BEAMABLE

Game Economy

March 8, 2021

Product Overview

Beamable provides rich game economy features that are designed to integrate into game developer workflows naturally, whether those are in the Unity Editor, in a Google Spreadsheet or on the web.

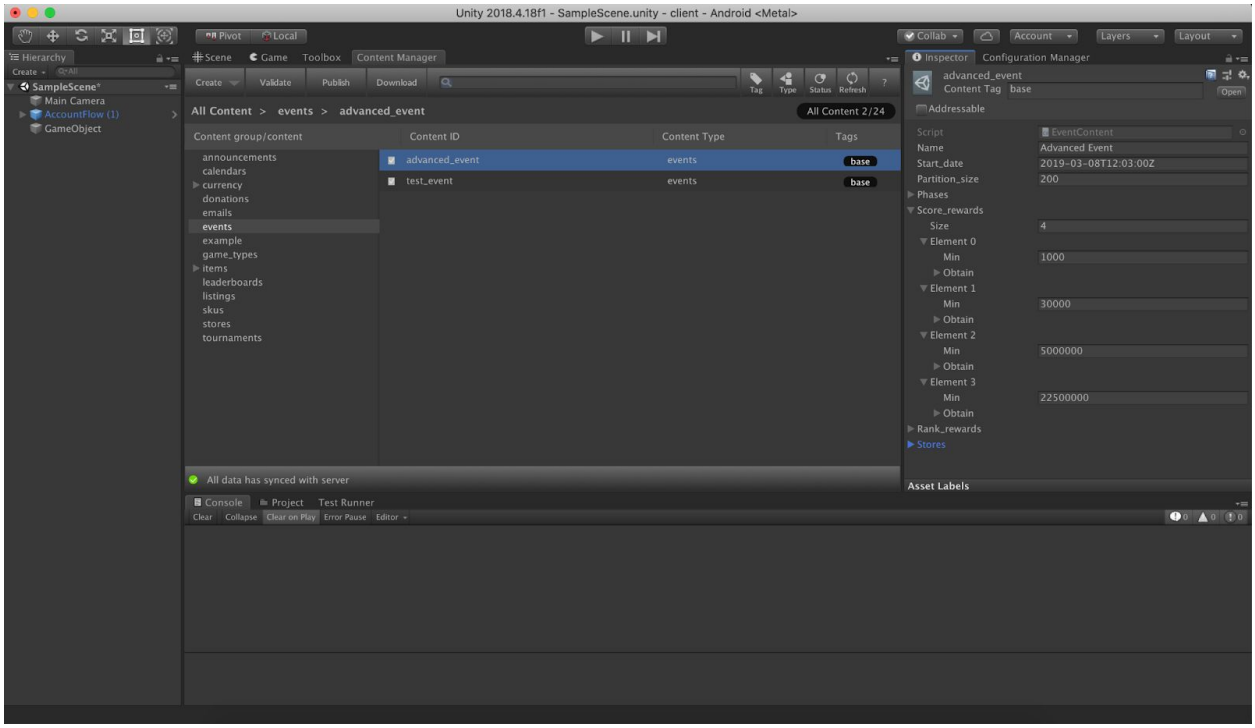
Content Authoring & Live Operations

Content Authoring

Having released a few games ourselves, we understand how much of a burden it can be on your designers to actually author new and exciting content into your game. This is exactly why we have built a few tools to make the content production process much easier.

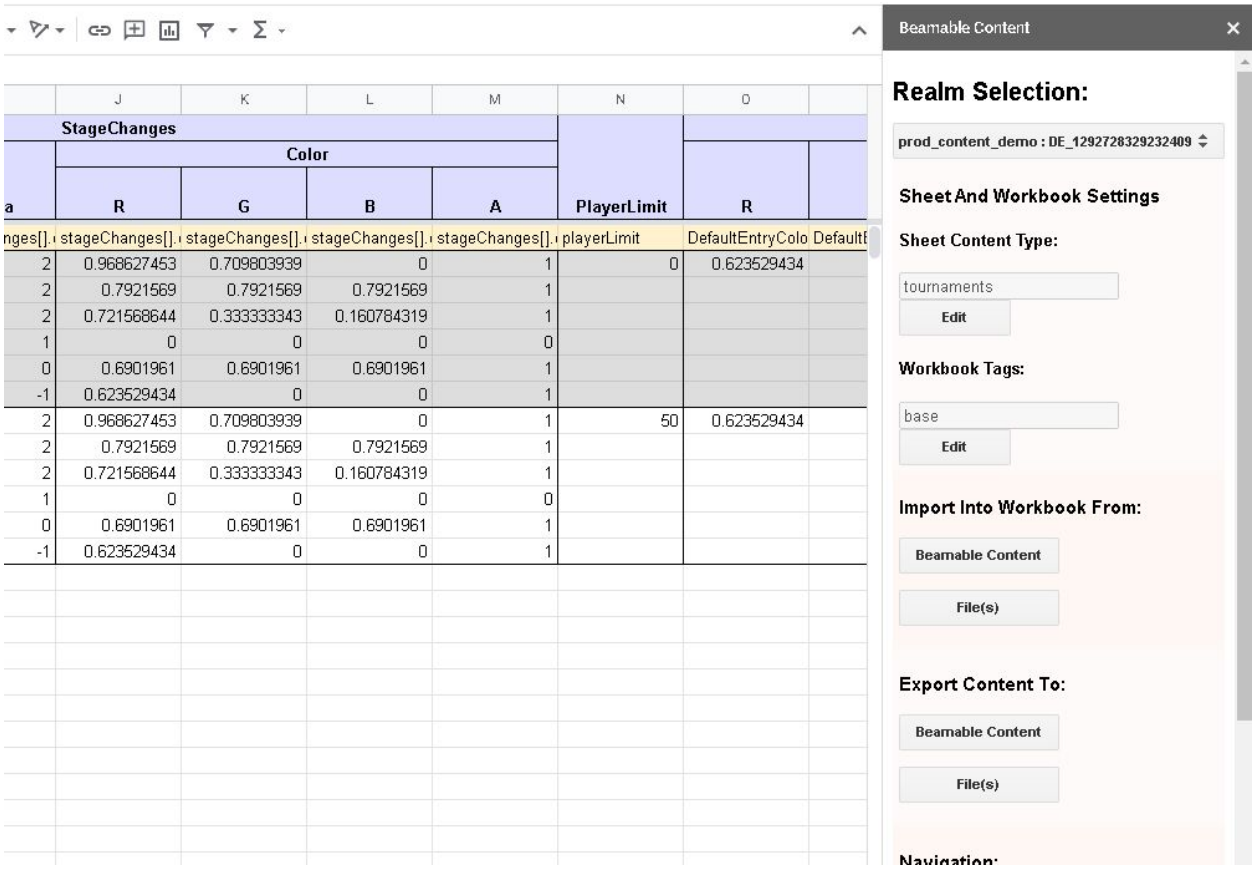
Content Creation

One of the things we found out early on is that proper tooling is necessary to produce high quality content at a high velocity. Beamable helps you with this by placing all of the necessary tools to both edit and publish content directly in the Unity editor.



As you can see, all of the content types are specified in the left-hand column and selecting a particular piece of content in the editor window loads the values for that piece of content in the inspector.

However, we do know that not all designers like to use the same workflow to author content and we have also realized how much designers love spreadsheets so we provide a Google Sheets plugin as well as the Unity Editor plugin to support that desire.



	J	K	L	M	N	O
	StageChanges					
	Color					
a	R	G	B	A	PlayerLimit	R
nges[].	stageChanges[].	stageChanges[].	stageChanges[].	stageChanges[].	playerLimit	DefaultEntryColo
2	0.968627453	0.709803939	0	1	0	0.623529434
2	0.7921569	0.7921569	0.7921569	1		
2	0.721568644	0.333333343	0.160784319	1		
1	0	0	0	0		
0	0.6901961	0.6901961	0.6901961	1		
-1	0.623529434	0	0	1		
2	0.968627453	0.709803939	0	1	50	0.623529434
2	0.7921569	0.7921569	0.7921569	1		
2	0.721568644	0.333333343	0.160784319	1		
1	0	0	0	0		
0	0.6901961	0.6901961	0.6901961	1		
-1	0.623529434	0	0	1		

Lastly, our content system is driven by a REST API so if neither of those solutions are sufficient, all of the necessary tools to build your own editing solution are available to you.

Content Publishing

Once your content is “ready to ship”, Beamable has you covered there as well. When you create your Beamable account, we automatically create a “Development to Production” pipeline of distinct environments for you. These environments are “Dev” -> “Staging” -> “Production”. This is how we enable you to “publish” your content to our servers for the development environment while allowing you the piece of mind to know that the “Production” environment has not been modified.

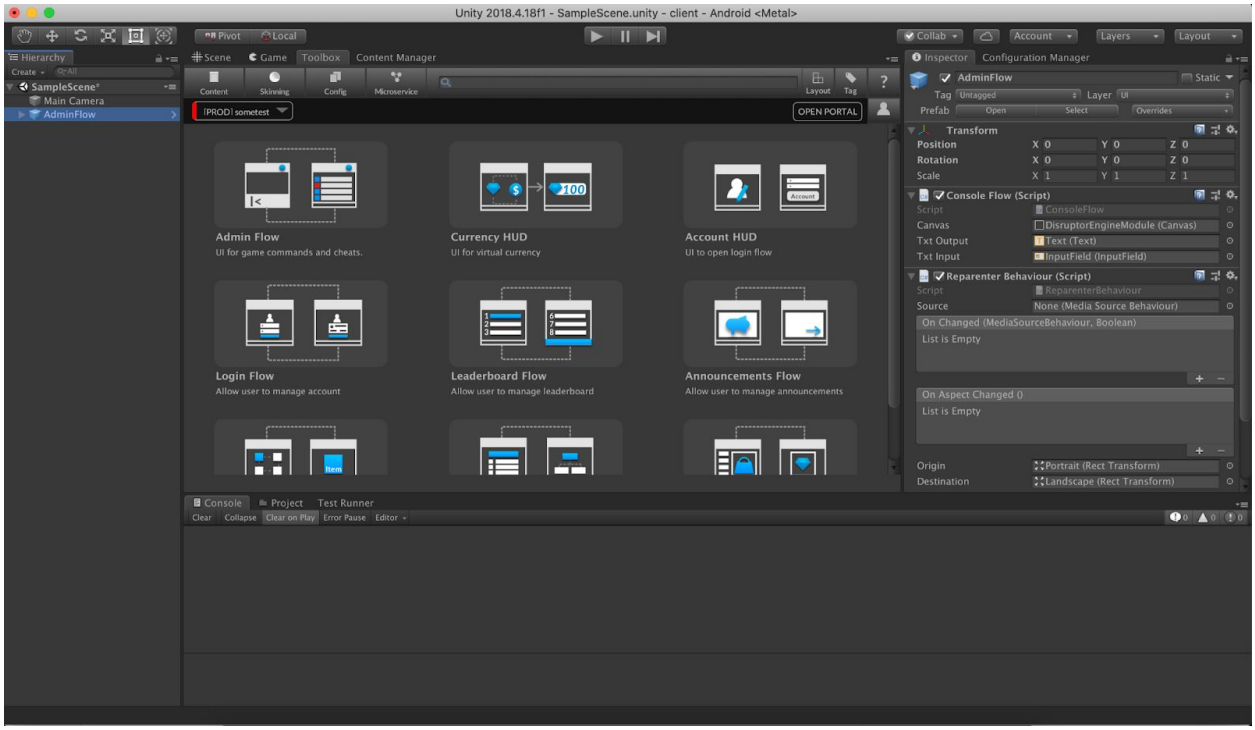
Then, once you have tested that the new content in your environment looks correct, you can go into the admin portal to promote the Dev content to Staging and eventually Production.

Live Refresh

One of the main benefits of our content system is that, when you hit publish and the content updates, we will automatically refresh that content on each of the game clients via a server-to-client message. This allows for the system to be incredibly interactive at development time as well as allowing for over the air updates to players who are playing in “Production” after a content promotion.

Testing

In order to help facilitate testing, we offer a number of in-game “cheats” that can be accessed by dragging in the “Admin Flow” to your games Scene hierarchy.



Here, you will find a number of cheats including but not limited to a very useful “SET TIME” cheat that allows your client to be able to make a request as if the time it were making the request is

the one specified rather than the actual time. This is incredibly useful for offer testing as well as a number of our other time-based services.

Localization

Our content service also provides text localization based on keys. This allows you to say that for the “greeting_dialog”, you would like the “French” version of that text when the content is returned to the client.

Live Operations

Portal & player administration

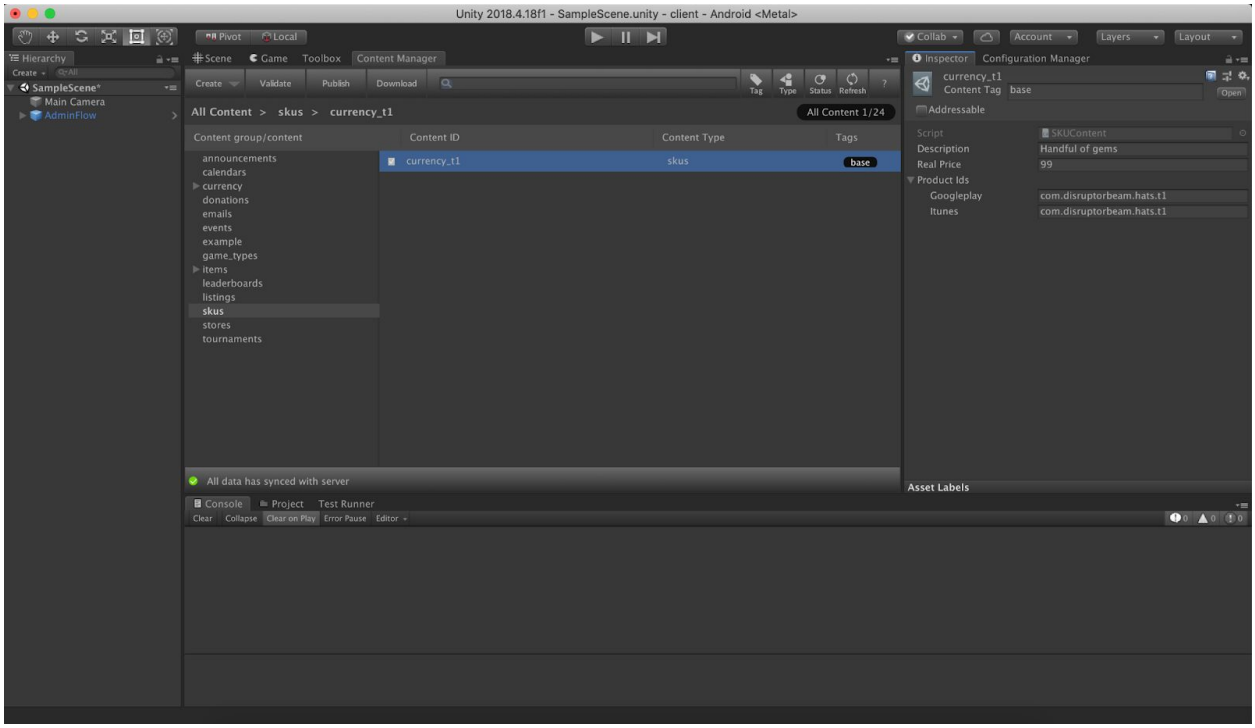
Once your game is in Production and you have a healthy set of players who love your game, you’re likely going to run into the problem that certain players will not be happy. Unfortunately, this is a fact of life for every successful game. You just can’t please everyone. Fortunately for you, you have Beamable’s player administration portal to help you find problems and deliver solutions. In the tool, we provide a number of features.

Payment Gateways

Beamable covers a number of in-app purchase use cases across a number of stores for virtual goods. Specifically, we support receipt verification for

- Google Play
- Itunes Store
- Facebook Payments
- Steam
- Windows Store

SKU Management



Instead of having to have separate configuration and code paths for each individual payment provider, we have created the idea of a “Beamable SKU” which collates all the interesting information about a provider’s products across stores into one structure. Once you have that content defined, our commerce system can use the identity of that content as the “price” for a particular item. Our services will use this content in verifying receipts sent to us from clients.

Receipt Validation

As previously mentioned, Beamable provides robust receipt verification for each of the providers listed. We, over the course of many years, have developed the experience and code necessary to ensure that only authentic, legitimate receipts for items in your games will be granted to the player and more importantly, we ensure that they are only granted once.

Hacking/Fraud Detection and Replay Guard

As you are probably aware, forging receipts or replaying previously redeemed receipts is a common cheat attempt in gaming to get an advantage upon other players as well as an attempt to not pay any money to your company. Fortunately, due to our payment verification system, we have the ability to detect and prevent fraudulent receipts before they wreak havoc on your player economy.

Coupon Redemption

One other thing worth mentioning is that we also have the ability for you to grant “coupons” to your players which can end up being used to purchase a particular offer of your choosing. This can be a great way to give your players free stuff to drive even more engagement.

Offer Behavior

Managed Content

All of the configuration for our offer system utilizes our content system, thus enabling you to update the store information as needed.

Offer Scheduling

Beamable allows you the ability to schedule particular offers in a store to be available to players starting at a particular date and ending on another date. For example, players will only see a Christmas special offer only during the week of Christmas." Pairing this to the previously mentioned "SET TIME" cheat and you can be sure that what will be delivered to players is what you'd expect.

Limited Time/Duration Offers

These are very similar to scheduled offers but they are player specific. For example, when a player first sees an offer, that player has N minutes to purchase before the offer is no longer available. This creates a sense of urgency to make sure to "get it before it's too late!"

Recurring offers

For the limited time offers, you can also specify a "cooldown" period until the offer becomes active again. For example, after seeing the super special offer but not purchasing, that player will not see the offer again until 5 days have gone.

Offer Text Localization

We offer the ability to specify different translations for offer text that are fetched by making a request with the appropriate 2 letter language code. For example, I, as a player, live in France therefore I naturally would like to see all of the text of the offer in French.

Show/Hide inactive listings

The backend API request can be configured to only return the offers active for this player to the client, or you can return all of the offers visible to the player yet not active.

Support for real money and virtual currency transactions

When specifying offers, you can either provide a virtual currency type plus amount OR you can specify a store provider SKU which would trigger a real money purchase flow. This allows you to be able to merely specify “price point” SKUs in Google Play or the iTunes store which we will use to grant the required items on our side. For example, in the Google Play developer console, we can specify a ‘product’ called ‘com.x.sku.5dollar’ and we can configure that with Beamable to be the “price” of a particular store listing.

Offer Price Schedule Increases

We allow you to specify a price schedule based on the number of purchases of a particular offer. For example, every time a player makes a purchase I want the price of the item to increase. It will start at 2. Then, after a purchase, it will become 3 then 5 then 20.

Store Offer Limits

We allow you to specify how many active offers are returned to the client. This is especially useful if a particular store has a large number of simultaneous active offers for a player. For example, A player is only able to see 1 special offer at a time but the special offer store has 4 special offers configured for the player which are active. In order to prevent various stateful timers to start ticking, you can restrict the API to only return the first active offer which ensures that the other offers are in a “not viewed” state.

Evented/real-time updating of the store

Any change made to the content of the stores will trigger the service to send an update message to the client prompting it to fetch the most recent commerce data.

API driven player state cleanup

We also have the ability to programmatically reset player status for a store via our API. This is useful if you want to allow a particular player to rebuy an offer or if you wanted to reuse the id of an offer for the future. For example, we made a mistake and released an offer that didn’t have enough reward specified. Rather than removing items or updating the offer without allowing the

players who already purchased to purchase again, we can clear all the player's state prior to releasing the fixed offer which would allow those players to purchase this particular offer again.

Segmentation

In addition to the features specified, we have a number of ways to control who gets to see any particular offer and who does not.

Player Stat based purchase requirements

For any offer, you can configure whether or not a player can get to see the offer based on a stat that the client or server can set for this player. For example, a player can only purchase this offer if that player's level is greater than level 5.

Offer-based purchase requirements

We provide the ability to specify that an offer can only activate for a player after they have purchased another offer. For example, after buying offer X 4 times, we will make the more expensive offer Y available.

Cohort based purchase requirements

Utilizing our Trials system, you can specify that a particular cohort/segment of players get an offer whereas another group or cohort will get a different offer. Example: "Players in Group A will receive the "very fancy" offer whereas players in Group B will receive the "splendid" offer."

Enforced Purchase Limits

For any particular offer, you can specify how many times this player has the ability to purchase it. We ensure this at the backend API level. For example, a player may only buy the "Extremely OP" offer once and never again.

Analytics

Commerce features automatic instrumentation of a variety of *analytic events* and *player stats*. These provide a view into historical and aggregate key performance indicators, as well as a vector for targeting players respectively.

Analytics Events

For a full list of analytics events that are automatically instrumented, visit our docs website here: Analytics (beamable.com).

Player Stats

Player stats offer both a way to store simple key/value pairs about a particular player, as well as a convenient way to target said player with offers. Additionally, a variety of services automatically set player stats, such as but not limited to:

- Total Player Purchases
- Total Player Spend
- Purchases in the last 1 day/3 days/7 days/14 days/28 days
- Spend (LTV) in the last 1 day/3 days/7 days/14 days/28 days

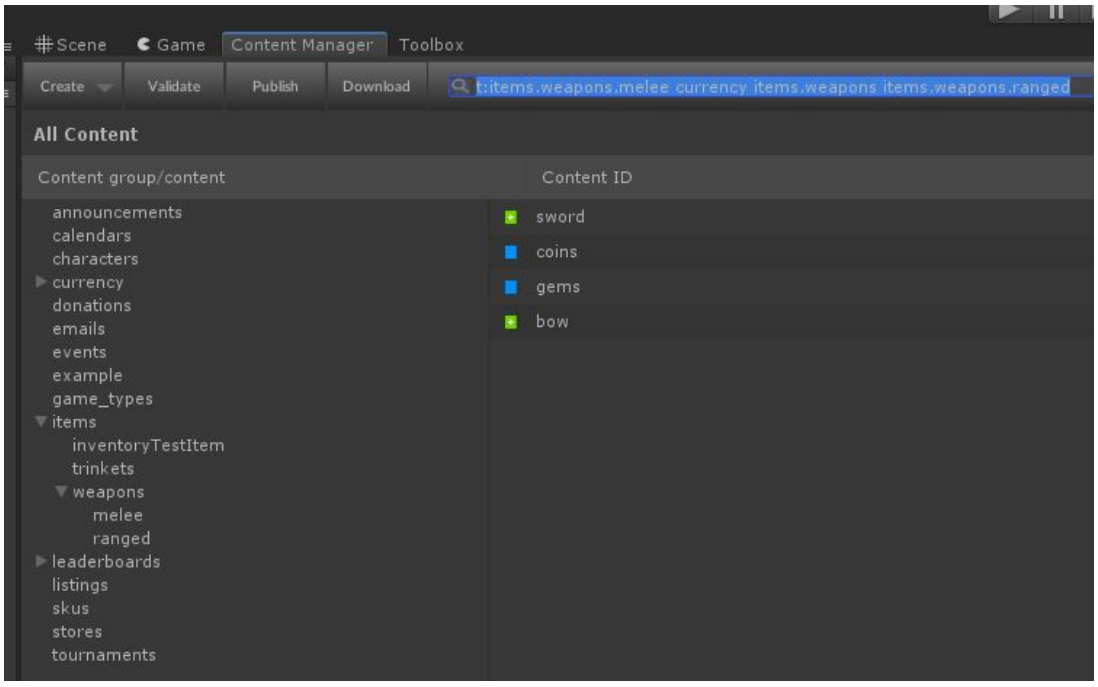
This allows developers to configure discount or upsell offers without having to track the past data themselves.

Inventory Integration

Beamable services natively integrate with the player inventory system, where they track statefulness. The inventory service tracks two types of grantables: items and currencies, which are definable in content.

Virtual Currencies

These are essentially items that can be granted to players which are represented by a symbol (currency id) and an amount (64 bit integer). They can be granted by services such as Commerce offers, Announcements, In-Game Mail, and many more. They are inherently fungible, and can be created at will with the Content system.



Items

These can also be specified Unlike currencies, items are inherent *non-fungible*. This means each instance of an item can hold dynamic properties unique to that particular item instance. Also, item content can be subclassed to attach a variety of static properties that are universal to that item type.

Live Events

Live events can have a significant multiplier effect on engagement and revenue. Beamable offers two out of the box features that support this.

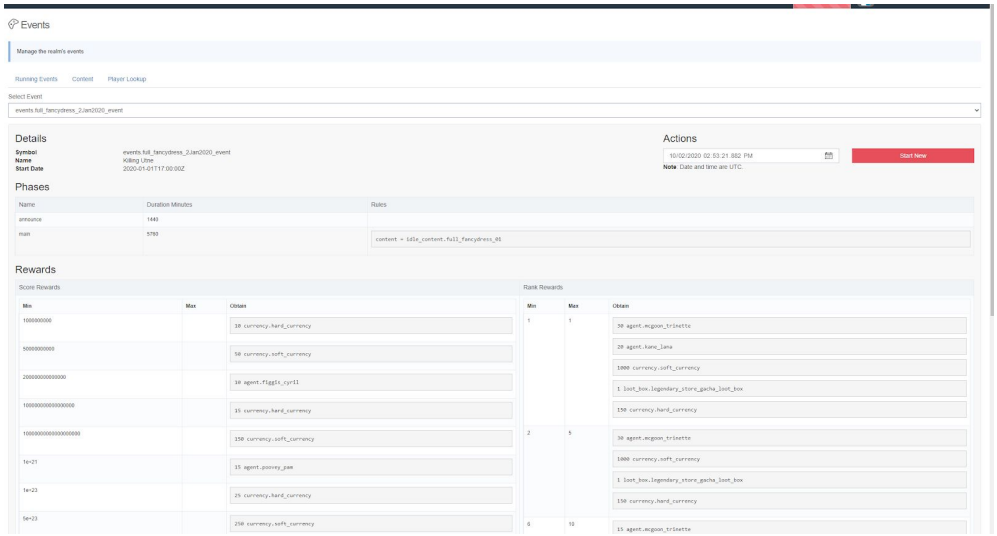
Events

A social/competitive feature that allows players to compete over a finite duration, where their in-game actions contribute a score to an event leaderboard. Events can have one or more phases, each with a defined duration. These phases can be interpreted by the game to mean whatever it wants.

You can also have multiple phases which map to different forms of gameplay. Phase I could be "collect gems", whereas Phase II could be "collect swords" -- these actions ultimately contribute to a single score on a leaderboard.

Lastly, Events deliver rewards on the basis of score and rank.

- Score: As players reach certain score thresholds, they can unlock rewards during the event. This encourages engagement and delivers rewards sooner than the end of the event.
- Rank: When the event ends, players can collect rewards on the basis of their final rank on the leaderboard.



Tournaments

A social/competitive feature, but with an additional layer of meta game and progression. Unlike Events, Tournaments are cyclical. This means that you compete on a periodic basis, and the results of that cycle feed into progression into new Stages and Tiers (i.e. Leagues).

An example Tournament might have:

- A start time at 12pm every day -- this means that the tournament cycles daily
- 5 defined Tiers (e.g. Bronze, Silver, Gold, Platinum, Diamond)
- 10 defined Stages each Tier (you can control how many stages for each tier granularly)
- Ranks 1-5 ranks advance 2 Stages, Ranks 6-10 advance 1 Stage
- Ranks 46-50 ranks regress 2 Stages, Ranks 40-45 regress 1 Stage
- Rank rewards for positions or ranges in the leaderboard

Progression/regression rules which determine whether you advance to the next stage/tier, regress to the previous stage/tier, or stay in the same stage/tier based on your rank in the previous cycle.

As you climb in Stages and Tiers, the competition would get fiercer -- but so too could the rewards. This can recur indefinitely, or you can shut down the tournament after a period of time. Players can claim their rewards at all times.

Lastly, unlike Events, Tournaments have three leaderboards that you can show up on:

- Your partitioned leaderboard by Stage/Tier
- A High Scores leaderboard, for who scored the highest across all Stages/Tiers
- A 30-day rolling view of the high scores winners

You can unlock rewards on these leaderboards based on your rank position


DAILY
HIGH SCORE
CHAMPIONS
5
Stage 6/10 1h Left
TODAY
YESTERDAY
1 Rocky_0 1M 123,400
2 Bullwinkle_100 995K 122,265
3 Natasha_200 985K 121,128
4 Boris_300 975K 119,989
5 Fearless Leader_400 965K 118,848
6 Narrator_500 955K 117,705
26 Bullwinkle_2500 94,425