CMS

Purpose

CMS is a web application designed to make it easy for non-technical users to add, edit and manage a website without programming or web hosting experience. Although Drupal has the functionality of a traditional CMS, we will be utilizing it as a headless CMS:

	Traditional CMS	Headless CMS
Platform independence	The back end is connected with Frontend.	The back end is run independently.
Security	ReqA traditional web server is easy to hack.	Backend issues don't affect the frontend.
Performance	A traditional CMS is slower.	It is designed to be fast.
Costs	Costs less when you include hosting.	More cost-effective (no hosting costs).
Customization	Difficult to customize and has limitations on flexibility.	Easy to customize and very flexible.
Scalability	Hard to scale.	Easy to share through different platforms.

Advantages

- Allowing non-technical users to easily update your own content
- Accommodate regularly changing content rather than have developers maintain these changes

Tradeoffs

- Requires higher level of maintenance
- Can complicate code and introduce unnecessary overhead and functionality

Use Case Evaluation

List of current supported CMS capabilities

Component/Area Proposed by Globe	Feature Instances	LF Proposal	Considerations
App Links	T&Cs Privacy Policy Globe Channels	Hardcode on app	1. These links shouldn't change that much 2. These will also need to be zero-rated, so if they're also in the CMS, this may cause inconsistencies with the links that are zero-rated on the backend
Spiels	Registration/Login Page Mobile OTP Account Enrollment	Hardcode on app	1. This text shouldn't change that much as it's better to have consistency with the user 2. Can't guarantee that the UI look and layout will be consistent with varying text lengths
Button Labels	Registration/Login Buttons	Hardcode on app	1. This text shouldn't change that much as it's better to have consistency with the user and registration buttons should follow the guidelines of the third party companies
Emails	Email Sender	Set on backend or CMS	1. This has no impact to client build, so this is more up to Globe
Interests	Categories	CMS	1. This depends on how the information is being set on the backend for rewards. If updating the configuration in CMS will update the values in the rewards platform correspondingly, I think this should be fine to have be CMS-able.

WalkMe	Onboarding Missions	WalkMe Integration	1. This depends on whether we are doing the integration with WalkMe, but likely they will integrate directly with the APIs and the SDK would handle the rendering of the UI corresponding to the configurations
Notifications	Reminders for Unfinished Onboarding Missions Promo Expiration Unactivated Offer/Freebie Segmentation	CMS	1. Some triggers will be on the backend, would this be set up by LF or Globe?
Promos	Display Details for Promos Discounts on Promos Usage Restriction per Campaign Categorization Search Offer External Links Mapping to Interests Banner Update	CMS	1. I think we can configure this information in the CMS, but this would be with the assumption that what is configured impacts what is stored on the backend for CXS services to ensure that when a promo is selected, those service details are provisioned for the user and charging orchestrations are also kicked off.
Load	Denominations of Load Offer Name of Load Offer Description of Load Offer Validity of Load Offer Inclusions of Load Offer Active Promotions Prerequisite Promos	CMS	1. I think we can configure this information in the CMS, but this would be with the assumption that what is configured impacts what is stored on the backend for CXS services to ensure that when a load is purchased, charging orchestrations are kicked off and the wallet is filled accordingly.
Discover	Globe Lifestyle Content (Articles, videos, quizzes, shop items) External Links Content Categories Search	CMS	1. Need to ensure that shop items is properly connected on the back end if it results in a specific purchase
Refund	Refund Load/Promo	?	1. Need more information on how we anticipate this to be in CMS. Are we thinking that for every configured promo, there would be a refund component that would allow an endpoint to be called?
App Alerts	Zero Rating App Maintenance New App Version Availability/Non-Availability of Transactions	CMS	1. I think we would need to determine the potential use cases and build the triggers for these beforehand so the app knows how to handle information that is passed over