PRÁCTICA 4. METODOLOGÍAS DE DESARROLLO SW

Análisis y Especificación de Sistemas Multimedia

Yolanda Torregrosa Hernández

Alberto García Garabal

Pablo Serna Martínez

Adrián González Herrera

# INTRODUCCIÓN

Para desarrollar nuestro proyecto debemos escoger una metodología de desarrollo de software que nos permita que la aplicación cumpla los requisitos que hemos establecido previamente. Para ello, debemos estudiar todas las metodologías existentes y seleccionar la que más se adapte a nuestro proyecto. Esto nos permitirá organizarnos mejor y establecer un orden a la hora de realizar las tareas, así como la relación entre las mismas. Dicha metodología deberá centrarse en enumerar las tareas que debemos realizar, sin entrar en detalles de cómo van a realizarse.

# METODOLOGÍAS ÁGILES

Atendiendo a la filosofía de desarrollo, las metodologías que más se adecúa a nuestro proyecto son las metodologías ágiles, que involucran activamente al cliente en el proceso y se dirigen a equipos de desarrollo pequeños. Esta elección se debe a que nuestro equipo de desarrollo se compone únicamente de cuatro miembros y la aplicación requiere un gran *feedback* con el cliente, ya que los requisitos pueden cambiar rápidamente durante el proceso de desarrollo. Además, todos los integrantes del equipo trabajaremos juntos durante todo el proyecto, facilitando la comunicación entre los diferentes miembros.

Con una metodología de SW más pesada, se invertiría más tiempo en plantear el proceso de desarrollo que en programar y probar el propio producto, y la sobrecarga de trabajo dominaría el proceso de desarrollo, lo que provocaría tanto la insatisfacción del cliente como la frustración del propio equipo de desarrollo. Sin embargo, la metodología ágil garantiza la satisfacción del cliente priorizando la entrega de software funcional de forma frecuente (en cada iteración).

# TIPOS DE METODOLOGÍAS ÁGILES

Las metodologías ágiles reducen la carga de gestión y control, siendo mucho más simples que el resto de metodologías. De entre las más conocidas destacan: **eXtreme Programming (XP), SCRUM y KANBAN.**

La metodología **eXtreme Programming** se basa en reforzar las relaciones interpersonales para llevar a cabo el proyecto, requiere gran *feedback* con el cliente y es útil para proyectos con requisitos inestables y cambiantes. En XP se distinguen 7 roles claramente diferenciados: un programador, un cliente, un encargado de pruebas o *tester*, un encargado de seguimiento o *tracker*, un entrenador, un consultor y un gestor o *Big boss*. El ciclo de vida de XP consta de 6 fases: exploración, planificación de la entrega, iteraciones, producción, mantenimiento y muerte del proyecto.

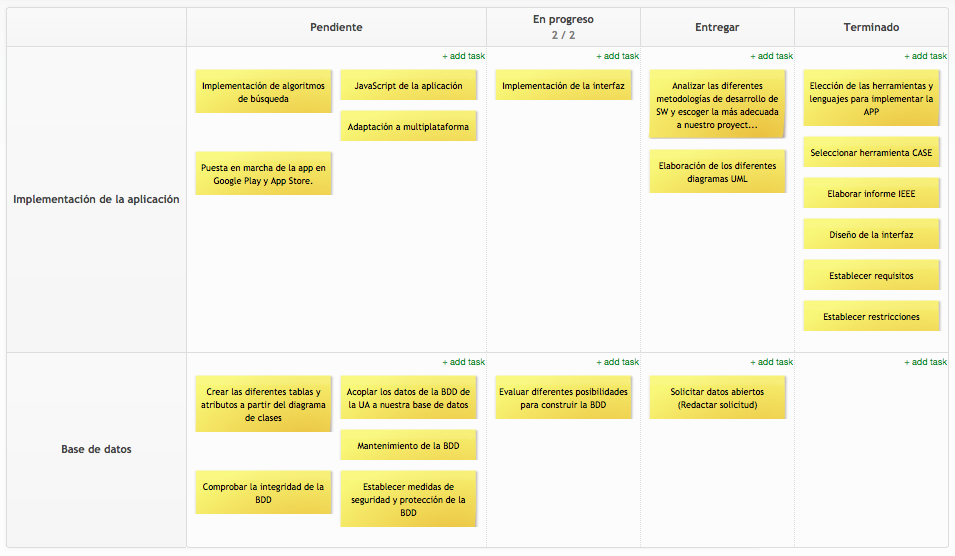
Por otro lado, **SCRUM** propone equipos cuyos miembros hacen un poco de todo durante todo el tiempo. Destaca por poseer un equipo auto-dirigido y auto-organizado en el que las iteraciones son conducidas por los clientes, entregándole a éste una demo al final de la iteración. Los roles que presenta esta metodología son: *Product owner*, *Scrum master*, equipo de desarrollo, clientes, usuarios y manager.

# METODOLOGIA UTILIZADA

Por último, **Kanban** es considerada la metodología más simple de las tres, ya que es mucho más visual. Por eso, hemos decidido utilizarla para llevar a cabo nuestro proyecto. Kanban divide el trabajo en bloques y utiliza diferentes columnas para mostrar en qué etapa está cada elemento. Además, no existen roles predeterminados y tampoco iteraciones de tiempo fijo, lo que proporciona una mayor libertad a la hora de desarrollar el sistema. Kanban mide en tiempo real la sucesión de las diferentes tareas, que es lo que se le muestra finalmente al cliente. Sin embargo, se pueden producir cuellos de botella, resultado de que una columna esté abarrotada mientras que la siguiente se encuentra vacía, aunque éstos son fácilmente detectables y solucionables.

Por tanto, para llevar a cabo nuestro proyecto, lo primero que tenemos que hacer es crear un tablero con columnas para representar los diferentes estados en el desarrollo de nuestro sistema. El siguiente paso es dividir el proyecto en diferentes tareas. A cada tarea se le deberá asociar un tiempo máximo y un valor de prioridad, además de la persona responsable de realizarla. Para ello, elaboraremos unas tarjetas en las que pondremos el nombre de la tarea, la persona que va a llevarla a cabo, la fecha de inicio y la fecha de fin, así como las restricciones que tiene cada tarea. Es tan importante evitar la sobrecarga de tareas como no asignar menos tareas de las que una persona es capaz realizar. Además, en nuestro caso sería interesante añadir secciones horizontales en mismo tablero para así dividir el proyecto en áreas de trabajo.

Este es un diagrama Kanban resultado de la simulación de utilizar esta metodología para el desarrollo software de nuestro proyecto, la aplicación transportUA:

****

# LISTA DE TAREAS

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **TAREA** | **Adrián González Herrera** | **Yolanda Torregrosa Hernández** | **Pablo Serna Martínez** | **Alberto García Garabal** |
| **Selección de la metodología de desarrollo SW** |  | **X** |  | **X** |
| **Elaboración de la presentación Power Point** | **X** |  | **X** |  |