

Upgrading to brainCloud from PlayFab

Player and Player Data



Table of Contents

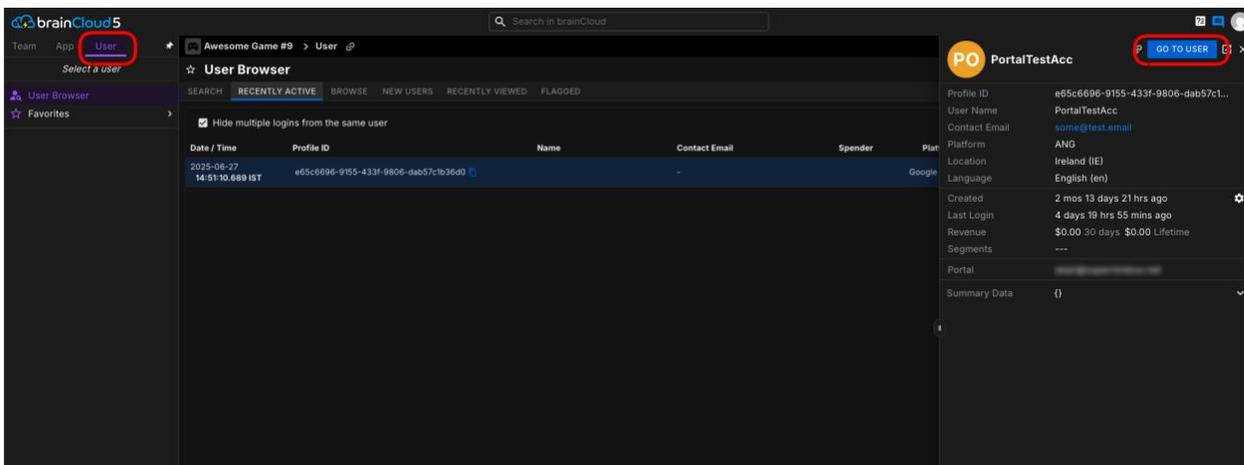
Introduction.....	3
Player Accounts.....	3
Player Data	5
Achievements.....	5
Inventory	6
Multiplayer Session	6
Social	7
Player Logs.....	7
Searching Player Records	8
Getting Player Account Details	10
Get Player State.....	11
Authentication Response	12

Introduction

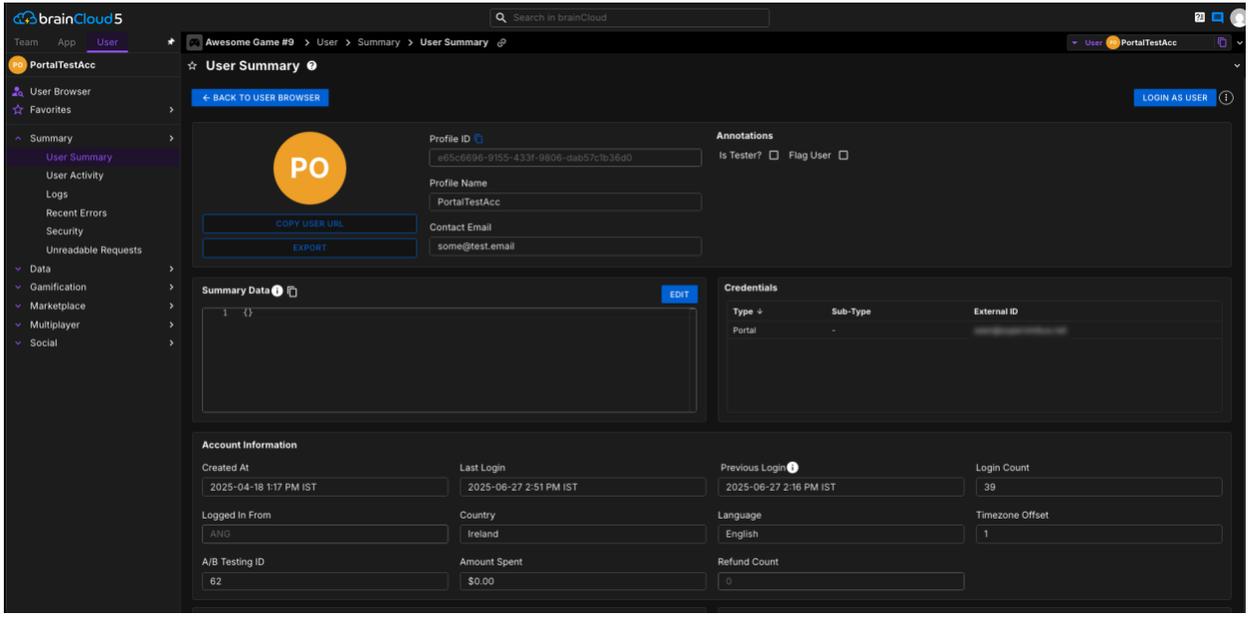
In the previous guide, we looked at authentication as the first step in migrating your game to work with brainCloud. In this guide, we will explore how to work with player data through the brainCloud portal. This is important for debugging during your migration process, as the tools and dashboards covered here can show if features attributed to player accounts are stored and updated correctly.

Player Accounts

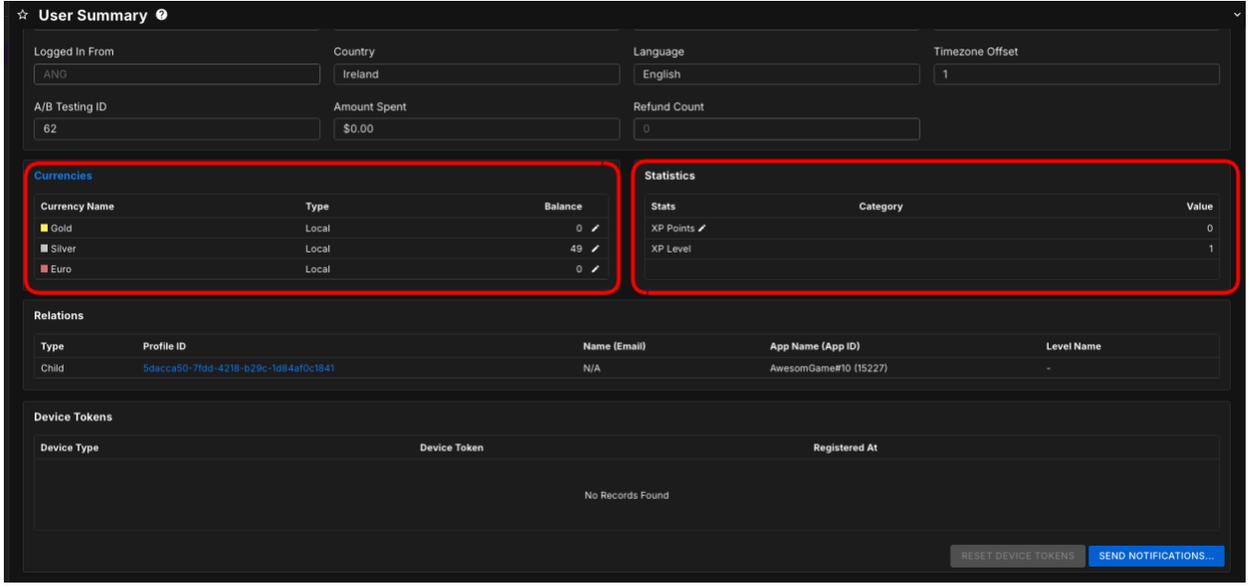
Once you have successfully authenticated a player, their account can be viewed from the brainCloud dashboard by going to the **User → User Browser** menu. This dashboard automatically shows the most recently authenticated players, so the player you just authenticated should appear at the top of the list. Clicking on this player displays a summary of their profile in a panel on the right-hand side of the dashboard. Click on the **Go To User** button to view their full profile.



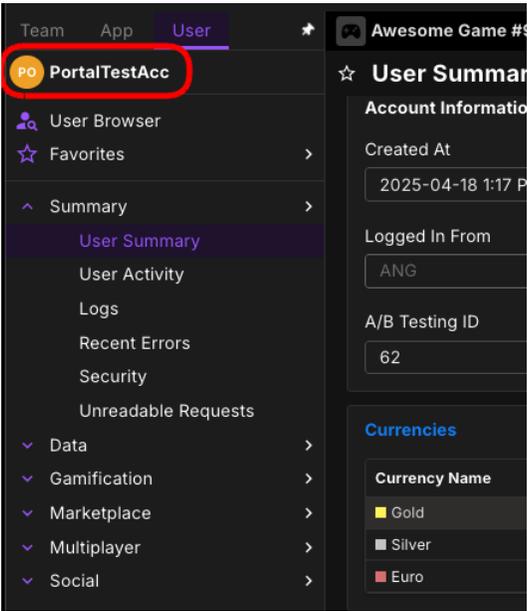
This will bring you to the User Summary dashboard, which displays basic information about the player, such as their ID, name, email, and login details.



Scrolling down this page you can view and edit the player’s virtual currency (covered in our [Virtual Currency Guide](#) and [Leaderboards and Statistics Guide](#)).



However, you will also see new options on the left-hand menu of the dashboard, with the name of the current player displayed at the top. This indicates that all elements in those sub-menus have context of this player.

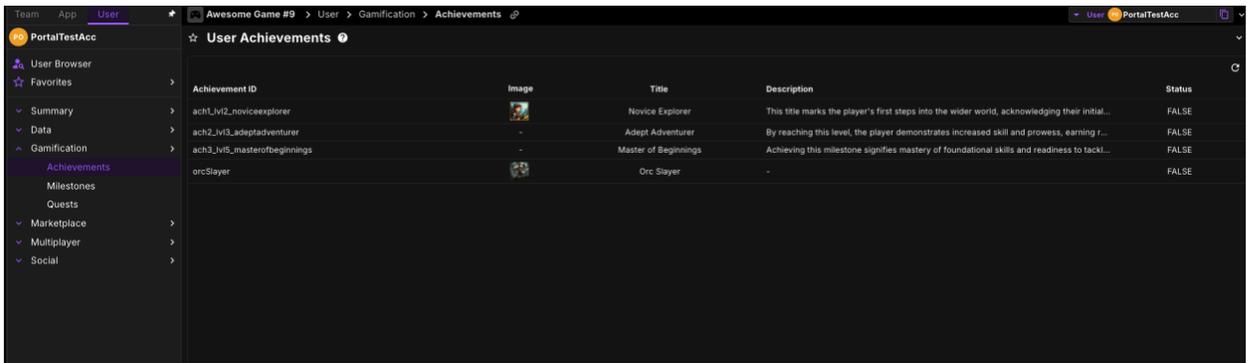


Player Data

The **User → Data** menus contain custom information about players, such as [Leaderboards and Statistics](#) and [Title Content, UGC and Files](#). These features are explored in the linked guides respectively, where we compared each feature to their PlayFab equivalent.

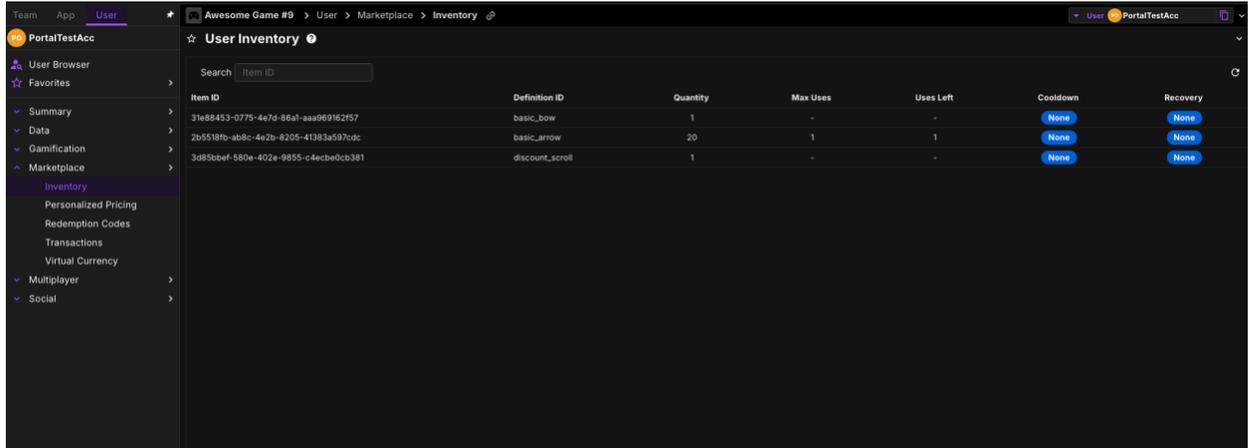
Achievements

The Gamification menu contains details about the players achievements, milestones and quests. These features are explored in our guides on [Leaderboards and Statistics](#) and [Virtual Currency](#).



Inventory

The Marketplace menu contains dashboards for exploring and editing the player's items. This is useful for debugging inventory, item delivery features, or custom cloud-code. We explore this in our guides on [Economy & Catalogs](#).

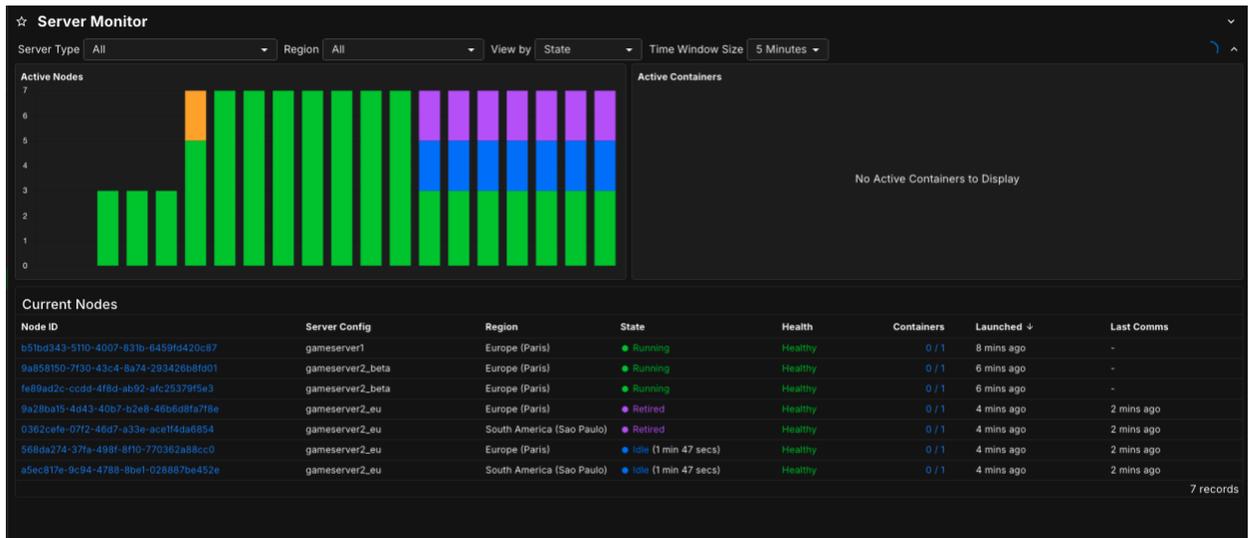


You can also see options for tracking and reviewing transactions and currency over time, as well as viewing a player's redemption codes, a feature similar to PlayFab's Coupons which is explored in this guide here <link>.



Multiplayer Session

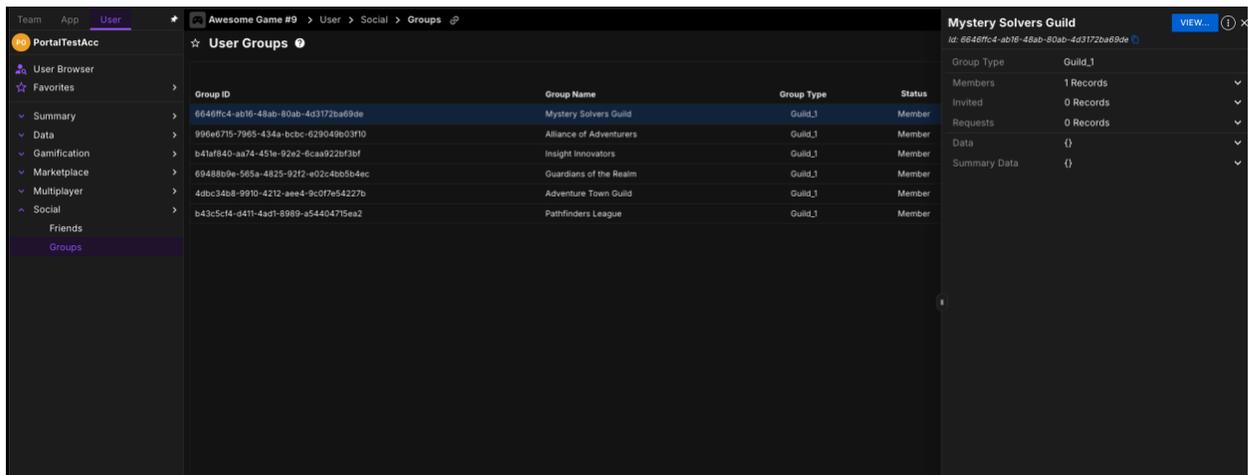
The Multiplayer menu displays records of a player's multiplayer sessions.



We go into more detail on multiplayer and servers in our guides on [Matchmaking](#) and [Servers](#).

Social

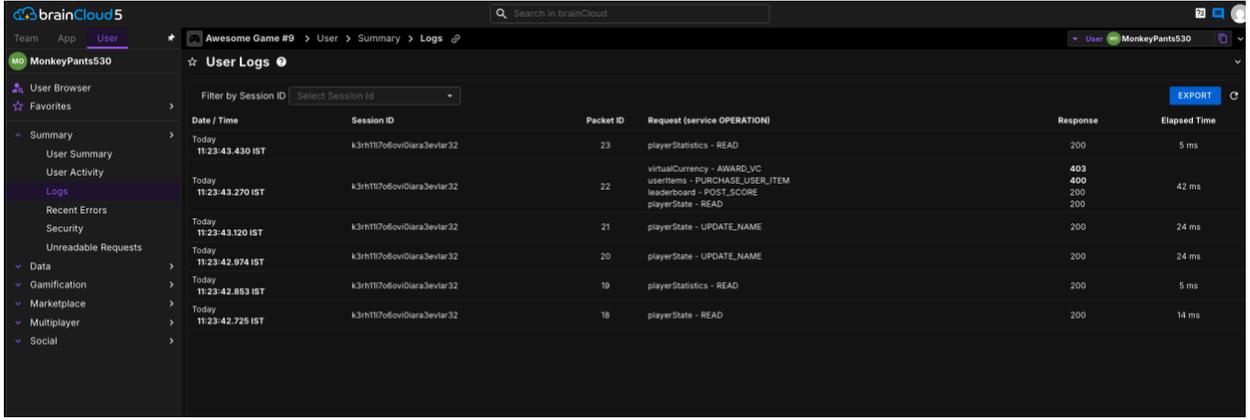
The Social menu contains dashboards for the player's [Groups, Friends and Messages](#), allowing you to click on these entities to view a summary or access other dashboards with the friend's player profile or the group's configuration details.



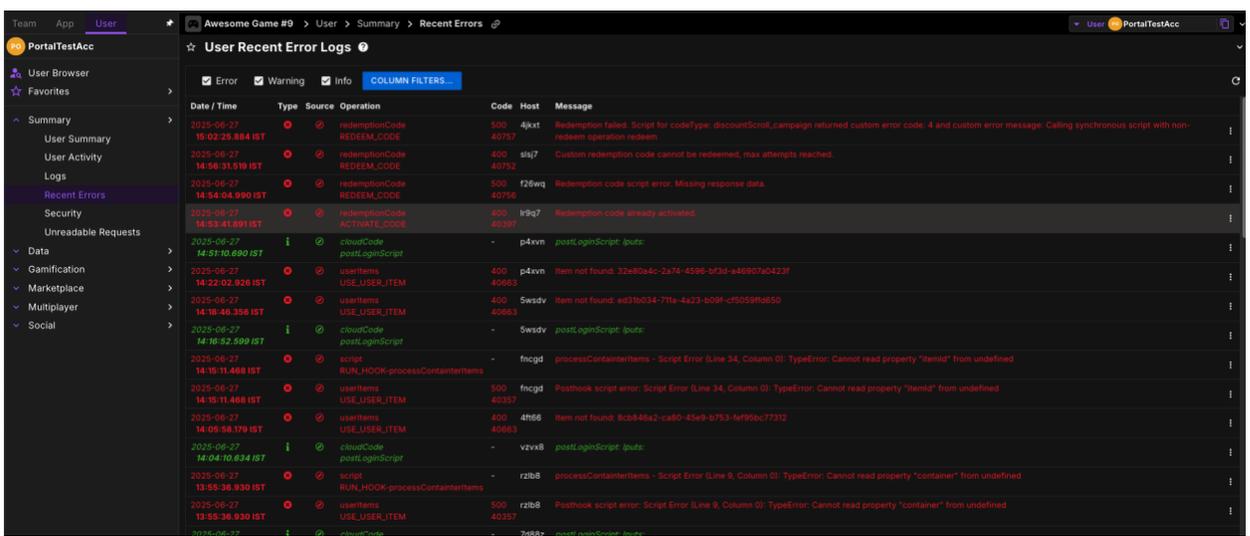
Player Logs

We skipped over an important section in the Summary menu we saw above, but it can be very useful during your migration. The Logs and Recent Errors menus are invaluable for debugging.

The Logs dashboard shows a record of all client calls and the specific APIs accessed in those calls. Note that this will only show client calls, so any debugging done through the portal's API Explorer will not be logged here.

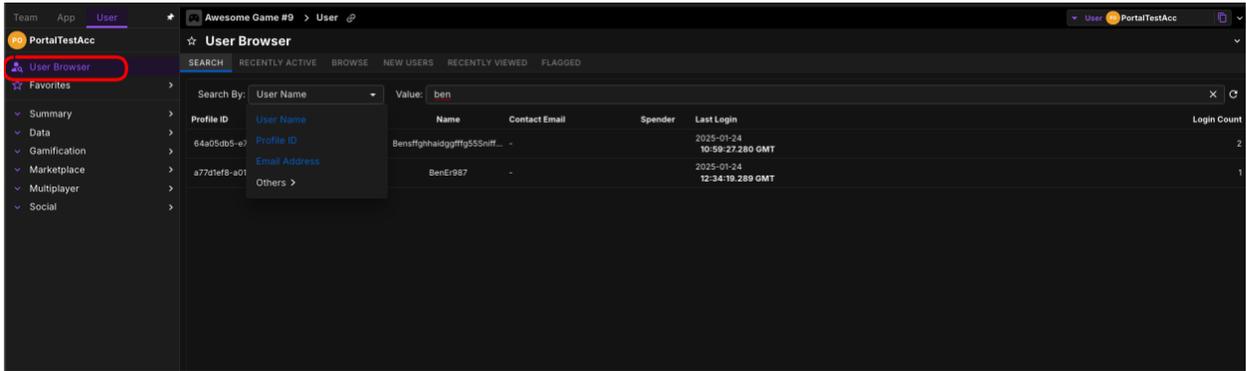


The Recent Errors dashboard actually shows errors, warnings and info logs and can be filtered to search for specific logs.

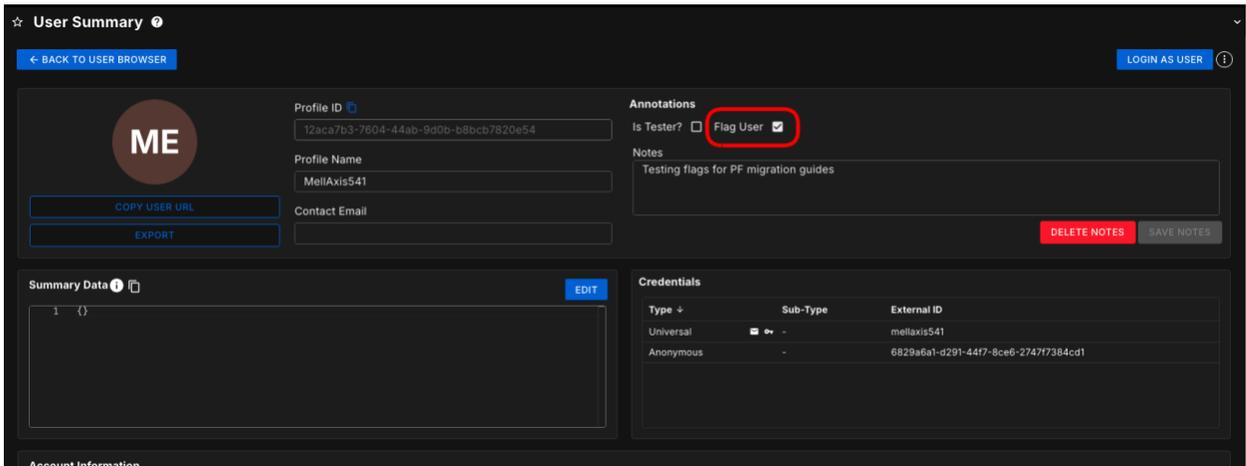


Searching Player Records

In the previous section, we looked at the Recently Active tab of the User Browser, but there are other options. Choosing the Search tab allows you to search using various player attributes such as name, email, and profile ID. It also supports partial string searches.

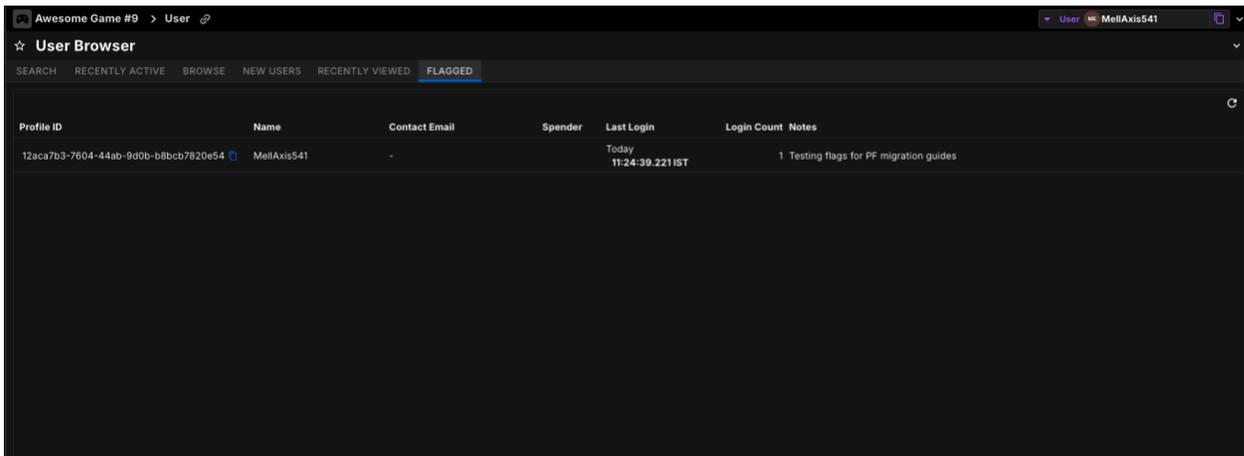


The New Users and Recently Viewed tabs are useful for debugging, as you'll always find players you've been testing with in these lists. The Flagged list is also helpful and is managed manually by selecting a player to view their summary and clicking the Flag User box.



This player will then appear in the Flagged list along with notes on why the player was flagged.

 Player and Player Data	Version: 1.0
	Issue Date: 2026-02-20



Getting Player Account Details

We've looked at viewing player details via the brainCloud portal dashboards, but you'll also need to get this information back to the client to process data for your game. There are various ways to do this for different player information.

Returning specific player information such as [Virtual Currency](#), [Player Inventory](#), [Leaderboards and Statistics](#), and [Groups, Friends and Messages](#) is covered in their respective guides. For now, we'll cover returning basic summary player information.

We'll use brainCloud's API Explorer to demonstrate examples of these APIs and their responses. More information about the API Explorer is [Cloud Scripts](#).

Note that brainCloud also allows you to log into the API Explorer as if you were a player connecting from the client or app. If you open the player summary dashboard for any player, you can click the **Login As User** button to open the API Explorer dashboard and automatically log in as that user.

User Summary

← BACK TO USER BROWSER LOGIN AS USER

Profile ID: e65c6696-9155-433f-9806-dab57c1b36d0

Profile Name: PortalTestAcc

Contact Email: some@test.email

Annotations: Is Tester? Flag User

Summary Data: 1 {}

Type	Sub-Type	External ID
Portal	-	sean@supernimbus.net

Account Information

Created At	Last Login	Previous Login	Login Count
2025-04-18 1:17 PM IST	2025-06-27 2:51 PM IST	2025-06-27 2:16 PM IST	39
Logged In From	Country	Language	Timezone Offset
ANG	Ireland	English	1

Get Player State

To get a general summary of the player you can use [ReadUserState](#) which would be similar to PlayFab's GetUserAccountInfo API. You can find this under the PlayerState service.

API Explorer

Summary

Profile ID: e65c6696-9155-433f-9806-dab57c1b36d0
Session ID: nqb44qcus109hfrum1sgovcpb
API Domain: api.braincloudservers.com

Request #1: playerState--READ
Response: 15ms

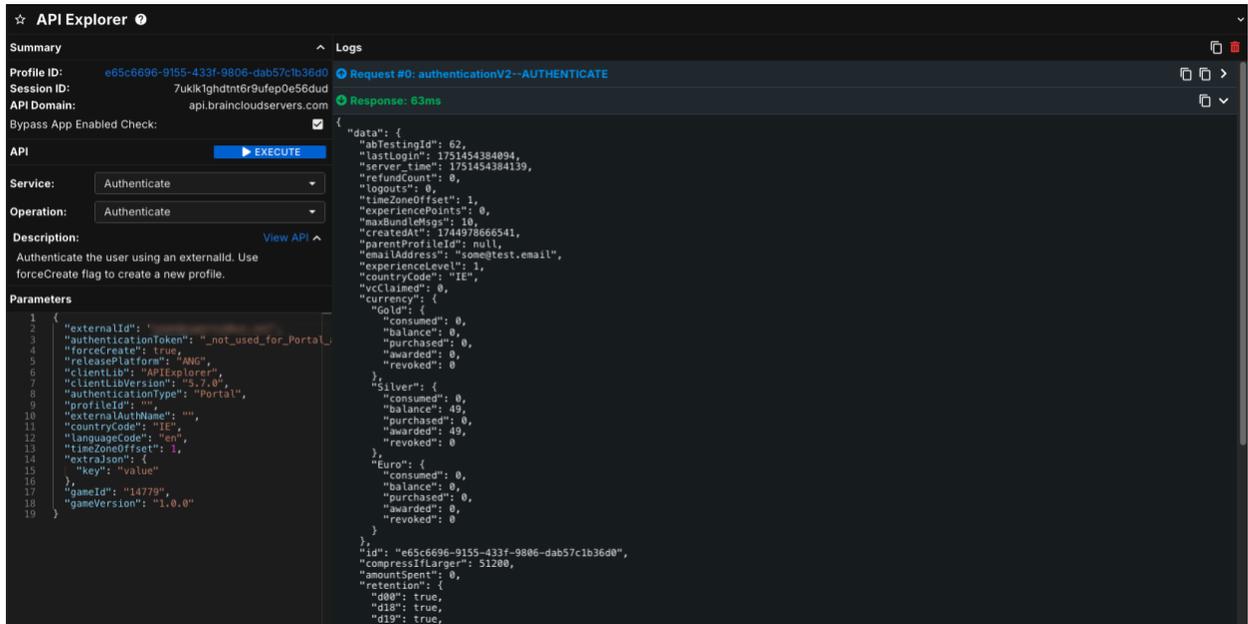
Service: PlayerState
Operation: ReadUserState

```
{
  "data": {
    "abTestingId": 62,
    "lastLogin": 1751832270625,
    "serverTime": 1751454119134,
    "refundCount": 0,
    "logouts": 0,
    "timeZoneOffset": 1,
    "experiencePoints": 0,
    "createdAt": 174497866541,
    "parentProfileId": null,
    "emailAddress": "some@test.email",
    "experienceLevel": 1,
    "countryCode": "IE",
    "acquired": 0,
    "currency": {
      "Gold": {
        "consumed": 0,
        "balance": 0,
        "purchased": 0,
        "awarded": 0,
        "revoked": 0
      },
      "Silver": {
        "consumed": 0,
        "balance": 49,
        "purchased": 0,
        "awarded": 49,
        "revoked": 0
      },
      "Euro": {
        "consumed": 0,
        "balance": 0,
        "purchased": 0,
        "awarded": 0,
        "revoked": 0
      }
    }
  },
  "id": "e65c6696-9155-433f-9806-dab57c1b36d0",
  "amountSpent": 0,
  "retention": {
    "d90": true,
    "d18": true
  }
}
```

You can see from the image above that there is a lot of basic player information returned from this request, along with currency, statistics, etc.

Authentication Response

The ReadUserState will return the basic player information at any time, however, it might only be required upon login or for a sync. brainCloud actually returns this basic information when the player authenticates too, so using the authentication response information you will require one less API call when your game boots.



The screenshot shows the API Explorer interface. On the left, the 'Summary' tab is active, displaying details for the 'Authenticate' API call. The 'Parameters' section shows a JSON object with fields like 'externalId', 'authenticationToken', 'forceCreate', 'releasePlatform', 'clientLib', 'authenticationType', 'profileId', 'externalAuthName', 'countryCode', 'languageCode', 'timeZoneOffset', 'extraId', 'gameId', and 'gameVersion'. The 'Response' section shows a JSON object with a 'data' field containing user account information such as 'abTestingId', 'lastLogin', 'server_time', 'refundCount', 'logout', 'timeZoneOffset', 'experiencePoints', 'maxDundelogs', 'createdAt', 'parentProfileId', 'emailAddress', 'experienceLevel', 'countryCode', 'vcClaimed', 'currency', and various currency balances (Gold, Silver, Euro) with their respective 'consumed', 'balance', 'purchased', 'awarded', and 'revoked' counts. The response also includes the 'id' of the user profile and 'compression' settings.

Remember that you pay for each API call, so if you are using the GetUserAccountInfo API post-authentication, consider replacing this with the authentication response to save on cost.