|  |  |  |
| --- | --- | --- |
| Case | Test | Result |
| Games get deleted | URL references get deleted | passed |
| Profile gets loaded, then deleted after loaded, Profile was only remaining image refs, image and other profile info currently on screen. | When save profile, creates new refs for images. | passed |
| Piece saved, Piece already exists, the image refs it had before were different and were the only remaining ones of that kind. | Previous piece image refs decremented/removed | passed |
| Selected same image for white and black, then saved. | 2 new image references are created | passed |
| Creating a custom game | Standard piece image URLs are NOT added to game data’s imgUrlStrs list. | passed |
| Creating a custom game, a piece with white/black using the same image is included in the game. | Only once reference is included for that image even though it appears in 2 different locations in the game data. |  |
| Game is stared, saved, then saved a second time with the same name | No new increment to the image URL count occurs for the second save | Passed |
| A game is saved overriding a game that already existed with the same name. | All imgStrUrls in the old game that are not in the new game, get their counts decremented. imgStrUrls in the old also in the new, counts remain the same, because they are losing a source, then gaining one | Passed |

Update:

An increment to the URl count is needed for every entity that can be added/deleted/changed by a user, which is not every place that the URL appears because a game’s individual attributes cannot be edited once a game is created. The entities for which one increment to the count is needed are:

1. An entry in the image pool
2. A global piece def’s white image
3. A global piece def’s black image
4. A created game

No matter how many times the same image appears in a game, only one increment to the count is given. This is because the purpose of image URL refs is to know when an image can be deleted from Firebase storage, and when a game is deleted, then if that game had all remaining usages of that image, than it can.

Games

**Game possible combinations for a new game:**

* Empty imgStrs array,
* Non-empty image strings array, all refs incremented.

**Game possible combinations for deleting a game:**

* All refs decremented, none deleted
* All refs decremented, some deleted
* All refs decremented, all deleted
* Empty array, skip

**Game combinations for overriding game with same name:**

* Old and new both have empty imgUrlStrs arrays
* Old has empty imgUrlStrs array
* New has empty imgUrlStrs array
* Neither has empty array

Piece Profiles

**The following combinations apply to both a new piece getting created, and a piece overriding one with the same name:**

Color:

W: white,

B: black,

Type :

n: new,

s: sample

example:

*Wn is for ‘white new’ Bs is for ‘Black sample’*

* WnBn
* WnBs
* WsBn
* WsBs

**The following combinations apply to deleting a piece:**

* W & B decremented, neither deleted.
* W & B decremented, W at 0 and is sample, ref removed but not storage image
* ~~W & B decremented, B at 0 and is sample, ref remove but not storage image~~
* W & B decremented, both at 0, both are samples, both refs removed, neither storage image is removed.
* ~~W & B decremented, W at 0 and is uploaded, ref + storage image removed~~
* W & B decremented, B at 0 and is uploaded, ref + storage image removed.
* W & B decremented, both at 0, both are uploaded, both refs removed, both storage images removed.

**The following combinations apply to overriding a piece:**

* New W matches Old W
* New W matches Old B
* New B matches Old B
* New B matches Old W
* Neither match