UNIVERSIDAD DE BURGOS ESCUELA POLITÉCNICA SUPERIOR



Grado en Ingeniería en Informática



TFG del Grado en Ingenier´ıa Inform´atica

**LucErik**

Presentado por Carolina Colina Zamorano

en Universidad de Burgos — 20 de enero de 2021 Tutor: D. José Ignacio Santos Martín y D. José Manuel Galán Ordax

**AGRADECIMIENTOS**

A todos mis profesores que a lo largo de tantos años me han enseñado que con esfuerzo y dedicación todo se consigue.

A mis amigos. A Javi por animarme todos y cada uno de los días desde que comencé este proyecto y a Sonia, que a pesar de vivir momentos tan inmensamente difíciles siempre tiene un momento para llamarme y darme su apoyo y confianza.

A mi hermano. Eres y serás mi persona en el mundo.

A mis padres, que con su infinito amor y la inmensa confianza en mí, han hecho que me vea capaz de cualquier cosa. No existe manera de agradeceros tanto. Soy lo que soy gracias a vosotros y estoy tremendamente orgullosa. Mis logros son todos vuestros.

Y por último, a mi hermanita. Gracias por haberme dado a Lucas y Erika que son mi inspiración y el nombre de este proyecto.

UNIVERSIDAD DE BURGOS ESCUELA POLITÉCNICA SUPERIOR



Grado en Ingeniería en Informática

D. José Ignacio Santos Martín y D. José Manuel Galán Ordax, profesores del departamento de nombre departamento, área de Ingeniería de Organización.

Expone:

Que el alumno D. Carolina Colina Zamorano, con DNI 71287077A, ha realizado el Trabajo final de Grado en Ingeniería Informática titulado LucErik.

Y que dicho trabajo ha sido realizado por el alumno bajo la dirección del que suscribe, en virtud de lo cual se autoriza su presentación y defensa.

En Burgos, 20 de enero de 2021

Vº. Bº. del Tutor: Vº. Bº. del co-tutor:

José Ignacio Santos Martín José Manuel Galán Ordax

**Resumen**

Se quiere optimizar el proceso de reparto de material en una fábrica. El proyecto desarrolla una herramienta de optimización general que pueda ser utilizada en un mayor número de supuestos. El objetivo del trabajo es optimizar la entrega de los materiales que requieren un conjunto de máquinas y puestos de trabajo

**Descriptores**

Palabras separadas por comas que identifiquen el contenido del proyecto Ej: servidor web, buscador de vuelos, android

**Abstract**

A **brief** presentation of the topic addressed in the project.

**Keywords**

keywords separated by commas

**Índice General**

**Índice de Figuras**

**Índice de Tablas**

# Introducción

En todo proceso productivo es de vital importancia mantener el stock de materiales siempre actualizado. En el momento que se produce un desajuste en las cantidades o un cambio no contemplado en la ubicación de los materiales hace que la recogida y transporte de dichos materiales deje de ser óptima y provoca paradas innecesarias en el proceso productivo lo cual conlleva a pérdidas económicas.

Cuando se producen una serie de paradas continuadas a causa del servicio de transporte de material, no sólo causa pérdidas económicas como ya se ha citado, sino que dificulta el trabajo de las personas que se encargan del reparto de material.

En este caso se propone un proyecto de reestructuración de un algoritmo ya implantado en una fábrica, que a pesar de que funciona, no es para nada óptimo. Las carencias principales son:

* No muestra la cantidad de material a llevar a una máquina, sino que el operario que lo transporta tiene que calcularlo.
* No indica el lugar donde se encuentra el material que tiene que llevar ni la cantidad disponible.
* El listado de órdenes no está ordenado, sino que va añadiendo en primer lugar la última orden solicitada. Esto implica que las primeras órdenes que aparecen en pantalla no sean de mayor prioridad.

Por todo ello se va a reprogramar el algoritmo para solventar los problemas descritos disminuyendo así el número de paradas y por consiguiente la pérdida económica.

Otro factor importante, aunque a nivel empresarial no siempre está contemplado, es que se facilitaría en gran medida el trabajo del operario que transporta el material. En muchas ocasiones, hace que al no estar optimizado, el operario no gestione bien los viajes en el transporte (por falta de información) e implica que a veces no disponga del tiempo suficiente para hacer los descansos programados en la jornada laboral.

* 1. Estructura de la memoria
* **Introducción:** Descripción de la situación actual y del proyecto. Estructura de la memoria y estructura de los anexos.
* **Objetivos del proyecto:** Se describen los objetivos a alcanzar tanto generales como técnicos.
* **Conceptos teóricos:** Se definen una serie de conceptos que permiten una mejor comprensión del proyecto.
* **Técnicas y herramientas:** Detalle de las herramientas usadas y metodologías empleadas en la realización del proyecto.
* **Aspectos relevantes en el desarrollo:** Aspectos relevantes y por lo tanto importante destacar durante el proceso.
* **Trabajos relacionados:** Búsqueda y estudio de trabajos relacionados previamente realizados.
* **Conclusiones y líneas de trabajo futuras:** Conclusiones que se han obtenido al finalizar y posibles ideas para continuar en el futuro.
  1. Estructura de los anexos
* **Plan de proyecto software:** Planificación temporal y estudio económico.
* **Especificación de requisitos:** Detalle de los requisitos necesarios y establecidos para la elaboración del proyecto.
* **Especificación de diseño:** Diseño de la interfaz y datos empleados.
* **Manual del programador:** Describe la estructura del proyecto, cómo se ejecuta y cómo está organizado.
* **Manual de usuario:** Manual o guía de uso para el usuario final.
  1. Enlaces adicionales

Repositorio

Wiki

YouTube

# Objetivos del Proyecto

En este apartado se detallan los objetivos generales de este proyecto y los objetivos técnicos, que serán necesarios para obtener los objetivos generales.

1. Objetivos generales

Descripción de los objetivos generales definidos al comienzo del proyecto.

* Obtener la cantidad de material que hay que transportar a las máquinas que está produciendo cubiertas.
* En función del stock y de la capacidad de almacenaje de la máquina generar las órdenes.
* Hacer una ordenación de las órdenes generadas mostrando en primer lugar las más urgentes y por lo tanto las primeras a llevar.
* Mostrar toda la información necesaria para el operario que transporta.
* Generar un listado de la cantidad de materiales que el operario ha transportado en su turno.
* Crear un sencillo manual de usuario.

1. Objetivos técnicos

Aquí se detalla la implementación y las herramientas utilizadas.

* El proyecto se desarrolla mediante la metodología SCRUM.
* La base de datos se aloja en un servidor para el cual se ha creado una máquina virtual con un sistema operativo Windows Server 2019.
* Uso de GutHub para el control de versiones.
* Uso de ZenHub para la gestión de las tareas y los sprints.
* Uso de GitHub Desktop para sincronizar el repositorio local con el remoto.
* Se usa Python como lenguaje de programación. Tanto la ejecución del programa principal como de los scripts que se ejecutan en segundo plano para simular el proceso productivo. Se usarán otras librerías adicionales como sqlite3 y PyQt5.
* Uso de PyCharm para el desarrollo del código fuente.
* Uso de DB Browser for SQLite para la creación y gestión de la base de datos.
* Uso de QT Designer para la creación del entorno gráfico.
* Uso de Zotero para guardar todas las referencias bibliográficas.
* Uso de Oracle VM VirtualBox para crear la máquina virtual que alojará la base de datos.
* Uso de Microsoft Word para la elaboración de la memoria del proyecto y los anexos al mismo.

# Conceptos teóricos

A continuación se van a detallar una serie de conceptos teóricos necesarios para facilitar la compresión del proyecto.

1. Servicio

En el proceso productivo que estamos desarrollando, es necesario que las máquinas que fabrican cualquier tipo de componente dispongan del material necesario para ello.

El material disponible en cada máquina se va agotando a medida que se produce y no es el propio personal de la máquina quien va a buscar el material cuando se agota, sino que hay una persona encargada para cada material necesario.

Cada material que hay que transportar para la elaboración de una cubierta pertenece a un servicio. En este caso nos vamos a centrar en el servicio de talones que explicaremos más adelante.

1. Cubierta

Una cubierta o neumático es una pieza compuesta de caucho que se coloca en la rueda de un vehículo. Es el único punto de unión del vehículo con el suelo.

Para la fabricación de una cubierta se necesitan varios materiales que se unen y forman el neumático. De entre todos esos materiales que la componen, nos vamos a centrar en la necesidad de los talones.

1. Máquina de construcción

Una cubierta la componen distintos materiales, y la unión de dichos materiales se realiza en una máquina de construcción.

1. Talón

El talón es la parte interna de una cubierta. Es la parte encargada de ajustar el neumático a la llanta de forma segura y firme, por lo tanto una cubierta necesita 2 talones.

1. RTB

Máquina de transformación de materiales.

Una RTB, produce talones, para ello, une un aro rígido formado principalmente por alambre y una tira de caucho formando así un talón.

Cada RTB fabrica talones de una dimensión, es decir, de un tamaño de llanta (15, 16, 17, 18 y 19 pulgadas).

A cada aro de alambre se le pega una tira de goma que dependiendo del ancho de la tira conformará el perfil de la rueda.

# Técnicas y herramientas

Técnicas y herramientas usadas en el desarrollo del proyecto.

1. Python

La decisión de usar Python como lenguaje de programación para este proyecto es debido a su uso en los últimos años de la carrera y por lo tanto el conocimiento de este.

Permite un desarrollo orientado a objetos y es un lenguaje interpretado, por lo que no hace falta compilar para su ejecución.

**Python [Ref] es un lenguaje de programación** de código abierto, orientado a objetos, muy simple y fácil de entender. Tiene una sintaxis sencilla que cuenta con una vasta biblioteca de herramientas, que hacen de Python un lenguaje de programación único.

**Una de las ventajas principales de aprender Python es la posibilidad de crear un código con gran legibilidad, que ahorra tiempo y recursos, lo que facilita su comprensión e implementación.**

1. PyCharm

PyCharm es un entorno de desarrollo integrado usado específicamente para el desarrollo en Python.

En un principio se estuvo dudando entre este IDE y Spyder. Si los comparamos vemos que PyCharm es mucho más completo y dispone de numerosas herramientas y características que facilitan mucho el trabajo [Ref]

* Asistencia y análisis de codificación, resaltado de sintaxis y errores.
* Refactorización del código.
* Depurador de Python integrado

1. DB Browser

DB Browser [Ref 1] es una aplicación gratuita diseñada para facilitar la creación y administración de las bases de datos con SQLite, el cual es uno de los motores de bases de datos SQL gratuito de código abierto más conocidos.

Es apto para usuarios y desarrolladores [Ref 2] que desean crear, buscar y editar bases de datos con una interfaz muy intuitiva y sencilla similar a una hoja de cálculo sin necesidad de introducir comandos SQL.

Los controles y asistentes están disponibles para que los usuarios:

* Cree y compacte archivos de base de datos
* Crear, definir, modificar y eliminar tablas
* Crear, definir y eliminar índices
* Examinar, editar, agregar y eliminar registros
* Registros de búsqueda
* Importar y exportar registros como texto
* Importar y exportar tablas desde / hacia archivos CSV
* Importar y exportar bases de datos desde / hacia archivos de volcado SQL
* Emitir consultas SQL e inspeccionar los resultados
* Examinar un registro de todos los comandos SQL emitidos por la aplicación.
* Trazar gráficos simples basados en datos de consulta o tabla

1. GitHub

GitHub [Ref 1] es una plataforma de desarrollo colaborativo para alojar proyectos utilizando el sistema de control de versiones Git. Se utiliza principalmente para la creación de código fuente de programas de ordenador.

Git [Ref 2] es uno de estos sistemas de control, que permite comparar el código de un archivo **para ver las diferencias entre las versiones**, restaurar versiones antiguas si algo sale mal, y fusionar los cambios de distintas versiones. También permite trabajar con distintas ramas de un proyecto, como la de desarrollo para meter nuevas funciones al programa o la de producción para depurar los bugs.

Las principales características de la plataforma es que ofrece las mejores características de este tipo de servicios sin perder la simplicidad, y es **una de las más utilizadas del mundo** por los desarrolladores. Es multiplataforma, y tiene multitud de interfaces de usuario.

Así pues, Github es **un portal para gestionar las aplicaciones que utilizan el sistema Git**. Además de permitirte mirar el código y descargarte las diferentes versiones de una aplicación, la plataforma también hace las veces de red social conectando desarrolladores con usuarios para que estos puedan colaborar mejorando la aplicación.

Por todo lo ya descrito y por la familiaridad con la herramienta puesto que ha sido usada con anterioridad para otros proyectos en otras asignaturas del grado.

1. ZenHub

Es una extensión del navegador que se integra con GitHub.

Con esta herramienta controlamos de manera muy visual y sencilla las tareas y sprints del proyecto, haciendo que su gestión sea muy sencilla.

Otra razón para su uso es que dispone de herramientas para elaborar gráficos del progreso del desarrollo.

1. GitHub Desktop

Herramienta para clonar nuestro repositorio local con el repositorio remoto.

Es gratuita, sencilla y muy intuitiva. Cuando se produce un cambio en un archivo en nuestro repositorio local hace una comparación con lo que hay alojado en el repositorio remoto, si detecta algún cambio lo muestra y permite escribir un comentario y una descripción para hacer *commit*.

Finalmente se hace un *push* para alojarlo en el repositorio remoto. De esta forma podemos ver todos los cambios realizados.

1. Zotero

Zotero [Ref] es un programa de software libre para la gestión de referencias bibliográficas. **Zotero** es una extensión libre para el navegador**,** que permite a los usuarios recolectar, administrar y citar investigaciones de todo tipo. Importa datos directamente desde las páginas web visualizadas en el momento.

Al navegar, Zotero detecta automáticamente cuándo estamos consultando una fuente de información. En este momento, mediante un clic del ratón, guarda en un fichero local la información completa (la referencia) de aquello que estemos consultando.

Muy útil para guardar toda la información recopilada de los diferentes sitios de manera organizada.

1. Qt Designer

Qt Designer [Ref] es una herramienta para diseñar y construir interfaces gráficas de usuario, con QT Widgets. Puede componer y personalizar sus ventanas o cuadros de diálogo en una forma de lo que ve es lo que obtiene y probarlos utilizando diferentes estilos y resoluciones.

Los widgets y formularios creados con Qt Designer se integran a la perfección con el código programado, utilizando el mecanismo de ranuras y señales de Qt, para que pueda asignar fácilmente el comportamiento a los elementos gráficos.

Todas las propiedades establecidas en Qt Designer se pueden cambiar dinámicamente dentro del código. Además, las funciones como la promoción de widgets y los complementos personalizados le permiten usar sus propios componentes con Qt Designer.

1. Oracle VM VirtualBox

Virtual Box se usa para hacer máquinas virtuales con instalaciones de sistemas operativos. Se ha decidido su uso porque se conoce el funcionamiento de la herramienta y por disponer ya de ella.

# Aspectos relevantes del desarrollo del proyecto

Es este apartado se recogen los aspectos más relevantes del desarrollo del proyecto.

1. Inicio del proyecto

La idea del proyecto nació en la primera reunión. Aún no se tenía ninguna idea de qué tipo de proyecto se podría realizar como trabajo de fin de carrera. Mi principal problema fue la poca disponibilidad de tiempo para el desarrollo por mis horarios y turnos de trabajo.

Expliqué a mis tutores en qué consistía mi trabajo y la aplicación que uso para el mismo. La pregunta que me hicieron fue la clave: ¿El programa que usas está optimizado? Y la respuesta fue rotunda: no está para nada optimizado y provoca numerosas paradas de producción, problemas para el operario y por consecuencia pérdidas económicas.

En ese momento se decidió desarrollar una versión mucho más completa y óptima de lo que actualmente está implementado. Yo, como usuaria de la aplicación conozco las carencias que tiene y cuáles podrían ser las mejoras a incluir para mejorar el proceso productivo.

1. Metodologías

La metodología seguida ha sido SCRUM la cual ya había sido estudiada en la asignatura de gestión de proyectos.

Se basa en hacer un desarrollo de manera incremental del proyecto, de tal manera que en cada cierto período de tiempo establecido se realizan una serie de tareas.

El desarrollo se ha dividido en sprints de una duración determinada en los cuales se marcan una serie de tareas u objetivos a cumplir y qué tiempo es necesario para cumplir cada una de esas tareas.

En la estructura del desarrollo existen diferencias con la metodología SCRUM, principalmente porque el proyecto es de desarrollo individual y no existe un equipo de desarrollo, pero se ha intentado seguir las características de la metodología lo máximo posible:

* Se han mantenido una reunión al comienzo de cada sprint y se han establecido los objetivos para ese sprint y la duración de cada tarea.
* Se debería haber mantenido una reunión al finalizar cada sprint para comentar si se han cumplido los objetivos y si ha habido problemas, en este caso, en la reunión de comienzo de cada sprint se han tratado estos temas al inicio de la reunión.
* Se ha utilizado la herramienta ZenHub para planificar en el tablón las tareas a realizar. Es una herramienta de gran utilidad para cambiar los estados de las tareas. De esta forma, en el momento que una tarea es completada, se mueve dentro del tablero a la lista de tareas cerradas eliminando así tareas pendientes.

1. Formación

El lenguaje de programación utilizado ha sido Python puesto que es el lenguaje que más se ha utilizado en todas las asignaturas durante los últimos cursos, por ello, se tiene más reciente que otros y resulta más sencillo desarrollar el proyecto.

**SQLITE**

Es un sistema de gestión de base de datos de dominio público. Hasta este momento no se había utilizado este sistema de gestión a lo largo del grado, por lo que se ha tenido que aprender su funcionamiento y sus peculiaridades.

Lo más importante es que Python incluye soporte para SQLite incorporado es su biblioteca estándar. Otro detalle importante que ayuda mucho durante la programación es que en cada columna se puede insertar un tipo de dato que no corresponda con el tipo de columna, por ejemplo, podemos insertar un string a una columna de tipo entero.

**PYQT5**

Es un enlace de Python al framework de Qt para la realización de interfaces gráficas.

Al instalar PyQt5, se dispone de la herramienta QT Designer, que facilita mucho el diseño de las interfaces gráficas, pero es necesario aprender a utilizarlo y cómo convertir el fichero que genera (xml) a lenguaje Python.

1. Desarrollo del proyecto / algoritmo

En primer lugar se hizo un algoritmo en pseudocódigo en el que se establecía a grandes rasgos lo que el programa debería mostrar al operario y lo que el operario debería hacer con dicha información.

A continuación, se decidió el diseño, es decir, cómo el programa iba a mostrar esos datos. Se decidió que fuese mediante una tabla con todo lo necesario.

VENTANA ÓRDENES

Datos del operario y del servicio

Órdenes

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |

Una vez que se tuvo decidido el entorno de desarrollo del proyecto se comenzó por diseñar la interfaz gráfica. Se hizo una tabla provisional en la base de datos llamada ORDENES para recoger los datos y mostrarlo en la ventana, de esta forma se aprendió cómo insertar los datos en la tabla de tipo QTableWidget.

En ese momento, se detectó el problema de que no existe un proceso productivo que indique qué material hace falta, es decir, cómo se va gastando material y cómo se van produciendo las cubiertas. Se decidió simular un proceso productivo y crear dos scripts. Uno de ellos va incrementando en la base de datos las cubiertas realizadas y decrementando el número de talones y el otro simula la producción de talones y modificando la tabla de stock de material. Ambos scripts corren en segundo plano y no interfieren en las tareas del operario.

Cuando el proceso productivo está activo, la ventana órdenes muestra los datos al operario de manera muy detallada y con toda la información necesaria para su ejecución. Las órdenes estarán ordenadas en función de la urgencia y la prioridad de la máquina. Aquellas máquinas que esté más próximas a parar se mostrarán en los primeros lugares de la lista.

En este momento, el operario sólo tiene que cumplir las órdenes en el orden en el que se le muestran. Recoge el material de la ubicación que se le muestra, llevarlo a la máquina que lo necesita y pulsar el botón llevar material, de esta manera, se actualizan los huecos de la máquina de destino, los materiales disponibles y se actualizan las órdenes.

En condiciones normales, el operario recoge el material de las RTBS y lo lleva a la máquina de construcción que lo necesita, pero existen el problema de que en un momento dado no exista stock de ese material en las RTBS y la máquina de construcción vaya a parar por falta de ese material. Para solventar dicho problema, el programa proporciona una funcionalidad para buscar el material necesario en otras ubicaciones. En ocasiones, hay varias máquinas usando el mismo tipo de material, por lo que se puede quitar de una que tenga excedente y llevarlo a aquella que lo necesita con más premura para que no pare.

Es importante controlar varios aspectos en este caso. En el momento que se abre la ventana que muestra las ubicaciones donde hay material disponible, es necesario tener en cuenta no sólo la cantidad de carros que se puedan retirar de la máquina que tiene el excedente, sino de los carros que caben en la máquina a la que se llevan, por lo tanto, sólo permitirá retirar tantos carros como quepan en la máquina de destino. Pongamos un ejemplo: Tengo que llevar 2 carros a la máquina V2 y los voy a retirar de V3 que tiene 3 carros de ese material en sus huecos. Se mostrará un stock de ese material en V3 de 2 carros, pero en la lista desplegable sólo nos permitirá retirar un máximo de 2 carros.

En el momento que se retira el material de otra máquina se actualizan sus huecos dejándolos vacíos y llenando los de la máquina de destino.

En este momento no se actualizan las órdenes. Es una manera que tiene el operario de comprobar que ha hecho la entrega correcta. Es necesario que el operario pulse el botón actualizar para que se refresquen las órdenes que se muestran.

1. Resolución de problemas técnicos

El principal problema que se ha encontrado durante todo el proceso ha sido al final del proyecto.

Cuando ya se ha tenido un programa completamente funcional, se ha intentado hacer un ejecutable para el usuario final. Se ha intentado hacerlo con diferentes herramientas, tales como, *pyinstaller, cpy2exe y cx\_freeze*. A pesar de haber invertido mucho tiempo en investigación y en hacer pruebas no se ha conseguido.

Se comentó el problema durante una de las reuniones de sprints y se decidió no hacer un ejecutable, puesto que Python ejecuta el programa sin necesidad de generar un fichero con extensión .exe.

El siguiente problema surge cuando se lanza el programa. Al ejecutarse, se muestra la consola en pantalla y no es adecuado para el usuario final, por lo tanto después de investigar diferentes opciones, se cambió la extensión de los ficheros .*py* por *.pyw* e importar las librerías *win32gui, win32con* y ocultar la consola durante la ejecución.

1. Wiki

La wiki es un aspecto relevante de este proyecto. A pesar de haber invertido bastante tiempo en su realización, porque es necesario explicar paso a paso el funcionamiento de la aplicación, creo que es de gran importancia porque facilita enormemente la comprensión por parte del usuario además de enriquecer el proyecto al añadir una funcionalidad adicional.