**MARCO TEORICO**

Para facilidad de entendimiento de este problema tenemos que entender términos y conceptos importantes que son:

* sistema de información.
* Ingeniería de software.
* Proceso de desarrollo de software.
* Scrum
* Procesamiento de imágenes.

**Sistema de Información.**

**A**ntes de entrar a mencionar un poco sobre que es un sistema de información, entremos investigar que es sistema y que es información. Según (Murdick,1988) un sistema es “*es un conjunto de elementos organizados que se encuentran en interacción, que busca alguna meta o metas comunes”*, por otra parte el estándar **IEE 1471-2000** define sistema como “*colección de componentes organizados para cumplir determinada función”*. Por otra parte (**Chiavenato,2006**), información “*es un conjunto de datos con un significado, o sea, que reduce la incertidumbre o que aumenta el conocimiento en algún tema. En verdad, la información es un mensaje o significado en un determinado contexto, disponible para uso inmediato y que proporciona orientación a las acciones por el echo de reducir el margen de inceditumbre con respecto a nuestra decisiones”*.

Con estas definiciones podemos obtener que un sistema de información es una parte vital de un proceso no solo para almacenar información. Si no también para reducir el riesgo al momento de tomar daciones o realizar una gestión interna.

**Ingeniería de** s**oftware.**

Debido al crecimiento y desarrollo tecnológico, los sistemas de información requieren ser cada vez más robustos y confiables, rápidos y agiles. Esto lleva al software a superar una serie de desafíos: ser capaz de cumplir en su totalidad los requerimientos o necesidades del cliente solicitando, ser capaz de ajustarse a la creciente demanda y una obsolescencia rápida, ser capaz de reducir sus mantenimientos al mínimo, sin importar la complejidad de la creación debe cumplir con las fechas estipuladas, tener una calidad lo suficiente mente confiable. Que el software de verdad consuma toda el potencial de hardware en el que está montado.

Debido a la complejidad de los mercados u organizaciones en la actualidad es indispensable que el software sea capaz de transformarse y adaptare, es por esto que debe realizarse una ingeniera para que este sea capaz de apartarse de forma rápida sin perder la funcionalidad ya existente.

La (IEEE,1990) define la ingeniería con una aplicación de un método sistemático, estructurado y cuantificable aplicado a sistemas o procesos, esto nos brinda una definición mas global de lo intangible del software. Dado esto podemos definir que la ingeniería de software busca la integración de diferentes principios ya sea de la ciencia aplicada a la computación o matemáticas con bases ingenieriles, las cuales son utilizadas para dar soluciones optimas a las soluciones planteadas por los requerimientos. Para dar una definición mas superior sobre la ingeniería de software (Pressman,2016) nos dice “*la ingeniería de software es una disciplina que integra el proceso, los métodos, y las herramientas para el desarrollo de software computacional*”. Pressman. Nombra la ingeniería del conocimiento como “*tecnología multicapa,*” donde cada capa se encuentra sostenida por un capa de certificación (enfoque de calidad), esto lleva a que en cada capa se brinda un estándar de calidad con el fin de generar productos mas robustos y adicionales en los procesos. Adicional nos nombra dos capas extras: procesos y métodos, donde se brindan los procedimientos que se aplican he identifican un software.

* **Proceso de desarrollo de software.**

Para la creación de un software implica un conjunto unificado de conocimiento, recursos y tiempo, por lo que se debe generar un adecuado proceso de actividades que den garantías de la culminación del proyecto. La ingeneira de software brinda la herramienta aparte de muchas, para planear y establecer esa trayectoria de actividades y obtener como resultado el ciclo de vida del Software.

* + **Proceso unificado de desarrollo de software.**

Para realizar un proceso detallado de construcción de software debe realizarce una metodología. Según (larman,2003), una metodología define: artefactos, roles y actividades, junto con las practicas y las técnicas recomendadas, con frecuencia es utilizado el termino “metodo” para referirse a técnicas, notacionales y guias asociadas, que son aplicables a una/s actividades dl proceso de desarrollo.

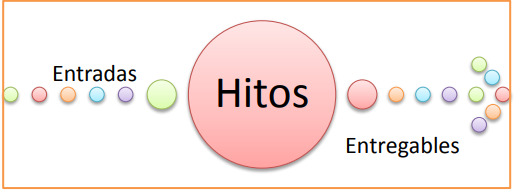
En este proceso unificado brinda una serie de escalones orientados a objetos, que se baza en: tener una dirección brindada por los casos de uso, por lo que debe crear modelos para el diseño y la implementación, esto con un entorno centralizado en la arquitectura planteada. En esta arquitectura se definen cuáles serán los procesos dinámicos o estáticos utilizados por el sistema. Esto debe de ser planeado de forma interactiva e incremental, siempre con el foco de crecimiento del software, para esto se realizan fases y entregas que pueden repetir o tener varios ciclos de vida.

**Fases y ciclo de vida un proyecto**

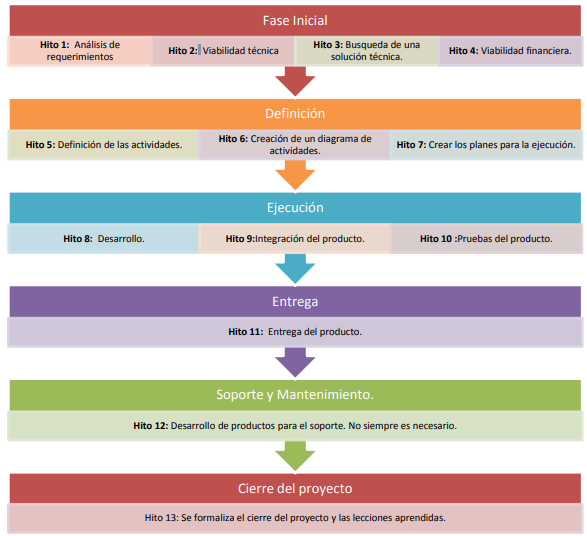
Como lo hablamos anterior mente en el ciclo de vida se ve identificado las actividades a realizar ya sea solo uno o sea incremental, estas actividades marcaran el inicio y el fin de las actividades para el software. Al inicial un proyecto de software una empresa siempre será solicitado un estudio de viabilidad.

Los ciclos de vida de un proyecto siempre se basan en 3 items:

* ¿Qué trabajo se va a realizar en cada fase?
* ¿Quién participara en cada fase?
* Se definirán las fases de forma secuencias y las que sean en paralelo.



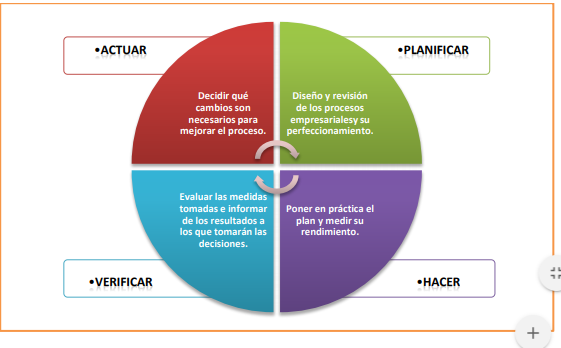
La mayoría de fases escogidas para el desarrollo de proyectos de software son:



EN cada una de las fases, se ve acompañada una fase adicional (Como lo mencionamos anterior mente), fase de certificación.

**Modelo tradicional de planificación PDCA**

El modelo PDCA fue popularizado por W .Edwards Deming y suele referise como el modelo de “espiral para el mejoramiento continuo.”



Planificar

* Reconocer el problema y establecer prioridad.
* Formar el equipo de resolución de problemas.
* Definir el problema y su ámbito de aplicación claramente.
* Analizar el problema/proceso.
* Determinar las posibles causas.
* Identidicar las posibles soluciones.
* Evitar la tentación de proponer soluciones rapidas e inmediatas.
* Los objetivos deben ser específicos, medibles, alcanzables y realiztas.
* Evaluar posibles soluciones.

Hacer

* Implementar la solución de cambio o proceso.
* Seguimiento de los resultados y recopilar datos.

Verificar

* Revisar y evaluar el resultado del cambio.
* Medir el progreso en contra de los hitos.
* Verificar que no haya consecuencias imprevistas

**SCRUM**

En este desarrollo utilizaremos esta metodología por lo que entraremos a identificar y conocer sobre esta metodología ágil.

**INTRODUCCION**

Scrum aparece como una practica destinada a los productos tecnológicos y será en 1993 cuando realmente Jeff sutherlan aplique un modelo de desarrollo de Software en Ease/Corporation.

En 1996, Jeff Sutherlan y ken Schaber presentaron las practicas que se usaban como proceso formal para el desarrollo de software y que pasarían a inclinarse en la lista de agile Alliance.

Scrum es adecuado para aquellas empresas en las que el desarrollo de los productos se realiza en entornos que se caracterizan por tener:

1. **Incertidumbre:** Sobre esta variable se plantea el objetivo que se quiere alcanzar sin proporcionar un plan detallado del producto.
2. **Auto-organización:** Los equipos son capaces de organizarse por si solos, no necesitan roles apra la gestión pero tienen que reunir las siguientes características:
   1. **Autonomía:** Son los encargados de encontrar la solución usando la estrategia que encuentren adecuada.
   2. **Auto separación**: La soluciones iniciales sufrirán mejoras.
3. **Control moderado:** se establecerá un control suficiente para evitar descontroles. Se basan en crear un “AUTO CONTROL ENTRE IGUALES”para no impedir la creatividad y espontaneidad de los miembros del equipo.

**Fases para definir el ciclo de vida de desarrollo ágil.**

1. **Concepto:** Se define de forma general las características del producto y se asigna el equipo que se encargara de su desarrollo.
2. **Especulación:** en esta fase se hacen disposiciones con la información obtenida y se establecen los limites que marcaran el desarrollo del producto, tales como costes y agendas.
3. **Exploración:** Se incrementa el producto en el que se añaden las funcionalidades de la fase de especulación.
4. **Revisión:** el equipo revisa todo lo que se ha construido y con el objetivo deseado
5. **Cierre:** se entregara en la fecha acordad una versión del producto deseado. Al tratarse de una versión, el cierre no indica que se ha finalizado el proyecto, sino que seguirá habiendo cambios, denominados “mantenimiento”, que ara el producto final se acerque al producto final. 

Scrum gestiona estas iteraciones a través de reuniones diarias, uno de los elementos fundamentales de esta metodología.



**Componentes Scrum**

Se pueden dividir en 2 fases:

1. **Las reuniones:**
   1. **Planificación del backlog:**

Se definirá un documento en el que estarán los requisitos del sistema por prioridades.

En esta fase se definirá la planificación del SPRING 0, este definirá cuales van a ser los objetivos y el trabajo que hay que realizar.

* 1. **Seguimiento del Sprint**

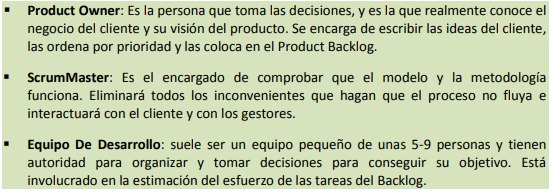
En esta fase se hacen reuniones diarias con 3 preguntas especificas:

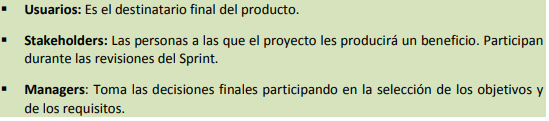
* + - * ¿Qué trabajo se realizo desde la reunión anterior?.
      * ¿Qué trabajo sehara hasta la siguiente reinon?.
      * Problemas que han sugrido y que soluciones hay.
  1. **Revisión del Sprint**

Cuando se finaliza el sprint se realizara una revisión del incremento que se ha generado. Se generara un resultado final y una nueva versión, esto ayuda a mejorar la conversación el cliente y que el pueda ver el crecimiento del requerimiento.

**2. Los Roles**

Los roles se dividen en 2 grupos:

1. Prime grupo: 
2. Segundo grupo:



**Elementos de Scrum**

Los elementos del scrum son:

1. **Product backlog:** lista de necesidades del cliente
2. **Sprint backlog:** lista de tareas que se realizan en un sprint.
3. **Incremento:** Parte añadida o desarrollada en un sprint, es una parte terminada y totalmente operativa.

**Ciclo del desarrollo scrum**

