

Login & connecting a Baziness account

How a player gets a Baziness identity, signs in from the splash screen or Settings, and how your game switches accounts safely.

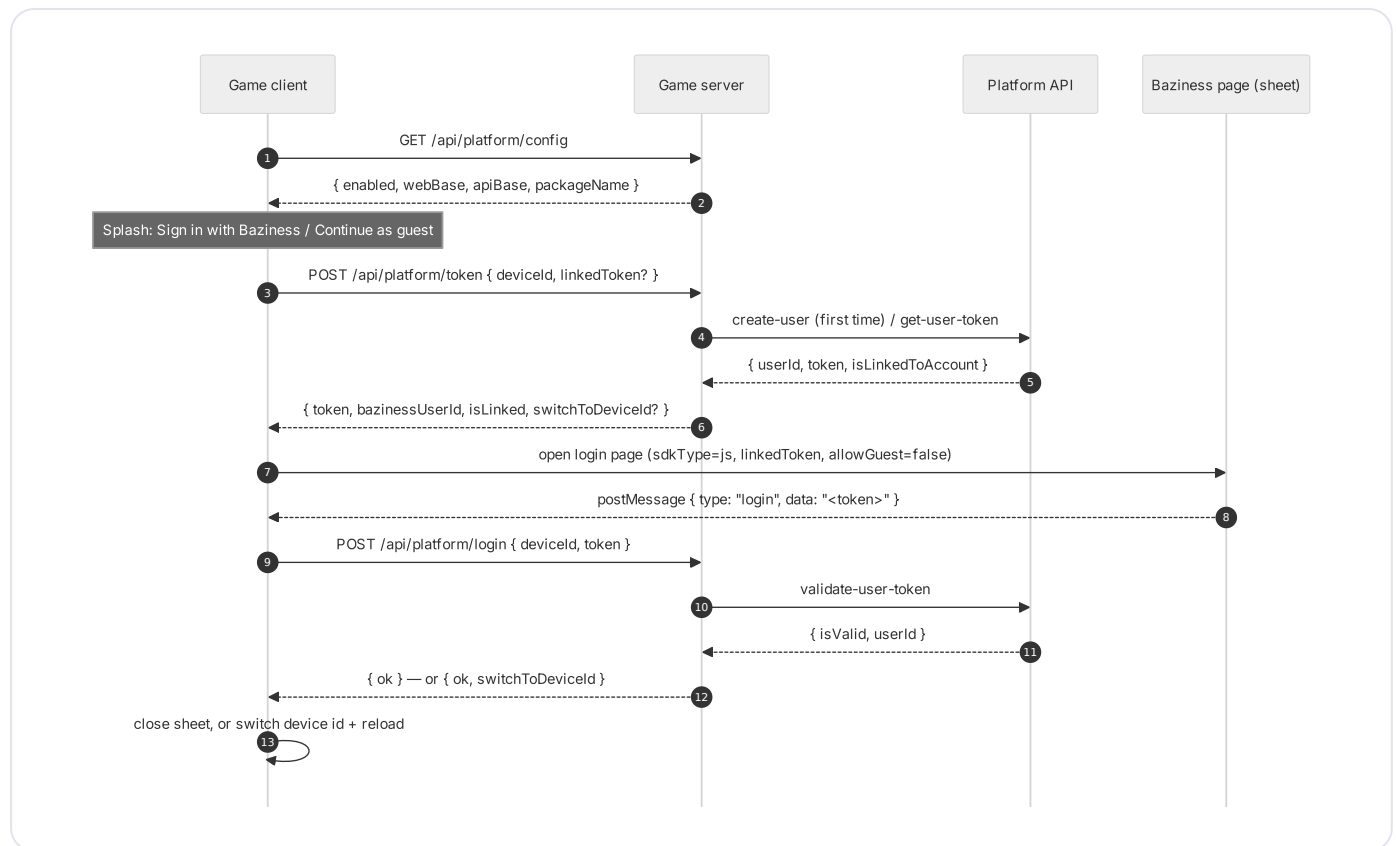
The identity model

Your game keeps **its own player accounts**, for example keyed by a device id generated on first launch. Baziness is attached **on top**:

1. The first time the platform is on, your server creates an **anonymous Baziness user** for the player (`create-user`) and saves its `userId` on the player's record.
2. The player stays **anonymous** (`isLinkedToAccount: false`) until they sign in with their phone number on a Baziness page. The account is then **linked**.
3. If the Baziness account they sign into **already belongs to another of your players** (the same person on another phone), your game **switches** to that player. It doesn't link the account twice.

State	Baziness <code>userId</code> saved?	<code>isLinkedToAccount</code>	What the player sees
Baziness off	—	—	no Baziness UI
Guest	yes (anonymous user)	<code>false</code>	coin balance and reward button work; "connect your account" in Settings
Signed in	yes	<code>true</code>	everything, including claiming Gita coin rewards

The whole flow



What your server provides

Your client never calls the platform's server API itself. It calls **three endpoints of your own server**:

Your endpoint	The client sends	Your server does	It answers
GET <code>/api/platform/config</code>	—	nothing; it only reports settings	<code>{ enabled, webBase, apiBase, packageName, gameName, gameOfferUrl }</code> , or <code>{ enabled: false }</code>
POST <code>/api/platform/token</code>	device id, <code>linkedToken</code> if any	no Baziness user yet → <code>create-user</code> ; otherwise → <code>get-user-token</code> . If the resulting <code>userId</code> belongs to another player, it does not link	<code>{ token, userId, isLinkedToAccount }</code> , or <code>{ switchToDeviceId }</code>
POST <code>/api/platform/login</code>	device id, the token from the sign-in page	<code>validate-user-token</code> → whose account is it? Another player's → switch; otherwise → link to this player	<code>{ ok: true }</code> , or <code>{ ok: true, switchToDeviceId }</code> , or <code>{ ok: false }</code>

If the platform is unreachable, `/token` answers `{ enabled: true, error: "platform_unavailable" }` without a token, and the game boots anyway.

The first launch: guest or sign in

When Baziness is on and the player has neither chosen yet nor signed in, the splash screen shows two buttons instead of the loading bar:

- «ورود با حساب بازینس» (sign in with Baziness): opens the sign-in page and waits.
- «ادامه به عنوان مهمان» (continue as guest): remembers the choice and continues.

The boot order:

1. Sign the player in to **your own** account system.
2. Ask your server for `/api/platform/config`. If Baziness is off, skip everything below.

3. If no choice was made and the player isn't signed in, show the two buttons and wait.
4. Call `/api/platform/token` . If it answers `switchToDeviceId` , adopt that id and restart.
5. If the player just signed in, load their profile again, so the screen shows the signed-in account.

If the player closes the sign-in page without signing in, show the two buttons again.

Account switching, step by step

A switch happens when the Baziness account the player signs into is already linked to **another** of your players, for example after a reinstall or on a new phone:

1. Your server answers `switchToDeviceId: "<that player's device id>"` .
2. The client saves that device id as its own and forgets its stored Baziness token.
3. It restarts, without `linkedToken` in the address.
4. The next boot loads the other player, with all their progress.

Opening the game from the Baziness app

Important

The Baziness app now also sends `?linkCode=` (one-time). Forward it to your server instead of `linkedToken` . See [Token migration](#).

The Baziness app opens games with `?linkedToken=<token>` in the address:

- Send that token as `linkedToken` to `/api/platform/token` , so the player lands in the right Baziness account. Remove it from the address on the next restart.
- If the device already holds a **different** token, a different Baziness account is arriving: clear the local data and start again before continuing.
- Signing in decides **which** account plays, never **whether the player is new**. A first-time player arriving this way must still see your tutorial and receive your welcome gifts.

Signing in later, from Settings

A guest can connect their account at any time: Settings → open the **profile page** ([parameters](#)). When the view closes, ask `/api/platform/token` again and refresh the coin balance.

Signing out

When the page sends `logout`, close the view, **delete everything your game stored locally** (identity, token, choices, tutorial progress) and restart as a brand-new visitor.

Testing logins on the dev platform

Tip

On the **development platform** (the one test games use), you can sign in with **any phone number** and the code `111111`. No SMS is sent. Make up numbers to create as many test accounts as you need: one for the account-switch test, one for a fresh guest, and so on. Production sends real SMS codes.

Common mistakes

Mistake	What happens	Fix
Treating the page's <code>token</code> message as a sign-in	the reward shop closes as soon as it opens; the first sign-in completes before the player signs in	only <code>login</code> is a sign-in
Linking a token without <code>validate-user-token</code>	anyone can claim any account	always verify on the server
Opening the sign-in page without the player's token	the first sign-in from the splash has no anonymous user to bind to	make sure <code>/api/platform/token</code> ran first and pass its token as <code>linkedToken</code>
No <code>allowGuest=false</code> on the sign-in page	the player sees a second "continue as guest"	add it on the sign-in, profile and offer pages
Rewarding the link more than once	signing out and in pays again	pay the "account connected" gift from the server, once per player

Ready-made code: [server endpoints](#), [client controller and messages](#).