

### Anexo de la Pregunta 3:

#### Tecnologías a utilizar:

**Clientes:** son las personas que visitan la web de CineManzana por medio de un browser de Internet.

**Servidor de Autenticación:** es el servidor que se encarga de validar las credenciales de los clientes y encargados de CineManzana.

**OpenID Connect Server:** servidor empleado para el protocolo de autenticación de los usuarios y clientes de CineManzana. Emplea formato JSON para estructurar su data y RESTful HTTP API.

**API Gateway HTTP Proxy:** es el API que se encarga de establecer un punto de entrada para cada una de los microservicios creados, además tiene la función de filtrar peticiones del exterior que no están registradas o autorizadas para el sistema. Tiene comunicación directa con el Servidor de Autenticación.

**Aplicación para Web Page:** representa la plataforma web de CineManzana donde los clientes ingresan para comprar sus entradas, está conformado por código HTML5, CSS (funciona también para las aplicaciones móviles), XML y Javascript.

**Payment Service:** representa el Sistema de pago cuando el cliente realiza el proceso de comprar una entrada, está conformada por una API Checkout para gestionar el servicio online, es multi-dispositivo.

**API Checkout (Payment Service):** se encarga de integrar y comunicar el proceso de compra al sistema, emplea protocolo REST, interfaces HTTPS y una API Key para uso público o privado.

**Flask REST Framework:** es una extensión del Web Framework Flask, está escrito en Python, es considerado un micro framework puesto que no requiere de librerías en general para sus aplicaciones.

API	Descripción	Tecnología
API Función	Permite realizar las funciones de escritura ( <b>command</b> ) y consultas ( <b>query</b> ) para las funciones de películas.	Flask REST Framework
API Película	Permite realizar las funciones de escritura ( <b>command</b> ) y consultas ( <b>query</b> ) para las películas.	Flask REST Framework
API Usuario	Permite realizar las funciones de escritura ( <b>command</b> ) y consultas ( <b>query</b> ) para gestionar el usuario del sistema de CineManzana.	Flask REST Framework
API Cine	Permite realizar las funciones de escritura ( <b>command</b> ) y consultas ( <b>query</b> ) para la gestión de los Cines de CineManzana.	Flask REST Framework

API Encargado	Permite realizar las funciones de escritura ( <b>command</b> ) y consultas ( <b>query</b> ) para los encargados de CineManzana.	Flask REST Framework
API Cliente	Permite realizar las funciones de escritura ( <b>command</b> ) y consultas ( <b>query</b> ) para los clientes.	Flask REST Framework
API Sala	Permite realizar las funciones de escritura ( <b>command</b> ) y consultas ( <b>query</b> ) para la gestión de las salas.	Flask REST Framework
API CadenaCine	Permite realizar las funciones de escritura ( <b>command</b> ) y consultas ( <b>query</b> ) para gestionar las cadenas de cines.	Flask REST Framework

#### Base de Datos:

**Servidor de BD:** Se empleará **Apache Cassandra DB**.

Objetivo y descripción: Apache Cassandra DB nos brinda un gran nivel de escalabilidad en clústers de servidores lo cual logra satisfacer los objetivos de CineManzana ofreciendo a los clientes mayor disponibilidad de sus servicios, además se caracteriza por su alto rendimiento para el manejo de grandes cantidades de data y tolerancia a fallas, realiza particionamiento de la data entre todos los nodos del clúster. Trabaja adicionalmente con Docker Container para una construcción y despliegue DevOps más ágil.

**Servidor de BD:** Se empleará **MongoDB**.

Objetivo y descripción: MongoDB nos permite realizar Sharding (particionamiento de data con gran densidad), replicación, emplea Javascript para las consultas haciendo estas más rápidas para ejecutarse. Recopila los datos utilizando múltiples servidores. Está diseñado para proporcionar escalabilidad horizontal y alta disponibilidad de los registros.

#### Redes Sociales:

**Base de Datos SQLite Database:** se emplea para la gestión de la base de datos de los dispositivos móviles (iOS y Android).

**API Facebook:** permite establecer comunicación entre el sistema de CineManzana con la red social de Facebook.

Objetivo y descripción: permitirá al sistema interactuar con la información de perfil de los usuarios de CineManzana mientras que estos podrán añadir información social usando su perfil, páginas, fotos, grupos. Trabaja mediante un protocolo RESTful mientras que las respuestas se generan en formato JSON.

**API Twitter:** permite establecer comunicación entre el sistema de CineManzana con la red social de Twitter.

Objetivo y descripción: permitirá al sistema trabajar con la data core de Twitter, suministrando los tuits de los usuarios (seguidores y seguidos), publicaciones, trending topics, etc. Trabaja mediante formato JSON, REST API, Search API y Streaming API.

**API Instagram:** permite establecer comunicación entre el sistema de CineManzana con la red social de Instagram.

Objetivo y descripción: permitirá al sistema trabajar con todo lo relacionado a sharing de fotos, mobile sharing, mensajería de usuarios. Trabaja por medio de instancias Nginx y balanceador de carga Elastic de Amazon, emplea también librerías Django.