



ugr

Universidad
de Granada

TRABAJO FIN DE GRADO
INGENIERÍA EN INFORMÁTICA

NO-INVENTORY

Sistema de gestión para inventarios

Autor

César Hugo Bárzano Cruz

Directores

Nombre Apellido1 Apellido2 (tutor1)

Nombre Apellido1 Apellido2 (tutor2)



ESCUELA TÉCNICA SUPERIOR DE INGENIERÍAS INFORMÁTICA Y DE
TELECOMUNICACIÓN

Granada, mes de 201



Título del proyecto

Subtítulo del proyecto.

Autor

Nombre Apellido1 Apellido2 (alumno)

Directores

Nombre Apellido1 Apellido2 (tutor1)

Nombre Apellido1 Apellido2 (tutor2)

NO-INVENTORY: Sistema de gestión para inventarios

César Hugo Bárzano Cruz

Gestión, Almacén , Activos, Eficiencia, Ahorro, Informes, **Palabras clave:** Gestión_clave1, Almacen_clave2, Activos_clave3, Eficiencia_clave4, Ahorro_clave5 Informes _clave6

Resumen

Poner aquí el resumen.hhhhhhhhhhhhhhhhhhhhhh

Project Title: Project Subtitle

First name, Family name (student)

Keywords: Keyword1, Keyword2, Keyword3,

Abstract

Write here the abstract in English.

Yo, **César Hugo Bárzano Cruz**, alumno de la titulación TITULACIÓN de la **Escuela Técnica Superior de Ingenierías Informática y de Telecomunicación de la Universidad de Granada**, con DNI 77138361h, autorizo la ubicación de la siguiente copia de mi Trabajo Fin de Grado en la biblioteca del centro para que pueda ser consultada por las personas que lo deseen.

Fdo: Nombre Apellido1 Apellido2

Granada a X de mes de 201 .

D. **Nombre Apellido1 Apellido2 (tutor1)**, Profesor del Área de XXXX del Departamento YYYY de la Universidad de Granada.

D. **Nombre Apellido1 Apellido2 (tutor2)**, Profesor del Área de XXXX del Departamento YYYY de la Universidad de Granada.

Informan:

Que el presente trabajo, titulado ***Título del proyecto, Subtítulo del proyecto***, ha sido realizado bajo su supervisión por **Nombre Apellido1 Apellido2 (alumno)**, y autorizamos la defensa de dicho trabajo ante el tribunal que corresponda.

Y para que conste, expiden y firman el presente informe en Granada a X de mes de 201 .

Los directores:

Nombre Apellido1 Apellido2 (tutor1) **Nombre Apellido1 Apellido2 (tutor2)**

Agradecimientos

Poner aquí agradecimientos...

Índice general

Capítulo 1

Introducción

Este proyecto tiene el objetivo de solucionar un problema real. Desde principios de 2012, la Oficina de Software Libre (OSL) tiene un acuerdo con Unidad de Calidad de la Universidad de Granada. Dicho acuerdo establece que la oficina es la encargada de recoger el material informático procedente de los distintivos organismos de la Universidad. Debido a la acumulación de material a lo largo de estos 5 años, la oficina de software libre necesita un sistema de gestión que le permita catalogar y clasificar este material.

El problema principal radica en el desconocimiento del estado en el que este material es recogido, es decir, la OSL no sabe si el material recibido funciona, no funciona, necesita ser actualizado o directamente es inservible y ha de ser reciclado. Hay que tener en cuenta el estado actual del almacén. Tras acumular material a lo largo de estos años, el almacén sufre una situación caótica debido al gran número de equipos sin catalogar ni reflejar el estado en el que se encuentran.

Por otra parte, el acuerdo mencionado anteriormente, obliga a la oficina a realizar informes cada cierto periodo de tiempo, reflejando el estado de las campañas de recogida, el estado del material recolectado, el peso...etc

El problema presentado no es algo aislado a la OSL, gran parte de las pequeñas y medianas empresas con almacenes que gestionar presentan problemas similares. Para las empresas, no usar plataformas de gestión para sus productos supone carencia de estrategias de crecimiento, una inadecuada utilización de las tecnologías y conocimientos, pérdidas de recursos, debilidad financiera y deficiencias en toda la organización.

En los últimos años, el análisis de la información asociada a los recursos o servicios de las empresas, ha tomado una gran importancia. El análisis de la información se ha convertido en un factor clave para la toma de decisiones en diversos sectores. El valor de la información ha alcanzado el nivel de otros recursos como por ejemplo los recursos financieros, materiales y humanos. El conocimiento del entorno y la información que podemos aprovechar de él son vitales para la toma de decisiones, ya sea fuera del entorno empresarial o

dentro de él. Decisiones como por ejemplo entrar en nuevos mercados, proteger la empresa de agentes externos, innovación, desarrollo, inversiones...etc. Por ello las Tecnologías de Información forman un factor decisivo para dar lugar al crecimiento tanto de las PyMEs como de cualquier empresa.

Lo principales motivos por los que estas empresas no utilizan sistemas de gestión para sus productos se deben a los costes elevados, a la carencia de recursos, la falta de acceso a la información, etc...

Las Tecnologías de la Información y Comunicaciones (TIC), se han convertido en la herramienta perfecta para llevar a cabo las tareas necesarias para aumentar la eficacia y eficiencia en el manejo de los recursos de la empresa.

Capítulo 2

Objetivos

Objetivo general a conseguir: se enuncia lo que se quiere hacer sin entrar en detalles.

Objetivos específicos: Es dividir el objetivo general en los pasos a seguir con sub-objetivos más simples. Dicho de otro modo, son cada uno de los pasos a realizar para alcanzar el objetivo general, es decir solucionando todos y cada uno de los objetivos específicos se resuelve el objetivo general. (Poner entre 3 y 5 como mucho).

Se puede decir en qué apartado se tratará cada objetivo específico.

Poner también los aspectos formativos previos utilizados, por ejemplo si se han usado técnicas de visión concretas como Transformada de Hough, Método de detección de rostros Viola-Jones, Filtro de Partículas, o técnicas de aprendizaje por SVM, etc. Se explica un poco cada método.

Capítulo 3

Resolución del trabajo

Como método de ingeniería del software decir que vamos a seguir la técnicas de modelo de prototipos rápido o también llamado modelado de prototipado rápido.

Ver http://www.ecured.cu/index.php/Modelo_de_Prototipos

3.1. Recursos

Decir los recursos humanos (autor y directores), hardware y software que se van a utilizar.

3.2. Especificación de requisitos

Decir que se partió de una especificación inicial de requisitos que a medida que se fueron implementando los prototipos se fue refinando posteriormente. Se puede poner la inicial y la final o solo la final indicando que se están poniendo los requisitos que finalmente tiene que tener el sistema.

Los requisitos se pueden referir a las necesidades del usuario del sistema (requisitos del usuario), a lo que tiene que hacer la aplicación (requisito funcional) o a cómo tiene que hacerlo (requisito no funcional). Ejemplo:

En este sistema un robot tiene que coger con sus pinzas un envase de medicamento y llevárselo a una persona anciana que por sí misma no puede recordar su medicación.

Requisitos del usuario:

RU1. La persona puede moverse libremente por una habitación donde está el robot.

RU2. La persona es capaz de coger el medicamento cuando se lo ofrece el robot.

RU3. La persona es capaz de tomarse el medicamento por sí misma.

Requisitos funcionales.

RF1. El robot mediante la cámara kinect debe poder localizar a la persona.

RF2. El robot conoce la posición de la mesa pues tiene un mapa de la habitación.

RF3. El robot debe identificar el medicamento correcto según un plan de medicación previamente establecido.

RF4. El robot debe poder coger el medicamento con sus pinzas.
etc...

Requisitos no funcionales

RNF1. El robot no puede atropellar ni dañar a la persona en ningún momento.

RNF2. La aplicación debe ejecutarse en entornos linux

RNF3. La aplicación debe utilizar pocos recursos para reaccionar con rapidez.

algo de la interfaz, como tratar posibles fallos, etc.

3.3. Planificación

Poner una tabla de tiempos con las planificación del proyecto diciendo cuando se tiene previsto alcanzar cada subobjetivo planteado. Con su correspondiente división en fases y tareas, y la posterior comparación con los datos reales obtenidos tras realizar el proyecto. Entre las fases está la realización de los diferentes prototipos I, II y III por ejemplo.

Poner presupuesto según horas de trabajo estimadas.

3.4. Análisis funcional

A partir de aquí nos referimos solamente al prototipo final que da lugar a la aplicación final.

Hay que describir la funcionalidad que debe poseer el sistema para poder cumplir con los objetivos y requisitos que se han dicho previamente. La descripción de esta funcionalidad puede hacerse analizando las tareas (que aparecerán en la planificación) y estudiando la inter-relación entre ellas y sus conexiones.

Para la realización de este análisis se pueden utilizar Diagramas de Flujo para poder conocer generalmente un único punto de inicio y un único punto de término o en varios.

https://es.wikipedia.org/wiki/Diagrama_de_flujo

Se pueden plantear también casos de uso. Los diagramas de casos de uso sirven para describir la inter-relación entre el sistema y el usuario del mismo. Se pueden utilizar para plantear diferentes casos de interacción entre el robot y la persona y cómo tiene que reaccionar el sistema en cada caso.

https://es.wikipedia.org/wiki/Diagrama_de_casos_de_uso

3.5. Implementación y pruebas

Decir qué lenguaje de programación se ha utilizado y las tecnologías implicadas aunque se hayan comentado en el apartado de recursos. Justificar su uso (rendimiento, disponibilidad, etc.). Si se ha usado open source decirlo y explicar las ventajas.

Las pruebas se realizan para comprobar la verificación y validación del producto software. La verificación consiste en comprobar que el producto realiza lo que está programado, es decir la programación no tiene errores y funciona en todos los casos cumpliendo los requisitos. La validación tiene que ver con que cumpla con lo que espera el usuario.

Verificación y Validación: Conjunto de procesos de comprobación y análisis que aseguran que el software que se desarrolla está acorde a su especificación y cumple las necesidades de los clientes.

Verificación: ¿Estamos construyendo el producto correctamente? e su especificacin.

Validación: ¿Estamos construyendo el producto correcto? Comprueba que el software cumple las expectativas que el cliente espera Importante: Nunca se va a poder demostrar que el software está completamente libre de defectos

las pruebas que pueden utilizarse son muy diversas. Aconsejo centrarnos en pruebas de caja blanca y de caja negra.

Las pruebas de la caja blanca se centran en la estructura interna del programa para elegir los casos de prueba. El objetivo de estas pruebas consiste en probar todos los posibles casos de ejecución de la aplicación para comprobar que los datos se comportan de manera correcta internamente.

Decir que se han hecho las pruebas de caja blanca.

Las pruebas de caja negra son aquellas que se centran en las salidas y entradas de los módulos, sin atender a su comportamiento interno (comprobando mediante las pruebas de caja blanca). Las pruebas de caja negra garantizan la interconectividad entre los diferentes módulos de la aplicación, así como su correcto funcionamiento final.

Poner algunos casos de prueba de caja negra.

Capítulo 4

Conclusiones y trabajo futuro

Decir lo que se ha conseguido realizar comentando sus puntos fuertes y débiles. Decir si se han alcanzado los objetivos específicos y el general propuesto y en qué grado.

Indicar las asignaturas del grado más relacionadas con la ejecución del TFG y cómo el TFG ha ayudado a afianzar los conocimientos adquiridos en el Grado.

Valoración personal si se quiere.

Avanzar algunas líneas de trabajo futuro para solucionar las debilidades detectadas o para conseguir nuevas funcionalidades interesantes.

Capítulo 5

Bibliografía

Poner las citas bibliográficas, direcciones de internet, etc.

Capítulo 6

Anexo

Al final de la memoria hay que añadir un anexo de una página o dos, explicando como se usa el software a modo de manual de usuario. Es decir, como se llaman los comandos, qué parámetros hay que darle, como se llaman los ficheros de datos de entrada, etc.

