

Documentación UML

El UML adjunto corresponde a la estructura (tanto de *front-end* como de *back-end*) del proyecto contenido en este repositorio de GitHub.

El *front-end*, denotado en el diagrama como 'Front', consiste en una aplicación construida mediante el *framework* React. A su vez, el *front-end* hace uso de una interfaz otorgada por Auth0 (denotada en el diagrama como 'Auth0') para manejar la autenticación y verificación de los usuarios que utilicen la plataforma. Por último, el contenido y la interfaz entregada por esta aplicación se entrega a Amazon CloudFront (denotado en el diagrama como 'CloudFront') para distribuirlo al *end user* de manera rápida y eficiente.

CloudFront, a su vez, le hace entrega de este contenido a un API Gateway (denotado en el diagrama como API GAT) encargado de intermediar las comunicaciones entre *front-end* y *back-end*. Este API Gateway está además construido junto a un Custom Authorizer (el cual se encuentra [en este repositorio de GitHub](#), también denotado en el diagrama como Custom Auth) que autoriza solicitudes asegurándose de que los *tokens* entregados correspondan a JWTs (*JSON Web Token*).

Este API Gateway, a su vez, se comunica con el *back-end* de la aplicación, componente construido en base a la librería 'FastAPI' de Python. A pesar de que éste no está denotado de manera explícita en el diagrama, consiste en los siguientes componentes:

- **Broker:** Corresponde a un servidor configurado por el cuerpo docente del curso. Es capaz de emitir información sobre la compra y venta de acciones correspondientes a distintas empresas. Además, es capaz de recibir *requests* enfocadas en la compra de estas acciones.
- **API:** Corresponde a la lógica de la aplicación, la cual se encarga de obtener eventos desde el *broker* correspondiente. Esta API también ofrece la opción de comprar estas acciones, notificando al *broker* cuando esto ocurre.
- **MQTT:** La comunicación entre API y *broker* se realiza en base a mensajes MQTT, los cuales destacan por ser livianos y de fácil manejo.
- **Database:** El *back-end* mantiene una base de datos destinada a almacenar cierta información sobre los usuarios que hacen uso de la aplicación, tal como la cantidad de dinero que estas cuentas poseen.
- **Nginx:** El servicio Nginx se encarga de otorgar el servicio de *proxy* para manejar las interacciones de *back-end*.
- **EC2:** Todo lo anterior (a excepción del Broker, ya que éste no es de la autoría de este grupo) se encapsula y deployea mediante EC2, servicio de Amazon Web Services que permite a los usuarios obtener máquinas virtuales para encargarse de ejecutar la aplicación en la nube.