

# DOC UTILISATEUR - LOUP GAROU ALEXA

## Installation

Télécharger le .apk de l'application dans la section Release puis installer le fichier sur un smartphone Android

Le Skill Alexa est disponible via le fichier lambda\_alexapy, utiliser ce code pour créer un skill via un compte développeur Amazon Alexa

## Déroulement d'une partie:

Pour démarrer une partie, il faut appeler l'intent de démarrage de partie.  
Alexa ne peut pas faire d'appel automatique à l'application de jeu, ainsi, il faut appeler l'intent d'état de partie pour que Alexa fasse la narration.

La nuit, la voyante et les loups garou font leur choix via l'application mobile, il n'y a donc pas d'interaction avec Alexa la nuit.

Néanmoins, le jour, les villageois choisissent leur cible à éliminer. Cela se fait via Alexa.  
Les joueurs peuvent désigner un porte parole qui dira l'intent de vote des villageois.  
Si le joueur tué par les villageois est le chasseur, ce dernier peut éliminer un autre joueur via l'intent du chasseur.

## Invocation:

Pour utiliser des intents de l'application, dites  
**alexa loup garou**, alexa se met en écoute et vous pouvez dire l'intent souhaité  
ou  
**alexa demande à loup garou <intent>**

## Intents:

### Démarrer la partie:

- Commence la partie
- Démarre la partie

### Etat de la partie:

- Etat de la partie
- Statut de la partie
- Récupère l'état de la partie
- Récupère le statut de la partie

### Vote des villageois:

- Le village vote contre le joueur <Numéro du Joueur>
- Nous votons contre le joueur <Numéro du Joueur>

- Nous choisissons le joueur <Numéro du Joueur>

Choix du chasseur:

- J'élimine le joueur <Numéro du Joueur>
- Je tue le joueur <Numéro du Joueur>

# DOC TECHNIQUE - LOUP GAROU ALEXA

## Serveur - API

Le serveur est codé en Python, avec Flask.

### Classes

TypeCarte(Enum): Contient la liste de toutes les cartes du jeu (Simple villageois, Loup, Chasseur...), ainsi qu'une carte Undefined, qui sert de placeholder lorsque les rôles n'ont pas encore été attribués.

EtatPartie(Enum): Contient tous les états de la partie. Par état, on entend surtout "Quel joueur / type de joueur peut faire une action".

Partie: Contient tout ce qu'il faut pour gérer une partie.

## API

Le serveur dispose de plusieurs points d'api.

Le texte entre <> est un paramètre obligatoire.

- / : Affiche une page basique avec l'état de la partie, et la liste des joueurs.
- /status : Renvoie un json avec les détails d'une partie en cours, incluant la liste des joueurs, leurs rôles et leur état (Vivant ou non)
- /join/<pseudo> : Ajoute un joueur avec comme nom <pseudo>
- /listejoueur : Renvoie un json avec la liste des joueurs.
- /joueur/<id> : Renvoie un json avec les informations du joueur numéro <id>
- /demarrerpartie : Démarre une partie. Utilisable dès qu'il y a 4 joueurs.
- /voyante/<id> : Renvoie un json avec la carte du joueur <id>. Utilisable seulement quand c'est au tour de la voyante de jouer
- /loupsvote/<idloup>/<idvictime> : Permet au loup <idloup> de voter pour manger le joueur <idvictime>. Ne peut être utilisé qu'au tour des loups, et que par des loups vivants.
- /villageoisvote/<idvictime> : Permet aux villageois de lyncher le joueur <id>. Utilisable seulement le jour. Ce point d'API est censé être utilisé par Alexa
- /chasseur/<idvictime> : Permet au chasseur de tuer le joueur <id>. Utilisable seulement quand le joueur lynché est le chasseur.
- /reset : Permet de réinitialiser le serveur, supprimant tous les joueurs, et réinitialisant la partie.

Tous les jsons contiennent un champ "response", qui contient un code: "200" lorsque la requête est valide et que le serveur accepte la commande, "403" sinon.

## Front - Application Flutter

L'application est codée à l'aide de Flutter.

Elle dispose de plusieurs pages :

- Page de Login : Un nom d'utilisateur est entré pour permettre la connexion à une partie à l'aide du bouton "Envoyer"
- Page de Lobby : Affichage de tous les membres de la partie en temps réel, lorsque Alexa démarre la partie, l'application passe directement à la page de jeu.
- Page de Jeu : Le joueur retrouve son pseudo, son numéro de joueur ainsi que son rôle dans la partie, il visualise également la narration d'Alexa. Lors de vote (Loup, Voyante), l'application affiche la page de vote.
- Page de Vote : Le joueur en question doit choisir une personne dans la liste des joueurs présent dans la partie, si le joueur est une Voyante : l'application affichera une notification du rôle de la personne choisi, si le joueur est un Loup : la personne sélectionnée meurt, son application passe sur la page de Mort.
- Page de Fin de partie : Lorsque tous les Loups ou Villageois sont mort, l'application indique les gagnants de la partie à l'aide de cette page dédiée qui contient une image dynamique, le nom du groupe gagnant et un bouton de retour à la page Login.
- Page de Mort : Si un joueur meurt, cette page s'affiche elle contient une image dynamique, un message indiquant que le joueur est mort et également un bouton de retour à la page de Login.
- Fichier api.dart : ce fichier contient toute les fonctions et classes relatives à la communication avec l'API, des commentaires indiquent leurs utilisations.