Requerimientos técnicos de la solución

Se realizaron **tres carpetas distintas** y cada una contiene un proyecto distinto. Con el fin de reducir costos, es posible ejecutar las tres aplicaciones en el mismo servidor, o en caso de tener una alta demanda por parte los jugadores, es posible dividir la arquitectura en dos servidores.

El modelo pensado para desarrollar la aplicación se basa en el patrón de arquitectura de **Microservicios**, pues al granular el sistema, tenemos ventajas como las siguientes:

* En caso de fallo, el único servicio que queda indisponible es aquel que falló; el resto de la aplicación puede seguir funcionando con algunas bajas dependiendo de la relación de información entre aplicaciones.
* Se facilita encontrar y solucionar errores en la aplicación.
* La aplicación es **escalable**
* Entre otros.

**Atributos de calidad**

* Disponibilidad
* Seguridad
  + Protocolos seguros de transmisión de información a través de la red (https)
* Desempeño
  + Minimizar el margen de error al realizar peticiones a las aplicaciones
  + Rápida respuesta por parte de los servidores

# Videojuego

* Desarrollo en Unity
* Servidor para hosting

# Dashboard

* Servidor para hosting
* Desarrollo en React + Node.js

# Análisis de datos

* Servidor para hosting
* Desarrollo en python, Flask
* Conocimientos de Machine Learning