Fases de desarrollo de una aplicación

Metodologías de gestión de proyectos

Métrica 3.0

Este tipo de metodología define o detalla los pasos a seguir desde que se nos presenta una oportunidad de negocio hasta que conseguimos desarrollar la aplicación y entregarla al cliente.

Nos ayuda a que todo el proceso desde desarrollo hasta entrega, se ofrezca con ciertas garantías, desviándonos el mínimo de los problemas que podríamos tener y maximizando los beneficios a obtener al final.

Es una metodología tradicional y, por lo tanto, no es tolerante a los cambios sobre la marcha, ya que está orientado a proyectos que tienen un desarrollo lento y largo En caso de que se realice algún cambio, se encontrarán problemas durante la ejecución, lo que complicará la resolución y la entrega final.

Es de estilo waterfall.

Equipo de trabajo

Stakeholders **Personas interesadas** en que el proyecto se lleve a cabo.

Pueden ser **Inversores** o **Altos cargos**.

Jefe de proyecto Es un perfil directivo que se encargan de dirigir y asegurarse de que el proyecto esté avanzando de manera correcta.

Se encargan de hacer pasar la información entre departamentos y de coordinar el equipo que está trabajando en el proyecto.

Es un perfil entre medio de los programadores y los jefes de la empresa.

Consultores Profesionales expertos en una materia que nos pueden asesorar, ayudar o impulsar en una parte del desarrollo del producto.

Analistas Son las personas que se encargan de hablar con todos para recoger información y requisitos del proyecto.

Se encargan de conocer exactamente el funcionamiento de la aplicación demandada.

Programadores Se encargan de transformar toda la información recogida por los analistas en un código ejecutable o producto deseado por el cliente.

Fases de desarrollo

Estudio de viabilidad Busca determinar si un proyecto es viable o no es viable (si me interesa hacer el proyecto o no).

Mediante el estudio de viabilidad se determina si el proyecto se puede realizar con garantías de éxito:

- Económico.
- Legal.
- Técnico.
- Estratégico.
- Operativo.

Si se determina que el proyecto no es viable, se para en esta fase de desarrollo.

Análisis Es una tarea que lleva a cabo el analista.

Busca conseguir una especificación detallada de qué es lo que tiene que realizar la aplicación.

Es la tarea más complicada, pues en ella se ha de realizar una descripción exhaustiva de lo que desea el cliente.

Requisitos funcionales Que la aplicación permita hacer login, cambiar el nombre de usuario, hablar con otros usuarios.

Requisitos no funcionales Que la aplicación no se caiga, que sea robusta, que tenga una seguridad adecuada.

Scope En esta fase del desarrollo también se define el scope o alcance del proyecto. Para así, en el caso de que el proyecto demandado sea muy ambicioso, centrar el proyecto en funcionalidades concretas.

Diagramas UML En esta fase del desarrollo también se realizarían los diagramas de **casos de uso** y de **clases**.

Para definir que es lo que los diferentes usuarios pueden hacer en la aplicación y para definir la estructura u arquitectura de la misma.

Diseño En esta fase se deben de haber reconocido todas las características y requisitos demandados por el cliente.

Se definen la arquitectura de la aplicación, así como los componentes del sistema y la estructura de datos.

Se definen los procedimientos sde migración de datos (por si hay una aplicación antigua y hay que mover sus datos a la nueva).

Definir el plan de pruebas para testear la aplicación antes de realizar la entrega.

Construcción En esta fase los programadores ya se pone manos a la obra a realizar el código de la aplicación en función de la información recibida de los anteriores pasos.

Es muy importante montar aquí el sistema de pruebas unitarias, que testean el funcionamiento del código creado para comprobar los errores.

Llevar a cabo la migración de datos iniciales.

Implantación y aceptación En esta fase se entrega y se realiza la comprobación de la aplicación directamente en la casa del cliente, comprobando así que se adapta y funciona correctamente en su empresa.

La migración de datos final se realizará paulatinamente.

Los usuarios de la empresa que tengan que utilizarla, deberán de testear todas las funcionalidades de la aplicación, para asegurarnos así de que se cumple con todo lo demandado por el cliente.

Se deberá de realizar un plan de mantenimiento de la aplicación, definiendo el tiempo de reparación de la aplicación y también un sistema de ayuda o consulta al usuario.

Diagrama de Gantt

Los diagramas de Gantt, son una herramienta que nos ayuda a planificar el desarrollo de cualquier proyecto, permitiéndonos llevar un control sobre las tareas a hacer, cuál es la evolución de esta tarea y quien está realizando la tarea.

Estos diagramas son muy poco tolerantes a los cambios y es poco cómodo de editar sobre la marcha.

Está muy ligando a técnicas tradicionales.

Tareas

Las tareas pueden tener subtareas y se descomponen en:

- Nombre o descripción
- Fecha de inicio
- Duración (días/horas)
- Relación de dependencia (relación con otras tareas)

Recursos

Recursos son recursos humanos o técnicos. En esta parte se definen las personas que harán según que tareas.

Un recurso puede estar sobrecargado o no aprovechado.

SCRUM

Scrum es lo opuesto a la **Metrica 3.0**, ya que es una tecnología ágil, enfocada a proyectos que están evolucionando constantemente.

Se plantea un desarrollo constante con entregas periódicas.

La ventaja respecto a la anterior, es que permite realizar entregas mostrando el transcurso de la aplicación. Así como enseñando las funcionalidades creadas mediante los sprints.

Los objetivos del proyecto se revisan cada vez que acaba un sprint, permitiendo estar pendiente a las tendencias del mercado y así, en el caso de que se produzca un cambio, poder actualizar el proyecto sin mayor problema.

Equipo de trabajo

Stakeholders Personas interesadas de que el proyecto se lleve a cabo. Es bastante interesante de que estas personas también participen en alguna de las reuniones del desarrollo del proyecto.

Product Owner No tiene por qué estar formado en el tema, pero es la persona que se encarga de hacer la recogida de qué funcionalidades debería de tener la aplicación para triunfar.

Es de gran ayuda para el **Development Team**, ya que este ayuda decidir que tareas hay que hacer en los sprints.

Development Team Es un equipo que se autogestiona mediante reuniones, decidiendo así qué funciones se implementarán y cuáles no durante un sprint.

Scrum Master Es un gurú de scrum, que domina la metodología de scrum y la tiene bien clara. Actúa como un soporte de cara a Scrum, siendo como un consultor para cualquier persona del equipo, ayudando a seguir las guías de scrum y de las reuniones.

Artefactos de trabajo

Product Backlog El **Product Backlog** es un como un cajón en el que se guardan todas las tareas pendientes o por hacer de la aplicación.

Pueden ser bugs, nuevos requisitos, mejoras de eficiencia, etc.

En él se guardan elementos susceptibles de ser desarrollados, aportando así valor a la aplicación.

Es administrado por el **Product Owner**.

Sprint Backlog Es un subconjunto del **Product Backlog**, en él se define que elementos serán implementados en el siguiente sprint.

El **Development Team** es el que decide qué elementos se implementan y cuáles no.

Sprint Un **sprint** es la unidad de medida mínima del tiempo.

Esta es definida únicamente una vez en cada proyecto y se ha de mantener durante todo el desarrollo del mismo.

Increment Un incremento es una aglomeración de elementos completados durante un sprint.

Se puede entender como una nueva versión del programa.

Evolución de un sprint

Sprint planning & Sprint goal Su objetivo es planificar el trabajo que se va a realizar durante un sprint. Definir un objeto que se pretende conseguir una vez se ha completado el sprint, para saber qué elementos hay que coger/dejar en el backlog.

Daily scrum Pequeñas reuniones diarias de como máximo 15 minutos que se hacen con todo el equipo de desarrollo. Sirven para planificar que tareas se van a realizar durante el día. Evaluamos el progreso que estamos haciendo hacia el final del scrum. Para evaluar y comentar bugs, problemas o impedimentos a la hora de desarrollar. Se realiza de forma autoorganizada. El scrum master forma parte de la reunion y actua como un guía **aunque no actua sobre las decisiones, únicamente como mediador**.

Sprint review Es una reunión informal que se lleva a cabo una vez se acaba el sprint.

Es una retrospectiva de lo que se ha llevado a cabo a lo largo del sprint.

Mediante el sprint review podemos llegar a la conclusion de realizar una nueva version de la aplicación con todo lo desarrollado, añadiéndole así valor a la misma.

Esto permite realizar una retrospectiva del sprint, tanto como para sacar elementos del backlog como para dropear funcionalidades que no nos convenzan.

Sprint retrospective Mediante el sprint retrospective hacemos una autoevaluación del scrum team, teniendo en cuenta las cosas a mejorar como equipo para llegar a ser más productivos.

En esta reunión también participa el scrum master, asegurando que la reunión sea constructiva y que no dure más de 3 horas en el caso de un sprint de un mes.

Kanban

Una pizarra de Kanban son básicamente columnas que definen un estado y dentro de las mismas, tarjetas de tareas.