

SOFTWARE ARCHITECTURE DOCUMENTATION

ARQUIHUB

Alejandro Gracia Rey, Luis Miguel González Fernández, Anyella Valeria Pérez
Buendía, Santiago Rodríguez González.
Universidad de los Andes, Bogotá, Colombia

Fecha de presentación: mayo 19 de 2018

Índice

1. Restricciones y escenarios de calidad.....	1
1.1 Restricciones:	1
1.1.1 Impuestos por Yale:	1
1.2 Escenarios de Calidad	2
1.2.1 Desempeño:	2
1.2.2 Disponibilidad:	2
1.2.3 Escalabilidad:.....	2
1.2.4 Seguridad:.....	3
1.2.5 Recuperación ante fallos:.....	3
2. Puntos de vista del proyecto	4
2.1 Punto de vista contexto	4
2.1.1 Diagrama de contexto.....	4
2.2 Punto de vista funcional	5
2.2.1 Diagrama de casos de uso.....	5
2.2.2 Diagrama de componentes.....	6
2.3 Punto de vista de información	7
2.3.1 Diagrama Entidad/Relación.....	7
2.4 Punto de vista de desarrollo	8
2.4.1 Diagrama de paquetes.....	8
2.5 Punto de vista de despliegue	9
2.5.1 Diagrama de despliegue.....	9

1. Restricciones y escenarios de calidad

1.1 Restricciones:

1.1.1 Impuestos por Yale:

Tecnología:

- La aplicación debe utilizar un sistema inalámbrico a través de internet para comunicar la cerradura con el sistema.

Negocio:

- La arquitectura y la aplicación debe ser desarrollada en 4 meses.
- Se deben hacer entregas parciales cada 3-4 semanas para que se vaya verificando el avance en el proyecto, junto con sus pruebas correspondientes.

Trabajo en equipo:

- Se debe presentar una planeación de actividades faltantes en cada etapa del proceso.
- Se debe manejar un repositorio de versiones para el desarrollo del proyecto.

Estas restricciones se seleccionaron teniendo en cuenta que el diseño de la arquitectura fue escogido por el grupo de desarrollo del proyecto, por lo que no existen muchas restricciones en cuanto a tecnología. Se optó por manejar un estilo de arquitectura Lambda y los implementos utilizados para la entidad física fueron sugeridos.

1.2 Escenarios de Calidad**1.2.1 Desempeño:**

Escenario	Escenario 1.1
Identificador	EC1
Prioridad	Alta
Atributo de calidad	Desempeño
Fuente	Yale Connect Hub, Sistema Central Yale
Estímulo	Envío de emergencia/Fallos
Ambiente	Normal
Medida Esperada Respuesta	Se espera que el sistema pueda recibir al menos 1500 emergencias o fallos, en un tiempo de 1 min con 0% de error y tiempo de respuesta menor a 1 segundo.

1.2.2 Disponibilidad:

Escenario	Escenario 2.1
Identificador	EC2
Prioridad	Alta
Atributo de calidad	Disponibilidad
Fuente	Yale Connect Hub, Sistema Central Yale
Estímulo	Envío de emergencia/Fallos
Ambiente	Normal
Medida Esperada Respuesta	Se espera que el sistema pueda recibir alarmas al menos el 99.9999% del tiempo al año, es decir una caída de máximo 31.5s anualmente.

1.2.3 Escalabilidad:

Escenario	Escenario 3.1
Identificador	EC3
Prioridad	Alta
Atributo de calidad	Escalabilidad
Fuente	Yale Connect Hub, Sistema Central Yale

Estímulo	Envío de emergencia/Fallos
Ambiente	Normal
Medida Esperada Respuesta	Se espera que el sistema pueda atender 300.000 peticiones en una ventana de 1 minuto.

1.2.4 Seguridad:

Escenario	Escenario 4.1
Identificador	EC4
Prioridad	Alta
Atributo de calidad	Seguridad
Fuente	Yale Connect Hub, Sistema Central Yale
Estímulo	Envío de emergencia/Fallos
Ambiente	Normal
Medida Esperada Respuesta	Se espera que el sistema atienda y reciba información solo de dispositivos y usuarios autenticados.

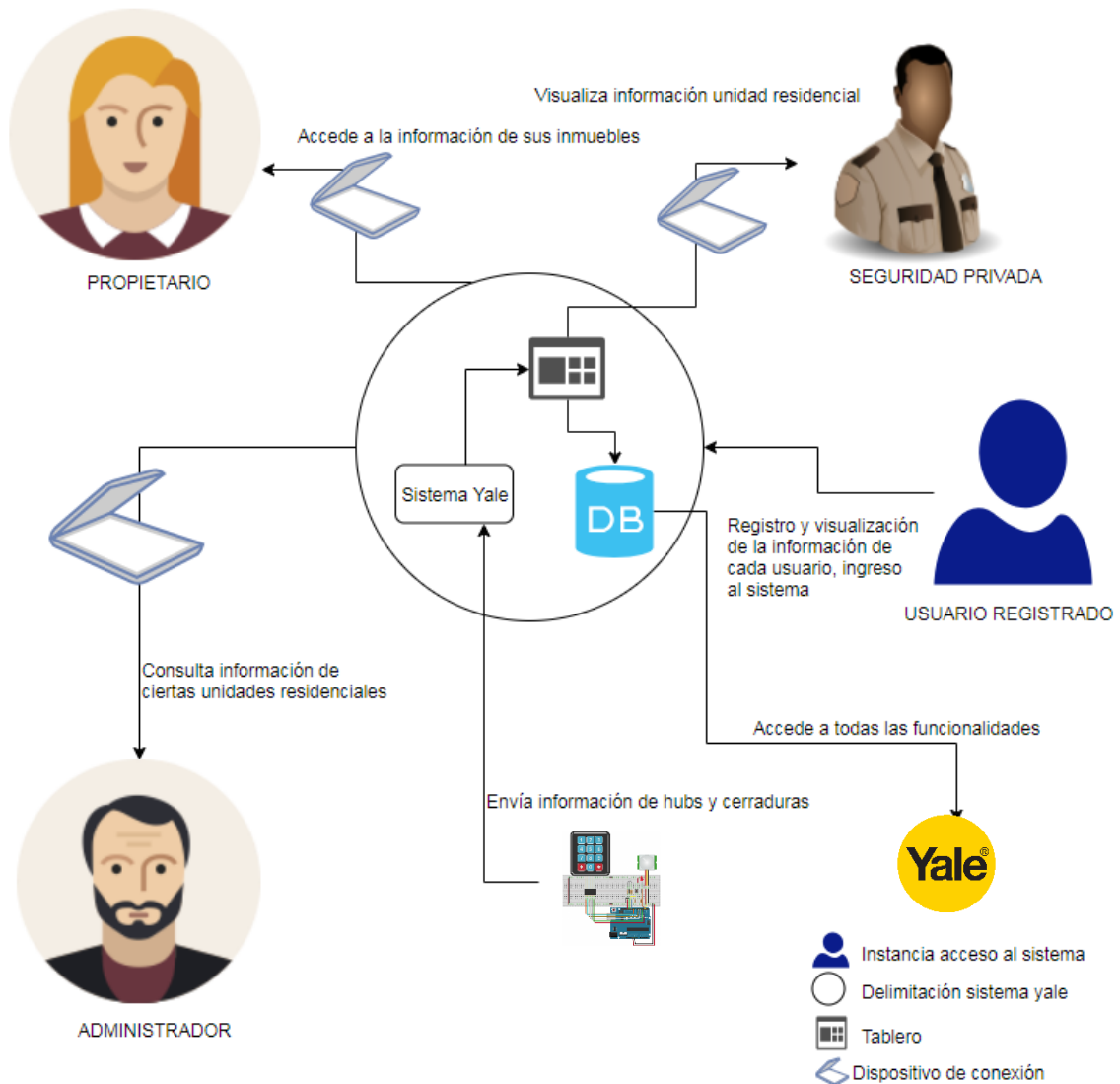
1.2.5 Recuperación ante fallos:

Escenario	Escenario 5.1
Identificador	EC5
Prioridad	Alta
Atributo de calidad	Mantenibilidad
Fuente	Yale Connect Hub, Sistema Central Yale
Estímulo	Envío de emergencia/Fallos
Ambiente	Saturación del sistema
Medida Esperada Respuesta	Se espera que el sistema en caso de fallos tarde máximo 10 segundos. En ser reemplazado por otro de los servidores funcionando.

2. Puntos de vista del proyecto

2.1 Punto de vista contexto

2.1.1 Diagrama de contexto



Información que fluye: Estados de las cerraduras, alarmas, estado de las unidades, información de los usuarios.

La aplicación será accedida por 3 tipos de usuarios:

- El usuario Administrador, el cual puede consultar las unidades residenciales y toda la información relacionada con los dispositivos en estas.
- El usuario Propietario, que posee únicamente acceso a la información de sus inmuebles, claves de acceso, y es notificado por el sistema en caso de emergencia.
- El usuario Seguridad Privada, que posee acceso a un tablero que contiene la información de la unidad residencial bajo su cuidado, con un reporte del estado de cada inmueble.

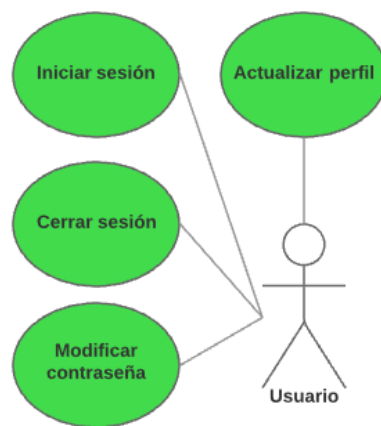
- El usuario Yale, que es un usuario único que tiene acceso a toda la información del sistema.

Se debe asegurar que el sistema está en capacidad de responder a las peticiones de todos los usuarios.

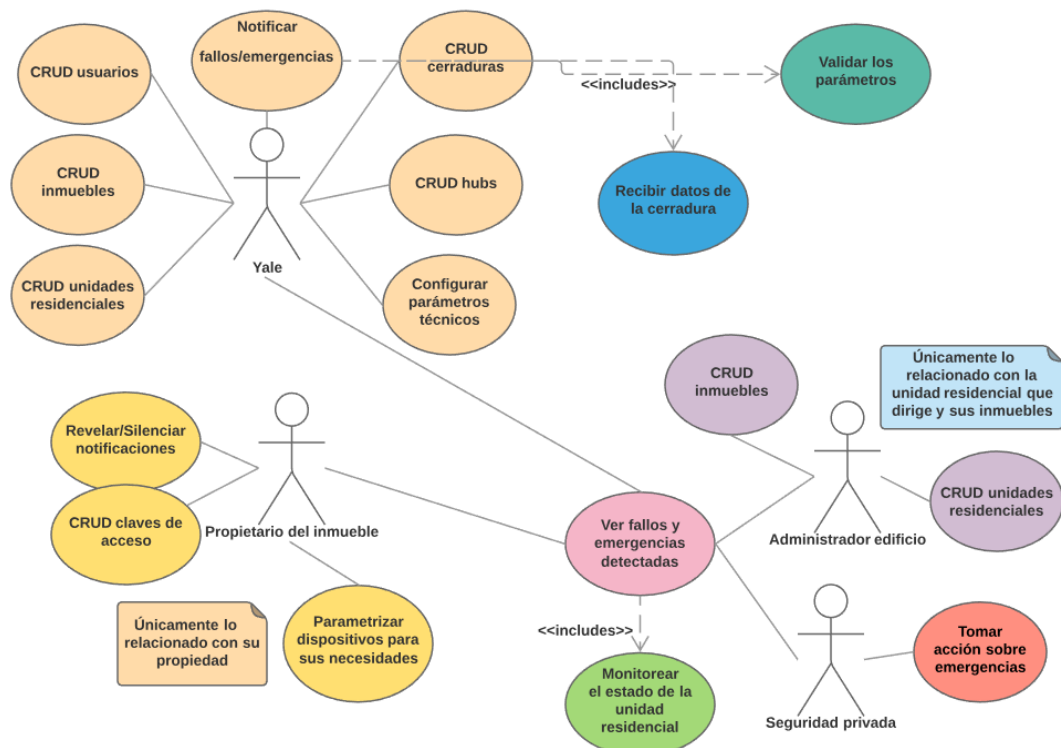
2.2 Punto de vista funcional

2.2.1 Diagrama de casos de uso

En esta arquitectura se implementa un usuario principal y simplificado para cada uno de los tipos de usuarios del sistema (Yale, PropietarioInmueble, SeguridadPrivada, Administrador), instanciándolos independientemente de su rol, y tiene como función permitir su conexión de forma autorizada a la plataforma y dar lugar a los siguientes casos de uso.



El Sistema Yale, tiene acceso libremente a todas las funcionalidades de la aplicación, desde crear un usuario hasta ver los fallos detectados en un inmueble específico.

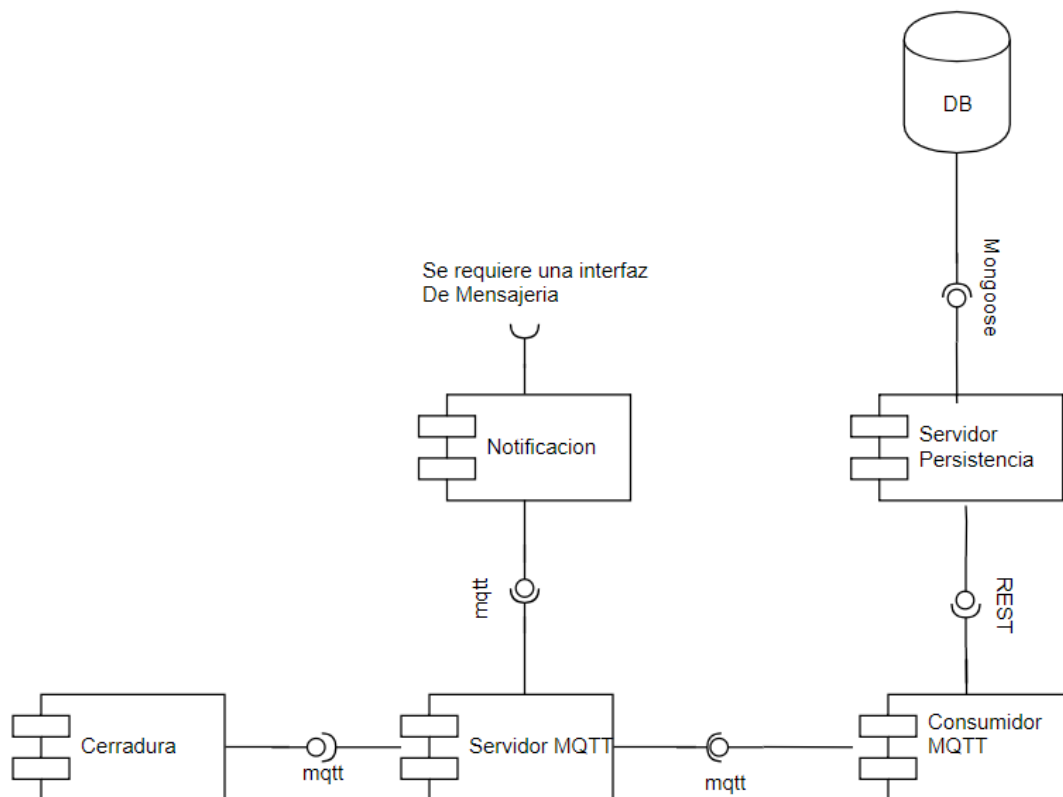


El usuario Propietario maneja únicamente toda la información de sus cerraduras de acuerdo a las necesidades que tenga.

El administrador del edificio maneja información de inmuebles y unidades que dirige.

Seguridad privada ve el estado de cada inmueble de la unidad residencial que está bajo su cuidado.

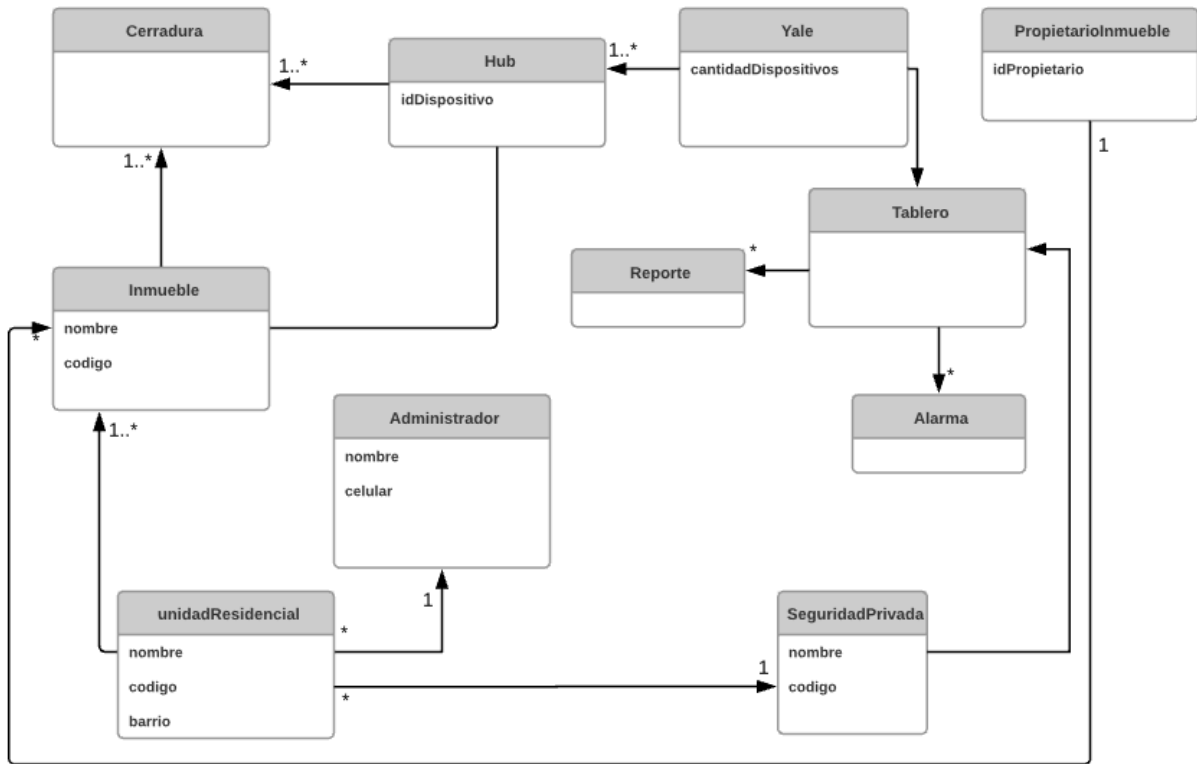
2.2.2 Diagrama de componentes



El nodo principal de nuestra aplicación es el servidor MQTT, usado para comunicar las diferentes partes del sistema. El servidor de mensajería comunica a la cerradura para darle sus claves, como para notificar de las diferentes alarmas que se pueden generar. Los mensajes de la cerradura pasan a un servidor de notificación por correo (por ahora es un *mock*) y a un programa que se encarga de redireccionar los mensajes al servidor de persistencia.

2.3 Punto de vista de información

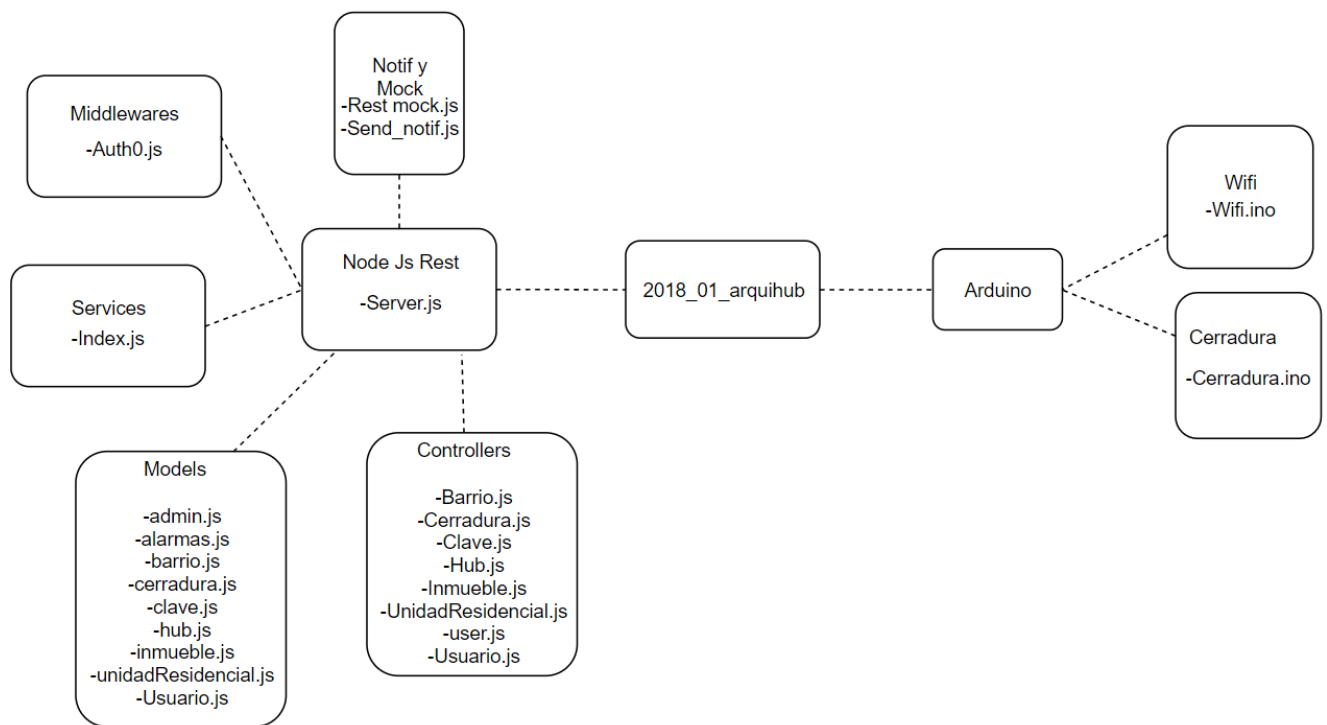
2.3.1 Diagrama Entidad/Relación



Principalmente, se encuentra la entidad principal “Yale”, la cual está relacionada con todos los hubs del sistema y con el tablero de visualización de alarmas en los inmuebles. Cada Unidad Residencial tiene muchos inmuebles y está dirigida por un Administrador y cuenta con una empresa de seguridad para los inmuebles de la misma. Por otro lado, cada inmueble está asociado a las posibles diferentes cerraduras que pueda tener (aunque en este caso sólo se maneja una por inmueble) y cada cerradura está asociada al hub de ese inmueble.

2.4 Punto de vista de desarrollo

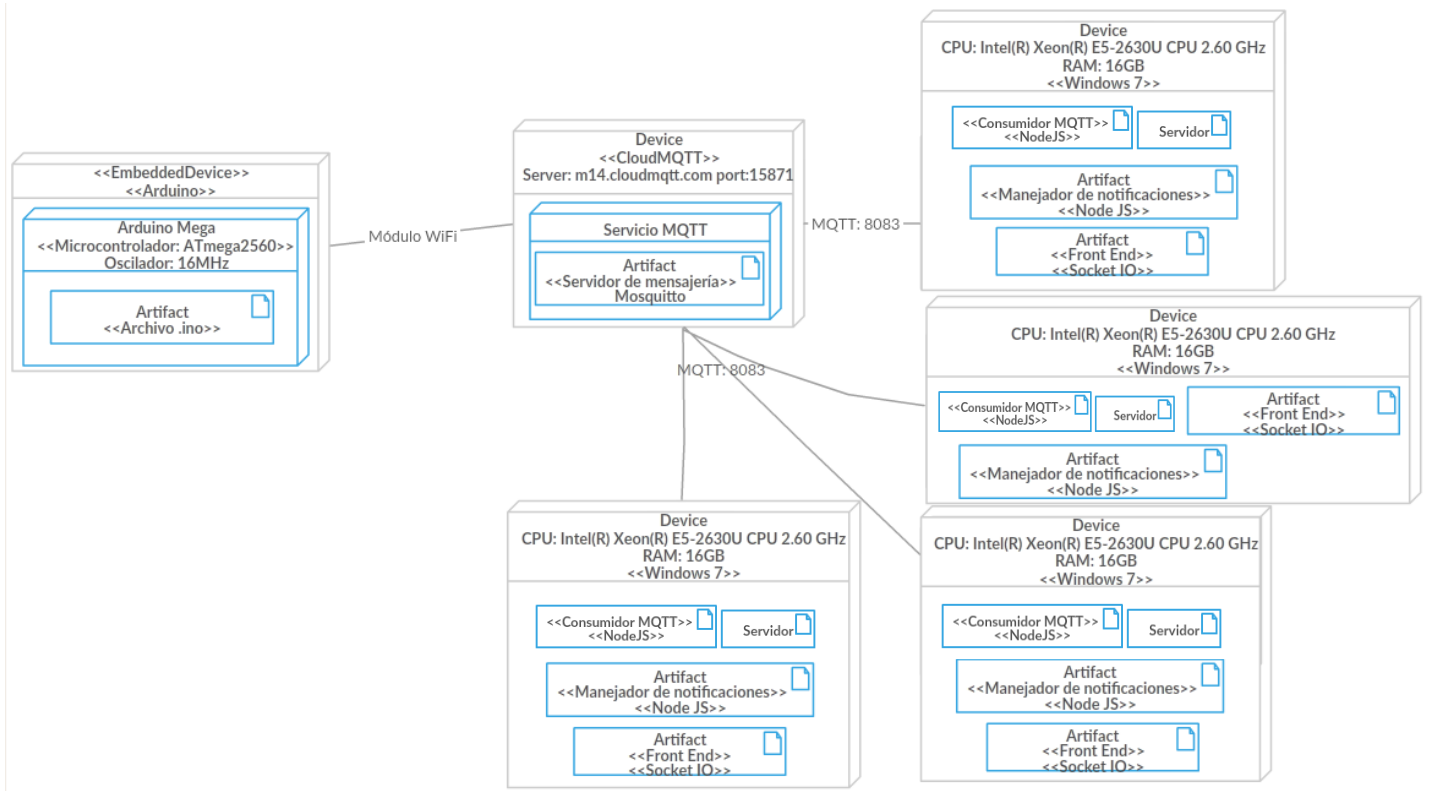
2.4.1 Diagrama de paquetes



Se puede ver en el diagrama que la gran mayoría de nuestros artefactos de software están hechos en JavaScript. Los archivos están en carpetas con nombres significativos para fácilmente saber qué hace lo que está adentro. Por ejemplo, la diferencia entre lo contenido entre la carpeta Arduino y la Node JS Rest. Una tiene el código del Hub (entidad virtual) mientras que otra tiene el código de la cerradura (entidad física).

2.5 Punto de vista de despliegue

2.5.1 Diagrama de despliegue



Una parte central de nuestra aplicación es el servidor de mensajería MQTT que esta implementado en la nube para así evitar el manejo de la seguridad o de disponibilidad de este. El servidor conecta la cerradura con las diferentes máquinas que se tienen para el manejo de las alarmas e información histórica.

Cada servidor tiene el software necesario para el manejo de alarmas y de persistencia por su cuenta. Esto se hizo, con el fin de manejar las diferentes peticiones en cada servidor, donde cada uno representa un barrio con diferentes unidades residenciales, para así, no sobrecargar un solo servidor atendiendo las señales enviadas por las cerraduras que almacena.