



PLAN DE GESTIÓN DE CALIDAD



Javier Santana Delgado

Jose Antonio Santacruz Gallego

Tamara Redondo Soto

Julio Sánchez de las Heras Martín Consuegra

Alejandro Riquelme Castaño

José Antonio Oliver González-Ortega



PLAN DE GESTIÓN DE CALIDAD

ADECUACIÓN FUNCIONAL	3
FIABILIDAD.....	3
SEGURIDAD	3
COMPATIBILIDAD	4
PORTABILIDAD	4
MANTENIBILIDAD	4
CAPACIDAD DE USO.....	4

PLAN DE GESTIÓN DE CALIDAD

Se procede a **dividir el modelo de calidad del producto software** para identificar los requisitos relacionados con la calidad de nuestro producto software.

ADECUACIÓN FUNCIONAL

Para que nuestro producto software tenga **adecuación funcional**, es necesario que sea completamente funcional, cumpliendo todos los requisitos especificados por el cliente, así como guardar las reservas, las comandas, asignar un camarero a una mesa, etc.

Además, debe tener **corrección funcional** asegurándose, por ejemplo, que la hora de las reservas que se guarde en la base de datos sea la correcta y además a la hora de secuenciar los diferentes estados de las mesas estos tengan una precisión exacta.

Por último, lo que podemos comentar es la **pertinencia funcional** que tiene nuestro proyecto dado que realiza todas las funciones establecidas, para esto disponemos una serie de elementos que nos ayudan a ejecutarlas sin hacer ninguna función innecesaria ya que supondría un gasto de recurso ya sea en costes y en tiempo.

FIABILIDAD

El **producto software** debe estar **disponible en todo momento** lo que se consigue a través de un servidor en el que se encontrará la base de datos donde se obtiene y almacena toda la información relacionada con camareros, reservas, etc.

También debe **cumplir la madurez** en tanto en cuanto a que, a no ser que la base de datos quede inactiva, todas las funcionalidades deben ejecutarse correctamente. La **tolerancia a fallos** se busca por ejemplo volviendo al menú principal en caso de que el servidor se cayese y avisando al usuario del error ocurrido o por ejemplo al intentar introducir una reserva que ya existe mostrar que es lo que ocurre.

Finalmente, la **capacidad de recuperación** también debe cumplirse al tener en la base de datos mucha información requerida en tiempo real.

SEGURIDAD

En cuanto a la seguridad, nuestro programa debe tener un **alto grado de calidad** en este apartado, debido a que se está trabajando con datos sensibles, por ejemplo, la información personal de cada empleado, que debe ser privada. Por ello, nuestro producto software debe **cumplir la característica de confidencialidad** para proteger dicha información de accesos no autorizados.

Además, debe ser **íntegro**, al tener que prevenir la información de los platos, restaurantes, empleados, etc. de ser modificada sin consentimiento.

COMPATIBILIDAD

Nuestro **software** lo consideramos **compatible con otros sistemas operativos**, ya que si se desplegara todo el programa a un sistema operativo distinto de Windows, ya sea por ejemplo Linux, si este tiene la herramienta adecuada podrá ejecutarlo correctamente y también, para la conexión a la base de datos podrán modificarla ambos sistemas operativos a la vez y comprobar los diferentes resultados de cada uno.

Esto nos lleva a asegurar que se cumplen las subcaracterísticas de compatibilidad que son la **Coexistencia** y la **Interoperabilidad**.

PORTABILIDAD

Nuestro producto debe cumplir el **requisito de adaptabilidad**, debido a que va a ser usado en infinidad de dispositivos, ya sean ordenadores, tablets, smartphones, etc. debiendo funcionar de igual manera en todas ellas. Por la misma razón, debe tener la **capacidad de ser instalado fácilmente** en cualquier plataforma o dispositivo.

MANTENIBILIDAD

Al tener la intención de mantener nuestro producto software funcional y operativo a lo largo del tiempo, se deben cumplir una serie de requisitos para poder llevarlo a cabo.

En primer lugar, se debe fomentar la **modularidad del código lo máximo posible**, para minimizar el impacto de un fallo de algún componente. Además, nuestro software debe tener la capacidad de **ser fácilmente probado**, debido a que, aunque no sea un sistema crítico, nuestra intención es lanzarlo al mercado con la mínima cantidad de fallos posible por lo que necesitamos probarlo exhaustivamente.

CAPACIDAD DE USO

Se buscará fomentar la **capacidad de aprendizaje y facilitar el uso** aportando un **manual de usuario** en el que se detallen todas las funcionalidades, posibles errores, etc. Además la **interfaz será minimalista** por lo que no se necesitará un conocimiento técnico sobre esta.