

**Sistemas de Computación**  
Departamento de Computación  
*FCEPyN - UNC*



**INGENIERÍA EN  
COMPUTACIÓN**

## Trabajo Final - Control Remoto de un portón doméstico

Tomattis, Natasha Mat:38.728.783

[natitomattis@gmail.com](mailto:natitomattis@gmail.com)

Trombotto, Agustín Mat:39.071.116

[gutitrombotto@gmail.com](mailto:gutitrombotto@gmail.com)

Aguerreberry, Matthew Mat:93.739.112

[mtaguerreberry@gmail.com](mailto:mtaguerreberry@gmail.com)

5 de julio de 2017

# Índice

1. Introducción	3
2. Marco Teórico	3
2.1. Comparación con potenciales competidores	3
2.1.1. OPEN-SESAME	3
2.1.2. GOGOGATE	4
3. Requerimientos Funcionales	5
3.1. Instalación sencilla	5
3.2. Fácil uso	5
3.3. Seguridad	5
4. Diseño	5
4.1. Modelado a nivel de sistema	5
4.2. Comportamiento	7
5. Decisión de diseño	8
6. Bibliografía	8

## 1. Introducción

En este informe se desarrolla el modelado, la implementación y la documentación de un sistema de Computación. Un Sistema de computación o también llamado “Cyber Physical System” está constituido tanto por componentes de hardware como por software además de la relación que el mismo tiene con el usuario y con el ambiente que lo rodea. En este caso, se desarrollará un sistema de control de un portón eléctrico con conexión remota mediante un dispositivo móvil.

Para la realización de este proyecto se tendrá en cuenta el standard SYSML para el modelado del problema. El mismo provee un esquema gráfico que facilita el diseño del sistema y a su vez ayuda en la obtención de requerimientos funcionales.

El sistema desarrollado tiene un enfoque principalmente al usuario. Por ende se tiene en cuenta el fácil uso y la sencilla implementación teniendo, por este motivo, algunas limitaciones en lo tecnológico.

## 2. Marco Teórico

Se va a desarrollar un controlador para portones automáticos que permite una rápida instalación en un portón automático convencional, controlado desde una aplicación de celular.

La motivación es que los usuarios no tengan riesgos de extraviar los controles de los portones automáticos, lo cual no solo afectaría al usuario en que no podrá ingresar a su garaje sino que también sería una amenaza a su seguridad ya que un control convencional no cuenta con una autenticación de usuario.

### 2.1. Comparación con potenciales competidores

Se analizó las ofertas en el mercado de productos similares o sustitutos al sistema que se va a desarrollar. En un principio tras un análisis en el mercado local (Argentina) no se obtuvieron resultados de un producto que se encontrara a la venta con estas características. Por otro lado en un mercado más amplio se obtuvieron dos resultados de productos sustitutos, ambos cumplen la misma funcionalidad pero implementados de diferente manera

#### 2.1.1. OPEN-SESAME

Open Sesame Garage Door Opener ofrece una pequeña caja bluetooth y una aplicación para smartphone que controla el portón a través de una conexión bluetooth. El precio de este producto es el de un control regular, incentivando a los clientes de esta forma a reemplazar los controles infrarrojos convencionales.

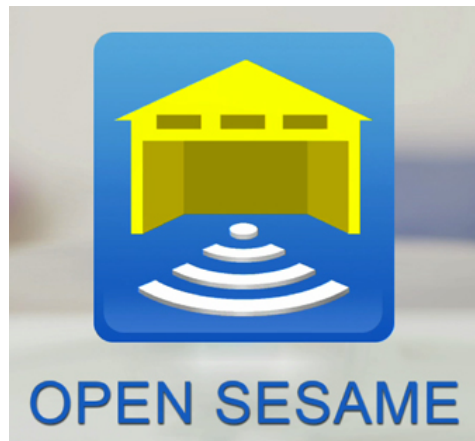


Figura 1: Logo de Gogogate

[Pagina web de OPEN-SESAME](#)

#### 2.1.2. GOGOGATE

Es una solución más completa, a diferencia del producto anterior este se comunica con el cliente a través de la red LAN del hogar. Tiene una aplicación móvil mucho más completa con control de acceso de usuarios, permitiendo el ingreso de diferentes clases de usuarios, un servicio de cámara para monitorear el garage, una sirena que advierte cuando el portón se abre o cierra, sensor de temperatura, entre otros servicios. Desde el punto de vista de hardware ofrece un dispositivo con cámara y alarma integrada, capacidad para conexión por ethernet o wireless



Figura 2: Logo de Gogogate



Figura 3: Dispositivo de Gogogate

### 3. Requerimientos Funcionales

#### 3.1. Instalación sencilla

Actualmente, los portones automáticos manejados por un simple control remoto independiente, dominan el mercado. Para que los usuarios tomen la iniciativa de cambiar de sistema este nuevo sistema debe ser de fácil y rápida instalación, de no ser así, los usuarios no optarán por realizar el cambio ya que implicaría un gran gasto de dinero y tiempo. Para lograr esto, el hardware utilizado será un simple embebido de bajo costo y gran disponibilidad.

#### 3.2. Fácil uso

La interfaz de usuario debe ser amigable e intuitiva. Tendrá un diseño orientado para dispositivos táctiles. No contendrá muchas opciones de configuración, ya que esto será concentrado en el servidor de control.

#### 3.3. Seguridad

Aquí se exige que solamente el usuario autorizado pueda ser capaz de controlar el comportamiento del portón. Para esto, se le solicitará al usuario se registre inicialmente en el sistema. Luego, cuando se desee abrir o cerrar el portón dicho usuario deberá proporcionar un PIN de autenticación extra.

### 4. Diseño

SysML define tres grupos de diagramas a alto nivel, estructura (modelo estático), comportamiento (modelo dinámico) y requerimientos. Los requerimientos en este caso ya fueron definidos anteriormente, ahora definiremos los bloques restantes:

#### 4.1. Modelado a nivel de sistema

Para este caso se utilizó un diagrama de definición de bloques para mostrar la relación entre los elementos del sistema y los elementos externos que afectarán al comportamiento del sistema.

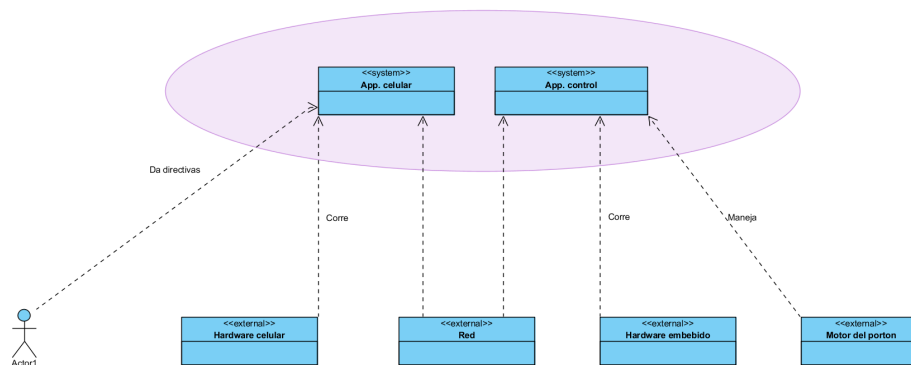


Figura 4: Diagrama de estructura

Se puede ver que la zona sombreada son los elementos que son parte del sistema y en el extremo inferior los elementos que afectan en su comportamiento

Una vez definida la relación del sistema con su entorno se definen los bloques internos de las partes de los elementos del sistema en este caso la App Móvil y App Control

- App Móvil Es la aplicación que correrá en el dispositivo móvil del cliente, por lo tanto se deberán tener en cuenta las librerías que permitan a la aplicación comunicarse con el entorno

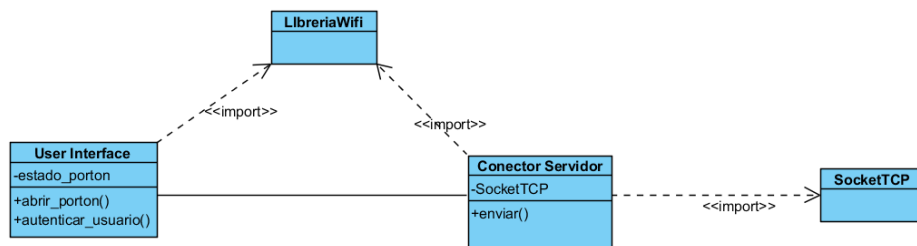


Figura 5: Diagrama de clases de App Movil

- App Control esta aplicación correrá en el hardware embebido, por lo tanto también deberá incluir las librerías de los módulos que utilizará.

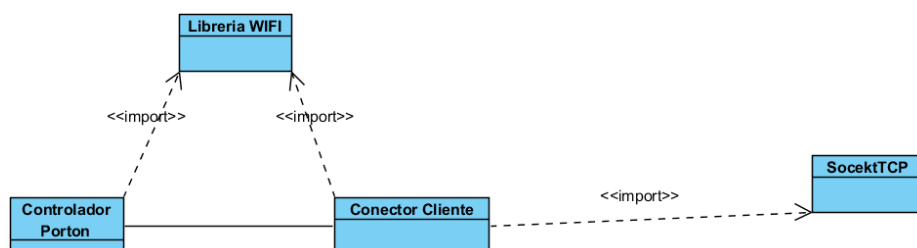


Figura 6: Diagrama de clases de App Control

## 4.2. Comportamiento

Para comenzar con el análisis de comportamiento se realizó un diagrama de casos de uso que brinda una descripción a muy alto nivel de las funcionalidades del sistema, respecto al usuario. Dado que el comportamiento del sistema es muy sencillo en este caso solo se analiza un caso de uso.

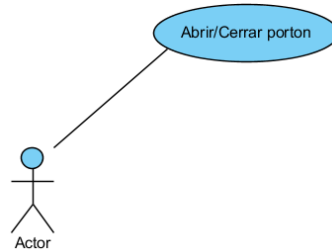


Figura 7: Diagrama de casos de uso

Una vez definido el caso de uso, se analizan las acciones que se deben llevar a cabo para cumplir con esa funcionalidad para esto se confecciona diagrama de actividad.

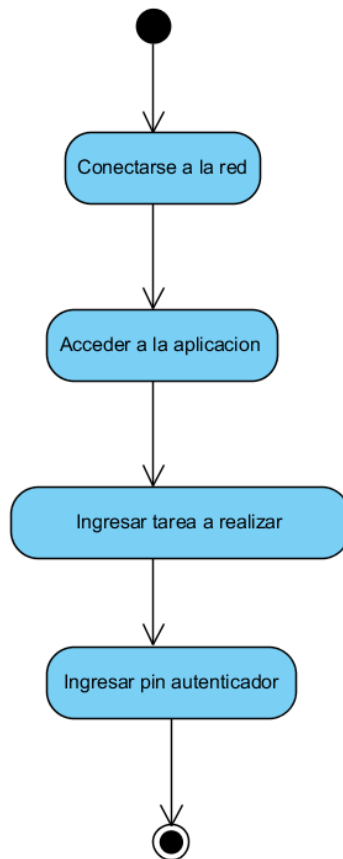


Figura 8: Diagrama de actividad de abrir/cerrar portón

Un diagrama de secuencia nos mostrara la interacción entre las diferentes partes del sistema cuando durante el caso de uso abrir/cerrar portón:

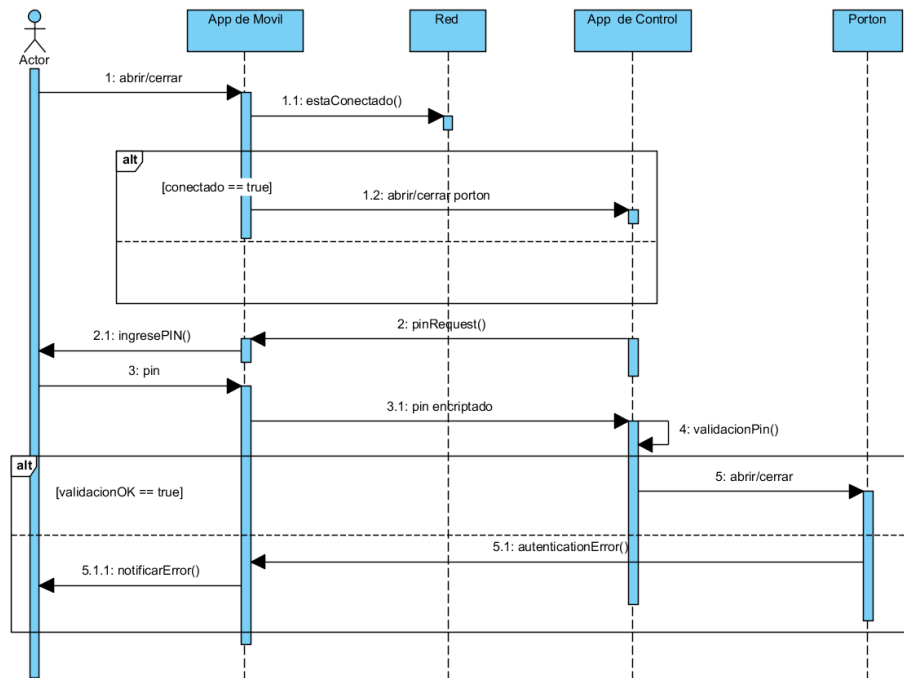


Figura 9: Diagrama de secuencia abrir/cerrar portón

## 5. Decisión de diseño

Luego de la descripción y modelado del sistema se define los recursos puntales que usaremos. Los mismos son los siguientes:

- Hardware Embebido: Raspberry Pi 3; este hardware tiene la capacidad de procesamiento necesaria para soportar los requerimientos del sistema, además de las interfaces necesarias como conectividad WiFi.
- App Control: se desarrollará en el lenguaje Python. Python es un lenguaje interpretado que posee gran cantidad de librerías para el manejo de interfaces en la Raspberry Pi. Además es de muy fácil uso y tiene una comunidad grande para enfrentar cualquier dificultad.
- App Celular: se usará Android Studio para crear la aplicación de celular. El mismo brinda un manejo muy fácil para programar, montando en java y con gran cantidad de clases de uso general del dispositivo.

## 6. Bibliografía