tema de las estimaciones de los tiquetes, Este será abordado de forma paralela en otra propuesta.

La implementación del modelo de desarrollo de software está fuera del alcance de este proyecto, debido a limitaciones con el tiempo establecido para la finalización del mismo.

1.7. Entregables

A continuación, se citan los entregables esperados del proyecto:

1. Análisis de la situación actual
2. Propuesta del proceso de desarrollo de software, basado en “Scrum”
3. Propuesta de una herramienta para soportar el proceso de desarrollo
4. Descripción de los roles necesarios para el proceso de desarrollo de software propuesto
5. Recomendaciones para la implementación del modelo

2. Marco Teórico

2.1. ¿Que es “Scrum”?

Según (S.Rubin, 2013), “Scrum” es un enfoque ágil para desarrollar productos y servicios innovadores. Ahora bien, según (Sliger, 2012) ágil se entiende como una filosofía de desarrollo de productos que usa modelos organizacionales basados en las personas, la colaboración y los valores compartidos.

(Sutherland, 2014) afirma que “Scrum” es un marco de trabajo que pone en práctica los valores que define el “Agile Manifesto”, además, es radical en indicar que no es una metodología. El “Agile Manifesto” fue escrito por un cónclave de diecisiete personas (incluido el mismo Sutherland), quienes eran líderes en el desarrollo de software en el año 2001. En dicho “manifiesto” se declaran los siguientes valores:

- ✓ Individuos e iteraciones sobre procesos y herramientas
- ✓ Software funcionando sobre documentación extensiva
- ✓ Colaboración con el cliente sobre negociación contractual
- ✓ Respuesta ante el cambio sobre seguir un plan

“Scrum” no es un proceso estandarizado donde periódicamente se siguen una serie de pasos secuenciales que garantizan producir en el tiempo y el presupuesto estimado productos de alta calidad. “Scrum” es un marco de trabajo para organizar y manejar el trabajo, se centra en los valores de la honestidad, apertura, respeto, enfoque, empoderamiento y colaboración (S.Rubin, 2013).

Los orígenes de “Scrum” se remontan hacia 1986, cuando la revista “Harvard Bussiness Review” publicó un artículo llamado “The New Product Development Game”, que

fue escrito por los profesores Hirotaka Takeuchi y Ikujiro Nonaka. Dicho artículo describía cómo compañías como Honda, Canon y Fuji-Xerox producían resultados de clase mundial utilizando un enfoque escalable, basado en el trabajo en equipo. Dicho artículo también enfatizaba en la importancia del empoderamiento y la auto organización de los equipos de trabajo, además de describir el rol de la administración en el proceso de desarrollo (S.Rubin, 2013).

Según (Sutherland, 2014), el artículo escrito por lo los profesores Takeuchi y Nonaka describía las características que ellos observaron en los equipos de las mejores compañías en el mundo, las cuales fueron:

Trascendencia: ellos tienen un sentido de propósito más allá de lo ordinario. Este objetivo auto realizado les da oportunidad de moverse de lo ordinario a lo extraordinario. Es una decisión que permite salirse del promedio y ser mejores, esto cambia la forma en cómo se ven a sí mismos y se percatan de lo que son capaces.

Autonomía: los equipos son auto organizados y auto gestionados, ellos tienen el poder de tomar decisiones sobre el trabajo que realizan.

Multifuncionales: el equipo tiene todas las habilidades requeridas para completar el proyecto. Las habilidades son mejoradas y reforzadas unos sobre otros.

El artículo mencionado fue una gran influencia en el entretejimiento de lo que se conoce hoy en día como “Scrum”. “Scrum” es un término que se tomó prestado del deporte rugby, donde se refiere a una forma de reanudar el partido después de una infracción accidental o cuando el balón ha salido fuera del campo (S.Rubin, 2013).

2.1.1. Beneficios de “Scrum”

(S.Rubin, 2013) afirma que las compañías que han aplicado “Scrum” han experimentado los siguientes beneficios:

Clientes satisfechos: pues reciben lo que ellos realmente quieren y necesitan.

Mejora en el retorno de la inversión: debido a que se pueden producir resultados de manera más frecuente.

Reducción de costos: rápidamente se hacen evidentes disfunciones y gastos innecesarios.

Resultados rápidos: “Scrum” se enfoca en entregar características que dan valor al negocio en cortos periodos de tiempo y de manera constante.

Confianza de tener éxito en un mundo complejo: permite a las organizaciones ajustarse rápidamente a los cambios del entorno.

Más diversión: no solo los clientes se sienten satisfechos, también el equipo de trabajo al colaborar de una manera significativa y frecuente.

2.1.2. ¿Cuándo utilizar “Scrum”?

Aunque “Scrum” es una excelente solución para muchas situaciones, no es la opción apropiada para todas las circunstancias.

El marco de trabajo “Cynefin” es considerado como una de las primeras aplicaciones prácticas de la teoría de la complejidad a la ciencia de la dirección. Su significado pretende recordar que nuestras raíces (culturales, religiosas, geográficas) influyen profundamente y determinan nuestro comportamiento y nuestras acciones. (Ignacio, Herrera, & Guillén, 2014)

Este marco de trabajo ayuda a interiorizar el hecho de que todas las situaciones son distintas, por lo cual, se requieren diferentes soluciones para afrontarlas con éxito (Wester, 2013).

El marco de trabajo “Cynefin” ayuda a comprender la situación en la cual opera una compañía, para así decidir cuál es el enfoque más apropiado. Este marco de trabajo define y compara las características de cinco dominios diferentes: simple, complicado, caótico, complejo y desordenado (S.Rubin, 2013).

(S.Rubin, 2013) realiza un análisis y expone sobre cuales dominios es más adecuado utilizar “Scrum”:

Complejo: lo que aquí sucede es más impredecible, requiere que se aprenda de los problemas, por medio de la inspección y la adaptación de lo aprendido. Trabajar bajo este dominio exige enfoques innovadores y creativos. Lo rutinario simplemente no funciona. En este tipo de ambientes los niveles de interacción y comunicación son esenciales.

Bajo este dominio se requiere de experiencia para afrontar las situaciones; sin embargo, el trabajo es evolutivo, no revolucionario (Wester, 2013).

El desarrollo de productos innovadores cae dentro de esta categoría, así como el mejoramiento de productos existentes con características innovadoras. “Scrum” es muy adecuado para operar en dominios complejos.

Complicado: Los problemas complicados son del dominio de las buenas prácticas dominada por expertos, en donde pueden existir múltiples respuestas correctas, pero es necesario el diagnóstico experto para poder encontrarlas. Por ejemplo, problemas de optimización. Para este dominio es más conveniente la utilización de “Six Sigma”.

Bajo este dominio, sin importar el tiempo invertido en análisis, no es posible predecir los riesgos, la solución y el esfuerzo requerido para resolver el problema (Wester, 2013).

Simple: en este dominio se tratan problemas simples, donde cualquiera puede ver las causas y los efectos. A menudo, las respuestas correctas son obvias y poco discutidas, además de que existen soluciones conocidas.

Bajo el dominio simple, la resolución de problemas requiere de experiencia mínima. Comúnmente los problemas de mesa de ayuda caen en esta categoría, donde existen guiones pre escritos que establecen cómo solucionar el problema (Wester, 2013).

“Scrum” puede ser utilizado en este dominio, pero puede no ser la herramienta más eficiente.

Caótico: Los problemas caóticos requieren respuestas rápidas, en medio de una crisis es necesario actuar de manera inmediata para al menos reestablecer el orden.

Bajo el dominio caótico, las soluciones que se aplican puede que no sean las mejores, siempre y cuando funcionen y sean suficientes para solucionar los problemas en el menor tiempo posible (Wester, 2013).

“Scrum” no es la mejor solución en este dominio.

Desordenado: se está en este dominio cuando no se encaja en alguno de los dominios antes descritos. Este es un lugar peligroso, debido a que no se puede dar sentido a la situación actual. La manera de abordar este dominio es descomponiendo la situación en distintas partes que se asignan a alguno de los otros cuatro dominios.

Bajo este dominio no se aplica “Scrum”, se trata primeramente salir de dicho estado.

2.2. Artefactos de “Scrum”

El marco de trabajo “Scrum” está compuesto por un conjunto de artefactos que son descritos a continuación.

2.2.1. “Sprint”

Según lo describe (S.Rubin, 2013), “Scrum” organiza el trabajo en iteraciones o ciclos, que tienen una duración máxima de un mes calendario. (Sutherland, 2014) afirma que *“Scrum funciona por medio de la definición de objetivos secuenciales que deben ser completados en una cantidad de tiempo fija”*. (Sutherland, 2014) establece además, que al final de cada “Sprint” debe existir un incremento sobre el producto trabajado, esto significa que se tendrá algo funcional para ser mostrado a cualquiera que pueda estar interesado e idealmente, para quienes usan el producto.

(S.Rubin, 2013), detalla que el objetivo de cada “Sprint” se fundamenta en el compromiso mutuo que se realiza entre el “Product Owner” y el equipo de trabajo. El equipo de trabajo se compromete a cumplir el objetivo planteado, y el “Product Owner” por su parte, se compromete a no cambiar dicho objetivo durante el “Sprint”.

La siguiente imagen (Figura 2: Características del “Sprint”), muestra como los “Sprints” son ciclos secuenciales, todos con un mismo tamaño. Además, establece que la duración de un “Sprint” debe ser de entre una semana a un mes calendario.

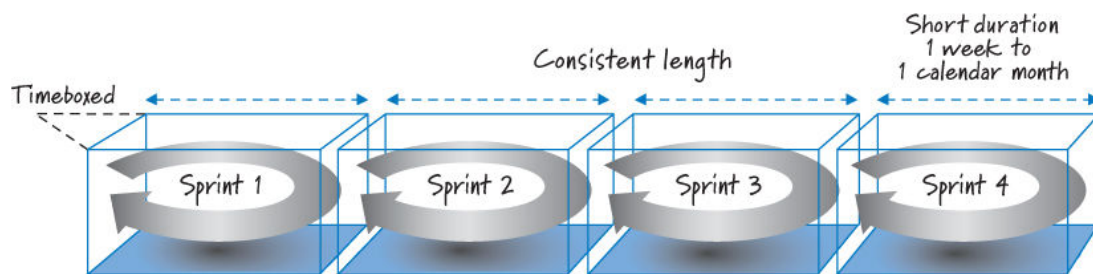


Figura 2: Características del “Sprint”

Fuente: (S.Rubin, 2013)

Según (S.Rubin, 2013), las ventajas de tener “Sprints” con un tamaño fijo establecido son:

- Se limita la cantidad de trabajo en progreso
- Se fuerza a la priorización
- Se hace más evidente el progreso realizado
- Evita perfeccionismos innecesarios
- Motivación al dejar atrás una etapa

Las ventajas de que los “Sprints” tengan una duración pequeña son:

- Facilita la planificación
- Se obtiene retroalimentación de manera rápida
- Se acota el margen de error
- Se mejora el retorno de la inversión
- El equipo rejuvenece su entusiasmo periódicamente

2.2.2. “User Stories”

Para (S.Rubin, 2013), en “Scrum” los detalles de los requerimientos son negociados a través de conversaciones que suceden de manera constante durante el desarrollo. “Scrum” les da a los requerimientos la libertad de que puedan ser manipulados en función de satisfacer las necesidades del negocio, contrario a lo que ocurre en procesos de desarrollo secuencial (como “Cascada”), donde los requerimientos son definidos en las etapas iniciales e idealmente no deberían cambiar. Cuando se utiliza “Scrum”, no es necesario invertir gran cantidad de tiempo definiendo detalles de los requerimientos de manera anticipada, porque se espera que estos cambien con el paso del tiempo, conforme se aprende más de lo que se está desarrollando.

(S.Rubin, 2013), describe un “User Story” como un formato utilizado para expresar el valor del negocio deseado en un producto. Los “User Stories” son escritos de manera que puedan ser entendidos por cualquier persona que los lea. Son estructuralmente simples y pueden ser escritos con varios niveles de granularidad, lo cual los hace fáciles de refinar progresivamente.

(Jeffries, 2001), establece que los “User Stories” tienen tres aspectos críticos:

1. **Tarjeta:** Son escritos en tarjetas o “post its”. Estas tarjetas no contienen toda la información del requerimiento, en su lugar, contiene un texto que es suficiente para identificar el requerimiento.
2. **Conversación:** El requerimiento es comunicado del cliente al equipo de trabajo a través de conversaciones, es un intercambio de pensamientos, opiniones y sentimientos. Esta conversación se lleva a cabo a través del tiempo y más particularmente, cuando el “User Story” es estimado.
3. **Confirmación:** Se realiza por medio de las pruebas de aceptación, las cuales describen qué es lo que se va a probar para verificar que el “User Story” fue implementado de manera correcta.

La imagen Figura 3: Plantilla de un "User Story", es una guía que describe como un “User Story” debería escribirse. En primer lugar, requiere un título que permita identificarlo. Ahora bien, la descripción del “User Story” debería comenzar por especificar el usuario que se verá beneficiado, seguidamente describiría el objetivo por lograr y finalmente, el beneficio que se obtendrá.



Figura 3: Plantilla de un "User Story"

Fuente: (S.Rubin, 2013)

2.2.3. “Product Backlog”

(Sutherland, 2014) afirma que el primer requerimiento para implementar “Scrum” es un “Backlog”. La idea detrás del “Backlog” es que debería contener todo lo que posiblemente tendrá un producto, reflejar su visión. Para (Schwaber, 2004), los requerimientos para el sistema que se está desarrollando deberían estar listados en el “Product Backlog”, y el “Product Owner” es el responsable de sus contenidos y priorización. (Schwaber, 2004) también afirma que el “Product Backlog” nunca está completamente terminado. (S.Rubin, 2013) lo describe como el corazón de “Scrum”, esto porque provee un entendimiento centralizado y compartido de lo que se quiere construir.

Características de un buen “Product Backlog”

Las características de un buen “Product Backlog”, según (S.Rubin, 2013) son:

- No todos los ítems se encuentran al mismo nivel de detalle. Los ítems que se van a trabajar pronto deberían encontrarse bien detallados y deberían ser

pequeños. Los ítems que no se trabajarán pronto pueden ser grandes y no contener muchos detalles.

- El “Product Backlog” debería ser actualizado en forma constante, basado en el flujo de información con valor económico que asiduamente se genera. Cuando nuevos ítems son agregados, o los existentes modificados, el “Product Owner” debe re balancear y re priorizar el “Product Backlog”.
- Todos los ítems del “Product Backlog” tienen que estar estimados

“Grooming”

Esta es una actividad que busca crear, refinar, estimar y priorizar los ítems del “Product Backlog”. Este es un esfuerzo continuo y colaborativo, liderado por el “Product Owner” e incluye la participación de interesados externos e internos, el ScrumMaster y el equipo de trabajo (S.Rubin, 2013).

La siguiente imagen (Figura 4: “Grooming” es un esfuerzo colaborativo) muestra como el “Grooming” es un esfuerzo continuo y colaborativo, donde se ven involucrados distintos actores, y tiene como finalidad priorizar, estimar, y refinar el “Product Backlog”.

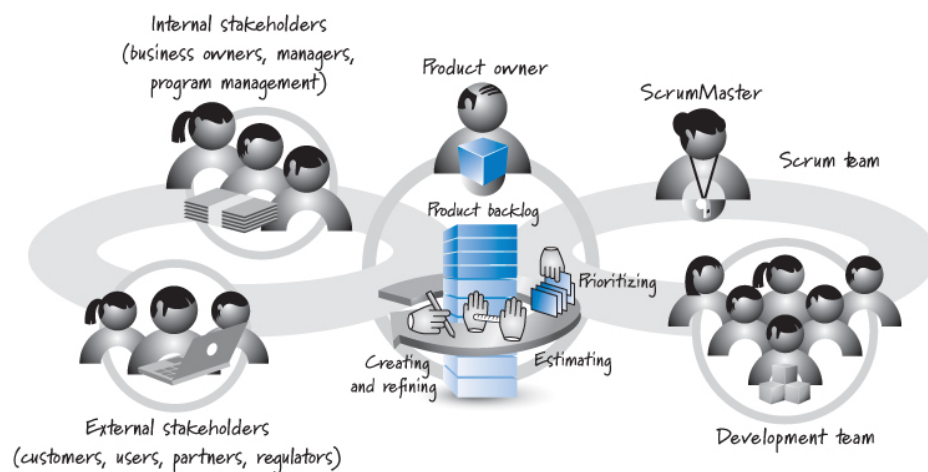


Figura 4: “Grooming” es un esfuerzo colaborativo

Fuente: (S.Rubin, 2013)

2.2.4. “Sprint Backlog”

(Schwaber, 2004) define el “Sprint Backlog” como el trabajo o las tareas que son seleccionadas en un “Sprint” para convertirlas en un incremento potencial del producto. Todas las tareas deberían estar divididas de manera que tomen de entre 4 a 16 horas para ser finalizadas. El “Sprint Backlog” es una fotografía en tiempo real del trabajo que el equipo planea lograr en el “Sprint”. (S.Rubin, 2013) argumenta que el “Sprint Backlog” describe a través de tareas detalladas cómo el equipo planea diseñar, construir, integrar y probar los “User Stories” seleccionados del “Product Backlog”.

La imagen que se muestra a continuación (Figura 5: Ejemplo de un “Sprint Backlog”) ejemplifica como debería ser un “Sprint Backlog”. Se puede apreciar que existe una lista de características (“User Stories”) éstas son las que se van a desarrollar en el “Sprint”, así mismo, cada característica se encuentra dividida en tareas que especifican su duración aproximada en horas.

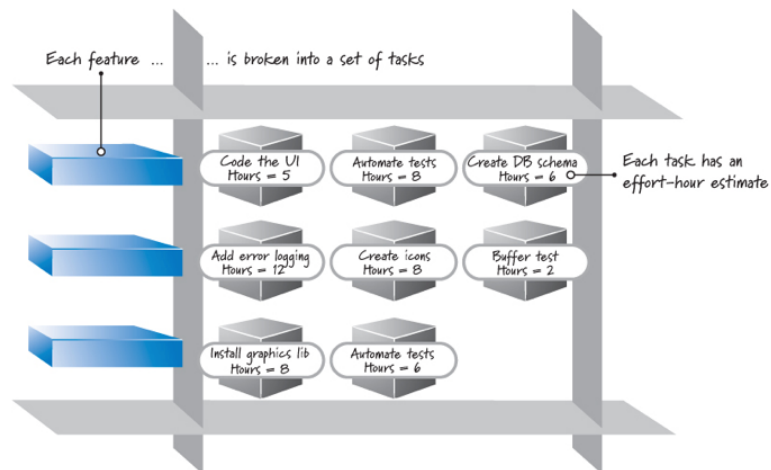


Figura 5: Ejemplo de un “Sprint Backlog”

Fuente: (S.Rubin, 2013)

2.3. Roles de “Scrum”

Según lo describe (Schwaber, 2004), en “Scrum” existen solo tres roles, el “ScrumMaster”, el “Product Owner” y el “Equipo de Trabajo”. Todas las actividades administrativas en el proyecto son divididas entre estos tres roles.

2.3.1. “ScrumMaster”

El “ScrumMaster” ayuda a las personas envueltas en el proyecto a comprender y adoptar los valores, principios y prácticas de “Scrum”. Como facilitador, el “ScrumMaster” ayuda al equipo a resolver inconvenientes, y a realizar mejoras en el uso de “Scrum”. Es responsable de proteger al Equipo de Trabajo de interferencia externa y toma el rol de liderazgo al remover impedimentos que afectan su productividad. El “ScrumMaster” no tiene autoridad para ejercer control sobre el equipo. Este rol no es el mismo del tradicionalmente conocido “Administrador de Proyecto” o “Administrador de Desarrollo”. El “ScrumMaster” funciona como líder, no como administrador. (S.Rubin, 2013)

Según (Schwaber, 2004), el “ScrumMaster” es en gran medida responsable por el éxito del proyecto y no gana medallas debido pues es solo un facilitador.

(S.Rubin, 2013) señala que el “ScrumMaster” debería tener las siguientes características:

- ✓ *Experto*: tener conocimiento sobre “Scrum”.
- ✓ *Inquisitivo*: saber utilizar sus habilidades como mentor, sus conocimientos sobre “Scrum” y su conocimiento sobre el negocio para hacer preguntas
- ✓ *Paciente*: darle al Equipo de Trabajo el tiempo que ellos necesitan para comprender las respuestas a las preguntas que surgen

- ✓ *Colaborativo*: las habilidades de colaboración son indispensables para trabajar tanto con el Equipo de Trabajo, el “Product Owner”, así como con otros involucrados
- ✓ *Transparente*: promueve una comunicación transparente tanto dentro como fuera del equipo

La siguiente imagen (Figura 6: Distribución del tiempo del “ScrumMaster”) muestra la distribución del tiempo en las diferentes tareas que lleva a cabo el “ScrumMaster”. Se puede apreciar que durante el primer y último día del “Sprint” es cuando debe dar más asistencia en las actividades de “Scrum” (“Sprint Planning”, “Sprint Restrospective”, “Sprint Review”). Por otro lado, gran cantidad de su tiempo es invertido en remover obstáculos, además de darle asistencia tanto al Equipo de Trabajo como al “Product Owner”.

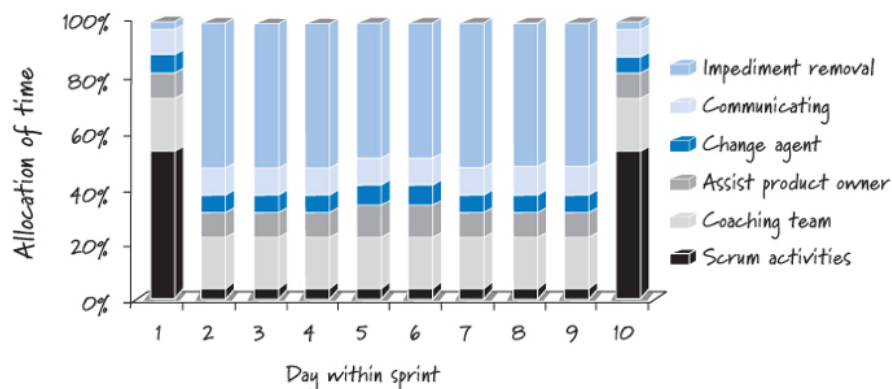


Figura 6: Distribución del tiempo del “ScrumMaster”

Fuente: (Jeffries, 2001)

2.3.2. “Product Owner”

Para (S.Rubin, 2013), el “Product Owner” es el único responsable de decidir qué características y funcionalidades se van a construir o agregar a un producto, además del orden de prioridad de las mismas. El “Product Owner” mantiene y comunica a todos los

involucrados (“Stakeholders”) una visión clara de lo que el Equipo de Trabajo está logrando. Es el principal responsable del éxito general de la solución que se está creando o manteniendo.

Para asegurarse que el Equipo de Trabajo construya rápidamente lo que el “Product Owner” quiere, el mismo debe colaborar activamente con el “ScrumMaster” y el Equipo de Trabajo. Debe estar disponible para responder preguntas tan pronto como sean planteadas.

(Sutherland, 2014) sostiene que el “Product Owner” necesita estar continuamente recolectando retroalimentación de los clientes y dándosela a conocer al Equipo de Trabajo. La mitad del tiempo del “Product Owner” debería invertirse en conocer la opinión de los clientes e involucrados en el producto y la otra mitad debería definir lo que el cliente quiere en futuras versiones del mismo.

El “Product Owner” no tiene autoridad y nadie le reporta a él, más bien él es quien reporta a otros grupos de personas sobre el avance del proyecto. El “Product Owner” no evalúa el desempeño, aumentos o ascensos.

Las características esenciales de un “Product Owner”, según (Sutherland, 2014) son:

- Debería tener conocimiento acerca del dominio. Debería comprender el proceso que el equipo está ejecutando, lo suficiente, como para saber qué puede hacerse y qué no. Debe tener la capacidad de traducir lo que puede convertirse en una realidad, y que a su vez tenga valor.
- Tiene que tener el poder de tomar decisiones
- Debe estar disponible en cualquier momento que el equipo lo requiera, y explicar qué debe estar listo y por qué

- Debe generar valor, en el contexto de negocios lo que importa son las ganancias

2.3.3. El Equipo de Trabajo

El desarrollo de software tradicional incluye varios tipos de trabajos, como lo es el arquitecto, el programador, el ingeniero de calidad, el administrador de bases de datos, otros. “Scrum” define el rol del “Equipo de Trabajo”: es un grupo diverso y funcional de personas, quienes son responsables de diseñar, desarrollar y probar el producto deseado (S.Rubin, 2013).

El “Equipo de Trabajo” debe ser capaz de auto organizarse con el fin de lograr los objetivos pactados con el “Product Owner”. Los miembros del Equipo de Trabajo deben tener, de manera colectiva, todas las habilidades necesarias para crear un producto de calidad.

(Sutherland, 2014) establece que el tamaño del equipo para un mejor funcionamiento debe oscilar entre siete y nueve personas.

2.3.4. Estructuras de equipos de “Scrum”

Según (S.Rubin, 2013), si se trabaja sobre un producto pequeño, basta con crear un Equipo de Trabajo como el antes descrito; sin embargo, si se trabaja sobre un producto o proyecto grande, se hace necesario coordinar el trabajo con múltiples equipos de “Scrum”. El autor propone crear equipos encargados de agregar características nuevas al producto y equipos dedicados a desarrollar componentes o subsistemas.

Equipo de características

Este es un equipo multifuncional, puede crear características nuevas al producto de principio a fin, debido a que sus miembros poseen diversas habilidades. “Scrum” favorece en mayor medida la creación de este tipo de equipos.

Equipo de componentes

Estos equipos, usualmente, están conformados por un conjunto de personas con habilidades similares y muy específicas. Se encargan del desarrollo de componentes o sub sistemas muy concretos. Un ejemplo podría ser un equipo dirigido a desarrollar únicamente características relacionadas con localización geográfica.

(Schwaber, 2004) menciona que previo a realizar una estructura de equipos de “Scrum” resulta necesario contar con una infraestructura de comunicación apropiada, especialmente, cuando existen equipos de trabajo remotos, debido a que deben existir mecanismos de sincronización periódicos y frecuentes. El producto sobre el cual trabajan los distintos equipos debe poseer una arquitectura técnica idónea, de manera que permita dividir el trabajo.

La imagen que se muestra a continuación (Figura 7: Un producto con múltiples equipos “Scrum”) ejemplifica como tres distintos equipos de “Scrum” trabajan en distintos tipos de componentes. Existe un “Product Backlog” principal, este contiene los componentes por desarrollar descritos a groso modo. Cada equipo, por su parte, cuenta con su propio “Backlog”, que describe en detalle el componente por desarrollar.

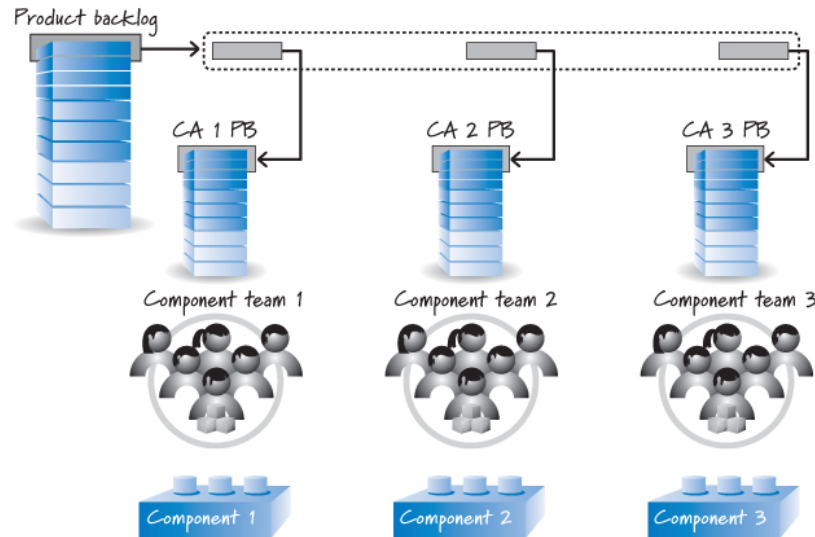


Figura 7: Un producto con múltiples equipos “Scrum”

Fuente: (S.Rubin, 2013)

Coordinación entre múltiples equipos de “Scrum”

Para (S.Rubin, 2013), “Scrum” escala no para tener muchos equipos; sino más bien, para tener equipos con el tamaño adecuado. Cuando se tiene más de un equipo “Scrum” se pueden presentar dificultades para coordinar los esfuerzos de todos los equipos.

“Scrum of Scrums”: (Schwaber, 2004) establece que este mecanismo consiste en que un miembro de cada equipo “Scrum” asiste a una reunión diaria donde el objetivo es coordinar las dependencias o bloqueos que pueden existir entre los distintos equipos.

Según (S.Rubin, 2013), las personas presentes en esta reunión deben responder tres preguntas:

1. ¿Qué ha hecho mi equipo desde la última reunión que pueda afectar a otros equipos?
2. ¿En qué va a seguir trabajando mi equipo que puede afectar otros equipos?
3. ¿Qué problemas está teniendo mi equipo de los cuales podría necesitar la ayuda de otros equipos?

2.4. Actividades de “Scrum”

Un proyecto que se desarrolla utilizando el marco de trabajo “Scrum” típicamente está compuesto por múltiples “Sprints”, cada una de las cuales producen valor al cliente final. Durante cada “Sprint” suceden una serie de actividades que define “Scrum”, las cuales se detallan a continuación.

2.4.1. “Daily Scrum”

Según (S.Rubin, 2013), ésta es una actividad que debe ocurrir cada día del “Sprint”, idealmente a la misma hora. Consiste en una reunión de no más de 15 minutos en la cual cada miembro del equipo responde tres preguntas:

1. ¿Qué he logrado desde el último “Daily Scrum”?
2. ¿Qué planeo seguir trabajando?
3. ¿Cuáles son los obstáculos que me impiden seguir avanzando?

Al responder estas preguntas, todas las personas obtienen una imagen de lo que ocurre en el proyecto, cómo se está progresando hacia el objetivo del “Sprint”, cuáles cambios se planean realizar en los días venideros y, lo más importante de todo, cuáles dificultades se deben resolver.

Esta actividad no es para solucionar problemas, en su lugar, muchos Equipos de Trabajo deciden abordar los problemas después del “Daily Scrum”, con un grupo pequeño de personas quienes pueden ayudar a resolver la dificultad.

(Schwaber, 2004) argumenta que el “Daily Scrum” debería:

- Ser siempre en el mismo lugar
- Todos los miembros del equipo deben asistir

- El “ScrumMaster” es el encargado de iniciar la reunión a la hora pactada, sin importar quien está presente
- Durante esta reunión, solo una persona habla a la vez
- A la reunión pueden asistir personas interesadas en el proyecto; sin embargo, estas no pueden hablar, solo el Equipo de Trabajo

2.4.2. “Sprint Planning”

Según (S.Rubin, 2013), el “Product Backlog” puede representar semanas o meses de trabajo, lo cual es mucho tiempo para poder ser completado en un solo “Sprint”. El “Sprint Planning” es la primera actividad que se realiza cuando comienza un “Sprint”. Durante esta reunión el Equipo de Trabajo acuerda cual será el objetivo del “Sprint” (a lo que se compromete). (Schwaber, 2004) agrega, además, que esta reunión no debería tardar más de 8 horas y debe constar de dos segmentos:

1. Seleccionar los ítems que se trabajarán del “Product Backlog”
2. Preparar el “Sprint Backlog”

(Schwaber, 2004) resalta los siguientes aspectos por considerar a la hora de llevar a cabo el “Sprint Planning”:

- Quienes deben atender esta reunión son el “ScrumMaster”, el “Product Owner” y el Equipo de Trabajo
- El “Product Owner” debe preparar el “Product Backlog” previo a la reunión
- En el primer segmento de la reunión, el equipo selecciona los ítems a los cuales se compromete para el “Sprint”

- El Equipo de Trabajo puede hacer sugerencias sobre que ítems se pueden agregar al “Sprint”; sin embargo, la decisión final recae sobre el “Product Owner”
- El Equipo de Trabajo es responsable de considerar qué tanto del “Product Backlog” puede trabajar en el “Sprint”

2.4.3. “Sprint Review”

Para (Schwaber, 2004), el propósito del “Sprint Review” es mostrar al “Product Owner” y a los involucrados en el proyecto, el trabajo finalizado por el equipo durante el “Sprint”. Todo aquello que no se finalizó no puede ser presentado. Al final de la presentación, los involucrados en el proyecto pueden comentar, dar retroalimentación, criticar o realizar observaciones. Los involucrados también pueden identificar nueva funcionalidad conforme avanza la presentación, y ésta es agregada al “Product Backlog”.

El “Sprint Review” es una de las últimas actividades que ocurren al finalizar el “Sprint” y provee a todos los involucrados en el proyecto la oportunidad de inspeccionar y adaptar lo construido hasta el momento. Este evento provee una mirada transparente del estado actual del proyecto; puede incluir verdades inconvenientes. El “Sprint Review” es una de las actividades más importantes de “Scrum”, pues es un ciclo de aprendizaje continuo (S.Rubin, 2013).

La siguiente imagen (Figura 8: Tareas a realizar previo al Sprint Review), muestra las actividades sugeridas para preparar el “Sprint Review”, donde primeramente se debe escoger a quien se debe invitar, seguidamente se agenda la actividad y se confirma que fue finalizado

en el “Sprint”. Finalmente, se debe preparar la demostración y definir quién va a presentar qué.

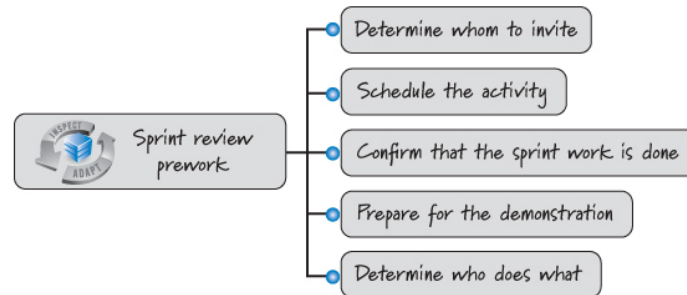


Figura 8: Tareas a realizar previo al Sprint Review

Fuente: (S.Rubin, 2013)

2.4.4. “Sprint Retrospective”

(S.Rubin, 2013) describe el “Sprint Retrospective” como una actividad para inspeccionar y adaptar el proceso de “Scrum” (Schwaber, 2004) afirma que esta reunión no debe durar más de tres horas, y deben asistir únicamente el Equipo de Trabajo, el “ScrumMaster” y el “Product Owner” (siendo el “Product Owner” opcional). La reunión debería comenzar preguntando a todos los miembros del equipo lo siguiente:

1. ¿Qué se hizo bien durante el “Sprint” pasado?
2. ¿Qué se puede mejorar en el siguiente “Sprint”?

El “ScrumMaster” es quien tiene la responsabilidad de tomar apuntes de manera resumida. De las mejoras potenciales que surjan, se debe generar una discusión para producir ítems de acción, no funcionales y agregarlos en el siguiente “Sprint”.

(S.Rubin, 2013) por su parte, argumenta que el “Sprint Retrospective” es una de las prácticas más importantes de “Scrum”; sin embargo, la menos apreciada, debido a que no es tiempo empleado en realizar “trabajo real”. Afirma, además, que es una actividad primordial

pues da la oportunidad al equipo de personalizar “Scrum” de acuerdo con sus circunstancias y necesidades.

3. Desarrollo metodológico

(Hernández, Fernández, & Baptista, 2010) definen la investigación como “*un conjunto de pasos sistemáticos, críticos y empíricos que se aplican al estudio de un fenómeno*”. El método de investigación lo define (Bernal, 2010) como el conjunto de aspectos operativos del proceso investigativo.

Los métodos de investigación pueden dividirse en cuantitativo o investigación cuantitativa y cualitativa o investigación cualitativa. El método cuantitativo o tradicional se basa en la medición de las características de fenómenos sociales, lo cual supone derivar de un marco conceptual pertinente al problema analizado, una serie de postulados que expresen relaciones entre las variables estudiadas de manera deductiva (Bernal, 2010).

El método cualitativo por su parte, se orienta a profundizar casos específicos y no generalizar. Su preocupación no es tanto medir, sino cualificar y describir un fenómeno social a partir de rasgos determinantes. Los investigadores que utilizan el método cualitativo buscan comprender una situación social como un todo, tomando en cuenta sus propiedades y dinámica (Bernal, 2010).

Según (Hernández, Fernández, & Baptista, 2010), algunas de las características del método cualitativo son:

- El estudio se basa en sí mismo y no en estudios previos
- Pretende describir, comprender e interpretar fenómenos a través de percepciones producidas por las experiencias de los participantes
- Puede ser que las preguntas de la investigación y la hipótesis se desarrollen antes, durante o después del análisis y la recolección de datos

- En lugar de iniciar con una teoría particular y luego confirmar si esta es apoyada por los hechos, el investigador comienza examinando el mundo social
- El investigador cualitativo utiliza técnicas para recolectar datos como la observación no estructurada, entrevistas abiertas, revisión de documentos, discusión en grupo, evaluación de experiencias personales, entre otros.

El método utilizado para llevar a cabo la investigación en el presente proyecto es de tipo cualitativo, busca comprender la situación actual a partir de los fenómenos que están ocurriendo, para así, posteriormente, realizar el análisis e interpretación de datos. Además, esta investigación se basa en sí misma y no en previas que pudiesen existir.

Por otro lado, el enfoque utilizado en la investigación es “Investigación-Acción”, la cual según lo describe (Hernández, Fernández, & Baptista, 2010) busca resolver problemas cotidianos e inmediatos, además de mejorar prácticas concretas. Su propósito fundamental se centra en aportar información que guíe la toma de decisiones para programas, procesos y reformas estructurales.

“La investigación-acción envuelve la transformación y mejora de una realidad (social, educativa, administrativa, etc.). De hecho, se construye desde ésta.” (Hernández, Fernández, & Baptista, 2010)

El diagrama que se muestra a continuación (Figura 9: Modelo metodológico utilizado en la investigación) presenta el modelo metodológico utilizado en la investigación. En los siguientes apartados de este capítulo se explicará en detalle cada etapa, así como las actividades involucradas en cada una de ellas.

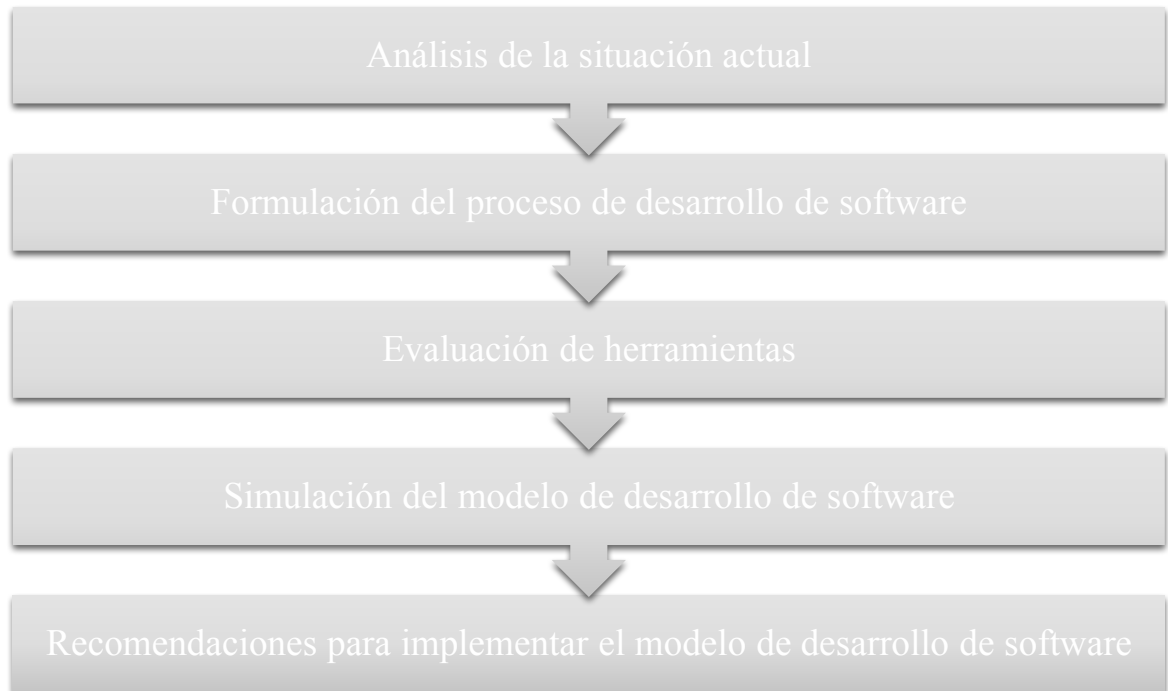


Figura 9: Modelo metodológico utilizado en la investigación

Fuente: Elaboración propia

3.1. Análisis de la situación actual

Esta es la primera etapa de la investigación, consiste en analizar y comprender cuál es la situación actual del proceso de desarrollo en el equipo de Integración en Dino I.T. Technologies. A continuación, se describen las actividades de esta etapa.

3.1.1. Caracterizar el proceso de desarrollo actual

Esta actividad tiene como finalidad determinar las características del proceso de desarrollo que emplea el equipo de Integración en la actualidad, además, identificar si dichas características son de alguna manera semejantes a los elementos y características de “Scrum”.

La técnica de recolección de datos empleada en esta actividad es el cuestionario, el cual se define en la sección 3.6.1: Cuestionario. La herramienta utilizada se expone en la sección 3.7.1: Cuestionario para caracterizar el proceso de desarrollo actual.

3.1.2. Determinar los estados de un ticket

Esta actividad busca identificar cuáles son los estados por los cuales pasan los tickets en el equipo de Integración, y adicionalmente, establecer si es posible agregar, modificar o eliminar estados.

La técnica de recolección de datos utilizada en esta actividad es la entrevista, la cual se define en la sección 3.6.2: Entrevista. El instrumento utilizado se presenta en la sección 3.7.2: Entrevista para determinar los estados de un ticket.

3.1.3. Revisar documentos

Esta actividad pretende revisar documentos relacionados con el proceso de desarrollo del equipo de Integración, con el fin de encontrar información que pueda dar aún más entendimiento de dicho proceso y sus características.

La técnica de recolección de datos utilizada es la revisión de documentos, ésta se define en la sección 3.6.3: Revisión de documentos. El instrumento empleado en esta actividad se expone en la sección 3.7.3: Tabla de documentos recopilados.

3.2. Formulación del proceso de desarrollo de software

En esta etapa se formula el proceso de desarrollo de software, el cual se basa en los fundamentos descritos en el Capítulo del Marco Teórico (2: Marco Teórico), así como en el

Análisis de la situación actual (3.1: Análisis de la situación actual), correspondiente a la fase 1 del modelo metodológico.

En esta etapa, las decisiones se toman con base en la realidad que está viviendo el equipo de Integración en cuanto a su proceso de desarrollo de software y tiene como objetivo final mejorar el mismo.

A continuación, se describen las actividades de la formulación del proceso de desarrollo de software.

3.2.1. Definir los artefactos del proceso de desarrollo

El propósito de esta actividad es definir los artefactos del proceso de desarrollo, los cuales son elementos complementarios con el fin de promover la transparencia, la inspección y la adaptación.

El instrumento empleado para llevar a cabo esta tarea se expone en la sección 3.7.4: Tabla de artefactos del proceso de desarrollo.

3.2.2. Definir los estados de un “User Story”

Esta actividad define los estados que puede tener un “User Story” dentro del proceso de desarrollo. El conjunto de estados definidos establecerán el flujo de ejecución que seguirán todos los “User Stories”.

El instrumento utilizado en esta tarea se expone en la sección 3.7.5: Tabla de estados de un “User Story”.

3.2.3. Definir los roles del proceso de desarrollo

Esta actividad tiene como objetivo definir los roles necesarios para en el proceso de desarrollo, así como las responsabilidades de cada uno. El instrumento utilizado se presenta en la sección 3.7.6: Tabla de roles del proceso de desarrollo.

3.2.4. Definir los eventos del proceso de desarrollo

El objetivo de esta actividad es definir los eventos que forman parte del proceso de desarrollo. Estos eventos son reuniones con un propósito: enriquecer y mejorar de manera continua el proceso de desarrollo.

La herramienta utilizada para llevar a cabo esta tarea se detalla en la sección 3.7.7: Tabla de eventos del proceso de desarrollo.

3.3. Evaluación de herramientas

El objetivo de esta etapa es en primera instancia evaluar la herramienta utilizada por el equipo de Integración para soportar su proceso de desarrollo (“Redmine”), seguidamente, identificar las características que busca el equipo de Integración en una herramienta, y finalmente, seleccionar un conjunto de herramientas que soporten procesos de desarrollo basados en el marco de trabajo “Scrum” y compararlas. A continuación, se describen las actividades de esta etapa.

3.3.1. Evaluar la herramienta “Redmine”

“Redmine” es la herramienta de software utilizada por el equipo de Integración para soportar su proceso de desarrollo. El propósito de esta actividad es evaluar en cuál nivel

“Redmine” está cumpliendo las necesidades del proceso que está soportando, qué tan adaptable y flexible es, además de plantear la interrogante de que si es posible cambiarla si fuese necesario.

Para llevar a cabo esta tarea, se utiliza la entrevista como técnica de selección de datos, misma que se define en la sección 3.6.2: Entrevista. El instrumento utilizado se expone en la sección 3.7.8: Entrevista para evaluar la herramienta “Redmine”.

3.3.2. Definir las características deseables en la herramienta

Esta actividad tiene como finalidad definir e identificar las características deseables en la herramienta que utilizaría el equipo de Integración para soportar el proceso de desarrollo de software.

La técnica de recolección de datos empleada es la entrevista, misma que se define en la sección 3.6.2: Entrevista. El instrumento utilizado se expone en la sección 3.7.9: Entrevista para determinar las características deseables en la herramienta.

3.3.3. Seleccionar y evaluar herramientas

Esta actividad consiste en revisar documentos electrónicos, con el propósito de encontrar herramientas que ofrece el mercado para soportar procesos de desarrollo de software basados en “Scrum”. Las herramientas seleccionadas deben cumplir con algunas de las características que se determinen en la actividad 3.3.2: Definir las características deseables en la herramienta.

La técnica de recolección de datos utilizada para esta tarea es la revisión de documentos, misma que se define en la sección 3.6.3. Revisión de documentos. El instrumento utilizado para llevar a cabo esta actividad se expone en la sección 3.7.10: Tabla de herramientas seleccionadas.

3.4. Simulación del modelo de desarrollo de software

(Shannon, 1998) define la simulación como el proceso de diseñar un modelo para un sistema real, y conducir experimentos y/o mediciones para comprender su comportamiento.

Esta etapa tiene como finalidad aplicar un modelo de simulación sobre el modelo de software propuesto, incluye el proceso de desarrollo y la herramienta seleccionada, a fin de poder determinar y medir cómo ese modelo trabajaba en el equipo de Integración.

A continuación, se describen las actividades por realizar en esta etapa.

3.4.1. Crear el modelo de simulación

Esta actividad consiste en elaborar el modelo de simulación, el cual describe todos los aspectos necesarios para simular el modelo de software planteado para el equipo de Integración.

El instrumento utilizado para llevar a cabo esta actividad se presenta en la sección 3.7.11: Modelo de simulación.

3.4.2. Realizar la simulación

Esta actividad tiene como finalidad ejecutar el modelo de simulación definido en la sección 3.4.1: Crear el modelo de simulación y documentar los hallazgos que ocurran durante

este proceso. El instrumento utilizado para llevar a cabo esta actividad se expone en la sección 3.7.12: Bitácora de la simulación del modelo de desarrollo.

3.5. Recomendaciones para implementar el modelo de desarrollo de software

El objetivo de esta etapa es definir un conjunto de recomendaciones que permitan aumentar la probabilidad de éxito del modelo de desarrollo de software propuesto, durante la fase de implementación. Las recomendaciones que se establezcan, van a estar basadas en las características y necesidades del equipo de Integración, información obtenida de la fase 3.1: Análisis de la situación actual.

3.6. Técnicas de recolección de información

A continuación, se presenta la definición de las técnicas utilizadas en la investigación para recolectar la información requerida.

3.6.1. Cuestionario

(Bernal, 2010) define el cuestionario como *“conjunto de preguntas organizadas sobre un tema de investigación y que se utiliza para obtener información relacionada con los objetivos en la investigación”*.

3.6.2. Entrevista

La entrevista es definida por (Hernández, Fernández, & Baptista, 2010) como una reunión para conversar e intercambiar información entre una persona que es el entrevistador

y otra que es el entrevistado. Además, menciona que la entrevista cualitativa es más flexible y abierta. (Bernal, 2010) sugiere que la entrevista puede estar soportada por un cuestionario muy flexible, pretende obtener información espontánea y abierta.

3.6.3. Revisión de documentos

(Hernández, Fernández, & Baptista, 2010) consideran que los documentos son una fuente muy valiosa de datos cualitativos, debido a que pueden ayudar a entender el fenómeno central de estudio.

3.7. Instrumentos utilizados en la investigación

Esta sección presenta los instrumentos utilizados para llevar a cabo la investigación, así como una descripción de los mismos.

3.7.1. Cuestionario para caracterizar el proceso de desarrollo

actual

El cuestionario (Tabla 1: Cuestionario para caracterizar el proceso de desarrollo actual) utilizado para reconocer las características del proceso de desarrollo actual se basó en (Ribeiro, 2015). El mismo contiene algunas preguntas de carácter general, y otras por su parte, son para determinar si el proceso de desarrollo actual presenta características y/o elementos semejantes a los de “Scrum”.

Evaluación del proceso de desarrollo en el equipo de Integración

1. ¿Cuál es su rol dentro del equipo de trabajo?

- Desarrollador
- “Product Owner”
- Líder Técnico
- “ScrumMaster”
- Administrador del Proyecto
- Ingeniero de Calidad

2. ¿Cuál metodología o marco de trabajo utiliza el equipo de Integración?

- “Scrum”
- “Kanban”
- Otra
- Ninguna

3. ¿Considera que el equipo de Integración es auto-organizado? Es decir, que no existe una figura que está continuamente diciendo qué se debe hacer.

- Sí
- No

4. ¿Existe alguna persona dentro del equipo de Integración que define qué se debe trabajar y su prioridad?

- Sí
- No

5. ¿Cuenta el equipo de Integración con una lista de tiquetes previamente establecidos y priorizados en los cuales debe trabajar?

- Sí

No

6. ¿Qué tan frecuentes son los pases a producción en los sistemas en los cuales labora el equipo de Integración?

Semanal

Quincenal

Mensual

Semestral

7. ¿Qué tan frecuente los tiquetes en los que usted trabaja cuentan con toda la información necesaria para poder ser implementados?

Siempre

Casi siempre

Algunas veces

Casi nunca

Nunca

8. ¿Tiene el equipo de Integración una definición establecida de lo que significa que un tiquete está listo?

Sí

No

9. ¿Encuentran las personas del equipo de Integración bloqueos que le impiden avanzar con el trabajo asignado?

Siempre

Casi siempre

Algunas veces

Casi nunca Nunca

10. ¿ Los bloqueos son abordados a la mayor brevedad posible?

 Siempre Casi siempre Algunas veces Casi nunca Nunca

11. ¿Considera que el equipo de Integración colabora diariamente por medio de algún tipo de reunión de estatus?

 Sí No

12. ¿Considera que el equipo de Integración se encuentra en mejora continua?

 Siempre Casi siempre Algunas veces Casi nunca Nunca

13. ¿Utiliza el equipo de Integración algún tipo de métrica para determinar cuál es su desempeño?

 Sí No

14. ¿Realiza el equipo de Integración algún tipo de planeación sobre el trabajo por realizar con cierta regularidad?

- Siempre
- Casi siempre
- Algunas veces
- Casi nunca
- Nunca

15. ¿Tiene usted alguna idea de qué se podría mejorar en el proceso de desarrollo que emplea el equipo de Integración?

Experiencia con “Scrum”

1. ¿Tiene usted alguna experiencia previa en equipos que han utilizado “Scrum”?

- Sí
- No

2. ¿Cuenta usted con la certificación de “ScrumMaster”?

- Sí
- No

Tabla 1: Cuestionario para caracterizar el proceso de desarrollo actual

Fuente: Elaboración propia, basado en (Ribeiro, 2015)

3.7.2. Entrevista para determinar los estados de un ticket

Esta entrevista es utilizada para identificar cuáles son los estados por los cuales pasan los tickets en el equipo de integración dentro del flujo normal de trabajo, además, establecer si algunos de estos estados pueden ser modificados o eliminados. La entrevista se basó en las preguntas que presenta la siguiente tabla.

Pregunta	Respuesta
¿Podría describir los estados por los cuales pasan los tickets dentro del flujo normal de desarrollo, desde su comienzo hasta su finalización?	
¿Podrían algunos de estos estados ser eliminados o modificados?	

Tabla 2: Entrevista para determinar los posibles estados de un ticket

Fuente: Elaboración Propia

3.7.3. Tabla de documentos recopilados

Este instrumento (Tabla 3: Documentos recopilados de Dino) es utilizado para listar y describir los documentos recopilados, que están relacionados con el proceso de desarrollo del equipo de Integración.

Nombre del documento	Tipo	Descripción
Documento 1		
Documento 2		
Documento 3		

Nombre del documento	Tipo	Descripción
Documento n		

Tabla 3: Documentos recopilados de Dino

Fuente: Elaboración Propia

3.7.4. Tabla de artefactos del proceso de desarrollo

La tabla de artefactos del proceso de desarrollo (Tabla 4: Tabla de artefactos del proceso de desarrollo) lista y detalla cada artefacto del proceso de desarrollo. A su vez, incluye una serie de recomendaciones por considerar para cada artefacto.

Nombre del artefacto	Descripción	Recomendaciones
Artefacto 1		
Artefacto 2		
Artefacto n		

Tabla 4: Tabla de artefactos del proceso de desarrollo

Fuente: Elaboración Propia

3.7.5. Tabla de estados de un “User Story”

La tabla de estados de un “User Story” (Tabla 5: Tabla de estados de un “User Story”) es utilizada para listar y describir los posibles estados que pueden tener los “User Stories” dentro del proceso de desarrollo.

Nombre del estado	Descripción
Estado 1	

Nombre del estado	Descripción
Estado 2	
Estado n	

Tabla 5: Tabla de estados de un "User Story"

Fuente: Elaboración Propia

3.7.6. Tabla de roles del proceso de desarrollo

Esta tabla (Tabla 6: Tabla de roles del proceso de desarrollo) detalla los roles que son necesarios en el proceso de desarrollo propuesto, así como las responsabilidades de cada uno.

Nombre del Rol	Responsabilidades
Rol 1	
Rol 2	
Rol n	

Tabla 6: Tabla de roles del proceso de desarrollo

Fuente: Elaboración Propia

3.7.7. Tabla de eventos del proceso de desarrollo

Esta tabla (Tabla 7: Tabla de eventos del proceso de desarrollo) es utilizada para definir los eventos del proceso de desarrollo, incluye el nombre del evento, una descripción, los distintos roles que deben atender cada evento, así como una serie de recomendaciones por considerar durante la ocurrencia de cada una de estas actividades.

Nombre del evento	Descripción	Asistentes	Recomendaciones
Evento 1			
Evento 2			
Evento n			

Tabla 7: Tabla de eventos del proceso de desarrollo

Fuente: Elaboración Propia

3.7.8. Entrevista para evaluar la herramienta “Redmine”

A continuación se presentan las preguntas que forman parte de la entrevista para evaluar la herramienta “Redmine”. El objetivo de esta herramienta es determinar en qué medida “Redmine” satisface las necesidades del equipo de Integración, si es posible cambiarla, y además, conocer qué tan flexible es, según el criterio del entrevistado.

Pregunta	Respuesta
¿Considera usted que “Redmine” le provee todas las características necesarias para soportar el proceso de desarrollo de software actual?	
¿Tiene el equipo de Integración la voluntad de cambiar “Redmine” si fuese necesario?	
¿Considera usted que tendría un efecto negativo sobre el equipo de Integración	

Pregunta	Respuesta
utilizar otra herramienta que no sea “Redmine”?	
¿Sabe usted si “Redmine” posee las características para soportar un proceso de desarrollo basado en “Scrum”?	

Tabla 8: Entrevista para evaluar la herramienta “Redmine”

Fuente: Elaboración Propia

3.7.9. Entrevista para determinar las características deseables en la herramienta

Esta entrevista (basada en la pregunta que presenta la siguiente tabla), tiene como objetivo plantear las características deseables por parte del equipo de Integración, en una herramienta que soporte procesos de desarrollo de software basado en “Scrum”.

Pregunta	Respuesta
¿Podría mencionar las características deseables en la herramienta que se utilizaría para soportar el proceso de desarrollo de software?	

Tabla 9: Entrevista para determinar las características deseables en la herramienta

Fuente: Elaboración Propia

3.7.10. Tabla de herramientas seleccionadas

Esta tabla (Tabla 10: Tabla de herramientas seleccionadas) es el instrumento utilizado para documentar las herramientas que han sido seleccionadas como posibles candidatas para soportar el proceso de desarrollo de software del equipo de Integración.

Nombre de la herramienta	Características
Herramienta 1	
Herramienta 2	
Herramienta n	

Tabla 10: Tabla de herramientas seleccionadas

Fuente: Elaboración Propia

3.7.11. Modelo de simulación

Este instrumento (Tabla 11: Modelo de simulación) define los aspectos requeridos para realizar la simulación del modelo de desarrollo de software propuesto.

Tamaño del equipo de trabajo	
Duración del Sprint	
“ Product Backlog ”	
“ Sprint Backlog ”	
“ Scrum Master ”	
“ Product Owner ”	

Tabla 11: Modelo de simulación

Fuente: Elaboración Propia

3.7.12. Bitácora de la simulación del modelo de desarrollo

Este instrumento (Tabla 12: Bitácora de la simulación del modelo de desarrollo) es una bitácora que va a permitir documentar los hallazgos que se den durante la ejecución de la simulación del modelo de software propuesto.

Fecha	Hallazgos

Tabla 12: Bitácora de la simulación del modelo de desarrollo

Fuente: Elaboración Propia

4. Análisis de resultados

(Bernal, 2010) define el análisis de resultados como la interpretación de los hallazgos relacionados con el problema de investigación, los objetivos propuestos, la hipótesis y/o las preguntas formuladas, a fin de evaluar si confirman las teorías o no.

El presente capítulo, mostrará los resultados obtenidos de la ejecución de las etapas que conformaron el modelo metodológico, así como de sus respectivas actividades.

4.1. Análisis de la situación actual

En esta sección se presentan los resultados obtenidos de las actividades que se llevaron a cabo para analizar la situación actual del proceso de desarrollo del equipo de Integración. Al final, se hace una síntesis para describir los hallazgos del análisis realizado.

4.1.1. Características del proceso de desarrollo actual

Para reconocer las características del proceso de desarrollo actual, se aplicó el cuestionario definido en la sección 3.7.1: Cuestionario para caracterizar el proceso de desarrollo actual.

El cuestionario fue aplicado por medio de la herramienta “Google Forms” a once integrantes del equipo de Integración. A todos ellos se les proporcionó el link por medio del cual podían completar el cuestionario. El administrador y líder técnico del equipo de Integración les solicitó que lo completaran, y para ello se les dio 30 minutos de tiempo.

A continuación, se presentan los resultados obtenidos en cada una de las preguntas que conformaron la primera sección del cuestionario.

1. ¿Cuál es su rol dentro del equipo de trabajo? (11 responses)

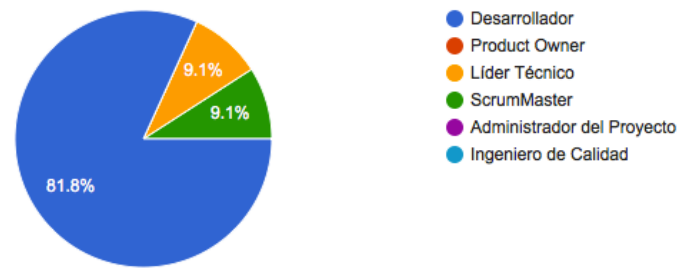


Figura 10: Gráfico de la pregunta #1 del cuestionario para conocer la situación actual

Fuente: Fuente: "Google Forms"

De la pregunta anterior, se puede observar que el 81.8% de las personas que completaron el cuestionario fueron desarrolladores, hubo un líder técnico y un "ScrumMaster".

Es importante recalcar en estos resultados que una persona dentro del equipo de Integración se considera "ScrumMaster", a pesar de que dicho rol aún no está establecido dentro de la compañía. Lo anterior podría deberse a que esta persona asocia las actividades que desempeña con dicho rol.

2. ¿Cuál metodología o marco de trabajo utiliza el equipo de Integración? (11 responses)

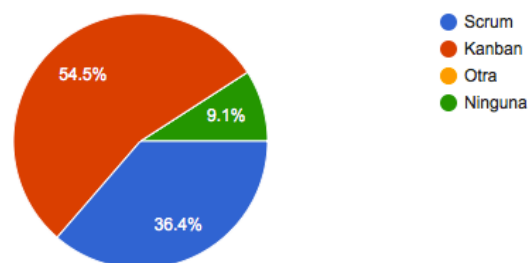


Figura 11: Gráfico de la pregunta #2 del cuestionario para conocer la situación actual

Fuente: "Google Forms"

En la pregunta anterior, se puede observar que el 54.5% de las personas que completaron el cuestionario piensan que el proceso de desarrollo actual se basa en “Kanban”, el 36.4% creen que es “Scrum”, y el 9.1% por su parte no cree que se utilice ningún marco de trabajo o metodología particular.

3. ¿Considera que el equipo de Integración es auto-organizado? Es decir, que no existe una figura que está continuamente diciendo que se debe hacer.
(11 responses)

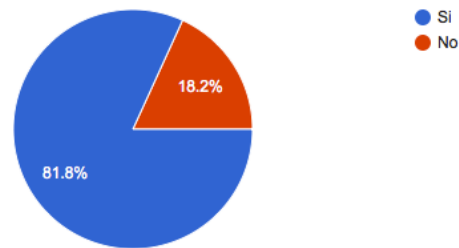


Figura 12: Gráfico de la pregunta #3 del cuestionario para conocer la situación actual

Fuente: "Google Forms"

En la pregunta anterior, se puede observar que el 81.8% de las personas que completaron el cuestionario creen que el equipo de integración es auto-organizado, el 18.2% por su parte, no lo considera así.

4. ¿Existe alguna persona dentro del equipo de Integración que define que se debe trabajar y su prioridad?
(11 responses)

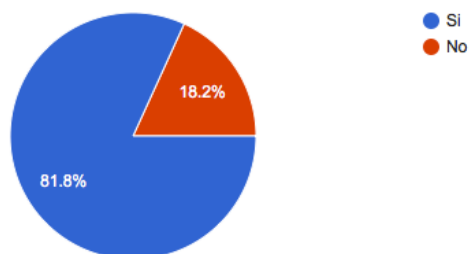


Figura 13: Gráfico de la pregunta #4 del cuestionario para conocer la situación actual

Fuente: "Google Forms"

De los resultados obtenidos en la pregunta anterior, se puede observar que el 81.8% de las personas consultadas consideran que existe una persona dentro del equipo de Integración que define el trabajo, así como su prioridad, el 18.2% por su parte, considera que nadie realiza ese trabajo.

5. ¿Cuenta el equipo de Integración con una lista de tiquetes previamente establecidos y priorizados en los cuales debe trabajar?

(11 responses)

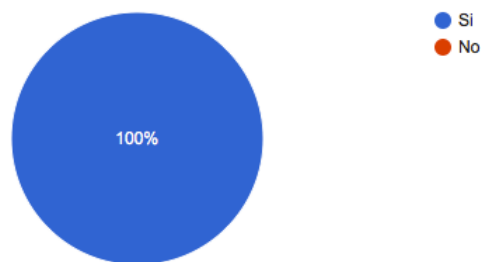


Figura 14: Gráfico de la pregunta #5 del cuestionario para conocer la situación actual

Fuente: "Google Forms"

En la pregunta anterior, se puede observar que el 100% de las personas consultadas consideran que existe una lista de tiquetes previamente establecidos y priorizados.

6. ¿Qué tan frecuentes son los pases a producción en los sistemas en los cuales labora el equipo de Integración?

(11 responses)

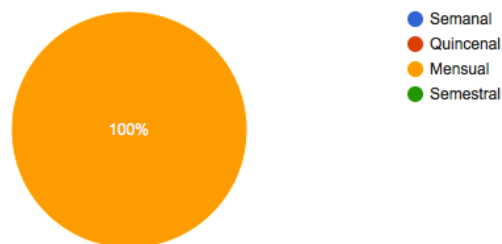


Figura 15: Gráfico de la pregunta #6 del cuestionario para conocer la situación actual

Fuente: "Google Forms"

Los resultados obtenidos en la pregunta anterior, reflejan que el 100% de la muestra que contestó el cuestionario considera que los pases a producción en los sistemas que trabaja el equipo de Integración se realizan de manera mensual.

7. ¿Qué tan frecuente los tiquetes en los que usted trabaja cuentan con toda la información necesaria para poder ser implementados?

(11 responses)

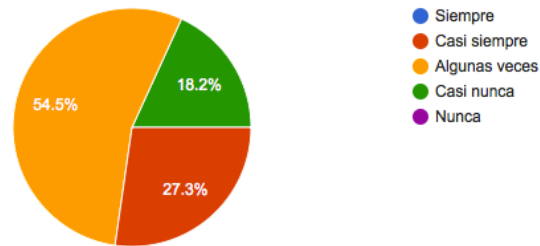


Figura 16: Gráfico de la pregunta #7 del cuestionario para conocer la situación actual

Fuente: "Google Forms"

En la pregunta anterior, es posible observar que el 54.5% de las personas consultadas consideran que algunas veces los tiquetes contienen toda la información requerida para poder ser implementados. El 18.2% considera que esto casi nunca ocurre, y el 27.3% considera que casi siempre ocurre.

8. ¿Tiene el equipo de Integración una definición establecida de lo que significa que un tiquete está listo?

(11 responses)

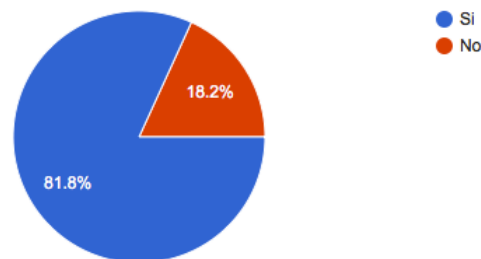


Figura 17: Gráfico de la pregunta #8 del cuestionario para conocer la situación actual

Fuente: "Google Forms"

En la pregunta anterior, el 81.8% de las personas consideran tener una definición clara de lo que significa que un tiquete está listo, el 18.2% por su parte no lo considera así.

9. ¿Encuentran las personas del equipo de Integración bloqueos que le impiden continuar avanzando con el trabajo asignado?

(11 responses)

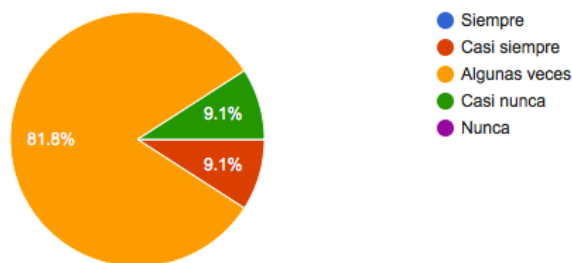


Figura 18: Gráfico de la pregunta #9 del cuestionario para conocer la situación actual

Fuente: "Google Forms"

De la pregunta anterior se desprende que el 81.8% de las personas consideran que algunas veces encuentran bloqueos que les impiden continuar avanzado sobre el trabajo asignado, el 9.1% considera que esto casi siempre ocurre, y el 9.1% por su parte, cree que esto casi nunca pasa.

10. ¿Son los bloqueos abordados a la mayor brevedad posible? (11 responses)

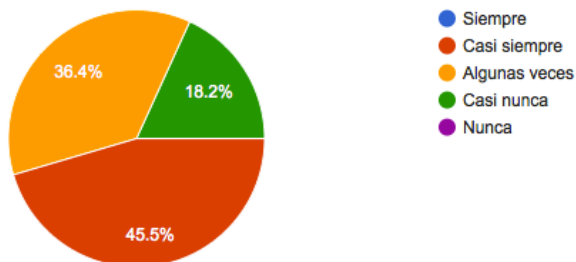


Figura 19: Gráfico de la pregunta #10 del cuestionario para conocer la situación actual

Fuente: "Google Forms"

De la pregunta anterior se desprende que el 45.5% de las personas consultadas consideran que casi siempre los bloqueos son abordados con la mayor brevedad posible, el 36.4% considera que esto ocurre algunas veces, y el 18.2% por su parte, piensa que esto casi nunca se da.

11. ¿Considera que el equipo de Integración colabora diariamente por medio de algún tipo de reunión de estatus?

(11 responses)

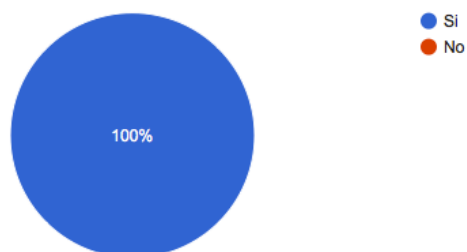


Figura 20: Gráfico de la pregunta #11 del cuestionario para conocer la situación actual

Fuente: "Google Forms"

En la pregunta anterior, el 100% de los consultados consideraron que se colabora diariamente con una reunión de estatus en el equipo de Integración.

12. ¿Considera que el equipo de Integración se encuentra en mejora continua?

(11 responses)

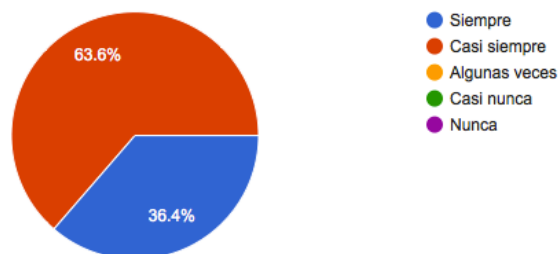


Figura 21: Gráfico de la pregunta #12 del cuestionario para conocer la situación actual

Fuente: "Google Forms"

En las respuestas obtenidas en la pregunta anterior, se puede observar que el 63.6% de los consultados consideran que el equipo de Integración casi siempre se encuentra en mejora continua, el 36.4% por su parte, considera que esto siempre ocurre.

13. ¿Utiliza el equipo de Integración algún tipo de métrica para determinar cuál su desempeño?

(11 responses)

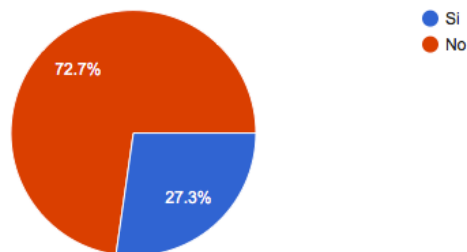


Figura 22: Gráfico de la pregunta #13 del cuestionario para conocer la situación actual

Fuente: "Google Forms"

En la pregunta anterior, el 72.7% de las personas consultadas considera que el equipo de Integración no tiene ningún tipo de métrica para medir su desempeño, el 27.3% considera por su parte, que sí se cuenta con alguna métrica para determinar esto.

14. ¿Realiza el equipo de Integración algún tipo de planeación sobre el trabajo a realizar con cierta regularidad?

(11 responses)

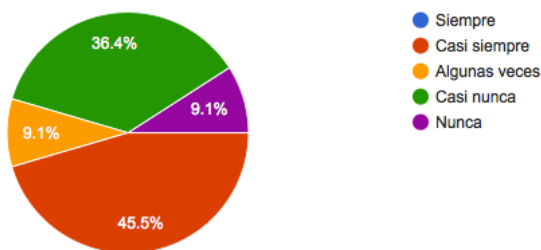


Figura 23: Gráfico de la pregunta #14 del cuestionario para conocer la situación actual

Fuente: "Google Forms"

De la pregunta anterior, se desprende que el 45.5% de los consultados considera que casi siempre se realiza algún tipo de planeación sobre el trabajo por realizar, el 36.4% considera que esto casi nunca ocurre. El 9.1% por su parte, cree que esto nunca se da, y finalmente, el 9.1% restante considera que algunas veces se da.

A continuación, se presentan los resultados de la segunda sección del cuestionario, éste pretendía conocer la experiencia de las personas que respondieron el cuestionario en “Scrum”

1. ¿Tiene usted alguna experiencia previa en equipos que han utilizado Scrum?
(11 responses)

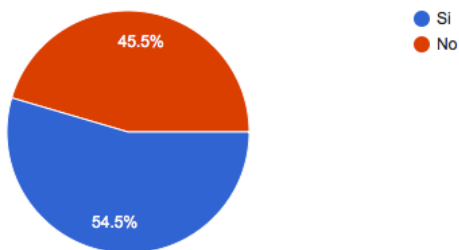


Figura 24: Gráfico de la pregunta #15 del cuestionario para conocer la situación actual

Fuente: "Google Forms"

De la pregunta anterior se extrae que el 54.5% de los consultados tiene alguna experiencia previa en equipos de trabajo que han utilizado “Scrum”, el 45.5% por su parte, no tiene experiencia alguna en “Scrum”.

2. ¿Cuenta usted con la certificación de ScrumMaster? (11 responses)

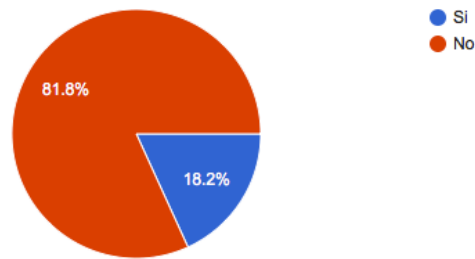


Figura 25: Gráfico de la pregunta #16 del cuestionario para conocer la situación actual

Fuente: Google Forms

En la pregunta anterior, el 81.8% de los consultados dijeron no tener la certificación como “ScrumMaster”, el 18.2% por su parte, si cuenta con dicha certificación.

El cuestionario también incluyó una pregunta abierta, en la cual se les consultó a los participantes sobre ideas para mejorar el proceso de desarrollo actual del equipo de Integración. Las respuestas obtenidas se encuentran en el capítulo de anexos (6.1: Ideas para mejorar el proceso de desarrollo actual). A continuación, se presentan las ideas propuestas:

- Implementar métricas de desempeño que permitan realizar acciones de mejoramiento y optimización tanto a nivel de equipo como a nivel individual
- Definir procesos sobre el accionar cuando hay tiquetes bloqueados
- Realizar monitoreo sobre los procesos establecidos
- Determinar áreas de mejora de manera continua
- Realizar reuniones para evaluar el rendimiento del equipo

4.1.2. Estados de un ticket

Para conocer los estados de los tiquetes dentro del proceso de desarrollo del equipo de Integración, se realizó una entrevista con dos líderes técnicos de dicho equipo. Esta

entrevista se llevó a cabo por medio de una llamada realizada a través de “Google Hangouts”. La información recolectada en dicha entrevista se encuentra en el capítulo de anexos (6.2: Entrevista para determinar los estados de un ticket).

El diagrama que se presenta a continuación (Figura 26: Diagrama de estados de un ticket), muestra los estados y el flujo que siguen los tickets durante el proceso de desarrollo.

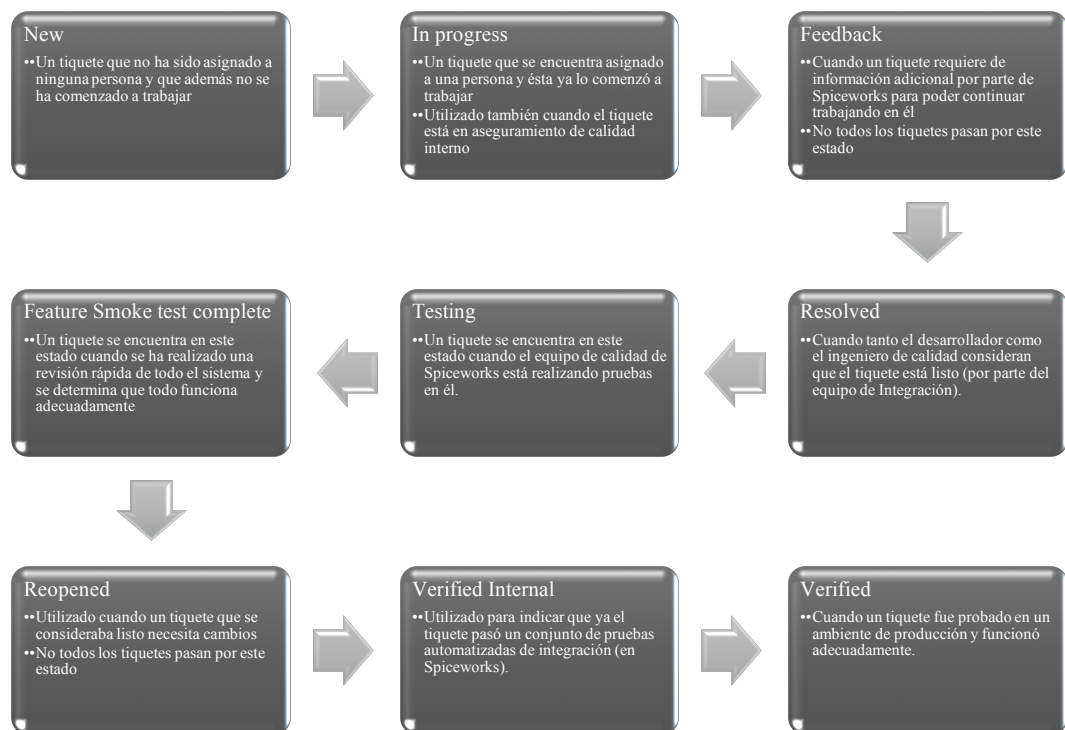


Figura 26: Diagrama de estados de un ticket

Fuente: Elaboración Propia basado en 6.2: Entrevista para determinar los estados de un ticket

Se consultó a los entrevistados si es posible cambiar o modificar los estados y el flujo que siguen los tickets, ellos respondieron que no, pues fueron establecidos por Spiceworks, y están configurados en la herramienta “Redmine”.

4.1.3. Revisión de documentos

La revisión de documentos se realizó sobre cuatro documentos que fueron proporcionados por el equipo de Integración. Los mismos se encuentran en el Capítulo de Anexos (6.3: Documentos revisados). La tabla que se presenta a continuación (Tabla 13: Documentos recopilados de Dino), describe los hallazgos presentes en estos documentos.

Nombre del documento	Tipo	Descripción
"05 06 2016 Status April 2016.xlsx"	Documento Excel	<p>Este documento corresponde a un reporte semanal enviado por el equipo de Integración a Spiceworks, para notificar el estado del trabajo que se tiene asignado. Los tiquetes, incluidos en este reporte, pertenecen a las características o mejoras que se incluirán en el siguiente pase a producción que se realice. El documento contiene la siguiente información:</p> <ul style="list-style-type: none"> ✓ Listado de tiquetes que aún no se han comenzado a trabajar ✓ Listado de tiquetes que se encuentran en progreso

Nombre del documento	Tipo	Descripción
		<ul style="list-style-type: none"> ✓ Listado de tiquetes que están en espera de retroalimentación, por lo cual están bloqueados ✓ Listado de tiquetes que fueron resueltos durante la semana
"Patch April and 7.6"	Correo electrónico	Este documento es un correo electrónico, donde el contacto principal por parte de Spiceworks, especifica cuáles son las prioridades sobre el trabajo que tiene asignado el equipo de Integración.
"Bug #72409"	Tiquete de "Redmine"	Este documento corresponde a un tiquete en "Redmine", en él se puede apreciar los detalles de dicho tiquete, y adicionalmente los estados por los cuales dicho tiquete pasó a lo largo de su ciclo de desarrollo.
"7.5 - Patch May"	Documento PDF	Este documento contiene el listado de tiquetes que fueron seleccionados para ser incluidos en la siguiente versión de los productos en los cuales trabaja el equipo de Integración. En dicho

Nombre del documento	Tipo	Descripción
		documento es posible observar que se tiene una prioridad establecida, además del estado correspondiente de cada tiquete.

Tabla 13: Documentos recopilados de Dino

Fuente: Elaboración Propia

4.1.4. Situación Actual del proceso de desarrollo del equipo de Integración

En el análisis de la información obtenida de las actividades llevadas a cabo para comprender la situación actual del proceso de desarrollo del equipo de Integración, es notable que existen fortalezas, y oportunidades de mejora. Además, sale a relucir que el proceso actual comparte algunas similitudes con el marco de trabajo “Scrum”.

Dentro de las fortalezas del proceso de desarrollo actual se encuentran:

- ✓ El equipo de trabajo se considera auto-organizado, lo cual significa que ninguna persona está continuamente indicando qué se debe hacer. Esto es substraído de la tercera pregunta del cuestionario para conocer la situación actual (Figura 12: Gráfico de la pregunta #3 del cuestionario para conocer la situación actual), donde el 81.8% de los encuestados afirmaron ser un equipo auto-organizado
- ✓ Existe una persona que establece de manera clara qué se debe trabajar, así como su prioridad. Esto es revelado en la cuarta pregunta del cuestionario para conocer la situación actual (Figura 13: Gráfico de la pregunta #4 del

cuestionario para conocer la situación actual), donde el 81.8% de los encuestados así lo afirmaron

- ✓ El equipo de Integración cuenta con una lista de tiquetes previamente establecidos y priorizados. Lo anterior es posible afirmarlo gracias a la quinta pregunta del cuestionario para conocer la situación actual (Figura 14: Gráfico de la pregunta #5 del cuestionario para conocer la situación actual), donde el 100% de los encuestados afirman tener dicha lista de tiquetes definida y priorizada
- ✓ Se tiene claramente establecida la periodicidad de los pases a producción. Esto se desprende de la sexta pregunta del cuestionario para conocer la situación actual (Figura 15: Gráfico de la pregunta #6 del cuestionario para conocer la situación actual), donde el 100% de los encuestados afirman que los pases a producción se realizan de manera mensual
- ✓ El equipo tiene una definición establecida de lo que significa que un tiquete está listo. Esto es revelado en la octava pregunta del cuestionario para conocer la situación actual (Figura 17: Gráfico de la pregunta #8 del cuestionario para conocer la situación actual), donde el 81.8% de los encuestados así lo confirmaron
- ✓ El hecho de que se den reuniones diarias para mantener al equipo informado sobre el estado del trabajo que se está realizando, demuestra la existencia de una buena comunicación dentro del mismo. Esto es posible deducirlo por medio de la onceava pregunta del cuestionario para conocer la situación actual (Figura 20: Gráfico de la pregunta #11 del cuestionario para conocer la

situación actual), donde el 100% de los encuestados afirman tener reuniones diarias con el equipo

- ✓ Los estados y el flujo que deben seguir los tiquetes está establecido de manera clara y contundente por parte de Spiceworks. Esto es posible afirmarlo por medio de la información descrita en la sección 4.1.2: Estados de un tiquete, donde se presenta un diagrama que muestra dicho flujo y sus estados. Esto también se confirma en uno de los documentos revisados en la sección 4.1.3: Revisión de documentos, específicamente el tiquete de “Redmine”, donde se aprecia que el tiquete sigue los estados y el flujo descrito.

Las oportunidades de mejora identificadas son las siguientes:

- ✓ Establecer para el proceso de desarrollo, la base sobre la cual se fundamenta el mismo, resulta evidente que el equipo de Integración no tiene un consenso sobre qué tipo de proceso de desarrollo están utilizando.

Esto se desprende de la segunda pregunta del cuestionario para conocer la situación actual (Figura 11:Gráfico de la pregunta #2 del cuestionario para conocer la situación actual), donde el 54.5% de los encuestados utilizan “Kanban”, el 36.4%, Scrum, y finalmente, el 9.1% dice no utilizar ninguna metodología o marco de trabajo. El hecho de que un gran porcentaje de encuestados seleccionaron “Kanban” se puede deber a que el equipo de Integración en algún momento intentó utilizar dicho enfoque.

- ✓ Es necesario que la calidad de los tiquetes sea mejorada pues el equipo considera en mayor proporción que el detalle de los tiquetes no siempre incluye toda la información requerida para su implementación, esto probablemente influya en que continuamente se encuentren bloqueos. Lo

anterior es posible afirmarlo gracias a la información obtenida en la séptima pregunta del cuestionario para conocer la situación actual (Figura 16: Gráfico de la pregunta #7 del cuestionario para conocer la situación actual) donde:

- El 54.5% de los encuestados afirman que algunas veces se cuenta con toda la información necesaria
- El 27.3% manifiesta que casi siempre se cuenta con toda la información necesaria
- El 18.2% afirma que casi nunca se cuenta con la información necesaria
- Nadie consideró que siempre se cuenta con toda la información necesaria

Adicionalmente, el 81.8% de los encuestados afirman que algunas veces se encuentran bloqueos que impiden seguir avanzando con el trabajo asignado, esto sale a relucir de la información obtenida en la novena pregunta del cuestionario para conocer la situación actual (Figura 18: Gráfico de la pregunta #9 del cuestionario para conocer la situación actual)

- ✓ Establecer métricas que permitan identificar el nivel de desempeño tanto individual como grupal. La información obtenida de la pregunta décimo tercera del cuestionario para conocer la situación actual (Figura 22: Gráfico de la pregunta #13 del cuestionario para conocer la situación actual) evidencia que el 72.7% de los encuestados no considera que el equipo de Integración utiliza alguna métrica para determinar su desempeño
- ✓ Establecer algún mecanismo para planificar el trabajo por realizar, debido a que en la actualidad se hace evidente que dicha planeación no se da de manera regular. Lo anterior es posible afirmarlo gracias a la información obtenida en

la pregunta décimo cuarta del cuestionario para conocer la situación actual (Figura 23: Gráfico de la pregunta #14 del cuestionario para conocer la situación actual), donde:

- El 9.1% considera que dicha planeación nunca se da
 - El 9.1% considera que algunas veces se realiza planeación
 - El 36.4% considera que casi nunca se realiza planeación
 - El 45.5% afirma que casi siempre se realiza planeación
 - Nadie afirmó que siempre se realice planeación
- ✓ Es necesario establecer un mecanismo que permita atender los bloqueos de manera más oportuna. La información obtenida de la décima pregunta del cuestionario para conocer la situación actual (Figura 19: Gráfico de la pregunta #10 del cuestionario para conocer la situación actual) muestra que:
- El 45.5% de los encuestados afirman que casi siempre los bloqueos son atendidos con la mayor brevedad
 - El 36.4% considera que los bloqueos algunas veces son atendidos con la mayor brevedad posible
 - El 18.2% de los encuestados afirman que casi nunca los bloqueos son atendidos con la mayor brevedad posible

Las características o elementos de “Scrum” que se asemejan al proceso de desarrollo actual son las siguientes:

- ✓ Existe una persona que establece cuál es el trabajo que se debe realizar, así como su prioridad, esto se asemeja a la labor del “Product Owner”. Lo anterior es posible afirmarlo gracias a la información obtenida en la cuarta pregunta del cuestionario para conocer la situación actual (Figura 13: Gráfico de la

pregunta #4 del cuestionario para conocer la situación actual), donde el 81.8% de los encuestados afirman que existe una persona que realiza dicho trabajo. Adicionalmente, en uno de los documentos revisados en la sección 4.1.3: Revisión de documentos, específicamente el correo electrónico “Patch April and 7.6”, se aprecia que el contacto principal por parte de Spiceworks indica las prioridades sobre el trabajo asignado.

- ✓ Los pases a producción se realizan siempre de manera mensual, características que se asemeja a los “Sprints”, esto porque son iteraciones que tienen siempre la misma duración. Esto fue revelado en la sexta pregunta del cuestionario para conocer la situación actual (Figura 15: Gráfico de la pregunta #6 del cuestionario para conocer la situación actual), donde el 100% de los encuestados afirmaron que los pases a producción son mensuales.

En el documento revisado en la sección 4.1.3: Revisión de documentos, específicamente el reporte “05 06 2016 Status April 2016.xlsx”, se puede apreciar una lista de tiquetes que brinda el estado del trabajo asignado para el siguiente pase a producción.

- ✓ El equipo se considera auto-organizado. Esto se desprende de la tercera pregunta del cuestionario para conocer la situación actual (Figura 12: Gráfico de la pregunta #3 del cuestionario para conocer la situación actual), donde el 81.8% de los encuestados afirmaron que el equipo de Integración es auto-organizado
- ✓ Se cuenta con una lista de tiquetes previamente establecida y priorizada, característica que se asemeja al “Product Backlog”. Lo anterior se deduce de la quinta pregunta del cuestionario para conocer la situación actual (Figura

14: Gráfico de la pregunta #5 del cuestionario para conocer la situación actual), donde el 100% de los encuestados afirmaron contar con dicho insumo. Esto además se aprecia en uno de los documentos revisados en la sección 4.1.3: Revisión de documentos, específicamente en el documento “7.5 - Patch May”, en donde se presenta una lista de tiquetes previamente establecida y priorizada.

- ✓ Se realizan reuniones diarias para brindar el estado del trabajo que se está desarrollando, característica que se asemeja al “Daily Scrum”. Esta información se obtiene a partir de la onceava pregunta del cuestionario para conocer la situación actual (Figura 20: Gráfico de la pregunta #11 del cuestionario para conocer la situación actual), donde el 100% de los encuestados afirmaron tener dicha reunión
- ✓ El equipo de Integración pareciera ser multifuncional, debido a que está conformado por personas con distintos perfiles profesionales, lo anterior se extrae de la primera pregunta del cuestionario para conocer la situación (Figura 10: Gráfico de la pregunta #1 del cuestionario para conocer la situación actual), donde el 81.8% de los encuestados dicen ser desarrolladores, el 9.1% dice ser líder técnico y el 9.1% ScrumMaster

Es importante recalcar además, que más de la mitad de las personas que conforman el equipo de Integración tienen experiencia previa con “Scrum”, y existen dos personas que cuentan con la certificación de “ScrumMaster”. Esto se desprende de la segunda parte del cuestionario para conocer la situación actual, específicamente en la primera (Figura 24: Gráfico de la pregunta #15 del cuestionario para conocer la situación actual) y segunda

pregunta (Figura 25: Gráfico de la pregunta #16 del cuestionario para conocer la situación actual), donde el 54.5% de los encuestados afirmaron tener experiencia previa con “Scrum”.

Equipo de Integración en el marco de trabajo “Cynefin”

El marco de trabajo “Cynefin” permite identificar por medio de distintos dominios, cuando es más apropiado utilizar “Scrum”. A continuación, se presenta un análisis de cada uno de los dominios que establece este marco de trabajo, desde la perspectiva del equipo de Integración, para finalmente establecer si “Scrum” resulta una opción adecuada.

- **Dominio complicado:** se está en este dominio cuando el trabajo que se realiza requiere de expertos y altos niveles de especialización, además, no es posible predecir ni el esfuerzo, ni los riesgos requeridos para resolver los problemas.

El equipo de Integración está conformado, en su gran mayoría, por programadores (información revelada en la Figura 10: Gráfico de la pregunta #1 del cuestionario para conocer la situación actual), quienes sin bien son profesionales en las áreas de la programación, no son expertos. Por otro lado, el equipo de Integración realiza cierta planeación sobre el trabajo que realiza (esto lo muestra la Figura 23: Gráfico de la pregunta #14 del cuestionario para conocer la situación actual), lo cual significa que las características del trabajo sí permiten estimar esfuerzo y riesgos.

Debido a lo anterior, se puede afirmar que el equipo de Integración no pertenece al dominio complicado.

- **Dominio simple:** en este dominio se requiere de experiencia mínima, la solución a los problemas es obvia y ya se encuentra documentada.

En el caso del equipo de Integración, de manera frecuente se enfrentan a bloqueos que impiden avanzar con el trabajo asignado (esto se muestra en Figura 18: Gráfico de la pregunta #9 del cuestionario para conocer la situación actual). Adicionalmente, no siempre se cuenta con toda la información necesaria para avanzar con el trabajo (esto lo muestra la Figura 16: Gráfico de la pregunta #7 del cuestionario para conocer la situación actual).

Lo anterior muestra que el dominio simple no se ajusta a las características del equipo de Integración, debido a que las soluciones no son simples ni están documentadas.

- **Dominio caótico:** se está en este dominio cuando es necesario resolver problemas con prontitud, y no necesariamente con la mejor solución.

Los productos sobre los cuales trabaja el equipo de Integración realizan sus pases a producción de manera mensual (esto lo muestra la Figura 15: Gráfico de la pregunta #6 del cuestionario para conocer la situación actual), lo cual significa que los mismos no responden a situaciones donde es necesario actuar de manera urgente para resolver problemas.

Lo anterior evidencia que las características del equipo de Integración no se ajustan al dominio caótico.

- **Dominio complejo:** en este dominio se requiere de inspección y adaptación, además, la comunicación es esencial. El trabajo realizado es evolutivo, no revolucionario. El desarrollo de productos innovadores caen dentro de esta categoría.

El equipo de Integración considera que se encuentra en mejora continua (esto lo revela la Figura 21: Gráfico de la pregunta #12 del cuestionario para conocer la

situación actual). Para lograr lo anterior, es necesario inspeccionar de manera periódica el trabajo que se realiza e incrementar ajustes (inspección y adaptación). En cuanto a la comunicación, pareciera que este es un hito muy importante en el equipo de Integración, debido a que sostienen reuniones diarias para mantener a todo el equipo informado (esto lo muestra la Figura 20: Gráfico de la pregunta #11 del cuestionario para conocer la situación actual).

El hecho de que se realizan pases a producción de manera mensual (esto lo muestra la Figura 15: Gráfico de la pregunta #6 del cuestionario para conocer la situación actual) evidencia el que trabajo realizado por el equipo de Integración es evolutivo, pues en dichos pases a producción se agregan características y mejoras a los productos sobre los cuales trabaja el equipo de Integración.

El análisis antes expuesto demuestra que las características del equipo de Integración se ajustan al dominio complejo.

- **Dominio desordenado:** se está en este dominio, cuando no se encaja en ninguno de los dominios antes expuestos. El equipo de Integración encaja en el dominio complejo, por lo cual no pertenece a este dominio.

Del análisis antes expuesto se puede deducir que las características del equipo de Integración se ajustan al dominio complejo del marco de trabajo “Cynefin”, esto permite que “Scrum” sea una opción adecuada para gestionar su trabajo.

4.2. Formulación el proceso de desarrollo de software

El proceso de desarrollo de software que se diseñó para el equipo de Integración, está basado en el marco de trabajo “Scrum” por tres razones fundamentales:

- ✓ Al analizar las características del trabajo realizado por el equipo de Integración, se determinó que las mismas se ajustan al dominio “Complejo” del marco de trabajo “Cynefin”, esto hace a “Scrum” una opción adecuada para soportar el proceso de desarrollo
- ✓ El equipo de Integración presenta similitudes con elementos de “Scrum”, tal y como se explica en la sección 4.1.4: Situación Actual del proceso de desarrollo del equipo de Integración
- ✓ Los ejecutivos de la compañía habían considerado “Scrum” como una opción viable. A continuación, se describen los elementos que componen el proceso de desarrollo.

4.2.1. Artefactos del proceso de desarrollo

Los artefactos del proceso de desarrollo son insumos del proceso que hacen posible su funcionamiento. En la tabla que se expone a continuación (Tabla 14: Tabla de artefactos del proceso de desarrollo), se describe cada uno de los artefactos que componen el proceso de desarrollo, además, se listan una serie de recomendaciones para cada artefacto.

El equipo de Integración cuenta con algunos insumos que tienen similitudes con algunos de los artefactos descritos a continuación, esto haría más sencillo su adopción y utilización.

Nombre del artefacto	Descripción	Recomendaciones
“Sprint”	Es la manera en la que “Scrum” organiza el trabajo, por medio de ciclos secuenciales.	<ul style="list-style-type: none"> • El tamaño máximo de cada “Sprint” no debe ser mayor a un mes calendario • Los “Sprints” deben tener todos los mismos tamaños • El equipo de Integración simula “Sprints” mensuales, se recomienda seguir utilizando dicho esquema
“User Story”	Formato utilizado para escribir los requerimientos de un sistema.	<ul style="list-style-type: none"> • Debe ser escrito de manera que pueda ser entendido por cualquier persona • Debe especificar el usuario que se verá beneficiado • Debe especificar el objetivo y su beneficio • Debe contener una sección donde se confirme lo que se desea construir o mejorar • La descripción de los tickets en “Redmine” podría ser

Nombre del artefacto	Descripción	Recomendaciones
		mejorada para utilizar el formato propuesto.
“Product Backlog”	Lista de requerimientos que se desean implementar en el sistema.	<ul style="list-style-type: none"> • El “Product Owner” debería estar constantemente alimentando y mejorando el “Product Backlog”. • No es necesario que todos los ítems se encuentren al mismo nivel de detalle. • Los ítems con más alta prioridad deben contener más detalles que aquellos con menor prioridad • El equipo de Integración ya cuenta con una lista de tickets por implementar, lo que se asemeja a un “Product Backlog”. Se podría trabajar más sobre esta lista para que sea alimentada de manera

Nombre del artefacto	Descripción	Recomendaciones
		<p>más frecuente y con mayor calidad en el nivel de detalle de lo que se describe.</p>
<p>“Sprint Backlog”</p>	<p>Lista de “User Stories” que son seleccionados durante el “Sprint Planning” para convertirlos en un potencial incremento del producto.</p>	<ul style="list-style-type: none"> • Los ítems que se encuentran en el “Sprint Backlog” deben ser desglosados en tareas, y para cada tarea se debería tener una estimación en horas. • En el equipo de Integración, existe una persona que define qué se debe trabajar para el siguiente pase a producción. Esto se asemeja al “Sprint Backlog”, con la discrepancia de que el equipo de trabajo también debería participar en la escogencia de los ítems. <p style="text-align: center;">Se recomienda involucrar al equipo de trabajo en la definición de</p>

Nombre del artefacto	Descripción	Recomendaciones
		esta lista, dado que finalmente sería el compromiso que se adquiriría para cada “Sprint”

Tabla 14: Tabla de artefactos del proceso de desarrollo

Fuente: Elaboración Propia

4.2.2. Estados de un “User Story”

La tabla que se muestra a continuación (Tabla 15: Estados de un “User Story”), presenta los posibles estados que puede tener un “User Story”. El conjunto de estados establecidos, define el flujo que seguirán los “User Stories” dentro del proceso de desarrollo.

Nombre del estado	Descripción
1. “New”	Un “User Story” que no ha sido asignado a ninguna persona y que además no se ha comenzado a trabajar. Lo anterior significa que se encuentra en el “Sprint Backlog”.
2. “In Progress”	Un “User Story” que se encuentra asignado a una persona y ésta ya lo comenzó a trabajar.
3. “Code completed”	Un “User Story” que ya fue finalizado y que está listo para la fase de pruebas.
4. “Internal Testing”	Un “User Story” que se encuentra en la fase de pruebas por parte del equipo de Integración.

Nombre del estado	Descripción
5. “Feedback”	Cuando un “User Story” requiere de información adicional por parte de Spiceworks para poder continuar trabajando en él. Lo anterior significa que el “User Story” se encuentra bloqueado.
6. Resolved	Un “User Story” que ya se encuentra listo, o sea, que ya fue aprobado por el ingeniero de calidad del equipo de Integración.
6. Reopened	Este estado sería utilizado cuando un “User Story” que se consideraba listo necesita cambios, o es necesario corregir comportamientos indeseados. Una vez realizados los cambios requeridos, el siguiente estado sería Resolved.

Tabla 15: Estados de un “User Story”

Fuente: Elaboración Propia

La figura que se presenta a continuación (Figura 27: Flujo de ejecución de los “User Stories”), expone el flujo de ejecución que seguirían los “User Stories”.

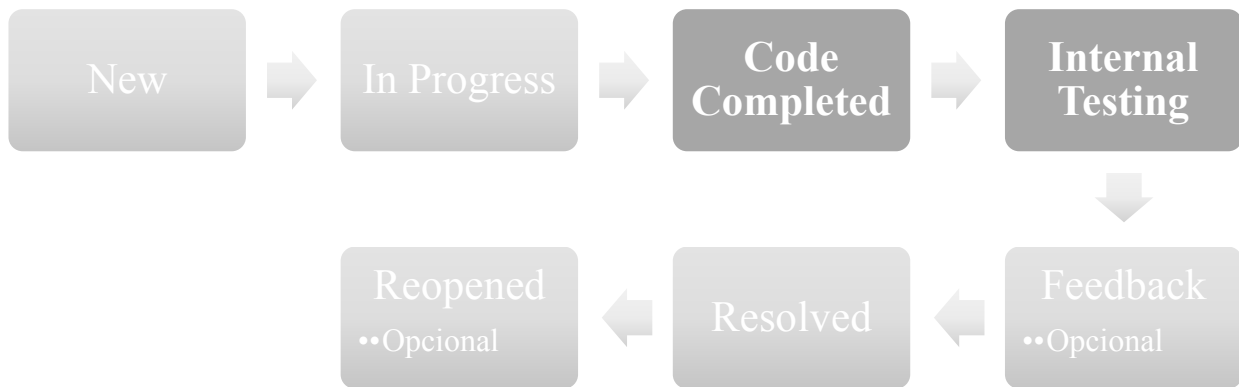


Figura 27: Flujo de ejecución de los “User Stories”

Fuente: Elaboración Propia

Los estados para los “User Stories” no son iguales a los estados que se manejan para los tickets, aunque sí existen similitudes en algunos. Los estados de los tickets definen el flujo de trabajo que establece Spiceworks en “Redmine”. Los estados para los “User Stories” definen el flujo que seguirían los “User Stories” en el proceso de desarrollo propuesto. La siguiente lista resume las diferencias y similitudes que existen entre ambos:

- Los estados “New”, “Resolved”, “Feedback” y “Reopened” mantienen el mismo significado tanto para los tickets como para los “User Stories”
- El estado “In Progress” desde el contexto de los “User Stories” no involucra la fase de pruebas, solo su desarrollo
- Se agregó el estado “Code Completed”, para demarcar aquellos “User Stories” que ya están listos por parte de la persona encargada de su desarrollo; pero que tienen pendiente la fase de pruebas internas
- Se agregó el estado “Internal Testing”, para demarcar los “User Stories” que se encuentran en la fase de pruebas por parte del equipo de Integración

- Los estados “Feature Smoke test complete”, “Testing”, “Verified Internal” y “Verified”, no fueron incluidos dentro del flujo que siguen los “User Stories”, pues corresponden a las actividades que realiza Spiceworks, y el equipo de Integración no tiene interferencia alguna en ellas.

La siguiente tabla (Tabla 16: Comparación de estados utilizados en “Redmine” versus “User Stories”), muestra los estados que se utilizarían para “Redmine”, así como aquellos que se emplearían para los “User Stories”.

Estado	¿Utilizado en “Redmine”?	¿Utilizado en los “User Stories”?
“New”	√	√
“In Progress”	√	√
“Feedback”	√	√
“Feature Smoke test complete”	√	
“Testing”	√	
“Resolved”	√	√
“Reopened”	√	√
“Verified Internal”	√	
“Verified”	√	
“Code Completed”		√
“Internal Testing”		√

Tabla 16: Comparación de estados utilizados en “Redmine” versus “User Stories”

Fuente: Elaboración Propia

4.2.3. Roles del proceso de desarrollo

La tabla que se expone a continuación (Tabla 17: Tabla de roles del proceso de desarrollo) describe los roles necesarios durante el proceso de desarrollo propuesto, así como las responsabilidades de cada uno.

Nombre del Rol	Responsabilidades
“Product Owner”	<ul style="list-style-type: none"> • Definir las características y funcionalidades que se van a construir, es decir, mantener actualizado el “Product Backlog” • Establecer cuál es la visión y metas de cada “Sprint” • Definir prioridades • Estar disponible para resolver los bloqueos con la mayor brevedad posible • Responder preguntas que puedan surgir durante el “Sprint”
“ScrumMaster”	<ul style="list-style-type: none"> • Ayudar a todas las personas involucradas en el proyecto a comprender y adoptar los valores y principios de “Scrum” • Ayudar a resolver problemas • Remover obstáculos • Evitar que el equipo de trabajo tenga interferencia externa que pueda afectar el “Sprint”

Nombre del Rol	Responsabilidades
	<ul style="list-style-type: none"> • Dar asistencia y ayudar a organizar el “Daily Scrum”, el “Sprint Planning”, el “Sprint Retrospective” y el “Sprint Review” • Debido a que en el equipo de Integración existen dos personas que cuentan con la certificación de “ScrumMaster”, se recomienda que éstas sean las seleccionadas para ejercer este rol.
Equipo de trabajo	<ul style="list-style-type: none"> • Diseñar, desarrollar y probar “User Stories” • Solicitar ayuda tan pronto como sea posible cada vez que haya un obstáculo • Auto gestionarse, es decir, organizar el trabajo por realizar de la mejor manera posible

Tabla 17: Tabla de roles del proceso de desarrollo

Fuente: Elaboración Propia

4.2.4. Eventos del proceso de desarrollo

Los eventos son ceremonias que deben ocurrir en cada “Sprint”. Son parte del proceso, son elementos fundamentales, lo reinician y mejoran de manera constante. La siguiente tabla (Tabla 18: Tabla de eventos del proceso de desarrollo), describe cada uno de los eventos que se deberían llevar a cabo, las personas que deberían participar; además, un conjunto de recomendaciones por considerar en la ocurrencia de estos eventos.

Nombre del evento	Descripción	Asistentes	Recomendaciones
“Daily Scrum”	<p>Esta es una reunión que debe ocurrir de manera diaria, en donde cada integrante del equipo de trabajo da su estado al responder las siguientes preguntas:</p> <ul style="list-style-type: none"> • ¿Qué he logrado desde el último “Daily Scrum”? • ¿Qué planeo seguir trabajando? • ¿Cuáles son los obstáculos que me impiden seguir avanzando? 	<p>Equipo de trabajo “ScrumMaster” “Product Owner” (opcional)</p>	<ol style="list-style-type: none"> 1. Esta reunión no debe durar más de 15 minutos. Si hubiese temas específicos por discutir deberían tratarse después de la reunión, con solo aquellas personas que están involucradas. 2. Debería realizarse siempre en el mismo lugar y a la misma hora 3. Todos los miembros del equipo deben atender esta reunión
“Sprint Planning”	<p>Esta reunión tiene como objetivo definir los “User Stories” que se trabajarán en el “Sprint”. En esta</p>	<p>“ScrumMaster” Equipo de Trabajo</p>	<ol style="list-style-type: none"> 1. El “Product Owner” debe preparar el “Product Backlog” previo a la reunión.

Nombre del evento	Descripción	Asistentes	Recomendaciones
	<p>ceremonia el Equipo de Trabajo establece su compromiso para el “Sprint” (el trabajo al que se compromete finalizar). El Equipo de Trabajo puede sugerir que se puede trabajar durante el siguiente “Sprint”; pero es el “Product Owner” quien tiene la decisión final.</p>	<p>“Product Owner”</p>	<p>2. Esta reunión se debería dividir en dos segmentos, uno para seleccionar los ítems que se trabajarán en el siguiente “Sprint” y otro para preparar el “Sprint Backlog”</p> <p>3. La ceremonia no debería tardar más de 8 horas</p>
“Sprint Review”	<p>Esta reunión tiene como objetivo mostrar al “Product Owner” y a los involucrados en el proyecto, el trabajo finalizado en el “Sprint”. Al finalizar la ceremonia, los involucrados pueden</p>	<p>“ScrumMaster”</p> <p>“Product Owner”</p> <p>Involucrados en el proyecto</p> <p>Equipo de Trabajo</p>	<p>1. La preparación de esta reunión no debería tardar más de 1 hora</p>

Nombre del evento	Descripción	Asistentes	Recomendaciones
	dar retroalimentación o hacer sugerencias.		
“Sprint Retrospective”	<p>Esta reunión tiene como objetivo el mejoramiento continuo por medio del análisis del “Sprint” que finalizó. A cada miembro del equipo se le deberían hacer dos preguntas:</p> <p>¿Qué se hizo bien durante el “Sprint” pasado que se pueda repetir?</p> <p>¿Qué se puede mejorar en el siguiente “Sprint”?</p> <p>De la retroalimentación recibida, se deberían establecer planes de acción, a fin de continuar mejorando en los “Sprints” futuros.</p>	<p>Equipo de Trabajo</p> <p>“ScrumMaster”</p> <p>“Product Owner”</p>	<p>1. Esta reunión no debe tardar más de 3 horas</p>

Nombre del evento	Descripción	Asistentes	Recomendaciones
“Grooming”	Esta actividad tiene como finalidad crear, refinar, estimar y priorizar los ítems del “Product Backlog”.	Equipo de Trabajo (no es necesario que estén todos los miembros del equipo) “Product Owner” “ScrumMaster”	1. Esta actividad se debe dar de manera regular, a fin de que se mantengan en el “Product Backlog” suficientes “User Stories” para trabajar en el siguiente “Sprint”.

Tabla 18: Tabla de eventos del proceso de desarrollo

Fuente: Elaboración Propia

4.3. Evaluación de herramientas

En esta etapa, se llevaron a cabo distintas actividades relacionadas con evaluar herramientas para soportar procesos de desarrollo basados en “Scrum”, además de determinar en qué medida la herramienta utilizada por el equipo de Integración, cumple con las necesidades requeridas. Lo anterior se realizó con la finalidad de proponer una herramienta para soporte del proceso de desarrollo formulado, y que a su vez cumpla con las características que el equipo de Integración busca.

4.3.1. Evaluación la herramienta “Redmine”

“Redmine” es la herramienta de software utilizada por el equipo de Integración para dar soporte a su proceso de desarrollo. En esta actividad, se condujo una entrevista grupal realizada por medio de una vídeo conferencia a dos líderes técnicos del equipo de Integración, con fin de conocer en cuánta medida “Redmine” estaba cumpliendo las necesidades del equipo, además de plantear la interrogante de un posible reemplazo. En el Capítulo de Anexos, la sección 6.4: Entrevista para evaluar la herramienta “Redmine” muestra la información obtenida en dicha entrevista. A continuación, se describen los principales hallazgos:

- “Redmine” no está cumpliendo a cabalidad con las necesidades del equipo de Integración, esto debido a dos factores principales:
 - Los reportes proporcionados por la herramienta no proveen la información que necesita el equipo
 - La creación de sub-tareas es considerada engorrosa, además de que lo que se genera en lugar de una sub-tarea, es otro tiquete, lo cual no resulta conveniente
- No es posible reemplazar “Redmine” con otra herramienta, debido a que su uso está establecido desde cláusulas contractuales
- Por medio de “Redmine” se da visibilidad en Spiceworks al trabajo que realiza el equipo de Integración
- “Redmine” provee mecanismos para soportar procesos de desarrollo basados en “Scrum”, pero se requiere instalar extensiones, operación que no es permitida por Spiceworks

- A pesar de que “Redmine” no puede ser reemplazado, sí podría ser complementado con una herramienta adicional. Los entrevistados sugirieron la utilización de un tablero físico, debido a que la oficina ya tiene un espacio acondicionado para ello

4.3.2. Características deseables en una herramienta

Con el objetivo de identificar las características deseables en una herramienta que ayude a soportar el proceso de desarrollo propuesto, se condujo una entrevista grupal, realizada por medio de una vídeo conferencia con dos líderes técnicos del equipo de Integración, uno de ellos, el administrador de dicho equipo. La información recolectada en dicha entrevista se encuentra en el Capítulo de Anexos, en la sección 6.5: Entrevista para determinar las características deseables en la herramienta. Las características deseables mencionadas en dicha entrevista se resumen a continuación:

- ✓ Preferiblemente gratuita
- ✓ Debe permitir visualizar el estado actual de trabajo que está en progreso
- ✓ No debe conllevar una carga adicional de trabajo para las personas (fácil de usar)
- ✓ Debe ser atractivo, de manera que la personas se vean motivadas a utilizarla
- ✓ Debe complementarse con “Redmine” de alguna manera

4.3.3. Selección y evaluación de herramientas

Esta actividad consistió en consultar documentos electrónicos, con el fin de encontrar herramientas que ofrece el mercado para soportar procesos de desarrollo basados en “Scrum”.

A partir de la investigación realizada, se seleccionó a “VersionOne” y “Rally” como las herramientas por analizar y evaluar, esto debido a que fueron las que se encontraban mejor calificadas por los usuarios en los sitios web consultados.

Se consideró como una tercera opción, analizar las características de un tablero físico, ya que en la entrevista para evaluar la herramienta Redmine (6.4: Entrevista para evaluar la herramienta “Redmine”) los entrevistados externaron su interés por contar con un tablero físico.

La tabla que se presenta a continuación (Tabla 19: Tabla de herramientas seleccionadas), describe las características de las herramientas analizadas.

Nombre de la herramienta	Características
“VersionOne”	<ul style="list-style-type: none"> • Herramienta web que no requiere instalación • Provee distintos esquemas de precios • Es gratuita hasta para diez usuarios • Soporta colaboración para equipos multifuncionales • Posee robustas habilidades de planeación: “User Stories”, requerimientos, proyectos • Su precio ronda entre \$20 y \$39 por usuario por mes. La variación en el precio depende del paquete que se quiera adquirir
“Rally Dev”	<ul style="list-style-type: none"> • Herramienta web que no requiere instalación

Nombre de la herramienta	Características
	<ul style="list-style-type: none"> • Provee distintos esquemas de precios • Soporta colaboración para equipos multifuncionales • Incluye manejo de defectos • Su precio ronda entre \$35 y \$49 por usuario por mes. La variación en el precio depende del paquete que se quiera adquirir
Tablero físico	<ul style="list-style-type: none"> • Los gastos en los que es incurren son mínimos. Su elaboración requiere: <ul style="list-style-type: none"> ○ “post-its”: son pequeñas hojas de papel adhesivo de varias formas, dimensiones y colores ○ Un espacio físico en el cual se pueda crear el tablero, puede ser una pared o una pizarra • Promueve la conversación y la colaboración • Es más interactiva y divertida • Es actualizada con más frecuencia

Tabla 19: Tabla de herramientas seleccionadas

Fuente: Elaboración Propia basada en (Erickson, 2013), (Salisbury, 2015), (VersionOne, 2016), (Hodzic, 2012)

4.3.4. Herramienta recomendada

Basado en el análisis antes expuesto, se recomienda utilizar el tablero físico como herramienta para soportar el proceso de desarrollo del equipo de Integración por las razones que se exponen a continuación:

- Cumple a cabalidad con los aspectos que busca el equipo de Integración en una herramienta:
 - El tablero físico es más interactivo y divertido que una herramienta virtual, lo que promueve su actualización de manera frecuente
 - Los gastos en los que se incurren son mínimos, solo se requiere “post its” y un espacio físico (éste ya existe)
 - Permite visualizar el estado actual del trabajo que está en progreso
 - No reemplazaría “Redmine”, más bien, lo complementaría
- Ya se cuenta con un espacio físico dentro de la oficina acondicionado para un tablero físico
- Debido a que el equipo de Integración debe mantener actualizado “Redmine” porque es un requerimiento de Spiceworks, podría resultar abrumador para los integrantes del equipo tener que estar actualizando dos herramientas digitales

La imagen que se muestra a continuación: Figura 28: Ejemplo de un tablero físico, ejemplifica un tablero físico en una pared.



Figura 28: Ejemplo de un tablero físico

Fuente: (Koontz, 2013)

4.4. Simulación del Modelo de desarrollo de software

La simulación del modelo de software propuesto, tenía como objetivo evaluar el funcionamiento del proceso de desarrollo y la herramienta, al mismo tiempo que se utilizaba “Redmine”, con la finalidad de simular las propuestas en un ambiente similar al cual operarían.

4.4.1. Modelo de simulación

La definición del modelo de simulación se realizó con el objetivo de definir los aspectos con los cuales se realizaría la simulación. La siguiente tabla: Tabla 20: Datos del modelo de simulación, muestra dichos aspectos.

Tamaño del equipo de trabajo	5 personas
Duración del “Sprint”	1 semana

“Product Backlog”	“Product Backlog” existente en el equipo de Integración
“Scrum Master”	Una de las personas del equipo que cuenta con la certificación de “ScrumMaster”
“Product Owner”	Uno de los líderes técnicos de equipo de Integración

Tabla 20: Datos del modelo de simulación

Fuente: Elaboración Propia

El equipo de trabajo incluyó dos desarrolladores, un ingeniero de calidad, el “ScrumMaster” y el “Product Owner”. La duración del “Sprint” se fijó en una semana pues la empresa estableció dicho tamaño de acuerdo con la disponibilidad de las personas que conformaron el equipo.

4.4.2. Hallazgos

La simulación del modelo de desarrollo se llevó a cabo sin mayores contratiempos bajo los parámetros descritos en la sección anterior (4.4.1: Modelo de simulación). Durante dicho proceso, el “ScrumMaster” designado tenía la tarea de documentar los hallazgos más importantes que se dieran día a día. En el Capítulo de Anexos, en la sección 6.6: Bitácora de la simulación del modelo de desarrollo se puede encontrar dicha bitácora. Al analizar los datos recabados en la misma, se desprende lo siguiente:

- ✓ El proceso de desarrollo de software propuesto pudo ser ejecutado por el equipo de manera correcta
- ✓ Los roles definidos en el proceso de desarrollo funcionaron de manera adecuada

- ✓ Los eventos definidos en el proceso de desarrollo se llevaron a cabo exitosamente
- ✓ Los artefactos definidos en el proceso de desarrollo fueron utilizados correctamente, aunque se debe mejorar la redacción de los “User Stories”
- ✓ El tablero físico funcionó adecuadamente, y las personas se sintieron a gusto con él
- ✓ El tablero físico permitió visualizar el estado del “Sprint” en todo momento
- ✓ Los integrantes del equipo resaltaron el beneficio de tener la figura del “ScrumMaster”, que ayuda a remover bloqueos
- ✓ Las estimaciones realizadas para las tareas no resultaron acertadas, debido a que se duró mucho más de lo que se estimó en un principio

En las imágenes que se presentan a continuación, se puede apreciar el tablero físico al inicio y al final de la simulación.



Figura 29: Tablero físico al inicio de la simulación

Fuente: Dino I.T. Technologies



Figura 30: Tablero físico al final de la simulación

Fuente: Dino I.T. Technologies

La simulación realizada demostró que si bien hay elementos de mejora, el modelo de desarrollo de software propuesto, constituido por el proceso de desarrollo y el tablero físico funcionan, y además las personas participantes en la simulación pudieron encontrar beneficios con respecto a lo que se cuenta hoy en día en el equipo de Integración.

4.5. Recomendaciones para implementar el modelo de desarrollo de software

En esta sección se presentan una serie de recomendaciones que se sugiere sean consideradas a la hora de implementar el modelo de desarrollo propuesto.

4.5.1. Formato para definir “User Stories”

Una de las debilidades hallada en la etapa de simulación del modelo propuesto fue que los “User Stories” no contenían toda la información necesaria para su implementación. Esto puede ser atacado por medio de la definición de una plantilla que especifique qué debería contener un “User Story”.

A continuación, se presenta una propuesta de una plantilla (Tabla 21: Plantilla para la creación de “User Stories”) que podría ser utilizada para la creación de User Stories”.

Sección	Descripción
Título	Se refiere a una descripción general del “User Story”.
Descripción	Descripción detallada de lo que se requiere implementar, indicando: <ul style="list-style-type: none"> • ¿Quién sería el usuario beneficiado? • ¿Qué es lo que se quiere lograr? • ¿Cuál sería el beneficio obtenido?
Razonamiento	Describe porque el “User Story” debe ser implementado.

Sección	Descripción
Criterios de aceptación	Esta es una lista de chequeo en donde se describe cuáles criterios serán utilizados para aceptar un “User Story”, esto es, darlo por finalizado.
Notas	Describe información adicional del “User Story”, por ejemplo, si tiene algún tipo de dependencia.

Tabla 21: Plantilla para la creación de “User Stories”

Fuente: Elaboración Propia

A continuación, se da un ejemplo de cómo se visualizaría la plantilla propuesta.

Sección	Descripción
Título	Pantalla de autenticación para administradores
Descripción	Como Administrador, deseo poder autenticarme en la aplicación por medio de un usuario y una clave, a fin de poder obtener acceso a todas las funcionalidades del sistema.
Razonamiento	Todas las funcionalidades a las que tiene acceso un administrador son habilitadas una vez que el mismo se ha autenticado en el sistema.
Criterios de aceptación	<ul style="list-style-type: none"> • Debe tener una caja de texto para ingresar el usuario y otra para ingresar la clave • Debe tener un botón con la etiqueta “Ingresar” • Debe mostrar una alerta cuando el usuario ha fallado ya sea el usuario o la clave

Sección	Descripción
	<ul style="list-style-type: none"> • Debe mostrar una alerta cuando el usuario presiona el botón “Ingresar” pero aún no se ha ingresado el usuario • Debe mostrar una alerta cuando el usuario presiona el botón “Ingresar” pero aún no se ha ingresado la clave
Notas	Esta pantalla se puede realizar cuando el servicio web que realiza la autenticación está listo.

4.5.2. Estructuras de Equipos de “Scrum”

El equipo de Integración está conformado por doce personas. Se recomienda la creación de dos equipos de trabajo, compuestos por seis personas cada uno, a fin de evitar tener un equipo con más de nueve personas. Lo anterior implicaría lo siguiente:

- Cada equipo debe tener un “ScrumMaster”
- Los equipos deben mantenerse sincronizados por medio la reunión denominada “Scrum of Scrums”
- Cada equipo debería tener su propio tablero físico
- Cada equipo debe tener su propio “Sprint Backlog”
- Ambos equipos podrían compartir el “Product Backlog”
- Actualmente se cuenta con solo una persona de aseguramiento de calidad, la misma podría trabajar medio tiempo para un equipo y medio tiempo para otro equipo

4.5.3. Establecer con el “Product Owner” sus responsabilidades

El “Product Owner” es un rol indispensable, pues define la dirección que sigue el proyecto(s), basado en su entendimiento del negocio y de los clientes. Aunque Spiceworks tiene una persona que en la actualidad realiza funciones compatibles con las de un “Product Owner” para el equipo de Integración, se recomienda aclarar qué es lo que se espera de este rol y sus responsabilidades. Para esto, se puede utilizar la información presentada en la sección 4.2.3: Roles del proceso de desarrollo.

4.5.4. Capacitar al equipo de Integración

La mayoría de personas que componen el equipo de Integración han tenido alguna experiencia con “Scrum”; sin embargo, es altamente recomendable que se capacite a todos los miembros del equipo, para que se tenga un nivel de conocimiento estandarizado. Se propone realizar el proceso de capacitación de la siguiente manera:

- Debido a que el equipo cuenta con dos personas certificadas en “Scrum”, las mismas podrían encargarse de esta tarea
- El proceso de capacitación se podría dar en tres sesiones con una hora de duración cada una, en donde se tendría:
 - Una sesión para explicar los artefactos de Scrum
 - Una sesión para abordar el tema de las ceremonias y su importancia
 - Una sesión para abordar el tema de los roles, exponiendo las responsabilidades de cada uno
- A fin de que todos los miembros del equipo participen, se recomienda que la capacitación se dé en horas laborales

La cantidad de tiempo estimado para invertir en este proceso de capacitación serían ciento cincuenta y seis horas:

- 12 horas para preparar el material de cada sesión (4 horas por sesión)
- 144 horas para impartir las sesiones a todos los miembros del equipo (12 horas por persona para las 3 sesiones x 12 personas que componen el equipo)

Se estima que el costo de esta actividad sería de \$4,680 (utilizando un precio promedio de \$30 por hora).

4.5.5. “Sprints” mensuales

Debido a que Spiceworks agrega de manera mensual mejoras y nuevas características a los proyectos en los cuales trabaja el equipo de Integración, se recomienda utilizar “Sprints” de un mes de duración, de esta manera cada vez que hay un pase a producción se finaliza el “Sprint” y se comienza con uno nuevo.

4.5.6. Gestión del cambio

La implementación del modelo de desarrollo propuesto supone un cambio en la manera en la que el equipo de Integración ha venido trabajando, así, se debería gestionar el cambio con el objetivo de aumentar las probabilidades de que la implementación sea exitosa.

A continuación, se presentan algunas recomendaciones sobre cómo se podría gestionar el cambio:

1. Comunicar al equipo de Integración lo que se pretende lograr con la implementación del modelo de software, enfatizando en los beneficios que

éste traería al equipo, de manera que todos los involucrados comprendan por qué es necesario realizar el cambio

2. Todo cambio conlleva en muchos casos resistencia por parte de las personas afectadas. A fin de mitigar una posible resistencia al cambio, se recomienda:
 - a. Identificar al menos dos personas que se encuentren identificadas y comprometidas con el cambio. Estas dos personas podrían encargarse de motivar al equipo de trabajo, y además estar disponibles en todo momento para ayudar a los miembros del equipo de trabajo a resolver posibles situaciones que se puedan presentar
3. Realizar reuniones de manera periódica con el equipo de trabajo, en la cuales se presentaría el avance de la implementación del modelo de software y los logros obtenidos

4.5.7. Gestión del riesgo

La implementación del modelo de desarrollo propuesto trae consigo riesgos, para los cuales se debe tener un plan de acción, a fin de evitarlos de ser posible; y si no fuera posible, mitigar su impacto. La tabla que se presenta a continuación (Tabla 22: Riesgos identificados que se podrían dar durante la fase de implementación), enumera algunos de los riesgos a los que se podría enfrentar el equipo de Integración durante la implementación del modelo propuesto. Para cada riesgo se establece un posible grado de ocurrencia, y un plan de acción que se recomienda tomar en caso de que el riesgo se materialice.

Riesgo	Posible grado de ocurrencia	Plan de acción en caso de que ocurra
El tablero físico no se adecua a las necesidades del equipo	Bajo, esto debido a que es utilizado por muchas empresas y ha demostrado funcionar.	Considerar utilizar alguna de las otras herramientas analizadas y evaluadas: “Rally” o “VersionOne”
Resistencia al cambio	Bajo, pues la mayoría de los miembros del equipo tienen experiencia con “Scrum”	- Presentar los beneficios que se van obteniendo a través del tiempo del modelo, esto con el fin de tener más personas comprometidas con la causa
El “Product Owner” no ejerce su rol correctamente	Bajo, debido a que en el equipo de Integración, existe una persona que lleva a cabo las responsabilidades que son esperadas del “Product Owner”	Agendar una reunión con el “Product Owner” y establecer cuáles son sus responsabilidades y lo que se espera de dicho rol.

Tabla 22: Riesgos identificados que se podrían dar durante la fase de implementación

Fuente: Elaboración Propia

5. Conclusiones

A continuación, se presentan las conclusiones que se desprenden del trabajo realizado.

- El proyecto culmina de manera exitosa por medio del cumplimiento de los objetivos establecidos. Lo anterior se da gracias al planteamiento de la propuesta del proceso de desarrollo basado en “Scrum” para el equipo de Integración, que establece roles y una herramienta para soportar dicho proceso. Por medio del análisis de la situación actual se conocieron las características y necesidades del equipo de Integración, y basado en ello, se dieron recomendaciones para la implementación exitosa de este proceso en el futuro.
- El análisis de la situación actual realizado sobre el equipo de Integración, permitió conocer que dicho equipo no tenía un proceso de desarrollo claramente establecido. De la información obtenida, fue posible deducir que existía confusión entre los miembros del equipo sobre cuál metodología o marco de trabajo se estaba empleando.
- En el análisis de la situación actual llevado a cabo en el equipo de Integración, en uno de los cuestionarios aplicados se encontraron gran cantidad de divergencias en las respuestas brindadas por las personas. Al tratar de comprender esta situación, se descubrió que algunas personas pensaban que utilizan Kanban en su proceso de desarrollo, pues en el pasado se hicieron algunos esfuerzos por implementar dicho sistema.

De esto, es posible anotar que las personas pueden confundir experiencias previas con la realidad de una empresa, solo porque encuentran ciertos elementos y características similares o familiares.

- A partir del análisis de la situación actual, fue posible identificar que el equipo de Integración contaba con ciertos elementos e insumos que compartían características con elementos de “Scrum”, como lo es el “Product Backlog”, la figura del “Product Owner”, y el “Daily Scrum”.
- El marco de trabajo “Cynefin” se aplicó sobre el equipo de Integración, con el fin de analizar sus características y determinar si “Scrum” resultaba adecuado para gestionar el trabajo de este equipo. Dicho análisis permitió ubicar al equipo de Integración bajo el dominio Complejo, o sea, “Scrum” es una opción viable para gestionar el trabajo de este equipo según lo que establece este marco de trabajo.

“Cynefin” es una herramienta muy valiosa, permite identificar las circunstancias y características bajo las cuales es más adecuado utilizar “Scrum”.

- A pesar de que la empresa Dino cuenta con una cláusula contractual con su cliente único (“Spiceworks”) la cual establece que el equipo de Integración debe utilizar una herramienta de software específica (“Redmine”), y el flujo de trabajo definido en esta, fue posible formular un proceso de desarrollo, que utiliza como base los estados definidos en dicho flujo de trabajo.

Así mismo, la formulación del proceso de desarrollo basado en “Scrum” fue posible gracias a la existencia de otra cláusula contractual donde Dino puede gestionar el trabajo de la manera que mejor le convenga.

- Un elemento que facilitó la formulación del proceso de desarrollo basado en “Scrum” fue que Spiceworks provee un punto de contacto principal, encargado de asignar el trabajo y las prioridades al equipo de Integración. Estas labores son las que desempeña el “Product Owner”.
- El proceso de desarrollo propuesto para el equipo de Integración incorpora, en primera instancia, los insumos y actividades que sugiere el marco de trabajo “Scrum”. El planteamiento ofrece recomendaciones para cada elemento del proceso, éstas se basan en lo identificado durante el análisis de la situación actual. Este proceso de desarrollo podrá ser adaptado y mejorado de acuerdo con las necesidades que vayan surgiendo mediante su análisis en cada “Sprint Retrospective”.
- La incorporación de “Scrum” al proceso de desarrollo del equipo de Integración supone la necesidad de una herramienta que apoye dicho proceso. La herramienta de software utilizada por el equipo de Integración (“Redmine”) en la actualidad, no provee mecanismos para soportar procesos basados en “Scrum”, y no puede ser reemplazada debido a que su utilización está supeditada a cláusulas contractuales por su cliente único “Spiceworks”. Debido a lo anterior, fue necesario analizar otras herramientas para apoyar el proceso de “Scrum” propuesto, y que al mismo tiempo fuera compatible con “Redmine”. Se concluyó que la mejor opción disponible era un tablero físico, pues permitía complementar “Redmine”, sin agregar una carga de trabajo extra que podría ser acarreada si se planteaba la utilización de otra herramienta de software.

- El tablero físico es una alternativa para complementar herramientas de software que no soportan procesos de desarrollo basados en “Scrum”. Su principal ventaja reside en que promueve la conversación y la colaboración, además, es interactiva. Lo anterior promueve su actualización de manera frecuente.
- Dino I.T Technologies es una PYME en la cual fue posible adaptar el proceso de “Scrum” de manera exitosa, sin aumentar costos por recursos adicionales.

5.1. Limitaciones

A continuación, se presentan las limitaciones encontradas en la realización de este proyecto:

- El modelo de desarrollo fue simulado solo un “Sprint” debido a limitaciones de tiempo.
- No fue posible valorar el reemplazo de la herramienta “Redmine”, debido a que su utilización es un requerimiento de “Spiceworks”.

5.2. Trabajos Pendientes

- Simular en el modelo de desarrollo varios “Sprints”, a fin de confirmar que tanto el proceso de desarrollo como el tablero físico se integran adecuadamente con “Redmine”.
- Definir métricas para el proceso de desarrollo formulado.
- Definir un mecanismo que permita atender los bloqueos con mayor prontitud.

6. Anexos

A continuación, se presentan los documentos utilizados en la realización del presente proyecto.

6.1. Ideas para mejorar el proceso de desarrollo actual

La imagen que se muestra a continuación, presenta las ideas propuestas por algunos miembros del equipo de Integración, para mejorar el proceso de desarrollo.

15. ¿Tiene usted alguna idea de que se podría mejorar en el proceso de desarrollo que emplea el equipo de Integración?

(6 responses)

-
Sería conveniente implementar algunas métricas de desempeño para poder realizar acciones de mejoramiento y optimización del proceso de desarrollo tanto a nivel de equipo como a nivel individual.
Definir procesos que tambien queden por escrito para eventos que cada vez se frecuenten más e impiden en desarrollar o terminar un ticket.
Seguir utilizando la wiki
Realizar monitoreo de los procesos establecidos puede señalar dónde y cómo se pueden mejorar.
Reuniones semanales para evaluar rendimiento y establecer metricas

Figura 31: Respuestas de la pregunta #15 del cuestionario para conocer la situación actual

Fuente: "Google Forms"

6.2. Entrevista para determinar los estados de un tiquete

La tabla que se presenta a continuación (Tabla 23: Aplicación de la entrevista para determinar los estados de un tiquete), contiene las respuestas brindadas por los dos líderes técnicos que participaron en la entrevista grupal realizada para conocer los estados de un tiquete.

Pregunta	Respuesta
<p>¿Podría describir los estados por los que pasan los tickets dentro del flujo normal de desarrollo, desde su comienzo hasta su finalización?</p>	<p><u>“New”</u>: Un ticket que no ha sido asignado a ninguna persona y que además no se ha comenzado a trabajar</p> <p><u>“In Progress”</u>: Un ticket que se encuentra asignado a una persona y ésta ya lo comenzó a trabajar. Cuando a un ticket se le están realizando pruebas del lado del equipo de Integración (“Internal testing”), el ticket se mantiene en este estado.</p> <p><u>“Feedback”</u>: Cuando un ticket requiere de información adicional por parte de Spiceworks para poder continuar trabajando en él.</p> <p><u>“Resolved”</u>: Cuando tanto el desarrollador como el ingeniero de calidad consideran que el ticket está listo.</p> <p><u>“Testing”</u>: Un ticket se encuentra en este estado cuando el equipo de calidad de Spiceworks está realizando pruebas en él.</p>

Pregunta	Respuesta
	<p><u>“Feature Smoke test complete”</u>: un ticket se encuentra en este estado cuando se ha realizado una revisión rápida de todo el sistema, a fin de verificar que no hay afectación en otras funcionalidades del mismo.</p> <p><u>“Reopened”</u>: este estado es utilizado cuando un ticket que se consideraba listo necesita cambios, o es necesario arreglar comportamientos indeseados.</p> <p><u>“Verified internal”</u>: Cuando un ticket se encuentra en este estado significa que ya pasó un conjunto de pruebas automatizadas de integración.</p> <p><u>“Verified”</u>: Cuando un ticket fue probado en el ambiente de producción y funcionó adecuadamente.</p>
¿Podrían algunos de estos estados ser eliminados o modificados?	No, debido a que fueron establecidas por Spiceworks, además, están configurados en la

Pregunta	Respuesta
	herramienta “Redmine”, la cual es utilizada por diversos equipos de trabajo de Spiceworks.

Tabla 23: Aplicación de la entrevista para determinar los estados de un ticket

Fuente: Elaboración Propia

6.3. Documentos revisados

A continuación, se presentan los documentos que fueron revisados en la etapa de análisis de la situación actual, del modelo metodológico utilizado en la investigación.

ID	Priority	Description	Category	Status
Community				
62333	Normal	404 errors with the software advisories injection	Community Future	
62996	Normal	Manage Engine KB article generation: Tests	Community Next	
77910	Normal	Parts Finder not showing up for manually managed device	Community Next	
Costa Rica				
77953	Normal	Reports - timeframe options not converting properly		
78470	Normal	Update software license modal to scroll page properly		
78506	Normal	7.5.00077 Social sign in settings toggle is reversed		
Desktop				
78524	Normal	remote sites settings - outdated site help popup display problem	7.5 - Patch	
69414	High	Turning Unknowns into partially scanned devices		7.6
72029	High	Settings: Not everything saving in new Settings pages		7.6
76319	High	Fix/update software widget		7.6
44851	Normal	MS Office 2013 Product Key not being retrieved		7.6
45207	Normal	surface os and processor architecture in inventory		7.6
45400	Normal	Up/Down Check Manually Managed Devices		7.6
45767	Normal	In case of same software title with different vendor names, last scan 'wins'		7.6
46195	Normal	Finder: Devices are skipped during scan if device has same ip address as previously reported by remote agent		7.6
49499	Normal	Network Scan Page: Single IP exclusion appears in its scan range, range of exclusions appear in all ranges		7.6
49582	Normal	Cloud Services: Adding a new regex Cloud Service requires a page refresh to show up		7.6
51064	Normal	Spiceworks Desktop should notify admins when a failure to backup network device configurations occurs		7.6
51470	Normal	Finder: VMs tab no longer links back to child VM inventory pages for Server 2012 r2 Hyper-V hosts		7.6
52242	Normal	Email and Schedule Reports: Ensure Graph is included		7.6
52880	Normal	Finder: "OpenSSL::Cipher::CipherError: bad decrypt" exception is not being sent to the sky		7.6
56898	Normal	Linux Software not scanning in for multiple OS and versions		7.6
56906	Normal	reports exported to XLS don't open properly		7.6
58024	Normal	Cloud Services: Older detections causing app to think there aren't any		7.6
58447	Normal	Email and Schedule Reports: Add loading animation/feedback to rules section in edit page		7.6
58485	Normal	Email and Schedule Reports: Add "Purchases" linking in report show page		7.6
59268	Normal	Include Maas360 account number on MDM Reconnect popup		7.6
60069	Normal	DNS Checkup errors caused by WAD scan		7.6
60158	Normal	Reporting: Reports exported to CSV have misc. character with foreign currency		7.6
60399	Normal	MDM: FF: Selecting expired enrollments doesn't allow resend button to become enabled		7.6
60901	Normal	report column widths no longer resizable		7.6
61153	Normal	Reports: Criteria "Free Disk Space" expecting bytes as a value		7.6
62481	Normal	TestCases:C186563 - Should only display months in datepicker if "end of the month" selected in report schedule		7.6
62487	Normal	TestCases: C14666 - Should be able to unenroll a device from Fiberlink		7.6
62511	Normal	Test Cases: C890 - Set a monitor for web domain expiration		7.6
63421	Normal	Reports: Time zone not converted back to local time when editing a report schedule		7.6

Figura 32: 05 06 2016 Status April 2016.xlsx

Fuente: Dino I.T. Technologies

7.5 - Patch May

#	Tracker	Status	Priority	Subject	Assignee ▲	Updated
<input type="checkbox"/> New 26						
68341	Bug	New	Normal	Devices with login errors don't populate card pile when scanning	redmine-costa-rica-team	03/21/2016 02:4
78225	Bug	New	Normal	Software Updates/Uninstalls Not Showing in Timeline	redmine-costa-rica-team	05/05/2016 10:1
70761	Bug	New	Normal	Backbone save function not working on inventory	redmine-costa-rica-team	03/21/2016 02:5
71538	Bug	New	Normal	when "on ready" event for plugin is called, it should handle exceptions	Bruno Sanchez	03/21/2016 02:5
75888	Bug	New	Normal	Manage Apps screen does not indicate when a new version of an app is available.	Bruno Sanchez	03/23/2016 09:5
74425	Bug	New	Normal	Settings - Inventory - Device Scanning Page will not load	Carlos Mora	05/06/2016 12:5
77018	Bug	New	Normal	"Activity title failed to load" error on Cloud Services	Carlos Mora	05/04/2016 02:4
76383	Bug	New	High	Remote Agent not	Eduardo	05/04/2016 10:4

Figura 33: 7.5 - Patch May

Fuente: Dino I.T. Technologies

You guys should have noticed a bunch of stuff move into patch April. This is going to be an inventory-only patch and currently cannot include anything settings (too hard to merge back into the single Ember app for 7.6). Current code freeze is targeted at 4/22.

7.6 Beta is now targeted for around May 20th (release, not code freeze), although this date may change based on the Rails 3 project. They hope to determine a more accurate time frame in the next few weeks.

Thus, people with feature tickets in the 7.6 queue (and the high priority tickets in there) need to be aware that the dates are probably going to overlap some and cannot wait until patch April ships before beginning work on 7.6. Some percentage of time should be spent on the important 7.6 stuff as well. Once again, the 7.6 projects that need to make Beta are:






- Linux Agent
- IP names to hostname in bulk delete
- Fix VMs
- Up/Down check manually managed devices
- Custom OIDs

Let me know if any of this doesn't make sense or if you have questions.

Figura 34: Patch April and 7.6

Fuente: Dino I.T. Technologies

History

-  Updated by **Samantha Salinas** 5 months ago #1
 - **Related to** *Ticket #72369: Stored XSS activate on complete profile screen when device name is JS, Helpdesk version 7.5.00059* added
-  Updated by **Connie Krueger** 5 months ago #2
 - **Priority** changed from *Normal* to *High*
-  Updated by **Ben Brookshire** 5 months ago #3
 - **Target version** changed from *App Triage* to *7.5 - Patch Feb*
-  Updated by **Jackie Gross** 5 months ago #4
 - **Target version** changed from *7.5 - Patch Feb* to *7.5 - Patch Jan*
 - **Assignee** changed from *Jackie Gross* to *redmine-costa-rica-team*
-  Updated by **Sergio Fernandez** 5 months ago #5
 - **Status** changed from *New* to *In Progress*
 - **Assignee** changed from *redmine-costa-rica-team* to *Sergio Fernandez*

Associated revisions

Revision f6194e6c
Added by Sergio Fernandez 5 months ago

Refs #72409 Prevent XSS with custom device name on profile screen

Revision 24a0e9a2
Added by Lucia Serrano 5 months ago

Merge branch 'bug/72409_cr_xss_device_name_on_profile_' into 'target/7_5_patch'

Refs #72409 Prevent XSS with custom device name on profile screen

Assignee: @sergiof
Branched off target/7_5_patch
Redmine:
<https://redmine.spice.spiceworks.com/issues/72409>
This ticket has no modified stats.
The CI is passing on bug/72409_cr_xss_device_name_on_profile_s
It has been verified on: FF, Chrome.

See merge request !719

Revision 8ae12d5b
Added by Sergio Fernandez 4 months ago

Refs #72409 Prevent XSS with custom

Figura 35: Bug #72409

Fuente: Dino I.T. Technologies

6.4. Entrevista para evaluar la herramienta “Redmine”

La tabla que se muestra a continuación (Tabla 24: Aplicación de la entrevista para evaluar la herramienta “Redmine”), presenta las respuestas recibidas en la entrevista para evaluar la herramienta “Redmine”.

Pregunta	Respuesta
¿Considera usted que “Redmine” le provee todas las características necesarias	No, debido a que la “Redmine” es compartida con otros equipos (de Spiceworks), y no es posible configurar la herramienta con características

Pregunta	Respuesta
<p>para soportar el proceso de desarrollo de software actual?</p>	<p>específicas para cada equipo de acuerdo con sus necesidades.</p> <p>Se han dado situaciones en las cuales “Redmine” ofrece extensiones para suplir una necesidad específica, pero no es posible instalarla.</p> <p>Considero que “Redmine” no es muy flexible en la definición de sub-tareas, debido a que la creación de las mismas resulta engorroso, y al final se genera otro tiquete, no una sub-tarea.</p> <p>Además, los reportes que ofrece considero son insuficientes y no siempre aportan información valiosa</p>
<p>¿Tiene el equipo de Integración la protestad de cambiar “Redmine” si fuese necesario?</p>	<p>No es posible cambiarlo debido a cláusulas contractuales con Spiceworks, pero sí podría ser posible utilizar de manera paralela otra herramienta para soportar el proceso de desarrollo del equipo de Integración, siempre y cuando se actualice “Redmine” de manera periódica para reflejar el estado del trabajo que se está realizando.</p>

Pregunta	Respuesta
¿Considera usted que tendría un efecto negativo sobre el equipo de Integración utilizar otra herramienta que no sea “Redmine”?	No, hace algunos años se había utilizado otra herramienta paralela a “Redmine”, y no hubo efectos negativos.
¿Sabe usted si “Redmine” posee las características para soportar un proceso de desarrollo basado en “Scrum”?	Sí, pero es necesario instalar extensiones sobre “Redmine” que afectarían otros equipos, y no existe autorización para ello.
¿Tendría usted preferencia por una herramienta en particular?	Sería muy bueno si se pudiera utilizar un tablero físico (en la pared) en lugar de una herramienta virtual, a fin de aprovechar un espacio en la oficina acondicionado para ello.

Tabla 24: Aplicación de la entrevista para evaluar la herramienta “Redmine”

Fuente: Elaboración Propia

6.5. Entrevista para determinar las características deseables en la herramienta

La tabla que se presenta a continuación (Tabla 25: Entrevista para determinar las características deseables en la herramienta), expone un resumen de las respuestas obtenidas de la entrevista grupal realizada para determinar las características deseables en una herramienta que soporte el proceso de desarrollo propuesto.

Pregunta	Respuesta
¿Podría mencionar las características deseables en la herramienta que se utilizaría para soportar el proceso de desarrollo de software?	<ul style="list-style-type: none"> • Que sea gratuita, o que los gastos en los que se incurran sean mínimos • Debe permitir visualizar de manera simple el estado actual del trabajo que está realizando el equipo • Simple, que no conlleve una carga de trabajo extra para las personas que lo utilizan • Atractivo, de manera que los miembros del equipo de trabajo se vean motivados a utilizarla • Debe complementarse de alguna manera con “Redmine”, debido a que dicha herramienta no puede ser reemplazada.

Tabla 25: Entrevista para determinar las características deseables en la herramienta

Fuente: Elaboración Propia

6.6. Bitácora de la simulación del modelo de desarrollo

La tabla que se presenta a continuación (Tabla 26: Bitácora de la simulación del modelo de desarrollo), es la bitácora que se generó al simular el modelo de software propuesto.

Fecha	Descripción
09/05/2015	<p>- Este día se llevó a cabo el “Sprint Planning”. Para el Equipo de Trabajo fue difícil saber a cuanto trabajo comprometerse, debido a que no sabían cuál era su velocidad. Se fijó que la meta iba a ser 8 puntos de dificultad, esto correspondió a 4 “User Stories” con 2 puntos de dificultad cada uno.</p> <p>La cantidad de puntos por “User Story” fue definida por un consenso entre el Product Owner y el “ScrumMaster”, dado que no se cuenta con un proceso de estimación establecido.</p> <p>- Los “User Stories” seleccionadas fueron divididos en tareas, y para cada tarea se estimó un tiempo en horas</p> <p>- El tablero físico fue creado en la pared con los estados definidos para los “User Stories”.</p> <p>- Una vez finalizado el “Sprint Planning”, se procedió a actualizar el tablero físico, agregando los correspondientes “User Stories” del “Sprint”.</p>
10/05/2015	<p>El “Daily Scrum” se llevó a cabo sin mayores complicaciones, cada miembro del equipo dijo su estado.</p> <p>Este día, un desarrollador hizo actualizaciones sobre el tablero físico</p>
11/05/2015	<p>El “Daily Scrum” se llevó a cabo sin mayores complicaciones, cada miembro del equipo dijo su estado; sin embargo, hubo un miembro del equipo que se encontraba bloqueado, debido a que necesitaba aclaración por parte de Spiceworks en un “User Story”.</p>

Fecha	Descripción
12/05/2015	- El "Daily Scrum" se llevó a cabo sin mayores complicaciones - Este día el tablero físico fue actualizado

Fecha	Descripción
12/05/2015	<p>- Este día se preparó la presentación que se iba a realizar en el “Sprint Review”, reunión que fue previamente agendada.</p> <p>- En el “Sprint Review”, un miembro del equipo mostró el trabajo realizado durante el “Sprint”</p> <p>- Se llevó a cabo el “Sprint Retrospective”, reunión que fue agendada con anticipación. De esta reunión se recolectó la siguiente información:</p> <ul style="list-style-type: none"> ✓ El Equipo de Trabajo se sintió a gusto con el proceso de desarrollo, debido a que tenían claro cuál era el objetivo del “Sprint” ✓ Se finalizaron 3 de los 4 “User Stories” definidos en el “Sprint” ✓ El tiempo estimado para las tareas de cada “User Story” estuvo muy por debajo de lo que realmente se tardó en cada tarea ✓ El Equipo de Trabajo mantuvo actualizado el tablero físico en todo momento. ✓ El Equipo de Trabajo sintió que el contar con el tablero físico le da más interacción al equipo, y no lo sintieron como una carga extra ✓ Los “User Stories” no contenían todo el detalle requerido, el Equipo de Trabajo aclaró que este aspecto se puede mejorar ✓ El Equipo de Trabajo resaltó el trabajo realizado por el “ScrumMaster”, debido a que fue muy responsable al organizar

Fecha	Descripción
	<p>los distintos eventos, además ayudó a resolver bloqueos con la mejor disposición</p> <ul style="list-style-type: none"> ✓ El tener un tablero físico hizo posible visualizar el estado del “Sprint” en todo momento ✓ El equipo observó que los estados definidos para los “User Stories” se ajustan a sus necesidades, ven principalmente beneficioso tener un estado que refleje cuando un “User Story” se encuentra en la fase de pruebas internas

Tabla 26: Bitácora de la simulación del modelo de desarrollo

Fuente: Elaboración Propia

7. Bibliografía

- Alfaro, S. A., & Castro, L. C. (2014). *Metodología para la gestión de proyectos Tradicionales y Á giles para el departamento de TI de La Costa Dorada Publicidad*. Instituto Tecnológico de Costa Rica, Area académica de gerencia de proyectos.
- Bernal, C. A. (2010). *Metodología de la investigación* (Tercera ed.). Bogotá D.C.: Prentice Hall.
- Erickson, K. (19 de Setiembre de 2013). *A Comparative Look at Top Agile Tools*. Recuperado el 13 de Mayo de 2016, de CapTech: <https://www.captechconsulting.com/blogs/a-comparative-look-at-top-agile-tools>
- Figuroa, A. P., & Villanueva, H. Z. (2009). *Propuesta de una metodología integral de desarrollo de software para el área de estadística de una institución pública*. Instituto Tecnológico de Costa Rica, Escuela de Computación.
- Hernández, R., Fernández, C., & Baptista, M. (2010). *Metodología de la investigación* (Quinta ed.). México D.F.: McGrawHill.
- Hodzic, E. (3 de Julio de 2012). *PM project-management.com*. Recuperado el 10 de Abril de 2016, de Rally Software Review: Overview – Features – Pricing: <http://project-management.com/rally-software-review/>

Ignacio, D., Herrera, A., & Guillén, J. (2014). *La complejidad en dirección de proyectos: Dimensiones y marcos de trabajo a nivel internacional*. Universidad Politécnica de Madrid, Dpto. de Proyectos y Planificación Rural.

Jeffries, R. (30 de Agosto de 2001). *Essential XP: Card, Conversation, Confirmation*. Recuperado el 12 de Abril de 2016, de Ron Jeffries: <http://ronjeffries.com/xprog/articles/expcardconversationconfirmation/>

Koontz, D. (19 de Noviembre de 2013). *Agile Complexification Inverter*. Recuperado el 20 de Abril de 2016, de Elements of an Effective Scrum Task Board: <http://agilecomplexificationinverter.blogspot.com/2013/11/elements-of-effective-scrum-task-board.html>

Ribeiro, E. (8 de Diciembre de 2015). *Agile Maturity Self-Assessment Survey*. Recuperado el 2 de Mayo de 2016, de Scrum Alliance: <https://www.scrumalliance.org/community/articles/2015/december/agile-maturity-self-assessment-survey>

S.Rubin, K. (2013). *Essential Scrum*. New Jersey: Addison-Wesley.

Salisbury, R. (30 de Noviembre de 2015). *Scrum Alliance*. Recuperado el 15 de Mayo de 2016, de 5 Reasons Why a Physical Scrum Board Rocks: <https://www.scrumalliance.org/community/articles/2015/november/5-reasons-why-a-physical-scrum-board-rocks>

Schaeffer, A. (23 de Setiembre de 2013). *Spiceworks Now Influences More Than \$600 Billion in Annual IT Spend*. Obtenido de Reuters: <http://www.reuters.com/article/2014/09/23/idUSnMKWbnvk7a+1c0+MKW20140923>

Schaeffer, A. (10 de Diciembre de 2014). *Spiceworks Named a Top Ten Best Place to Work in 2015 by Glassdoor*. Obtenido de Spiceworks: <http://www.spiceworks.com/press/releases/2014-12-10/>

Schwaber, K. (2004). *Agile Project Management with Scrum (Developer Best Practices)*. Redmond: Pearson Education.

Shannon, R. E. (1998). *Introduction to the art and science of simulation*. Texas A&M University.

Sliger, M. (13 de Setiembre de 2012). *What Agile Is — And What It Isn't*. (S. Alliance, Productor) Obtenido de Scrum Alliance: <https://scrumalliance.org/community/articles/2012/september/what-agile-is-%E2%80%94-and-what-it-isn%E2%80%99t>

Sutherland, J. (2014). *Scrum: The Art of Doing Twice the Work in Half the Time*. Crown Publishing Group.

VersionOne. (2016). *VersionOne*. Recuperado el 22 de Mayo de 2016, de Pricing & Editions:

https://www.versionone.com/product/lifecycle/pricing-and-editions/?utm_content=content_link

Wester, J. (29 de Setiembre de 2013). *Understanding the Cynefin framework – a basic intro*.

Obtenido de Everyday Kanban:

<http://www.everydaykanban.com/2013/09/29/understanding-the-cynefin-framework/>