

**UNIVERSIDAD DE LAS FUERZAS ARMADAS  
ESPE**

**Metodología de desarrollo de Software**

**Especificación de requerimientos de software (ERS)**

**Proyecto: Sistema de control de inventario y facturación de una tienda de productos electrónicos**

**Criollo Odalys**

**Moreno Douglas**

**Lema Alex**

**Solorzano Deivy**

**Suquillo Fernando**

**3 de diciembre de 2021**

**Sangolquí – Ecuador**

1. **Consultar las metodologías más usadas actualmente en la industria del software. Realizar un cuadro comparativo.**

|  |  |  |
| --- | --- | --- |
| Metodologías | Concepto | Importancia |
| Scrum | Estructura de desarrollo incremental. Se descompone en pequeños proyectos divididos en etapas (análisis, desarrollo y testing). | Las reuniones son el pilar fundamental. Estas reuniones son:  Reuniones de planificación, diaria, revisión y de retrospectiva. |
| Kanban | También conocida como Tarjeta Visual. Donde se visualiza las distintas etapas de su trabajo hasta finalizar. | Se debe realizar un cuadro o diagrama donde se visualizará tres columnas con las tareas pendientes, en proceso y terminadas. EL cuadro debe estar alcance de todos para evitar que se repitan tareas. |
| Agile Inception | Esta constituida por una serie de dinámicas orientadas a la definición de generales de la compañía | Gira entorno al método “elevator pitch” el cual consta de pequeñas reuniones entre socios-equipo. No solo se puede utilizar en software. |
| Extreme Programming XP | Su objetivo principal es apoyar las relaciones entre empleados y clientes. Por lo que se encuentra orientada a empresas emergentes | Entre sus objetivos se encuentra el potenciar las relaciones personales, el trabajo en equipo, fomenta la comunicación y elimina los tiempos muertos. |
| Design Sprint (Metodología de Google) | Permite prototipar y validar ideas con los usuarios finales de forma directa. Propone un método de trabajo con el que rápidamente se podrá lanzar un producto de éxito. | Acelera la toma de decisiones y reduce el riesgo en los proyectos. La finalidad es construir un prototipo de testeo con los futuros clientes o usuarios. |

1. **Definir el ciclo de vida de desarrollo aplicado a un caso práctico. Utilizar el proyecto planteado en clase.**

El modelo en cascada considera las actividades fundamentales del proceso de especificación, desarrollo, validación y evolución, y los representa como fases separadas del proceso, tales como la especificación de requerimientos, el diseño del software, la implementación, las pruebas y el funcionamiento y mantenimiento del sistema.

Las principales etapas de este modelo se transforman en actividades fundamentales de desarrollo:

1. **Análisis y definición de requerimientos.** - Los servicios, restricciones y metas del sistema se definen a partir de las consultas con los usuarios. Entonces, se definen en detalle y sirven como una especificación del sistema.
2. **Diseño del sistema y del software. -** El proceso de diseño del sistema divide los requerimientos en sistemas hardware o software. Establece una arquitectura completa del sistema. El diseño del software identifica y describe las abstracciones fundamentales del sistema software y sus relaciones.
3. **Implementación y prueba de unidades. -** Durante esta etapa, el diseño del software se lleva a cabo como un conjunto o unidades de programas. La prueba de unidades implica verificar que cada una cumpla su especificación.
4. **Integración y prueba del sistema. -** Los programas o las unidades individuales de programas se integran y prueban como un sistema completo para asegurar que se cumplan los requerimientos del software. Después de las pruebas, el sistema software se entrega al cliente.
5. **Funcionamiento y mantenimiento. -** Por lo general (aunque no necesariamente), ésta es la fase más larga del ciclo de vida. El sistema se instala y se pone en funcionamiento práctico. El mantenimiento implica corregir errores no descubiertos en las etapas anteriores del ciclo de vida, mejorar la implementación de las unidades del sistema y resaltar los servicios del sistema una vez que se descubren nuevos requerimientos.

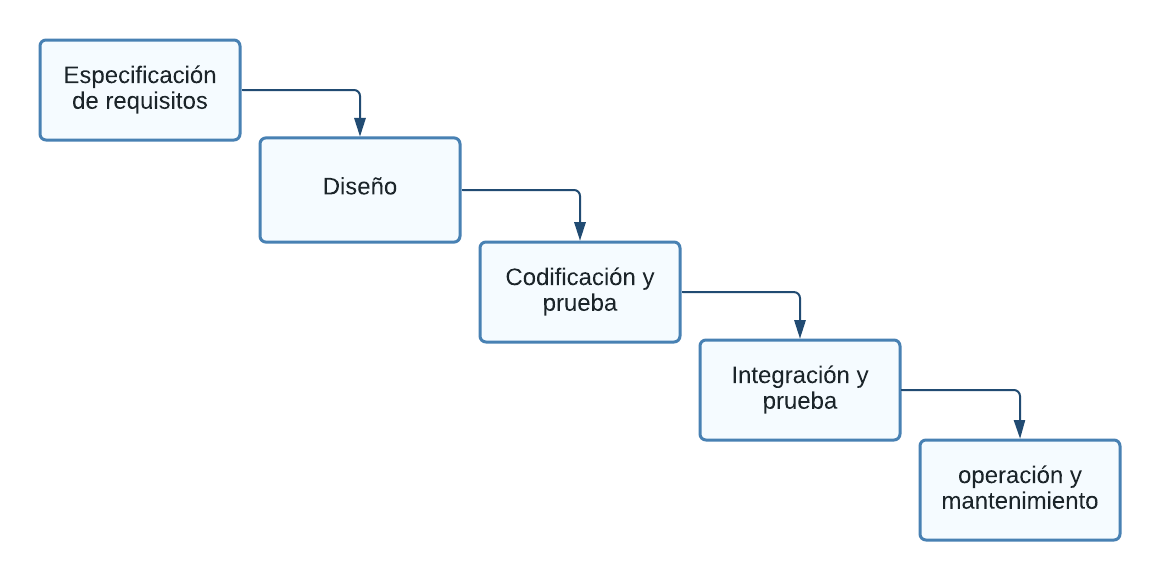


Ilustración 1. Esquema del modelo del ciclo de vida del software

Las fases que vamos a aplicar en el proyecto son:

* **Análisis y definición de requerimientos:** para poder automatizar el proceso de administración de productos de la tienda, se analizaron los requerimientos de los usuarios finales del software y los requerimientos del sistema para determinar qué objetivos debe cubrir.
* **Diseño del sistema y del software:** el diseño se aplica a unas características distintas del software que pueden elaborarse por separado, para así aprovechar las ventajas del trabajo en equipo:
* Un programa que nos permita realizar un inventario de los productos de entrada y salida, como también nos permita obtener facturación por ventas.
* Una base de datos que permita tener un stock real de los productos existentes.
* Consultas de ventas por factura.
* **Implementación y prueba de unidades:** en esta fase de programación propiamente dicha, se desarrolla el código fuente, haciendo uso de prototipos, así como pruebas y ensayos para corregir errores.

El software está constituido por tres niveles:

* ***Capa de bases de datos:*** Se implementará con persistencia en JDBC mediante JPA 2.0, a un esquema de base de datos Oracle.
* ***Capa de aplicación:*** En la parte de desarrollo se implementará la modelo vista controlador con clases, en donde se usarán DTO´s. para vincular las tablas con la aplicación.

Con un diseño orientado a objetos, en donde demostraremos cómo interactúan un conjunto de objetos entre sí, se administra su estado y operaciones.

Este será desarrollado en lenguaje Java, ya que este provee los recursos para definir las clases y un sistema para crear los objetos correspondientes a las clases.

* ***Capa de presentación:*** se desarrollará en JSF, modelos y controladores siguiendo el patrón de diseño MVC (modelo vista controlador) ejecutándose sobre un servidor Glass Fish.
* **Funcionamiento y mantenimiento:** una vez superadas la fase de pruebas, el software se entrega al cliente y comienza la vida útil del mismo, la fase de utilización se adhiere con las posteriores de mantenimiento, y durara hasta que sea remplazado por otro y deje de utilizarse.

1. **Realizar una especificación de requerimientos de acuerdo a un estándar. Utilizar el estándar IEEE830.**

<https://drive.google.com/file/d/1OS4DwdeEKc5X6D9iIrQt5M-vOkKYBKF9/view?usp=sharing>

1. **Definir un diagrama de clases y diagrama de secuencia para un caso de estudio. Utilizar un proyecto planteado.**
2. **Definir casos de prueba para una aplicación. Utilizar el proyecto planteado en clases**