TRABAJO FIN DE GRADO

Grado en Ingeniería Informática

TFG - DASIoT

DASIoT: Desarrollo y Auditoría de Seguridad para prototipo de dispositivos IoT

Autor
Luis Aróstegui Ruiz

Directores
José Manuel Soto Hidalgo
Alberto Guillén Perales

Escuela Técnica Superior de Ingenierías Informática y de Telecomunicación

Granada, 11 de abril de 2022

TFG - DASIoT: Desarrollo y Auditoría de Seguridad para prototipo de dispositivos IoT

Luis Aróstegui Ruiz

Palabras clave: Internet de las Cosas, Seguridad

Resumen

En el contexto de IoT, hay un ecosistema de entornos de desarrollo específicos. En este TFG se hará uso de alguno de ellos para llevar a cabo la implementación de un dispositivo IoT y analizar los potenciales problemas de seguridad que pueden aparecer durante la etapa de implementación.

Project Title: Project Subtitle

First name, Family name (student)

Keywords: Keyword1, Keyword2, Keyword3, ...

Abstract

Write here the abstract in English.

Yo, **Nombre Apellido1 Apellido2**, alumno de la titulación TITULACIÓN de la **Escuela Técnica Superior de Ingenierías Informática y de Telecomunicación de la Universidad de Granada**, con DNI XXXXXXXXX, 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

Índice de figuras

Índice de tablas

CAPÍTULO 1

Introducción

1.1. Descripción y contexto

El Internet de las Cosas, o IoT, es un sistema de dispositivos informáticos, máquinas mecánicas y digitales, objetos, animales o personas interrelacionados que cuentan con identificadores únicos (UID) y la capacidad de transferir datos a través de una red sin que sea necesaria la interacción entre personas o entre ordenadores. [?]

Una "cosa" en el Internet de las Cosas puede ser una persona con un implante de monitor cardíaco, un animal de granja con un chip, un automóvil que tiene sensores incorporados para alertar al conductor cuando la presión de los neumáticos es baja o cualquier otro objeto natural o artificial al que se le pueda asignar una dirección de Protocolo de Internet (IP) y que sea capaz de transferir datos a través de una red. Cada vez más, las organizaciones de diversos sectores utilizan el IoT para operar de forma más eficiente, comprender mejor a los clientes para ofrecerles un mejor servicio, mejorar la toma de decisiones y aumentar el valor del negocio.

El concepto de *Internet of Things* fue propuesto en 1999 por el laboratorio de identificación automática del Instituto Tecnológico de Massachusetts (MIT). La UIT lo dio a conocer en 2005, empezando por China. El IoT puede definirse como *"datos y dispositivos continuamente disponibles a través de Internet"*. La interconexión de "cosas" (objetos) que pueden dirigirse de forma inequívoca y las redes heterogéneas constituyen el IoT. La identificación por radiofrecuencia (RFID), los sensores, las tecnologías inteligentes y las nanotecnologías son los principales contribuyentes a al IoT para una variedad de servicios. Con la drástica

reducción del coste de sensores y con la evolución de tecnologías como el ancho de banda, el procesamiento, los teléfonos inteligentes, la migración hacia el IPv6 y el 5G se está facilitando la adopción del IoT.

El IoT también ve todo como lo mismo, sin discriminar entre humanos y máquinas. Las "cosas" incluyen a los usuarios finales, los centros de datos (DC), las unidades de procesamiento, los teléfonos inteligentes, las tabletas, el Bluetooth, el ZigBee, la Asociación de Datos por Infrarrojos (IrDA), la banda ultraancha (UWB), las redes celulares, las redes Wi-Fi, los DC de comunicación de campo cercano (NFC), la RFID y sus etiquetas, los sensores y los chips, los equipos domésticos, los relojes de pulsera, los vehículos y las puertas de las casas. [?]

1.2. Motivación

Las personas de todo el mundo estan ya preparadas para disfrutar de las ventajas del Internet de las cosas (IoT). El IoT lo incorpora todo, desde el sensor corporal hasta la computación en la nube. Comprende los principales tipos de redes, como la distribuida, la de red, la ubicua y la vehicular, que han conquistado el mundo de la informática durante una década. Desde de estacionamiento de vehículos a su seguimiento, de la introducción de datos de pacientes a la observación de los pacientes a la observación de los pacientes, de la atención a los niños a la atención a los ancianos, de las tarjetas inteligentes a las tarjetas de campo cercano, los sensores están haciendo sentir su presencia. Los sensores desempeñan un papel fundamental en el IoT.

El IoT funciona en redes y estándares heterogéneos. Excepcionalmente, ninguna red está libre de amenazas y vulnerabilidades de seguridad. Cada una de las capas del IoT está expuesta a diferentes tipos de amenazas. Este proyecto se centra en las posibles amenazas que hay que abordar y mitigar para conseguir una comunicación segura en el IoT. [?]

En otras palabras, el IoT combina "lo real y lo virtual" en cualquier lugar y en cualquier momento, atrayendo la atención tanto de desarrolladores como de ciberdelicuentes. Inevitablemente, dejar los dispositivos sin intervención humana durante un largo periodo podría dar lugar a robos. La seguridad ha sido finalmente reconocida como un requisito esencial para todo tipo de sistemas informáticos, incluidos los de IoT. Sin embargo, muchos sistemas IoT son mucho menos seguros que los típicos sistemas Windows/Mac/Linux.

Los problemas de seguridad de IoT se derivan de seguridad de la IO provienen de una serie de causas: características de seguridad inadecuadas en el hardware, software mal diseñado con una serie de vulnerabilidades, contraseñas por defecto

y otros errores de de seguridad. Los nodos IoT inseguros crean problemas para la seguridad de todo el sistema IoT. Dado que los nodos suelen tener una vida útil de varios años, la gran base instalada de de dispositivos inseguros creará problemas de seguridad durante algún tiempo. Los nodos IoT inseguros son ideales para los ataques de denegación de servicio. El ataque Dyn es un ejemplo de ataque basado en el IoT contra la infraestructura tradicional de Internet. Los sistemas IoT inseguros también causan problemas de seguridad al resto de Internet.

La privacidad está relacionada con la seguridad, pero requiere medidas específicas a nivel de aplicación la red y los dispositivos. No sólo hay que proteger los datos de los usuarios contra el robo, sino que la red debe diseñarse de forma que los datos menos privados no puedan utilizarse fácilmente para inferir datos más privados. [?]

1.3. Objetivos

El principal objetivo de este proyecto es hacer uso de un entorno de desarrollo específico para llevar a cabo una implementación de un dispositivo IoT y analizar los potenciales problemas de seguridad que pueden aparecer durante la etapa de implementación.

1.3.1. Objetivos específicos

- **OE.1:** Estudiar los distintos entornos de desarrollo para IoT y analizar funcionalidades y propiedades que se ajusten al proyecto.
- **OE.2:** Desarrollar un prototipo de aplicación para IoT utilizando un framework de desarrollo.
- **OE.3:** Explotación de una vulnerabilidad a nivel de dispositivo, protocolo, SO, aplicación o HW.

CAPÍTULO 2

Estado del arte

Antes de empezar a desarrollar los objetivos del proyecto es necesario conocer primero los fundamentos del Internet de las Cosas, esto engloba desde su funcionamiento, tipos de arquitecturas hasta los distintos estándares que existen.

2.1. Internet de las Cosas

El término "Internet de las Cosas" (IoT) se conoce desde hace unos años. En los últimos tiempos, está recibiendo más atención debido al avance de la tecnología inalámbrica. La idea básica se debe a la variedad de objetos, como RFID, NFC, sensores, actuadores, teléfonos móviles, que pueden interactuar entre sí teniendo una dirección distinta. El IoT permite a los objetos ver, oír, pensar y realizar trabajos haciendo que 'hablen' entre sí, para compartir y sincronizar información. El IoT transforma estos objetos de convencionales a inteligentes mediante la manipulación de sus tecnologías subyacentes, como los dispositivos integrados, las tecnologías de comunicación, las redes de sensores, los protocolos y las aplicaciones. [?]

La premisa básica y el objetivo de IoT es "conectar lo que no está conectado". Esto significa que los objetos que no están actualmente unidos a una red informática, es decir, a Internet, se conectarán para que puedan comunicarse e interactuar con personas y otros objetos. IoT es una transición tecnológica en la que los dispositivos nos permitirán sentir y controlar el mundo físico haciendo que los objetos sean más inteligentes y conectándolos a través de una red inteligente. Cuando los objetos y las máquinas pueden ser detectados y controlados a distancia a través de una red, se consigue una mayor integración entre el mundo físico y los ordenadores. Esto permite mejoras en las áreas de eficiencia, preci-

sión, automatización y habilitación de aplicaciones avanzadas.

El mundo del IoT es amplio y puede resultar algo complicado al principio debido a la abundancia de componentes y protocolos que engloba. En lugar de considerar IoT como un único tecnología, es bueno verlo como un paraguas de varios conceptos, protocolos y tecnologías, todos ellos dependientes a veces de un sector concreto. Aunque la amplia gama de elementos de IoT está diseñado para crear numerosos beneficios en las áreas de productividad y automatización, al mismo al mismo tiempo, introduce nuevos retos, como la ampliación del gran número de dispositivos y de la cantidad de datos que deben procesarse. [?]

2.1.1. Arquitecturas

En esta sección se muestran distintas arquitecturas para el IoT, que suele describirse como un proceso de cuatro etapas en el que los datos fluyen desde los sensores conectados a las çosas.ª través de una red y, finalmente, a un centro de datos corporativo o a la nube para su procesamiento, análisis y almacenamiento.

2.1.1.1. Arquitectura de tres capas

Normalmente, la arquitectura de IoT se divide en tres capas básicas: capa de aplicación, capa de red y capa de percepción, que se describen con más detalle a continuación.

- **Capa de percepción.** También conocida como capa de sensores, se implementa como la capa inferior en la arquitectura de IoT. La capa de percepción interactúa con los dispositivos y componentes físicos a través de dispositivos inteligentes (RFID, sensores, actuadores, etc.). Sus principales objetivos son conectar las cosas a la red IoT, y medir, recoger y procesar la información de estado asociada a estas cosas a través de los dispositivos inteligentes desplegados, transmitiendo la información procesada a la capa superior a través de las interfaces de la capa.

- **Capa de red.** También es conocida como la capa de transmisión, se implementa como la capa intermedia en la arquitectura de IoT. La capa de red se utiliza para recibir la información procesada proporcionada por la capa de percepción y determinar las rutas para transmitir los datos y la información al centro del IoT, los dispositivos y las aplicaciones a través de redes integradas. La capa de red es la más importante en la arquitectura de IoT, ya que varios dispositivos (hub, switching, gateway, cloud computing perform, etc.), y varias tecnologías de comunicación (Bluetooth o Wi-Fi) se integran en esta capa. La capa de red debe transmitir datos hacia o desde diferentes cosas o aplicaciones, a través de interfaces o pasarelas entre redes heterogéneas, y utilizando diversas tecnologías y protocolos de comunicación.

- **Capa de aplicación.** También conocida como la capa de negocio, se implementa como la capa superior en la arquitectura de IoT. La capa de aplicación recibe los datos transmitidos desde la capa de red y utiliza los datos para proporcionar los servicios u operaciones requeridos. Por ejemplo, la capa de aplicación puede proporcionar el servicio de almacenamiento para respaldar los datos recibidos en una base de datos, o proporcionar el servicio de análisis para evaluar los datos recibidos para predecir el estado futuro de los dispositivos físicos. Existen varias aplicaciones en esta capa, cada una con requisitos diferentes. Algunos ejemplos son la red inteligente, el transporte inteligente, ciudades inteligentes, etc.

La arquitectura de tres capas es básica para el IoT y se ha diseñado y realizado en varios sistemas. Sin embargo, a pesar de la simplicidad de la arquitectura multicapa de IoT, las funciones y operaciones en las capas de red y aplicación son diversas y complejas. Por ejemplo, la capa de red no sólo necesita determinar rutas y transmitir datos, sino también proporcionar servicios de datos como agregación de datos, computación, etc. La capa de aplicación no sólo necesita proporcionar servicios a los clientes y dispositivos, sino que también debe proporcionar servicios de datos tales como minería de datos, análisis de datos, por ejemplo. Por lo tanto, para establecer una arquitectura multicapa genérica y flexible para la IoT, debe desarrollarse una capa de servicio entre la capa de red y la capa de aplicación para proporcionar los servicios de datos. Basándose en este concepto, recientemente se han desarrollado arquitecturas orientadas a los servicios (SoAs).

2.1.1.2. Arquitectura basada en SoA

En general, SoA (Arquitectura Orientada a Servicios) es un modelo basado en componentes, que puede ser diseñado para conectar diferentes unidades funcionales, también conocidas como servicios, de una aplicación a través de interfaces y protocolos. SoA puede centrarse en el diseño del flujo de trabajo de los servicios coordinados, y permitir la reutilización de los componentes de software y hardware, mejorando la viabilidad de SoA para su uso en el diseño de la arquitectura IoT. Así, SoA puede integrarse fácilmente en la arquitectura de IoT, en la que servicios de datos proporcionados por la capa de red y la capa de aplicación en la arquitectura tradicional de tres capas pueden ser de la arquitectura tradicional de tres capas pueden extraerse y formar una nueva capa, la capa de servicios, también conocida como capa de interfaz o capa de middleware. Así, en una arquitectura arquitectura de IoT basada en SoA, existen cuatro capas que interactúan entre sí, siendo éstas la capa de percepción, la capa de red red, la capa de servicio y la capa de aplicación.

En la arquitectura de IoT basada en cuatro capas, la capa de percepción desempeña la función de la capa inferior de la arquitectura, y se utiliza para medir,

recoger y extraer los datos asociados a los dispositivos físicos. La capa de red se utiliza para determinar las rutas y proporcionar soporte de transmisión de datos a través de redes heterogéneas integradas. La capa de servicio se sitúa entre la capa de red y la capa de aplicación, proporcionando servicios para apoyar la capa de aplicación. La capa de servicio consiste en el descubrimiento de servicios, la composición de servicios, la gestión de servicios y las interfaces de servicios. Aquí, el descubrimiento de servicios se utiliza para descubrir las solicitudes de servicio deseadas, la composición de servicios se utiliza para interactuar con los objetos conectados, y dividir o integrar los servicios para satisfacer las solicitudes de servicio de una manera eficiente, la gestión de servicios se utiliza para gestionar y determinar los mecanismos de confianza para satisfacer las solicitudes de servicio, y las interfaces de servicio se utilizan para soportar las interacciones entre todos los servicios proporcionados. La capa de aplicación se utiliza para dar soporte a las solicitudes de servicio de los usuarios. La capa de aplicación puede soportar una serie de aplicaciones, como la red inteligente, el transporte inteligente, las ciudades inteligentes, etc. [?]

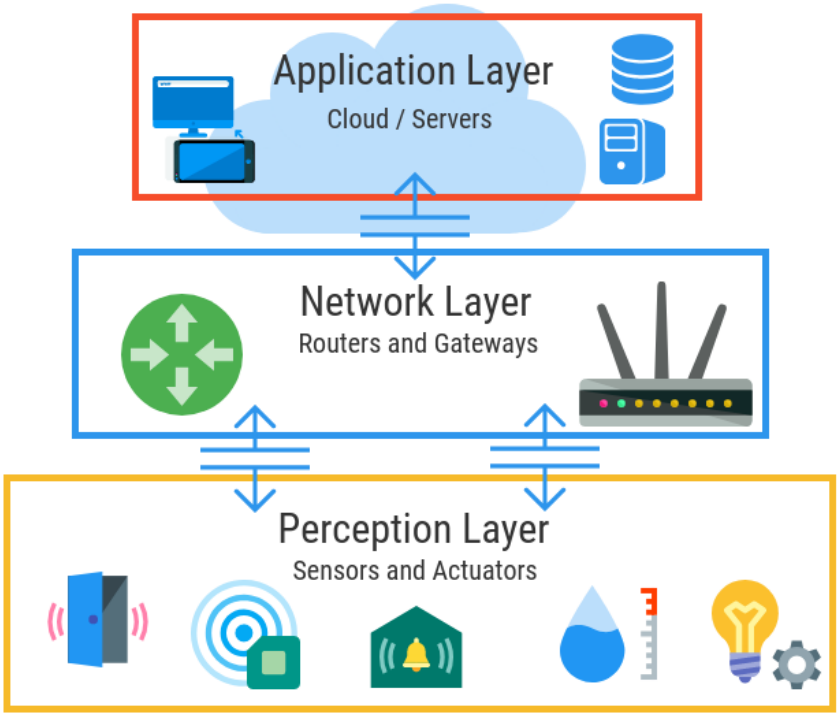

Figura 2.1: Arquitectura de 3 capas. [?]

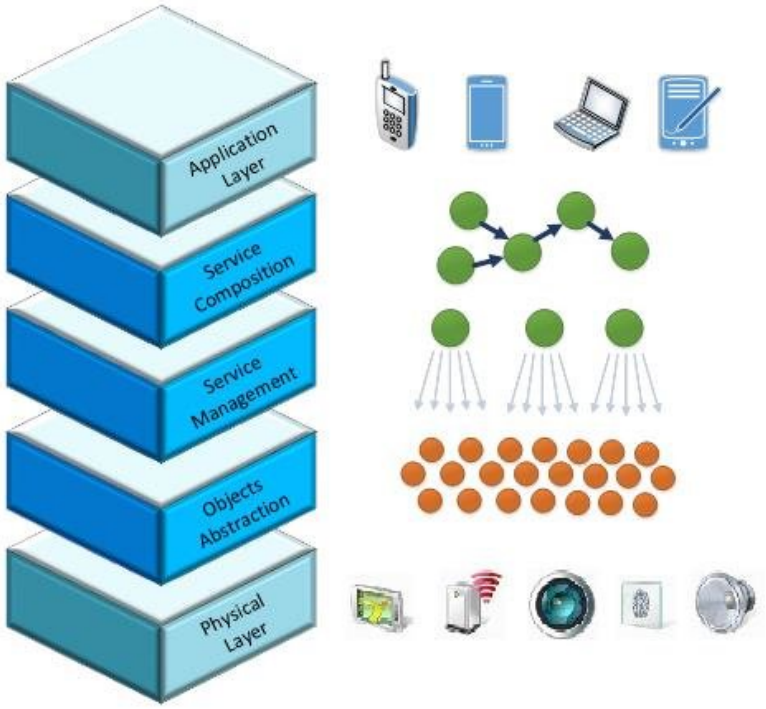

Figura 2.2: Arquitectura de 4 capas. [?]

2.1.2. Tecnologías asociadas

Teniendo en cuenta las arquitecturas mencionadas anteriormente, el IoT cuenta con varias tecnologías que hacen posible la cumplir el objetivo de cada capa de la arquitectura. A continuación se comentan sobre las tecnologías que nos encontramos en la Arquitectura basada en SoA. [?]

2.1.2.1. Capa de percepción

En la capa de percepción, la función principal es identificar y rastrear objetos. Para lograr esta función, se pueden implementar las siguientes tecnologías pueden ser implementadas.

- **Radio frequency identification (RFID)**. El sistema RFID se compone de uno o varios lectores y varias etiquetas RFID. Utiliza campos electromagnéticos de radiofrecuencia para enviar los datos adjuntos. Las etiquetas que se adjuntan a ella, almacenan datos electrónicamente que pueden ser leídos por RFID cuando se encuentra en la proximidad del lector.
- **Redes de sensores inalámbricos (WSN)**

[?]

2.1.3. Protocolos

[?]

2.1.4. Retos

2.2. Arquitecturas middleware IoT

2.3. Seguridad

2.3.1. Aspectos legales y éticos

2.3.2. Privacidad

2.3.3. Auditoría de seguridad

2.3.4. Seguridad de datos

CAPÍTULO 3

Planificación

3.1. Planificación a priori

Se trata de planificar como se espera desarrollar el proyecto en el tiempo, para ello se va a hacer uso de un diagrama de Gantt donde se va a proporcionar una vista general de las tareas programadas, estas tareas tendrán que completarse en unas fechas estipuladas.

3.1.1. Diagrama de Gantt

Un diagrama de Gantt es una herramienta útil para planificar proyectos. Proporciona una vista general de las tareas programadas, indicando el periodo de tiempo que tienen para completarse.

El diagrama se mostrará:

- Fecha de inicio y finalización del proyecto.
- Las tareas del proyecto.
- Fecha de programada de cada tarea, tanto la de inicio como la de final.
- Como se superponen las tareas y si hay relación entre ellas.

Con esta planificación se conseguirá una mayor claridad en las tareas a realizar y una mejor gestión del tiempo.

3.1.2. Etapas de desarrollo

- **1ª etapa**: Revisar entornos de desarrollo para IoT.

- **2ª etapa**: Desarrollar aplicación para IoT.
- **3ª etapa**: Explotación de una vulnerabilidad.
- **4ª etapa**: Documentación.

3.1.3. Temporización

De los 3 meses y medios a que se van a dedicar al proyecto, en todos ellos se va a desarrollar las etapas indicadas anteriormente. Se muestra la fecha de inicio y de fin de cada etapa.

Etapas del desarrollo	Fecha de comienzo	Fecha de finalización
Documentación del proyecto	9 de Marzo	23 de Junio
Revisar entornos de desaroollo IoT	21 de Marzo	6 de Abril
Desarrollar aplicación para IoT	7 de Abril	12 de Mayo
Explotar vulnerabilidad	13 de Mayo	22 de Junio

Tabla 3.1: Tabla con la organización temporal del proyecto.

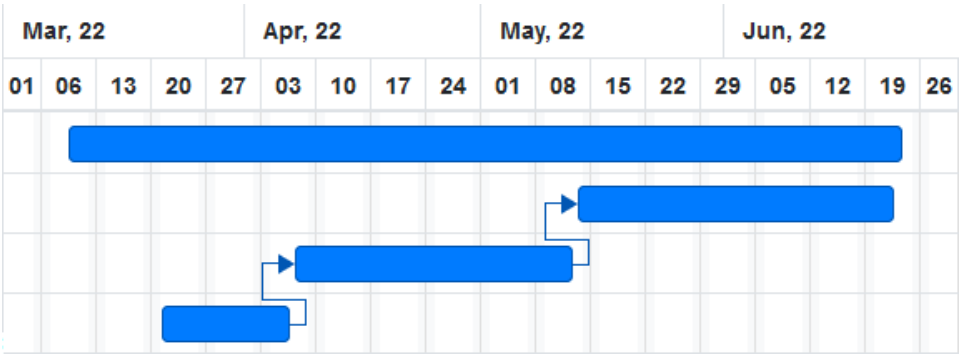

Figura 3.1: Diagrama de Gantt. Gráfico.

Este gráfico corresponde con la duración de cada etapa, comenzando con la documentación. También se indica que hay etapas que para que estas se puedan empezar a desarrollar necesitan que la anterior este terminada, es el caso de la etapa de *Desarrollo de la aplicación* y de *Explotación de una vulnerabilidad*.

3.1.4. Seguimiento del desarrollo

Con el fin de facilitar la visualización del progreso del proyecto se usan distintas herramientas.

3.1.4.1. Trello

Permite gestionar proyectos de forma sencilla y muy visual. Por cada proyecto se crea un tablero donde podemos crear listas y cada lista almacenará *tarjetas* donde se describirá una tarea u objetivo a cumplir. En nuestro caso el tablero se divide en 7 listas:

- **Dudas**, se dejan ancladas preguntas a los tutores.
- **Objetivos**, para recordar los objetivos que se tienen que completar para el proyecto.
- **Pendientes**, tareas que se tienen que realizar pero no se están desarrollando.
- **En proceso**, tareas que se están trabajando.
- **Pendientes de revisión**, tareas finalizadas que requieren de revisión por parte de los tutores.
- **Terminadas**, tareas que han sido aceptadas en la revisión.
- **Hitos**, se agrupan tareas que forman parte de una misma etapa.

3.1.4.2. Github

Se ha creado un repositorio [1] donde se van añadiendo *issues* por cada tarea que se tenga que realizar. Esta tarea es la misma que nos encontramos en el tablero de Trello, por tanto, cuando se crea una nueva tarea en Trello, creamos un nuevo issue con el mismo nombre y descripción para seguir un progreso coherente.

También se crean *Milestones* que se corresponde con el nombre y descripción de la lista de *Hitos* del tablero de Trello.

Cada vez que terminemos de trabajar en un *Milestone* se creará un *Pull Request* para que los tutores puedan revisar los cambios del proyecto y aceptarlos o rechazarlos. Cada acción que se realice tanto en Trello como en Github se verá reflejado en ambos, de esta manera conseguimos un desarrollo del proyecto realista de principio a final.

3.1.4.3. Clockify

Esta herramienta nos permite seguir el tiempo que estamos trabajando. Cada vez que nos pongamos a trabajar, iniciaremos el contador y nombraremos a ese contador con el nombre de la tarea que estemos realizando en ese momento.

[1] Enlace al repositorio: https://github.com/LuisArostegui/TFG

CAPÍTULO 4

Análisis del problema

CAPÍTULO 5

Diseño

CAPÍTULO 6

Implementación

CAPÍTULO 7

Pruebas

CAPÍTULO 8

Conclusiones y trabajos futuros