



Universidad de
Castilla~La Mancha



Plan de Mantenimiento

Ingeniería del Software II

Empresa:
NETA.SA

1. Definición.

Hemos decidido emplear un mantenimiento **preventivo** ya que tenemos las funcionalidades logradas pero se pueden mejorar bastante las interfaces, el código fuente y la implementación de las mismas, por esto, se ve conveniente emplear este tipo de mantenimiento que permite comprobar validez de datos sobre todo en las páginas de inicio de sesión /registro y permite reestructurar el código para aportar más legibilidad al mismo.

Para esto, se ha definido para cada componente qué aspectos deberían considerarse oportunos de observar en el mantenimiento:

-Frontend: pruebas de funcionalidad de la interfaz así como pruebas para aportar un mejor diseño el cual sea funcional. Por ejemplo, se puede añadir botones que introduzcan nuevas funcionalidades de calidad para el usuario o enfocarse más en mejorar la experiencia del usuario con la interfaz. Se podría hacer **semanalmente y con 1 persona encargada**.

-Backend: revisar los logs y reestructurar el código para evitar cualquier tipo de error posible que pueda suceder y prevenir que se genere más código que no sea legible. En este apartado, conviene seguir los principios del **“clean code”** o de algunos principios software como **DRY, LoB (Locality of Behaviour), SoC (Separate of Concerns)**. En resumen, estos mantienen los principios de no repetir código, respetar la localidad de los bloques de código según el comportamiento deseado y organizar el sistema de tal forma que el planteamiento del mismo puede ser seguido en base a la localización de los bloques de código. Todo esto se puede realizar cada **4-5 días y con 3-4 personas responsables**.

-Base de datos: revisar que no haya habido ningún fallo en la BD o alguna entrada extraña, así como detectar entradas obsoletas o tablas que no se están usando apenas. Revisar los datos puede ayudar a dar ciertas ideas en algunos cambios en el backend. De frecuencia se podría hacer **cada 1 semana y por 1-2 personas**.

-Integraciones: comprobar que las integraciones del proyecto con otros software funcionan correctamente como puede ser con **Sonar, Jira** o algún otro programa. Esto se puede ir haciendo **mensualmente** ya que no suelen lanzarse versiones nuevas tan constantemente y además, estas versiones traen pequeñas actualizaciones las cuales no suelen afectar al rendimiento de proyectos que usen dichas aplicaciones. Puede ocuparse **1 persona**.

También podemos considerar otros campos que tienen que ver con el proyecto pero al ser un grupo pequeño hemos decidido limitarnos a lo básico y más importante.

2. Implementación.

NO REALIZADA.