

**UNIVERSIDAD NACIONAL DEL ALTIPLANO**  
**FACULTAD DE INGENIERÍA MECÁNICA ELÉCTRICA, ELECTRONICA**  
**Y SISTEMAS**  
**ESCUELA PROFESIONAL DE INGENIERÍA DE SISTEMAS**



**“MODELO DE SISTEMA DE INFORMACIÓN DE REGISTRO Y  
MONITOREO SOCIO AMBIENTAL PARTICIPATIVO DEL PROYECTO  
DE EXPLORACION MINERO CHUCAPACA COMPARANDO LAS  
METODOLOGIAS ÁGILES SCRUM Y KANBAN”**

**T E S I S**

PRESENTADO POR:

**ALFREDO SVEN PALAO VILLASANTE**  
**EDER JOSÉ NUÑEZ DELGADO**

PARA OPTAR EL TÍTULO PROFESIONAL DE:  
**INGENIERO DE SISTEMAS**

PUNO - PERÚ

2014

UNIVERSIDAD NACIONAL DEL ALTIPLANO  
FACULTAD DE INGENIERIA MECANICA ELÉCTRICA, ELECTRONICA  
Y SISTEMAS  
ESCUELA PROFESIONAL DE INGENIERIA DE SISTEMAS

“MODELO DE SISTEMA DE INFORMACIÓN DE REGISTRO Y  
MONITOREO SOCIO AMBIENTAL PARTICIPATIVO DEL PROYECTO DE  
EXPLORACION MINERO CHUCAPACA COMPARANDO LAS  
METODOLOGIAS AGILES SCRUM Y KANBAN”

TESIS PRESENTADA POR:

ALFREDO SVEN PALAO VILLASANTE  
EDER JOSÉ NUÑEZ DELGADO

PARA OPTAR EL TITULO PROFESIONAL DE:  
INGENIERO DE SISTEMAS

APROBADA POR EL JURADO REVISOR CONFORMADO POR:

PRESIDENTE

DR. ANGEL MANUEL OLAZABAL GUERRA

PRIMER MIEMBRO

M. SC. ELMER COYLA IDME

SEGUNDO MIEMBRO

ING ADOLFO CARLOS JIMENEZ CHURA

DIRECTOR

M.SC. CARLOS BORIS SOSA MAYDANA

PUNO - PERÚ

2014

ÁREA: Informática

TEMA: Sistemas de información tradicionales y expertos

## ÍNDICE DE CONTENIDO

	Pág.
INDICE DE FIGURAS.....	ix
INDICE DE TABLAS.....	xii
RESUMEN.....	xiii
INTRODUCCIÓN.....	1
CAPÍTULO I.....	2
PLANTEAMIENTO DEL PROBLEMA, ANTECEDENTES Y OBJETIVOS DE LA INVESTIGACIÓN.....	3
1.1. PLANTEAMIENTO DEL PROBLEMA.....	3
1.1.1. Formulación del problema.....	3
1.1.2. Justificación de la investigación.....	4
1.1.3. Alcances y limitantes.....	5
1.1.3.1. Alcances.....	5
1.1.3.2. Limitantes.....	5
1.2. ANTECEDENTES DE LA INVESTIGACIÓN.....	6
1.1.4. Utilización de software en evaluación de impacto ambiental y monitoreo ambiental.....	6
1.1.5. Identificación y valoración de técnicas ágiles de gestión de proyectos software.....	7
1.3. OBJETIVOS DE LA INVESTIGACIÓN.....	8
1.3.1. Objetivo General.....	5
1.3.2. Objetivos Específicos.....	6
1.4. HIPÓTESIS DE LA INVESTIGACION.....	8
1.4.1. Hipótesis general.....	8
1.4.2. Hipótesis específica.....	9
CAPÍTULO II.....	10
MARCO TEÓRICO Y CONCEPTUAL.....	11
2.1. MARCO TEÓRICO.....	11
2.1.1. Programa de Monitoreo Socio Ambiental Participativo.....	11
2.1.2. Servidores y Base de datos.....	14
2.1.2.1. Apache.....	14
2.1.2.2. MySQL.....	15
2.1.2.3. PHP.....	15

2.1.2.4. Tecnología cliente/servidor.....	16
2.1.3. Metodologías ágiles.....	17
2.1.3.1. Metodología de desarrollo ágil SCRUM.....	17
2.1.3.2. Metodología de desarrollo ágil Kanban.....	39
2.1.4. Relación entre SCRUM y KANBAN.....	47
2.1.5. Modelamiento de datos.....	51
2.2. MARCO CONCEPTUAL.....	52
2.2.1. AMP.....	52
2.2.2. Ancho de banda.....	52
2.2.3. Apache.....	52
2.2.4. Base de datos.....	53
2.2.5. ECA.....	53
2.2.6. EIA.....	53
2.2.7. HTML.....	53
2.2.8. HTTP.....	53
2.2.9. IA.....	54
2.2.10. Kanban.....	54
2.2.11. Lead Time.....	54
2.2.12. LMP.....	54
2.2.13. MySQL.....	55
2.2.14. Navegador Web.....	55
2.2.15. PMA.....	55
2.2.16. PHP.....	55
2.2.17. RUP.....	56
2.2.18. Scrum.....	56
2.2.19. Sprint.....	56
2.2.20. SGA.....	56
2.2.21. WIP.....	56
CAPÍTULO III.....	58
METODOLOGÍA E INSTRUMENTOS.....	59
3.1. LUGAR DE INVESTIGACIÓN.....	59
3.2. TIPO DE INVESTIGACIÓN.....	59
3.3. TÉCNICAS DE RECOLECCIÓN DE DATOS.....	60

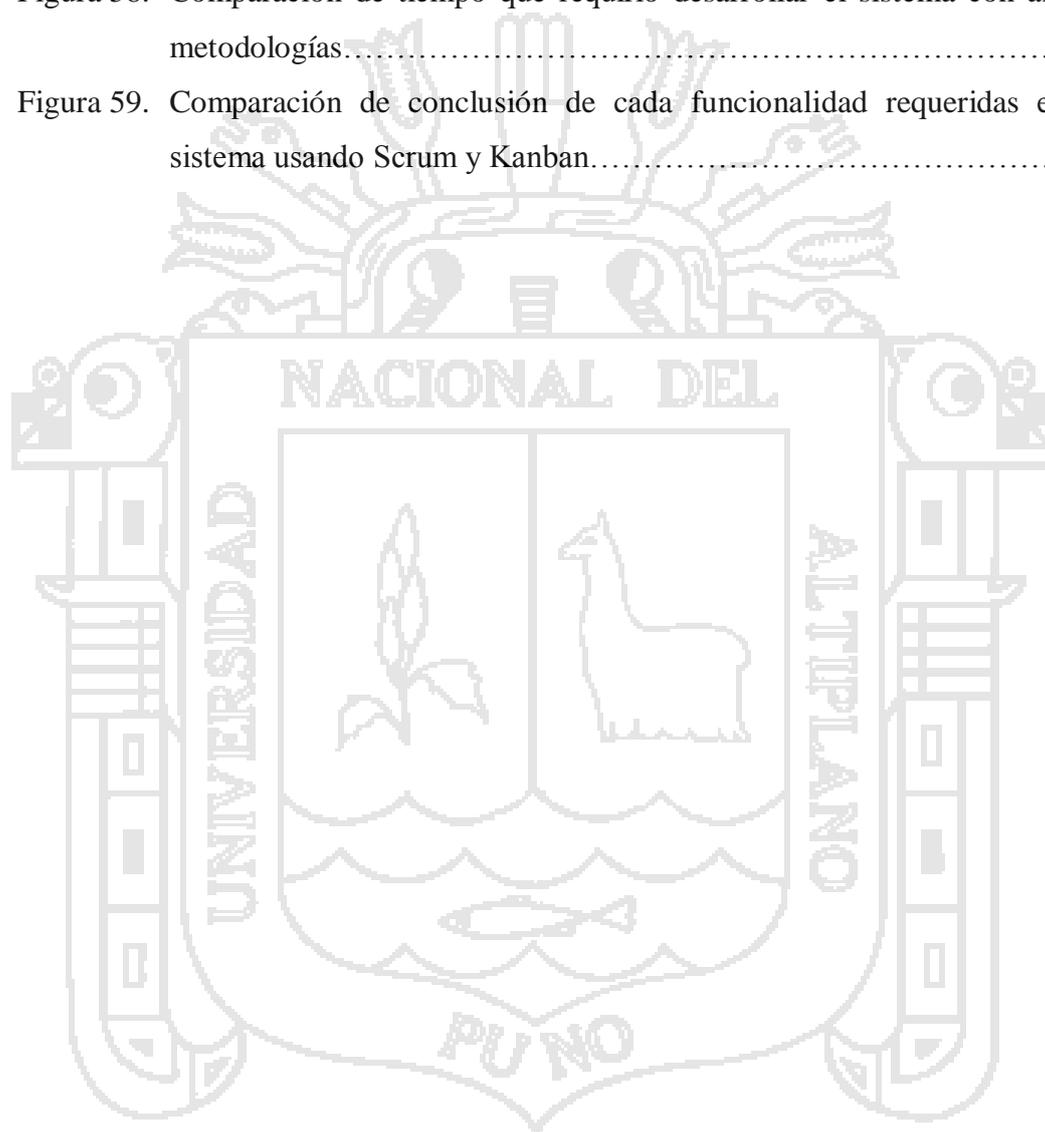
3.3.1. La observación estructurada o formalizada.....	60
3.3.2. El cuestionario.....	60
3.4. METODOLOGÍAS ÁGILES SCRUM Y KANBAN.....	61
3.5. INSTRUMENTOS.....	63
3.6. EVALUACIÓN DE RESULTADOS PROCEDENTES DE LA INVESTIGACIÓN.....	63
CAPÍTULO IV.....	65
EXPOSICIÓN Y ANALISIS DE RESULTADOS.....	66
4.1. APLICANDO SCRUM.....	66
4.1.1. Roles en Scrum.....	66
4.1.2. Pila de Producto.....	66
4.1.3. Reunión de planificación del Sprint 1.....	71
4.1.4. Reunión de Scrum diario del Sprint 1.....	75
4.1.5. Tableros de Desarrollo del Primer Sprint.....	76
4.1.6. Resultado del Sprint 1.....	77
4.1.7. Reunión de planificación del Sprint 2.....	80
4.1.8. Reunión de Scrum diario del Sprint 2.....	84
4.1.9. Tableros de Desarrollo Segundo Sprint.....	85
4.1.10. Resultado del Sprint 2.....	88
4.1.11. Reunión de planificación del Sprint 3.....	92
4.1.12. Reunión de Scrum diario del Sprint 3.....	94
4.1.13. Tableros de Desarrollo tercer Sprint.....	95
4.1.14. Resultado del Sprint 3.....	96
4.2. APLICANDO KANBAN.....	98
4.3. SCRUM COMPARADO CON KANBAN.....	108
CONCLUSIONES.....	113
RECOMENDACIONES.....	115
BIBLIOGRAFÍA.....	116
ANEXOS.....	120

## INDICE DE FIGURAS

	<b>Pág.</b>
Figura 1. Tecnología cliente / servidor.....	17
Figura 2. Proceso Scrum.....	19
Figura 3. Pila de producto.....	24
Figura 4. Estimando horas disponibles.....	29
Figura 5. Estimando horas disponibles.....	30
Figura 6. Estimado de cantidad de trabajo.....	33
Figura 7. Grafica restante del Sprint.....	33
Figura 8. Proceso Kanban.....	41
Figura 9. Comparación escala restrictivo vs adaptativo.....	49
Figura 10. Diferencias Pizarras Scrum y Kanban.....	50
Figura 11. Ejemplo de diagramas E-R.....	51
Figura 12. Flujograma de metodología Scrum.....	62
Figura 13. Flujograma de metodología Kanban.....	62
Figura 14. Se desea una interfaz fácil e intuitiva y una consistente Base de datos.....	76
Figura 15. El administrador desea agregar, quitar o modificar datos de los monitores.....	76
Figura 16. El administrador desea agregar, quitar o modificar datos de ubicaciones del lugar donde se realiza el análisis impacto ambiental.....	77
Figura 17. Se desea que el sistema cuente con una validación (clave).....	77
Figura 18. Interfaz de usuario.....	78
Figura 19. Modelado de Base de datos.....	78
Figura 20. Elemento: “Agregar, quitar, modificar datos de monitores”.....	79
Figura 21. Elemento: “Agregar, quitar, modificar datos de lugares”.....	79
Figura 22. Elemento: “validación (autenticación) para ingresar”.....	80
Figura 23. El monitor desea registrar las fichas de información.....	86
Figura 24. El monitor desea evaluar la información ingresada calificando el estado de alerta que deben tener.....	87
Figura 25. El administrador desea hacerle seguimiento a cada análisis según estado de alerta.....	87
Figura 26. El monitor desea que existan controles de validación en el ingreso de data evitando errores de tipeo.....	87

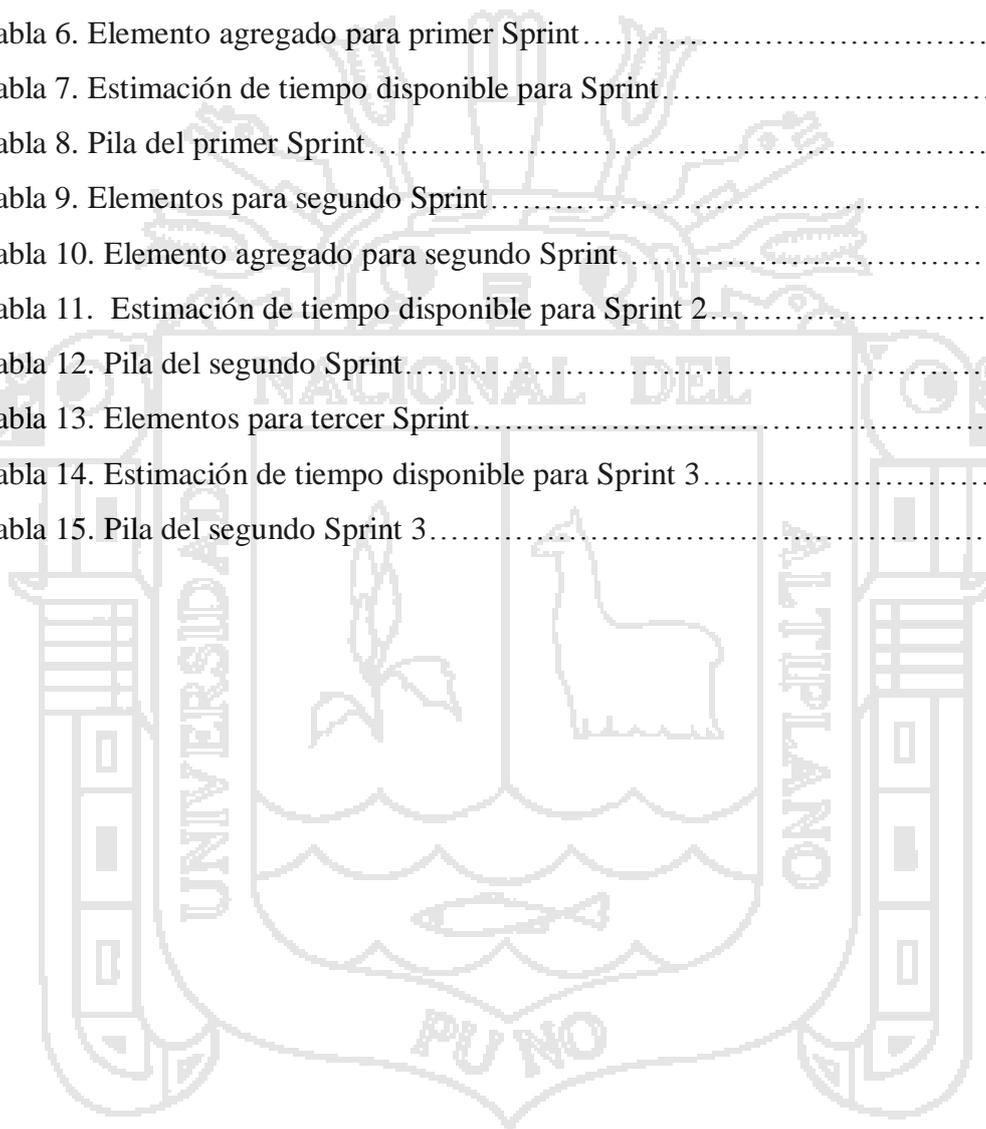
Figura 27. Ingreso de datos Fuentes de Agua.....	88
Figura 28. Ingreso de datos Campamentos.....	89
Figura 29. Ingreso de datos Caminos de Acceso.....	90
Figura 30. Ingreso de datos Almacenamiento de Manejo de Residuos.....	91
Figura 31. El administrador desea generar reportes respecto a la información ingresada.....	95
Figura 32. El administrador desea hacer búsquedas fácilmente.....	95
Figura 33. Se desea levantar las observaciones (réplicas).....	96
Figura 34. Investigar soluciones para generar reportes visuales.....	96
Figura 35. El administrador desea generar reportes respecto a la información ingresada.....	97
Figura 36. El administrador desea hacer búsquedas fácilmente.....	97
Figura 37. Investigar soluciones para generar reportes visuales.....	98
Figura 38. Se desea una interfaz fácil e intuitiva y una consistente Base de datos....	99
Figura 39. El administrador desea agregar, quitar o modificar datos de los monitores.....	99
Figura 40. El administrador desea agregar, quitar o modificar datos de ubicación de lugares donde se realiza el análisis impacto ambiental.....	100
Figura 41. Se desea que el sistema cuente con una validación (clave) para ingresar..	100
Figura 42. Visualización proceso de desarrollo de Sprint Miembro 1.....	101
Figura 43. Visualización proceso de desarrollo de Sprint Miembro 2.....	101
Figura 44. El monitor se desea registrar las fichas de información.....	102
Figura 45. El monitor se desea evaluar la información ingresada calificando el estado de alerta que deben tener.....	103
Figura 46. El administrador desea hacer el seguimiento a cada análisis según estado de alerta.....	103
Figura 47. El monitor desea que existan controles de validación en el ingreso de data evitando errores de tipeo.....	104
Figura 48. Visualización proceso de desarrollo de Sprint Miembro 1.....	104
Figura 49. Visualización proceso de desarrollo de Sprint Miembro 2.....	105
Figura 50. El administrador desea generar reportes respecto a la información ingresada.....	105
Figura 51. El administrador desea hacer búsquedas fácilmente.....	106

Figura 52. El administrador desea levantar las observaciones (réplicas).....	106
Figura 53. Investigar soluciones para generar reportes visuales.....	107
Figura 54. Visualización proceso de desarrollo de Sprint Miembro 1.....	107
Figura 55. Visualización proceso de desarrollo de Sprint Miembro 2.....	108
Figura 56. Resultados Cuestionario tomado antes de la implantación del sistema.....	110
Figura 57. Resultados Cuestionario tomado después de la implantación del sistema.	111
Figura 58. Comparación de tiempo que requirió desarrollar el sistema con ambas metodologías.....	111
Figura 59. Comparación de conclusión de cada funcionalidad requeridas en el sistema usando Scrum y Kanban.....	112



## ÍNDICE DE TABLAS

	Pág.
Tabla 1. Lista de elementos.....	67
Tabla 2. Asignación de prioridades y Estimación de valor.....	68
Tabla 3. Estimación de esfuerzo inicial.....	69
Tabla 4. Pila de producto.....	70
Tabla 5. Elementos para primer Sprint.....	71
Tabla 6. Elemento agregado para primer Sprint.....	72
Tabla 7. Estimación de tiempo disponible para Sprint.....	73
Tabla 8. Pila del primer Sprint.....	73
Tabla 9. Elementos para segundo Sprint.....	80
Tabla 10. Elemento agregado para segundo Sprint.....	81
Tabla 11. Estimación de tiempo disponible para Sprint 2.....	82
Tabla 12. Pila del segundo Sprint.....	82
Tabla 13. Elementos para tercer Sprint.....	92
Tabla 14. Estimación de tiempo disponible para Sprint 3.....	93
Tabla 15. Pila del segundo Sprint 3.....	93



## RESUMEN

El presente estudio se realizó durante los años 2011 – 2012 en la ciudad de Puno en gerencia Programa de Monitoreo Socio Ambiental Participativo (PMSAP) del Proyecto Minero Chucapaca que se desarrolla en la provincia de Ichuña región Moquegua. Con el objetivo de Implementar un sistema de registro y procesamiento de datos del Programa de Monitoreo Socio Ambiental Participativo en el proyecto de exploración minera Chucapaca, comparando las metodologías ágiles Scrum y Kanban. La metodología utilizada fue de investigación de tipo cuantitativo - descriptivo y experimental; para la recolección de la información se realizó un cuestionario con preguntas abiertas, las cuales fueron aplicada a los monitores encargados del Programa de Monitoreo Socio Ambiental Participativo (PMSAP) a pedido del Proyecto Minero Chucapaca. Para implementar un sistema de registro y procesamiento de datos del Programa de Monitoreo Socio Ambiental Participativo en el proyecto de exploración minera Chucapaca, se realizó la secuencia de pasos las metodologías ágiles Scrum y Kanban, para posteriormente comparar las metodologías Scrum y Kanban, adoptando lo mejor de ambas, evaluando el nivel de optimización de los resultados del monitoreo del PMSAP en el proyecto de exploración minera Chucapaca. De los resultados obtenidos se ha llegado a la siguiente conclusión: El sistema de registro y procesamiento de datos del Programa de Monitoreo Socio Ambiental Participativo en el proyecto de exploración minera Chucapaca, comparando las metodologías ágiles Scrum y Kanban, logra optimizar los resultados en tiempo y calidad. El análisis, diseño e implementación permitió poner en funcionamiento las funcionalidades requeridas. La comparación de ambas metodologías aplicado en el proceso de desarrollo del sistema. permitió conocer la medida en que Scrum y Kanban son muy adaptables, pero en términos relativos Scrum es más restrictivo que Kanban Como cualquier herramienta, Scrum y Kanban no son ni perfectas ni completas, son herramientas que se deben juzgar a la herramienta por el éxito del trabajo realizado con ella.

**Palabras clave:** Análisis, Diseño, Kanban, Scrum, Monitoreo ambiental.

## ABSTRACT

This study was conducted during the years 2011 - 2012 in the city of Puno in management Environmental Monitoring Program Partner Participatory (PMSAP) Chucapaca Mining Project that develops in the province of Ichuña Moquegua region. In order to implement a system for recording and data processing Monitoring Program in Environmental Socio Participatory mining exploration project Chucapaca comparing the Agile Scrum and Kanban. The methodology used was quantitative research - descriptive and experimental; for data collection a questionnaire with open questions was conducted, which were applied to the monitors in charge of Environmental Monitoring Program Partner Participatory (PMSAP) on request Chucapaca Mining Project. To implement a system for recording and data processing Partner Program Participatory Environmental Monitoring in mining exploration project Chucapaca, the sequence of steps performed the Agile Scrum and Kanban, later to compare Scrum and Kanban methodologies, adopting best both by assessing the level of optimization of monitoring results PMSAP in mining exploration project Chucapaca. From the results it has come to the following conclusion: The system for recording and data processing Environmental Monitoring Program Partner Participatory in mining exploration project Chucapaca comparing the Agile Scrum and Kanban, optimizes the results in time and quality. The analysis, design and implementation allowed to operate the required functionality. The comparison of both methodologies applied in the process of system development. allowed to know the extent that Scrum and Kanban are very adaptable, but relatively speaking Scrum is more prescriptive than Kanban Like any tool, Scrum and Kanban are neither perfect nor complete, are tools that should judge the tool for the success of work done with it.

**Keywords:** Analysis, Design, Kanban, Scrum, environmental monitoring.

## INTRODUCCIÓN

El proyecto de exploración minera Chucapaca, es ejecutado por la empresa minera Canteras del Hallazgo SAC (CDH), ubicado en la región Moquegua, Provincia de Ichuña; la cual actualmente viene realizando actividades relacionadas a la Evaluación Impacto Ambiental de la zona. Ante esta situación la compañía CDH, ha considerado tener un rol más activo y cercano con la población del lugar, para ello se propuso diseñar e implementar un Programa de Monitoreo Socio Ambiental Participativo (PMSAP).

Un Programa de Monitoreo Socio Ambiental Participativo (PMSAP), es un programa que evalúa posibles impactos sobre las poblaciones y los recursos naturales de un área determinada; está conformado por representantes de las localidades y comunidades campesinas del área de influencia directa e indirecta del Proyecto de Exploración minera. La empresa encargada de llevar dicho PMSAP es “ProNaturaleza”, Fundación Peruana para la Conservación de la Naturaleza, desde sus inicios (1984) ha contribuido a nivel nacional en materia de conservación de la biodiversidad, la promoción del desarrollo sostenible y la promoción de una cultura ambiental.

El proyecto de investigación propone un modelo de sistema que recopile información alimentada por los monitores encargados para el seguimiento del proceso de cada monitoreo; el cual viene realizándose de forma manual, ante esta situación se ha realizado un análisis para obtener resultados rápidos y confiables, además dicha información puede ser objeto de futuros reportes históricos-gráficos y consultas de suma importancia y utilidad para la toma de decisiones gerenciales-ambientales en futuras empresas mineras.



## **PLANTEAMIENTO DEL PROBLEMA, ANTECEDENTES Y**

### **OBJETIVOS DE LA INVESTIGACIÓN**

#### **1.1. PLANTEAMIENTO DEL PROBLEMA**

El Programa de Monitoreo Socio Ambiental Participativo (PMSAP) del proyecto de exploración minera Chucapaca, actualmente viene realizando actividades relacionadas al monitoreo de fuentes de agua, plataformas de perforación, campamentos, caminos de acceso, almacenamiento y manejo de residuos.

El proceso de monitoreo se desarrolla de forma manual, es decir, los monitores previo proceso de capacitación llenan formularios en hojas de papel, indicando las posibles amenazas a la contaminación ambiental, dichas observaciones deben ser subsanadas en un tiempo determinado, dependiendo de la gravedad de la actividad; ello es muy deficiente pues como es sabido un proceso manual de este tipo se enfrenta a muchos riesgos de pérdida de información, confusión y mal registro de datos.

Otro factor que se observa, que dicha información queda archivada y difícilmente consultada para un trabajo multitemporal, pudiendo ser de suma importancia para obtener reportes históricos, gráficos que ayuden a la toma de decisiones gerenciales sobre temas de impacto socio ambiental que genera una exploración minera, siendo en la actualidad un tema muy delicado y de suma importancia en nuestro país.

#### **1.1.1. FORMULACION DEL PROBLEMA**

Frente a esta problemática nos formulamos las siguientes interrogantes:

1. ¿Qué beneficios brindará el modelo de sistema de información de registro y

- monitoreo socio ambiental participativo al proyecto de exploración minera Chucapaca comparando las metodologías ágiles Scrum y Kanban?
2. Dentro de la metodología ágil de Scrum y Kanban, ¿cuál da la mayor versatilidad y eficiencia?
  3. Es posible obtener un modelo integral de estas dos metodologías?

### 1.1.2. JUSTIFICACIÓN DE LA INVESTIGACIÓN

Los sistemas de diagnóstico por recopilación de información a partir de datos, se precisa la ordenación, interacción, correlación progresiva y multidisciplinaria; por ello, el presente proyecto de investigación propone un modelo de sistema que recopile información alimentada por los monitores encargados para el seguimiento del proceso de cada monitoreo; obteniendo resultados más rápidos y confiables, además dicho sistema utilizará esta información para realizar reportes históricos-gráficos y consultas de suma importancia para la toma de decisiones gerenciales-ambientales en futuras empresas mineras.

El sistema se realizará con herramientas de software libre como lo son GNU-AMP, ya que Apache es el servidor web por excelencia, con 65.24% de los servidores de internet confiando en él (Netcraft.com, 2012). Entre sus características más sobresalientes están: fiabilidad, gratuidad y extensibilidad. MySQL es un servidor de bases de datos relacionales muy rápido y robusto, Por su lado PHP es suficientemente versátil y potente como para hacer tanto aplicaciones grandes como pequeñas. Todas ellas sin costo de licenciamiento, lo cual es un gran ahorro en el proceso de implementación.

Y es por ello, que para el desarrollo del sistema de información se propone la utilización de una metodología ágil de desarrollo de software, en este caso se compara

“Kanban” y “Scrum”, muy conocidas por su eficiencia y rapidez, para lo cual el trabajo fue en equipo en una sola metodología, sin embargo esta parte de nuestra investigación ayudará a escoger una de estas metodologías o la combinación de ambas, para la implementación de un sistema de registro y procesamiento de datos del programa de monitoreo socio ambiental participativo en el proyecto de exploración minera Chupaca.

### **1.1.3. ALCANCES Y LIMITANTES**

#### **1.1.3.1. Alcances**

El sistema de registro y procesamiento de datos del Programa de Monitoreo Socio Ambiental Participativo en el proyecto de exploración minera Chupaca, utilizando las metodologías ágiles Scrum y Kanban, permite recopilar información alimentada por los monitores encargados para el seguimiento del proceso de cada monitoreo, emitiendo resultados más rápidos y confiables, dicha información puede ser objeto de futuros reportes históricos-gráficos y consultas de suma importancia y utilidad para la toma de decisiones gerenciales-ambientales en futuras empresas mineras.

Sí también se pretende colaborar con empresas interesadas en el estudio de Impacto ambiental de explotaciones de recursos naturales en general, siendo actualmente un tema muy importante y delicado en nuestro país.

#### **1.1.3.2. Limitantes**

- a) Necesidad de capacitación permanente para el registro y procesamiento de datos la evaluación de impacto ambiental en el Programa de monitoreo Socio Ambiental Participativo.
- b) El sistema será alimentado por los monitores, sin embargo no serán ellos quienes

podrán emitir reportes, sino la empresa dedicada al estudio Impacto ambiental, y por medio de ellas llegar a las empresas mineras propiamente dichas.

## **1.2. ANTECEDENTES DE LA INVESTIGACIÓN**

### **1.2.1. Utilización de software en Evaluación de Impacto ambiental y monitoreo ambiental**

Chávez (2012), en su trabajo de investigación concluye en que los estudios de caracterización de sitios contaminados y de evaluación de riesgos ambientales en salud humana están generalmente limitados por las incertidumbres presentes durante su desarrollo. La cantidad y variedad de información que fue necesaria analizar para llevar a cabo el proceso metodológico propuesto, dependió en gran medida del acervo de información sobre el sitio y las características propias de cada uno de los documentos, en particular, todos aquellos relacionados con información y datos relevantes en materia del impacto y riesgo ambiental del caso de estudio. Se ofrecen críticas/sugerencias de las acciones realizadas o que se pueden realizar para minimizar o superar estas limitaciones e incertidumbres, particularmente en torno a tres aspectos:

- a. Análisis Sistemático de la Información Histórica.
- b. Estimación de riesgos a la salud humana.
- c. Aplicación de Herramientas de Cómputo.

Luna (2007), en su trabajo de investigación logra diseñar un sistema de monitoreo ambiental cuya estación de monitoreo continuo registra 5 tipos de datos: Altura de nivel de agua, pH, Conductividad eléctrica, Total de sólidos disueltos y Total de sólidos en suspensión. La estación ha sido diseñada para la medición de los flujos de agua

superficial de quebradas pequeñas, en zonas de sierra; priorizando las condiciones críticas que dan en épocas de estiaje. El análisis de costo / beneficio del proyecto indica que los impactos de la instalación de estaciones de monitoreo no alteran el bienestar de las comunidades por el contrario los benefician tanto a ellos como a los centros mineros prologando una actividad que asegura un crecimiento continuo a la población.

### **1.2.2. Identificación y valoración de técnicas ágiles de gestión de proyectos software**

Sáez (2013), indica que, tras documentar sobre el porqué del fracaso de los proyectos software, explicar 6 modelos de ciclo de vida, diez metodologías ágiles y otras tantas técnicas ágiles, ver sus ventajas y desventajas, preparar listas de comprobación, formularios y árboles de decisión, se ha llegado a varias conclusiones:

- Los modelos de ciclo de vida y las metodologías presentados, muchas veces son complementarios entre sí. En muchos casos, los paradigmas pueden y deben combinarse de forma que puedan utilizarse las ventajas de cada uno en un único proyecto.
- No se debe establecer un modelo y una metodología por defecto para todos los proyectos. Cada proyecto es un mundo que depende además de otros importantes factores como son el equipo que lo va a desarrollar o el cliente que lo solicita. Aun así sí que hay forma de aproximar cual es aquel modelo y metodología que mejor se adapta a nuestras necesidades.
- La elección de un modelo y una metodología a seguir es un paso crítico en cualquier proyecto software, y así lo deben entender tanto los jefes de proyecto, como el equipo que lo forman, como la organización para la que trabajan.

### **1.3. OBJETIVOS DE LA INVESTIGACIÓN**

#### **1.3.1. Objetivo General**

Implementar un sistema de registro y procesamiento de datos del Programa de Monitoreo Socio Ambiental Participativo en el proyecto de exploración minera Chucapaca, comparando las metodologías ágiles Scrum y Kanban.

#### **1.3.2. Objetivos Específicos**

1. Analizar, diseñar e implementar el sistema de registro y procesamiento de datos del Programa de Monitoreo Socio Ambiental Participativo (PMSAP) en el proyecto de exploración minera Chucapaca.
2. Comparar las metodologías Scrum y Kanban, y obtener un modelo sinérgico o integrado.
3. Evaluar el nivel de optimización de los resultados del monitoreo del PMSAP en el proyecto de exploración minera Chucapaca.

### **1.4. HIPÓTESIS DE LA INVESTIGACION**

#### **1.4.1. Hipótesis general**

La implementación del sistema de información de registro y monitoreo socio ambiental participativo del proyecto de exploración minera Chucapaca comparando las metodologías ágiles Scrum y Kanban optimizará los resultados en tiempo y calidad, además de ser un potencial herramienta de consultas y tratamiento de datos para la toma de decisiones gerenciales - ambientales.

#### 1.4.2. Hipótesis específica

1. El Análisis, diseño e implementación de un sistema de registro y procesamiento de datos del Programa de Monitoreo Socio Ambiental Participativo (PMSAP) en el proyecto de exploración minera Chucapaca, será de manera integral en el tiempo.
2. Las metodologías Scrum y Kanban, se basaran en un modelo integrado.
3. El nivel de optimización de los resultados del monitoreo del PMSAP en el proyecto de exploración minera Chucapaca, es eficiente.





## MARCO TEÓRICO Y CONCEPTUAL

### 2.1. MARCO TEÓRICO

#### 2.1.1. Programa de Monitoreo Socio Ambiental Participativo

Un Programa de Monitoreo Socio Ambiental Participativo (PMSAP), es un programa que evalúa posibles impactos sobre las poblaciones y los recursos naturales de un área determinada. Está conformado por representantes de las localidades y comunidades campesinas del área de influencia directa e indirecta del Proyecto de Exploración minera (Fundación peruana para la conservación de la naturaleza, 2011)

##### **Objetivos del PMSAP:**

- Monitorear el desempeño ambiental y social de la Empresa descrito en el Estudio de Impacto Ambiental en su versión semi detallada.
- Generar información que va a servir para las comunidades y población local acerca de los impactos positivos y negativos de las actividades de exploración.

La Fundación peruana para la conservación de la naturaleza (2011), también da a conocer lo siguiente:

**Monitoreo Ambiental;** Se observa y evalúa los cambios ambientales relacionados con las actividades de Exploración como paisajes, agua, suelo, cultivos y animales.

**Monitoreo Social;** Se observa y evalúa los aspectos socio – culturales relacionados con las actividades de La empresa minera, coordinación con las autoridades y el cumplimiento de compromisos.

**Fines del PMSAP:**

- Disponer de información confiable, a través de la participación comunitaria, sobre los impactos socio ambientales del Proyecto.
- Contar con integrantes de las comunidades locales, dentro de la zona de impacto del Proyecto de Exploración.
- Informar oportunamente a las directivas de las comunidades y poblados sobre los incidentes y accidentes que puedan ocurrir en el ámbito de impacto directo del proyecto.
- Garantizar la transparencia en los temas socio ambiental relacionado con las actividades de la Empresa Minera en la zona.
- Difundir las experiencias de monitoreo en las poblaciones comprometidas.

**Beneficios del PMSAP:**

- Participar en el monitoreo socio ambiental de las actividades del Proyecto de Exploración.
- Alertar a la población para prevenir los posibles impactos de los recursos naturales involucrado en las actividades de Proyecto de Exploración, en cumplimiento de los compromisos asumidos por la empresa ante el Estado y las comunidades locales.

**¿Cómo participar en el PMSAP?**

La Fundación peruana para la conservación de la naturaleza (2011), menciona, que:

- Las comunidades u organizaciones locales están invitadas a participar en el PMSAP a través de sus representantes o monitores comunitarios. La decisión de

participar de la comunidad debe ser voluntaria.

- La comunidad que decide participar elige por mayoría a sus representantes, considerando sus capacidades y conocimientos.
- El representante se convertirá en monitor socio-ambiental y debe estar dispuesto a participar directamente en el PMSAP en representación de su comunidad.
- El monitor deberá cumplir con ciertos requisitos técnicos para realizar la actividad de monitoreo, por ello es importante que participen en los talleres de capacitación organizados por el equipo técnico de ProNaturaleza. Al finalizar la capacitación, las personas capacitadas serán evaluadas para la selección final como monitores socio- ambientales.

#### **Funciones de los Monitores en el PMSAP**

- Asistir a las capacitaciones programadas por el PMSAP. y revisar constantemente los materiales de consulta.
- Informar en asambleas y a la Junta Directiva acerca de sus actividades y los resultados del monitoreo.
- Proponer mejoras ambientales y sociales en el espacio comunal.
- Observar los posibles impactos al ambiente relacionados directamente con las actividades del Proyecto de Exploración (suelo, agua, vegetación, residuos) y registrar los datos en el formulario adecuado.
- Recoger información en asambleas o talleres informativos y hacer entrevistas a las autoridades de la comunidad sobre los posibles impactos sociales (salud, educación, vivienda, actividades culturales) que pudieran estar directamente relacionados con las actividades del Proyecto de Exploración. Esta información se registra en el formulario adecuado.

## 2.1.2. Servidores y Base de datos

### 2.1.2.1. Apache

El servidor HTTP Apache es un servidor Web HTTP de código abierto para plataformas Unix (BSD, GNU/Linux, etc.), Microsoft Windows, Macintosh y otras, que implementa el protocolo HTTP y la noción de sitio virtual. Cuando comenzó su desarrollo en 1995 se basó inicialmente en código del popular NCSA HTTPd, pero más tarde fue reescrito por completo. Su nombre se debe a que Behelendorf quería que tuviese la connotación de algo que es firme y enérgico pero no agresivo, y la tribu Apache fue la última en rendirse al que pronto se convertiría en gobierno de EEUU, y en esos momentos la preocupación de su grupo era que llegasen las empresas y "civilizasen" el paisaje que habían creado los primeros ingenieros de Internet. Además Apache consistía solamente en un conjunto de parches a aplicar al servidor de NCSA. Era, en inglés, a patchy server (un servidor "parcheado").

El servidor Apache se desarrolla dentro del proyecto HTTP Server (httpd) de la Apache Software Foundation. Apache presenta entre otras características altamente configurables, bases de datos de autenticación y negociado de contenido, pero fue criticado por la falta de una interfaz gráfica que ayude en su configuración.

Apache tiene amplia aceptación en la red: desde 1996, Apache, es el servidor HTTP más usado. Alcanzó su máxima cuota de mercado en 2005 siendo el servidor empleado en el 70% de los sitios Web en el mundo, sin embargo ha sufrido un descenso en su cuota de mercado en los últimos años.

La mayoría de las vulnerabilidades de la seguridad descubiertas y resueltas tan sólo pueden ser aprovechadas por usuarios locales y no remotamente. Sin embargo, algunas se pueden accionar remotamente en ciertas situaciones, o explotar por los

usuarios locales malévolos en las disposiciones de recibimiento compartidas que utilizan PHP como módulo de Apache (WIKIMEDIA, 2010).

#### **2.1.2.2. MySQL**

MySQL es un sistema de gestión de base de datos relacional, multihilo y multiusuario con más de seis millones de instalaciones. MySQL AB —desde enero de 2008 una subsidiaria de Sun Microsystems y ésta a su vez de Oracle Corporation desde abril de 2009— desarrolla MySQL como software libre en un esquema de licenciamiento dual.

Por un lado se ofrece bajo la GNU GPL para cualquier uso compatible con esta licencia, pero para aquellas empresas que quieran incorporarlo en productos privativos deben comprar a la empresa una licencia específica que les permita este uso. Está desarrollado en su mayor parte en ANSI C.

Al contrario de proyectos como Apache, donde el software es desarrollado por una comunidad pública y el copyright del código está en poder del autor individual, MySQL es propietario y está patrocinado por una empresa privada, que posee el copyright de la mayor parte del código.

Esto es lo que posibilita el esquema de licenciamiento anteriormente mencionado. Además de la venta de licencias privativas, la compañía ofrece soporte y servicios. Para sus operaciones contratan trabajadores alrededor del mundo que colaboran vía Internet. MySQL fue fundado por David Axmark, Allan Larsson y Michael Widenius (WIKIMEDIA, 2010).

#### **2.1.2.3. PHP**

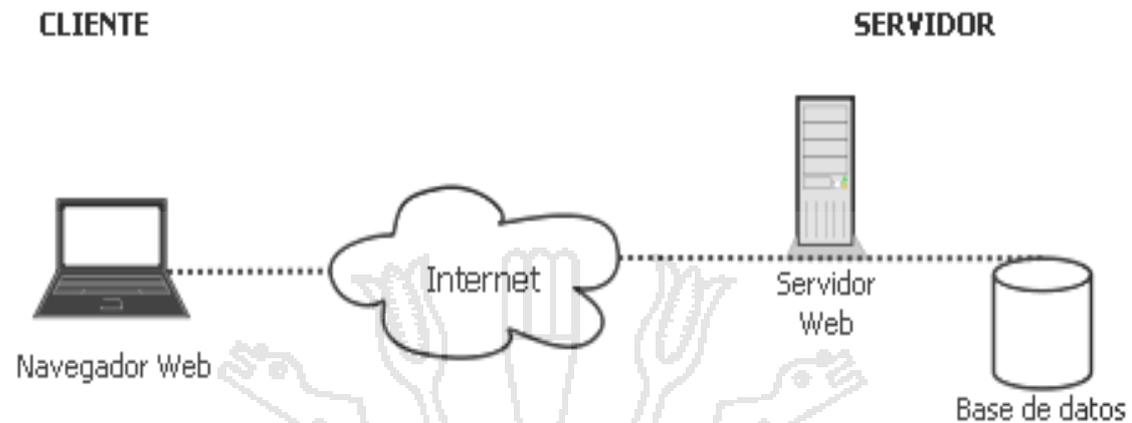
PHP es un lenguaje de programación interpretado, diseñado originalmente para la

creación de páginas Web dinámicas. Es usado principalmente en interpretación del lado del servidor (server-side scripting) pero actualmente puede ser utilizado desde una interfaz de línea de comandos o en la creación de otros tipos de programas incluyendo aplicaciones con interfaz gráfica usando las bibliotecas Qt o GTK+.

PHP es un acrónimo recursivo que significa PHP Hypertext Pre-processor (inicialmente PHP Tools, o, Personal Home Page Tools). Fue creado originalmente por RasmusLerdorf en 1994; sin embargo la implementación principal de PHP es producida ahora por The PHP Group y sirve como el estándar de facto para PHP al no haber una especificación formal. Publicado bajo la PHP License, la Free Software Foundation considera esta licencia como software libre (WIKIMEDIA, 2010).

#### **2.1.2.4. Tecnología cliente/servidor**

En la arquitectura cliente-servidor, una o varias computadoras son los servidores, que responden a un número grande de clientes conectados a través de la red. Los clientes suelen ser computadoras personales de propósito general orientadas al usuario final; en ocasiones, los servidores son intermediarios entre los clientes y otros servidores más especializados (por ejemplo, servidores de bases de datos). Con el auge de Internet, la arquitectura cliente-servidor ha adquirido una mayor relevancia, ya que es el principio básico de funcionamiento de la World Wide Web: un usuario que mediante un visualizador (cliente) solicita un servicio (páginas HTML, etc.) a una computadora que funciona como servidor (Trejo, 2002).

**Figura 1. Tecnología cliente / servidor**

Fuente: Trejo, 2002

### 2.1.3. Metodologías ágiles

#### 2.1.3.1. Metodología de desarrollo ágil SCRUM

Sáez (2013), da a conocer que, la palabra Scrum es un término utilizado en rugby para restablecer la unión después de una interrupción. Scrum es un proceso ágil que se puede usar para gestionar y controlar desarrollos complejos de software y productos usando prácticas iterativas e incrementales, así define un proceso empírico, iterativo e incremental de desarrollo que intenta obtener ventajas respecto a los procesos definidos (cascada, espiral, prototipos, etc.) mediante la aceptación de la naturaleza caótica del desarrollo de software, y la utilización de prácticas tendientes a manejar la impredecibilidad y el riesgo a niveles aceptables. Aunque Scrum estaba previsto que fuera para la gestión de proyectos de desarrollo de software, se puede usar también para la ejecución de equipos de mantenimiento de software o como un enfoque de gestión de programas (Scrum de Scrums).

Scrum es un marco de trabajo iterativo e incremental para el desarrollo de proyectos, productos y aplicaciones. Estructura el desarrollo en ciclos de trabajo llamados Sprints. Son iteraciones de 1 a 4 semanas, y se van sucediendo una detrás de otra. Los Sprints son de duración fija – terminan en una fecha específica aunque no se haya terminado el trabajo, y nunca se alargan. Se limitan en tiempo. Al comienzo de cada Sprint, un equipo multi-funcional selecciona los elementos (requisitos del cliente) de una lista priorizada. Se comprometen a terminar los elementos al final del Sprint. Durante el Sprint no se pueden cambiar los elementos elegidos (KNIBERG, 2007 y Deemer *et al*, 2009).

Scrum es un esqueleto de proceso que incluye un conjunto de prácticas y roles predefinidos. Los roles principales en Scrum son el “ScrumMaster” que mantiene los procesos y trabaja junto con el jefe de proyecto, el “Product Owner” que representa a las personas implicadas en el negocio y el “Team” que incluye a los desarrolladores. Un principio clave de Scrum es el reconocimiento de que durante un proyecto los clientes pueden cambiar sus pensamientos sobre lo que quieren y necesitan, y de que los desafíos que no se pueden predecir no se pueden tratar fácilmente de una forma predictiva o planificada tradicional. Por esto, Scrum adopta un enfoque empírico, aceptando que el problema no se puede entender o definir completamente, centrándose en cambio en maximizar las habilidades del equipo para entregar rápidamente y responder a los requisitos emergentes (Sáez, 2013).

Todos los días el equipo se reúne brevemente para informar del progreso, y actualizan unas gráficas sencillas que les orientan sobre el trabajo restante. Al final del Sprint, el equipo revisa el Sprint con los interesados en el proyecto, y les enseña lo que han construido. La gente obtiene comentarios y observaciones que se puede incorporar al siguiente Sprint. Scrum pone el énfasis en productos que funcionen al final del Sprint

que realmente estén “hechos”; en el caso del software significa que el código esté integrado, completamente probado y potencialmente para entregar. Los roles, artefactos y eventos principales se resumen en la siguiente Figura.

**Figura 2. Proceso Scrum**



Fuente: Deemer *et al*, 2009.

Un tema importante en Scrum es “inspeccionar y adaptar”. El desarrollo inevitablemente implica aprender, innovación y sorpresas. Por eso Scrum hace hincapié en dar un pequeño paso de desarrollo; inspeccionar el producto resultante y la eficacia de las prácticas actuales; y entonces adaptar el objetivo del producto y las prácticas del proceso. Y volver a repetir.

Los tres roles principales, son los siguientes (KNIBERG, 2007 y Deemer *et al*, 2009):

**El Dueño de Producto;** es el responsable de maximizar el retorno de inversión (ROI) identificando las funcionalidades del producto, poniéndolas en una lista priorizada de funcionalidades, decidiendo cuales deberían ir al principio de la lista para el siguiente Sprint, y repriorizando y refinando continuamente la lista. El Dueño de Producto tiene la responsabilidad de las pérdidas y ganancias del producto, asumiendo que es un producto comercial. En el caso de una aplicación interna, el DP no es responsable del ROI en el mismo sentido de un producto comercial (que dará beneficio), pero es responsable de maximizar el ROI en el sentido de elegir - en cada Sprint – los elementos de más valor de negocio y menos coste. En algunas ocasiones el DP y el cliente son la misma persona; esto es muy común en aplicaciones internas. En otras, el cliente podría ser millones de personas con diferentes necesidades, en cuyo caso el rol de DP es parecido al rol de jefe de producto o jefe de marketing del producto que hay en muchas empresas. Sin embargo el Dueño de Producto es diferente al tradicional jefe de producto porque interactúa activa y frecuentemente con el equipo, estableciendo personalmente las prioridades y revisando el resultado en cada iteración de 1 a 4 semanas, en vez de delegar las decisiones de desarrollo en el jefe de proyecto. Es importante destacar que en Scrum hay una persona y sólo una, que hace y tiene la autoridad final de Dueño de Producto.

**El Equipo;** construye el producto que va a usar el cliente, por ejemplo una aplicación o un sitio web. El equipo en Scrum es “multi-funcional” – tiene todas las competencias y habilidades necesarias para entregar un producto potencialmente distribuible en cada Sprint – y es “autoorganizado” (auto-gestionado), con un alto grado de autonomía y responsabilidad. En Scrum, los equipos se auto-organizan en vez de ser dirigidos por un jefe de equipo o jefe de proyecto. El equipo decide a que se compromete, y como hacer lo mejor para cumplir con lo comprometido; en el mundo de

Scrum, al equipo se le conoce como “Cerdos” y a todos los demás como “Gallinas” (que viene de un chiste sobre un cerdo y una gallina que están hablando sobre abrir un restaurante llamado “Huevos con jamón”, y el cerdo no lo ve claro porque “él estaría verdaderamente comprometido, pero la gallina solo estaría implicada”).

El equipo en Scrum consta de siete personas más o menos dos, y para un producto de software el equipo podría incluir analistas, desarrolladores, diseñadores de interface, y testers. El equipo desarrolla el producto y da ideas al DP de cómo hacer un gran producto. En Scrum, el equipo debería estar dedicado al 100% al trabajo en el producto durante el Sprint; intentando evitar hacer varias tareas en diferentes productos o proyectos. A los equipos estables se les asocia con una productividad más alta, así que evita cambiar miembros del equipo. A los grupos de desarrollo de aplicaciones con mucha gente se les organiza en varios equipos Scrum, cada uno centrado en diferentes funcionalidades del producto, coordinando sus esfuerzos muy de cerca. Dado que el equipo hace todo el trabajo (planificación, análisis, programación y pruebas) para una funcionalidad completa centrada en el cliente, a los equipos de Scrum también se les llama equipos por funcionalidades.

**El ScrumMaster;** ayuda al grupo del producto a aprender y aplicar Scrum para conseguir valor de negocio. El ScrumMaster hace lo que sea necesario para ayudar a que el equipo tenga éxito. El ScrumMaster no es el jefe del equipo o jefe de proyecto; el ScrumMaster sirve al equipo, le protege de interferencias del exterior, y enseña y guía al DP y al equipo en el uso fructífero de Scrum. El ScrumMaster se asegura de que todo el mundo en el equipo (incluyendo al DP y la gerencia) entienda y siga las prácticas de Scrum, y ayuda a llevar a la organización, a través de los cambios necesarios y frecuentemente difíciles, a conseguir el éxito con el desarrollo ágil.

Como Scrum hace visibles muchos impedimentos y amenazas a la efectividad del DP y el equipo, es importante tener un ScrumMaster comprometido y que trabaje enérgicamente para ayudar a resolver dichos asuntos, o si no el equipo y el DP tendrán dificultades para tener éxito. Los equipos de Scrum deberían tener un ScrumMaster a tiempo completo, aunque en un equipo más pequeño podría ser un miembro del equipo (llevando una carga de trabajo más ligera). Un gran ScrumMaster puede venir de cualquier experiencia o disciplina previa: ingeniería, diseño, testing, gestión de productos, gestión de proyectos o gestión de calidad.

El ScrumMaster y el Dueño de Producto no pueden ser la misma persona; a veces el ScrumMaster necesitará parar los pies al DP (por ejemplo si intenta meter nuevas funcionalidades en mitad de un Sprint). Y al contrario de un jefe de proyecto, el ScrumMaster no le dice a gente las tareas que tienen asignadas – lo que hace es facilitar el proceso, apoyando al equipo que se organiza y gestiona solo. Si el ScrumMaster tuvo un puesto de gestión en el equipo, necesitará cambiar radicalmente su forma de pensar y el estilo de comunicación con el equipo para tener éxito con Scrum. En el caso de una transición de antiguo jefe a ScrumMaster, es mejor que esté en un equipo diferente al equipo en el que era el jefe, si no habrá un conflicto potencial por las dinámicas sociales y de poder. Ten en cuenta que no existe el rol de jefe de proyecto en Scrum. A veces un (ex-) jefe de proyecto pasa a ser ScrumMaster, esto tiene un historial de éxito variado – hay una diferencia fundamental entre los dos roles en las responsabilidades del día a día y en la mentalidad necesaria para tener éxito. Una buena manera de comprender en profundidad el rol de ScrumMaster y empezar a desarrollar las habilidades necesarias es con los cursos de Certified ScrumMaster de la Scrum Alliance.

Además de estos tres roles, hay otros que contribuyen al éxito del producto, incluyendo los jefes y gestores. Aunque sus roles cambian en Scrum, siguen siendo valiosos. Por ejemplo:

- Ayudan al equipo respetando las reglas y el espíritu de Scrum
- Ayudan a quitar los impedimentos identificados por el equipo
- Ponen su experiencia y conocimiento a disposición del equipo

En Scrum, los jefes cambian el tiempo que dedicaban a hacer de “niñeras” (asignar tareas, pedir informes de estado y otras formas de micro-gestión) por tiempo como “gurus” o “sirvientes” del equipo (mentoring, coaching, ayudar a quitar obstáculos, ayudar a resolver problemas, dar ideas creativas y guiar el desarrollo de habilidades de los miembros del equipo).

Para llevar a cabo este cambio los gestores puede que necesiten cambiar su estilo de gestión; por ejemplo usar cuestionamiento socrático para ayudar al equipo a descubrir la solución a un problema en lugar de simplemente decidir una solución e imponérsela al equipo.

#### a) **Productos esenciales**

##### - **Pila de Producto**

Se describen de forma reducida todos los requerimientos (user stories) priorizados y con una estimación de tiempos. También se muestra información global del desarrollo. La gestión del Product Backlog se puede llevar a cabo mediante una simple hoja de cálculo o a través de herramientas software más completas (Sáez, 2013).

El primer paso en Scrum consiste en que el Dueño de Producto articule la visión del producto. Al final esto evolucionará hacia una lista priorizada de funcionalidades llamada la Pila de Producto. Esta pila de producto existe (y evoluciona) a lo largo de la vida del proyecto; es el plan de trabajo del producto.

**Figura 3. Pila de producto**

Elemento	Detalles (URL wiki)	Prioridad	Estimación del Valor	Estimación de Esfuerzo Inicial	Estimación de Esfuerzo Nueva Lo que queda del Sprint...					
					1	2	3	4	5	6
Como comprador, quiero poner un libro en el carrito de la compra (ver bocetos en el wiki)	...	1	7	5						
Como comprador, quiero quitar un libro del carrito de la compra	...	2	6	2						
Mejorar el rendimiento del procesador de transacciones (ver métricas de objetivos de rendimiento en el wiki)	...	3	6	13						
Investigar soluciones para acelerar la validación de tarjetas de crédito (ver métricas de objetivos de rendimiento en el wiki)	...	4	6	20						
Actualizar todos los servidores a Apache 2.2.3	...	5	5	13						
Diagnosticar y arreglar los errores de los scripts de procesamiento de órdenes (ID bugzilla 14823)	...	6	2	3						
Como comprador, quiero crear y guardar una lista de regalo	...	7	7	40						
Como comprador, quiero añadir o borrar elementos en mi lista de regalo	...	8	4	20						

Fuente: Deemer et al, 2009.

En este punto, la Pila de Producto es la vista única y definitiva de “todo lo que podría ser hecho por el equipo en algún momento, en orden de prioridad”. Solo existe una única Pila de Producto; esto significa que el Dueño de Producto tiene que decidir sobre la priorización de todo el espectro.

La Pila de Producto incluye una serie de elementos, principalmente funcionalidades nuevas del cliente (“permitir que todos los usuarios pongan libros en el carrito de la compra”), pero también objetivos de mejora de ingeniería (“rehacer el módulo de procesamiento de transacciones para hacerlo escalable”), trabajo de investigación o exploratorio (“investigar soluciones para acelerar la

validación de tarjetas de crédito”), y posiblemente, fallos conocidos (“diagnosticar y arreglar los errores de script de procesamiento de órdenes”), si hay solo unos pocos fallos. (Un sistema con muchos errores normalmente tendrá además un sistema de seguimiento de incidencias). A muchos les gusta articular los requisitos usando “historias de usuario”, descripciones claras y concisas de la funcionalidad en términos del valor que aporta al usuario final del producto (KNIBERG, 2007 y Deemer *et al*, 2009).

El subconjunto de la Pila de Producto que va en la entrega actual se conoce como la *Pila de Entrega*, y es en general el objetivo principal del Dueño de Producto.

La Pila de Producto es actualizada continuamente por el Dueño de Producto para reflejar los cambios de necesidad del cliente, nuevas ideas, movimientos de los competidores, dificultades técnicas, etc. El equipo da al Dueño de Producto las estimaciones del esfuerzo requerido para cada elemento de la Pila de Producto. Además, el Dueño de Producto es responsable de asignar una estimación del valor de negocio a cada elemento individual. Esto es normalmente una práctica desconocida para el Dueño de Producto. Por esa razón, a veces el ScrumMaster puede enseñar al Dueño de Producto a hacerlo. Con estas dos estimaciones (esfuerzo y valor) y quizás con estimaciones adicionales de riesgo, el Dueño de Producto prioriza la pila (realmente solo hace falta el subconjunto de la Pila de Entrega) para maximizar el ROI (eligiendo elementos de alto valor con poco esfuerzo) o secundariamente, para reducir algún riesgo importante. Como se verá más adelante, estas estimaciones del valor y esfuerzo se pueden actualizar en cada Sprint a medida que se aprenden cosas nuevas; por tanto, es una actividad continua

de re-priorización de la Pila de Producto que evoluciona constantemente (KNIBERG, 2007 y Deemer *et al*, 2009).

Scrum no dice como estimar la Pila de Producto, pero es común usar estimaciones relativas expresadas como “puntos” en vez de unidades absolutas de esfuerzo tales como personas semana.

#### - **Pila de Sprint (Sprint Backlog)**

Es un subconjunto de la pila de producto. Un documento detallado de los requerimientos (user stories) de cada sprint. Estos se pueden subdividir en tareas (items) de 4-16 horas. El producto de software resultante debe ser potencialmente usable. Para decidir la carga máxima de trabajo para cada sprint se define una puntuación máxima al equipo (trabajo máximo que puede soportar) y una puntuación a cada uno de los requerimientos. Los primeros sprints a desarrollar suelen abordar las historias de usuario o requerimientos más críticos (Saez, 2013).

#### **Planificación del Sprint**

Al comienzo de cada Sprint se hace la Reunión de Planificación del Sprint. Se divide en dos reuniones, la primera de las cuales se llama Planificación del Sprint Parte Uno.

En la Planificación del Sprint Parte Uno, el Dueño de Producto y Equipo (con la facilitación del Scrum Master) revisan los elementos de alta prioridad de la Pila de Producto que el DP está interesado en implementar para este Sprint. Hablan sobre los objetivos y el contexto de dichos elementos de alta prioridad de la Pila de Producto, de tal forma que el Equipo se haga una idea de lo que piensa el Dueño de Producto. El DP y Equipo también revisan la “Definición de Hecho” que todos los

elementos deben cumplir, tales como, “Hecho significa codificado según los estándares, revisado, implementado con desarrollo orientado a pruebas (TDD – test driven development), probado 100% con pruebas automáticas, integrado y documentado”. La Parte Uno se centra en entender qué quiere el DP. Según las reglas de Scrum, al final de la Parte Uno el (siempre ocupado) DP se puede ir aunque debe estar disponible (por ejemplo, por teléfono) durante la siguiente reunión. Sin embargo, son bienvenidos a estar en la Parte Dos (KNIBERG, 2007 y Deemer *et al*, 2009).

La Planificación del Sprint Parte Dos se centra en la planificación detallada de tareas para saber cómo implementar los elementos que el equipo decide hacer. El Equipo selecciona los elementos de la Pila de Producto a las que se comprometen que estará al final del Sprint, comenzando por la parte de arriba de la Pila de Producto (en otras palabras, comenzando con los elementos que tienen más prioridad para el DP) y escribiendo la lista en orden. Esto es una práctica clave en Scrum: El equipo decide a cuanto trabajo se compromete en vez de serles asignado por el Dueño de Producto. Esto hace que el compromiso sea más fiable porque el equipo lo está haciendo basado en su propio análisis y planificación en vez de que venga “hecho” por otros. Aunque el DP no tiene control sobre la cantidad de trabajo que el equipo se compromete a hacer, sabe que los elementos que el equipo coja son de lo alto de la Pila de Producto –en otras palabras, los elementos que ha clasificado como los más importantes. El equipo también tiene autoridad para seleccionar elementos de más abajo de la lista, esto normalmente pasa cuando el equipo y Dueño de Producto observan que algunos elementos de más baja prioridad encajan fácil y apropiadamente entre los de alta prioridad (KNIBERG, 2007 y Deemer *et al*, 2009).

La Reunión de Planificación del Sprint normalmente durará unas cuantas horas –el equipo se está comprometiendo seriamente a completar el trabajo, y este compromiso necesita ser meditado para poder tener éxito. El equipo probablemente empezará la Planificación del Sprint Parte Dos estimando cuánto tiempo tiene cada miembro para trabajo relacionado con el Sprint – en otras palabras, su día laboral medio menos el tiempo que pasarán en reuniones, leyendo el correo, comiendo, etc. Para la mayoría de la gente esto significa 4-6 horas de tiempo disponible al día para trabajo relacionado con el Sprint (KNIBERG, 2007 y Deemer *et al*, 2009).

**Figura 4. Estimando horas disponibles**

Longitud del Sprint		2 semanas	
Días laborables durante el Sprint		8 días	
Miembro del equipo	Días disponibles durante el Sprint*	Horas disponibles por día	Total horas disponibles
Tracy	8	4	32
Sanjay	7	5	35
Phillip	8	4	32
Jing	6	5	30

Fuente: Deemer *et al*, 2009.

Después de haber determinado el tiempo disponible el equipo empieza con el primer elemento de la Pila de Producto –en otras palabras, el elemento de más alta prioridad del DP- y entre todos los dividen en tareas individuales, que se guardan en un documento llamado Pila del Sprint. Como se ha mencionado anteriormente, el DP debe estar disponible durante la Parte Dos (por ejemplo vía telefónica) para cualquier clarificación necesaria. El equipo se irá moviendo secuencialmente hacia abajo de la Pila de Producto de esta forma hasta que se acaben las horas

disponibles. Al final de la reunión el equipo habrá producido una lista de tareas con estimaciones (típicamente en horas o fracciones de un día) (KNIBERG, 2007 y Deemer *et al*, 2009).

Scrum anima a tener trabajadores multi-funcionales en vez de simplemente “trabajar según el nombre del puesto de trabajo” tal como “tester” que solo prueba. En otras palabras, los miembros del equipo “van donde está el trabajo” y ayudan en todo lo posible. Si hay muchas tareas de pruebas, entonces todo el equipo puede ayudar. Esto no significa que todos son generalistas; por supuesto que hay personas entrenadas especialmente en probar (y lo mismo con otras cosas) pero los miembros del equipo trabajan conjuntamente y aprenden nuevas habilidades de los demás (KNIBERG, 2007 y Deemer *et al*, 2009).

Por lo tanto durante la generación de tareas y estimación de la Planificación del Sprint, no es necesario –ni apropiado- que la gente se ofrezca voluntario para las tareas que “pueden hacer mejor”. En vez de eso, es mejor ofrecerse voluntario para una sola tarea a la vez, y cuando hay que coger otra tarea considerar tareas cuyo propósito incluya aprender (quizá haciendo programación en parejas con un especialista). Dicho eso, hay algunas ocasiones excepcionales en las que Juan puede hacer una tarea particular porque a otros miembros del equipo les llevaría mucho o tiempo o sería imposible que lo aprendieran –quizás Juan es la única persona con cualidades artísticas para dibujar. Otros miembros del equipo no podrían dibujar un “muñecote” incluso si su vida dependiera de ello. En este caso excepcional – y si no es excepcional y no se va haciendo infrecuente a medida que el equipo aprende, entonces algo va mal- puede ser necesario preguntarse si las tareas de dibujo planeadas que Juan debe hacer son posibles dentro de un Sprint tan corto (KNIBERG, 2007 y Deemer *et al*, 2009).

**Figura 5. Estimando horas disponibles**

Elemento de la Pila de Producto	Tarea del Sprint	Voluntario	Esfuerzo estimado inicial	Nuevo esfuerzo estimado Lo que queda al final del día...					
				1	2	3	4	5	6
Como comprador quiero poner un libro en el carrito de la compra	modificar base de datos		5						
	crear página web (interfaz de usuario)		8						
	crear página web (lógica Javascript)		13						
	escribir pruebas de aceptación automáticas		13						
	actualizar la página de ayuda del comprador		3						
Mejorar el rendimiento de procesamiento de transacciones	...								
	juntar el código DCP y completar los test del nivel de capa		5						
	completar la orden de máquina para pRank		8						
	Cambiar el DCP y el lector para usar el API http de pRank		13						

Fuente: Deemer *et al*, 2009.

Muchos equipos también usan una herramienta visual de seguimiento de tareas, en forma de un gran tablero de tareas en la pared donde las tareas (escritas en post-its) se cambian durante el Sprint entre columnas etiquetadas con “No empezado”, “En progreso” y “Completado” (KNIBERG, 2007 y Deemer *et al*, 2009).

Uno de los pilares de Scrum es que una vez que el Equipo se compromete, cualquier cambio o adición debe esperar al siguiente Sprint. Esto significa que si a mitad del Sprint el Dueño de Producto decide que hay un nuevo elemento en el que debería trabajar el equipo, no puede hacer el cambio hasta el comienzo del siguiente Sprint. Si aparece alguna circunstancia externa que haga cambiar las prioridades significativamente, y significa que el equipo estaría perdiendo el tiempo si continua con el trabajo actual, el Dueño de Producto o el equipo puede terminar el Sprint. El equipo para, y una nueva reunión de Planificación del Sprint da comienzo a un nuevo Sprint. El trastorno de hacer esto suele ser grande; esto sirve para desincentivar que el Dueño de Producto o el equipo recurran frecuentemente a esta dramática decisión (KNIBERG, 2007 y Deemer *et al*, 2009).

Existe una influencia positiva y poderosa en el hecho de que el equipo esté protegido de cambios en los objetivos durante el Sprint. Primero, el equipo va a trabajar con la certeza absoluta de que sus compromisos no cambiarán, eso refuerza que el equipo se centre en asegurarse que terminan. Segundo, disciplina al Dueño de Producto para que piense seriamente en como prioriza los elementos de la Pila de Producto que da al equipo para el Sprint (KNIBERG, 2007 y Deemer *et al*, 2009).

El Dueño de Producto gana dos cosas siguiendo estas reglas de Scrum. Primero, tiene la confianza de saber que el equipo se ha comprometido a completar un conjunto de trabajo claro y realista que ellos han elegido. Con el tiempo puede convertirse en un equipo muy experto en la elección y el cumplimiento de un compromiso realista. Segundo, el Dueño de Producto puede hacer cualquier cambio que quiera a la Pila de Producto antes de que empiece el siguiente Sprint. En ese momento, adiciones, supresiones, modificaciones y repriorizaciones son posibles y aceptables. Aunque el Dueño de Producto no puede hacer cambios a los elementos seleccionados durante el Sprint actual, está solo a un Sprint de duración o menos de hacer los cambios que desee. Se acabó el estigma en torno al cambio – cambio de dirección, cambio de requisitos, o simplemente cambio de opinión- y puede ser por esta razón que los Dueños de Producto están normalmente tan entusiasmados por Scrum como cualquier otro (KNIBERG, 2007 y Deemer *et al*, 2009).

**Scrum Diario;** Una vez empezado el Sprint, el equipo participa en otra de las prácticas claves de Scrum: El Scrum Diario. Es una reunión corta (15 minutos o menos) que se celebra todos los días a una hora prefijada. Todo el equipo asiste a la reunión. Para hacerla corta, se recomienda que todos estén de pie. Es la oportunidad

del equipo de informar a los demás sobre el progreso y los obstáculos. En el Scrum Diario, cada miembro del equipo, uno por uno, informa sobre tres (y solo tres) cosas a los otros miembros del equipo: (1) Que han hecho desde la última reunión; (2) Que tienen planificado hacer antes de la siguiente reunión; y (3) cualquier bloqueo o impedimento que tengan. Hay que tener en cuenta que el Scrum Diario no es una reunión de estado para informar a un jefe; es tiempo que dedica un equipo auto-organizado para compartir entre sus miembros lo que está pasando, para ayudarles a coordinarse. Alguien anota los bloqueos, y el ScrumMaster se responsabiliza de ayudar a los miembros del equipo a resolverlos. No hay discusiones durante el Scrum Diario, solo se da respuesta a las tres preguntas; si se necesita hablar sobre algo se hace después del Scrum Diario en una reunión de seguimiento, aunque Scrum no dicta que sea obligatorio ir a dicha reunión. Esta reunión de seguimiento es algo común cuando el equipo se adapta a la información que oyeron en el Scrum Diario: en otras palabras, otro ciclo de inspección y adaptación. Normalmente se aconseja que no acudan al Scrum Diario ni jefes, ni gerentes ni otros puestos de autoridad percibida. Esto puede hacer sentir al equipo que está siendo “observado” –con la presión de informar de un gran progreso todos los días (una expectativa poco realista), y cohibe el informar sobre problemas –y tiende a socavar la autogestión del equipo, e invita a la microgestión.

Es más útil que los interesados en el proyecto se acerquen al equipo después de la reunión y se ofrezcan a ayudar con cualquier bloqueo que está haciendo disminuir el progreso del equipo (KNIBERG, 2007 y Deemer *et al*, 2009).

**Actualizando la Pila de Sprint y la Gráfica de Trabajo restante del Sprint;** Todos los días el equipo actualiza sus estimaciones de la cantidad de trabajo que queda para terminar sus tareas actuales en la Pila de Sprint.

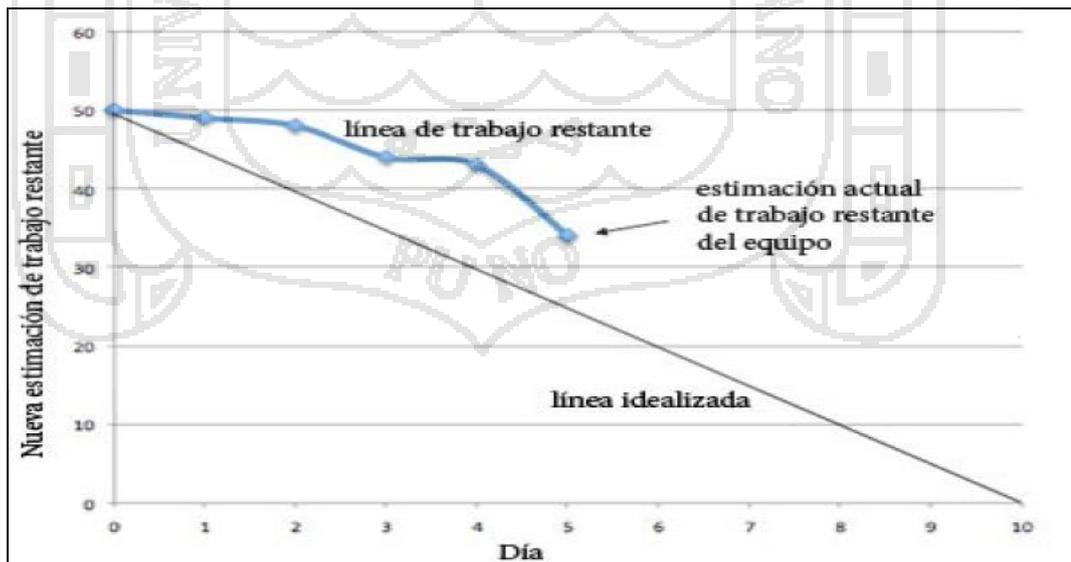
**Figura 6. Estimado de cantidad de trabajo**

Elemento de la Pila de Producto	Tarea del Sprint	Voluntario	Esfuerzo estimado inicial	Nuevo esfuerzo estimado Lo que queda al final del día...					
				1	2	3	4	5	6
Como comprador quiero poner un libro en el carrito de la compra	modificar base de datos	Sanjay	5	4	3	0	0	0	
	crear página web (interfaz de usuario)	Jing	3	3	3	2	0	0	
	crear página web (lógica Javascript)	Tracy & Sam	2	2	2	2	1	0	
	escribir pruebas de aceptación automáticas	Sarah	5	5	5	5	5	0	
	actualizar la página de ayuda del comprador	Sanjay & Jing	3	3	3	3	3	0	
	...								
Mejorar el rendimiento de procesamiento de transacciones	juntar el código DCP y completar los test del nivel de capa		5	5	5	5	5	5	
	completar la orden de máquina para pRank		3	3	8	8	8	8	
	Cambiar el DCP y el lector para usar el API http de pRank		5	5	5	5	5	5	
Total (personas-hora)			50	49	48	44	43	34	

Fuente: Deemer *et al*, 2009.

Después de esta actualización, alguien suma las horas restantes del equipo como un todo, y dibuja una Gráfica de Trabajo Restante del Sprint (KNIBERG, 2007 y Deemer *et al*, 2009).

**Figura 7. Grafica restante del Sprint**



Fuente: Deemer *et al*, 2009.

Esta gráfica muestra cada día una nueva estimación de cuanto trabajo queda (medido en personas-hora) para terminar las tareas del equipo. Idealmente es una gráfica de pendiente descendiente que está en la trayectoria para llegar a “no queda nada de esfuerzo” el último día del Sprint. Y aunque a veces tiene buena pinta, frecuentemente no la tiene; es la realidad del desarrollo de productos. Lo importante es que muestre al equipo el progreso hacia su objetivo, no en términos de cuanto hicimos en el pasado (un hecho irrelevante en término de progreso), sino en término de cuanto trabajo queda en el futuro –lo que separa al equipo de su objetivo. Si la línea de trabajo restante no está bajando hasta la finalización del trabajo cuando se acerca el final del Sprint, entonces el equipo necesita hacer ajustes, por ejemplo reducir el alcance del trabajo o encontrar una forma de trabajar más eficientemente al tiempo que mantienen un ritmo sostenible (KNIBERG, 2007 y Deemer *et al*, 2009).

**Refinamiento de la Pila del Producto;** Una de las directrices menos conocidas pero más útiles de Scrum es que el equipo debe dedicar un cinco o diez por ciento de cada Sprint a refinar o “asear” la Pila de Producto. Esto incluye análisis detallado de requisitos, separar elementos grandes en otros más pequeños, estimación de nuevos elementos, y reestimación de elementos existentes. Scrum no dice nada sobre cómo hay que hacerlo, pero nosotros sugerimos hacer un taller centrado en esto cerca del final del Sprint, para que el equipo y Dueño de Producto se puedan dedicar a este trabajo sin interrupciones. Para un Sprint de dos semanas, cinco por ciento de duración significa que en cada Sprint hay medio día de taller de Refinamiento de la Pila de Producto. Esta actividad de refinamiento no se hace sobre los elementos seleccionados para el Sprint actual; es para los elementos del

futuro, probablemente para el siguiente o el de dentro de dos Sprints. Con esta práctica, la Planificación del Sprint se hace relativamente más sencilla porque el Dueño de Producto y el Equipo empiezan la planificación con un conjunto de elementos claro, bien analizado y cuidadosamente estimado. Una señal de que este taller de refinamiento no se está haciendo (o se está haciendo mal) es que la Planificación del Sprint tenga muchas preguntas, muchos nuevos descubrimientos o mucha confusión (KNIBERG, 2007 y Deemer *et al*, 2009).

**Terminando el Sprint;** Uno de los principios fundamentales de Scrum es que nunca se prolonga la duración del Sprint –termina en la fecha asignada aunque el equipo no haya terminado el trabajo comprometido. Los equipos normalmente se comprometen en exceso en su primer Sprint y no cumplen sus objetivos. Podrían entonces compensarlo al alta y comprometerse por debajo, y terminar antes. Alrededor del tercer o cuarto Sprint, los equipos suelen haber averiguado de que son capaces de entregar (la mayor parte del tiempo), y cumplirán los objetivos del Sprint con más fiabilidad. A los equipos se les recomienda que escojan una duración para el Sprint (por ejemplo dos semanas) y que no la cambien. Una duración consistente ayuda al equipo a saber cuánto pueden hacer, que les ayuda en la estimación y en la planificación de la entrega a más largo plazo. También ayuda a que el equipo coja un ritmo de trabajo; en Scrum a esto se le llama el “pulso” del equipo (KNIBERG, 2007 y Deemer *et al*, 2009).

**Revisión del Sprint;** Cuando termina el Sprint se hace la Revisión del Sprint, donde el equipo revisa el Sprint junto con el Dueño de Producto. A menudo se le llama incorrectamente la “demo” pero esto no capta la intención real de esta

reunión. Una idea clave en Scrum es inspeccionar y adaptar. Ver y aprender lo que está pasando y entonces evolucionar basándose en los comentarios y observaciones, en ciclos repetidos. La Revisión del Sprint es una actividad de inspección y adaptación del producto. Es la oportunidad de que el Dueño de Producto vea lo que está pasando con el producto y con el equipo (o sea, revisión del Sprint); y la oportunidad del equipo de saber cómo va el Dueño de Producto y el mercado. Por lo tanto el elemento más importante de la Revisión es una conversación en profundidad entre el equipo y el DP para conocer la situación, recibir consejo, etc. La revisión incluye una demo de lo que el equipo construyó durante el Sprint, pero si el centro de la revisión es la demo en vez de la conversación, estará desbalanceado (KNIBERG, 2007 y Deemer *et al*, 2009).

Una directriz útil –aunque frecuentemente pasada por alto- es que el Scrum Master es responsable de saber la “Definición de Hecho” que fue definida durante la Planificación del Sprint, y durante esta reunión es responsable de decir al Dueño de Producto si alguno de los elementos implementados por el equipo no cumple esta definición. De esta forma, hay una mayor visibilidad de la calidad del trabajo; los equipos no pueden simular la calidad presentando software que parece que funciona bien, pero que está implementado con código desordenado, sin probar y de mala calidad. En esta reunión están presentes el DP, los miembros del Equipo, y el ScrumMaster, además de los clientes, gente interesada en el proyecto, expertos, directivos y cualquier otra persona que lo desee. La parte de demo de la Revisión del Sprint no es una “presentación” que da el equipo – no hay diapositivas. Una guía de Scrum es que no se debería dedicar más de 30 minutos a preparar la demo, si no quiere decir que algo va mal en el trabajo del equipo. Es simplemente una

demo de lo que se ha construido. Cualquiera es libre de preguntar y dar su opinión (KNIBERG, 2007 y Deemer *et al*, 2009).

**Sprint de Entrega;** La visión de perfección de Scrum es que el producto es potencialmente entregable al final de cada Sprint, lo que implica que no se necesita trabajos de acabado tales como pruebas o documentación. Por contra, implica que todo está completamente terminado en cada Sprint; y se podría realmente empaquetar o desplegar inmediatamente después de la Revisión del Sprint (KNIBERG, 2007 y Deemer *et al*, 2009).

Sin embargo, muchas organizaciones tienen prácticas de desarrollo débiles y no pueden alcanzar esta visión de perfección, o hay otras circunstancias atenuantes (por ejemplo que “la máquina se estropeó”). En este caso, quedará algo de trabajo por hacer, por ejemplo las últimas pruebas de integración del entorno producción, y en este caso se necesitará un “Sprint de Entrega” para terminar este trabajo pendiente (KNIBERG, 2007 y Deemer *et al*, 2009).

Hay que tener en cuenta que la necesidad de un Sprint de Entrega es una señal de alguna debilidad; lo ideal es que no sea necesario. Si fuera necesario, los Sprints continúan hasta que el Dueño de Producto decide que el producto está casi listo para entregar, en este punto habrá un Sprint de Entrega para tenerlo listo para el lanzamiento. Si el equipo ha seguido buenas prácticas de desarrollo, con refactorización e integración continua, y pruebas efectivas durante cada Sprint, debería ser necesario poco trabajo de estabilización y acabado antes de la entrega (KNIBERG, 2007 y Deemer *et al*, 2009).

### **Planificación de la entrega y Refinamiento de la Pila de Producto inicial;**

Una pregunta que la gente se hace a veces es cómo en un modelo iterativo se puede hacer planificaciones de entregas a largo plazo. Hay dos casos a considerar: (1) un producto nuevo en su primera entrega, y (2) un producto ya existente en una entrega posterior (KNIBERG, 2007 y Deemer *et al*, 2009).

En el caso de un producto nuevo, o un producto existente que acaba de adoptar Scrum, hay que refinar la Pila de Producto inicial antes del primer Sprint, el Dueño de Producto y el equipo dan forma a la Pila de Producto de Scrum. Podría llevar unos días o una semana, e implica un taller sobre la visión del producto, algo de análisis detallado de requisitos, y la estimación de todos los elementos identificados para la primera entrega (KNIBERG, 2007 y Deemer *et al*, 2009).

Es sorprendente en Scrum que en el caso de un producto establecido con una Pila de Producto establecida no se debería necesitar ninguna planificación especial o extensiva para la siguiente entrega. ¿Por qué? Porque el Dueño de Producto y el equipo debería estar refinando la Pila de Producto en cada Sprint (cinco o diez por ciento del tiempo de cada Sprint), preparándose continuamente para el futuro. Este modo de desarrollo de producto continuo evita la necesidad de las etapas dramáticas de preparar-ejecutar-concluir que se ven en desarrollos tradicionales de ciclo de vida secuencial (KNIBERG, 2007 y Deemer *et al*, 2009).

Durante el taller de refinamiento inicial de la Pila de Producto y durante los refinamientos continuos de la pila de cada Sprint, el Equipo y el Dueño de Producto planificarán la entrega, perfeccionarán las estimaciones, prioridades y contenido a medida que van aprendiendo.

Algunas entregas tienen una fecha fijada; por ejemplo: “Entregaremos la versión 2.0 de nuestro proyecto en la feria del 10 de noviembre”. En esta situación,

el equipo completará todos los Sprints posibles (con todas las funcionalidades posibles) en el tiempo disponible. Otros productos necesitan tener ciertas funcionalidades antes de estar terminados y el producto no se podrá lanzar hasta que cumpla dichos requisitos, independientemente del tiempo que lleve. Como Scrum enfatiza el producir código potencialmente entregable en cada Sprint, el Dueño de Producto puede optar por empezar a hacer entregas internas, para permitir que el cliente pueda beneficiarse del trabajo completado lo antes posible (KNIBERG, 2007 y Deemer *et al*, 2009).

Como no es posible saber todo de antemano, la atención se centra en crear y perfeccionar un plan para darle una dirección amplia a la entrega, y clarificar como se tomarán las decisiones (por ejemplo alcance frente a fechas de entrega). Piensa en esto como si fuera la guía que te lleva hasta el destino final; los caminos exactos que tomarás y las decisiones que harás durante el viaje se pueden llevar a cabo durante el trayecto. La mayoría de DP eligen el enfoque de una entrega. Por ejemplo, elegirán una fecha de entrega y trabajarán con el equipo para estimar los elementos de la Pila de Entrega que se podrán tener para esa fecha. En situaciones de compromisos de “precio fijo/fecha fija/funcionalidad fija”, al menos uno de estos parámetros tiene que tener un margen para permitir incertidumbre y cambio; en esto Scrum no difiere de otros enfoques (KNIBERG, 2007 y Deemer *et al*, 2009).

### **2.1.3.2. Metodología de desarrollo ágil Kanban**

La metodología de desarrollo Kanban (Kanban Software Development) está basada en la metodología de fabricación industrial del mismo nombre. Su objetivo es

gestionar de manera general como se van completando tareas, pero en los últimos años se ha utilizado en la gestión de proyectos de desarrollo software (Sáez, 2013).

Kanban es un sistema que controla el flujo de recursos en procesos de producción a través de tarjetas, las cuales son utilizadas para indicar abastecimiento de material o producción de piezas, está basada en la demanda y consumo del cliente, y no en la planeación de la demanda. Puede entenderse también, como un sistema de producción que determina el flujo de materiales a través de señales que indican cuando debe producirse un bien o producto y cuando debe reabastecerse de materias primas entre dos centros de trabajo que son consecutivos (Getting started, 2012).

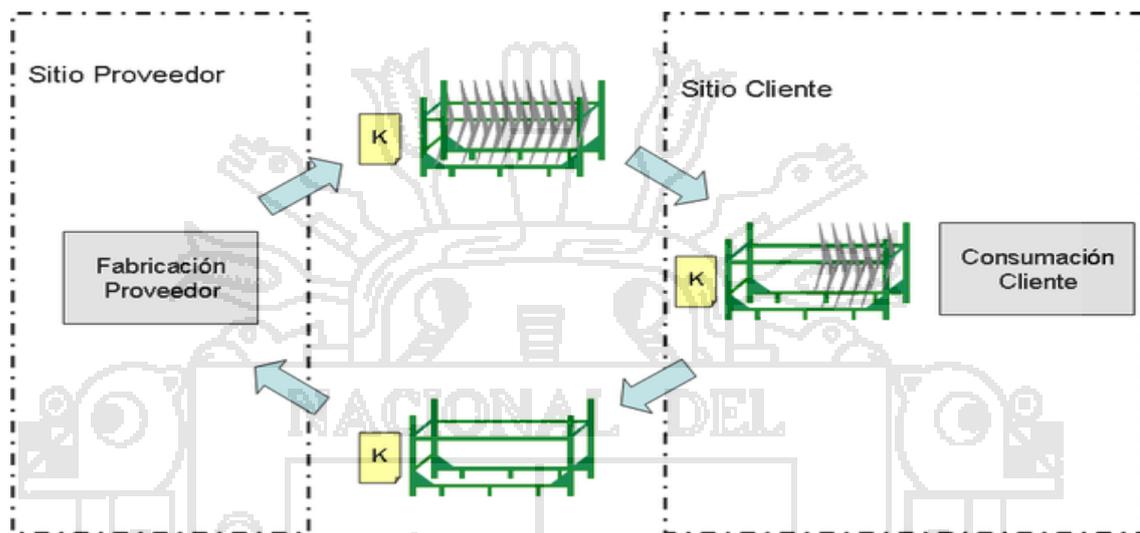
En los últimos años, los sistemas Kanban se han vuelto un tema de candente de creciente interés en la comunidad ágil, gracias particularmente a la excelente introducción que Tom y Mary Poppendieck realizaron sobre los principios propuestos por Lean aplicados al desarrollo de software. Dentro de esta línea se pueden destacar tres personas que han estado trabajando con sistemas Kanban dentro del marco de desarrollo ágil: David Anderson, Arlo Belshee y Kenji Hiranabe (Joyce, 2009).

a) **Funcionamiento**

Cuando un cliente retira productos de su lugar de almacenamiento, el Kanban, o la señal, viaja hasta el principio de la línea de fabricación o de montaje, para que se produzca un nuevo producto. Se dice entonces que la producción está guiada por la demanda y que el Kanban es la señal que el cliente indica para que un nuevo producto deba ser fabricado o montado para rellenar el punto de stock. Funcionando sobre el principio de los flujos "pull" (el cliente "apela" o "pide" el producto), el primer paso es definir la cantidad ideal de productos que hay que entregar, suficientemente grande para

permitir la producción, y no demasiado pequeño como para reducir las existencias (COMUNIDAD CRISP, 2010).

**Figura 8. Proceso Kanban**



Fuente: COMUNIDAD CRISP, 2010.

Kanban es un sistema basado en señales. Como su nombre sugiere, Kanban históricamente usa tarjetas para señalar la necesidad de un artículo. Sin embargo, otros dispositivos como marcadores plásticos, pelotas, o un carro vacío de transporte también pueden ser usados para provocar el movimiento, la producción, o el suministro de una unidad en una fábrica.

El sistema Kanban fue inventado debido a la necesidad de mantener el nivel de mejoras por la Toyota. Kanban se hizo un instrumento eficaz para apoyar al sistema de producción en total. Además, demostró ser una forma excelente para promover mejoras, porque al restringir el número de Kanban en circulación se destacan las áreas con problemas (COMUNIDAD CRISP, 2010).

**b) Reglas para la implantación de las tarjetas Kanban**

Para conseguir el propósito de la producción Kanban deben cumplirse las siguientes reglas (COMUNIDAD CRISP, 2010):

**Regla 1. El proceso posterior recogerá del anterior los productos necesarios en las cantidades precisas del lugar y momento oportuno.**

Se crea una pérdida si el proceso anterior abastece de partes y materiales al proceso subsiguiente en el momento que éste no los necesita o en una cantidad mayor a la que necesita. La pérdida puede ser muy variada, incluyendo la pérdida por el exceso de tiempo extra, pérdida en el exceso de inventario, y pérdida en la inversión de nuevos proyectos sin saber que la existente cuenta con la capacidad suficiente. La peor pérdida ocurre cuando los procesos no pueden producir lo que realmente es necesario, cuando éstos están produciendo lo que no es necesario.

Para asegurarse de que se cumpla esta regla:

- Se prohibirá cualquier retirada de piezas o elementos sin la correspondiente utilización del Kanban.
- Se prohibirá la retirada de piezas o elementos en cantidades mayores que las especificadas en los Kanban.

**Regla 2. El proceso precedente deberá fabricar sus productos en las cantidades recogidas por el proceso siguiente.**

Por tal motivo se prohíbe una producción mayor que el número de tarjetas Kanban. Por otra parte, cuando en un proceso anterior hayan de producirse varios tipos

de piezas, su producción deberá seguir la secuencia con que se han entregado los diversos tipos de Kanban.

Puesto que el proceso siguiente requerirá unidades únicas o lotes de tamaño reducido a fin de conseguir el nivelado de la producción, el proceso anterior deberá llevar a cabo frecuentes preparaciones de máquina según los requerimientos asimismo frecuentes del proceso posterior, preparaciones que habrán de realizarse con la mayor rapidez mediante la implementación del sistema SMED, ya comentado en apartados anteriores.

**Regla 3. Los productos defectuosos nunca deben pasar al proceso siguiente.**

El incumplimiento de esta regla comprometería la existencia misma del sistema Kanban. Si llegaran a identificarse en el proceso siguiente algunos elementos defectuosos, tendría lugar una parada de la línea, al no tener unidades extras en existencia y devolvería los elementos defectuosos al anterior proceso. La parada de la línea del proceso siguiente resulta obvia y visible para todos. El sistema se basa pues en la idea de autocontrol siendo su propósito el evitar la repetición de defectos.

El sentido del término defectuoso comprende asimismo las operaciones defectuosas, aquellas que no responden por completo a la estandarización y que suponen ineficiencia en las operaciones manuales, en las rutas o en los tiempos de trabajo. Tales ineficiencias son con frecuencia causa de que se produzcan elementos a su vez defectuosos. Así pues, las operaciones defectuosas deben eliminarse, a fin de asegurar un ritmo continuo en los pedidos.

**Regla 4. El número de Kanban debe minimizarse.**

Puesto que el número de Kanban expresa la cantidad máxima de existencias de un determinado elemento, habrá que mantenerse tan pequeño como sea posible. La autoridad final para modificar el número de Kanban se delega en el supervisor de cada proceso. Si un proceso se perfecciona gracias a la disminución de tamaño del lote y al acortamiento del plazo de fabricación será posible disminuir a su vez el número de Kanban necesarias. La delegación de autoridad para determinar el número de Kanban es el primer paso para promover el perfeccionamiento de las capacidades directivas. El número de Kanban a minimizar viene determinado por la expresión:  $N^{\circ}$  de Kanban=(Demanda media tiempo de terminación+Stock de seguridad)/(Capacidad de los contenedores)

**Regla 5. El Kanban habrá de utilizarse para lograr la adaptación a pequeñas fluctuaciones de la demanda.**

Con ello hacemos mención al rasgo más notable del sistema Kanban consistente en adaptarse a los cambios repentinos en los niveles de demanda o de las exigencias de la producción.

**Regla 6. Equilibrar la producción.**

Para que podamos producir solamente la cantidad necesaria requerida por los procesos subsiguientes, se hace necesario para todos los procesos mantener al equipo y a los trabajadores de tal manera que puedan producir materiales en el momento necesario y en la cantidad necesaria. En este caso si el proceso subsiguiente pide material de una manera discontinua con respecto al tiempo y a la cantidad, el proceso anterior requerirá personal y máquinas en exceso para satisfacer esa necesidad. Es aquí

cuando es más fácil apreciar los componentes básicos del sistema Kanban, que son los siguientes:

- Equilibrio, sincronización y flujo.
- Calidad: "Hacerlo bien la primera vez".
- Participación de los empleados.

### **Escenarios donde aplica**

Una vez consideradas las reglas de operación anteriores es posible pasar a la etapa de implementación. Como segundo paso es necesario identificar los requerimientos necesarios que se deben cumplir, para iniciar con la puesta en marcha. Es importante desde el inicio conocer en qué tipo de escenarios KANBAN proporciona los resultados esperados, ya que no siempre KANBAN puede resolver los problemas presentados en la línea de producción. Es altamente efectivo en ambiente donde el producto tiene un alto volumen de ventas, la complejidad de producción es baja y principalmente en aquellos ambientes en donde el proceso de producción se encuentra balanceado. De no ser así algunos proyectos de implementación pueden tener como etapa inicial, llevar a cabo la re-ingeniería del toda la Cadena de Suministro.

### **Requerimientos para la implementación**

Definir cuál será (Getting started with KANBAN, 2012):

- La estrategia de Administración, que puede ser sobre la base de una metodología visual o asistida por computadora.
- La estrategia de punto de re-orden:
  - o Punto de re-orden
  - o Máximos y mínimos

- Cantidad económica
- Stock de seguridad

Definir cuales componentes aplica debido al tipo de producto que se fabrica.  
Definir el sistema KANBAN, en donde se establezcan las posiciones de los centros de trabajo, distancias, tiempos de entrega, procesos precedentes y procesos subsecuentes.

Establecer los objetivos que KANBAN pretende resolver, cuando será alcanzado y las medidas de desempeño necesarias para medir el proceso. Monitoreo y ajustes, una vez identificados los KPI es recomendable realizar mediciones con el propósito de hacer los ajustes pertinentes de forma que se adicionen más contenedores, se definan nuevos espacios para almacenaje de partes y/o materia prima, o se tengan que establecer nuevos acuerdos con los proveedores externos ya que en algunos casos estos pueden intervenir de manera directa en esta filosofía de administración y flujo de materiales.

### **Beneficios**

Entre los principales beneficios se encuentran (Getting started with KANBAN, 2012):

- Reduce inventarios y obsolescencia de productos

Debido a que un material no es entregado hasta que es producido, provoca que se reduzcan las necesidades de espacio. Si el material sufre una actualización de diseño, el producto es entregado al siguiente proceso considerando las actualizaciones en diseño.

- Reduce desperdicios y basura

Al igual que en el punto anterior, debido a que los productos son entregados hasta que son requeridos, fomenta que no haya sobre producción, lo cual hace que se eliminen costos de almacenamiento.

- Provee flexibilidad en la producción

La forma en la que están dispuestas las líneas de producción facilita adaptarse a los cambios en la demanda del producto, ya sea por cambios en el diseño o por cambios en los requerimientos del cliente.

- Reduce el costo total

Esto se da como consecuencia de los anteriores, al no tener sobre producción, al tener unidades de producción flexibles, al minimizar stocks de seguridad, al reducir tiempos de espera todo ello conlleva a una reducción del costo total.

- Puede obtener una cuantificación de valor.

Esto se da cuando en el sistema Kanban se realiza previamente un estudio en la asignación de las cantidades requeridas para el llenado de los almacenes de productos en proceso. Si conocemos los pasos y las cantidades exactas del material requerido para las fases finales de Producción y Ensamble los almacenes de productos en proceso pueden tener asignado un valor interno para la Empresa, aunque su venta no sea posible en ese momento al no estar terminado el producto, ésta medición es posible cuando tenemos un control que el sistema Kanban puede proporcionarnos.

#### **2.1.4. Relación entre SCRUM y KANBAN**

Kniberg y Skarin (2010), indica lo siguiente:

Scrum y Kanban son ambas herramientas de proceso:

Herramienta = cualquier medio para realizar una tarea o propósito

Proceso = cómo trabajas.

Scrum y Kanban son herramientas de proceso que te ayudan a trabajar más eficazmente, en cierta medida, diciéndote qué hacer. Java también es una herramienta, te ofrece una forma más sencilla de programar una computadora. Un cepillo de dientes también es una herramienta, te ayuda a llegar a tus dientes para que puedas limpiarlos.

“Compara herramientas para entender, no para juzgar”

Como cualquier herramienta, Scrum y Kanban no son ni perfectas ni completas. No te dicen todo lo que tienes que hacer, solo proporcionan ciertas restricciones y directrices. Por ejemplo, Scrum te obliga a tener iteraciones de duración fija y equipos interdisciplinarios, y Kanban te obliga a usar tableros visibles y a limitar el tamaño de tus colas.

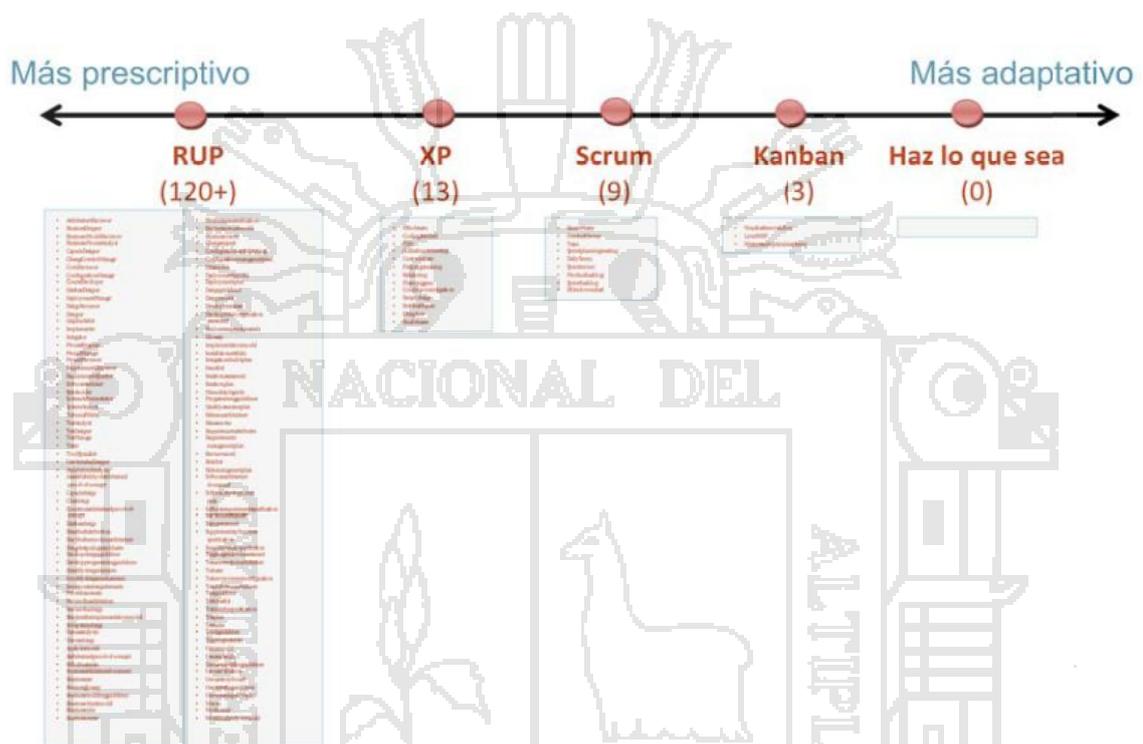
Curiosamente, el valor de una herramienta es que limita tus opciones. Una herramienta de proceso que te permite hacer cualquier cosa no es muy útil. Podríamos llamar a ese proceso "Hacer lo que sea" o qué tal "Hacer lo correcto". El proceso "Hacer lo correcto" garantiza que funciona, ¡es una bala de plata! Porque si no funciona, es obvio que no estabas siguiendo el Proceso. Usar las herramientas adecuadas te ayudará a triunfar, pero no garantizará el éxito. Es fácil confundir el éxito/fracaso del proyecto, con el éxito / fracaso de la herramienta.

- Un proyecto puede triunfar debido a una gran herramienta.
- Un proyecto puede triunfar a pesar de una pésima herramienta.
- Un proyecto puede fallar debido a una pésima herramienta.

- Un proyecto puede fallar a pesar de una gran herramienta.

Vamos a comparar algunas herramientas de proceso más en la escala restrictivo vs adaptativo:

**Figura 9. Comparación escala restrictivo vs adaptativo**



Fuente: Kniberg y Skarin, 2010.

RUP es bastante restrictivo- tiene más de 30 perfiles, más de 20 actividades, y más de 70 artefactos, una cantidad enorme de cosas a aprender. Realmente no se espera que uses todos los que hay, si bien, se supone que seleccionarás un subconjunto adecuado para tu proyecto. Lamentablemente, esto parece ser difícil en la práctica. ¿Necesitaremos artefactos para la Configuración de auditoría de los resultados? ¿Necesitaremos un perfil de gerente de Control de cambios? “No estoy seguro, así que mejor los mantenemos por si acaso”. Esto puede ser una de las razones

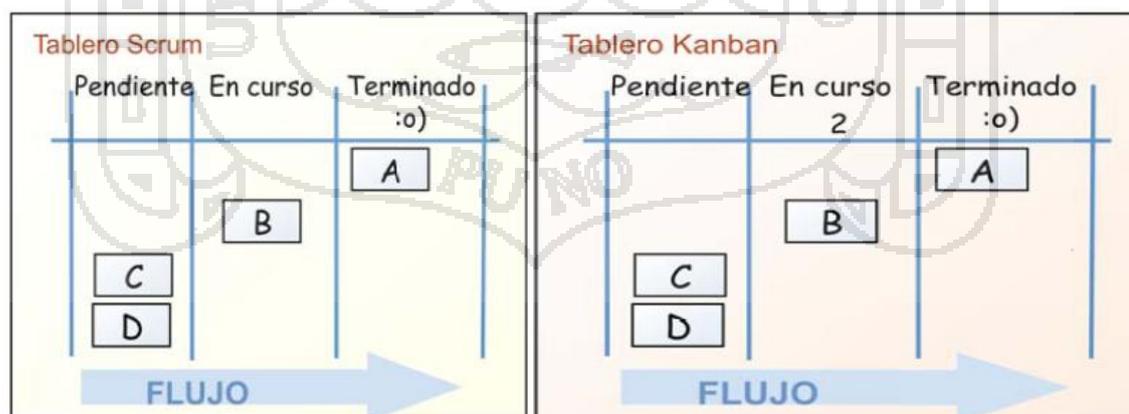
por las que las implementaciones de RUP suelen ser bastante pesadas en comparación con los métodos ágiles como Scrum y XP.

XP (eXtreme Programming) es más restrictivo en comparación con Scrum. Se incluye la mayoría de Scrum más un montón de buenas prácticas específicas de ingeniería, tales como desarrollo dirigido por pruebas y la programación en parejas.

Scrum es menos restrictivo que XP, ya que no establece ninguna práctica específica de ingeniería. Sin embargo, Scrum es más restrictivo que Kanban ya que prescribe cosas como iteraciones y equipos interdisciplinarios.

Una de las principales diferencias entre Scrum y RUP es que en RUP tienes demasiado, y se supone que quitarás aquello que no necesitas. En Scrum tienes demasiado poco, y se supone que añadirás el material que falta. Kanban deja casi todo abierto. Las únicas normas son Visualiza tu Flujo de trabajo y Limita tu WIP (Work In Progress, Trabajo en curso). A pocos centímetros de “Haz lo que Sea”, pero sigue siendo sorprendentemente poderoso.

**Figura 10. Diferencias Pizarras Scrum y Kanban**



Fuente: Kniberg y Skarin, 2010.

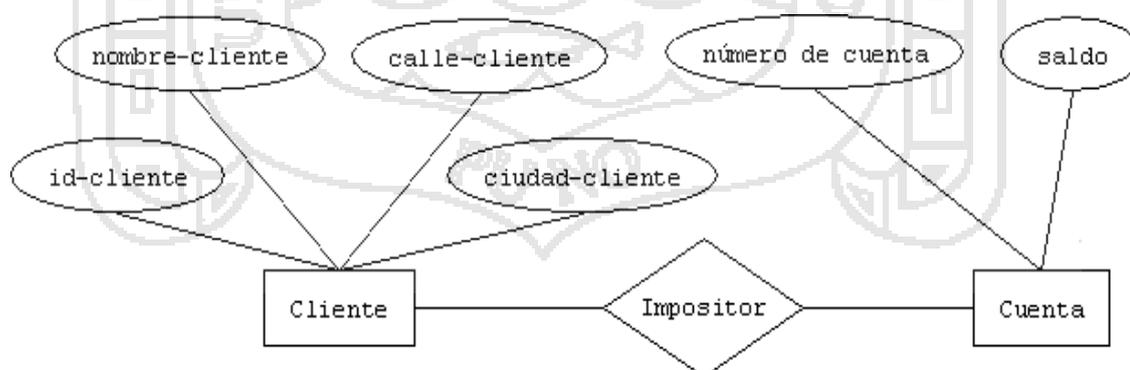
### 2.1.5. Modelamiento de datos

Bajo la estructura de la base de datos se encuentra el modelo de datos: una colección de herramientas conceptuales para describir los datos, las relaciones, la semántica y las restricciones de consistencia. Para ilustrar el concepto de un modelo de datos, describimos dos modelos de datos en este apartado: el modelo entidad-relación y el modelo relacional. Los diferentes modelos de datos que se han propuesto se clasifican en tres grupos diferentes: modelos lógicos basados en objetos, modelos lógicos basados en registros y modelos físicos (Trejo, 2010).

#### Modelo entidad relación

El modelo de datos entidad-relación (E-R) está basado en una percepción del mundo real que consta de una colección de objetos básicos, llamados entidades, y de relaciones entre estos objetos. La estructura lógica general de una base de datos se puede expresar gráficamente mediante un diagrama E-R (Silberschatz *et al*, 2002).

Figura 11. Ejemplo de diagramas E-R.



Fuente: Silberschatz *et al*, 2002.

Donde:

- Rectángulos representan entidades.
- Elipses representan atributos.
- Rombos representan relaciones.
- Líneas unen atributos con entidades y entidades con relaciones.

Además de entidades y relaciones, el modelo E-R representa una restricción importante que es la *cardinalidad*, que expresa el número de entidades con las que otra entidad se puede asociar.

## 2.2. MARCO CONCEPTUAL

### 2.2.1. AMP

La tecnología AMP se refiere a una lista de AMP (Apache, MySQL, Perl / PHP / Python) paquetes de software para todas las plataformas informáticas; estos paquetes se utilizan para ejecutar sitios Webs dinámicos o servidores (Zaldumbide, 2012).

### 2.2.2. Ancho de banda

En conexiones a Internet el ancho de banda es la cantidad de información que se puede enviar a través de una conexión de red en un período de tiempo dado (WIKIMEDIA, 2010).

### 2.2.3. Apache

Es un software que se encarga de gestionar un servidor Web, de distribución libre y de código abierto, multiplataforma, tiene un uso de aproximadamente un 50% del total

de servidores web alrededor del mundo (Zaldumbide, 2012).

#### **2.2.4. Base de datos**

Es una colección de datos interrelacionados (Luna, 2007).

#### **2.2.5. ECA**

El Estándar de Calidad Ambiental (ECA) es la medida que establece el nivel de concentración o del grado de elementos, sustancias o parámetros físicos, químicos y biológicos, presentes en el aire, agua o suelo, en su condición de cuerpo receptor, que no representa riesgo significativo para la salud de las personas ni al ambiente. Según el parámetro en particular a que se refiera, la concentración o grado podrá ser expresada en máximos, mínimos o rangos (Ministerio del ambiente, 2011).

#### **2.2.6. EIA**

La Evaluación de Impacto Ambiental (EIA), es el conjunto de esfuerzos científicos, técnicos y sociales para evaluar la dirección del cambio en el ambiente y su magnitud, producto de una acción antrópica determinada (O'Sullivan, 1999).

#### **2.2.7. HTML**

Lenguaje de Etiquetado de Hipertexto (HyperText Markup Language). Es un lenguaje comúnmente utilizado para la publicación de hipertexto en la Web (Zaldumbide, 2012).

#### **2.2.8. HTTP**

Protocolo de Transferencia de Hipertexto (Hypertext Transfer Protocol). Es un

protocolo utilizado para la transferencia de datos a través de Internet, y que está basado en operaciones sencillas de solicitud y respuesta (Zaldumbide, 2012).

### **2.2.9. IA**

Impacto ambiental (IA), es la alteración positiva o negativa de la calidad ambiental, provocada o inducida por cualquier acción del hombre. Es un juicio de valor sobre un efecto ambiental. Es un cambio neto (bueno o malo) en la salud del hombre o en su bienestar organización (Instituto Caro y Cuervo, 2009).

### **2.2.10. Kanban**

Kanban es una palabra japonesa que significa algo así como “tarjetas visuales” (kan significa visual, y ban tarjeta). Esta técnica se creó en Toyota, y se utiliza para controlar el avance del trabajo, en el contexto de una línea de producción. El Kanban está dentro de la estrategia Kaizen, es decir, la mejora continua y continuada.

### **2.2.11. Lead Time**

Tiempo promedio para completar un iteración, a veces llamado "tiempo de ciclo", optimizar el proceso para hacer tiempo de espera lo más pequeño y más predecible posible.

### **2.2.12. LMP**

El Límite Máximo Permisible (LMP), es la medida de la concentración o del grado de elementos, sustancias o parámetros físicos, químicos y biológicos, que caracterizan a un efluente o una emisión, que al ser excedida causa o puede causar daños a la salud, al bienestar humano y al ambiente. Su cumplimiento es exigible

legalmente por la respectiva autoridad competente. Según el parámetro en particular a que se refiera, la concentración o grado podrá ser expresada en máximos, mínimos o rangos (Ministerio del ambiente, 2011).

### **2.2.13. MySQL**

Es un sistema de gestión de bases de datos (SGBD) que utiliza el lenguaje SQL (Zaldumbide, 2012).

### **2.2.14. Navegador Web**

Un navegador o navegador Web (del inglés, Web browser) es un programa que permite visualizar la información que contiene una página Web (ya se encuentre ésta alojada en un servidor dentro de la World Wide Web o en un servidor local) (WIKIMEDIA, 2010).

### **2.2.15. PMA**

Plan de Manejo Ambiental (PMA); es el conjunto de actividades realizadas con el fin de identificar, valorar, prevenir, corregir, mitigar o compensar los impactos ambientales negativos y potenciar los positivos, derivados de la operación y el mantenimiento de los centros productivos organización (Instituto Caro y Cuervo, 2009).

### **2.2.16. PHP**

PHP es un lenguaje de programación interpretado, diseñado originalmente para la creación de páginas Web dinámicas (Zaldumbide, 2012).

### **2.2.17. RUP**

Es un marco de proceso integral que provee prácticas de desarrollo y testeo para la implementación y entrega de software y sistemas también para la administración de proyectos. Es uno de muchos procesos con Rational Process Library que ofrecen orientación sobre mejores y adecuadas prácticas para desarrollo o necesidad de proyecto.

### **2.2.18. Scrum**

Scrum es un proceso en el que se aplican de manera regular un conjunto de mejores prácticas para trabajar colaborativamente, en equipo, y obtener el mejor resultado posible de un proyecto. Estas prácticas se apoyan unas a otras y su selección tiene origen en un estudio de la manera de trabajar de equipos altamente productivos.

### **2.2.19. Sprint**

En una competición de carrera, aceleración final. Espacio en el que se tiene un tiempo límite para lograr objetivos.

### **2.2.20. SGA**

Sistema de Gestión Ambiental (SGA); es la integración armónica de los elementos requeridos para desarrollar una gestión enfocada en prevenir la contaminación, cumplir los requisitos y legislación ambiental, y mejorar continuamente desempeño ambiental de una organización (Instituto Caro y Cuervo, 2009).

### **2.2.21. WIP**

Material que es incluido en el proceso de producción pero que aún no ha sido

finalizado, Work In Progress se refiere a todo los materiales y productos parcialmente terminados que se encuentran en diversas etapas del proceso de producción, WIP incluye un inventario de materias primas al comienzo del ciclo de producción y los productos terminados al final del mismo.





## METODOLOGÍA DE LA INVESTIGACIÓN

### 3.1. LUGAR DE INVESTIGACIÓN

El Proyecto Chucapaca se encuentra asentado en los andes meridionales del Perú, a 835 km al sureste de la ciudad de Lima y 117 km al sur este de la ciudad de Moquegua (distancias en línea recta), a una altitud que varía entre los 4 700 m y 4 900 m. El Proyecto de Exploración Chucapaca, se ubica en el distrito de Ichuña, perteneciente a la provincia de General Sánchez Cerro, en el departamento de Moquegua. El centro del área del proyecto se encuentra entre las coordenadas UTM 8 208 300 N y 332 275E y 8 207 225N y 334 263E, a una altitud promedio de 4 800 m. Para el acceso al Proyecto, existen dos alternativas de acceso al proyecto. La primera alternativa es a través de la ruta: Puno – Titire – Santiago de Chucapaca, (147 km) y el recorrido dura aproximadamente 3,5 horas. La segunda ruta es a través de la vía Moquegua – Titire – Santiago de Chucapaca (197 km) y el viaje por carretera dura aproximadamente 4,5 horas (Knight Piésold, 2010).

#### o TIPO DE INVESTIGACIÓN

La Investigación pertenece al enfoque *cuantitativo*; pues utilizamos la recolección de datos para probar la hipótesis, con base en la medición numérica y el análisis estadístico, para establecer patrones de comportamiento y probar la teoría (Hernández *et al* 2006).

Así, el conocimiento actual del tema de investigación y la perspectiva que damos a su estudio, revelan un tipo de investigación *descriptivo*; pues en un estudio descriptivo se selecciona una serie de cuestiones y se mide o recolecta información sobre cada una

de ellas, para así (valga la redundancia) describir los que se investiga (Hernández *et al*, 2006).

El diseño de investigación es *Experimental*, sin embargo tomando en cuenta la tipología de Campbell y Stanley (1966) quienes dividen los diseños experimentales en tres clases; consideran uno de ellos el experimento puro que son aquellos que logran el control y la validez interna al reunir dos requisitos: grupos de comparación y equivalencia de los grupos; en este entender consideramos es el indicado para el presente trabajo de investigación (Hernández *et al*, 2006).

#### ○ **TÉCNICAS DE RECOLECCIÓN DE DATOS**

Para el presente trabajo de investigación se ha buscado todas las formas posibles para obtener la información necesaria en el proceso investigativo; las cuales vamos a pasar a nombrar según su clasificación (Reyes, 2008).

#### ▪ **La observación estructurada o formalizada**

La observación estructurada se lleva a cabo cuando se pretende probar una hipótesis, o cuando se quiere hacer una descripción sistemática de un fenómeno; es decir, cuando estamos realizando un estudio o investigación en el que sabemos exactamente lo que vamos a investigar. De esta manera los datos se pueden cuantificar más fácilmente, debido a su homogeneidad, y podemos tener la certeza de no haber olvidado registrar ninguno de los aspectos principales del problema en estudio.

#### ▪ **El cuestionario**

Las encuestas por lo general se centran en la gente, los hechos vitales de la gente, sus creencias, opiniones, motivaciones y conducta; la cual fue aplicada a todas las

personas involucradas con el manejo de la información referente a la recopilación, tabulación e interpretación de datos evaluados denominados “monitores.” La secuencia seguida para la obtención de información mediante la encuesta fue la siguiente:

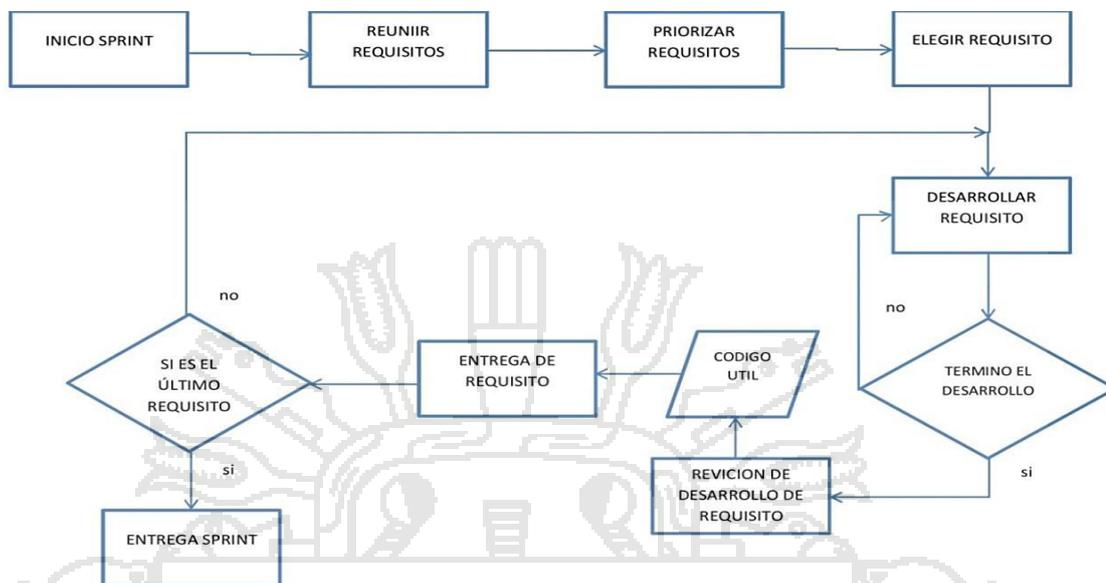
1. Elaboración del Cuestionario, Los sujetos se pueden sentir más libres para responder, no obstante requieren que las preguntas sean cuidadosamente diseñadas para que todos las interpreten de igual manera. Para asegurar la espontaneidad se sugiere que sean anónimas. Pueden aplicarse a varios sujetos simultáneamente (en grupo).
2. Entrevista por contacto personal, es la que proporciona mayor abundancia en los datos. Se observa el comportamiento en forma directa y permite repreguntar.
3. Encuestas telefónicas. Pueden ser rápidas y a bajo costo, pero son limitadas por la falta de cooperación o por la renuencia a contestar algunas preguntas que no sean sencillas ante una persona que no se conoce.

#### ○ **METODOLOGÍAS ÁGILES SCRUM Y KANBAN**

Para el presente trabajo de investigación, utilizamos las metodologías ágiles SCRUM y KANBAN descritas con base teórica en el *capítulo 2.2*, la cual se puede resumir en las siguientes figuras.

**Figura 12. Flujograma de metodología Scrum**

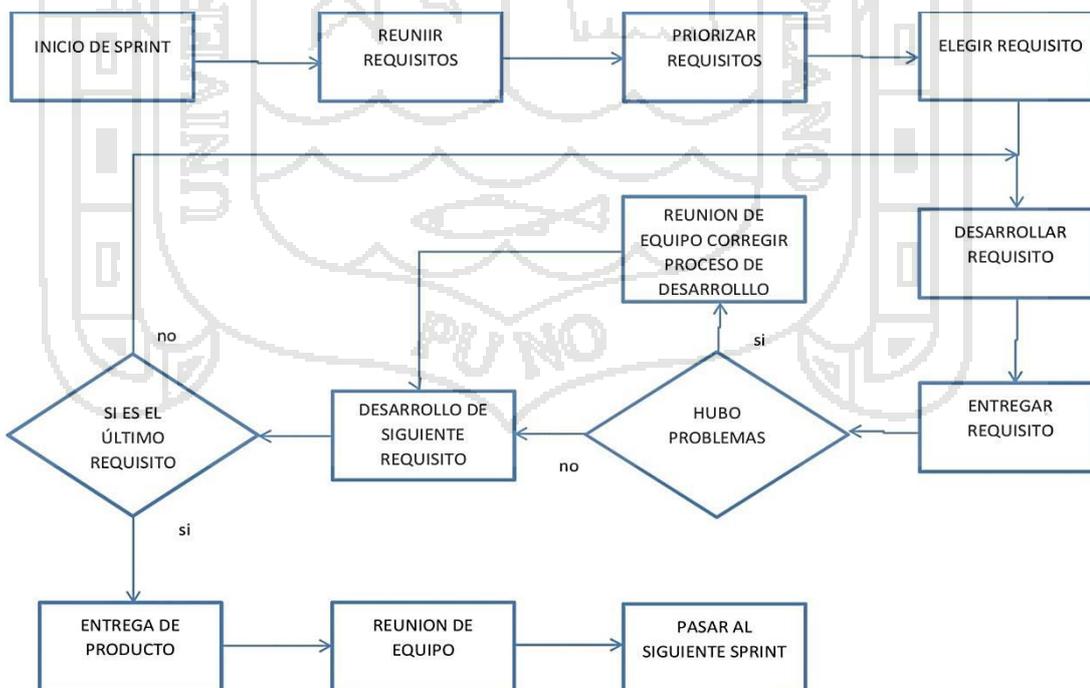
(realizada en la investigación)



Fuente: Elaboración propia.

**Figura 13. Flujograma de metodología Kanban**

(realizada en la investigación)



Fuente: Elaboración propia

○ **INSTRUMENTOS**

**Hardware**

- Computadora personal compatible 2Ghz, memoria 2G, disco duro 20G o superior.
- Impresora.
- Memoria USB.
- Discos Compactos.

**Software**

- Microsoft Windows7.
- Mozilla Firefox 5 o superior.
- ArgoUML 0.30.2.
- MySQLWorkbench 5.1.
- Apache 2.0 o superior.
- PHP 5.0 o superior.
- MySQL 5.0 o superior.
- PHPMyAdmin 2.1 o superior.

**Servicios**

- Conexión a Internet en el computador.

○ **EVALUACIÓN DE RESULTADOS PROCEDENTES DE LA INVESTIGACIÓN.**

Para la cuantificación y ponderación de respuestas, se ha realizado lo siguiente:

- a) Estimación de valores Scrum Kanban, (De donde sacamos las cifras de los cuadros en Scrum y Kanban)
- b) En Scrum el dueño de producto decide la prioridad de las funcionalidades según la cantidad de la misma; si existe 8 tareas que cumplir habrá 8 niveles de prioridad.
- c) Scrum no dice como calificar la estimación de valor, en este caso se tomaron como puntos en un rango de 1 a 10, donde 1 es el nivel más bajo y 10 es el nivel más complejo.
- d) En Scrum el valor del esfuerzo inicial es también tomado en un rango manejado por el ScrumMaster y el equipo puede tomarse de 1 a 10, donde 1 es el más sencillo de desarrollar y 10 el más complejo, tomando en cuenta las opiniones de todo el equipo.
- e) En Scrum se define en una reunión el tiempo para cada miembro el equipo para cada Sprint en semanas, días disponibles por semana y horas disponibles por día.



## EXPOSICIÓN Y ANALISIS DE RESULTADOS

### 4.1. APLICANDO SCRUM

#### 4.1.1. Roles en Scrum

1. Dueño de Producto: Representante del Programa de Monitoreo Socio Ambiental Participativo (PMSAP) del proyecto de exploración minera Chucapaca. A quien no se nombrará en el presente trabajo de investigación por políticas de seguridad de la Empresa en cuestión.
2. Equipo: Conformado por los tesisistas:
  - Alfredo Sven Palao Villasante
  - Eder Jose Nuñez Delgado.
3. ScrumMaster: Eder Jose Nuñez Delgado.

#### 4.1.2. Pila de Producto

Para iniciar con la visión del producto y determinar la “pila de producto” se requiere una reunión entre el “ScrumMaster” y el “Dueño de Producto”, en esta reunión el Dueño de Producto explica a detalle lo que desea que haga el sistema, poco a poco el ScrumMaster interioriza la idea y ayuda al Dueño de Producto a pulirla, por ejemplo después de haber escuchado las peticiones sobre el sistema, el Scrum Master aconseja al Dueño del Producto que por lógica es necesario trabajar con 2 perfiles: el de “Administrador” que se representa por algún integrante del PMSAP que tiene todos los permisos para manejar el sistema y por otro lado está el “Monitor” quien alimenta la base de datos que en este caso son los pobladores de Chucapaca y zonas aledañas.

El dueño de Producto está en todo el derecho de listar todo lo que crea por conveniente para formar parte de la pila de producto; por su lado el Scrum Master con la experiencia obtenida en anteriores proyectos busca un nivel de equilibrio entre el lenguaje del “Dueño de Producto” y el lenguaje del “Equipo”.

Como resultado de esta reunión de aproximadamente 2 horas se obtiene una primera lista de elementos que posteriormente se convierte en “la pila de producto”.

**Tabla 1. Lista de elementos**

<b>Elemento</b>
El administrador desea agregar, quitar o modificar datos de ubicaciones del lugares donde se realiza el análisis impacto ambiental
El administrador desea hacer búsquedas fácilmente
El monitor desea evaluar la información ingresada calificando el estado de alerta que deben tener
El administrador desea generar reportes respecto a la información ingresada
Investigar soluciones para generar reportes visuales
El monitor desea que existan controles de validación en el ingreso de data evitando errores de tipeo
Se desea una interfaz fácil e intuitiva y una consistente Base de datos
El administrador desea agregar, quitar o modificar datos de los monitores
El monitor desea registrar las fichas de información
El administrador desea hacerle seguimiento a cada análisis según estado de alerta
Se desea que el sistema cuente con una validación (clave) para ingresar.
El administrador deseo levantar las observaciones (replicas)

Fuente: Elaboración propia.

El dueño de producto tiene el deber de dar prioridades a cada elemento de la pila de producto, de tal manera que la lista queda de forma ascendente por prioridad; también es responsable de asignar una estimación del valor de negocio a cada elemento.

Esto es normalmente una práctica desconocida para el Dueño de Producto, por esa razón, el Scrum Master enseña al Dueño de Producto a hacerlo, esta reunión dura aproximadamente 1 hora.

**Tabla 2. Asignación de prioridades y Estimación de valor**

Elemento	Prioridad	Estimación de valor
Se desea una interfaz fácil e intuitiva y una consistente Base de datos	1	8
El administrador desea agregar, quitar o modificar datos de los monitores	2	8
El administrador desea agregar, quitar o modificar datos de ubicaciones del lugares donde se realiza el análisis impacto ambiental	3	8
El monitor desea registrar las fichas de información	4	10
El monitor desea evaluar la información ingresada calificando el estado de alerta que deben tener	5	9
El administrador desea hacerle seguimiento a cada análisis según estado de alerta	6	9
El administrador desea generar reportes respecto a la información ingresada	7	10
El administrador deseo hacer búsquedas fácilmente	8	5
El monitor desea que existan controles de validación en el ingreso de data evitando errores de tipeo	9	7
El administrador deseo levantar las observaciones (réplicas)	10	10
Se desea que el sistema cuente con una validación (clave) para ingresar	11	4
Se desea investigar soluciones para generar reportes visuales	12	8

Fuente: Elaboración propia.

El siguiente paso es asignar un valor de esfuerzo a cada elemento, esto es deber del “equipo”, Scrum no especifica cómo se hace este cálculo pero comúnmente es el producto de días por persona y así se considera en el presente trabajo de investigación.

**Tabla 3. Estimación de esfuerzo inicial**

Elemento	Días	Perso- nas	Estima- ción de esfuerzo inicial
Se desea una interfaz fácil e intuitiva y una consistente Base de datos	4	2	8
El administrador desea agregar, quitar o modificar datos de los monitores	3	1	3
El administrador desea agregar, quitar o modificar datos de ubicaciones del lugares donde se realiza el análisis impacto ambiental	3	1	3
El monitor desea registrar las fichas de información	4	2	8
El monitor desea evaluar la información ingresada calificando el estado de alerta que deben tener	1	2	2
El administrador desea hacerle seguimiento a cada análisis según estado de alerta	3	2	6
El administrador desea generar reportes respecto a la información ingresada	5	2	10
El administrador desea hacer búsquedas fácilmente	3	2	6
El monitor desea que existan controles de validación en el ingreso de data evitando errores de tipeo	1	1	1
El administrador desea levantar las observaciones (replicas)	4	2	8
Se desea que el sistema cuente con una validación (clave) para ingresar.	2	2	4
Se desea investigar soluciones para generar reportes visuales	4	2	8

Fuente: Elaboración propia.

Estas estimaciones del valor y esfuerzo se pueden actualizar en cada Sprint a medida que se aprenden cosas nuevas; por tanto, es una actividad continua de re-priorización de la Pila de Producto que evoluciona constantemente. Entonces después de este proceso tenemos la “pila de producto” lista para iniciar el primer Sprint.

**Tabla 4. Pila de producto**

Elemento	Prioridad	Estimación de valor	Estimación de esfuerzo inicial
Se desea una interfaz fácil e intuitiva y una consistente Base de datos	1	8	8
El administrador desea agregar, quitar o modificar datos de los monitores	2	8	3
El administrador desea agregar, quitar o modificar datos de ubicaciones del lugares donde se realiza el análisis impacto ambiental	3	8	3
El monitor desea registrar las fichas de información	4	10	8
El monitor desea evaluar la información ingresada calificando el estado de alerta que deben tener	5	9	2
El administrador desea hacerle seguimiento a cada análisis según estado de alerta	6	9	6
El administrador desea generar reportes respecto a la información ingresada	7	10	10
El administrador desea hacer búsquedas fácilmente	8	5	6
El monitor desea que existan controles de validación en el ingreso de data evitando errores de tipeo	9	7	1
El administrador desea levantar las observaciones (réplicas)	10	10	8
Se desea que el sistema cuente con una validación (clave) para ingresar.	11	4	4
Se desea investigar soluciones para generar reportes visuales	12	8	8

Fuente: Elaboración propia.

Los elementos de la Pila de Producto pueden variar significativamente en tamaño y esfuerzo. Los elementos grandes se parten en elementos más pequeños durante el taller de refinamiento de la Pila de Producto o la Reunión de Planificación del Sprint, y los elementos pequeños pueden ser consolidados.

#### 4.1.3. Reunión de planificación del SPRINT 1

➤ Reunión del sprint 1 parte uno:

En esta reunión el Equipo y el Dueño del Producto revisan los elementos de alta prioridad para definir la cantidad de elementos que se desarrollaran en el primer sprint, el equipo estima en función al esfuerzo y conocimiento del tema y se compromete a entregar los siguientes elementos, que son los de más alta prioridad:

**Tabla 5. Elementos para primer Sprint**

Elemento	Prioridad
Se desea una interfaz fácil e intuitiva y una consistente Base de datos	1
El administrador desea agregar, quitar o modificar datos de los monitores	2
El administrador desea agregar, quitar o modificar datos de ubicaciones del lugares donde se realiza el análisis impacto ambiental	3

Fuente: Elaboración propia.

Sin embargo existe en la pila de producto un elemento de muy baja prioridad que encaja perfectamente en este Sprint y como “El Equipo” en Scrum también tiene

autoridad para seleccionar elementos de más abajo de la lista para incluirlos en el siguiente Sprint, se incluye el elemento:

**Tabla 6. Elemento agregado para primer Sprint**

Elemento	Prioridad
Se desea que el sistema cuente con una validación (clave) para ingresar.	11

Fuente: Elaboración propia.

Ésta es una práctica muy frecuente en la metodología Scrum, de hecho una de sus principales características es su adaptabilidad.

La amena reunión del sprint parte uno tuvo una duración de aproximadamente 1 hora, donde clarifican los objetivos y el contexto de cada elemento.

Ya que el “Equipo” decide cuántos elementos desarrollar en el Sprint, el serio compromiso con el Dueño del Producto se hace más fiable.

➤ Reunión del sprint parte dos:

La Reunión del Sprint Parte Dos se centra en la planificación detallada de tareas para saber cómo implementar los elementos seleccionados en el Sprint.

En esta etapa del proceso el “Equipo” estima cuánto tiempo tiene cada miembro para trabajo relacionado con el Sprint – en otras palabras, su día laboral medio menos el tiempo que pasan en reuniones, leyendo el correo, comiendo, etc. Para la mayoría de la

gente esto significa 4-6 horas de tiempo disponible al día para trabajo relacionado exclusivamente con el Sprint.

**Tabla 7. Estimación de tiempo disponible para Sprint**

<b>Longitud del Sprint</b>			2 semanas
<b>Días laborables durante el Sprint</b>			8 días
<b>Miembro del equipo</b>	<b>Días disponibles durante el Sprint</b>	<b>Horas disponibles por día</b>	<b>Total horas disponibles</b>
Eder Nuñez	8	4	32
Sven Palao	8	5	40

Fuente: Elaboración propia.

El siguiente paso es dividir cada elemento en tareas individuales, que se guardan en un documento llamado Pila del Sprint.

**Tabla 8. Pila del primer Sprint**

Elemento de la pila de producto	Tarea del Sprint	Voluntario	Esfuerzo estimado inicial	Nuevo esfuerzo estimado al final del día									
				1	2	3	4	5	6	7	8		
Deseo una interfaz fácil e intuitiva y una consistente Base de datos	Diseñar la interfaz de todo el sistema	Sven	1	1	0								
	Codificar la interfaz	Sven	1	1	0								
	Diseñar la estructura lógica de Base de datos	Eder	1	1	0								
	Codificar la Base	Eder	2	3	2	1	0						



	de datos											
	Hacer pruebas de funcionamiento	Sven	1			1	1	0				
El administrador desea agregar, quitar o modificar datos de los monitores	Crear código PHP y JQuery que interactúe con la BBDD	Sven	2				3	2	1	0		
	Escribir pruebas de funcionalidad	Eder	1							1	0	
El administrador desea agregar, quitar o modificar datos de ubicaciones del lugares donde se realiza el análisis impacto ambiental	Crear código PHP y JQuery que interactue con la BBDD	Eder	2				2	1	0			
	Escribir pruebas de funcionalidad	Sven	1								1	0
Se desea que el sistema cuente con una validación (clave) para ingresar.	Crear un control de autenticación	Sven	1			1	1	0				
	Hacer pruebas de funcionamiento	Eder	1								1	0

Fuente: Elaboración propia.

#### 4.1.4. Reunión de Scrum diario del Sprint 1

Otra de las prácticas claves de Scrum una vez empezado el Sprint es el Scrum Diario.

Es una reunión corta (10 minutos en este Sprint) que se celebra todos los días a una hora prefijada, el equipo asiste a la reunión de pie para hacerla corta.

Es la oportunidad del equipo de informar a los demás sobre el progreso y los obstáculos. En el Scrum Diario, cada miembro del equipo, uno por uno, informa sobre tres cosas a los otros miembros del equipo:

- Que han hecho desde la última reunión.
- Que tienen planificado hacer antes de la siguiente reunión.
- Cualquier bloqueo o impedimento que tengan.

En este Sprint no hubo ningún bloqueo o impedimento durante los 8 días de duración y los informes diarios respecto a su avance permitió optimizar tiempo y recursos haciendo por ejemplo que Sven el día 3 dedique su tiempo a la tarea “Crear un control de autenticación”, pues Eder demoraría más de lo previsto en la tarea “codificación de la Base de datos”.

4.1.5. Tableros de Desarrollo del Primer Sprint

Figura 14. Se desea una interfaz fácil e intuitiva y una consistente Base de datos

PENDIENTE	EN PROCESO	TERMINADO
		Diseñar la interfaz de todo el sistema
	Diseñar la estructura lógica de Base de Datos	Codificar la interfaz
Codificar la Base de Datos		
Hacer pruebas de funcionamiento		

Fuente: Elaboración propia.

Figura 15. El administrador desea agregar, quitar o modificar datos de los monitores

PENDIENTE	EN PROCESO	TERMINADO
	Crear código PHP y JQuery que interactue con la BBDD	
Escribir pruebas de funcionalidad		

Fuente: Elaboración propia.

**Figura 16. El administrador desea agregar, quitar o modificar datos de ubicaciones del lugar donde se realiza el análisis impacto ambiental**

PENDIENTE	EN PROCESO	TERMINADO
	<div style="border: 1px solid black; background-color: #ADD8E6; padding: 5px; width: fit-content; margin: auto;">                     Escribir pruebas de funcionalidad                 </div>	<div style="border: 1px solid black; background-color: #ADD8E6; padding: 5px; width: fit-content; margin: auto;">                     Crear código PHP y Jquery que interactue con la BBDD                 </div>

Fuente: Elaboración propia.

**Figura 17. Se desea que el sistema cuente con una validación (clave)**

PENDIENTE	EN PROCESO	TERMINADO
<div style="border: 1px solid black; background-color: #ADD8E6; padding: 5px; width: fit-content; margin: auto;">                     Hacer pruebas de funcionamiento                 </div>	<div style="border: 1px solid black; background-color: #ADD8E6; padding: 5px; width: fit-content; margin: auto;">                     Crear un control de autenticación                 </div>	

Fuente: Elaboración propia.

#### 4.1.6. Resultado del Sprint 1

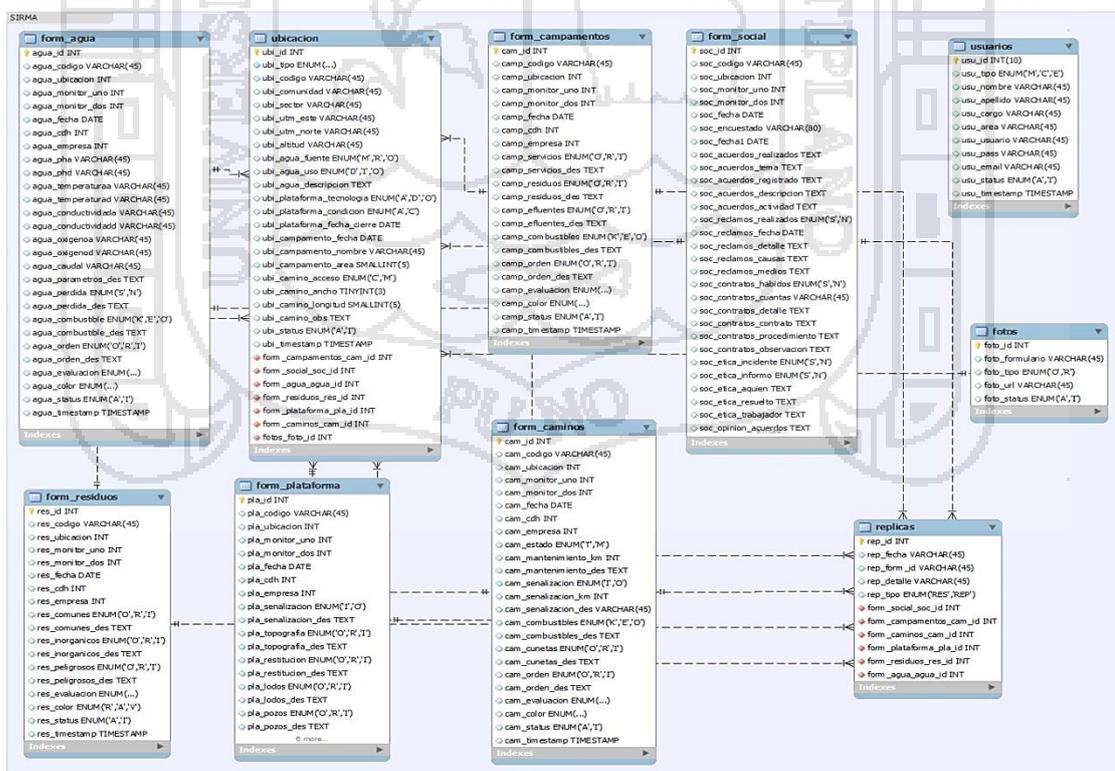
Los resultados del Sprint 1 fueron:

Figura 18. Interfaz de usuario



Fuente: Elaboración propia.

Figura 19. Modelado de Base de Datos



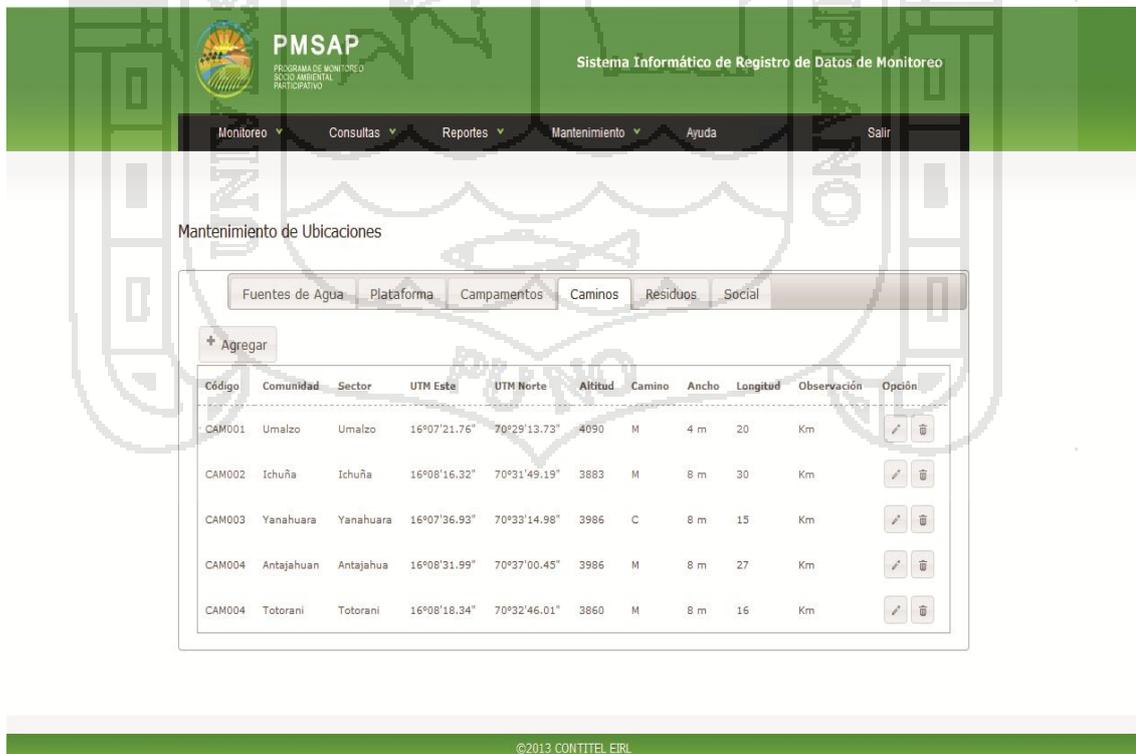
Fuente: Elaboración propia.

Figura 20. Elemento: “Agregar, quitar, modificar datos de monitores”

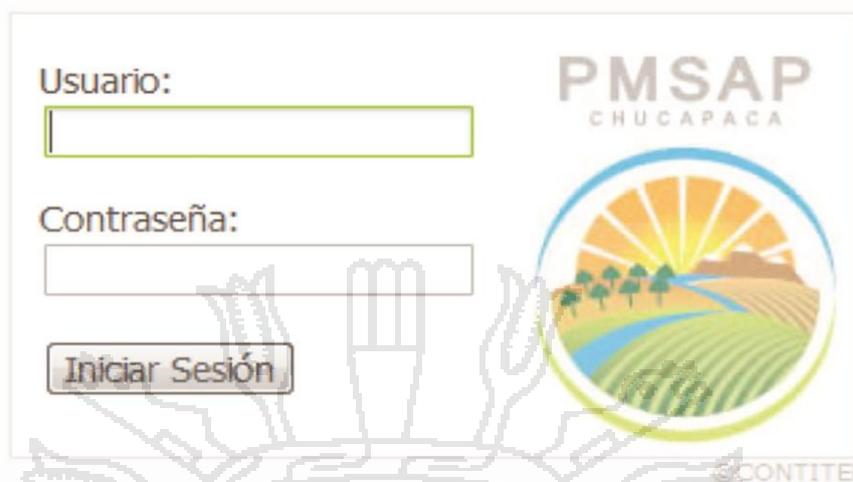


Fuente: Elaboración propia.

Figura 21. Elemento: “Agregar, quitar, modificar datos de lugares”



Fuente: Elaboración propia.

**Figura 22. Elemento: “validación (autenticación) para ingresar”**


Fuente: Elaboración propia.

#### 4.1.7. Reunión de planificación del SPRINT 2

➤ Reunión del sprint parte uno: En esta reunión el Equipo y el Dueño del Producto revisan los siguientes elementos de alta prioridad que quedan en la pila de producto para definir la cantidad de elementos que se desarrollaran en el segundo sprint, el equipo estima en función al esfuerzo y conocimiento del tema y se compromete a entregar los siguientes elementos:

**Tabla 9. Elementos para segundo Sprint**

Elemento	Prioridad
El monitor desea registrar las fichas de información	4
El monitor desea evaluar la información ingresada calificando el estado de alerta que deben tener	5
El administrador desea hacerle seguimiento a cada análisis según estado de alerta	6

Fuente: Elaboración propia.

Nuevamente se observa que existe en la pila de producto un elemento de muy baja prioridad que encaja perfectamente en este Sprint.

**Tabla 10. Elemento agregado para segundo Sprint**

Elemento	Prioridad
El monitor desea que existan controles de validación en el ingreso de data evitando errores de tipeo.	9

Fuente: Elaboración propia.

La reunión del sprint 2 parte uno tuvo una duración de aproximadamente 2 horas, siendo de suma importancia lograr entender la lógica con la que el Dueño de Producto (que en este Sprint fueron 2) deciden aplicar en el sistema; el enfoque en los resultados permite al “equipo” saciar todas sus dudas respecto al tratamiento de datos que debe darse en esta parte del sistema.

➤ Reunión del sprint 2 parte dos:

En la Reunión del Sprint 2 Parte Dos el “Equipo” estima cuánto tiempo tiene cada miembro para trabajo relacionado con el Sprint 2.

**Tabla 11. Estimación de tiempo disponible para Sprint 2**

<b>Longitud del Sprint</b>	2 semanas
<b>Días laborables durante el Sprint</b>	8 días

<b>Miembro del equipo</b>	<b>Días disponibles durante el Sprint</b>	<b>Horas disponibles por día</b>	<b>Total horas disponibles</b>
Eder Nuñez	6	5	30
Sven Palao	8	5	40

Fuente: Elaboración propia.

**Tabla 12. Pila del segundo Sprint**

<b>Elemento de la pila de producto</b>	<b>Tarea del Sprint</b>	<b>Voluntario</b>	<b>Esfuerzo estimado inicial</b>	<b>Nuevo esfuerzo estimado al final del día</b>									
				<b>1</b>	<b>2</b>	<b>3</b>	<b>4</b>	<b>5</b>	<b>6</b>	<b>7</b>	<b>8</b>		
El monitor desea registrar las fichas de información	Crear la interfaz para modulo Monitoreo - Fuentes de agua	Sven	2	1	0								
	Crear código PHP y Jquery que interactue con la BBDD para modulo Fuentes de agua	Sven	3	2	1	0							
	Escribir pruebas de funcionalidad para modulo Fuentes de agua	Eder	1			1	0						
	Crear la interfaz para modulo Monitoreo - Plataforma de perforacion	Sven	2			1	0						



Crear código PHP y Jquery que interactue con la BBDD para modulo Plataforma de perforación	Sven	3	2	1	0				
Escribir pruebas de funcionalidad para modulo Plataforma de perforación	Eder	1		1	0				
Crear la interfaz para modulo Monitoreo - Campamentos	Sven	2		1	0				
Crear código PHP y Jquery que interactue con la BBDD para modulo Campamentos	Sven	3		1	0				
Escribir pruebas de funcionalidad para modulo Campamentos	Eder	1			1	0			
Crear la interfaz para modulo Monitoreo - Caminos de acceso	Sven	2			1	0			
Crear código PHP y Jquery que interactue con la BBDD para modulo Caminos de acceso	Sven	3			1	0			
Escribir pruebas de funcionalidad para modulo Caminos de acceso	Eder	1				1	0		
Crear la interfaz para modulo Monitoreo -	Sven	2				1	0		



	Almacenamiento y manejo de residuos												
	Crear código PHP y JQuery que interactue con la BBDD para modulo Almacenamiento y manejo de residuos	Sven	3						1	0			
	Escribir pruebas de funcionalidad para modulo Almacenamiento y manejo de residuos	Eder	1										
	Crear la interfaz para modulo Monitoreo – Social	Sven	2								1	0	
	Crear código PHP y JQuery que interactue con la BBDD para modulo Social	Sven	3								1	0	
	Escribir pruebas de funcionalidad para modulo Social	Eder	1									1	0
El monitor desea evaluar la información ingresada calificando el estado de alerta que deben tener	Crear la interfaz para este modulo	Eder	1				1						
	Crear código PHP y JQuery que interactue con la BBDD	Eder	2					2	1	0			
	Hacer pruebas de funcionalidad	Sven	1									1	0
El	Crear la interfaz para	Eder	1								1	0	



administrador	este modulo																		
desea hacerle seguimiento a cada análisis según estado de alerta	Crear código PHP y JQuery que interactue con la BBDD	Eder	6																
	Hacer pruebas de funcionalidad	Sven	1																
El monitor desea que existan controles de validación en el ingreso de data evitando errores de tipeo	crear codigo Mysql con especificaciones	Eder	1																

Fuente: Elaboración propia.

**4.1.8. Reunión de Scrum diario del Sprint 2**

Como ya se mencionó antes en el Scrum Diario, cada miembro del equipo, informa sobre tres cosas a los otros miembros del equipo:

- Que han hecho desde la última reunión.
- Que tienen planificado hacer antes de la siguiente reunión.
- Cualquier bloqueo o impedimento que tengan.

En este Sprint hubo muchísima coordinación en el Scrum Diario es necesario que ambos integrantes del equipo se ayuden, pues Sven inicia con el código para ingresar las fichas de monitoreo dos días antes, Eder se incorpora al 3er día como lo planificado,

Sven informa de los avances en esos 2 días, lo cual facilita al compañero buscar uniformidad en los resultados, sin quitarle tiempo de dedicación al Sprint.

#### 4.1.9. Tableros de Desarrollo Segundo Sprint

Figura 23. El monitor desea registrar las fichas de información

PENDIENTE	EN PROCESO	TERMINADO
		Interfaz Monitoreo Fuentes de agua PHP y Jquery - BBDD Fuentes de agua Pruebas modulo Fuentes de agua
	Interfaz Monitoreo Plataforma de perforación PHP y Jquery - BBDD Plataforma de perforación Pruebas modulo Plataforma de perforación Interfaz Monitoreo Campamentos PHP y Jquery - BBDD Campamentos Pruebas modulo Campamentos	
Interfaz Monitoreo Caminos de acceso PHP y Jquery - BBDD Caminos de acceso Pruebas modulo Caminos de acceso Interfaz Monitoreo Almacenamiento y manejo de residuos PHP y Jquery - BBDD Almacenamiento y manejo de residuos Pruebas modulo Almacenamiento y manejo de residuos Interfaz Monitoreo Social PHP y Jquery - BBDD Social Pruebas modulo Social		

Fuente: Elaboración propia.

**Figura 24. El monitor desea evaluar la información ingresada calificando el estado de alerta que deben tener**

PENDIENTE	EN PROCESO	TERMINADO
	<div style="border: 1px solid black; padding: 5px; width: fit-content; margin: auto;">                     Crear código PHP y jquery que interactue con la BBDD                 </div>	<div style="border: 1px solid black; padding: 5px; width: fit-content; margin: auto;">                     Crear interfaz para este modulo                 </div>
<div style="border: 1px solid black; padding: 5px; width: fit-content; margin: auto;">                     Hacer pruebas de funcionalidad                 </div>		

Fuente: Elaboración propia.

**Figura 25. El administrador desea hacerle seguimiento a cada análisis según estado de alerta**

PENDIENTE	EN PROCESO		TERMINADO
	ANÁLISIS (1)	DESARROLLO (1)	
<div style="border: 1px solid black; padding: 5px; width: fit-content; margin: auto;">                     Hacer pruebas de funcionalidad                 </div>	<div style="border: 1px solid black; padding: 5px; width: fit-content; margin: auto;">                     Crear código PHP y jquery que interactue con la BBDD                 </div>	<div style="border: 1px solid black; padding: 5px; width: fit-content; margin: auto;">                     Crear la interfaz para este modulo                 </div>	

Fuente: Elaboración propia.

**Figura 26. El monitor desea que existan controles de validación en el ingreso de data evitando errores de tipeo**

PENDIENTE	EN PROCESO	TERMINADO
	<div style="border: 1px solid black; padding: 5px; width: fit-content; margin: auto;">                     Crear codigo Mysql con especificaciones                 </div>	

Fuente: Elaboración propia.

4.1.10. Resultado del Sprint 2

Figura 27. Ingreso de datos Fuentes de Agua

**PMSAP**  
PROGRAMA DE MONITOREO SOCIOAMBIENTAL PARTICIPATIVO

Sistema Informático de Registro de Datos de Monitoreo

Monitoreo Consultas Reportes Mantenimiento Ayuda Salir

Registro de Objeto Ambiental : Fuente de Agua

**Datos Generales**

Ubicación: Seleccione... CDH: Seleccione...  
 Monitor (1): Seleccione... Empresa Sub-Contratista: Seleccione...  
 Monitor (2): Seleccione...  
 Fecha: 2013-12-13

**Variables de la Observación**

1. Parámetros físicos del agua (muestra antes y después de la toma):

PH:  Antes  Después  
 Temperatura (C°):   
 Conductividad Eléctrica (us/cm):   
 Oxígeno Disuelto (mg/L):   
 Caudal L/Seg:  Descripción:

Fotografías:  
 ningún .cionado  
 ningún .cionado  
 ningún .cionado

2. Pérdida de Agua:  Si  No Descripción:

Fotografías:  
 ningún .cionado  
 ningún .cionado  
 ningún .cionado

3. Manejo de lubricantes y combustibles:  Kit Antiderrames  Extintor  Otros Descripción:

Fotografías:  
 ningún .cionado  
 ningún .cionado  
 ningún .cionado

4. Orden y Limpieza:  Óptimo  Regular  Inadecuado Descripción:

Fotografías:  
 ningún .cionado  
 ningún .cionado  
 ningún .cionado

**Evaluación Final**

Positiva  Sin Observación  Alerta  Riesgo  Emergencia

Impacto con visible aporte de trabajo que mejora el ecosistema. Impacto manejable y control inmediato mediante respuestas normales. Impacto con peligro inminente que salga fuera de los límites y amerita una acción inmediata. Impacto donde se pierde control y hay acumulación de otros impactos.

© 2013 CONITTEL EURL

Fuente: Elaboración propia.

Figura 28. Ingreso de datos Campamentos

**PMSAP**  
PROGRAMA DE MONITOREO SOCIOAMBIENTAL PARTICIPATIVO

Sistema Informático de Registro de Datos de Monitoreo

Monitoreo ▾
Consultas ▾
Reportes ▾
Mantenimiento ▾
Ayuda

Registro de Objeto Ambiental : Campamentos

**Datos Generales**

Ubicación:

Monitor (1):

Monitor (2):

Fecha:

CDH:

Empresa Sub-Contratista:

**Variables de la Observación**

**1. Descripción del Estado de Servicios Higiénicos**

Óptimo

Regular

Inadecuado

Descripción:

Fotografías:

ningún .cionado

ningún .cionado

ningún .cionado

**2. Manejo de Residuos sólidos:**

Óptimo

Regular

Inadecuado

Descripción:

Fotografías:

ningún .cionado

ningún .cionado

ningún .cionado

**3. Manejo de efluentes líquidos:**

Óptimo

Regular

Inadecuado

Descripción:

Fotografías:

ningún .cionado

ningún .cionado

ningún .cionado

**4. Manejo de combustibles y lubricantes:**

Kit Antiderrame

Extintor

Otros

Descripción:

Fotografías:

ningún .cionado

ningún .cionado

ningún .cionado

**5. Orden y Limpieza:**

Óptimo

Regular

Inadecuado

Descripción:

Fotografías:

ningún .cionado

ningún .cionado

ningún .cionado

**Evaluación Final**

Positiva

Impacto con visible aporte de trabajo que mejora el ecosistema

Sin Observación

Impacto manejable y control inmediato mediante respuestas normales

Alerta

Impacto con peligro inminente que salga fuera de los límites y amerita una acción inmediata

Riesgo

Impacto donde se pierde control y hay acumulación de otros impactos

Emergencia

Impacto donde se pierde control y hay acumulación de otros impactos

©2013 CON TITEL EIRL

Fuente: Elaboración propia.

Figura 29. Ingreso de datos Caminos de Acceso

Sistema Informático de Registro de Datos de Monitoreo

Monitoreo ▾
Consultas ▾
Reportes ▾
Mantenimiento ▾
Ayuda
Salir

Registro de Objeto Ambiental : Caminos de Acceso

**Datos Generales**

Ubicación: <input type="text" value="Seleccione..."/>	CDH: <input type="text" value="Seleccione..."/>
Monitor (1): <input type="text" value="Seleccione..."/>	Empresa Sub-Contratista: <input type="text" value="Seleccione..."/>
Monitor (2): <input type="text" value="Seleccione..."/>	
Fecha: <input type="text" value="2013-12-13"/>	

**Variables de la Observación**

1. Estado del camino de acceso:  Tránsito Regular  En mantenimiento

2. Mantenimiento:

Fotografías:  
 ningún \_cionado  
 ningún \_cionado  
 ningún \_cionado

3. Señalización:  Informativa  Otros

Fotografías:  
 ningún \_cionado  
 ningún \_cionado  
 ningún \_cionado

3. Manejo de combustibles y lubricantes:  Kt Antiderrame  Extintor  Otros

Fotografías:  
 ningún \_cionado  
 ningún \_cionado  
 ningún \_cionado

4. Manejo de Cunetas:  Optimo  Regular  Inadecuado

Fotografías:  
 ningún \_cionado  
 ningún \_cionado  
 ningún \_cionado

5. Orden y Limpieza:  Optimo  Regular  Inadecuado

Fotografías:  
 ningún \_cionado  
 ningún \_cionado  
 ningún \_cionado

**Evaluación Final**

<input type="radio"/> Positiva Impacto con visible aporte de trabajo que mejora el ecosistema	<input type="radio"/> Sin Observación	<input type="radio"/> Alerta Impacto manejable y control inmediato mediante respuestas normales	<input type="radio"/> Riesgo Impacto con peligro inminente que salga fuera de los límites y amerita una acción inmediata	<input type="radio"/> Emergencia Impacto donde se pierde control y hay acumulación de otros impactos
--	---------------------------------------	--	---	---

©2013 CONTITEL EIRL

Fuente: Elaboración propia.

Figura 30. Ingreso de datos Almacenamiento de Manejo de Residuos

Sistema Informático de Registro de Datos de Monitoreo

Monitoreo ▾
Consultas ▾
Reportes ▾
Mantenimiento ▾
Ayuda
Salir

Registro de Objeto Ambiental : Almacenamiento y Manejo de Residuos

**Datos Generales**

Ubicación: <input type="text" value="Seleccione..."/>	CDH: <input type="text" value="Seleccione..."/>
Monitor (1): <input type="text" value="Seleccione..."/>	Empresa Sub-Contratista: <input type="text" value="Seleccione..."/>
Monitor (2): <input type="text" value="Seleccione..."/>	
Fecha: <input type="text" value="2013-12-13"/>	

**Variables de la Observación**

**1. Residuos Comunes**

Optimo  
 Regular  
 Inadecuado

Fotografías:  
 ningún \_cionado  
 ningún \_cionado  
 ningún \_cionado

Descripción

---

**2. Residuos Inorgánicos:**

Optimo  
 Regular  
 Inadecuado

Fotografías:  
 ningún \_cionado  
 ningún \_cionado  
 ningún \_cionado

Descripción

---

**3. Residuos Peligrosos:**

Optimo  
 Regular  
 Inadecuado

Fotografías:  
 ningún \_cionado  
 ningún \_cionado  
 ningún \_cionado

Descripción

**Evaluación Final**

<input type="radio"/> Positiva	<input type="radio"/> Sin	<input checked="" type="radio"/> Alerta	<input type="radio"/> Riesgo	<input type="radio"/> Emergencia
Impacto con visible aporte de trabajo que mejora el ecosistema	<b>Observación</b>	Impacto manejable y control inmediato mediante respuestas normales	Impacto con peligro inminente que salga fuera de los límites y amerita una acción inmediata	Impacto donde se pierde control y hay acumulación de otros impactos

©2013 CONTITEL EIRL

Fuente: Elaboración propia.

#### 4.1.11. Reunión de planificación del SPRINT 3

- Reunión del sprint 3 parte uno:

En esta reunión el Equipo y el Dueño del Producto revisan los últimos elementos de por desarrollar en la pila de producto para completar el desarrollo del producto, el equipo estima en función al esfuerzo y conocimiento del tema y se compromete a entregar los siguientes elementos:

**Tabla 13. Elementos para tercer Sprint**

Elemento	Prioridad
El administrador desea generar reportes respecto a la información ingresada	7
El administrador desea hacer búsquedas fácilmente	8
El administrador desea levantar las observaciones (réplicas)	10
Se desea investigar soluciones para generar reportes visuales	12

La reunión del sprint 3 parte uno tuvo una duración de aproximadamente 1 hora, esta vez el enfoque es la generación de reportes, sin dejar de lado las búsquedas y las réplicas.

Respecto a la nueva petición del elemento “Deseo ver la ubicación satelital (grafica) para modulo lugares” fue el resultado de un fragmento de conversación entre el equipo y el Dueño de Producto en la reunión de planificación del Sprint 2, si bien lo ideal era manifestarlo en este momento, el DP puede hacerlo ahora esto inevitablemente causa molestia a un integrante del equipo, pero gracias al ScrumMaster se mantuvo el buen y cordial trato entre el “equipo” y el “Dueño de Producto”.

- Reunión del sprint 3 parte dos:

En la Reunión del Sprint 3 Parte Dos el “Equipo” estima su tiempo.

**Tabla 14. Estimación de tiempo disponible para Sprint 3**

<b>Longitud del Sprint</b>	3 semanas
<b>Días laborables durante el Sprint</b>	8 días

<b>Miembro del equipo</b>	<b>Días disponibles durante el Sprint</b>	<b>Horas disponibles por día</b>	<b>Total horas disponibles</b>
Eder Nuñez	8	4	32
Sven Palao	8	5	40

Fuente: Elaboración propia.

**Tabla 15. Pila del segundo Sprint 3**

<b>Elemento de la pila de producto</b>	<b>Tarea del Sprint</b>	<b>Voluntario</b>	<b>Esfuerzo estimado inicial</b>	<b>Nuevo esfuerzo estimado al final del día</b>									
				<b>1</b>	<b>2</b>	<b>3</b>	<b>4</b>	<b>5</b>	<b>6</b>	<b>7</b>	<b>8</b>		
¿El administrador desea generar reportes respecto a la información ingresada?	Crear la interfaz para este modulo	Sven	2	1	0								
	Crear código PHP y JQuery que interactue con la BBDD	Eder	3	2	1	0							
	Hacer pruebas de funcionalidad	Sven	1			1	0						
¿El administrador desea hacer búsquedas fácilmente?	Crear la interfaz para este modulo	Sven	2			1	0						
	Crear código PHP y JQuery que interactue con la	Eder	3				1	0					



	BBDD													
	Hacer pruebas de funcionalidad	Sven	1					1	0					
¿El administrador desea levantar las observaciones (réplicas)?	Crear la interfaz para este modulo	Eder	2						1	0				
	Crear código PHP y JQuery que interactue con la BBDD	Sven	3						1	0				
	Hacer pruebas de funcionalidad	Eder	1							1	0			
¿Se desea investigar soluciones para generar reportes visuales?	Crear la interfaz para este modulo	Sven	2							1	0			
	Crear código PHP y JQuery que interactue con la BBDD	Eder	3									1	0	
	Hacer pruebas de funcionalidad	Sven	1									1	0	

Fuente: Elaboración propia.

**4.1.12. Reunión de Scrum diario del Sprint 3**

Como ya se mencionó antes en el Scrum Diario, cada miembro del equipo, informa sobre tres cosas a los otros miembros del equipo:

- ¿Qué han hecho desde la última reunión?
- ¿Qué tienen planificado hacer antes de la entrega del producto?
- ¿Qué bloqueo o impedimento han tenido?.

En este Sprint algunos problemas de tiempo y muchas tareas no se concluyeron satisfactoriamente, requiriendo que se reúnan con del dueño del producto y revisar el sistema en su integridad.

4.1.13. Tableros de Desarrollo tercer Sprint

**Figura 31. El administrador desea generar reportes respecto a la información ingresada**

PENDIENTE	EN PROCESO		TERMINADO
	ANÁLISIS (1)	DESARROLLO (1)	
hacer pruebas de funcionalidad		crear código PHP y JQuery para que interactue con la BBDD	crear interfaz para este modulo

Fuente: Elaboración propia.

**Figura 32. Como administrador deseo hacer búsquedas fácilmente**

PENDIENTE	EN PROCESO		TERMINADO
	ANÁLISIS (1)	DESARROLLO (1)	
hacer pruebas de funcionalidad	crear código PHP y JQuery para que interactue con la BBDD	crear interfaz para este modulo	

Fuente: Elaboración propia.

**Figura 33. El administrador desea levantar las observaciones (réplicas)**

PENDIENTE	EN PROCESO		TERMINADO
	ANÁLISIS (1)	DESARROLLO (1)	
		<div data-bbox="831 607 1094 680" style="background-color: #ADD8E6; padding: 2px;">hacer pruebas de funcionalidad</div>	<div data-bbox="1123 443 1386 510" style="background-color: #ADD8E6; padding: 2px;">crear interfaz para este modulo</div> <div data-bbox="1123 517 1386 584" style="background-color: #ADD8E6; padding: 2px;">crear codigo PHP y JQuery para que interactue con la BBDD</div>

Fuente: Elaboración propia.

**Figura 34. Investigar soluciones para generar reportes visuales**

PENDIENTE	EN PROCESO		TERMINADO
	ANÁLISIS (1)	DESARROLLO (1)	
			<div data-bbox="1123 1126 1386 1193" style="background-color: #ADD8E6; padding: 2px;">crear interfaz para este modulo</div> <div data-bbox="1123 1200 1386 1267" style="background-color: #ADD8E6; padding: 2px;">crear codigo PHP y JQuery para que interactue con la BBDD</div> <div data-bbox="1123 1296 1386 1364" style="background-color: #ADD8E6; padding: 2px;">hacer pruebas de funcionalidad</div>

Fuente: Elaboración propia.

#### 4.1.14. Resultado del Sprint 3

Los resultados del Sprint 3 fueron:

**Figura 35. El administrador desea generar reportes respecto a la información ingresada**

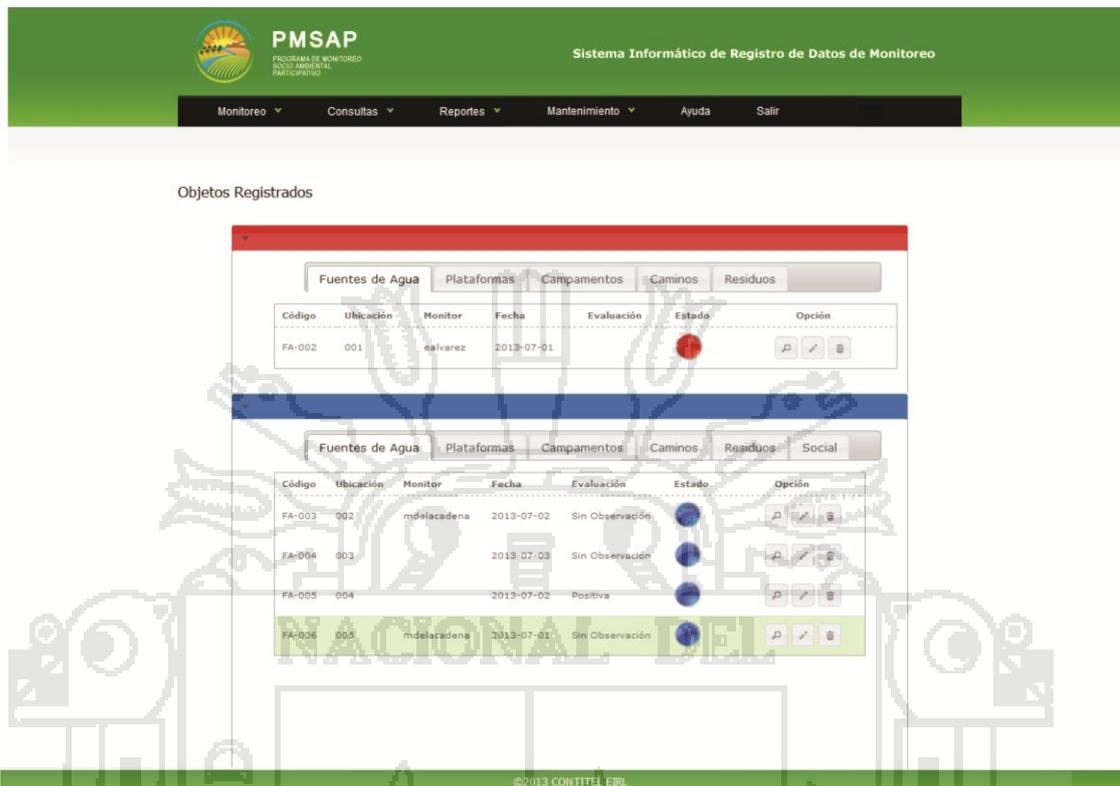


Fuente: Elaboración propia.

**Figura 36. El administrador desea hacer búsquedas fácilmente**



Fuente: Elaboración propia.

**Figura 37. Se desea investigar soluciones para generar reportes visuales**


Objetos Registrados

Código	Ubicación	Monitor	Fecha	Evaluación	Estado	Opción
FA-002	001	salvarez	2013-07-01			

Código	Ubicación	Monitor	Fecha	Evaluación	Estado	Opción
FA-003	002	mdelacadena	2013-07-02	Sin Observación		
FA-004	003		2013-07-03	Sin Observación		
FA-005	004		2013-07-02	Positiva		
FA-006	005	mdelacadena	2013-07-01	Sin Observación		

©2013 COMITTEL EIRL

Fuente: Elaboración propia.

#### 4.2. APLICANDO KANBAN

Kanban limita el WIP (Work in Progress - trabajo en curso) asigna límites concretos a cuántos elementos pueden estar en progreso en cada estado del flujo de trabajo.

Kanban mide el lead time (tiempo medio para completar un elemento, a veces llamado "tiempo de ciclo"), optimiza el proceso para que el lead time sea tan pequeño y predecible como sea posible, en este caso se ha tomado en cuenta el mismo tiempo que tomamos para los Sprint como se ilustra en las figuras siguientes. Reutilizaremos la Pila de Producto hecho en el análisis de Scrum.

Según la cantidad de entregables de cada Sprint en Kanban podemos sugerir la cantidad de tareas en proceso y análisis, ya conocido que hay existen tareas muy

similares ponemos varias en estas columnas sin temor a no cumplirlas es así que planificamos, desarrollamos y analizamos el proceso Kanban.

Ya realizado los cálculos del tiempo necesario para realizar cada ítem o tarea, lo cual no genero los siguientes tableros muestra del trabajo en progreso.

**Figura 38. Se desea una interfaz fácil e intuitiva y una consistente Base de datos**

PENDIENTE	EN PROCESO		TERMINADO
	ANÁLISIS (1)	DESARROLLO (2)	
			Diseñar la interfaz de todo el sistema
		Codificar la interfaz	
		Diseñar la estructura lógica de Base de Datos	
	Codificar la Base de Datos		
Hacer pruebas de funcionamiento			

Fuente: Elaboración propia.

**Figura 39. El administrador desea agregar, quitar o modificar datos de los monitores**

PENDIENTE	EN PROCESO		TERMINADO
	ANÁLISIS (1)	DESARROLLO (1)	
			Crear código PHP y JQuery que interactue con la BBDD
		Escribir pruebas de funcionalidad	

Fuente: Elaboración propia.

**Figura 40. El administrador desea agregar, quitar o modificar datos de ubicación de lugares donde se realiza el análisis impacto ambiental**

PENDIENTE	EN PROCESO		TERMINADO
	ANÁLISIS (1)	DESARROLLO (1)	
Escribir pruebas de funcionalidad		Crear código PHP y JQuery que interactue con la BBDD	

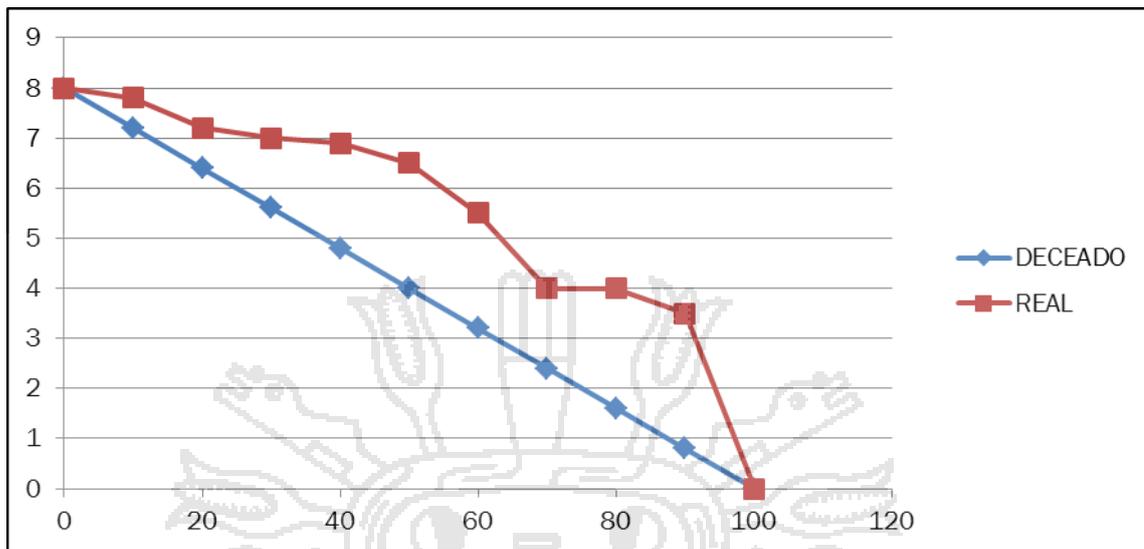
Fuente: Elaboración propia.

**Figura 41. Se desea que el sistema cuente con una validación (clave) para ingresar**

PENDIENTE	EN PROCESO		TERMINADO
	ANÁLISIS (1)	DESARROLLO (2)	
		Hacer pruebas de funcionamiento	Crear un control de autenticación

Fuente: Elaboración propia.

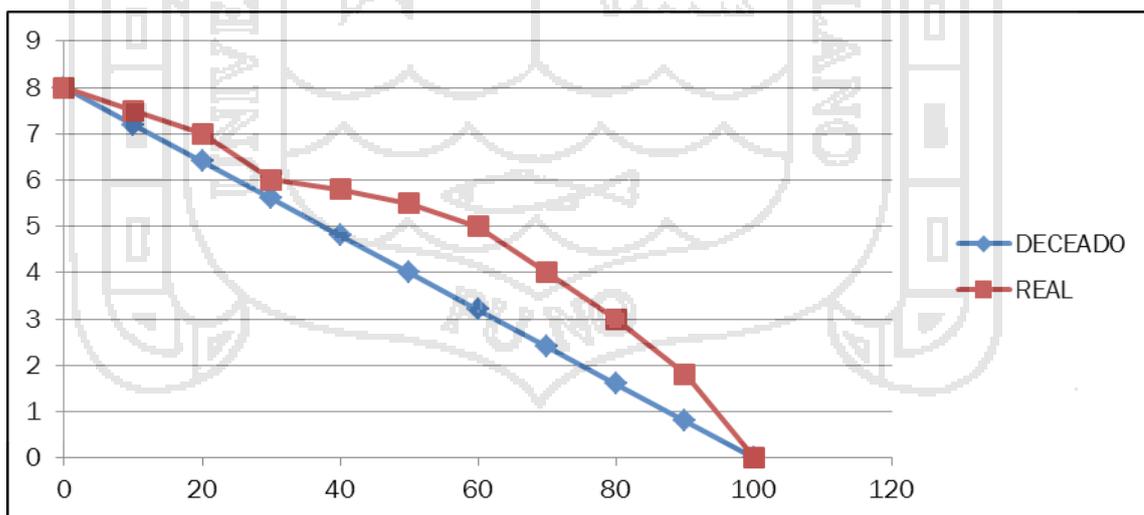
**Figura 42. Visualización proceso de desarrollo de Sprint Miembro 1**



X = días Programados  
 Y = porcentaje de avance en este Sprint

Fuente: Elaboración propia.

**Figura 43. Visualización proceso de desarrollo de Sprint Miembro 2**



X = días Programados  
 Y = porcentaje de avance en este Sprint

Fuente: Elaboración propia.

Figura 44. El monitor desea registrar las fichas de información

PENDIENTE	EN PROCESO		TERMINADO
	ANÁLISIS (3)	DESARROLLO (3)	
			Interfaz Monitoreo Fuentes de agua PHP y Jquery - BBDD Fuentes de agua Pruebas modulo Fuentes de agua
		Interfaz Monitoreo Plataforma de perforación PHP y Jquery - BBDD Plataforma de perforación Pruebas modulo Plataforma de perforación	
	Interfaz Monitoreo Campamentos PHP y Jquery - BBDD Campamentos Pruebas modulo Campamentos		
Interfaz Monitoreo Caminos de acceso PHP y Jquery - BBDD Caminos de acceso Pruebas modulo Caminos de acceso			
Interfaz Monitoreo Almacenamiento y manejo de residuos PHP y Jquery - BBDD Almacenamiento y manejo de residuos Pruebas modulo Almacenamiento y manejo de residuos			
Interfaz Monitoreo Social PHP y Jquery - BBDD Social Pruebas modulo Social			

Fuente: Elaboración propia.

**Figura 45. El monitor desea evaluar la información ingresada calificando el estado de alerta que deben tener**

PENDIENTE	EN PROCESO		TERMINADO
	ANÁLISIS (1)	DESARROLLO (1)	
Hacer pruebas de funcionalidad	Crear código PHP y jquery que interactue con la BBDD	Crear interfaz para este modulo	

Fuente: Elaboración propia.

**Figura 46. El administrador desea hacerle seguimiento a cada análisis según estado de alerta**

PENDIENTE	EN PROCESO		TERMINADO
	ANÁLISIS (1)	DESARROLLO (1)	
Hacer pruebas de funcionalidad	Crear código PHP y jquery que interactue con la BBDD	Crear la interfaz para este modulo	

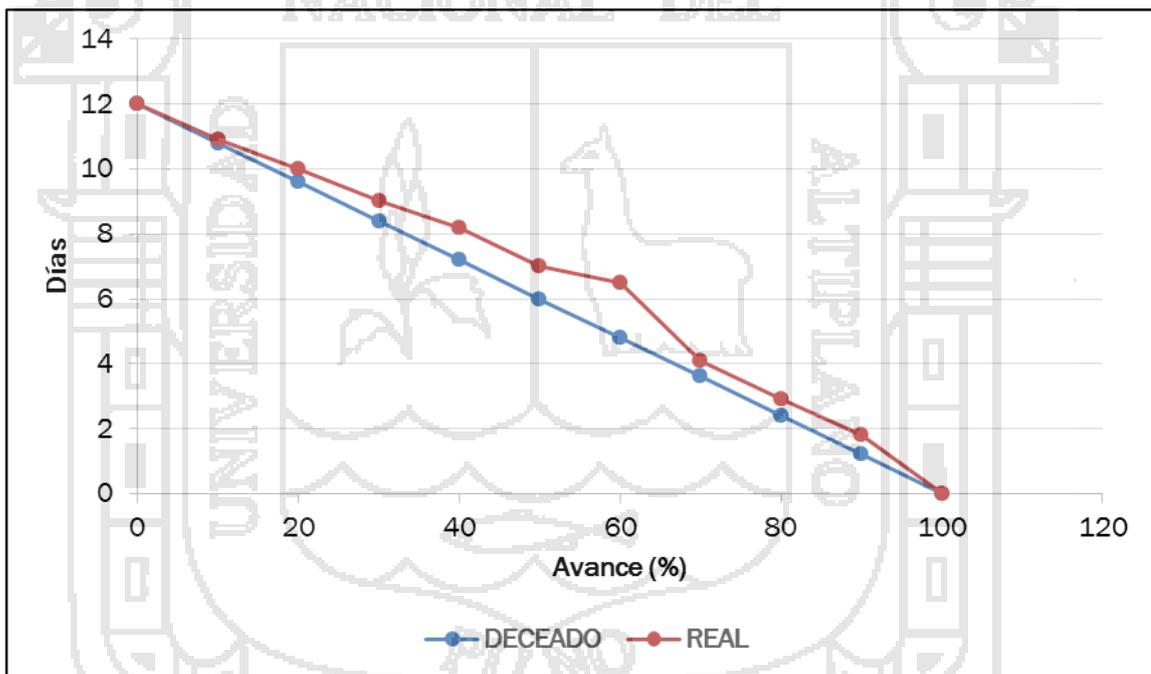
Fuente: Elaboración propia.

**Figura 47. El monitor desea que existan controles de validación en el ingreso de data evitando errores de tipeo**

PENDIENTE	EN PROCESO		TERMINADO
	ANÁLISIS (1)	DESARROLLO (1)	
		<div style="border: 1px solid black; background-color: #ADD8E6; padding: 5px; display: inline-block;">                     Crear código Mysql con especificaciones                 </div>	

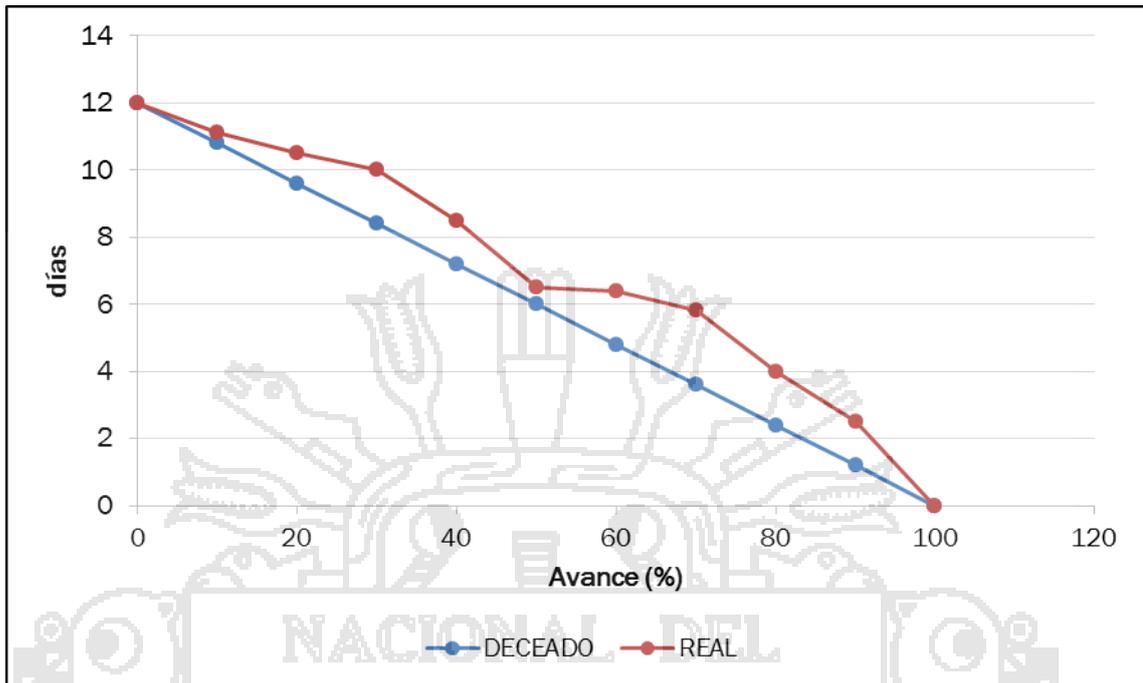
Fuente: Elaboración propia.

**Figura 48. Visualización proceso de desarrollo de Sprint Miembro 1**



Fuente: Elaboración propia.

**Figura 49. Visualización proceso de desarrollo de Sprint Miembro 2**



Fuente: Elaboración propia.

**Figura 50. El administrador desea generar reportes respecto a la información ingresada**

PENDIENTE	EN PROCESO		TERMINADO
	ANÁLISIS (1)	DESARROLLO (1)	
hacer pruebas de funcionalidad		crear código PHP y JQuery para que interactue con la BBDD	crear interfaz para este módulo

Fuente: Elaboración propia.

**Figura 51. El administrador desea hacer búsquedas fácilmente**

PENDIENTE	EN PROCESO		TERMINADO
	ANÁLISIS (1)	DESARROLLO (1)	
hacer pruebas de funcionalidad	crear código PHP y Jquery para que interactue con la BBDD	crear interfaz para este modulo	

Fuente: Elaboración propia.

**Figura 52. El administrador desea levantar las observaciones (réplicas)**

PENDIENTE	EN PROCESO		TERMINADO
	ANÁLISIS (1)	DESARROLLO (1)	
		hacer pruebas de funcionalidad	crear interfaz para este modulo crear código PHP y Jquery para que interactue con la BBDD

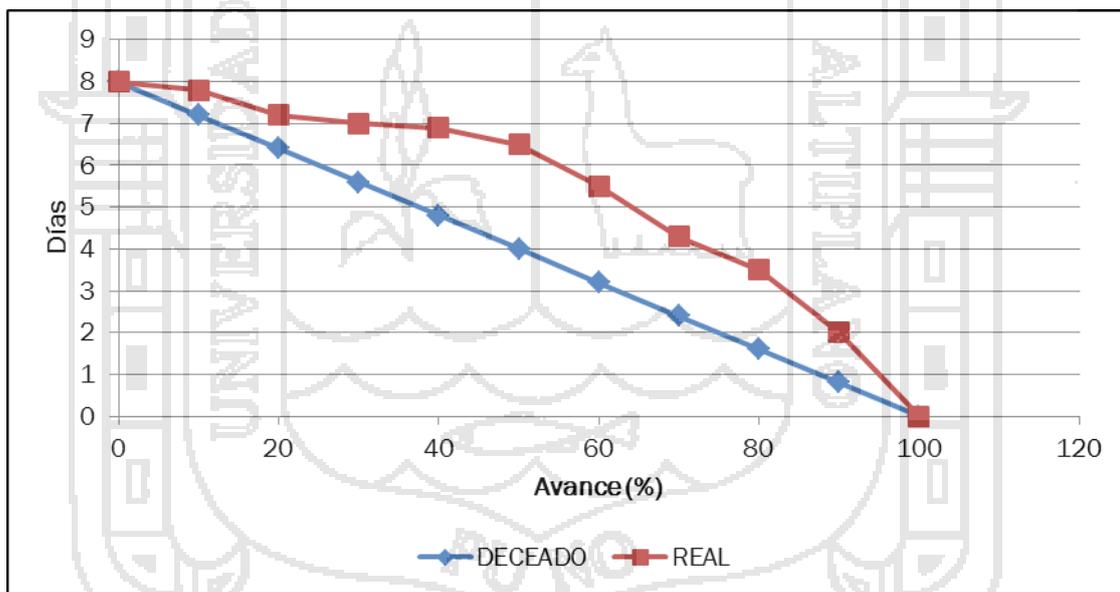
Fuente: Elaboración propia.

**Figura 53. Se desea investigar soluciones para generar reportes visuales**

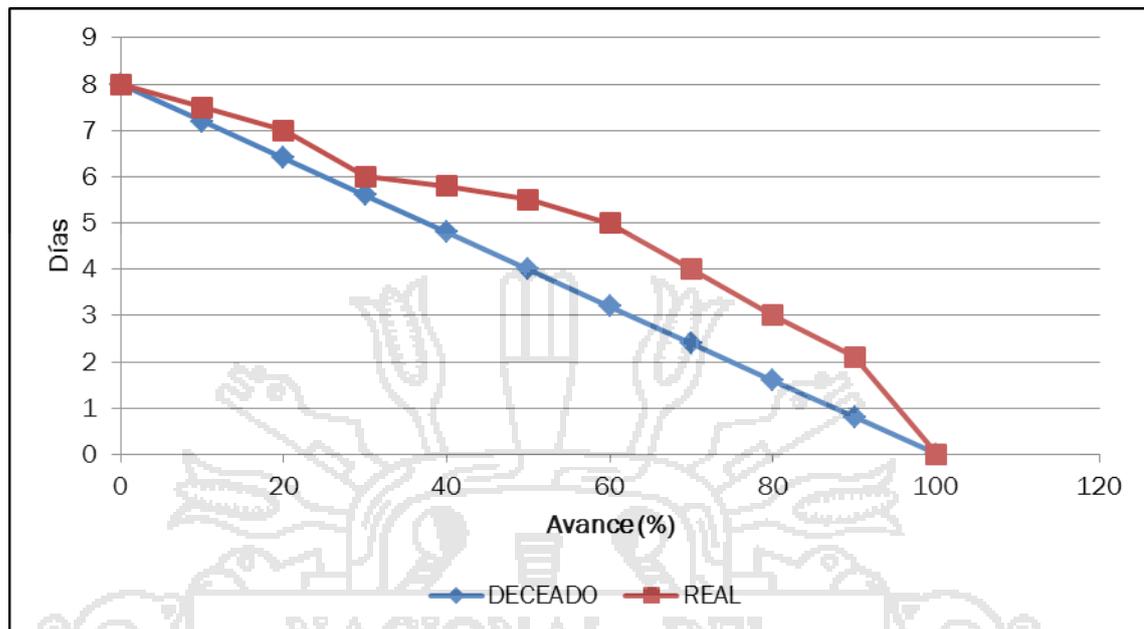
PENDIENTE	EN PROCESO		TERMINADO
	ANÁLISIS (1)	DESARROLLO (1)	
			<div style="border: 1px solid black; background-color: #ADD8E6; padding: 5px; margin-bottom: 5px;">crear interfaz para este modulo</div> <div style="border: 1px solid black; background-color: #ADD8E6; padding: 5px; margin-bottom: 5px;">crear codigo PHP y JQuery para que interactue con la BBDD</div> <div style="border: 1px solid black; background-color: #ADD8E6; padding: 5px;">hacer pruebas de funcionalidad</div>

Fuente: Elaboración propia.

**Figura 54. Visualización proceso de desarrollo de Sprint Miembro 1**



Fuente: Elaboración propia.

**Figura 55. Visualización proceso de desarrollo de Sprint Miembro 2**

Fuente: Elaboración propia.

#### 4.3. SCRUM COMPARADO CON KANBAN

En el presente trabajo de investigación se ha utilizado Kanban como consecuencia de haber utilizado Scrum, como por ejemplo la definición de los procesos que se ha trabajado aplicando la metodología Scrum permitiendo convertir los cuadros de Scrum a Kanban. El lead time o tiempo de progreso fue el mismo que el Sprint de Scrum.

Los tableros entre ambas metodologías son muy similares Kanban limita el WIP (Work in Progress - trabajo en curso) asigna límites concretos a cuántos elementos pueden estar en progreso en cada estado del flujo de trabajo.

Scrum es más prescriptivo que Kanban Podemos comparar herramientas viendo cuántas reglas proporcionan.

Prescriptivo significa "más reglas a seguir" y adaptativo significa "menos reglas a seguir".

Los métodos ágiles se denominan a veces los métodos ligeros, en concreto, porque son menos restrictivos que los métodos tradicionales. De hecho, el primer principio del Manifiesto Ágil es "Individuos e interacciones sobre procesos y herramientas".

Scrum y Kanban son muy adaptables, pero en términos relativos Scrum es más restrictivo que Kanban. Scrum da más limitaciones, y así deja menos opciones abiertas. Por ejemplo Scrum prescribe el uso de iteraciones de duración fija, Kanban no.

Kanban limita el WIP por estado en flujo de trabajo, Scrum limita el WIP por Iteración. En Scrum, la pila de sprint muestra qué tareas han de ser ejecutadas durante la iteración actual "sprint" en terminología Scrum.

Entonces, ¿cuál es la diferencia entre una pizarra de Scrum y una pizarra de Kanban? Vamos a empezar con un proyecto simple y comparar las dos:

En ambas metodologías estamos siguiendo un grupo de elementos a medida que avanzan a través de un flujo de trabajo. Hemos seleccionado tres estados: pendiente, en proceso, y terminado. Se puede elegir los estados deseados. No olvidemos que para estas metodologías "menos es más".

¿En Scrum no hay ninguna regla que impida que el equipo ponga todos los elementos en la columna en curso al mismo tiempo! Sin embargo, existe un límite implícito ya que la iteración en sí tiene un alcance fijo.

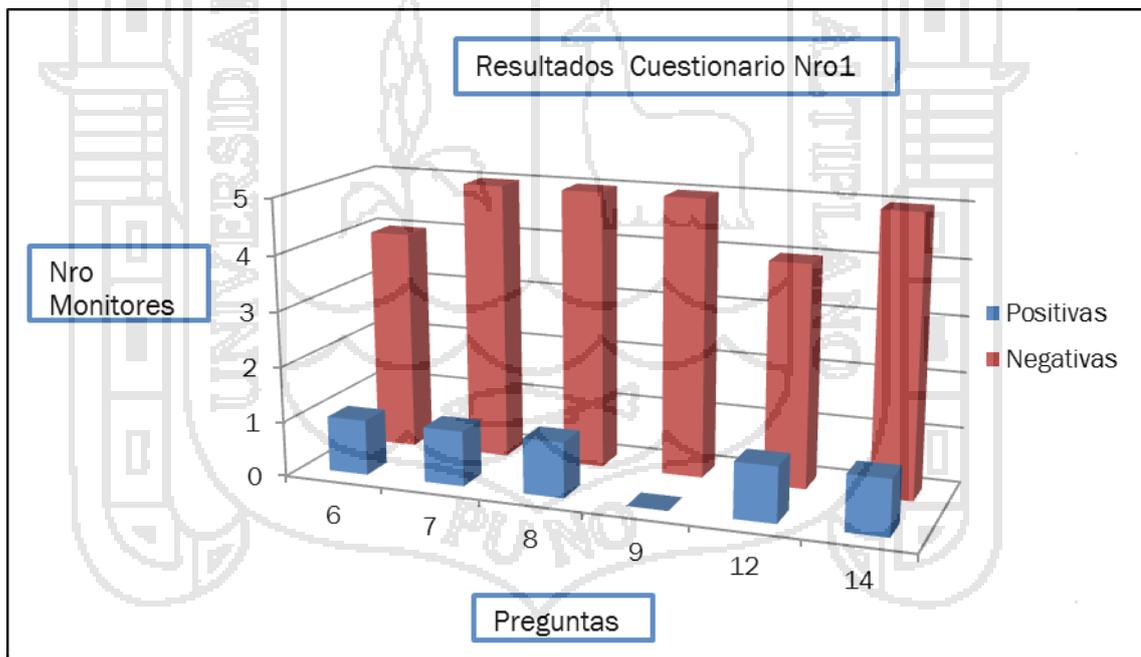
Así que Scrum limita el WIP indirectamente, mientras que Kanban limita el WIP directamente. La mayoría de los equipos de Scrum aprenden finalmente que es una mala idea tener demasiados elementos en curso, y desarrollan una cultura de intentar tener los elementos actuales terminados antes de comenzar con nuevos elementos.

De forma que en Scrum el WIP se limita por unidad de tiempo. En Kanban el WIP se limita por el estado del flujo de trabajo.

Tienes que elegir qué límite aplicar a qué estados del flujo de trabajo. Pero la idea general es limitar el WIP de todos los estados del flujo de trabajo, empezando por "lo más pronto posible" y terminando "lo más tarde posible" a lo largo de la cadena de valor. Una vez que tenemos los límites de WIP en su lugar, podemos empezar a medir y predecir el tiempo de entrega, es decir, el tiempo medio de un elemento para realizar todo el camino a través de la pizarra. Tener tiempos de entrega predecibles nos permite comprometer los SLA (acuerdos de nivel de servicio, en inglés service-level agreements) y hacer planes de entrega realistas.

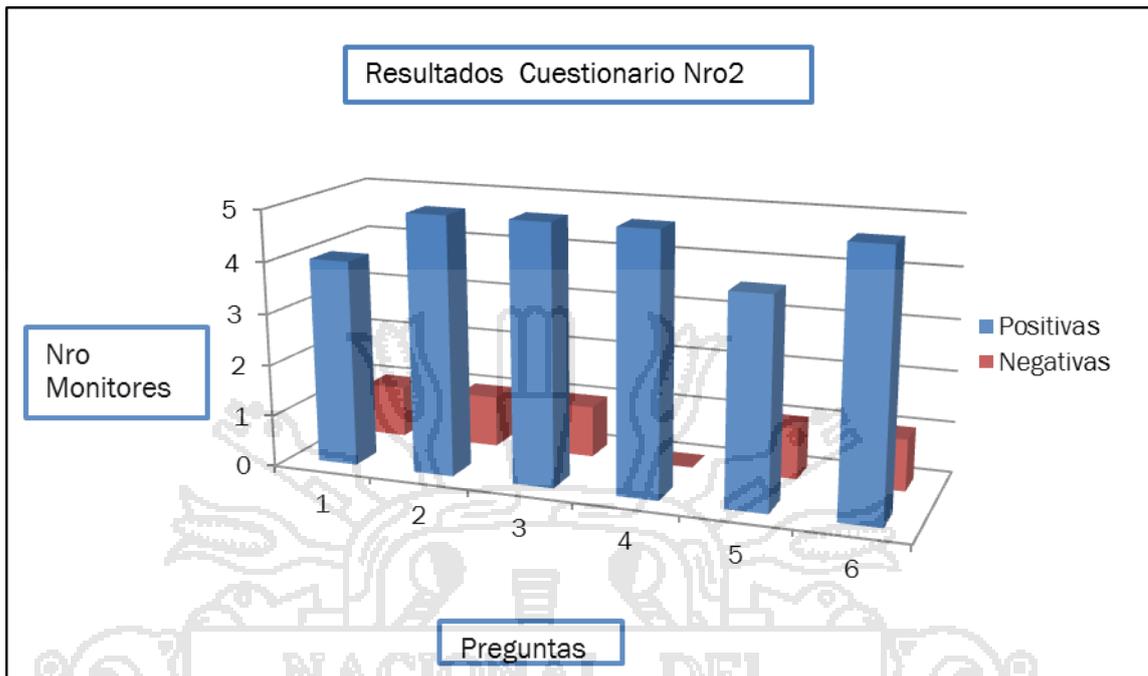
Las diferencias entre las dos metodologías se pueden observar en las siguientes figuras:

**Figura 56. Resultados Cuestionario tomado antes de la implantación del sistema**



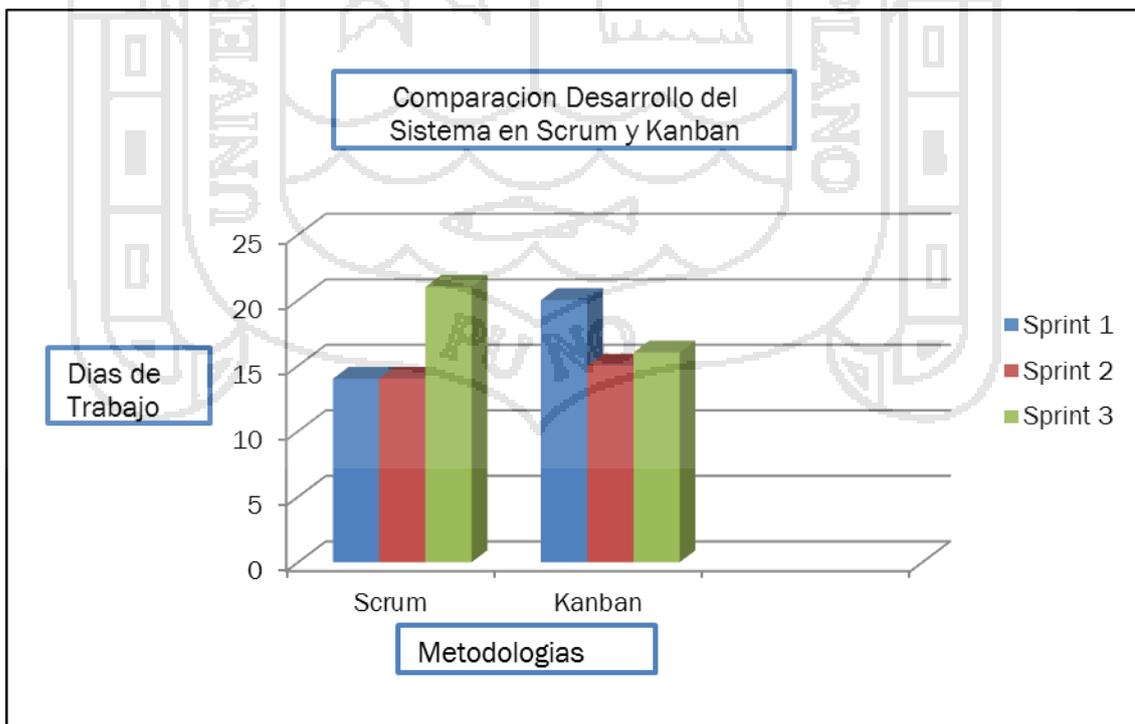
Fuente: Elaboración propia.

**Figura 57. Resultados Cuestionario tomado después de la implantación del sistema**



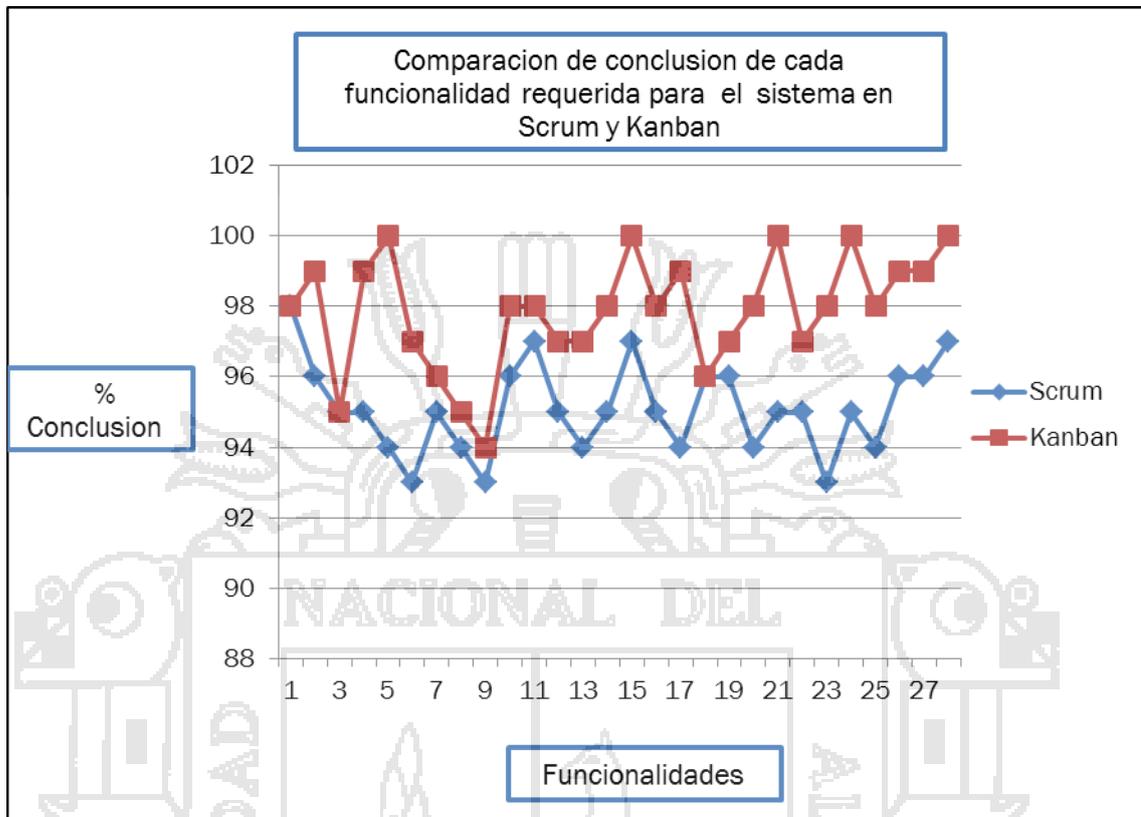
Fuente: Elaboración propia.

**Figura 58. Comparación de tiempo que requirió desarrollar el sistema con ambas metodologías**



Fuente: Elaboración propia.

**Figura 59. Comparación de conclusión de cada funcionalidad requeridas en el sistema usando Scrum y Kanban**



Fuente: Elaboración propia.

## CONCLUSIONES

**Primera:** El sistema de registro y procesamiento de datos del Programa de Monitoreo Socio Ambiental Participativo en el proyecto de exploración minera Chucapaca, comparando las metodologías ágiles Scrum y Kanban, logra optimizar los resultados en tiempo y calidad; además de ser una potencial herramienta de consultas y tratamiento de datos para la toma de decisiones gerenciales - ambientales.

**Segunda:** El análisis, diseño e implementación permitió poner en funcionamiento el sistema de registro y procesamiento de datos del Programa de Monitoreo Socio Ambiental Participativo (PMSAP) en el proyecto de exploración minera Chucapaca, para el ingreso de 6 fichas de evaluación, 2 tipos de búsquedas, 2 tipos de reportes, las ubicaciones satelitales de las zonas y la administración de usuarios.

**Tercera:** La comparación de ambas metodologías aplicado en el proceso de desarrollo del sistema registro y procesamiento de datos del Programa de Monitoreo Socio Ambiental Participativo (PMSAP) en el proyecto de exploración minera Chucapaca permitió conocer la medida en que Scrum y Kanban son muy adaptables, pero en términos relativos Scrum es más restrictivo que Kanban. Scrum da más limitaciones, y así deja menos opciones abiertas. Como cualquier herramienta, Scrum y Kanban no son ni perfectas ni completas. No dicen todo lo que se debe hacer, solo proporcionan ciertas restricciones y directrices. Scrum obliga a tener iteraciones de duración fija y equipos interdisciplinarios, y Kanban obliga a usar tableros visibles y a limitar el tamaño de colas. Curiosamente, el valor de una herramienta es que limita opciones. Una herramienta de proceso que permite hacer cualquier cosa no es muy útil.

Usar las herramientas adecuadas ayudará al equipo a triunfar, pero no garantizará el éxito. Es fácil confundir el éxito/fracaso del proyecto, con el éxito / fracaso de la herramienta (un proyecto puede triunfar debido a una gran herramienta, un proyecto puede triunfar a pesar de una pésima herramienta, un proyecto puede fallar debido a una pésima herramienta y un proyecto puede fallar a pesar de una gran herramienta).

**Cuarta:** El nivel de optimización del sistema de registro y procesamiento de datos del Programa de Monitoreo Socio Ambiental Participativo en el proyecto de exploración minera Chucapaca, ha sido muy bueno, según las encuestas realizadas a 3 monitores y un integrante del PMSAP.



## RECOMENDACIONES

**Primera.-** Se sugiere realizar mejoras al sistema propuesto en el presente trabajo de investigación de forma que se elimine el registro de data en fichas de papel y se haga mediante alguna alternativa de tecnología móvil.

**Segunda.-** Se sugiere ampliar el sistema de manera que involucre directamente a la mina Canteras del Hallazgo, ello permitiría una toma de decisiones a nivel superior involucrando no solo al PMSAP sino a la mina propiamente dicha.

**Tercera:** Se sugiere combinar las metodologías Scrum o Kanban con alguna otra que ayude en la orientación hacia el objetivo de manera más eficaz, por ejemplo escribir algunos elementos de la pila (en Scrum) como casos de uso (una práctica de RUP) o utilizar metodología XP haciendo reuniones diarias (práctica de Scrum), entre otras.

**BIBLIOGRAFÍA**

ANDERSON A.; EFFRIES, R. y HENDRICKSON, Chet. Scrum y XP desde las trincheras. Libro online disponible en:

<http://www.proyectalis.com/wp-content/uploads/2008/02/Scrum-y-xp-desde-las-trincheras.pdf>

CHÁVEZ, C. 2012. Evaluación de riesgos ambientales para sitios mineros: caso del distrito minero Santa María de la paz”. Universidad Autónoma San Luis de Potosí, Facultad de Ciencias Químicas, Ingeniería y Medicina.

COMUNIDAD CRISP. 2010. “Extreme Programming Explained: Embrace Change”. Disponible en: <http://www.crisp.se/Kanban>

DEEMER, P.; BENEFIELD, G.; LARMAN, C. y VODDE, B. 2009. The Scrum primer. Certified Scrum Training Worldwide. ISBN: 978-1-4303-2264-1. Traducción al castellano: Leo Antoli - Agile-Spain. Disponible en: <http://creativecommons.org/licenses/by-nc-nd/3.0/es/>

FUNDACIÓN PERUANA PARA LA CONSERVACIÓN DE LA NATURALEZA – PRO NATURALEZA. 2011. Memoria anual 2011. Programas de Monitoreo Socio Ambiental Participativo. Lima.

GETTING STARTED WITH KANBAN. 2012. Graphics Products Home Page. Recuperado en agosto de 2013. Disponible en: <http://www.graphicproducts.com/Kanban/Kanban-getting-started.php>

HERNANDEZ, R.; FERNANDEZ, C.; BAPTISTA, Pilar. 2006. Metodología de la Investigación. 4ta. ed. McGraw-Hill. 2006. 850p. ISBN: 978-970-10-5753-7.

INSTITUTO CARO Y CUERVO. 2009. Plan de Manejo Ambiental. 56 p. Disponible en:

[http://apoyoambiental.bligoo.com.co/media/users/19/973332/files/225766/plan\\_de\\_manejo\\_ambiental\\_CaraCuervo.pdf](http://apoyoambiental.bligoo.com.co/media/users/19/973332/files/225766/plan_de_manejo_ambiental_CaraCuervo.pdf)

JOYCE, D. 2009. Pulling Value: Lean and Kanban. 27 de Junio de 2009. Consultado de Systems Thinking, Lean and Kanban. Disponible en: <http://leanandKanban.files.wordpress.com/2009/06/pulling-value-lean-and-Kanban.pdf>.

KNIBERG, H. 2007. Prólogo de Jeff Sutherland y Mike Cohn. “Scrum y XP desde las trincheras”. Media, editor de InfoQ.com. Libro online. ISBN: 978-1-4303-2264-1. Traducción al castellano: Ángel Medinilla. Disponible en: <http://www.proyectalis.com/2008/02/26/Scrum-y-xp-desde-las-trincheras/>

KNIBERG, Henrik y SKARIN, Mattias. 2010. Prólogo de Mary Poppendieck & David Anderson. “Kanban y Scrum –obteniendo lo mejor de ambos”. © 2010 C4Media, editor de InfoQ.com. Libro online. ISBN: 978-0-557-13832-6. Traducción al castellano: Equipo de contenidos de Agile Spain ([www.agileSpain.com](http://www.agileSpain.com)). Disponible en: <http://www.proyectalis.com/2010/01/28/Scrum-vs-Kanban-en-castellano>

KNIGHT PIÉSOLD CONSULTING. 2010. Resumen ejecutivo. Proyecto de Exploración Chucapaca. Modificación del Estudio de Impacto Ambiental, Semidetallado Categoría II. Canteras del Hallazgo S.A.C. 40 p. Disponible en: [http://intranet2.minem.gob.pe/web/archivos/dgaam/inicio/resumen/RE\\_1983232.PDF](http://intranet2.minem.gob.pe/web/archivos/dgaam/inicio/resumen/RE_1983232.PDF)

LUNA, F. 2007. Sistema Remoto de Monitoreo Ambiental”. Universidad Nacional de Ingeniería. Facultad de ingeniería mecánica, Especialidad de ingeniería mecatrónica.

MINISTERIO DEL AMBIENTE. 2011. Ley General del Ambiente - Ley N° 28611. Ley Marco del Sistema Nacional de Gestión Ambiental - Ley N° 28245. Reglamento

de la Ley Marco del Sistema Nacional de Gestión Ambiental-Decreto Supremo N° 008 - 2005 – PCM. Ley de Creación, Organización y Funciones del Ministerio del Ambiente-Decreto Legislativo N° 1013. Disponible en:

<http://www.minam.gob.pe/wp-content/uploads/2013/06/ley-general-del-ambiente.pdf>.

MINISTERIO DE ADMINISTRACIONES PÚBLICAS. Métrica [en línea]. Versión 3.

Técnicas y Prácticas. Disponible en:

<http://www.csae.map.es/csi/metrica3/tecnicas.pdf>

O’SULLIVAN, M. 1999. Evaluación de Impacto Ambiental. En: Kiely G. Ingeniería Ambiental. Ed. McGraw-Hill. España. 1117 – 1150 pp.

REYES, I. 2008. Método de recolección de datos. Universidad de Carabobo [en línea]. [fecha de consulta: 17 Julio 2010].

SAES, P. 2013. Identificación y valoración de técnicas ágiles de gestión de proyectos software. Trabajo Fin de Master. Universidad de Oviedo. Departamento de explotación y prospección de minas. Master interuniversitario en dirección de proyectos. 120 p. Disponible en:

<http://digibuo.uniovi.es/dspace/bitstream/10651/18211/10/TFMPedroJSaezMartinezProteg.pdf>

SILBERSCHATZ , A.; KORTH , H.F.; y SUDARSHAN, S. 2002. Fundamentos de Bases de Datos. 4a. ed. Madrid : McGraw-Hill. 787 p. ISBN: 84-481-3654-3

TREJO, I. 2002. Módulo XML para acceder al Sistema Administrador de Bases de Datos SQLmx a través de Internet. Tesis (Magíster en Ciencias de la Computación). México D. F.: Instituto Politécnico Nacional, Centro de Investigación en Computación. 143 p.

WIKIMEDIA. 2010. Servidor HTTP Apache. WIKIPEDIA La enciclopedia libre [en línea]. Junio 2010 [fecha de consulta: 13 Julio 2012].

Disponible en: [http://es.wikipedia.org/wiki/Servidor\\_HTTP\\_Apache#cite\\_note-0](http://es.wikipedia.org/wiki/Servidor_HTTP_Apache#cite_note-0)

WIKIMEDIA. 2010. MySQL. WIKIPEDIA La enciclopedia libre [en línea]. Junio 2010 [fecha de consulta: 12 Julio 2010].

Disponible en: <http://es.wikipedia.org/wiki/MySQL>

WIKIMEDIA. 2010. PHP. WIKIPEDIA La enciclopedia libre [en línea]. Junio 2010 [fecha de consulta: 12 Julio 2010]. Disponible en:

<http://es.wikipedia.org/wiki/MySQL>

ZALDUMBIDE, J.P. 2012. Interfaz de servicios del Centro de Naturoterapia Reiki SPA, para Sistemas móviles IPHONE y IPAD. Proyecto de Investigación. Magister en gerencia de Proyectos. Escuela Politécnica del Ejército. 106 p. Disponible en: <http://repositorio.espe.edu.ec/bitstream/21000/5714/1/T-ESPE-033870.pdf>



## ANEXO A: CUESTIONARIOS

## A.1 CUESTIONARIO Nro. 01

*Sistema de información de registro y monitoreo socio ambiental participativo del proyecto de exploración minera Chucapaca*

*Entrevista Recopilación de Requisitos*

LUGAR		FECHA	
CARGO QUE OCUPA			
NOMBRE COMPLETO			

1. ¿Qué clase de usuarios requiere para el acceso al sistema?
2. ¿Cómo administrador que requisitos en el sistema desearía?
3. ¿Cómo monitor que requisitos en el sistema desearía?
4. ¿Investigar soluciones para generar reportes visuales
5. ¿Qué clase de interfaz desearía en el sistema?
6. ¿Cómo monitor deseo registrar las fichas de información?
7. ¿Qué tipo de validación para el acceso requiere para el sistema?

## A.1 CUESTIONARIO Nro. 02

*Sistema de información de registro y monitoreo socio ambiental participativo del proyecto de exploración minera Chucapaca*

LUGAR		FECHA	
CARGO QUE OCUPA			
NOMBRE COMPLETO			

1. ¿Cuántos monitores se tiene en operación?
2. ¿Cuál es el procesamiento de datos que se le da a la información obtenida?
3. ¿Qué problemática(s) presenta el proceso de lectura de monitores?
4. ¿utilizan alguna tecnología, dispositivo como herramienta en la tarea de registro de datos?
5. ¿Cuál es el horario de trabajo de los monitores?
6. ¿Cuánto tiempo se requiere para realizar la tarea de monitoreo?
7. ¿Cuánto tiempo demora vaciar los datos obtenidos y procesarlos?
8. ¿Cuántos días demora el procesamiento de datos? favor detallar los procesos
9. ¿Cuánto tiempo demora el traslado de la información al sitio encargado del procesamiento de datos?
10. ¿En la actualidad puede Ud. generar algún tipo de reporte con la información obtenida?
11. ¿Qué problemas ha detectado en el traslado de datos al sitio de procesamiento?

12. En la actualidad, ¿Como calificaría Ud. el proceso de evaluación de datos?
13. ¿Es necesario la revisión de información obtenida como parte de procesos anteriores?
14. ¿Considera eficiente el proceso de revisión de información obtenida?



**A.1 CUESTIONARIO Nro. 03**

*Sistema de información de registro y monitoreo socio ambiental participativo del proyecto de exploración minera Chucapaca*

LUGAR		FECHA	
CARGO QUE OCUPA			
NOMBRE COMPLETO			

1. ¿Cuánto tiempo el sitio de procesamiento demora en recibir los datos obtenidos por los monitores?
2. ¿Cuánto tiempo demora vaciar los datos obtenidos y procesarlos?
3. ¿Cómo se realiza el procesamiento de información captada por los monitores?
4. ¿Cuánto tiempo demora el proceso de procesamiento de datos?
5. ¿Qué problemas ha detectado en el traslado de datos al sitio de procesamiento?
6. en la actualidad, ¿Cómo calificaría Ud. el proceso de evaluación de datos?
7. ¿Cuánto tiempo toma consolidar la información recolectada en el sistema?
8. ¿Cómo realiza las acciones como resultado de la lectura de información brindada?
9. ¿Cómo realiza la revisión de información obtenida como parte de procesos anteriores?
10. ¿Se considera eficiente el proceso de revisión?
11. ¿es para usted más cómoda la recolección de datos utilizando fichas manuales o digitar las mismas?
12. ¿Cuáles son su sugerencias y recomendaciones para implementar un nuevo sistema monitoreo?

**ANEXO B: MAPA DE UBICACIÓN DE LA EXPLORACIÓN MINERA**

