**Universidad Latina De Costa Rica**

**Escuela de Ingeniería de Sistemas Informáticos**

**Materia:** Programación

**Profesor:** Jorge Isaac Vásquez Valencia

**Estudiante:** Brayan Guzmán Sánchez

**Fecha:** 21/05/2018

**Modelo de Cascada**

En este modelo cada etapa representa una unidad de desarrollo con un pequeño descanso en el medio. Por lo tanto, cada siguiente etapa inicia tan pronto como la anterior haya culminado, y esos descansos son usados para confirmaciones del lado del cliente.

Adicionalmente, este es considerado como el método tradicional de explicar el proceso de desarrollo de software en ingeniería de software, por lo que actualmente es visto como anticuado. Sin embargo, aún sigue siendo aplicado a proyectos con metas claras y requisitos que demandan hasta 100 horas de desarrollo, sobre todo considerando que este enfoque permite a los negocios deshacerse del papeleo innecesario, reuniones regulares que consumen mucho tiempo y retrasos en sus procesos de negocios.

### ****Modelo de Espiral****

Mientras que la metodología de la cascada ofrece una estructura ordenada para el desarrollo de software, las demandas de tiempo reducido al mercado hacen que sus pasos en serie sean inapropiados.

El siguiente paso evolutivo desde la cascada es donde se realizan los diversos pasos para múltiples entregas o traspasos. La última evolución de la caída del agua es la espiral, aprovechando el hecho de que los proyectos de desarrollo funcionan mejor cuando son incrementales e iterativos.

La metodología espiral refleja la relación de tareas con prototipos rápidos, mayor paralelismo y concurrencia en las actividades de diseño y construcción. El método en espiral debe todavía ser planificado metódicamente, con las tareas y entregables identificados para cada paso en la espiral.

### ****Metodología de Prototipo****

Es un procedimiento de desarrollo especializado que permite a los desarrolladores la posibilidad de poder solo hacer la muestra de la resolución para poder  validar su esencia funcional ante los clientes, y hacer los cambios que sean fundamentales antes de crear la solución final auténtica. De hecho, la mejor parte de esta metodología es que tiende a resolver un conjunto de problemas de diversificación que ocurren con el método de la cascada.

Además de esto, la gran ventaja de optar por este enfoque es que da una idea clara sobre el proceso funcional del software, reduce el riesgo de falla en una funcionalidad de software y asiste bien en la recolección de requisitos y en el análisis general.

### ****Desarrollo Rápido de Aplicaciones (RAD)****

Con el objetivo de otorgar resultados rápidos, se trata de un enfoque que está destinado a proporcionar un excelente procesos de desarrollo con la ayuda de otros enfoques, pero además, está diseñado para aumentar la viabilidad de todo el procedimiento de desarrollo de software para resaltar la participación de un usuario activo.

### ****Metodología de Programación Extrema (XP)****

Esta metodología, se utiliza principalmente para evitar el desarrollo de funciones que actualmente no se necesitan, pero sobre todo para  para atender proyectos complicados. Sin embargo, sus métodos peculiares pueden tomar más tiempo, así como recursos humanos en comparación con otros enfoques.

**Otros tipos de metodologías de desarrollo**

**1970**

* [Programación estructurada sol](https://es.wikipedia.org/w/index.php?title=Programaci%C3%B3n_estructurada_sol&action=edit&redlink=1) desde 1969
* [Programación estructurada Jackson](https://es.wikipedia.org/w/index.php?title=Programaci%C3%B3n_estructurada_Jackson&action=edit&redlink=1) desde 1975

**1980**

* [Structured Systems Analysis and Design Methodology](https://es.wikipedia.org/w/index.php?title=Structured_Systems_Analysis_and_Design_Methodology&action=edit&redlink=1) (SSADM) desde 1980
* [Structured Analysis and Design Technique](https://es.wikipedia.org/w/index.php?title=Structured_Analysis_and_Design_Technique&action=edit&redlink=1) (SADT) desde 1980
* [Ingeniería de la información](https://es.wikipedia.org/w/index.php?title=Ingenier%C3%ADa_de_la_informaci%C3%B3n&action=edit&redlink=1) (IE/IEM) desde 1981

**1990**

* [Rapid application development](https://es.wikipedia.org/wiki/Rapid_application_development) (RAD) desde 1991.
* [Programación orientada a objetos](https://es.wikipedia.org/wiki/Programaci%C3%B3n_orientada_a_objetos) (OOP) a lo largo de la década de los 90's
* [Virtual finite state machine](https://es.wikipedia.org/w/index.php?title=Virtual_finite_state_machine&action=edit&redlink=1) (VFSM) desde 1990s
* [Dynamic Systems Development Method](https://es.wikipedia.org/wiki/Dynamic_Systems_Development_Method) desarrollado en UK desde 1995.
* [Scrum](https://es.wikipedia.org/wiki/Scrum) (desarrollo), en la última parte de los 90's
* [Rational Unified Process](https://es.wikipedia.org/wiki/Rational_Unified_Process) (RUP) desde 1999.
* [Extreme Programming](https://es.wikipedia.org/wiki/Extreme_Programming)(XP) desde 1999

**Nuevo milenio**

* [Enterprise Unified Process](https://es.wikipedia.org/w/index.php?title=Enterprise_Unified_Process&action=edit&redlink=1) (EUP) extensiones RUP desde 2002
* [Constructionist design methodology](https://es.wikipedia.org/w/index.php?title=Constructionist_design_methodology&action=edit&redlink=1) (CDM) desde 2004 por [Kristinn R. Thórisson](https://es.wikipedia.org/w/index.php?title=Kristinn_R._Th%C3%B3risson&action=edit&redlink=1" \o "Kristinn R. Thórisson (aún no redactado))
* [Agile Unified Process](https://es.wikipedia.org/wiki/Agile_Unified_Process) (AUP) desde 2005 por [Scott Ambler](https://es.wikipedia.org/w/index.php?title=Scott_Ambler&action=edit&redlink=1)

**MyISAM:** Es el motor de almacenamiento por defecto de [MySQL](https://www.ecured.cu/MySQL" \o "MySQL). Se basa en el código [ISAM](https://www.ecured.cu/index.php?title=ISAM&action=edit&redlink=1) y tiene muchas extensiones útiles.

[MySQL](https://www.ecured.cu/MySQL) soporta varios motores de almacenamiento que tratan con distintos tipos de tabla. Los motores de almacenamiento de [MySQL](https://www.ecured.cu/MySQL" \o "MySQL) incluyen algunos que tratan con tablas transaccionales y otros que no lo hacen: MyISAM trata tablas no transaccionales, proporciona almacenamiento y recuperación de datos rápida

**InnoDB.** Brinda a [MySQL](https://www.ecured.cu/MySQL" \o "MySQL) un motor de almacenamiento transaccional (conforme a [ACID](https://www.ecured.cu/index.php?title=ACID&action=edit&redlink=1)) con capacidades de commit (confirmación), rollback (cancelación) y recuperación de fallas.

Características de MyISAM

* Bloqueo a nivel de tabla. Si hay actualizaciones (UPDATE) de registros, bloquea la tabla entera, lo cual es más ineficiente.
* Si se realiza una gran cantidad de consultas (SELECT), el rendimiento es más elevado. Por lo tanto, se puede decir que tiene una mayor velocidad al recuperar datos.
* En caso de errores, las tablas pueden dar problemas en la recuperación de datos. Una de las claves para ganar rendimiento viene dado por esta peor gestión de la recuperación ante errores.
* Permite hacer búsquedas full-text, las cuales son más precisas que las búsquedas con like.
* Menos consumo de recursos (memoria RAM).
* Ausencia de características de atomicidad. No tiene comprobaciones de integridad.

Características de InnoDB

* Bloqueo a nivel de registro. Lo cual es más eficiente que a nivel de tabla, ya que se permite que otras transacciones trabajen con el resto de registros.
* Soporte de transacciones.
* Mayor rendimiento para operaciones de inserción y actualización (INSERT/UPDATE).
* Mejor respuesta a la concurrencia.
* Más confiabilidad, al tener características [ACID](https://es.wikipedia.org/wiki/ACID) (Atomicidad, Consistencia, Aislamiento y Durabilidad. Esto también repercute en una mejor integridad de los datos.
* Mejor recuperación de errores a partir de sus logs.
* A partir de mySQL 5.6, también permite búsquedas full-text.

MyISAM vs InnoDB

Si aún así estás indeciso sobre qué motor de base de datos elegir, quizás estás cuestiones te puedan ayudar a decidir.

* Un solo gestor para una plataforma que requiera muchas consultas (SELECT): MyISAM.
* Se requieren muchas modificaciones de los datos (INSERT/UPDATE/DELETE): InnoDB.
* Más velocidad con pocos recursos (RAM, disco, etc.): MyISAM.
* Varios gestores de mantenimiento: InnoDB.
* Se utiliza un diseño relacional de base de datos: InnoDB.
* Es necesario una buena integridad de datos y recuperación ante errores: InnoDB.
* Necesitas búsquedas full-text: A partir de MySQL 5.6, ambos pueden realizarlas.