

Actividades del desarrollo de software



Hola:

Hasta ahora nos hemos podido enterar que la mayoría de ciclos tienen las mismas fases, aunque algunos en orden diferente y, aunque hay algunos que tienen una fase diferenciadora, la mayoría son similares. A continuación estarán los conjuntos de fases, que son las fases primarias y de gestión para el desarrollo de software y de esta manera se podrá saber qué se hace en cada una, qué productos y artefactos se trabajan en ellas.

Lo primero que se tiene que saber es que en el mundo del desarrollo de software hay dos grandes grupos de actividades: las primarias, que están conformadas por requisitos, análisis y diseño, codificación o implementación, pruebas y despliegue, que son las que conforman la parte técnica o de ingeniería. Y las secundarias, un conjunto de actividades de gestión y soporte, como por ejemplo planeación, manejo de versiones, aseguramiento de la calidad, entre otras.

La combinación de las actividades primarias y de gestión es lo que permiten tener proyectos exitosos y, por supuesto, dependiendo del tamaño del proyecto, es posible que se haga énfasis en algunas, pero casi siempre, se necesita de ambas.

Requisitos, análisis ydiseño, codificación, pruebas y despliegue

planeación, manejo de versiones, aseguramiento de la calidad

Entre las actividades primarias se encuentran:

En la parte de requisitos lo que se hace es definir claramente qué se necesita, que se desea por parte de los clientes y los usuarios. Para esto hay que interactuar con las personas para determinar las funciones y características del sistema y cuál es el valor esperado cuando se entregue ese software al cliente. Por lo general, en esta fase, se trabajan algunos artefactos como casos de uso, historias de usuario, documentos de requisitos, diagramas de proceso, entre otros.



En la anterior fase se definió el "qué" y en esta fase de análisis y diseño, se definirá el "cómo". Aquí se analizan alternativas de solución, se miran diferentes estrategias para definir cuál es la mejor, cómo se va a implementar una solución que sea exitosa con las restricciones que se tienen, pero también cumpliendo con todos los requisitos. En el análisis y diseño es muy común trabajar modelos o diagramas, como diagramas de clase, diagramas de flujo, modelos entidad-relación, también prototipos de pantalla (para el diseño visual), etc. Cuando esto ya está listo se puede pasar a la codificación o implementación, en la que esa solución que se seleccionó en el diseño se implementa utilizando uno o más lenguajes de programación. Aquí los resultados están relacionados con el código del programa, pero también con algunas pruebas manuales y documentación interna, entre otros.

0 1 0 1 0 1 1 1 1 0 1 0 1 1 1

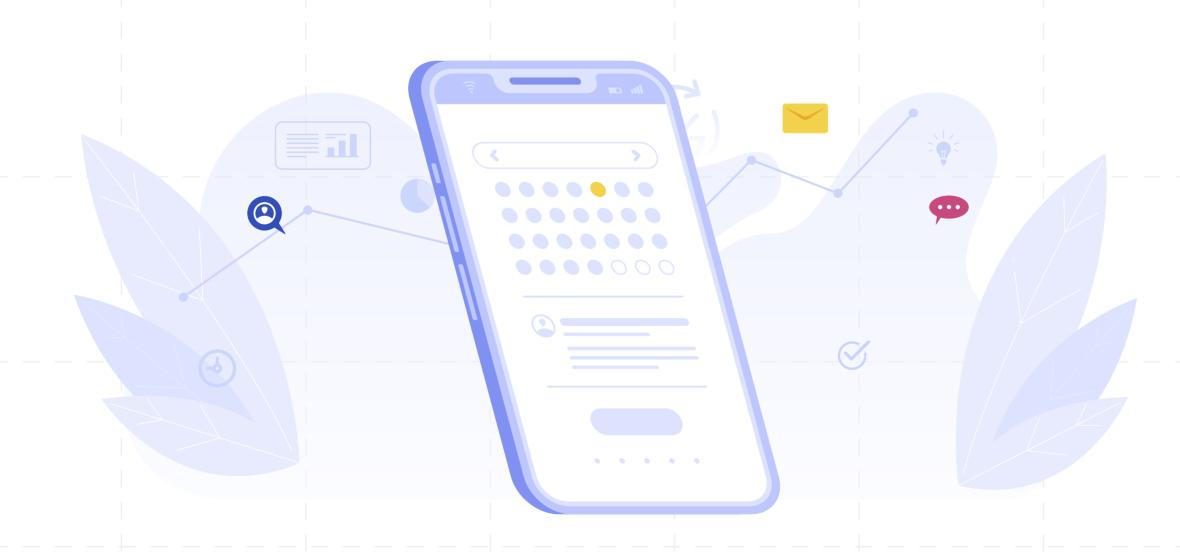
00010100001101

1001000011111000

1 0 0 0 1 1 0 0 1 1 1 1 1 0 0 1

Lo anterior lleva a la fase de pruebas. En esta actividad se tiene que verificar que realmente estamos obteniendo los resultados y los comportamientos esperados. Las pruebas se pueden hacer en diferentes momentos, aunque, la mayoría se hacen después de tener unos ciertos módulos desarrollados. Los resultados que se trabajan en pruebas incluyen el diseño de las pruebas, los resultados de la ejecución (que muchas veces son errores), pero también cierta información de calidad y algunas recomendaciones, etc.

La actividad de despliegue consiste en hacer el software disponible para su uso, por lo general en un entorno web (la nube). Anteriormente se hablaba de "instalar", que era algo diferente porque era llevar el código fuente para que fuera ejecutable y así, llevarlo a una máquina. Era un proceso relativamente sencillo. Hoy están involucrados varios servidores, muchos de ellos en la nube, con diferentes protocolos de comunicación, por lo que ya no se habla de instalar, sino de desplegar, porque implica un conjunto más amplio de actividades. Los resultados de esta fase incluyen, no solo el programa funcionando, sino también elementos de configuración de los servidores.



Entre las actividades de gestión y soporte una de las más importantes es la planeación. En esta actividad se define el tiempo y los recursos necesarios para el proyecto de desarrollo de software. Una buena planeación puede llevar a hacer buenos acuerdos con los clientes, a poner metas claras, a usar muy bien los recursos. Por eso es fundamental aprender no solo elementos técnicos, sino algunos de gestión para contribuir a una buena planeación.

Algunos de los productos o artefactos que se trabajan en esta actividad son los planes. No solo hay cronogramas, sino que hay planes de adquisición, planes de calidad, entre otros. También se trabajan tableros de control o diagramas que muestran cómo va avanzando el proyecto para hacerle seguimiento, tomar correcciones, etc.



Otra actividad es el manejo de versiones, que también se llama control de la configuración y control de cambios. En esta actividad lo que se busca es garantizar que todos estén trabajando con los componentes correctos del software. Esto se ha vuelto fundamental, sobre todo porque se está trabajando con varias personas, incluso en diferentes lugares de trabajo. Es importante sincronizarse de acuerdo a esta actividad y gracias a esta, se emplean repositorios, sistemas de control de versiones, entre otros. Además, hay una actividad de suma importancia, porque no solo las pruebas ayudan a identificar errores, existe toda una actividad de aseguramiento de calidad, que ayuda a monitorear los procesos y productos para garantizar la calidad de los mismos. En esta actividad se trabajan cosas o subactividades como revisiones, estándares, listas de chequeo, definiciones de procesos, entre otras.

Ya se ha visto un panorama de las diferentes actividades que se pueden hacer en un proceso de desarrollo de software. Así que cabe preguntar ¿Cuáles de las actividades crees que se pueden automatizar, o hasta qué nivel? y ¿en cuáles es fundamental el aporte de las personas?



Investiga un poco sobre las siguientes actividades del desarrollo de software, para saber en qué consisten: Integración y Mantenimiento.



MISION TIC2022



