**INFORME DE AUDITORÍA DE LA APLICACIÓN “CARNICERÍA PEPE”**

**Integrantes**

Manuel Guerrero

Enrique Maestro

Erik Velasquez

Yifei Liu

Sorin Draghici

# 

# **ÍNDICE**

[**ÍNDICE**](#_xp2i9ln7pdsp) **2**

[**DESTINATARIO**](#_uk3u0r7lp5yd) **3**

[**ENTIDAD AUDITADA**](#_ug00dyvvhhz3) **3**

[**ALCANCE**](#_tpubju2ym2ac) **3**

[**COMPARABILIDAD**](#_7pw93u1650gw) **3**

[**SALVEDADES**](#_hiuxu9m6r037) **3**

[**INCUMPLIMIENTOS**](#_889xnscnsmpx) **4**

[Evaluación Detallada de los Incumplimientos](#_fccw88j33092) 5

[**ÉNFASIS**](#_r2qgnaxhhjbw) **9**

[**INFORME DE GESTIÓN**](#_e9aiquno679) **9**

[**RESUMEN**](#_ta75sfz0vyk1) **9**

[**RESULTADO**](#_q6akclylem5x) **9**

[**FECHA Y FIRMA**](#_sakhfyv1aht) **9**

# 

# **DESTINATARIO**

El destinatario de este informe es Adrián Riesco Rodríguez.

# **ENTIDAD AUDITADA**

La entidad auditada es *TecnoService S.L*. Esta empresa de desarrollo de software está especializada en aplicaciones web para *Ventas online*. En particular, se audita el proyecto *Carnicería Pepe* desarrollado para el cliente *Carnicería Pepe S.L*.

# **ALCANCE**

Como veremos en la sección de objetivos, en esta auditoría vamos a analizar los siguientes puntos:

* **Contrato**, estudiando cómo se han gestionado los requisitos.
* **Plan de desarrollo**, fijándonos especialmente en el plan de desarrollo, las tareas críticas y los hitos.
* **Requisitos**, viendo si los requisitos están correctamente definidos, documentados y comprobando la existencia de requisitos de mantenibilidad.
* **Diseño**, estudiando la documentación y revisiones formales del diseño, verificando las revisiones tanto por pares como por expertos.
* **Implementación**, nos fijamos en las pruebas de unidad, el testing, su documentación y en las revisiones por pares.
* **Seguimiento**, comprobando los hitos, riesgos seguidos y los riesgos solucionados.
* **Documentación**, comprobando si los documentos tienen algún tipo de control, la existencia de un responsable de dichos documentos, la documentación para el testing, la documentación de instalación y uso.

# **COMPARABILIDAD**

No se dispone de información de informes de auditoría anteriores.

# **SALVEDADES**

Se han detectado las siguientes salvedades:

* No se pudo contactar con los desarrolladores, por lo que no se pudieron confirmar ciertos aspectos poco desarrollados en la memoria.
* No se pudo contactar con el cliente, para conocer el estado del proyecto en el momento de la auditoría.
* Falta información acerca de los tipos de revisiones llevadas a cabo.
* No se tiene constancia de la participación de expertos externos en el desarrollo del proyecto.

# 

# **INCUMPLIMIENTOS**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Item** | **Objetivo** | **Sí** | **No** | **Parcial** | **N.A.** |
| **1.** | **Contrato** | | | | |
| 1.1. | Se reflejan los requisitos | **X** |  |  |  |
| 1.2. | Se justifican los requisitos rechazados |  | **X** |  |  |
| 1.3. | Se tienen en cuenta los riesgos |  |  | **X** |  |
| **2.** | **Plan de desarrollo** | | | | |
| 2.1. | Plan de desarrollo definido | **X** |  |  |  |
| 2.2. | Plan de desarrollo documentado |  |  | **X** |  |
| 2.3. | Tareas críticas establecidas |  | **X** |  |  |
| 2.4. | Hitos establecidos | **X** |  |  |  |
| **3.** | **Requisitos** | | | | |
| 3.1. | Requisitos claramente definidos |  |  | **X** |  |
| 3.2. | Requisitos claramente documentados |  |  | **X** |  |
| 3.3. | Existen requisitos de mantenibilidad |  |  |  | **X** |
| **4.** | **Diseño** | | | | |
| 4.1. | Diseño claramente documentado |  | **X** |  |  |
| 4.2. | Revisiones formales de diseño | **X** |  |  |  |
| 4.3. | Revisiones por pares |  | **X** |  |  |
| 4.4. | Revisiones por expertos |  |  |  | **X** |
| **5.** | **Implementación** | | | | |
| 5.1. | Pruebas de unidad |  |  |  | **X** |
| 5.2. | Testing documentado |  |  | **X** |  |
| 5.3. | Revisiones por pares |  | **X** |  |  |
| **6.** | **Seguimiento** | | | | |
| 6.1. | Hitos seguidos |  |  | **X** |  |
| 6.2. | Riesgos seguidos |  |  | **X** |  |
| 6.3. | Riesgos solucionados |  | **X** |  |  |
| **7.** | **Documentación** | | | | |
| 7.1. | Documentos controlados |  |  | **X** |  |
| 7.2. | Responsable de documentos controlados |  | **X** |  |  |
| 7.3. | Documentación para testing |  |  | **X** |  |
| 7.4. | Documentación de instalación y uso |  |  | **X** |  |

## **Evaluación Detallada de los Incumplimientos**

**1.1.** Los requisitos funcionales y no funcionales se reflejan en el documento “*Contrato Carnicería*”, en la sección de anexos (Anexo I) entre las páginas 12 y 14.

**1.2.** No hay constancia en ningún documento de los requisitos rechazados.

* **Recomendación**: Mencionar los requisitos rechazados junto con la justificación, permitirá realizar un trazado de la evolución de las funcionalidades del sistema, y evitar que el el futuro se introduzcan de nuevo.

Es de buenas prácticas el uso de control de documentos para facilitar el seguimiento de los cambios.

**1.3.** Los riesgos se reflejan en el documento “*Plan de riesgos*”, aunque no hay detalles o planes de acción para cada riesgo en específico.

* **Recomendación**: Especificar los planes de acción para cada riesgo presentado en el documento, tomando en cuenta todos los riesgos, y de esta forma preparar un plan para prevenir cualquier clase de problema que pueda surgir antes, durante y después del desarrollo del proyecto.

Agregar el plan de reducción, plan de supervisión y plan de contingencia para los riesgos que no contengan.

**2.1.** En el documento “*Contrato Carnicería*” se especifican las cláusulas: “SEGUNDA.- Fases de desarrollo”, “TERCERA.- Entrega de los productos de software”, “CUARTA.- Validación de las entregas parciales”, “QUINTA.- Duración”. En dichas cláusulas queda recogido formalmente lo que se espera del plan para el desarrollo del proyecto. Además, puede encontrarse información adicional que desglosa estos puntos en el Anexo I “Especificación de Requisitos Software” y el Anexo II “Fases del desarrollo”.

**2.2.** Existe un documento “*Gantt Carnicería*” que muestra el plan de desarrollo de forma diagrama de gantt. “Plan de riesgos” contiene la parte de análisis de riesgo pero está simplificada y sin explicación de los riesgos.

* **Recomendación**: Establezca una planificación temporal detallada que incluye estimación de recursos de actividades específicas con información de distribución de horarios, y hacer el análisis de riesgos más detallado.

**2.3.** No se ha encontrado ninguna referencia a las tareas más críticas para lograr el cumplimiento del proyecto en las fechas señaladas.

* **Recomendación:** Es importante calcular el camino crítico, es decir, el conjunto de tareas en las que si se produce un retraso, automáticamente producirán un retraso en la planificación del proyecto. Dicho camino crítico sería beneficioso calcularlo haciendo el grafo PERT equivalente al diagrama de Gantt del proyecto, esto permitirá calcular ciertas métricas, por ejemplo el tiempo esperado para la finalización del proyecto o la varianza de las actividades que lo componen.

**2.4.** Los hitos, así como sus fechas y las actividades necesarias para alcanzarlos, están detallados tanto en el documento “*Contrato Carnicería*” en el apartado Anexo II “Fases del desarrollo”, como en el documento “*Gantt Carnicería*”.

**3.1.** Aunque los requisitos se reflejan en el documento del Contrato, en la sección de anexos (Anexo I) entre las páginas 12 y 14, carecen de suficiente detalle, tanto en especificación, como en alcance necesario para su desarrollo. Además, los requisitos funcionales son descritos de la misma forma que los requisitos no funcionales, y ocurre que algunos de ellos están compuestos por varios requisitos.

* **Recomendación:** Tiene que haber una clara diferencia entre los requisitos funcionales y no funcionales. Para esto, no se debe utilizar el lenguaje natural para evitar causar ambigüedades. Se recomienda el uso del lenguaje estructurado para acotar de forma clara los requisitos funcionales, los cuales deben indicar servicios que proporcionan el sistema, la respuesta ante determinadas entradas, o el comportamiento en situaciones particulares. Para ello, se ha de especificar en detalle las características de cada requisito, incluyendo al menos: prioridad, estabilidad, descripción, entrada, salida, origen, destino, acciones, precondición, postcondición, efectos laterales, y los usuarios con permiso de realizar la función. Desglosar cada requisito en una única funcionalidad, sin necesidad de combinar varios requisitos en uno mismo.

**3.2.** Los requisitos están documentados en un apartado anexo al contrato del software.

* **Recomendación:** Realizar un documento de “Especificación de Requisitos de Software (SRS)” donde se defina claramente el sistema a desarrollar, y mencionarlo en el Contrato como documento anexo al mismo. En relación al apartado 3.1. de la auditoría, en la SRS se deben incluir los requisitos funcionales y no funcionales, además de los casos de uso para aclarar las interacciones que los usuarios pueden tener con el sistema.

**3.3.** Debido al reducido tamaño, alcance del sistema, y el hecho de que en el contrato se menciona en la cláusula decimocuarta que la actividad de mantenimiento no es requerida puesto que tendría que quedar fijada en un nuevo contrato, se ha decidido que los requisitos de mantenibilidad no son de interés en el presente proyecto.

**4.1.** No se han encontrado documentos relacionados al diseño del sistema.

* **Recomendación:** Los documentos de diseño son los relacionados a la estructura arquitectónica del sistema. El no tener dichos documentos impiden definir claramente el alcance y/o construcción del sistema y su despliegue. Se deben de desarrollar los siguientes documentos antes de comenzar con la fase de desarrollo: diseño de sistema, diseño de Base de datos y diseño de interfaz de usuarios.

**4.2.** La información se refleja en el documento “*Revisiones formales*”.

**4.3.** No hay constancia de revisiones por pares de los documentos de diseño.

* **Recomendación**: Realizar la revisión por pares de los documentos de diseño y revisiones formales para así mantener los estándares de calidad, mejorar el rendimiento y proporcionar credibilidad al sistema.

**4.4.** No se da ninguna información en la documentación sobre revisiones por expertos. Sin embargo, dado que la aplicación *Carnicería Pepe* es suficientemente sencilla y la empresa *TecnoService S.L* cuenta con sobrada experiencia en el desarrollo de este tipo de aplicaciones, como se indicó en la explicación de la entidad auditada, consideramos que no era relevante en este caso.

**5.1.** Realizar pruebas de unidad permite comprobar de forma automática el correcto funcionamiento de las distintas partes no triviales del código. Sin embargo, en este caso concreto, debido al reducido tamaño, alcance, y el hecho de que es más preferible realizar las pruebas de forma manual por la naturaleza del sistema, se ha decidido que las pruebas de unidad (junto con la integración continua) no son de elevado interés en el presente proyecto.

Adicionalmente, en el documento “*Revisiones formales*” queda reflejado que se prevé realizar las pruebas de unidad en los siguientes hitos.

**5.2.** Los resultados obtenidos de las pruebas realizadas han quedado grabados en el documento “*Plan de pruebas funcionales*” aunque no de manera detallada.

* **Recomendación**: Detallar la evolución de las pruebas a lo largo del desarrollo, permitirá realizar un trazado de la evolución del sistema, y en caso de incongruencias futuras, se podrá emplear como base para la subsanar los problemas.

**5.3.** No hay constancia de revisiones por pares de los documentos de diseño.

* **Recomendación:** Realizar la revisión por pares de los documentos de diseño y revisiones formales para así mantener los estándares de calidad, mejorar el rendimiento y proporcionar credibilidad al sistema.

**6.1.** Se cumplieron en su mayoría los hitos pero no todos. Los hitos iniciales definidos en el proyecto según el documento Gantt se completaron en su totalidad (Hito 1, Hito 2, Hito 3), pero se observa que no fueron completados los siguientes hitos en el proyecto (Hito 4 y Hito 6) referentes a integración y entrega al cliente.

* **Recomendación**: Hacer un seguimiento completo de los Hitos del proyecto, como su correcta culminación y entrega. Minutas de reuniones de finalización de hitos ayudarían a llevar este control.

**6.2.** Los riesgos y planes de contingencia están considerados en el documento “Plan de riesgos”, pero no podemos concretar si se han seguido las pautas en el caso de que ocurriera algún problema, esto debido a la falta de documentación sobre el seguimiento o aplicación de estos riesgos en el proyecto, pero debido al inclumiento de hitos en el desarrollo del proyecto, podemos considerar que no se siguieron en su totalidad los riesgos.

* **Recomendación**: Realización de un seguimiento y control de riesgos en el proyecto, como un control de acciones y eventualidades que surjan en el desarrollo del proyecto..

**6.3.** Los riesgos no fueron solucionados, debido a que no se alcanzaron todos los hitos especificados en el proyecto.

* **Recomendación**: Realización de un mejor plan de riesgo y ejecución de una mejor gestión de riesgo.

**7.1.** Todos los ficheros relacionados con la aplicación se controlan a través de un repositorio de github como puede deducirse del apartado “Descargar proyecto” del documento “*Manual de instalación*”.

Los siguientes documentos se controlan de forma manual: Contrato, manual de instalación, manual de usuario, plan de pruebas funcionales y revisiones formales. Todos ellos siguen una plantilla especificada en el documento “*Plantilla Documentos*” para el control de versiones, esta incluye la siguiente información: El tipo de documento, el número de versión, la fecha en la que entra en vigor la versión y el autor del cambio.

* **Recomendación:** Utilizar una nomenclatura que permita identificar de forma unívoca cada documento, esta habrá de incluir a parte del nombre del nombre, su versión, su revisión y una fecha asociada. Comenzar a utilizar un software para la gestión de los documentos del proyecto, este ha de gestionar el almacenamiento de los documentos y permitir la recuperación de las versiones, revisiones y eliminaciones que se realicen. Además de controlar: El autor del documento, los personas que aprobaron el documento, la fecha de aprobación, las firmas del autor y de quienes lo aprobaron, la descripción de los cambios introducidos, la lista de personas a quienes corresponde recibir dicho documento y las cláusulas de confidencialidad asociadas a la información que contenga el documento.

**7.2.** No se ha encontrado ninguna referencia a una persona que se haga responsable de control de la documentación del proyecto.

* **Recomendación:** Se requiere nombrar a un responsable de la documentación del proyecto que se ocupe de definir la lista de los documentos controlados y los procedimientos de documentación, estos incluirán: Los requisitos para su preparación, aprobación, almacenamiento y recuperación.

**7.3.** La documentación encontrada en el documento “Plan de pruebas funcionales” solamente recoge los test de caja negra, es decir, aquellos que comprueban el correcto funcionamiento de la funcionalidad ofrecida por el sistema sin tener en cuenta como se ha realizado su codificación. Los test de caja negra no recogen las entradas ni las salidas del sistema, solamente recogen si el test ha resultado exitoso o no, y en caso de fallo, un breve comentario indicando su origen a alto nivel. Con respecto a las pruebas de caja blanca, aquellas en las que se pone a prueba el correcto funcionamiento, la eficiencia y la seguridad del código de las funciones aplicación, se han encontrado los siguientes ficheros vacíos: apps/core/tests.py, apps/product/tests.py, apps/shop/tests.py, apps/ticket/tests.py. Por lo que no se dispone de ninguna prueba de este tipo.

* **Recomendación:** Acompañar la documentación de las pruebas de caja negra con todos los valores de entrada y salida que se han generado. Definir y ejecutar un plan de pruebas para los requisitos no funcionales del proyecto.

**7.4.** Ambos ficheros están disponibles en la carpeta “Documentación A Cliente” y sus cambios son controlados de forma manual.

* **Recomendación:** Utilizar software para el control de la documentación permita gestionar la siguiente información: Versiones, revisiones, cambios y eliminaciones del documento, autor del documento, fecha de aprobación, autores y fechas de los cambios introducidos, etc. El documento de “Manual de instalación” es importante que especifique los requerimientos de hardware y software mínimo necesarios para la instalación del sistema, especificación de contingencias en caso de posibles errores en el proceso, referencias a documentación de las herramientas usadas.

# **ÉNFASIS**

Los problemas más graves encontrados han sido la definición de los requisitos y la falta de documentos de diseño necesarios para la construcción y el despliegue del producto.

El documento de Especificación de los Requisitos de Software es uno de los documentos claves del desarrollo de un sistema informático ya que en él se fija de forma detallada y precisa los requisitos del producto, de forma que sirvan como base para elaborar un contrato.

# **INFORME DE GESTIÓN**

En esta auditoría no hemos tenido acceso a documentación sobre la gestión.

# **RESUMEN**

Hemos encontrado incumplimientos en los hitos de entrega, el seguimiento y la documentación. No queda constancia por parte del cliente que estos incumplimientos ocasionarán un problema mayor en la entrega y puesta en producción del sistema.

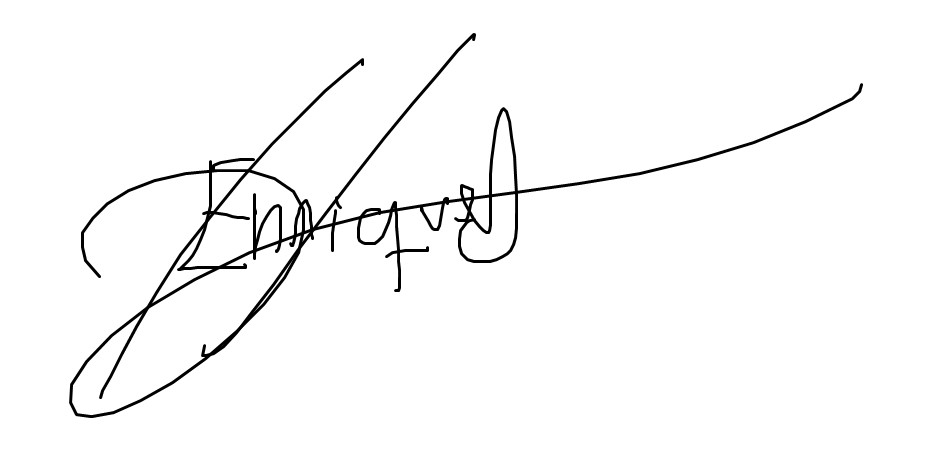
Por ello, recomendamos realizar un seguimiento de los hitos de entrega en conjunto con el cliente, mejorar la definición de los requerimientos, y realizar los procesos de planificación y diseño del sistema. Esto mediante un documento de SRS, planificación y seguimiento de riesgos.

# **RESULTADO**

Favorable **CON SALVEDADES.**

Se recomienda revisar los requerimientos y la documentación, tal y como se ha indicado en los puntos anteriores.

# **FECHA Y FIRMA**



Firmado a 14 de noviembre de 2020 en Madrid