****

**Universidad EAFIT**

**Escuela de Ingenierías**

**Departamento de Informática y Sistemas**

**Especialización en Desarrollo de Software**

**Ingeniería del Proceso Software**

**TRABAJO EVOLUTIVO – PRIMERA PARTE**

**Presentado a**

*Luis Fernando Londoño L.*

**Presentado por**

*Hans Linderman Parra Padilla*

*Juan David Cuartas León*

*Mario Andrés Galeano Montoya*

Medellín, Julio 24 de 2012

**Tabla de Contenido**

[1. Introducción 2](#_Toc333686062)

[2. Estado del arte 3](#_Toc333686063)

[2.1 Optimización de Procesos de Software 3](#_Toc333686064)

[2.1.1 Definiciones 3](#_Toc333686065)

[2.1.2 Mapa conceptual 5](#_Toc333686066)

[2.2 Innovación en Software 5](#_Toc333686067)

[2.2.1 Definiciones 5](#_Toc333686068)

[2.2.2 Mapa conceptual 6](#_Toc333686069)

[2.3 Compañías innovadoras 7](#_Toc333686070)

[2.3.1 A nivel global 7](#_Toc333686071)

[2.3.2 A nivel local 9](#_Toc333686072)

[3. Metodologías ágiles 11](#_Toc333686073)

[3.1 SCRUM 11](#_Toc333686074)

[3.1.1 Definición 11](#_Toc333686075)

[3.1.2 Fundamentos 11](#_Toc333686076)

[3.1.3 Roles 11](#_Toc333686077)

[3.1.4 Documentos 12](#_Toc333686078)

[3.1.5 Reuniones 13](#_Toc333686079)

[3.1.6 Mapa conceptual 14](#_Toc333686080)

[3.2 AUP 15](#_Toc333686081)

[3.2.1 Definición 15](#_Toc333686082)

[3.2.2 Ciclo de vida 15](#_Toc333686083)

[3.2.3 Fases 16](#_Toc333686084)

[3.2.4 Disciplinas 16](#_Toc333686085)

[3.2.5 Mapa conceptual 18](#_Toc333686086)

[3.3 Ventajas y desventajas de AUP/SCRUM en el proyecto integrador 19](#_Toc333686087)

[3.3.1 BVC Móvil 19](#_Toc333686088)

[3.3.2 Sistema de gestión porcina 21](#_Toc333686089)

[4. Conclusiones 23](#_Toc333686090)

[5. Referencias 24](#_Toc333686091)

# 1. Introducción

En este documento vamos a plasmar los temas de optimización e innovación, en donde explicaremos definiciones, características y sus aplicaciones en el mundo real. Así mismo daremos a conocer algunas organizaciones que tienen ideas innovadoras y de optimización.

La optimización nos expone la mejora de los procesos existentes para ser más competitivos en el mercado local y global. Si apuntamos hacia el campo de la informática, conllevamos a que el software requiera menos recursos para su construcción y funcione mejor, lo que obviamente resulta en una mayor rentabilidad, mejor calidad y así mismo una mayor aceptación por parte de los usuarios.

La innovación es un concepto muy importante en el mundo tecnológico, que introduce cambios y genera valor. En el ámbito empresarial, las ideas que generan valor son de tres tipos: gerenciales, organizativas, comerciales y tecnológicas. En esta última, se ha otorgado una mayor importancia, por ser las que pueden tener mayores y mejores consecuencias en las organizaciones y en el mercado global. La innovación en las empresas va de la mano de la gestión del conocimiento, ya que al realizar un manejo adecuado del conocimiento (transferencia), este se va a diseminar mucho más fácil entre los empleados de la compañía, lo que podría facilitar la innovación.

Luego daremos un vistazo general a las metodologías ágiles de desarrollo de software como SCRUM y AUP, con el objetivo de tomarlas como punto de partida para la realización de nuestro proyecto integrador de la Especialización en Desarrollo de Software.

Nuestro equipo de trabajo, está divido en dos proyectos integradores diferentes: BVC Móvil, liderado por los integrantes Hans Parra y Juan Cuartas, y Sistema de Gestión porcino liderado por el integrante Mario Galeano. En los último apartes del presente trabajo, explicaremos las ventajas/desventajas de aplicar las diferentes metodologías ágiles (SCRUM/AUP) para la realización de estos proyectos.

# 2. Estado del arte

## 2.1 Optimización de Procesos de Software

### 2.1.1 Definiciones

Podemos definir la optimización de procesos de software como un intento de cambiar la forma en que la gente ejecuta las actividades para satisfacer mejor los objetivos de una empresa.

Un proceso de mejora tiene que generar cambios en la compañía. Este cambio es necesario administrarlo, planificarlo, comunicarlo a los interesados y organizar el cambio mediante la identificación de personas responsables de la implementación del cambio.

La optimización de procesos, surge de la necesidad de las empresas por ser más competitivas [1]. La capacidad para optimizar o mejorar un proceso, depende de la capacidad para controlarlo y para esto se necesita de mediciones confiables. Una optimización industrial exitosa implica un enfoque estratégico que abarque la siguiente cadena:

* Medición
* Control
* Optimización

La optimización de procesos se constituye como una de las principales herramientas cuantitativas en la toma de decisiones industriales [2]. Algunos de los objetivos más comunes de optimización son:

* Minimizar los costos
* Maximizar el rendimiento
* Maximizar la eficiencia

Desde una perspectiva general, Lorenz T. Biegler, en su libro sobre Programación Lineal, define la optimización de procesos de la siguiente manera: “Dado un sistema o proceso, encontrar la mejor solución a este proceso dentro de sus limitaciones” [3]. El autor plantea los siguientes elementos necesarios para llevar a cabo una tarea de optimización:

* Una función objetivo cuantificable que necesite ser maximizada o minimizada.
* Un modelo predictivo que describa el comportamiento del sistema.
* Un ajuste de las variables del modelo predictivo, para satisfacer las restricciones.

En muchos casos, los procesos de optimización se realizan a través de prueba y error, por lo que es necesario un enfoque sistemático que elimine el camino tedioso, y garantice en cierto grado la mejor solución posible. Este enfoque sistemático, se puede abordar desde diferentes niveles que deberían complementarse entre sí, pero que frecuentemente son considerados por separado en las siguientes disciplinas:

* **Programación matemática**: Se centra en la comprensión de las propiedades fundamentales de los problemas de optimización y algoritmos.
* **Computación científica**: Está fuertemente influenciada por las propiedades matemáticas, y la aplicación del método de optimización para uso eficiente y práctico.
* **Investigación de operaciones**: Se centra en la formulación del problema de optimización y desarrollo de estrategias de solución sobre modelos bien establecidos.
* **Ingeniería**: Las estrategias de optimización se aplican a retos, y a problemas del mundo real mal definidos, acoplándose con la eficiencia y fiabilidad de los métodos aplicables.

Ahora, desde una perspectiva más específica de la Ingeniería de Software, Masao Ito, en su publicación sobre los Elementos en la Optimización de Procesos de Software [4], aborda el enfoque de la optimización en función del chequeo constante del objetivo del producto, y la reorganización del proceso software en función de ello. Se sabe que el objetivo definido en la especificación de un producto, su costo y su calidad, son vulnerables de cambio por varias razones, por ejemplo, si la competencia va a lanzar un producto similar al de mi compañía en un tiempo menor y a un menor coste, entonces debo replantear los objetivos de mi producto.

Si los objetivos de un producto de software son fijos, el control de su proceso puede ser más fácil, definiéndolo en una etapa temprana, pero como normalmente los objetivos cambian, no se puede tener un proceso estrictamente definido para un objetivo. La esencia del problema de optimización radica en la capacidad de predecir la posición del objetivo final y en la comprobación constante del objetivo del producto para reorganizar el proceso en función de ello.

Masao Ito, propone un marco de referencia sistematizado (SPO framework) para hacer el proceso de desarrollo de software optimizado, basado en los siguientes elementos:

* **Ingeniería de software**: Es el conocimiento sistematizado sobre la Ingeniería de Software.
* **Gestión organizacional**: Es la técnica de gestión de la organización respaldada por algunos métodos.
* **Técnicas de procesos**: Son las técnicas para manejar el problema del proceso de software, por sí mismo.
* **Modelo base**: Es un meta modelo que sirve como base para la modificación del proceso continuo.
* **Funciones de autocomunicación**: Abordan la auditoria de los procesos y algunas actividades de ajuste.

Algunos modelos sobre los cuales nos podemos apoyar [5]:

* ***ISO 9001:2000:*** Propone un enfoque basado en procesos al desarrollar, implementar y mejorar el desempeño de un Sistema de Gestión de la Calidad
* ***CMMI:*** Es un marco de referencia desarrollado por el SEI para determinar el nivel de madurez que han alcanzado las organizaciones en función del nivel de capacidad de sus procesos involucrados en el desarrollo de software.
* **MR.MPS**: Es un modelo de mejora y evaluación de proceso de software para las micro, pequeñas y medianas empresas relacionadas a la industria del software del Brasil.

### 2.1.2 Mapa conceptual

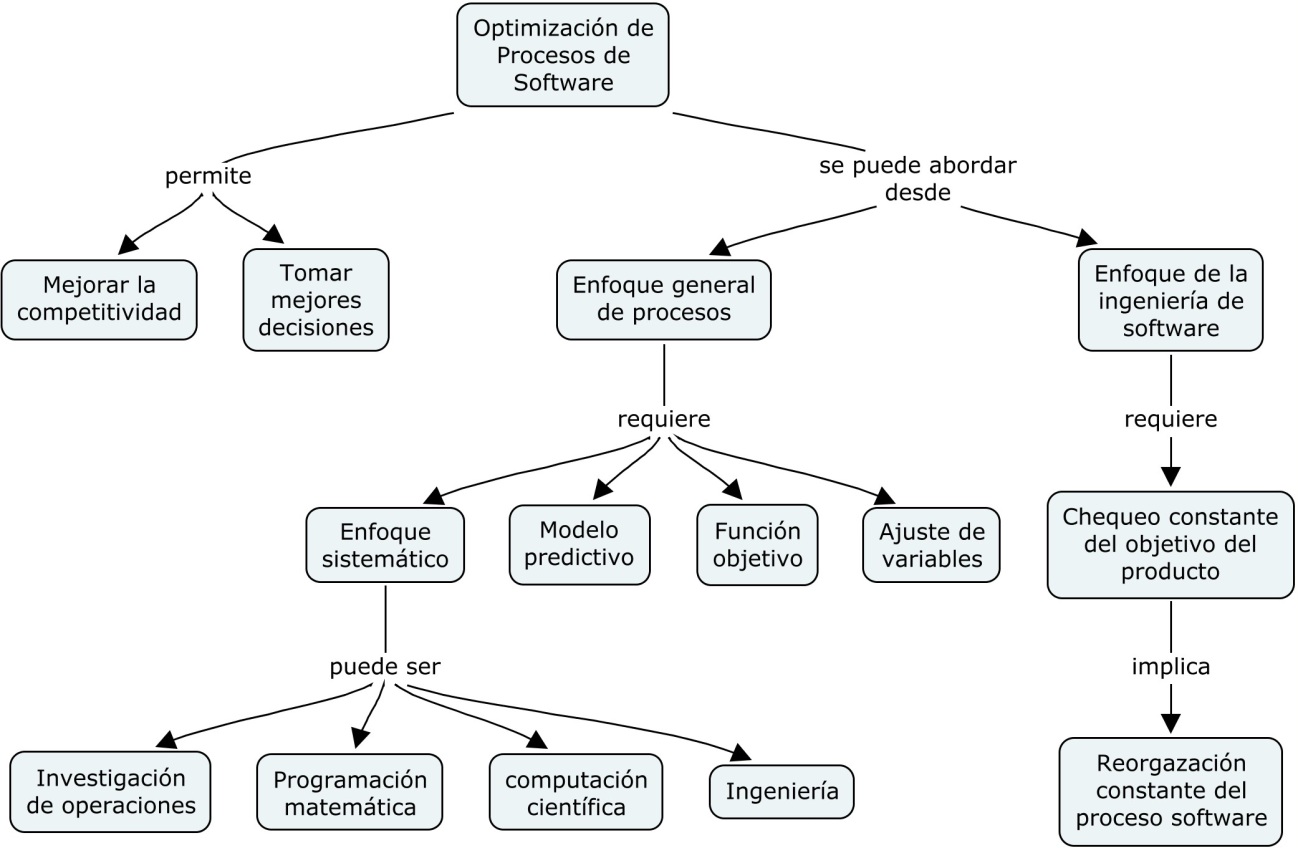


Figura 1. Optimización de Procesos de Software

## 2.2 Innovación en Software

### 2.2.1 Definiciones

Empecemos por definir innovación. Según el diccionario de la Real Academia Española, es la modificación de un producto, y su introducción en un mercado.

Innovar proviene del latín innovare, que significa acto o efecto de innovar, tornarse nuevo o renovar, introducir una novedad [1].

Introducción de un nuevo método de producción o metodología organizativa [2].

Una frase muy importante: *“La investigación científica, la innovación y el desarrollo tecnológico son el único camino que tiene un país para salir de la recesión en un mundo global e hiperconectado”.* José Pardin [3].

Software innovation es el proceso que busca el cambio y mejora a la hora de construir software. Software innovation no es software innovador, es mejorar, cambiar y volver cada vez mejor el proceso de creación de software. Este proceso es el que agrupa cada una de las actividades que tarde o temprano nos da un resultado, en este caso un producto o software.

La innovación es el elemento clave de la competitividad. Es crear o modificar un producto e introducirlo en el mercado

Según Michael Porter: *"La competitividad de una nación depende de la capacidad de su industria para innovar y mejorar. Las empresas consiguen ventajas competitivas si consiguen innovar"*

Existen distintos tipos de innovación, todos ellos se clasifican según su aplicación o su grado de originalidad:

Según su aplicación:

* **Innovación de producto**: Comercialización de un producto tecnológicamente distinto o mejorado, la innovación se da cuando las características de un producto cambian
* **Innovación de proceso**: Ocurre cuando hay un cambio significativo en la tecnología de producción de un producto o servicio también ocurre cuando se producen cambios significativos en el sistema de dirección y/o métodos de organización; reingeniería de procesos, planificación estratégica, control de calidad, etc...

Según su grado de originalidad:

* **Innovación radical**: aplicaciones nuevas de una tecnología o combinación original de nuevas tecnologías.
* **Innovación incremental**: mejoras que se realizan sobre un producto, servicio o método existente.

Según su nivel:

* Corrección de errores
* Mejoramiento continuo
* Mejoramiento con algún impacto
* Productos con diferenciadores
* Diferenciadores que son sorprendentes
* Generan conocimiento

### 2.2.2 Mapa conceptual

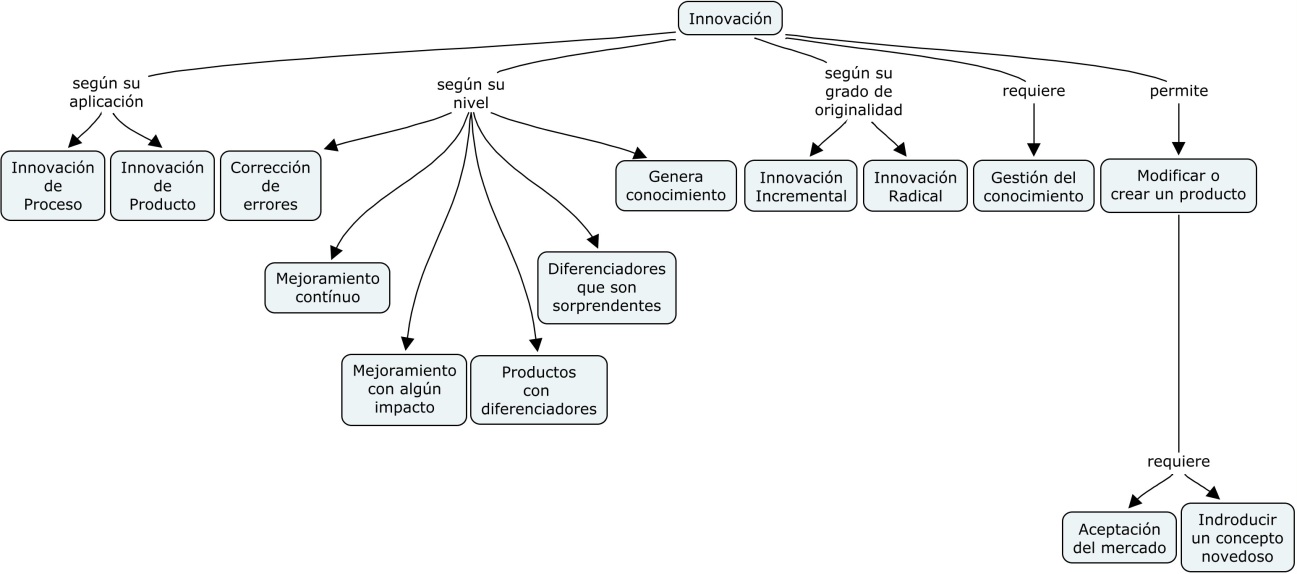


Figura 2. Innovación en Software

## 2.3 Compañías innovadoras

### 2.3.1 A nivel global

#### 2.3.1.1 Apple

Apple está construyendo una de las carreras financieras más impresionantes en la historia mundial, vendiendo más de 375 millones de dispositivos digitales en los últimos 5 años, con un promedio de ingresos mensuales de casi 4 mil millones de dólares, una acumulación en efectivo de 110 mil millones y una capitalización bursátil de 623 mil millones es actualmente la empresa más importante del mundo [1].

Gracias a una combinación de anticipación y suerte, Apple, que se fue casi a la quiebra antes del boom de las burbujas tecnológicas, está listo para alcanzar alturas aún mayores si puede continuar pensando e innovando como hasta ahora.

Las tres principales fuerzas que han empujado la compañía hacia adelante han sido:

* **Innovación**: Una de las principales lecciones que dejó Steve Jobs es que se debe anticipar la acción clave, y evitar el golpe tardío. En lugar de confiar en lo que el cliente dice que quiere, Steve Jobs insistió en definir las necesidades de los consumidores que no sabían lo que querían. Millones de personas quizás estaban satisfechas con los reproductores analógicos de Sony Walkman, pero eso fue antes de que el iPod y iTunes revolucionaran la industria musical y la forma en que escuchamos música.
* **Oportunidad**: Tres tendencias en curso han creado el ambiente ideal para que Apple continúe su asenso: El auge de los teléfonos inteligentes y las tabletas, y el crecimiento del mercado asiático en donde Apple está penetrando fuertemente.
* **Ejecución**: El actual CEO de Apple Tim Cook, quien asumió el frente de la compañía después de la muerte de Steve Jobs, ha realizado una labor excepcional como director de operaciones y tuvo un papel clave en el perfeccionamiento de la cadena de suministro de la compañía. Su papel clave en la empresa, ha sido ejecutar la visión de Steve Jobs y ha tenido tanto éxito que ahora Apple se encuentra en una posición inusual y envidiable: No puede producir suficientes cantidades para satisfacer la demanda de sus consumidores.

#### 2.3.1.2 Google

El presidente de la compañía, Eric Schmidt afirma que: "Google probó que la innovación puede ser sistematizada" y que esto significa "crear un entorno en el que se pregunte por qué las cosas son como son y si se pueden hacer de una manera diferente" [2].

Google ha logrado una fortuna que a todo el mundo le gustaría emular: $ 87.000 dólares en ganancias por cada empleado, y ha hecho un trabajo impresionante en innovación y crecimiento continuo. Por el otro lado, Google es una compañía realmente pequeña, con apenas 30.000 empleados, en comparación con otras empresas que tienen que poner millones de personas menos calificadas para trabajar. La mayoría de sus empleados son más listos que el resto de empleados de otras empresas.

Estos son los factores que han permitido hacer de Google, una de las compañías más innovadoras en el mundo [3]:

* **Cultura organizacional**: Los fundadores de la empresa trajeron desde el principio una cultura que ha sido continuamente desarrollada para dirigir a toda la compañía en la dirección de la innovación constante.
* **Selección de personas**: Es impresionante ver el enfoque que Google tiene cuando se trata de llevar a las personas adecuadas a la empresa, destinando grandes cantidades de recursos a la selección de personal.
* **Valores**: La organización está estructurada para ayudar a sus individuos a realizar y crear innovaciones en consonancia con la filosofía de la empresa. Ser "Googley" significa que un empleado se comporta de acuerdo con los valores de la compañía. Estos valores están documentados a partir de 11 principios, 3 de los cuales tienen una pasión por cambiar el mundo a través de internet.

#### DropBox

Fue fundada por Drew Houston. Este concibió la idea tras olvidar constantemente su memoria USB mientras era estudiante en el MIT. Él decía que los servicios existentes sufrían problemas de latencia, errores o eran muy complicados. Desde esto el empezó a crear una solución para el mismo pero se dio cuenta que podría beneficiar a otros con el mismo problema. Fundó Dropbox Inc en junio de 2007 [4].

Su modelo de negocio se cataloga como *freemium*, es un modelo de negocios que funciona ofreciendo servicios básicos gratuitos, mientras se cobra por otros más avanzados o especiales

#### 2.3.1.4 Otras compañías

A continuación, presentamos la lista de las 50 compañías más innovadoras del mundo en el año 2012, de acuerdo a la revista Fast Company, para su referencia [5].

|  |  |  |  |
| --- | --- | --- | --- |
| **Ranking** | **Empresa** | **Ranking** | **Empresa** |
| **1** | Apple | **26** | LegalZoom |
| **2** | Facebook | **27** | Tapjoy |
| **3** | Google | **28** | Polyvore |
| **4** | Amazon | **29** | Red Bull Media House |
| **5** | Square | **30** | LinkedIn |
| **6** | Twitter | **31** | Liquid Robotics |
| **7** | Occupy Movement | **32** | Gogo |
| **8** | Tencent | **33** | Bug Agentes Biológicos |
| **9** | Life Technologies | **34** | Chipotle |
| **10** | SolarCity | **35** | James Corner Field Operations |
| **11** | HBO | **36** | Narayana Hrudayalaya Hospitals |
| **12** | Southern New Hampshire U. | **37** | Recyclebank |
| **13** | Tesla Motors | **38** | UPS |
| **14** | Patagonia | **39** | Networked Insights |
| **15** | NFL | **40** | Chobani |
| **16** | National Marrow Donor Program | **41** | Kickstarter |
| **17** | Greenbox | **42** | SoundCloud |
| **18** | Jawbone | **43** | PayPal |
| **19** | Airbnb | **44** | Berg |
| **20** | 72andSunny | **45** | Boo-box |
| **21** | Siemens AG | **46** | Amyris |
| **22** | Dropbox | **47** | Knewton |
| **23** | Kiva Systems | **48** | RedBus |
| **24** | Starbucks | **49** | OpenSky |
| **25** | Genentech | **50** | Y Combinator |

Tabla 1. Compañías innovadoras

### 2.3.2 A nivel local

#### 2.3.2.1 Liberty Seguros [6]

El Director de Gestión de Conocimiento Alejandro Moreno Salamanca, define la Gestión de Conocimiento como “Un proceso continuo de adquisición, distribución, análisis y administración de la información organizacional, para obtener trabajadores más competitivos que tomen mejores decisiones y den una respuesta más rápida a las necesidades del mercado, logrando un desarrollo sostenible y una mayor productividad y rentabilidad corporativas”. Como Director de este proceso expone que “El valor agregado de la Organización es la gente” y en ese mismo sentido sostiene que “el conocimiento es el camino hacia la productividad”.

La Gestión de conocimiento toma en cuenta a todos los públicos de Liberty Seguros, incluyendo los canales de Distribución -que incluye a los Gerentes y Gestores Comerciales- los funcionarios de la Oficina Principal, los proveedores y los clientes asegurados. De esta manera todo el negocio está en función de la estrategia principal. “La estrategia de Liberty en Colombia está enfocada totalmente hacia el Cliente Distribuidor ofreciéndole soluciones creativas en Productos y Servicios de Calidad Superior”.

El conocimiento virtual se transmite mediante cursos de e-learning, juegos, simulaciones y certificaciones. Del mismo modo, a través de la modalidad presencial se promueven cátedras, casos y talleres, en un programa que se ha denominado Liberty University. También existe un plan de capacitación que apunta al mejoramiento de las competencias técnicas, administrativas, comerciales, gerenciales y empresariales de los funcionarios de la organización.

#### 2.3.2.2 Ecopetrol [7]

Ecopetrol ha dispuesto en el interior de su compañía una serie de herramientas tecnológicas que soportan las estrategias de Gestión de Conocimiento y aprendizaje al interior de la empresa, a través de una plataforma en ambiente web 2.0 que facilita el trabajo colaborativo en toda la organización. Así, los usuarios tienen la oportunidad de generar flujos de trabajo y aprobación, creación de blogs, encuestas y foros de discusión, permitiendo relaciones más amigables, ahorro de tiempo y mayor productividad. Dentro de estas herramientas, se encuentran:

* **Mi Sitio**: Con más de 4.600 usuarios, es el minisite en la intranet para compartir y administrar información e intereses, conocer la experiencia y experticia del personal de la empresa y hacer parte de la red de contactos de Ecopetrol. Entre 2005 y julio de 2011 pasó de 48 mil visitas anuales a 183 mil.
* **Lecciones Aprendidas**: Sitio en la intranet donde se publican las lecciones aprendidas documentadas por múltiples equipos de trabajo para compartir sus experiencias, aprender de ellas y mejorar actividades similares en el futuro. A la fecha han sido publicadas más de 1.400 lecciones sobre diversos procesos de la organización y el número de visitas anuales ha pasado de 35 mil en 2006 a más de 145 mil a julio de 2011.
* **Sitios de Comunidades de Práctica**: Ecopetrol brinda a sus comunidades de práctica espacios virtuales en su intranet para facilitar la interacción de sus integrantes, compartir información y establecer flujos de trabajo. Además, cuenta con funcionalidades como foros de discusión, blogs y wikis.
* **Sitios de Foros de Entorno Tecnológico**: Contienen la información y memorias de los Foros de Entorno Tecnológico, uno de los mecanismos definidos en el ciclo de planeación de Ecopetrol para identificar tendencias y cambios que puedan afectar el desempeño del negocio, y para plantear nuevas oportunidades a mediano plazo.

# 3. Metodologías ágiles

## 3.1 SCRUM

3.1.1 Definición[1]

Scrum es un proceso en el que se aplican de manera regular un conjunto de buenas 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. En Scrum se realizan entregas parciales y regulares del producto final, priorizadas por el beneficio que aportan al receptor del proyecto. Por ello, Scrum está especialmente indicado para proyectos en entornos complejos, donde se necesita obtener resultados pronto, donde los requisitos son cambiantes o poco definidos, donde la innovación, la competitividad, la flexibilidad y la productividad son fundamentales.

3.1.2 Fundamentos[2]

* El desarrollo incremental de los requisitos del proyecto en bloques temporales cortos y fijos (iteraciones de un mes natural y hasta de dos semanas, si así se necesita).
* La priorización de los requisitos por valor para el cliente y coste de desarrollo en cada iteración.
* El control empírico del proyecto. Por un lado, al final de cada iteración se demuestra al cliente el resultado real obtenido, de manera que pueda tomar las decisiones necesarias en función de lo que observa y del contexto del proyecto en ese momento. Por otro lado, el equipo se sincroniza diariamente y realiza las adaptaciones necesarias.
* La potenciación del equipo, que se compromete a entregar unos requisitos y para ello se le otorga la autoridad necesaria para organizar su trabajo.
* La sistematización de la colaboración y la comunicación tanto entre el equipo y como con el cliente.
* El timeboxing de las actividades del proyecto, para ayudar a la toma de decisiones y conseguir resultados.

### 3.1.3 Roles

**Product Owner** [3]**:** El rol del Product Owner, puede venir de parte del cliente o dentro de la empresa misma. Generalmente no se aconseja que el Product Owner sea parte también del equipo de desarrollo, o el Scrum Master mismo, sus intereses se pueden ver contradichos, pero esto puede variar según el caso. Por defecto, probablemente venga de parte del cliente. Según la definición de la Scrum Alliance de los roles en Scrum, los roles del Product Owner son:

* Definir las características del producto;
* Decidir sobre las fechas de lanzamiento y contenido;
* Ser responsable de la rentabilidad del producto (ROI);
* Priorizar las características según el valor de mercado;
* Ajustar las características y prioridades por iteración, según sea necesario; y
* Aceptar o rechazar resultados del trabajo.

**Scrum Master** [4]**:** El trabajo del Scrum Master es quitar cualquier impedimento, interno o externo al equipo que le impida alcanzar el objetivo de construir el software que comprometieron al inicio del Sprint., y es el responsable de:

* Mejorar la vida y productividad de los miembros del equipo, facilitándole la creatividad de cualquier manera posible.
* Asegurar la cooperación de todos los roles, y quitar cualquier barrera que lo impida.
* Proteger al equipo de interferencias externas, y quitar impedimentos.
* Asegurar que se siga el proceso
* Invitar a personas al Scrum diario, y a reuniones de revisión del sprint y planificación
* Seguir la Lista De Impedimentos, quitando las barreras entre el desarrollo y el cliente, de manera que el cliente maneje directamente la funcionalidad desarrollada.
* Enseñarle al cliente cómo maximizar su retorno de inversión y cumplir sus objetivos con Scrum
* Mejorar las prácticas y herramientas usadas, de manera que cada incremento de funcionalidad sea potencialmente productivo.

**Equipo de desarrollo:** El equipo del proyecto es un grupo multifuncional de personas con todas las habilidades diferentes que son necesarias para convertir los requerimientos en algo que es un incremento de una funcionalidad potencialmente productiva. Un pequeño equipo de 3 a 9 personas con las habilidades transversales necesarias para realizar el trabajo (análisis, diseño, desarrollo, pruebas, documentación, etc.).

**Stakeholders:** Se refiere a la gente que hace posible el proyecto y para quienes el proyecto producirá el beneficio acordado que justifica su producción. Sólo participan directamente durante las revisiones del sprint.

### 3.1.4 Documentos

**Product backlog** [5]**:** La lista de objetivos/requisitos priorizada representa la visión y expectativas del cliente respecto a los objetivos y entregas del producto o proyecto. El cliente es el responsable de crear y gestionar la lista (con la ayuda del Facilitador y del equipo, quien proporciona el coste estimado de completar cada requisito). Dado que reflejar las expectativas del cliente, esta lista permite involucrarle en la dirección de los resultados del producto o proyecto.

En la lista se indican las posibles iteraciones y las entregas (releases) esperadas por el cliente (los puntos en los cuales desea que se le entreguen los objetivos/requisitos completados hasta ese momento), en función de la velocidad de desarrollo del (los) equipo(s) que trabajará(n) en el proyecto. Es conveniente que el contenido de cada iteración tenga una coherencia, de manera que se reduzca el esfuerzo de completar todos sus objetivos.

**Sprint backlog** [7]: Lista de tareas que el equipo elabora en la reunión de planificación de la iteración (Sprint planning) como plan para completar los objetivos/requisitos seleccionados para la iteración y que se compromete a demostrar al cliente al finalizar la iteración, en forma de incremento de producto preparado para ser entregado. Esta lista permite ver las tareas donde el equipo está teniendo problemas y no avanza, con lo que le permite tomar decisiones al respecto. Para cada uno de los objetivos/requisitos se muestran sus tareas, el esfuerzo pendiente para finalizarlas y la auto asignación que han hecho los miembros del equipo.

**Burn down** [8]**:** Un gráfico de trabajo pendiente a lo largo del tiempo muestra la velocidad a la que se está completando los objetivos/requisitos. Permite extrapolar si el Equipo podrá completar el trabajo en el tiempo estimado. Se pueden utilizan los siguientes gráficos de esfuerzo pendiente:

* Días pendientes para completar los requisitos del producto o proyecto (product burndown chart), realizado a partir de la lista de requisitos priorizada (Product Backlog).
* Horas pendientes para completar las tareas de la iteración (sprint burndown chart), realizado a partir de la lista de tareas de la iteración (Iteration Backlog).

### 3.1.5 Reuniones

**Sprint Planning** [9]: La planificación de las tareas a realizar en la iteración se divide en dos partes:

Primera parte de la reunión. Se realiza en un timebox de cómo máximo 4 horas:

* El cliente presenta al equipo la lista de requisitos priorizada del producto o proyecto, pone nombre a la meta de la iteración (de manera que ayude a tomar decisiones durante su ejecución) y propone los requisitos más prioritarios a desarrollar en ella.
* El equipo examina la lista, pregunta al cliente las dudas que le surgen, añade más condiciones de satisfacción y selecciona los objetivos/requisitos más prioritarios que se compromete a completar en la iteración, de manera que puedan ser entregados si el cliente lo solicita.

Segunda parte de la reunión. Se realiza en un timebox de cómo máximo 4 horas:

* El equipo planifica la iteración, elabora la táctica que le permitirá conseguir el mejor resultado posible con el mínimo esfuerzo. Esta actividad la realiza el equipo dado que ha adquirido un compromiso, es el responsable de organizar su trabajo y es quien mejor conoce cómo realizarlo.
* Define las tareas necesarias para poder completar cada objetivo/requisito, creando la lista de tareas de la iteración (Sprint Backlog) basándose en la definición de completado.
* Realiza una estimación conjunta del esfuerzo necesario para realizar cada tarea.
* Cada miembro del equipo se auto asigna a las tareas que puede realizar.

**Sprint** [10]**:** En Scrum un proyecto se ejecuta en bloques temporales cortos y fijos (iteraciones de un mes natural y hasta de dos semanas). Cada iteración tiene que proporcionar un resultado completo, un incremento de producto que sea susceptible de ser entregado con el mínimo esfuerzo cuando el cliente (Product Owner) lo solicite.

* Cada día el equipo realiza una reunión de sincronización, donde cada miembro inspecciona el trabajo de los otros para poder hacer las adaptaciones necesarias, comunica cuales son los impedimentos con que se encuentra, actualiza el estado de la lista de tareas de la iteración (Sprint Backlog) y los gráficos de trabajo pendiente (Burndown charts).
* El Facilitador (Scrum Master) se encarga de que el equipo pueda cumplir con su compromiso y de que no se merme su productividad.

**Scrum daily meeting** [11]**:** El objetivo de esta reunión es facilitar la transferencia de información y la colaboración entre los miembros del equipo para aumentar su productividad, al poner de manifiesto puntos en que se pueden ayudar unos a otros. Cada miembro del equipo inspecciona el trabajo que el resto está realizando (dependencias entre tareas, progreso hacia el objetivo de la iteración, obstáculos que pueden impedir este objetivo) para al finalizar la reunión poder hacer las adaptaciones necesarias que permitan cumplir con el compromiso conjunto que el equipo adquirió para la iteración (en la reunión de planificación de la iteración).

Cada miembro del equipo debe responder las siguientes preguntas en un timebox de cómo máximo 15 minutos:

* ¿Qué he hecho desde la última reunión de sincronización? ¿Pude hacer todo lo que tenía planeado? ¿Cuál fue el problema?
* ¿Qué voy a hacer a partir de este momento?
* ¿Qué impedimentos tengo o voy a tener para cumplir mis compromisos en esta iteración y en el proyecto?

**Sprint Review:** Reunión informal donde el equipo presenta al cliente los requisitos completados en la iteración, en forma de incremento de producto preparado para ser entregado con el mínimo esfuerzo, haciendo un recorrido por ellos lo más real y cercano posible al objetivo que se pretende cubrir. En función de los resultados mostrados y de los cambios que haya habido en el contexto del proyecto, el cliente realiza las adaptaciones necesarias de manera objetiva, ya desde la primera iteración, replanificando el proyecto. Se realiza en un timebox de como máximo 4 horas.

### 3.1.6 Mapa conceptual

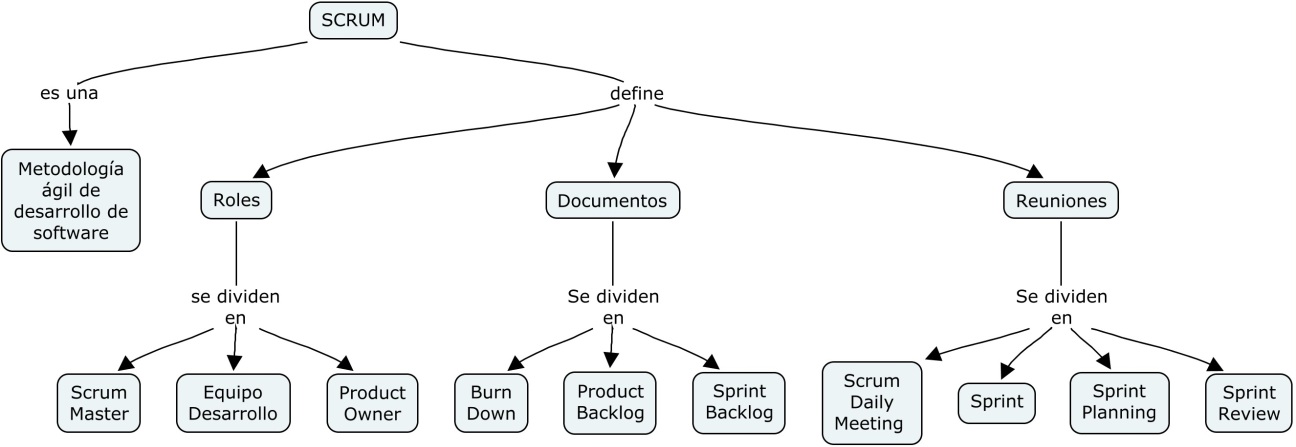


Figura 3. SCRUM

## 3.2 AUP

### 3.2.1 Definición

El Proceso Unificado Ágil (AUP), es una versión simplificada del Proceso Unificado de Desarrollo de Software (RUP). Este describe de una forma simple y fácil de entender, una aproximación al desarrollo de aplicaciones de software utilizando técnicas ágiles y conceptos de RUP. La aproximación aplica técnicas ágiles incluido el Desarrollo Guiado por Pruebas (TDD), el Desarrollo Ágil Dirigido por Modelos (AMDD), la Administración Ágil del Cambio y la Refactorización de Bases de datos para mejorar la productividad [1].

### 3.2.2 Ciclo de vida

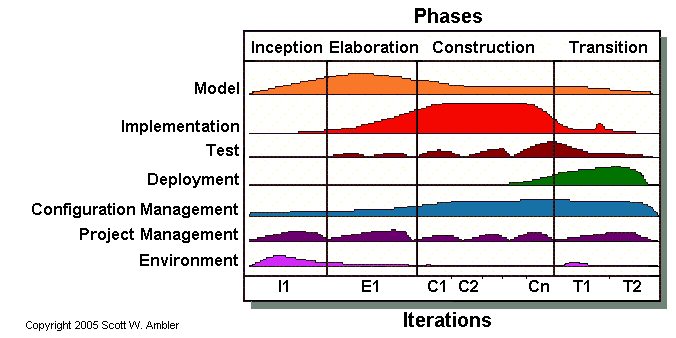


Figura 4. Ciclo de vida AUP

La figura 3, presenta el Ciclo de Vida del proceso Unificado Ágil. Respecto a RUP, se puede notar que las disciplinas han cambiado. En primer lugar, la disciplina de Modelado abarca las disciplinas de Modelado de Negocios, Requisitos y Análisis y Diseño de RUP. Esto es debido a que sí bien el modelado es importante, esta disciplina no domina el proceso, y lo que se pretende es hacerla lo más ágil posible creando los modelos y documentos necesarios para el resto del proceso. En segundo lugar, la disciplina de Gestión de la Configuración y Cambio es ahora la disciplina de Gestión de la Configuración. En el desarrollo ágil, las actividades de gestión del cambio son típicamente parte de los esfuerzos de la gestión de requisitos, lo cual hace parte de la disciplina Modelado.

La metodología AUP hace énfasis en sólo desarrollar los artefactos o hitos que son útiles para el proyecto, no desarrollarlos todos.

### 3.2.3 Fases

Las fases se realizan en forma secuencial, tal como en RUP, a continuación se describe cada una de estas:

**Concepción**: El Objetivo es identificar el alcance inicial del proyecto, una potencial arquitectura para el sistema, obtener financiamiento para el proyecto y aceptación de los involucrados stakeholders). Las principales actividades, incluyen:

* Definir alcance del proyecto
* Estimar costos y plazos
* Definir riesgos
* Determinar factibilidad del proyecto
* Preparar el ambiente

**Elaboración**: El objetivo es obtener la arquitectura del sistema. Las principales actividades, incluyen:

* Identificar arquitectura
* Validar arquitectura
* Desarrollar el ambiente del proyecto
* Definir el equipo del proyecto

**Construcción**: El objetivo es implementar un software sobre una base incremental la que debe estar relacionada con los objetivos de los involucrados. Las principales actividades, incluyen:

* Modelado, construcción y testeo del sistema
* Creación de documentación de apoyo

**Transición**: El objetivo es validar y entregar el sistema en un ambiente de producción. Las principales actividades, incluyen:

* Test del sistema
* Test de usuarios
* Retrabajo del sistema
* Instalación del sistema

### 3.2.4 Disciplinas

Se desarrollan de modo iterativo, definiendo las actividades que el equipo debe desarrollar para satisfacer las necesidades de los stackholders.

**Modelado**: El objetivo de esta disciplina es entender el negocio de la organización, el dominio del problema que se aborda en el proyecto, e identificar una solución viable para hacer frente al dominio del problema. Los principales elementos que se pueden desarrollar en esta disciplina son:

* Modelo de requerimientos: Modelos de casos de uso, modelo de interface de usuario, modelo del proceso de negocio, requerimientos técnicos, pruebas de aceptación, modelo organizacional, modelo de dominio, oportunidades de automatización, reglas de negocio específicas, glosario del proyecto.
* Modelo de diseño: Modelo de objetos, modelo de hilo de seguridad, diseño físico de datos, modelo de despliegue, modelo de interface de usuario, documento de información general.

**Implementación**: El objetivo de esta disciplina es transformar el modelo en código ejecutable y realizar un nivel básico de las pruebas (pruebas de unidad). Los principales elementos que se pueden desarrollar en esta disciplina son:

* Pruebas de regresión
* Código fuente
* Esquema del conjunto de datos
* Modelo del sistema

**Pruebas**: El objetivo de esta disciplina consiste en realizar una evaluación objetiva para asegurar la calidad. Esto incluye encontrar defectos, validar que el sistema funcione como está diseñado, y verificar que se cumplen los requisitos. Los principales elementos que se pueden desarrollar en esta disciplina son:

* Plan del proyecto
* Pruebas de regresión
* Reporte de defectos
* Expediente de revisión

**Despliegue**: El objetivo de esta disciplina es planificar el desarrollo del sistema y ejecutarlo para que esté a disposición de los usuarios finales. Los principales elementos que se pueden desarrollar en esta disciplina son:

* Plan del proyecto
* Notas de la versión
* Scripts de instalación
* Documentación de soporte
* Documentación de usuario
* Material de capacitación
* Documentación de operación.

**Gestión de la configuración**: El objetivo de esta disciplina es administrar el acceso a los productos de trabajo del proyecto. Esto incluye no sólo el seguimiento de versiones de productos de trabajo a través del tiempo, sino también el control y la gestión de los cambios a los mismos. Los principales elementos que se pueden desarrollar en esta disciplina son:

* Ambiente de configuración.
* Artefactos.

**Gestión de proyectos**: El objetivo de esta disciplina es dirigir las actividades que lleva a cabo en el proyecto. Esto incluye la gestión de riesgos, las personas que dirigen (la asignación de tareas, seguimiento de los progresos, etc.), y la coordinación con las personas y los sistemas fuera del alcance del proyecto para asegurarse de que se entregue a tiempo y dentro del presupuesto. Los principales elementos que se pueden desarrollar en esta disciplina son:

* Guía de recursos humanos
* Estimación individual
* Hardware
* Software
* Patrocinadores
* Plan de iteración
* Lista de riesgos
* Cronograma del proyecto
* Estimación
* Presupuesto de plan de proyecto.

**Ambiente**: El objetivo de esta disciplina es apoyar el resto de los esfuerzos para asegurar que el proceso tenga una orientación adecuada (normas y directrices), y las herramientas (hardware, software, etc.), de modo que estén disponibles para el equipo según sea necesario. Los principales elementos que se pueden desarrollar en esta disciplina son:

* Dirección
* Plantillas
* Herramientas

### 3.2.5 Mapa conceptual

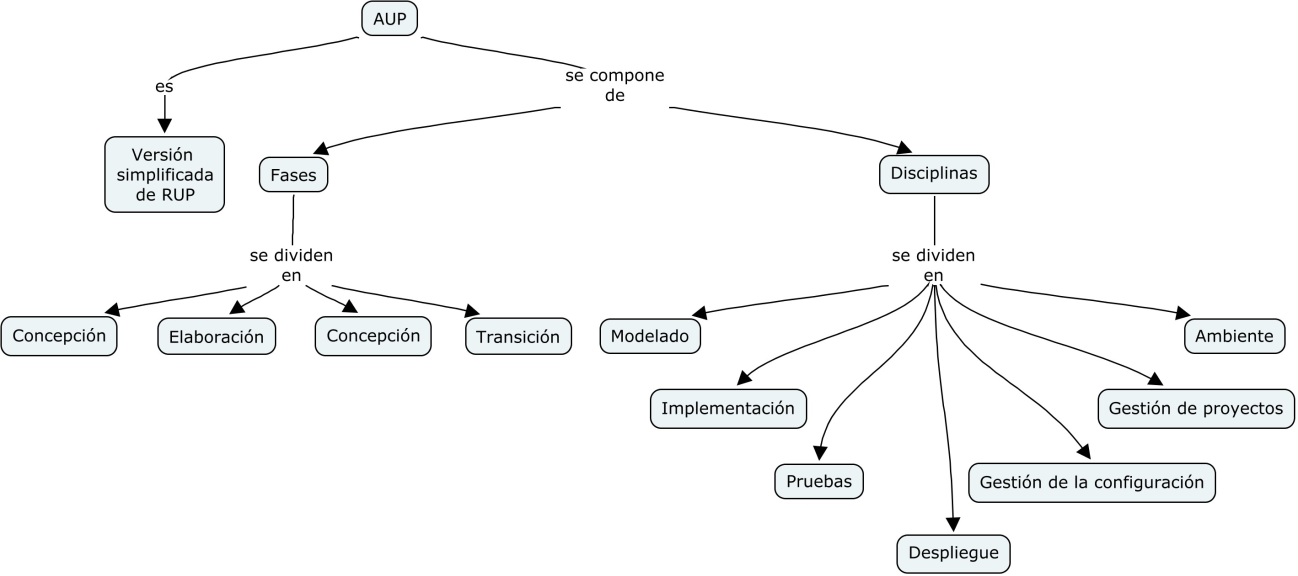
****

Figura 5. AUP

## 3.3 Ventajas y desventajas de AUP/SCRUM en el proyecto integrador

### 3.3.1 BVC Móvil

#### 3.3.1.1 SCRUM

|  |  |
| --- | --- |
| **VENTAJAS** | **DESVENTAJAS** |
| Debido a la naturaleza experimental, adaptativa y cambiante de nuestro proyecto, que involucra las tecnologías móviles y el mercado bursátil, SCRUM se adapta muy bien como modelo metodológico para responder a esta naturaleza.  Al no tener conocimientos exactos de los requisitos de nuestro sistema, ya que este se encuentra en etapa de planeación y no tenemos claro que alcances y limitaciones daremos al mismo, SCRUM nos da la posibilidad de realizar los cambios a los requisitos en el momento que sea necesario.  Al realizar revisiones quincenales o mensuales podemos ver resultados anticipados por lo que podremos hacer cambios en cualquier momento sin necesidad de re-hacer todo el proyecto.  Podemos crear una lista de tareas con ideas principales del core del proyecto y así ir creando sus clases y métodos, sin necesidad de perder tiempo ya que todo puede ser reutilizable.  Al observar los avances, nuestro grupo de trabajo estará más motivado por lo que es factible lanzar un producto de mayor calidad y aceptación.  No es nuestro caso ya que no hay inversión de capital inicialmente, pero la maximización del ROI se vería reflejada en menor tiempo.  En esta metodología no existe un producto final, por lo que es factible seguir mejorando la versión de nuestro producto, satisfaciendo las necesidades de los posibles usuarios, siendo más aceptado en el mercado. | No es nuestro caso, pero en grupos de trabajo muy grandes, el proyecto se vuelve incontrolable.  No tener suficiente conocimiento de la metodología, por lo que el Scrum master no se puede definir; los integrantes de nuestro equipo seremos product owner, scrum master y equipo de desarrollo.  La documentación en entidades del sector financiero es obligatoria, por lo que estaríamos consumiendo tiempo y recursos en algo no exigido por la metodología.  Falta de especializados en cada área. Al ser desarrolladores y dba’s, no tenemos un claro conocimiento de procesos como pruebas, documentación, etc. |

#### 3.3.1.2 AUP

|  |  |
| --- | --- |
| **VENTAJAS** | **DESVENTAJAS** |
| Podemos descomponer nuestro proyecto en proyectos más pequeños que se pueden hacer de modo iterativo:   * Servicio Web de la BVC * Sistema para los dispositivos móviles * Sistema de sincronización   Podemos hacer una detección temprana de los riesgos como:   * Garantizar la consulta de acciones en tiempo real * Inconvenientes para establecer una interface con la BVC   Podemos centrarnos en la arquitectura de nuestro sistema, diseñando todos los planos antes de la construcción.  Podemos definir nuestro sistema desde diferentes puntos de vista.  Durante la etapa de desarrollo, sabremos lo que tendremos que hacer, basados en los planos que diseñamos. | Es un modelo más pesado ya que se basa en RUP, esto puede generar inconvenientes en nuestro proyecto, ya que las tecnologías móviles tienen una naturaleza experimental, adaptativa y cambiante que requieren de una respuesta muy ágil, más aún, los planos del sistema podrían cambiar en tiempo de desarrollo.  No se ajusta muy bien a nuestro proyecto, ya que nuestro equipo de trabajo está conformado por dos integrantes, que intervendremos en todos los roles del proceso de desarrollo.  Podríamos invertir tiempo en la creación de modelos que luego podrían cambiar, debido a la naturaleza experimental que requiere nuestro proyecto. |

### 3.3.2 Sistema de gestión porcina

#### 3.3.2.1 SCRUM

|  |  |
| --- | --- |
| **VENTAJAS** | **DESVENTAJAS** |
| Al ser iterativo se puede realizar investigación y levantamiento de requisitos a medida que el proyecto avanza.  Los módulos se pueden desarrollar durante los sprint. De esta manera se pueden validar una vez termine cada uno.  Las iteraciones ayudan a tener una retroalimentación del cliente más recurrente, lo que se vuelve un tema a favor de la calidad/funcionalidad del proyecto.  A medida que se hacen las entregas, se pueden ir probando. Esto es ventajoso ya que las funcionalidades se pueden ir probando y recibir retroalimentación tanto de errores como de mejoras.  La motivación que de el(los) desarrollador(es) va en beneficio del proyecto.  El(los) desarrollador(es) tienen libertad para poner su creatividad en practica  La libertad que tiene el equipo es muy positiva ya que este se puede autogestionar. Maneja su tiempo a su manera siempre y cuando cumpla con los entregables.  Como los sprint son siempre del mismo tiempo, los desarrolladores se van a ir acostumbrando a trabajar en periodos determinados y cuando se tenga que estimar algo, estas estimaciones van a ser mucho más precisas. Además, las métricas de c/u se vuelven mas precisas a medida que pasa el tiempo. | Si los requisitos varían mucho, es probable que el producto final varíe mucho de lo que inicialmente se planteó y se pueda retrasar su finalización  NO es una desventaja propiamente, pero es necesario planear bien los sprint para que se tenga éxito. |

#### 3.3.2.2 AUP

|  |  |
| --- | --- |
| **VENTAJAS** | **DESVENTAJAS** |
| Los requisitos están definidos desde el principio, lo que demarca el alcance del proyecto y permitirá modelar la arquitectura desde el principio.  Siendo una versión light de RUP, es más sencillo adoptarlo para las personas acostumbradas a RUP. | Un cambio en una etapa tardía es muy costoso para el proyecto  Al tener tantas fases se necesita de un equipo muy grande para que estas se puedan realizar a cabalidad.  La cantidad de diagramas que hay que realizar es mucha para un proyecto que es propio  Si un requisito inicial cambia después de la mitad del proyecto es muy probable sea muy costoso afrontar el reproceso. |

# 4. Conclusiones

La innovación en las empresas va de la mano de la gestión del conocimiento, ya que al realizar un manejo adecuado del conocimiento (transferencia), este se va a diseminar mucho más fácil entre los empleados de la compañía, lo que podría facilitar la innovación.

La innovación es una de las herramientas más efectivas para alcanzar nuevos mercados, pero a la vez presenta más riesgos ya que al involucrarse en un tema desconocido no se sabe la aceptación que se pueda a llegar a tener en el entorno.

Para realizar innovación, las organizaciones deben tener como política una gestión de la innovación como función, además de invertir en desarrollar capacidades tecnológicas y usar adecuadamente los recursos.

La optimización no necesariamente es una innovación y se define como un intento de cambiar la forma en que la gente ejecuta las actividades para satisfacer mejor los objetivos de una empresa, esta se logra basada en la experiencia de la organización mediante conocimiento formal y tácito.

La optimización se hace necesaria debido a la necesidad de las empresas por ser más competitivas en un mercado globalizado. En función del desarrollo de software, la optimización de procesos, debe permitir a una compañía estar reorientando el proceso de desarrollo según cambie el objetivo del producto el cual es impuesto por las condiciones del mercado.

Las metodologías ágiles como SCRUM y AUP, ofrecen un modelo adecuado para el desarrollo de productos de software, basados en la simplicidad y la evolución constante. Surgen como alternativa para aquellos proyectos en los que no se debe seguir un diseño estricto y una documentación excesiva, y que deben estar preparados para cambios durante el proyecto.

En la metodología SCRUM, no se tiene la idea de producto final, sino la de un producto que va evolucionando en función del tiempo a través de las entregas parciales del proyecto (Los Sprint). Las funciones pueden ser validadas por el cliente en cada entrega, obteniendo la retroalimentación necesaria, para así tomar las decisiones correctas en un tiempo oportuno. Esta metodología promueve la creatividad y la libertada en el equipo de desarrollo.

La metodología AUP, es un modelo basado en el Proceso Unificado de Desarrollo de Software (RUP), que busca mayor simplicidad y facilidad de entendimiento, y sirve como aproximación en el desarrollo de software utilizando técnicas ágiles. A la vez que promueve la agilidad, no descuida el diseño de software basado en la arquitectura y los modelos, permitiendo un control anticipado de los riesgos.

# 5. Referencias

**Optimización de procesos:**

1. Metso, Inc. "Process optimization", <http://www.multisimplex.com/optimization.htm>.
2. Wikipedia, "Process Optimization", <http://en.wikipedia.org/wiki/Process_optimization>.
3. Lorenz T. Biegler, "Nonlinear Programming: Concepts, Algorithms, and Applications to Chemical Processes", Chapter 1 "Introduction to Process Optimization", SIAM Books 2010.
4. Masao Ito, "The Elements of Software Process Optimization: Dealing with the Process Dynamics", Nil Software Corp.
5. Gonzalo Alonso Sánchez Lorenzo, "Mejora de proceso software de una pequeña empresa desarrolladora de software: Caso CompetiSoft - Perú TAU", Pontificia Universidad Católica del Perú.

**Innovación en Software:**

1. Wikipedia, "Innovación", <http://es.wikipedia.org/wiki/Innovaci%C3%B3n>.
2. Web and macros, "Definición y concepto de Innovación", <http://www.webandmacros.com/innovacionconceptos.htm>.
3. Premios innovación, "Michael Porter", <http://premiosmuyinnovacion.muyinteresante.es/tag/michael-porter>.
4. Marco Álvarez, "La Innovación en la Industria de Software", Artículos relacionados a la innovación, 17/06/2011, <http://blog.pucp.edu.pe/item/135789/la-innovacion-en-la-industria-de-software>.
5. Universidad Nacional de Trujillo, "Gestión del Conocimiento en el Proceso de Desarrollo de Software”, <http://es.scribd.com/doc/18994532/Gestion-del-Conocimiento-en-el-Desarrollo-de-Software>.

**Empresas innovadoras:**

1. Sam Gustin, "Why Apple is Winning: Innovation, Opportunity and Execution", Time Business, 04/25/2012, <http://business.time.com/2012/04/25/why-apple-wins-innovation-opportunity-and-execution/>.
2. Quentin Hardy, "Google's Innovation -And Everyone's?", Forbes, 07/16/2011, <http://www.forbes.com/sites/quentinhardy/2011/07/16/googles-innovation-and-everyones/>.
3. Annika Steiber, "Why is Google one of the most innovative companies?", Hindustan Times, 25/05/2012, <http://www.hindustantimes.com/technology/Chunk-HT-UI-Technology-OtherStories/Why-is-Google-one-of-the-most-innovative-companies/SP-Article1-845935.aspx>.
4. Wikipedia, "Dropbox (service)", <http://en.wikipedia.org/wiki/Dropbox_(service)>.
5. Fast Company, "The World's 50 Most Innovative Companies", 2012, <http://www.fastcompany.com/most-innovative-companies/2012/full-list>.
6. "Gestión de Conocimiento en Liberty Seguros", <http://www.comunikandonos.com/sitio/comunicacion-interna/25-gesti-de-conocimiento-en-liberty-seguros.html>.
7. "Herramientas de tecnología de información (TI) en Gestión de Conocimiento", <http://www.ecopetrol.com.co/especiales/ecopetroweb/esp/herramientas_esp.html>.

**SCRUM:**

1. “Qué es Scrum”, <http://www.proyectosagiles.org/que-es-scrum>.
2. “Fundamentos de Scrum", <http://www.proyectosagiles.org/fundamentos-de-scrum>.
3. “Rol y perfil del Product Owner”, <http://www.aplicandoscrum.com/rol-perfil-product-owner/>.
4. “Scrum Master”, <http://www.dosideas.com/wiki/Scrum_Master>.
5. “Lista de objetivos / requisitos priorizada (Product Backlog)”, <http://www.proyectosagiles.org/lista-requisitos-priorizada-product-backlog>.
6. “Sprint Backlog”, <http://www.mountaingoatsoftware.com/scrum/sprint-backlog>.
7. “Lista de tareas de la iteración (Sprint Backlog)”, <http://www.proyectosagiles.org/lista-tareas-iteracion-sprint-backlog>.
8. “Gráficos de trabajo pendiente (Burndown charts),“ <http://www.proyectosagiles.org/graficos-trabajo-pendiente-burndown-charts>.
9. “Planificación de la iteración (Sprint Planning)”, <http://www.proyectosagiles.org/planificacion-iteracion-sprint-planning>.
10. “Ejecución de la iteración (Sprint)”, <http://www.proyectosagiles.org/ejecucion-iteracion-sprint>.
11. “Reunión diaria de sincronización del equipo”, <http://www.proyectosagiles.org/reunion-diaria-de-sincronizacion-scrum-daily-meeting>.
12. “Reunión diaria de sincronización del equipo (Scrum daily meeting)”, <http://www.proyectosagiles.org/reunion-diaria-de-sincronizacion-scrum-daily-meeting>.
13. “Demostración de requisitos completados (Sprint Review)”, <http://www.proyectosagiles.org/demostracion-requisitos-sprint-review>.
14. “Beneficios de SCRUM”, Proyectos Ágiles.ORG, <http://www.proyectosagiles.org/beneficios-de-scrum>.
15. “Defectos de SCRUM”, Calidad y Software, <http://calidadysoftware.blogspot.com/2011/11/defectos-de-scrum-i.html>

**AUP:**

1. Scott W. Ambler, “The Agile Unified Process (AUP)”, Ambysoft 2009, <http://www.ambysoft.com/unifiedprocess/agileUP.html>.
2. Luis Fernando Londoño, “Metodología AUP”, Especialización en Desarrollo de Software EAFIT 2012.
3. Jorge Luis Cordero, Metodología Ágiles "Proceso Unificado Ágil (AUP)". <http://es.scribd.com/doc/97756785/METODOLOGIAS-AGILES-AUP>.