bot-lg documentation

Table of content

1. What is bot-lg

bot-lg is a Discord bot giving access to helpful commands, making Game Master's life easier during a Werewolf game on Discord.

DISCLAIMER: Some commands and the messages sent by bot-lg are in French. I will work later on a translation support.

2. Prerequisite

2.1. For your Discord server

1. Create a Discord application
2. In the OAuth2 tab:
   1. Select the "bot" scope
   2. Select the following bot permissions:
      - Manage Roles
      - View Channels
      - Send Messages
      - Embed Links

        - Read Message History
        - Mention Everyone
        - Mute Members
        - Deafen Members
        - Move Members

3. Add your new bot to your Discord server using the generated link

Note: Please, be sure to keep your bot token to fill the app/config.json file later.

2.2. For Docker application run

- Install Docker
- Install docker-compose

2.3. Optionnal

If you are a make user, a Makefile is available!
Intall the `make` command.

3. Install bot-lg

1. Clone or download bot-lg project from github
2. Create the required configuration files in the project using the following *.dist.* files:
    - `.dist.env => .env`
    - `app/config.dist.json => app/config.json`
    - `app/commands/cmd_config.dist.json => app/commands/cmd_config.json`
    - `init-postgres/initdb.dist.sql => init-postgres/initdb.sql`
    - `php-api/config.dist.php => php-api/config.php`

At this point, bot-lg is ready to start.

4. Finalise the install

In order to initialise and start bot-lg, you need to follow these steps:

4.1. Classic initialisation

1. Open a ***bash*** command line at the project root
2. Issue the following commands:
    1. `sudo docker image build --no-cache --tag node:bot-lg --file bot-lg.Dockerfile`
    2. `sudo docker image build --no-cache --tag php:php-api --file php-api.Dockerfile`
    3. `sudo docker run --detach --rm --name bot-lg --volume "$PWD/app/:/app/" node:bot-lg npm install`
    4. `sudo docker-compose up --detach`

4.2. Initialisation with the make command

1. Open a ***bash*** command line at the project root
2. `make prepare start`

You can see all other make rules using `make` or `make help`.

Start and stop bot-lg

Classic commands

To start and stop bot-lg you can issue the following commands:

- Start bot-lg: `sudo docker-compose up --detach`
- Stop bot-lg: `sudo docker-compose down`

Make commands

To start and stop bot-lg you can use the make rule available:

- Start bot-lg: `make start`
- Stop bot-lg: `make stop`

5. Project structure

```
bot-lg
+-- app                             : contains the bot-lg
app
|   +-- commands                    : contains bot-lg
commands
|   |   +-- *.js                    : all bot-lg commands
|   |   +-- cmd_config.dist.json    : configuration file
template for the bot-lg commands. Contains the required role
and channel ids
|   |
|   +-- config.dist.json            : configuration file
template for the bot-lg app
|   +-- index.js                    : main file to run bot-
lg
|   +-- package-lock.json           : node dependencies to
be installed
|   +-- package.json                : node dependencies to
be installed
|
+-- init-postgres                   : contains the files
used to init the PostgreSQL database
+-- php-api                         : contains the files
needed by the PHP API
|
+-- .dist.env                       : docker-compose
environment variables file template
+-- .dockerignore                   : elements to ignore by
docker
+-- .gitignore                      : elements to ignore by
```

```
git
+-- docker-compose.yml              : docker-compose file
+-- bot-lg.Dockerfile               : "bot-lg" container
image
+-- php-api.Dockerfile              : "php-api" container
image
+-- Makefile                        : all make rules
available to manage "bot-lg" container
+-- README.md                       : project documentation
+-- logo.zip                        : bot-lg logo by Kévin
BOURBASQUET
```

6. Discord server requirements

In this section you can find the required elements for you Discord server. In order to use bot-lg, you need to copy-paste `/app/commands/cmd_config.dist.json` as `/app/commands/cmd_config.json` and fill this file with the following required element ids.

6.1. Roles

Here is the list of the required role:

Role ids	Name	Types	Descriptions
idRoleAdmin	Admin	Moderator	Administrator of the Discord server
idRoleGameMaster	Game Master	Moderator	Game Master.
idRoleVillager	Villager	Villager	Has no ability.
idRoleCupid	Cupid	Villager	During the first night, chooses 2 lovers.
idRoleLovers	Lovers	Additional	Chosen by Cupid. If the 2 lovers are villagers, they win with the village. Else, they win in solo. If one of the lovers fall, so is the other.

Role ids	Name	Types	Descriptions
idRoleGuard	Guard	Villager	Once per night, protects someone from the werewolf attack, but cannot protect the same person twice in a row.
idRoleWerewolf	Werewolf	Werewolf	Once per night, votes to kill a villager.
idRoleWhiteWerewolf	White Werewolf	Solo Werewolf	Acts with the werewolf, but can kill one of his mates every other night.
idRoleInfectWerewolf	Infect Werewolf	Werewolf	Once per game, after the werewolves vote, chooses to infect the villager chosen by his mates, making him become a werewolf.
idRoleInfected	Infected	Additional, Werewolf	Chosen by Infect Werewolf. Becomes a werewolf, but keeps his original role. If inspected by Seer, the role shown is the original one.
idRoleWitch	Witch	Villager	Once per night, chooses to use a potion or not. Has 2 potions per game: one to resurrect the werewolves' victim, another to kill someone.
idRoleSeer	Seer	Villager	Once per night, can see someone's role.
idRoleAssassin	Assassin	Solo	Once per night, can kill someone.

Role ids	Name	Types	Descriptions
idRolePyromaniac	Pyromaniac	Solo	Once per night, chooses to oil someone or to ignite those who were previously oiled
idRoleOiled	Oiled	Additional	Chosen by Pyromaniac. Dies when Pyromaniac chooses to burn his victims.
idRoleFlutist	Flutist	Solo	Once per night, can enchant 2 players who cannot attempt to kill their master and have to defend him.
idRoleEnchanted	Enchanted	Additional	Chosen by Flutist. Cannot attempt to kill Flutist, and have to defend him.
idRoleReaper	Reaper	Villager	Alive, acts as a villager. Dead, can talk to the other dead players and vote during the two daily vote following the player's death.
idRoleAncient	Ancient	Villager	Can survive to 1 werewolf attack. If the player dies during the daily vote, all villager-type players lose their abilities.

Role ids	Name	Types	Descriptions
idRoleAngel	Angel	Solo, then Villager	Has to die in the first daily vote. If it is a success, the game is over and the player wins. Else, he becomes a villager.
idRoleShaman	Shaman	Villager	At night, can listen the dead players.
idRoleHunter	Hunter	Villager	When eliminated, the player can kill someone else.
idRoleDead	Dead	Dead Player	When a player is eliminated, this role replaces the other one(s). Can talk to Shaman at night.
idRoleMuted	Muted	Additional	Role given by the !nuit command to mute players.
idRoleEveryone	@everyone	Default	Default Discord role allowing to mention all the server members.

Types explaination:

- Moderator: Server and game manager. A player can be Admin, but not Game Master
- Villager: Wins when all threats (werewolves and solos) are eliminated.
- Werewolf: Wins when all villagers are eliminated and solos.
- Solo: Wins when all other players are eliminated.
- Dead Player: Players who have been eliminated.
- Additional: This role is added to the player roles. The player can use the abilities provided by his first role, and all his Additional roles.
- Default: Default Discord roles.

6.2. Channels

Here is the list of the required channels:

Channels ids	Name	Description

Channels ids	Name	Description
idVocalChannelMain	vocal	Vocal channel use by the bot-lg to determine the Game Master and players
idTextChannelGameMaster	game-master	Text channel used by the Game Master (idRoleGameMaster)
idTextChannelWitch	witch	Text channel used by the !vote command in the "sor" case
idTextChannelPyromaniac	pyromaniac	Text channel used by the !vote command in the "pyr" case

7. Bot commands

Here is the list of the bot-lg commands:

Commands	Roles required to use commands	Descriptions
commandes	None	Prints available commands
roles	None	Prints still alive game roles
joueurs	Game Master	Prints roles by player in the "game-master" text channel
assigner	Game Master	Assigns the given role to the given players
commencer	Game Master	Assigns specified roles randomly to all the players
reset	Game Master	Remove all game roles from the players
nuit	Game Master	Mutes all players
jour	Game Master	Unmutes all players
poll	Game Master	Prints a vote form with given question and choices
vote	Game Master	Prints the specified vote form with !poll

Commands	Roles required to use commands	Descriptions
timer	Game Master	Runs a timer for n s/m/h (default 3m)
clear	Admin	Clears the current text channel
deconnexion	Admin	Disconnects bot-lg from the Discord server

8. Authors

- Lilian POULIQUEN: Bot creation and development, documentation

9. Contributors

- Léandre KERUZEC: Command ideas, Documentation review
- Kévin BOURBASQUET: bot-lg logo designer and creator