

**Tecnológico Nacional de México  
Instituto Tecnológico de Tijuana**

**Subdirección Académica  
Departamento de Sistemas y Computación**

**SEMESTRE:**  
FEBRERO-JULIO

**CARRERA:**  
Ingeniería en Sistemas Computacionales

**MATERIA:**  
Fundamentos de ingeniería de software

**Nombre del trabajo:**  
Tarea no. 4 Metodología tradicionales y ágiles.

**UNIDAD A EVALUAR: I**

**NOMBRE Y NÚMERO DE CONTROL DEL ALUMNO:**  
Rodriguez Vásquez Eduardo Antonio 19211724

**NOMBRE DEL MAESTRO (A):**  
REYNALDO ANDRES SALAS CARMONA

**FECHA DE ENTREGA:** 07/09/2021

## Tarea no. 4 Metodología tradicionales y agiles.

### Metodologías de desarrollo de software tradicionales

Estas se caracterizan por definir total y rígidamente los requisitos al inicio de los proyectos de ingeniería de software.

La organización del trabajo de las metodologías tradicionales es lineal, esto quiere decir que las etapas se suceden una tras otra y no se puede empezar la siguiente sin terminar la anterior. Además de que tampoco se puede volver hacia atrás una vez de que se realiza una tarea. Estas metodologías, no se adaptan nada bien a los cambios, y el mundo actual cambia constantemente, lo cual marca una gran desventaja. Las principales metodologías tradicionales o clásicas son:

- **Waterfall (Cascada):** En esta las etapas se organizan de arriba a abajo, de ahí su nombre característico. Estas se desarrollan en etapas diferenciadas y obedeciendo un riguroso orden. Antes de cada etapa se debe revisar el producto para ver si está listo para pasar a la siguiente fase. Los requisitos y especificaciones iniciales no están diseñados para cambiarse, por lo que no se pueden ver los resultados hasta que el proyecto ya esté bastante avanzado por lo que en ocasiones puede hacer que el proyecto se lleve a cabo más lento de lo normal, pero seguro.
- **Prototipado:** se realiza la construcción de un prototipo de software que se construye rápidamente para que los usuarios puedan probarlo y aportar feedback. Así, se puede arreglar lo que está mal e incluir otros requerimientos que puedan surgir. Es un modelo iterativo que se basa en el método de prueba y error para comprender las especificidades del producto.
- **Espiral:** es una combinación de los dos modelos anteriores, que añade el concepto de análisis de riesgo. Se divide en cuatro etapas: planificación, análisis de riesgo, desarrollo de prototipo y evaluación del cliente. El nombre de esta metodología da nombre a su funcionamiento, ya que se van procesando las etapas en forma de espiral. Cuanto más cerca del centro se está, más avanzado está el proyecto.
- **Incremental:** en esta metodología de desarrollo de software se va construyendo el producto final de manera progresiva. En cada etapa incremental se agrega una nueva funcionalidad, lo que permite ver resultados de una forma más rápida en comparación con el modelo en

cascada. El software se puede empezar a utilizar incluso antes de que se complete totalmente y, en general, es mucho más flexible que las demás metodologías.

- **Diseño rápido de aplicaciones (RAD):** esta metodología permite desarrollar software de alta calidad en un corto periodo de tiempo. Los costes son mucho más altos y el desarrollo más flexible, aunque requiere una mayor intervención de los usuarios. Por otro lado, el código puede contener más errores, y sus funciones son limitadas debido al poco tiempo del que se dispone para desarrollarlas. El objetivo es iterar el menor número posible de veces para conseguir una aplicación completa de forma rápida.