* Intro
  + Features
  + Installation
  + Support
  + License
* Command
  + ClanManager
    - udpate\_leagues
    - get\_clan
    - get\_league\_infos
    - add\_to\_master\_roster
    - remove\_from\_master\_roster
    - get\_master\_roster
  + DBManager
    - update
    - beforeupdate
    - update\_db
    - create\_league
  + DiscordManager
    - myaccount
  + LineupManager
    - get\_best\_hitters
    - create\_lineup
  + PlayerManager
    - add\_player
    - delete\_player
    - get\_all
    - get\_all\_linked
    - get\_player\_stats
  + UpgradeManager
    - add\_upgrade
    - get\_upgrades
  + WarManager
    - claim\_clan
    - post\_hits
    - unclaim\_clan
    - my\_clan
    - on\_war\_attack
    - on\_war\_state\_change
    - current\_war\_status
    - war\_lineup
* Schemas
  + ER
  + UML
* Test
  + Unity
  + Integrity
* Exceptions
* utility.py
* Diagrams

Lambot

Hey Clasher, Lambot is the Discord bot made for you!

* It will help you out managing your clan, it will make your leader life easier!

Features

* Save your clan hit rates
* Keep a look on your master roster
* Manage all the competitions your clan is involved in
* Create the best lineup for each war

Installation

Install Lambot by following the next steps:

1. add the bot to your server clicking on the following link <https://discord.com/api/oauth2/authorize?client_id=727563602860703855&permissions=2148006976&scope=bot>
2. create a channel and link it to your clan by typing lb claimclan <clan\_tag>
3. let your members to link their accounts to their Discord id by typing lb claim<account\_tag>

Support

If you are having issues, please let us know.

* You can contact me at the following Discord ID: •Karot•#6042

License

The project is intellectual property of the developer Raffaele Scarano.

ClanManager

* udpate\_leagues

perms: bot owner

it has autorun every 10 hours

[update\_leagues|updateleagues|updateleague]

updates the league infos to dabatase

* get\_clan

perms: any

[get\_clan|getclan|clan] <clan\_tag>

Returns clan infos

* get\_league\_infos

perms: any

[get\_league\_infos|getleagueinfos|gli|getleaguenames]

Returns league infos in order to update master roster properly

* add\_to\_master\_roster

perms: leader

[add\_to\_master\_roster|addmr|addtomr|mradd|addtomasterroster][league\_name] [league\_season] [league\_division] [player\_tags...]

Add player to league master roster You can provide multiple players too

* remove\_from\_master\_roster

perms: leader

[remove\_from\_master\_roster|removemr|removefrommr|mrremove|removefrommasterroster] [league\_name] [league\_season] [league\_division] [player\_tags...]

Remove player from league master roster You can provide multiple players too

* get\_master\_roster

perms: leader

[get\_master\_roster|getmasterroster|gmr][league\_name][league\_season][league\_division]

Returns all linked accounts who played in clan Can be filtered to league

DBManager

* update

it has autorun every 6 minutes

updates the database

* beforeupdate

prevents the task from running before the database is connected

* update\_db

perms: bot owner

updates the database

* create\_league

perms: bot owner

creates a new league. It takes name, season, division, description

DiscordManager

* myaccount

perms: any

returns discord account infos

LineupManager

* get\_best\_hitters

perms: any

lb [get\_best\_hitters|gbh|get\_hitrate|getbesthitters] [th] [clan\_tag]

Returns the stats of the players in database.

Add th level to filter on th level and/or clan to filter on a clan NOTE: if channel is linked to a clan, clan filter will be applied

* create\_lineup

perms: any

lb [create\_lineup|createlineup] [league\_name] [league\_season] [league\_division] [ths...]

Creates the best lineup possible based on who is in master roster and has the best performances.

Please provide the war breakdown from max th level to min required Ex: lb create\_lineup 0 5 10 15

PlayerManager

* add\_player

perms: any

Links a coc account to a discord user

lb [add\_player|avatar|claim] <player\_tag> [user]

If no user is given, default user is the command invoker

* delete\_player

perms: any

Unlinks a coc account to a discord user

lb [delete\_player|unclaim] <player\_tag>

If no user is given, default user is the command invoker

* get\_all

perms: bot owner

Returns all players in the database linked to their discord id

* get\_all\_linked

perms: any

Returns all coc accounts linked to a discord user

lb [get\_all\_linked|list|getme] [user] If no user is given, default user is the command invoker

* get\_player\_stats

perms: any

Returns player stats

lb [get\_player\_stats|getplayerstats|gps|mystats] [user] [clan\_tag] If no user is given, default user is the command invoker

Type a clan tag to filter stats by a clan

UpgradeManager

* add\_upgrade

perms: any

lb [add\_upgrade|upgrade|setugrade|set\_upgrade] <player\_tag> <previous\_th> <target\_th> Adds an account upgrade to the database

* get\_upgrades

perms: any

lb [get\_upgrades|upgrades|getupgrades] Returns all upgrades currently ongoing

WarManager

* claim\_clan

perms: any

Claim your war clan to a discord channel

lb claimclan <clan\_tag> Clan attacks will be shown as they come

* post\_hits

perms: leader

Turn on/off post hits feature

lb [post\_hits|posthits] <arg>

Type lb posthits false to turn off post hits

Type lb posthits true to turn on post hits

* unclaim\_clan

perms: any

lb [unclaim\_clan|unclaimclan]

Unclaim your war clan from a discord channel

* my\_clan

perms: any

lb [my\_clan|myclan]

Returns clan linked to this discord channel

* on\_war\_attack

This method overwrites an event into the coc library.

It is called every time one attack happens, sends it to discord and saves the result onto the database

Only attacks from friendly wars between same level townhall are saved onto database.

* on\_war\_state\_change

This method overwrites an event into the coc library.

It is called every time war state changes and it sends a message with the new state.

* current\_war\_status

perms: any

lb [get\_status|status|getstatus|war]

Returns the current war status

* war\_lineup

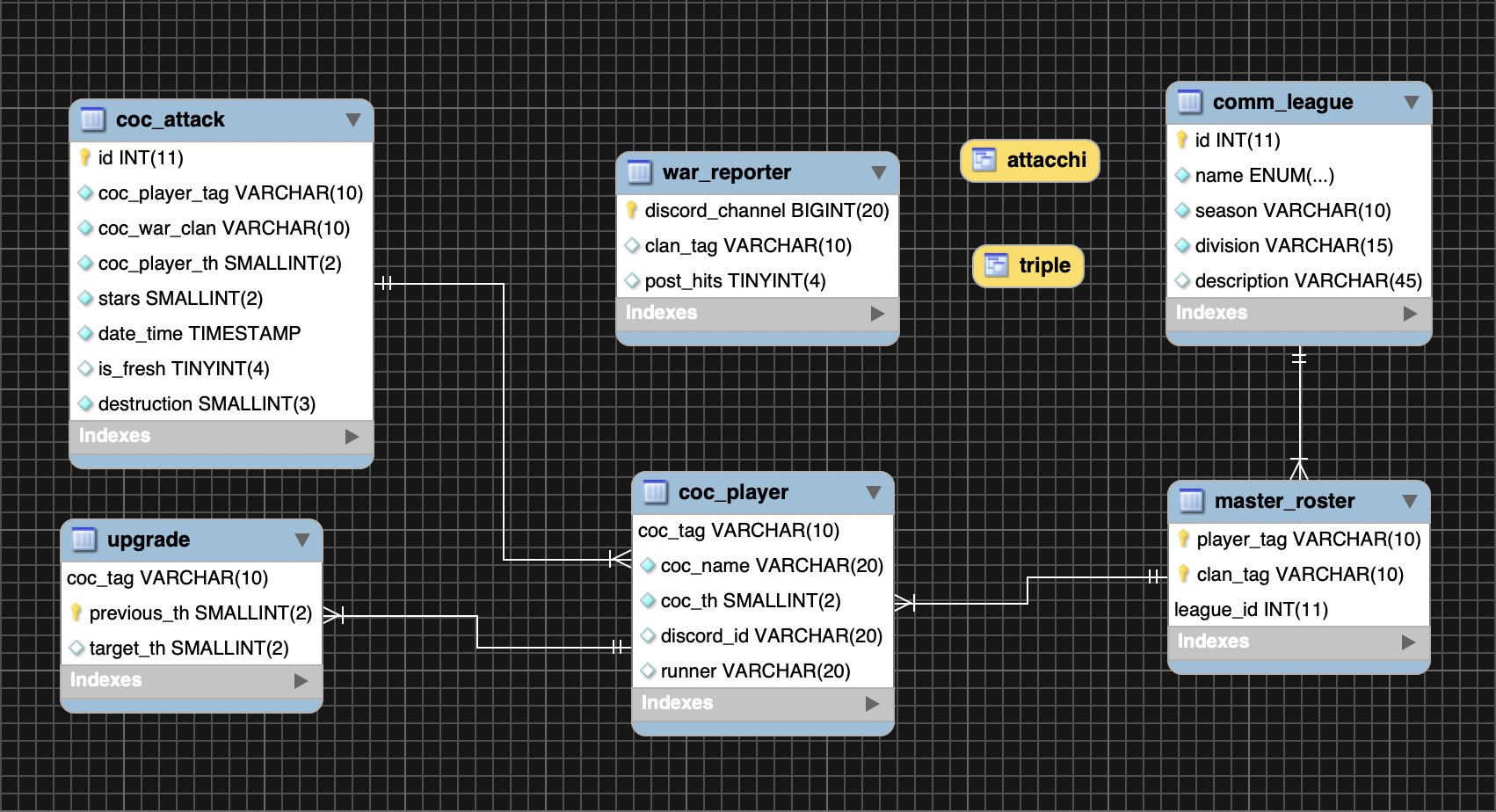
perms: any

lb [war\_lineup|attacks|warlineup|goa|getourattacks]

Returns the lineup of the current war.

Schemas

* ER



* UML

Gli schemi sono disponibili in una cartella [Google Drive](https://drive.google.com/drive/folders/1lwi5dVD3xSXReaPr1u70FEZof4jv6ljV?usp=sharing)

Test

* Unity

|  |  |  |  |
| --- | --- | --- | --- |
| Class | Method | Expected | Test |
| PlayerService | createPlayerTest | to format | Passed |
| WarService | getFirstChannelIDByTag | to find | Passed |
| WarService | getFirstChannelIDByTag | to not find | Passed |
| WarService | getAllChannelsIDByTagPostTrue | to find | Passed |

* Integrity
  + ClanManager

|  |  |  |  |
| --- | --- | --- | --- |
| Method | Command | Expected | Test |
| udpate\_leagues | lb udpate\_leagues | to pass | Passed |
| get\_clan | lb get\_clan | to pass | Passed |
| get\_league\_infos | lb get\_league\_infos | to pass | Passed |
| add\_to\_master\_roster | lb addmr ndl 9 lightweigth #single\_tag | to pass | Passed |
| add\_to\_master\_roster | lb addmr ndl 9 lightweigth #multiple\_tags | to pass | Passed |
| add\_to\_master\_roster | lb addmr ndl 10 lightweigth #multiple\_tags | to not pass (wrong league infos) | Passed |
| remove\_from\_master\_roster | lb removemr ndl 9 lightweigth #single\_tag | to pass | Passed |
| remove\_from\_to\_master\_roster | lb removemr ndl 10 lightweigth #multiple\_tags | to not pass (wrong league infos) | Passed |
| remove\_from\_master\_roster | lb removemr ndl 9 lightweigth #multiple\_tags | to pass | Passed |
| get\_master\_roster | lb gmr ndl 9 lightweight | to pass | Passed |
| get\_master\_roster | lb gmr ndl 10 lightweight | to not pass (wrong league infos) | Passed |
| get\_master\_roster | lb gmr | to pass | Passed |

* + DBManager

|  |  |  |  |
| --- | --- | --- | --- |
| Method | Command | Expected | Test |
| update\_db | lb udpate\_db | to pass | Passed |
| create\_league | lb\_createleague wcl sko blizzard | to pass | Passed |
| create\_league | lb create\_league ndl 9 lightweight | to pass | Passed |

* + DiscordManager

|  |  |  |  |
| --- | --- | --- | --- |
| Method | Command | Expected | Result |
| myaccount | lb myaccount | to pass | Passed |

* + LineupManager

|  |  |  |  |
| --- | --- | --- | --- |
| Method | Command | Excpected | Test |
| get\_best\_hitters | lb gbh | to pass | Passed |
| get\_best\_hitters | lb gbh 12 | to pass | Passed |
| get\_best\_hitters | lb gbh 12 #clantag | to pass | Passed |
| get\_best\_hitters | lb gbh #clantag | to pass | Passed |
| get\_best\_hitters | lb gbh #wrong\_clantag | to not pass | Passed |
| create\_lineup | lb create\_lineup ndl 9 lightweight 5 10 5 10 | to pass | Passed |
| create\_lineup | lb create\_lineup ndl 9 lightweight 5 10 2 | to not pass (wrong format) | Passed |
| create\_lineup | lb create\_lineup ndl 10 lightweight 5 10 5 | to not pass (wrong league infos) | Passed |
| create\_lineup | lb create\_lineup ndl 9 lightweight 5 10 5 35 | to not pass (wrong format) | Passed |

* + PlayerManager

|  |  |  |  |
| --- | --- | --- | --- |
| Method | Command | Expected | Test |
| add\_player | lb claim #accounttag | to pass | Passed |
| add\_player | lb claim #wrongaccounttag | to not pass | Passed |
| add\_player | lb claim #accounttag @discordtag | to pass | Passed |
| delete\_player | lb unclaim #accounttag | to pass | Passed |
| delete\_player | lb unclaim #wrongaccounttag | to not pass | Passed |
| get\_all | lb getall | to pass | Passed |
| get\_all\_linked | lb getme | to pass | Passed |
| get\_all\_linked | lb getme @discordid | to pass | Passed |
| get\_player\_stats | lb gps | to pass | Passed |
| get\_player\_stats | lb gps @discordid | to pass | Passed |

* + UpgradeManager

|  |  |  |  |
| --- | --- | --- | --- |
| Method | Command | Expected | Test |
| add\_upgrade | lb upgrade #accounttag 12 13 | to pass | Passed |
| add\_upgrade | lb upgrade #accounttag 13 12 | to not pass | Passed |
| get\_upgrades | lb getupgrades | to pass | Passed |

* + WarManager

|  |  |  |  |
| --- | --- | --- | --- |
| Method | Command | Expected | Test |
| claim\_clan | lb claimclan #clantag | to pass | Passed |
| claim\_clan | lb claimclan #clantag | to not pass (channel already linked) | Passed |
| post\_hits | lb posthits false | to pass | Passed |
| post\_hits | lb posthits true | to pass | Passed |
| post\_hits | lb posthits true | to not pass (channel not linked) | Passed |
| unclaim\_clan | lb unclaimclan | to not pass (channel not linked) | Passed |
| unclaim\_clan | lb unclaimclan | to pass (channel linked) | Passed |
| my\_clan | lb myclan | to pass (channel linked) | Passed |
| my\_clan | lb myclan | to not pass (channel not linked) | Passed |
| current\_war\_status | lb status | to pass (channel linked) | Passed |
| current\_war\_status | lb status | to not pass (channel not linked) | Passed |
| war\_lineup | lb warlineup | to pass (channel linked) | Passed |
| war\_lineup | lb warlineup | to not pass (channel not linked) | Passed |

Exceptions

Any exceptions not checked in the code, is handled by Discord error handler and shown as a result of the command sent.

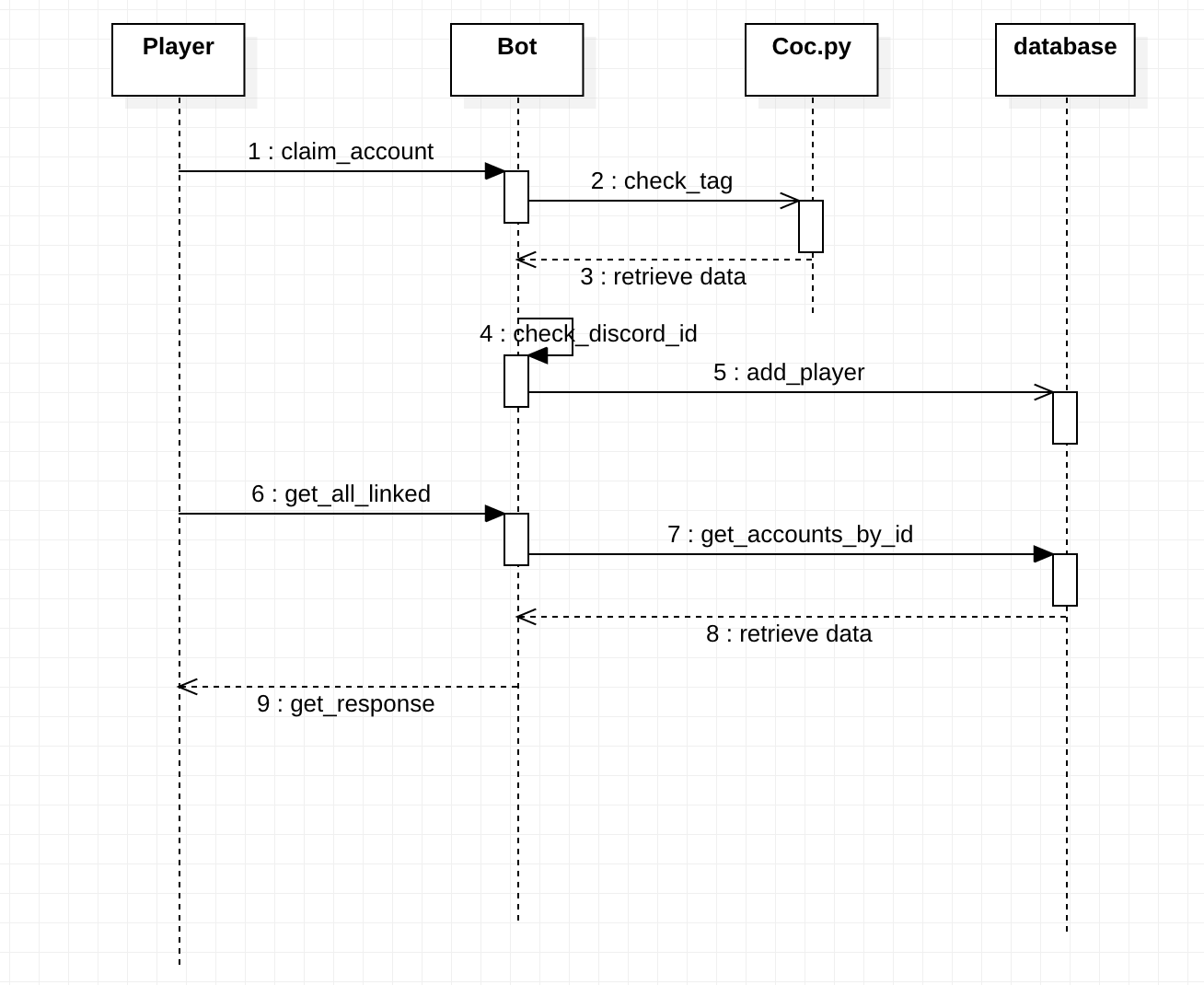
utility.py

This file contains emojis, images and townhall level database.

It is used in graphic functions in order to retrieve the right image/emoji for the command requested.

Diagrams

* The very first action to be made is to link every single discord ID to their own coc accounts. The next chart explains how the process works and what happens after the player sends the right command.



* Most complex feature is to auto create lineup based on members of a clan and their performance in passed events.

