

Memoria del Trabajo de Fin de Grado

Álvaro Ruifernández Palacios

18 de enero de 2018



UNIVERSIDAD DE BURGOS

Índice

1. Introducción	3
1.1. Parte software	3
1.2. Parte hardware	3
2. Anexo 1	4
2.1. Desarrollo de la memoria	4
2.2. Desarrollo del código	4
2.3. Planteamiento de las tareas	4
2.4. Metodología de gestión y herramientas asociadas	5
3. Diagrama de clases del sistema	6
4. Diagrama de secuencia	7

1. Introducción

En este proyecto se va a desarrollar un programa que permita detectar objetos en el rango de visión de un láser de seguridad. En un principio estas medidas serán tomadas con el Hokuyo Safety Laser Scanner. Este aparato es capaz de desarrollar tres áreas de detección dependiendo de la distancia a la que detecte un elemento (puede ser de 20, 10 y 5 metros). Aunque sea este láser con el que se va a realizar el proyecto, el principal objetivo es el de poder hacer que el programa pueda funcionar en diferentes tipos de láser con diferentes formas de obtención de datos.

1.1. Parte software

La explicación básica del funcionamiento de este proyecto es la utilización de tres elementos bien diferenciados:

- Láser: esta parte es la que se encarga de obtener los datos de su entorno. Estos datos serán enviados a través de una interfaz (en nuestro caso u cable ethernet) hacia la unidad de tratamiento de datos.
- Unidad de tratamiento: esta es la parte más importante del proyecto, esto es debido a que es la parte que gestiona las inmensas cantidades de datos recibidas del láser, procesarlos y manejarlos. La otra parte por la que esta parte es esencial es la de que también se comunica con la parte encargada de establecer las áreas en las que el programa debe de comprobar si hay un objeto con los datos recibidos del láser en un momento determinado. Esta es la parte con la que va a interactuar el sujeto que utilice la aplicación.
- Gestor de áreas: Es otra de las partes del proyecto con la que le usuario va a actuar sobre el programa ya que es aquella en la que va a definir las áreas donde quiere saber si existe un objeto o no.

Este proyecto se va a desarrollar en Un sistema operativo en base Linux, concretamente en un sistema Ubuntu en su version 16.04. Esto es debido a que ciertos componentes no son compatibles con sistemas Windows.

1.2. Parte hardware

A demás de la creación de este sistema software formado por las tres partes anteriormente definidas, es necesario destacar la creación de un sistema hardware para implementar estos comportamientos. El sistema físico se puede definir en varias partes bien diferenciadas:

- Láser: encargado de recibir las ordenes de detección a través de su interfaz de comunicación con el PCB para que este procese la información de sus lecturas.
- PCB: encargado de guardar y hacer operativo todo el sistema software. Es el que controla la puesta en marcha del sistema, encargándose de recibir tanto las áreas en las que se desea realizar el análisis de detección de objetos como la de enviar esas órdenes de detección hacia el para luego analizar la información de las lecturas de este aparato y poder hacer saber al usuario si el laser ha detectado un objeto o no en la áreas que previamente se le deben de haber pasado.
- Interfaz de áreas: en este elemento es en el que el usuario introduce dentro del sistema las áreas en las que desea detectar la existencia de objetos dentro del espectro de visión del láser empelado en el sistema.

2. Anexo 1

En este anexo se presenta las aplicaciones que se van a emplear para el desarrollo de diferentes aspectos des este proyecto, explicando las diferentes razones por las cuales se ha decido emplear cada una de ellas.

2.1. Desarrollo de la memoria

En este aspecto se ha querido emplear LaTeX en contraposición a otros programas de propósito similar como pueden ser Microsoft Word o Open Office Writer ya que esta aplicación nos permite una mayor cantidad de posible acciones a realizar en el documento a realizar.

Por ejemplo, a presentación del documento puede quedar más limpia con esta aplicación ya que el usuario puede manejar el espacio de cada hoja a su antojo y poder disponer de todo él ya que se puede variar los márgenes de las páginas de una forma sencilla (introducción de un comando), lo que en los otros programas antes mencionados resultaría demasiado complicado a demás de ser potencialmente peligroso debido a que, gracias a las acciones automáticas que poseen, podrían desajustar todo el contenido de la hoja en sí haciendo que el usuario tenga que ajustar cada uno de los elementos que antes poseía. LaTeX permite saltarse los márgenes establecidos, por ejemplo, para insertar una imagen. en cuanto a los encabezados, pies de página y numeración de página también se puede realizar con un simple comando.

Con todo esto y con la característica de que permite guardar el archivo directamente en PDF sin realizar ninguna conversión (aunque para ello necesite crear algún archivo debido a la compilación) es por la que se ha escogido para este

2.2. Desarrollo del código

Debido a que es un programa familiar debido a su utilización en diferentes asignaturas a lo largo del grado, se ha escogido Eclipse para el desarrollo de esta parte del proyecto. Para poder emplearlo de forma dirigida a este proyecto se ha necesitado integrar este programa con el entorno de desarrollo conocido como MTX-GTW. Este entorno es el que permitirá la extracción de medidas necesarias para saber la distancia a la que se encuentra un objeto del láser si es que se detecta un objeto. Otra característica de este entorno es la compatibilidad con el sistema operativo, ya que solo es compatible con sistemas Ubuntu y demás distribuciones de Linux.

Como ya se ha dicho en la introducción, para el desarrollo del código se necesita tener la máquina virtual del sistema operativo Ubuntu 16.04. Este sistema se ha escogido, al igual que Eclipse, debido a que se han empleado en otras asignaturas del grado con lo que resulta familiar para poder trabajar de manera más cómoda y eficiente que si se debiera aprender su modo de empleo. A demás de estas características, se ha escogido por el tema de la compatibilidad explicado anteriormente.

2.3. Planteamiento de las tareas

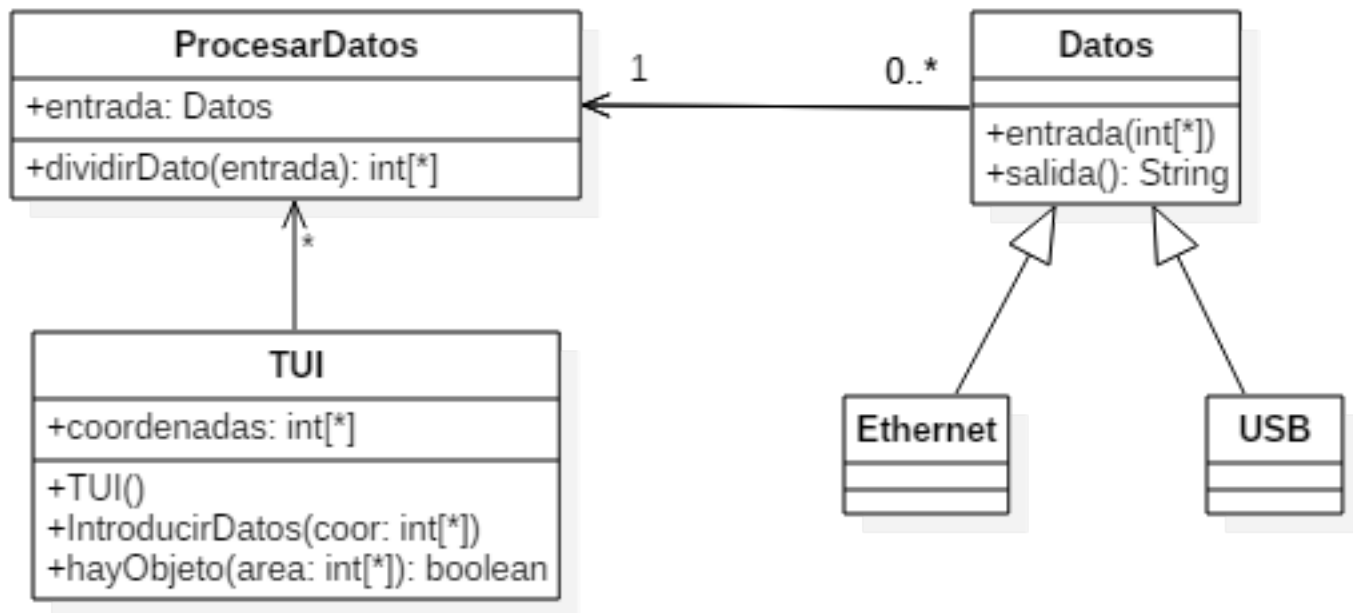
Para organizar las actividades a realizar se ha escogido la aplicación Trello. Esta aplicación se ha escogido debido a su facilidad de uso (debido a que su interfaz de usuario es muy intuitiva), a demás de otros muchos aspectos como el hecho de que a un tablero (unidades organizativas en las que se gestionan las tareas de los diferentes proyectos que se puede gestionar desde una misma cuenta de usuario) se puede acceder más de un usuario. Esta característica permite que tanto el alumno como el profesor pueden acceder al mismo tablero para gestionar las tareas a realizar.

2.4. Metodología de gestión y herramientas asociadas

Para este proyecto se ha decidido utilizar la metodología de SCRUM. Esta metodología esta basada en entregas incrementales pero funcionales. Para la realización de esta metodología, perteneciente a las denominadas metodologías ágiles, se va a utilizar la aplicación ZenHub. Esta aplicación se puede emplear para presentar los sprints y se puede planificar la fecha de comienzo y final cada una de las tareas.

3. Diagrama de clases del sistema

El diagrama de clases que definen las funciones y relaciones entre objetos que conforman el sistema seria el siguiente:



4. Diagrama de secuencia

El diagrama de secuencia que definen mensajes y funciones a utilizar en el empleo de la aplicación seria el siguiente:

