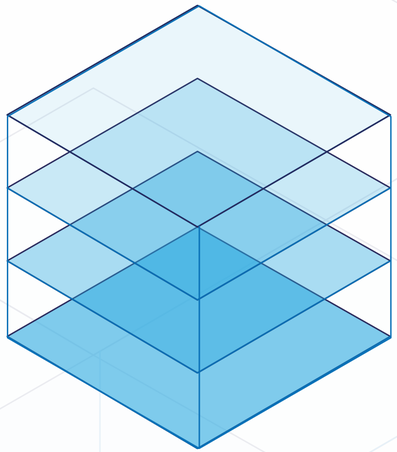


MISO

Maestría en Ingeniería de Software

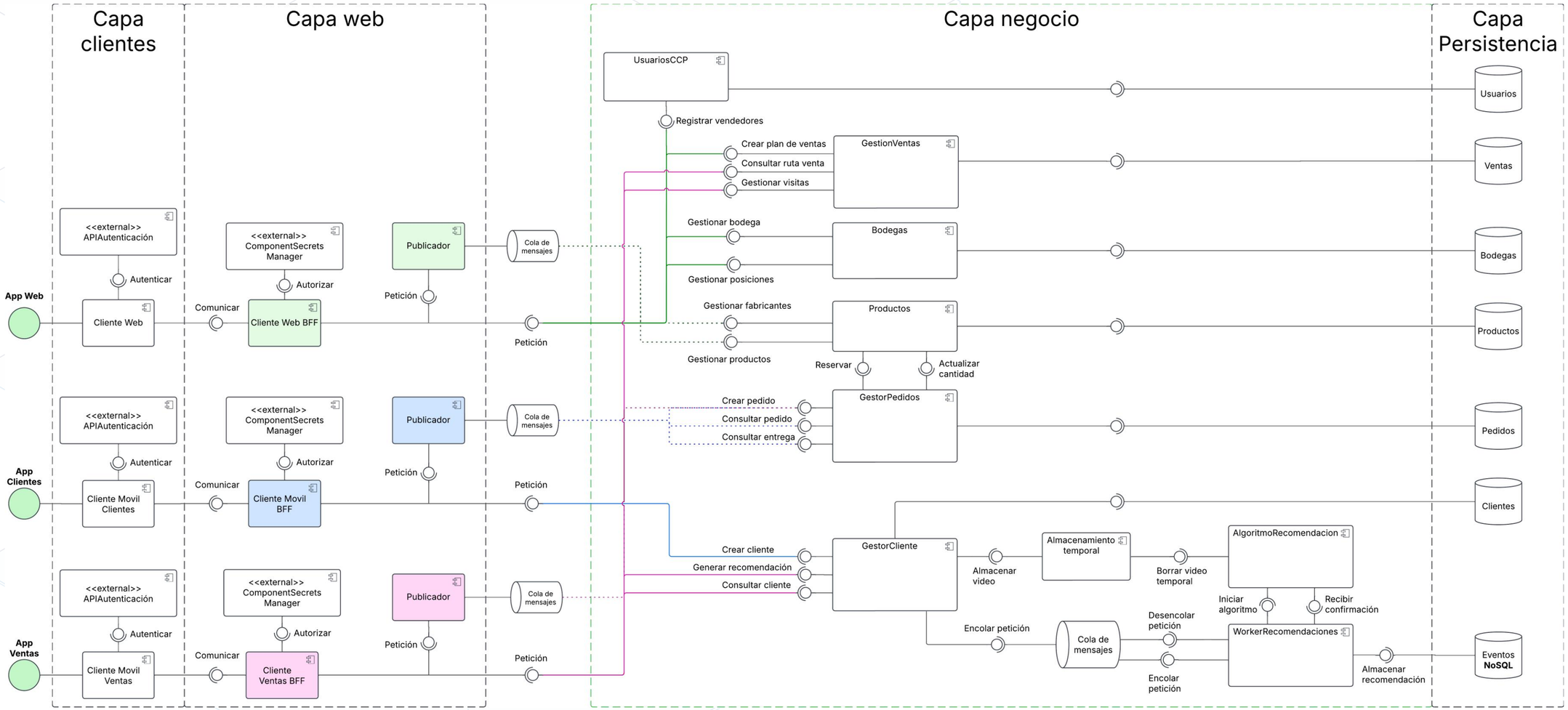
Hoja de Trabajo

Semana 5 – Diseño detallado y experimentación

Semana 5 – Diseño detallado y experimentación

Vista Funcional – (Estilo Microservicios)

Proyecto	CCP Supply	ID	VC-001	Elaboración	Grupo 7	Versión	1.0
Vista	Funcional		Modelo	Componentes			



Convención

Estilo microservicios

Interfaz

Estímulo

Componente

Base de datos

Vista Funcional – (Estilo Microservicios)

Proyecto	CCP Supply	**ID**	VC-001	**Elaboración**	Grupo 7	**Versión**	1.0
Vista	Funcional		**Modelo**	Componentes			

Patrones y tácticas de arquitectura utilizadas

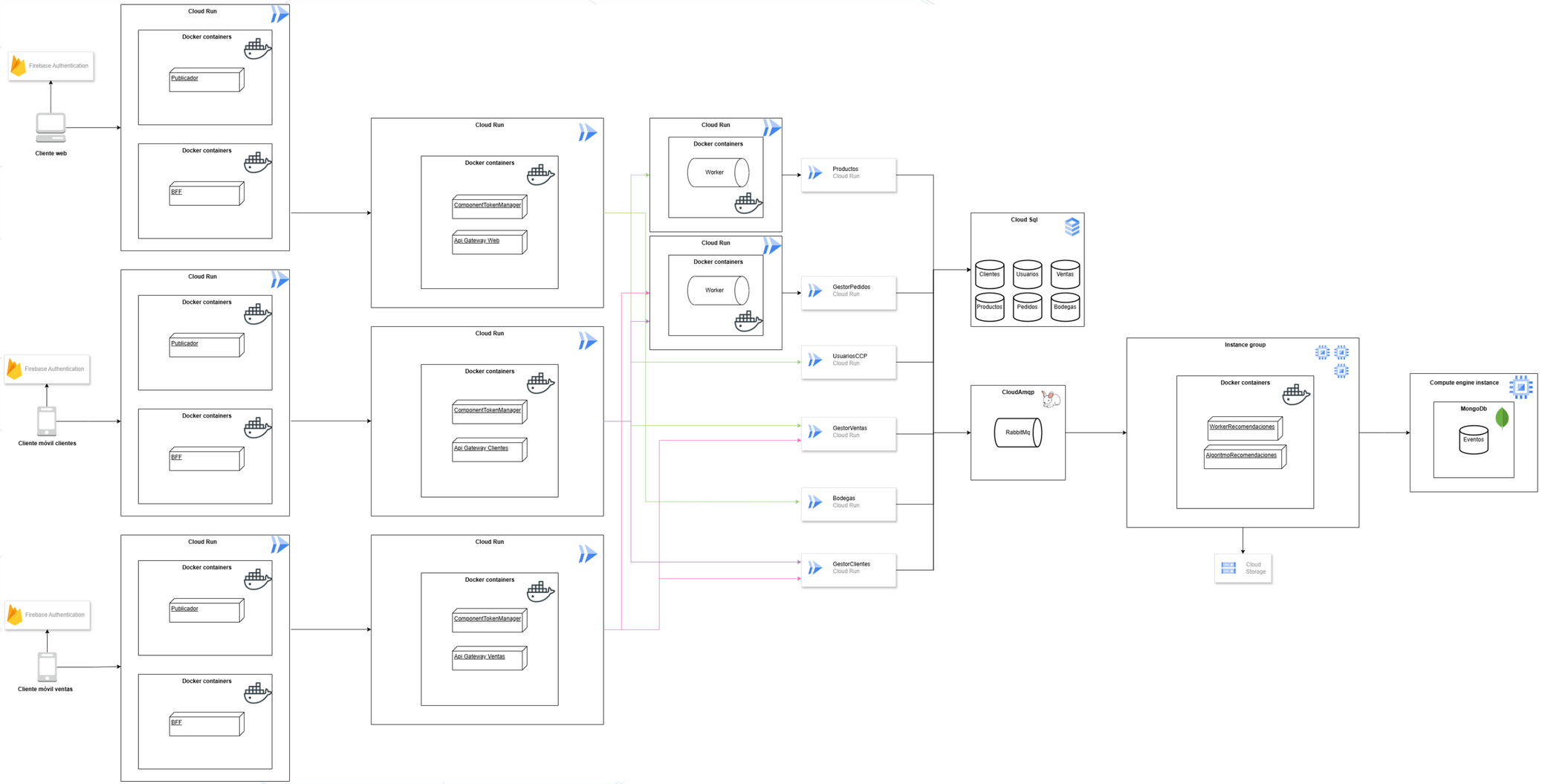
- Microservicios
- Backend For Frontend (BFF)
- Autenticación
- Autorización
- Colas de mensajería

Razonamiento sobre las principales decisiones de arquitectura tomadas en este modelo

- Por medio del estilo de microservicios se espera incrementar la disponibilidad del sistema y tener lógicas de negocio más focalizadas en los componentes asociados, mejorando la facilidad de modificación
- El patrón BFF se escogió en contraposición al patrón de API Gateway para reducir el tamaño de los diferentes aplicativos a realizar. En este caso particular, reducirá la dependencia de un único componente, aumentará la disponibilidad y permitirá redistribuir adecuadamente las cargas
- Se introdujo un servicio externo de autenticación. Este permitirá mantener, por medio del estándar OAuth2, mantener la identidad de los usuarios, tener trazabilidad de sus actividades y delegar roles y permisos. Así mismo, este servicio reducirá la carga de código y es más seguro que un servicio in-house
- Se utilizará un servicio cloud para el almacenamiento y uso de secrets, el cual permitirá autorizar a los componentes para realizar sus debidas operaciones
- Para el manejo del procesamiento de videos para la generación de recomendaciones se implementará una cola de mensajería que permitirá activar un worker que realizará el proceso de forma asíncrona. De esta forma se reduce el tiempo de respuesta y al implementar la lógica “exactly once”, se garantizará el procesamiento de todos los videos

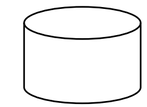
Vista de despliegue

Proyecto	CCP Supply	**ID**	VC-001	**Elaboración**	Grupo 7	**Versión**	1.0
Vista	Funcional		**Modelo**	Despliegue			



Convención

Nodos de ejecución



Base de datos

Cola de mensajes.

Vista de despliegue

Proyecto	CCP Supply	**ID**	VC-001	**Elaboración**	Grupo 7	**Versión**	1.0
Vista	Funcional		**Modelo**	Despliegue			

Patrones y tácticas de arquitectura utilizadas

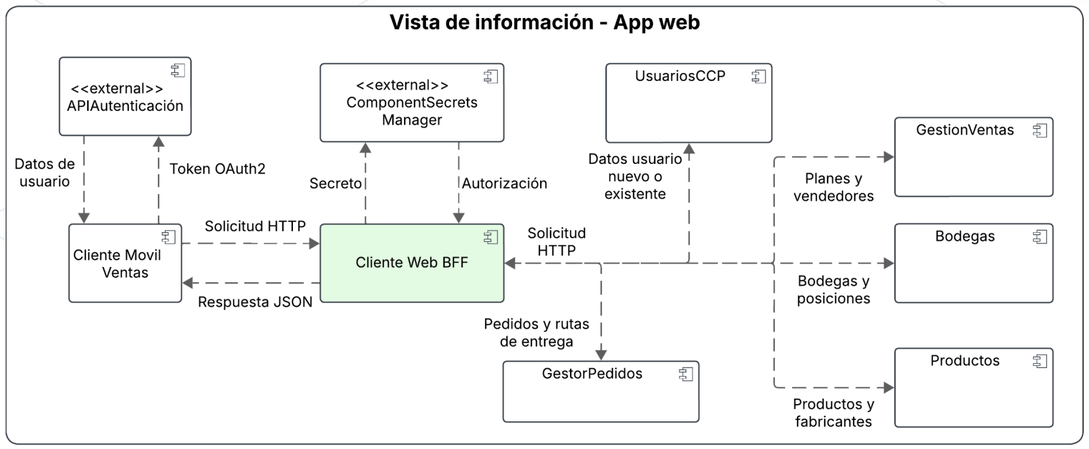
- Microservicios.
- Cliente-servidor

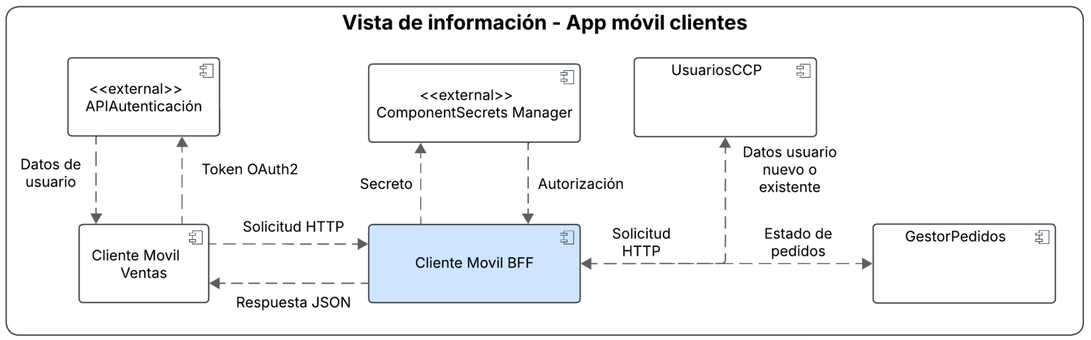
Razonamiento sobre las principales decisiones de arquitectura tomadas en este modelo

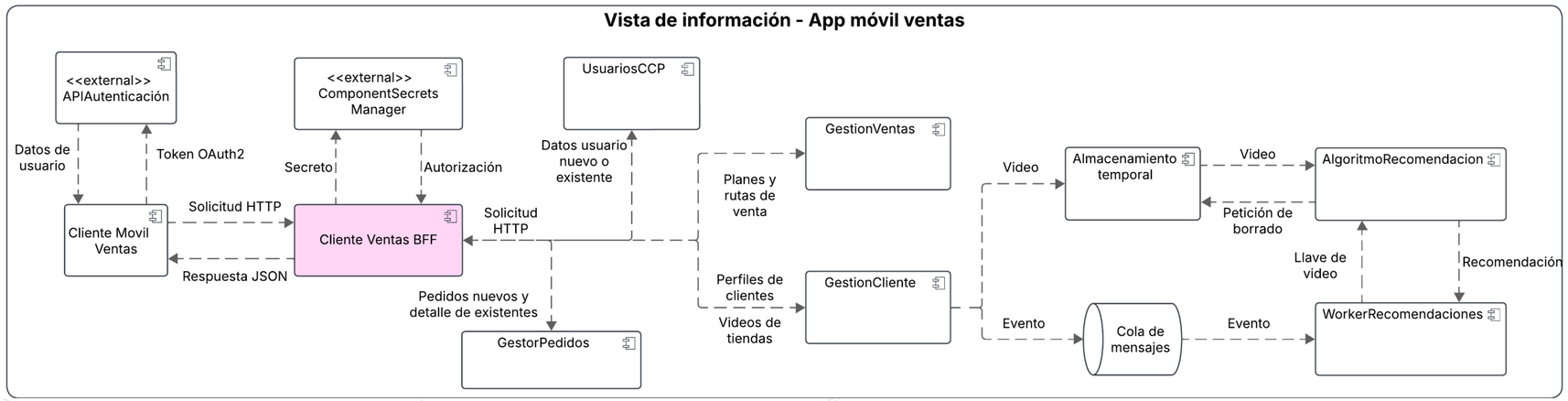
- El despliegue de la solución se hará principalmente utilizando la nube y los servicios de GCP.
- Para el proceso de autenticación de cada área del proyecto se utilizarán los servicios de Firebase authenticator, esto nos ayudara principalmente a acelerar el proceso de desarrollo y gestionar una misma herramienta para la parte web y móvil.
- Los diferentes servicios serán desplegados en un compute engine y cada uno en su propio container. Con esto en mente, se debe tener en cuenta la configuración de la VPC para la interconexión de todos los servicios (compute engines, bases de datos, sistema de encolamiento, entre otros.).
- Debido a la complejidad del sistema de recomendaciones para los clientes (basándonos en videos tomados por los vendedores), se utilizará una base de datos No relacional con el fin de tener mayor flexibilidad en el proceso de desarrollo del modelo y los objetos a persistir en una base de datos.

Vista de Información

Proyecto	CCP Supply	ID	VC-001	Elaboración	Grupo 7	Versión	1.0
Vista	Información		**Modelo**	Información			







Convención

Componente

Vista de Información

Proyecto	SportApp	**ID**	VC-004	**Elaboración**		**Versión**	1.1
Vista	Información		**Modelo**	Información			

Patrones y tácticas de arquitectura utilizadas

- Backend For Frontend (BFF)
- Colas de eventos

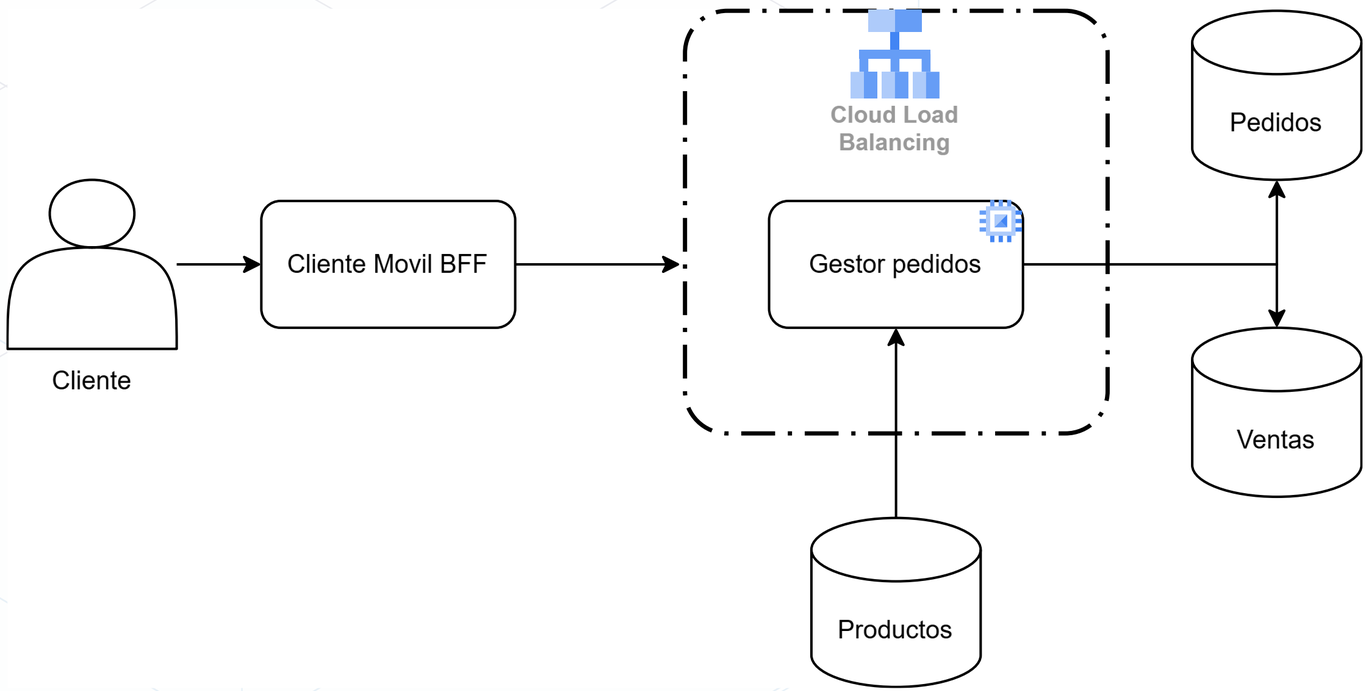
Razonamiento sobre las principales decisiones de arquitectura tomadas en este modelo

- Para el manejo del procesamiento de videos para la generación de recomendaciones se implementará una cola de mensajería que permitirá activar un worker que realizará el proceso de forma asíncrona. Esto, en conjunto con una lógica exactly once, muestra que el sistema será resiliente a fallos ya que podrá procesar más adelante los videos cargados cuándo el sistema esté disponible, y tener un espacio eficiente al eliminarlos del almacenamiento temporal enlazado a él
- El patrón BFF muestra en este caso como el sistema distribuido se parte en partes más pequeñas. Por tal razón, cada uno deberá manejar menos tipos de datos, aumentando significativamente la facilidad de modificación

Vista de Información

Proyecto	CCP Supply	ID	VC-001	Elaboración	Grupo 7	Versión	1.0
Vista	Concurrencia		Modelo	Concurrencia			

Vista de concurrencia de la operación de escalamiento para la creación de pedidos



Convención

Vista de Información

Proyecto	SportApp	**ID**	VC-004	**Elaboración**		**Versión**	1.1
Vista	Concurrencia		**Modelo**				

Patrones y tácticas de arquitectura utilizadas

- Backend For Frontend (BFF)
- Load balancer

Razonamiento sobre las principales decisiones de arquitectura tomadas en este modelo

- Como uno de los requisitos de calidad principales en nuestra arquitectura es la escalabilidad, y dado que vamos a hacer un experimento relacionado a ello. Desarrollamos a un alto nivel el segmento del diagrama completo de arquitectura para el componente de realización de pedidos. Esto con el fin de adelantar los experimentos de arquitectura, documentar la hipótesis y probarla para este ciclo.
- El principal componente que debe escalar y al cual se evaluara su escalamiento es el Gestor de pedidos.
- Se busca probar el patrón BBF en un ambiente de alta concurrencia, evaluar cómo se comporta el balanceador de cargar y el BFF y evaluar los límites del BFF en este tipo de condiciones.
- El backend de la aplicación independientemente de ser móvil o web estará alojada en la nube. En este caso las peticiones serán hechas directamente a la nube.