## GitGudDojo LFG Matching and Recommendation Design

Last updated: 12/27/2019

Owner: Sean, Kevin

## **Principle**:

The goal is to provide a more competitive, role / team composition based matching process for dedicated casual R6 players.

Problem to solve: R6 matchmaking only matches based on MMR, which is measuring only W/L ratio. However, a “quality” match has much more than W/L. It needs good communication, good team composition, similarly skilled players.

Solution: By using R6 user data exposed via public API we can create a new LFG user recommendation engine, using publicly available gameplay data and post-game data GitGudDojo will collect to build a machine learning model to recommend players who are most likely to generate a positive teamplay experience for the user.

## MVP V0 Methodology:

V0 LFG model is a bare minimum matching rule to find a player who has a different play style than the user initiating the LFG. The goal is to complete the GitGudDojo platform so we have a working MVP as a POC for testing. Once the entire platform MVP is completed and tested, we will quickly evolve to the next version of the model, as this is the **critical differentiator** for our service.

* **Operator Groupings**: We will group different operators into **operator\_cat** in the big Attacker and Defender category:
  + Attacker
    - entry\_fragger: Ash, Twitch, Amaru
    - soft\_breacher: Sledge, Buck
    - hard\_breacher: Hibana, Thermite, Maverick
    - support: Thatcher, Jackal, Gridlock, Nomad, Capitao
    - active\_ability: Dokebi, Lion, Ying, Finka, Zofia
    - passive\_ability: Blackbeard, Nook, Kali
    - shield: Blitz, Montagne
  + Defender
    - deep\_roamer: Vigil, Jaeger, Bandit, Ela, Mozzie, Pulse, Alibi, Caveira
    - shallow\_anchors: Smoke, Mute, Clash, Goyo
    - deep\_anchors: Mira, Maestro, Warden, Doc, Rook, Tachanka
    - traps: Frost, Kapakan, Lesion, Castle, Kaid, Wamai
    - intelligence: VALKYRIE, Echo
* **Player Recommendation**
  + When a user clicks “LFG”, GRE searches his/her identifier (for now Steam ID) in the database to locate the player’s profile, and locates the **Top 3 Operators** (as shown on R6 tracker website) for the user.
  + Based on Operator Groupings, GRE categorizes the user into entry\_fragger, soft\_breachers based on his/her Top Operators. A user could have multiple categories.
  + GRE then finds **four** different other registered users in the Database at random who has a different set of top operator\_cat than the user’s, and recommends them.
    - For example, if a user’s top 3 operators are Sledge, Bandit and Frost, he will have top operator\_cats as 1) soft\_breacher 2) deep\_roamer 3) traps.
    - Then the system will recommend 4 other users at random whose top 3 operators are NOT in either operator\_cat of 1) soft\_breacher 2) deep\_roamer 3) traps.
  + If user dismisses one or multiple recommendation, GRE will keep recommending the next random user whose operator\_cat is Not the same as the user’s.
* **Data source**: <https://r6.tracker.network>

|  |  |  |
| --- | --- | --- |
| **operator\_type** | **operator\_cat** | **operator** |
| Attackers | entry\_fragger | ash, twitch |
| soft\_breacher | sledge, buck |
| hard\_breacher | hibana,  thermite,  maverick |
| support | thatcher,  jackal,  gridlock,  nomad,  capitao |
| active\_ability | dokkaebi,  lion, ying, finka, zofia |
| passive\_ability | blackbeard,  nokk |
| shield | blitz,  montagne |
| Defenders | deep\_roamer | vigil, jager, bandit, ela, mozzie, pulse, alibi |
| shallow\_anchors | smoke, mute |
| deep\_anchors | mira, maestro, warden,  doc,  rook |
| traps | frost,  kapkan,  lesion,  castle |
| intelligence | valkyrie,  echo |

Next version ideas:

Recommend users to try to match the following team composition:

* **Attackers**: 1 Entry Fragger, 1 Hard breacher, 1 soft breacher, 1 support, 1 Active ability
* **Defenders**: 1 Deep Anchor, 2 Shallow Anchor, 1 Shallow Roamer, 1 Deep Roamer

**Incorporate GitGudDojo “Reputation” Data**

* After each match, the User can rate other players who are matched, in the following categories to endorse them.
  + Good Communication
  + Leadership
  + Fun
  + Positive
  + Respectful
* These 5 attributes would be rated based on cumulative sums. As in the user can’t rate a 1 to retaliate a user.
* At version 1, The target user will have the attribute that has the highest count displayed as their “User Type”
* For the matching for the next round, the players with the highest “Reputation” rating would get priority matching

**Machine learning**

* Build ML model to train the model to learn from the matching results (most notably post LFG reviews)