**CONTROL DE CAMBIOS**

|  |  |
| --- | --- |
| Versión | 1.0 |
| Proyecto | MEPPP |
| Fecha | 28/05/2018 |
| Elaborado por | Jonathan Jair Alfaro Sánchez |
| Localización del Documento | [url del github] |

|  |  |  |  |
| --- | --- | --- | --- |
| **Historial de Cambios** | | | |
| **Versión** | **Fecha** | **Responsable** | **Descripción del Cambio** |
| 1.0 | 28/05/2018 | JJAS | Creación del documento. |
|  |  |  |  |
|  |  |  |  |

**Contenido**

[CICLO DE VIDA 3](#_Toc515656560)

[DEFINICIÓN 3](#_Toc515656561)

[MODELOS DE CICLO DE VIDA 3](#_Toc515656562)

[Ciclo de vida en cascada 3](#_Toc515656563)

[Ciclo de vida en “V” 5](#_Toc515656564)

[Ciclo de vida Scrum 7](#_Toc515656565)

[JUSTIFICACIÓN DEL CICLO DE VIDA ELEGIDO 12](#_Toc515656566)

[TABLA DE ACTIVIDADES 13](#_Toc515656567)

# CICLO DE VIDA

## DEFINICIÓN

El ciclo de vida del desarrollo Software (SDLC en sus siglas inglesas), es una secuencia estructurada y bien definida de las etapas en Ingeniería de software para desarrollar el producto software deseado.

Roger S. Pressman los define como Modelos prescriptivos de proceso y señala que un modelo prescriptivo de proceso es “un conjunto distinto de actividades, acciones, tareas, fundamentos y productos de trabajo que se requieren para desarrollar software de alta calidad. Estos modelos de proceso no son perfectos, pero proporcionan una guía útil para el trabajo de la ingeniería del software” (S. Pressman, 2005).

## MODELOS DE CICLO DE VIDA

### Ciclo de vida en cascada

* **Definición:** El modelo en cascada, algunas veces llamado el ciclo de vida clásico, sugiere un enfoque sistemático, secuencial hacia el desarrollo de software, que inicia con la especificación de requerimientos del cliente y que continua con la planeación, el modelado, la construcción y el despliegue para culminar en el soporte del software terminado (S. Pressman, 2005).
* **Características:** Ordenar de manera rigurosa las etapas del ciclo de vida de software, dado que el comienzo de cada etapa debe esperar a la finalización de la inmediata anterior. Cuando la revisión determina que el proyecto no está listo para pasar a la siguiente etapa, permanece en la etapa actual hasta que esté preparado. Y debido a que el proceso está planeado es más fácil determinar costos y los plazos. Esté modelo puede ser visto como un modelo con forma de cascada de agua con varios saltos, en la que cada salto representa cada una de las fases del ciclo de vida [Ver Figura 2 Ciclo de vida en cascada].

La metodología en cascada es esencialmente:

* El inicio y el alcance del proyecto.
* La planificación del proyecto (calendario, recursos necesarios, costo).
* Definición de las necesidades del negocio y el análisis en detalle dela solución.
* La creación de la solución.
* Prueba que la solución funciona.
* La entrega de la solución a su público objetivo.
* Cierre del proyecto.

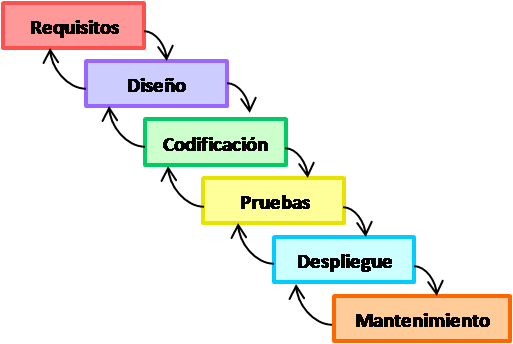


Figura 1 Ciclo de vida en cascada

* **Ventajas:**
* Permite la departamentalización y control de gestión.
* El horario se establece con los plazos normalmente adecuados para cada etapa de desarrollo.
* Este proceso conduce a entregar el proyecto a tiempo.
* Es sencilla y facilita la gestión de proyectos.
* Permite tener bajo control el proyecto.
* Limita la cantidad de interacción entre equipos que se produce durante el desarrollo.
* **Desventajas:**
* “Es muy raro que los proyectos reales sigan el flujo secuencial que propone el modelo” (S. Pressman, 2005).
* “Con frecuencia es difícil para el cliente establecer todos los requisitos de manera explícita. El modelo en cascada lo requiere y se enfrentan dificultades al incorporar la incertidumbre natural presente en el inicio de muchos proyectos” (S. Pressman, 2005).
* “El cliente debe tener paciencia. Una versión que funcione de los programas estará disponible cuando el proyecto esté muy avanzado” (S. Pressman, 2005).
* Se tarda mucho tiempo en pasar por todo el ciclo.
* La aplicación de la metodología en cascada se orienta mejor al desarrollo de proyectos de corto plazo, de poca innovación y proyectos definitivos y detallados.
* Metodología pueden confundir al equipo profesional en las etapas tempranas del proyecto.

### Ciclo de vida en “V”

* **Definición:** También conocido como modelo de cuatro niveles. El modelo de ciclo de vida V proviene del principio que establece que los procedimientos utilizados para probar si la aplicación cumple las especificaciones ya deben haberse creado en la fase de diseño [Ver Figura 5 Ciclo de vida en "V"].
* **Características:** Busca hacer la actividad de pruebas más efectiva y productiva. Los planes (y casos de prueba) se van elaborando a medida que se avanza en el desarrollo del proyecto.

El modelo en V es una variación del modelo en cascada que muestra cómo se relacionan las actividades de prueba con el análisis y el diseño.

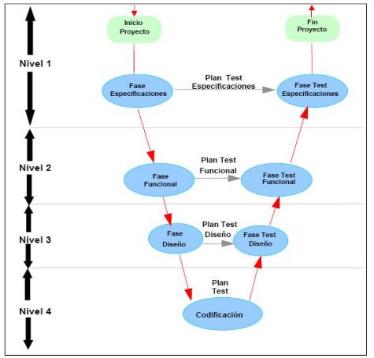


Figura 2 Ciclo de vida en "V"

* **Ventajas:**
* La relación entre las etapas de desarrollo y los distintos tipos de pruebas facilitan la localización de fallos.
* Es un modelo sencillo y de fácil aprendizaje.
* Hace explícito parte de la iteración y trabajo que hay que revisar.
* Especifica bien los roles de los distintos tipos de pruebas a realizar
* Involucra al usuario en las pruebas.
* **Desventajas:**
* Es difícil que el cliente exponga explícitamente todos los requisitos.
* El cliente debe tener paciencia pues obtendrá el producto al final del ciclo de vida.
* Las pruebas pueden ser caras y, a veces, no lo suficientemente efectivas.
* El producto final obtenido puede que no refleje todos los requisitos del usuario.

### Ciclo de vida Scrum

* **Definición:** “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.” (Deemer, Benefield, Larman, & Vodde, 2009)
* **Características:** (Bahit, 2012)
* Un Sprint tiene dos características fundamentales:
* Una duración fija entre 1 a 4 semanas.
* Cada Sprint se ejecuta de forma consecutiva (sin tiempo muerto entre un sprint y otro).
* El objetivo principal es transformar un conjunto de ítems requeridos por el cliente en un incremento de funcionalidad 100% operativa para el software.



Figura 3 Marco de trabajo Scrum

* **Roles de Scrum:** (Bahit, 2012)
* El **Director Scrum (Scrum Master)**, responsable de asegurar los procesos.
* El **Dueño de Producto (Product Owner)**,, responsable de maximizar el valor del producto.
* El **Equipo (Team Members)**, responsable de realizar el trabajo.
* **Ceremonias en Scrum:** (Bahit, 2012)
* **Planificación del Sprint:**
* La planificación es lo primero que debe hacerse al comienzo de cada Sprint.
* Durante esta ceremonia, participan el Dueño de Producto, el Scrum Master y el Scrum Team.
* El objetivo de esta ceremonia, es que el Dueño de Producto pueda presentar al equipo, las historias de usuario prioritarias.
* **Reunión diaria:**
* Las reuniones diarias para Scrum, son "conversaciones" de no más de 5-15 minutos, que el Scrum Master tendrá al comienzo de cada día, con cada miembro del equipo.
* Ponerse al día de lo que cada miembro ha desarrollado (en la jornada previa), lo que hará en la fecha actual, pero por sobre todo, conocer cuáles impedimentos estén surgiendo, a fin de resolverlos y que el Scrum Team pueda continuar sus labores, sin preocupaciones.
* **Revisión:**
* El equipo presentará al Dueño de Producto las funcionalidades desarrolladas. Las explicará y hará una demostración de ellas.
* El Dueño de Producto podrá sugerir mejoras a las funcionalidades desarrolladas, aprobarlas por completo o eventualmente, rechazarlas si considera que no se ha cumplido el objetivo.
* La ceremonia de revisión se lleva a cabo el último día del Sprint.
* **Retrospectiva:**
* “La búsqueda de la perfección”.
* Última ceremonia del Sprint.
* El objetivo de esta retrospectiva, es realizar un análisis de lo que se ha hecho y sus resultados correspondientes, y decidir qué medidas concretas emplear, a fin de mejorar esos resultados.
* **Artefactos Scrum:** (Bahit, 2012)
* **Backlog de Producto:** Una lista priorizada de todo lo que requiere el software.
* **Backlog de Sprint:** Una lista de tareas necesarias para convertir parte del Backlog de Producto, en incremento de funcionalidad del Software.
* **Scrum taskboard:** Un tablero físico que permite la transparencia de los ítems de trabajos pendientes, en curso y terminados de un mismo Sprint.
* **Diagrama de Burndow:** Un gráfico que permite medir visualmente, el progreso de los items del Backlog de Producto.
* **Ventajas:**
* **Gestión de las expectativas de los clientes.** Los clientes pueden participar en cada una de las iteraciones y proponer soluciones. (EAE Business School, 2016)
* **Resultados anticipados.** Cada iteración arroja una serie de resultados. No es necesario, por tanto, que el cliente espere hasta el final para ver el producto. (EAE Business School, 2016)
* **Flexibilidad y adaptación a los contextos.** Se adapta a cualquier contexto, área o sector de la gestión. No es una técnica exclusiva de ninguna disciplina. (EAE Business School, 2016)
* **Gestión sistemática de riesgos.** Del mismo modo, los riesgos que pueden afectar a un proyecto son gestionados en el mismo momento de su aparición. La intervención de los equipos de trabajo es inmediata. (EAE Business School, 2016)
* El cliente puede comenzar a utilizar el producto rápidamente. (Comentarios Informaticos, s.f.)
* El cliente puede decidir los nuevos objetivos a realizar. (Comentarios Informaticos, s.f.)
* Se agila el proceso, porque se divide el problema en pequeñas tareas. (Comentarios Informaticos, s.f.)
* Menos probabilidad de que se den sorpresas o desarrollos inesperados porque el cliente va viendo poco a poco lo que se está desarrollando. (Comentarios Informaticos, s.f.)
* **Desventajas:**
* **Funciona sobre todo con equipos reducidos.** (EAE Business School, 2016)
* **Requiere una exhaustiva definición de las tareas y sus plazos.** Cuando estos dos aspectos no se definen adecuadamente, Scrum se desvanece. Recuerda que la división del trabajo en iteraciones (y de éstas en tareas específicas) son la esencia de esta metodología. (EAE Business School, 2016)
* **Exige una alta cualificación o formación.** No es una modalidad de gestión propia de grupos junior o que apenas estén en proceso de formación. Gran parte del éxito de Scrum radica en la experiencia que aportan los profesionales de los equipos, quienes por lo general acumulan años de experiencia. (EAE Business School, 2016)
* Existe la tendencia que si se deja una tarea sin terminar y que por las exigencias del Dueño del Producto se deban realizar otras nuevas. Estas tareas no terminadas puedan obstaculizar la planeación de nuevas Sprints y se deba volver al problema original. (Comentarios Informaticos, s.f.)
* Alto nivel de estrés de los miembros del equipo, el desgaste puede ser excesivo y estresante lo que puede disminuir el rendimiento. (Comentarios Informaticos, s.f.)
* La necesidad de contar con equipos multidisciplinarios puede ser un problema, porque cada integrante del equipo debe estar en capacidad de resolver cualquier tarea y no siempre se cuenta con este perfil en la empresa. (Comentarios Informaticos, s.f.)
* El equipo puede estar tentado de tomar el camino más corto para cumplir con un sprint, que no necesariamente puede ser el de mejor calidad en el desarrollo del producto. (Comentarios Informaticos, s.f.)

## JUSTIFICACIÓN DEL CICLO DE VIDA ELEGIDO

Para el desarrollo de este proyecto se decidió implementar la metodología Scrum, ya que es una de las metodologías agiles más utilizadas y populares. Esta metodología busca entregar un producto final que de un mayor valor al cliente por lo que pensamos es una buena opción. Además, esta metodología se acopla en algunos aspectos de nuestro proyecto, como por ejemplo el hecho de que cada Sprint debe ser de una duración mínima de una semana, y nosotros durante el proyecto nos estaremos reuniendo cada semana para determinar las actividades a realizar. Otro motivo por el cual lo elegimos es porque es una metodología que solo funciona con equipos de trabajo pequeños.

Las reuniones son un punto clave para esta metodología, cosa que como empresa también consideramos muy importante, y que esperamos que el buen uso de estas reuniones nos lleve al desarrollo de un producto de calidad.

# TABLA DE ACTIVIDADES

|  |  |  |
| --- | --- | --- |
| Nombre de la actividad | | Justificación |
| Inicio | | |
|  | Acta de inicio | Se realizará el documento que dará inicio al proyecto de manera formal para posteriormente comenzar con las demás actividades. |
| Planeación | | |
|  | Presupuesto | Se realizará un análisis del proyecto para obtener el presupuesto disponible para el desarrollo del proyecto, de esta actividad depende uno de los factores de éxito del proyecto. |
|  | Estudio de roles | Se realizará un estudio de roles para obtener el equipo de trabajo adecuado, y así organizarlo de acuerdo a las habilidades y conocimientos que tiene cada uno de ellos y cumplir con un factor de éxito. |
|  | Selección del ciclo de vida | Se hará la selección entre algunos ciclos de vida, con el fin de obtener el que mejor se adapta para el tipo de proyecto solicitado, además, describe la metodología de trabajo con la que el equipo de JOMIMAN trabajará durante la ejecución del proyecto. |
|  | Declaración del alcance | Define hasta donde llegará el proyecto, con él, tenemos conocimiento especifico de las funciones y contenido que debe tener el proyecto, es decir, lo necesario para cumplir con los requerimientos del proyecto. |
|  | Estructura de desglose de trabajo (EDT) | Actividad clave que se necesita para conocer el plan de trabajo que se llevará a cabo durante la ejecución del proyecto, además, al utilizarlo se controlarán los recursos, presupuesto y tiempo del proyecto. |
|  | Plan de comunicación | El plan de comunicación nos ayudará para conocer los responsables de cada una de las actividades del proyecto. |
|  | Políticas de calidad | Estas políticas se definirán con el objetivo de que todos los integrantes del equipo conozcan los factores de calidad que JOMIMAN ha declarado para el producto final. |
| Análisis | | |
|  | Análisis de Proveedores | Se hará un análisis a nuestros principales proveedores cercanos, con el fin de obtener los mejores recursos para el desarrollo del proyecto. |
|  | Evaluar proveedores de requerimientos | Seleccionar al personal adecuado que aportará información importante para la planificación del proyecto por parte del cliente, así obtenemos información de ayuda de personal adecuado e involucrado en él proyecto. |
|  | Selección de la técnica de recolección de datos | Se definirá una técnica de recolección de datos elegida en base a un análisis, deberá ser la más adecuada para el tipo de proyecto a desarrollar. Esta herramienta servirá para obtener los requerimientos funcionales y no funcionales del proyecto. |
|  | Aplicación de la técnica de recolección de datos seleccionada | Se dará inicio a la aplicación de la técnica seleccionada anteriormente para obtener información importante y relevante para la ejecución del proyecto. |
|  | Especificación de requisitos de software (ERS) | Documento clave que en base a la información obtenida por la herramienta de recolección de datos ayudará a conocer los requerimientos que el proyecto deberá cumplir para satisfacer al cliente. |
| Diseño | | |
|  | Diagramas UML | Se harán los diagramas UML para especificar la interacción del sistema con el usuario y así mejorar el entendimiento al programador de las funcionalidades que deberá contener el proyecto. |
|  | Diagramas de la Base de datos | Actividad clave para modelar la estructura y relación de la información que deberá contener el proyecto, los diagramas a realizar son: diagrama E-R (Entidad relación) y diagrama relacional, además proporcionará una vista de como se almacenará la información necesaria para el proyecto. |
|  | Mapa de navegación | Dará conocimiento de la relación que habrá entre las diferentes paginas de la aplicación web, además de saber como esta estructurada dicha relación. |
|  | Bocetos | Información clave para el programador, ya que los bocetos darán conocimiento de la estructura del sitio y ordenamiento de cada uno de los elementos que deberá contener. |
| Codificación | | |
|  | Base de datos | Se realizará la correcta codificación de la base de datos en base a los diagramas E-R y relacional, deberá cumplir su función adecuadamente, la cual es almacenar toda la información necesaria para el funcionamiento de la aplicación web. |
|  | Maquetación | En base a los bocetos se codificará la estructura de la aplicación tanto del frontend y del backend. |
|  | Frontend | Con la maquetación del frontend se comenzará a codificar la funcionalidad de la aplicación de acuerdo a los requerimientos funcionales. |
|  | Backend | Con la maquetación del backend se comenzará a codificar el diseño de la aplicación. |
| Pruebas | | |
|  | Pruebas unitarias | Para asegurar la calidad y el cumplimiento de los requerimientos funcionales y no funcionales se realizarán pruebas unitarias a cada función o diseño funcional del sitio. |
|  | Pruebas de módulo | Para asegurar la calidad y el cumplimiento de los requerimientos funcionales y no funcionales se realizarán pruebas de módulo a cada uno de los módulos completos de la aplicación. |
| Implementación | | |
|  | Selección del hosting | En base a un análisis de diferente hosting, se hará la selección del mas adecuado para la aplicación. |
|  | Subir el sitio web | Se realizará la integración de la aplicación y base de datos al hosting web seleccionado anteriormente. |
|  | Pruebas de sistema | Una vez dentro del hosting web, se realizarán pruebas de sistema, para asegurar el correcto funcionamiento al haber cambiado de ambiente. |
| Cierre del proyecto | | |
|  | Carta de aceptación | Documento clave que especificará la aceptación del proyecto por parte del cliente, el proyecto deberá cumplir con todo lo especificado anteriormente. |
|  | Acta de cierre del proyecto | Documento clave que especificará el cierre formal del proyecto por parte del cliente. |