

Challenge Técnico — Team Lead VTEX

Galletas Chinas de la Fortuna

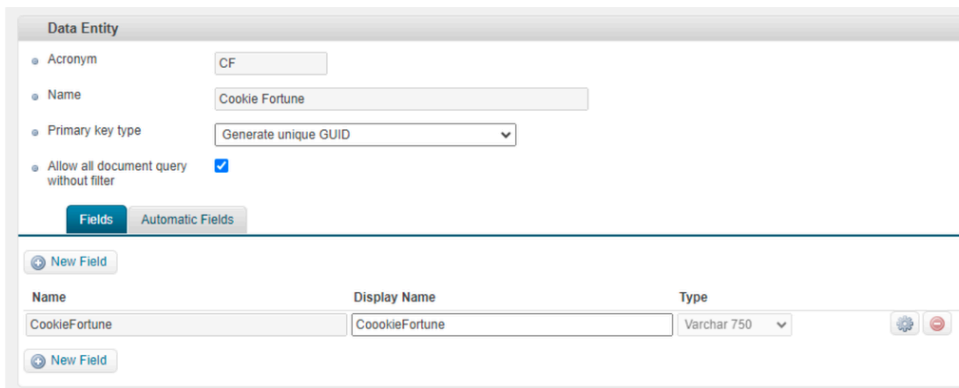
Este desafío consiste en desarrollar un conjunto de aplicaciones en el ecosistema de **VTEX IO**, utilizando **Master Data** como origen de datos principal, con el objetivo de construir una experiencia interactiva y administrable basada en “Galletas de la Fortuna”.

Objetivo General

Consumir y gestionar frases almacenadas en una entidad de datos de Master Data previamente creada. Estas frases podrán visualizarse desde una app de Store (Front-end) y administrarse desde una app de Admin, incluyendo operaciones como agregar y eliminar frases.

Información técnica clave

- La entidad de Master Data ya se encuentra creada y tiene como acrónimo: **CF** (Cookie Fortune), su estructura de datos es la siguiente:



The screenshot shows the 'Data Entity' configuration page in VTEX IO. The 'Acronym' field is set to 'CF', the 'Name' field is 'Cookie Fortune', and the 'Primary key type' is set to 'Generate unique GUID'. The 'Allow all document query without filter' checkbox is checked. Below the configuration fields, there is a table of fields. The table has three columns: 'Name', 'Display Name', and 'Type'. The first row shows 'CookieFortune' in the 'Name' column, 'CookieFortune' in the 'Display Name' column, and 'Varchar 750' in the 'Type' column. There are 'New Field' buttons at the top and bottom of the table.

| Name | Display Name | Type |
|---------------|---------------|-------------|
| CookieFortune | CookieFortune | Varchar 750 |

- Los registros actuales pueden visualizarse ingresando a la sección “**Cookie Fortune**” dentro de Master Data en el entorno: <https://valtech.myvtex.com>
- Las credenciales de acceso al ambiente de VTEX y las claves de API necesarias serán enviadas por correo electrónico.

Fase 1: App de Store (Front-end)

Se deberá desarrollar una app de Store en VTEX IO que renderice un componente interactivo de “Galleta de la Fortuna”.

Requisitos funcionales:

- El componente debe incluir:
 - Un botón para generar una frase.
 - Un encabezado <h3> para mostrar la frase obtenida.
 - Un subtítulo <h5> para mostrar un número de la suerte aleatorio.
- Comportamiento esperado:
 - Al hacer clic en el botón, se debe mostrar un **spinner de carga** mientras se obtiene la información.
 - Una vez completada la solicitud, se debe mostrar:
 - Una **frase aleatoria** proveniente de la entidad de Master Data (CF), almacenada en el estado del componente.

- Un **número de la suerte aleatorio**, con el siguiente formato: XX-XX-XXXX.
- El encabezado (h3) y el subtítulo (h5) **no deben renderizarse** hasta que haya un valor disponible en el estado.

Consideraciones de diseño: [🔗](#)

- Se valorará el uso de animaciones, microinteracciones y un diseño visual atractivo.
- El componente debe ser responsivo y permitir su integración posterior al Store Theme (Fase 2).

Buenas prácticas esperadas: [🔗](#)

- Uso de estado local con React Hooks (useState, useEffect).
- Separación clara entre lógica, presentación y estilos.
- Modularización y uso de componentes reutilizables si corresponde.

Fase 2: Maquetación de Landing en el Store Theme [🔗](#)

Se deberá maquetar una landing page en el Store Theme de VTEX Store Framework que integre el componente desarrollado en la Fase 1.

Requisitos: [🔗](#)

- Crear una landing page dedicada que incorpore el módulo de la **Fase 1** (frase + número + botón).
- Integrar el componente como un **custom block** dentro del Theme, permitiendo su uso a través del CMS de VTEX.
- Aplicar estilos personalizados mediante el store.theme utilizando CSS handles.

Consideraciones de estilo: [🔗](#)

- La maquetación debe seguir una metodología responsive, preferentemente **Mobile First**.
- Debe contemplarse una experiencia visual cuidada, con animaciones o microinteracciones que aporten valor a la interfaz.
- El diseño debe respetar las buenas prácticas de accesibilidad y jerarquía visual.

Buenas prácticas esperadas: [🔗](#)

- Separación de estilos y lógica de presentación.
- Uso adecuado de **interfaces.json** para definir el bloque.
- Uso de **store builder**, **flex-layout** o **rich-text** si fuera necesario para estructurar contenido adicional en la landing.

Fase 3: App de Admin [🔗](#)

Se deberá desarrollar una aplicación de Admin utilizando el VTEX Admin Framework que permita gestionar los registros de frases de la entidad “CF” (Cookie Fortunes).

Requisitos funcionales: [🔗](#)

- Visualizar en una tabla todos los registros actuales de frases provenientes de la entidad de Master Data.
- Incluir un formulario para agregar nuevas frases:
 - Puede implementarse como un input en la interfaz principal o mediante un modal.
- Añadir una opción para eliminar frases individualmente desde la tabla.

Requisitos técnicos: [🔗](#)

- La App debe comunicarse con la App de Servicios (Fase 4) para realizar operaciones de lectura, escritura y eliminación.
- Toda la gestión de datos debe realizarse a través de resolvers GraphQL o endpoints controlados desde la capa de servicios.

Buenas prácticas esperadas: [🔗](#)

- Interfaz clara, responsiva y coherente con el Admin de VTEX.
- Validaciones mínimas de entrada (por ejemplo, evitar campos vacíos).
- Feedback visual para las acciones (exitosas o con errores).

Fase 4: App de Servicios [🔗](#)

Se deberá desarrollar una app de servicios (Service app) que centralice y exponga todos los métodos utilizados en las fases anteriores relacionados con la interacción con Master Data y otras operaciones de backend.

Requisitos: [🔗](#)

- Debe exponer endpoints HTTP y/o resolvers GraphQL (queries y mutations) para:
 - Obtener frases disponibles.
 - Obtener una frase aleatoria.
 - Crear nuevas frases.
 - Eliminar frases.
- Las funciones implementadas en esta app serán consumidas por:
 - La Store App (fase 1 y 2).
 - La Admin App (fase 3).

Seguridad: [🔗](#)






- Todos los endpoints y resolvers deberán contar con mecanismos de autenticación y autorización apropiados.
- Se debe evitar el acceso público o no autorizado a los métodos expuestos.

Buenas prácticas esperadas: [🔗](#)




- Modularización clara del código (service, routes, resolvers, middlewares, etc.).
- Manejo adecuado de errores y validaciones.
- Documentación mínima de los endpoints y resolvers expuestos.

Criterios de Evaluación [🔗](#)

Durante la revisión del challenge, se tendrán en cuenta los siguientes aspectos:

-  **Documentación:** Claridad, completitud y calidad de la documentación entregada sobre el desarrollo realizado.
-  **Buenas prácticas en VTEX IO:** Uso correcto de tecnologías y convenciones del ecosistema VTEX, incluyendo Store Framework, IO Services, Master Data y GraphQL.
-  **Calidad de la maquetación:** Organización visual, uso de estilos mobile-first y cumplimiento de criterios de diseño responsive y accesibilidad.
-  **Componentización:** Modularización efectiva de las apps (Store, Admin y Services), reutilización de componentes, y claridad en la estructura del código.
-  **Consumo de recursos de backend:** Forma en la que las apps interactúan con la capa de servicios (Master Data y servicios propios), incluyendo control de errores y performance.

Qué esperamos como entrega [↗](#)

-  **Repositorio en GitHub** con el proyecto completo y su historial de commits.
-  **Workspace de VTEX IO** donde se haya desarrollado y probado el proyecto.
-  **Documentación técnica** que consideres relevante, incluyendo:
 - Instrucciones de instalación y despliegue.
 - Descripción general de la arquitectura.
 - Detalles sobre endpoints o integraciones realizadas.