

**Escola Tècnica Superior d’Enginyeria  
Electrònica i Informàtica La Salle**

Trabajo Fin de Máster

Máster en Programción Web de Alto Rendimiento

Alumno Profesor Ponente

Daniel Salgado Población Víctor Caballero Codina

**ACTA DEL EXAMEN**

**DEL TRABAJO FIN DE MÁSTER**

Reunido el Tribunal calificador en la fecha indicada, el alumno

D. Daniel Salgado Población

expuso su Trabajo Fin de Máster, titulado:

Social Access Controller

Acabada la exposición y contestadas por parte del alumno las objeciones formuladas por los Sres. miembros del tribunal, éste valoró dicho Trabajo con la calificación de

Barcelona,

VOCAL DEL TRIBUNAL VOCAL DEL TRIBUNAL

PRESIDENTE DEL TRIBUNAL

Social Access Controller

Universitat Ramon Llull – La Salle Barcelona

Daniel Salgado Población

1 de septiembre de 2019

# Glosario

* **Web Thing** (*WT)*. Objeto con conexión a internet que ofrece información interna vía http(s) y arquitectura REST. En el presente proyecto *WT* devuelven formato de datos JSON. Posee dos zonas:
  + Zona Pública
    - Nombre
    - Brand
  + Zona Privada
    - Action
    - Property
* **Property**: Almacena un estado de un *WT*.
* **Action**: Es una iteracción con un *WT* que permite invocar una función en un *WT*. Una acción permite ver el estado de una *Property*.
* **Iot\_emulator**: recurso web donde configurar emulaciones de conjuntos de *Iots*.
* **Owner**: persona que posee *Iots*, conoce las credenciales para acceder a zona privada de *Iot*. Ademas posee cuenta de facebook y red de amigos dentro de esta red social
* **Friend**: persona conectada como amigo en facebook del owner.
* **Social Access Controller** (*SAC)*. Aplicación acoplada a facebook donde un *Owner* puede compartir cierta *Action* con un *Friend*.

# Resumen

Internet of Things (Iot) es uno de los anticipos tecnológicos más destacados en los últimos tiempos. Cada vez está más presente en el día a día y su aplicabilidad en el futuro está fuera de cualquier discusión. Este nuevo marco anticipa varios retos, entre los que se suele destacar la accesibilidad y la manera de compartir (o compartición) de estos entre personas de confianza.

Por no disponer de *Iot* y para abordar la accesibilidad de los *WT*  hemos creado un **emulador de Iot** que nos proporciona colecciones de *WTs.*

Para abordar la compartición de los *Iots* hemos creado el **Social Access Controller** hemos implementado un sistema para compartir *Wts*  WEB THING esta seguridad y delegarla en redes ya construidas para esto, como puede ser Facebook. Es esta aplicacbilidad la que hemos querido implementar con el Social access controller (SAC). Donde poder compartir con otras personas informacion de internet of things (Iot) propiedad de 1 dueño. Usando facebook para saber la red de amigos conectadas con un *Owner*. El *Owner* puede decidir cuales *Actions* muestra a cada *Friend*. SAC almacena la mínima informacion posible en su sistema, y consulta en tiempo real lo que el *Owner* o *Friend* están demandando.

El grueso de la información se encuentra en Facebook y en los propios Iots. SAC es un intermediario entre la red de amigos y las credenciales en os Iots.

# Abstract

# Índice General

# Introducción

## Motivación

Actualmente los *Owners* de *WT* no poseen una manera segura y homogénea de compartir *Actions* con sus *Friends*. El objetivo es proporcionar un sistema seguro donde compartir *Actions* de *WT.*

## Problemas a compartición de *WT* a resolver

Para poder compartir *WT* el *Owner* tiene estas soluciones pero tienen varios problemas

### *WT* multiusuario

Usar *WT* multiusuarios y condecer permiso a cada *Friend* para dicho *WT*.

Problemas:

* *Owner* debe conocer la manera de dar de alta un usuario nuevo a su *WT*.
* Todos los *WT* deben ser multiusuario
* Permiso a nivel *WT*

### Compartir credenciales

Compartir las credenciales con *Friend*.

Problemas:

* *Owner* debe confiar en la buena fe del *Friend* ya que pierde el control de la credencial
* *Owner* a llevar de alguna manera el control de a quién dejó cual WT
* Permiso a nivel de *WT*

## Requisitos

Para resolver las anteriores problemáticas proponemos diseñar una aplicación. que con estos requisitos*:*

* Un *Owner* tiene un lugar único donde administrar la compartición de *WTs* poseídas.
* Un *Owner* puede compartir indistintamente distintos *Actions* de un mismo *WT*
* Las *Actions* compartidas puedes serlo a número indeterminado de *Friends*
* La aplicación almacena la cantidad mínima e imprescindible de información
* La aplicación se loga al *WT* con las credenciales de *Owner*
* La red de amigos es almacenada externamente a la aplicación

## Objetivos

El objetivo es crear una aplicación que cumpla los requistios anteriormente mencionados.

### Social Access Controller

Siguiendo la idea de un Social Access Controller (SAC) propuesta por Dominique Guirnard *et al* en [LINK1] y la idea de la Compartición basada en Redes Sociales

El *Owner* accederá al SAC por primera vez usando Facebook. SAC almacena el user\_id. A partir de entonces SAC usa una autenticación delegada en Facebook para saber quien es *Friend* del *Owner*

El *Owner* da de alta los *WTs* en SAC. De esta manera SAC obtiene las credenciales para logarse como *Owner*

Al administrar la compartición desde SAC el *Owner* tendrá un lugar único para administrar su *WT*

Como se loga con las credenciales del *Owner* puede ver todos los *Actions* de un *WT* permitiendo compartir indistintamente cualquier *Action*

Para disponer de *WT* para trabajar y poder compartirlos y con los que trabajar hemos creado una aplicación que funciona a modo de hub de *WTs*. Esta aplicación se llama *iot\_emulator*. Se apoya en el modelo de *WT* propuesto por W3Consortium [LINK2].

El objetivo es poner en marcha y comunicar estas dos aplicaciones para poder permitirnos evaluar la aplicabilidad de la accesibilidad y compartición de *WT* por medio de Facebook.

Resolver un problemas intrínsecos de *Iot* que es la manera de Compartir de manera eficiente y segura uno o varios *WT* con *Friends*

Estas son las premisas para la aplicación:

* Esta aplicación consulta, (no almacena) *Actions* ni *Properties* de los *WT*
* Dicha aplicación se loga en *WT* con las credenciales aportadas por el *Owner*
* Consulta los datos necesario
* Esta aplicación sirve de intermediario entre un request de *Friend* para consultar

# Bibliografía

1-"Sharing Using Social Networks in a Compossable Web of Things"

2-El de w3Consorcium