

Alvarez_Valencia_Gianfranco_ED01_Tarea

Índice

1.	Apartado 1.....	3
1.1	Análisis de requisitos.....	3
1.2	Diseño.....	4
2.	Apartado 2.....	5
2.1	Codificación	5
3.	Apartado 3.....	5
3.1	Pruebas.....	5
3.2	Documentación	6
3.3	Explotación.....	6
3.4	Mantenimiento	6

Enunciado.

La empresa BK ha recibido un nuevo encargo de software.

Se trata de diseñar una aplicación para una tienda especializada en vender productos estéticos.

La tienda desea trabajar con software libre. Además, desea explícitamente que la aplicación sea capaz de cumplir las siguientes tareas:

- Proporcionar facturas de las ventas.
- Llevar la cuenta de lo que vende cada trabajador.
- Controlar el stock de productos en almacén.
- Operar con lector de código de barras y tarjetas de crédito.
- Controlar los precios de los productos y ofrecer la posibilidad de operar con ellos.
- El tiempo de respuesta de la aplicación ha de ser lo menor posible.
- No se podrán procesar dos peticiones a la vez, aunque haya varios equipos funcionando simultáneamente.
- La empresa también quiere almacenar información de sus trabajadores: DNI, nombre, apellidos, número de la Seguridad Social, fecha de nacimiento, teléfono y localidad. Asimismo, de los productos interesa almacenar: código, marca, nombre comercial, precio, cantidad.

Tendrás que diseñar una planificación del proyecto de desarrollo de ese software que cumpla con las premisas estudiadas en la presente unidad de trabajo.

1. Apartado 1

Sintetiza el análisis de requerimientos del sistema para nuestro cliente. Plantea el diseño y determina el modelo de ciclo de vida más idóneo para esta aplicación.

1.1 Análisis de requisitos

Hay que organizar las reuniones que van a tener lugar, planificando los tiempos y las etapas de las fases del desarrollo de la aplicación previniendo los posibles problemas.

Hay que conocer los requisitos hardware, los equipos en los que se usará la aplicación.

Análisis, requerimientos funcionales:

- Proporcionar facturas de las ventas.

Hay que saber cómo quieren que se muestren estas facturas.

- Llevar la cuenta de lo que vende cada trabajador.

Hay que saber de qué forma quieren que muestre esta información y por ejemplo si los trabajadores trabajan a comisión.

- Controlar el stock de productos de almacén.

Hay que saber la información que hay que almacenar de cada producto como nombre precio, así como la forma de gestionar las entradas y salidas de productos.

- Operar con lector de código de barras y tarjetas de crédito.

La aplicación tiene que ser compatible con estos “medios”.

- Controlar los precios de los productos y ofrecer la posibilidad de operar con ellos.

Hay que saber la manera en la que van a entrar los datos de los precios de los productos y como se va a ir actualizando.

- La empresa también quiere almacenar información de sus trabajadores: DNI, nombre, apellidos, número de la Seguridad Social, fecha de nacimiento, teléfono y localidad. Asimismo, de los productos interesa almacenar: código, marca, nombre comercial, precio, cantidad.

Hay que saber cómo se van almacenar estos datos y cómo van a entrar en la aplicación.

Análisis, requerimientos no funcionales:

- El tiempo de respuesta de la aplicación ha de ser lo menor posible.

Hay que controlar los tiempos de respuesta de la aplicación.

- No se podrán procesar dos peticiones a la vez, aunque haya varios equipos funcionando simultáneamente.

Hay que controlar las peticiones que se hacen a la vez.

- Respecto a todos los datos que se guarden, como los datos de los empleados, hay que estar al día con la legislación aplicable sobre estos.

1.2 Diseño

Por un lado, están los productos que tiene la empresa y las operaciones que hacen sobre estos. Hay que tener en cuenta que pueden operar con lector de código de barras y tarjetas de crédito.

Por otro lado, están los datos personales que tiene la empresa sobre los empleados.

Como uno de los requisitos es el uso de software libre se utilizará java como lenguaje de programación y MariaDB como sistema gestos de base de datos, teniendo en cuenta la eficiencia y el acceso simultaneo a los datos.

Ciclo de vida

Al parecer una ampliación sencilla en la que, por lo visto no se prevén muchos cambios, que además se especificarán los datos de entrada y salida de la aplicación correctamente, el modelo elegido para el ciclo de vida será el Modelo en Cascada con Realimentación.

2. Apartado 2

Planifica la codificación, indicando el lenguaje de programación y las herramientas que usarías para la obtención del código fuente, objeto y ejecutable, explicando por qué eliges esas herramientas.

2.1 Codificación

Como un gran porcentaje de aplicaciones están desarrolladas en lenguajes de programación orientados a objetos, el lenguaje elegido será Java, además de ser el que se enseña en el curso.

Se usará el JDK de java y el IDE blue J para editar textos compilarlo y ejecutar mediante la máquina virtual de java. Se eligen estas herramientas porque una de los requisitos es que se use SW libre.

Código fuente: Blue J.

Código objeto: Compilar desde Blue J+JDK java.

Código objeto: Ejecutar el código objeto a través de la máquina virtual de Java.

3. Apartado 3

Planifica las restantes fases del ciclo de vida, indicando en cada una el objetivo que persigues y cómo lo harías

3.1 Pruebas

Cuando ya tenga el software habrá que hacer pruebas de funcionamiento de este.

Pruebas unitarias:

Mediante JUnit habría que probar de forma separada las distintas partes del software.

Prueba integral:

Una vez pasada las pruebas independientes de las partes del software, hay que probar el sistema completo.

Habría que contactar con el cliente para planificar una prueba final en el entorno de producción de este, lo que se conoce como Beta Test. Esta prueba se llevará a cabo en los equipos del cliente.

3.2 Documentación

Hay que documentar mediante un editor de textos todas las fases de desarrollo del software. Generando los siguientes documentos:

- Guía técnica

Reflejará el diseño de la aplicación, el código, las pruebas. Estará dirigido al personal técnico encargado de mantener o controlar la aplicación.

El objetivo es facilitar el correcto desarrollo de forma que permita un fácil mantenimiento.

- Guía de uso

Reflejará la descripción de la funcionalidad de la aplicación, ejemplos de uso, los requerimientos para su ejecución y posibles casuísticas o errores. Estará dirigido a las personas que usaran la aplicación.

El objetivo es facilitar el uso a los usuarios finales.

- Guía de instalación

Reflejará la información necesaria para la puesta en marcha y explotación de la aplicación, así como la seguridad de esta. Estará dirigida al personal de informática encargado de realizar las instalaciones y actualizaciones.

El objetivo es facilitar el proceso al equipo encargado de la implementación.

3.3 Explotación

Cuando la aplicación ha pasado las pruebas y se ha documentado (Doc guía técnica) todo correctamente llega el momento de usarla en el entorno de producción final del cliente. Para ello hay que instalarlas (Doc guía de instalación) en los equipos del cliente y enseñarles a estos el funcionamiento de la aplicación (Doc guía de uso).

Cuando esto ocurra la aplicación ya podrá ser explotada por los usuarios finales en su entorno de producción final.

3.4 Mantenimiento

Habrá que ocuparse del mantenimiento de la aplicación garantizando su correcto funcionamiento en caso de que aparezcan cambios ya sean:

- Errores: cambios correctivos para solventarlos.

Adaptativos: actualizaciones de la aplicación conforme a los productos del mercado, nuevas versiones de hardware/software.

- Perfectivos: mejorar la funcionalidad del software.
- Evolutivos: añadir nuevas utilidades demandadas por el cliente.