

BotZilla, robot Discord

Un projet personnel par Yann Vogel

Sommaire

BotZilla	4
Contexte:	5
Préfixe :	5
Commandes	6
aks	7
avatar	8
delete	9
egs	10
help	11
kick	12
listening	13
quote	14
role	15
roll	16
server	17
twitch	18
version	19
Fonctions automatisées	20
Accueil des nouveaux venus	21
egsFetcher	22
steamFetcher	23
steamFeed	24
streamStarting	25
Gestion des erreurs	26
LootZilla	28
Contexte	29
Commandes	30
Jeu de base	31
fiche	32
loot	33
loutre	34
shop	35
buy	36
useltem	37
money	38

and the Charles	26
moneyStats	
myInventory	
mycd	41
admin	42
Challenges	43
challenge	44
acceptChallenge	45
refuseChallenge	46
myChallenges	47
Endurance, maps et aventures	48
maps	49
adventure	50
Craft, mutations et attaques	51
mutations	52
mutate	53
myMaterials	54
myMutations	55
power	56
attack	57
Améliorations	58
improve	59
improveHelp	60
Annexes	61
Système de niveau et d'endurance	62
Fiche de personnage, inventaires et limitations	63
Les sacs de butin, les sacs maudits et les rubis	
Les items	
Les animaux (huffs)	66

BotZilla

Contexte:

BotZilla est un robot Discord que j'ai créé dans le but de faciliter certaines tâches aux utilisateurs du serveur Discord de Bobzilla.

BotZilla utilise une base de données NoSQL avec MongoDB.

Préfixe:

Le préfixe à utiliser pour communiquer avec botZilla est **bz!** (il faut le renseigner devant chaque commande).

Commandes

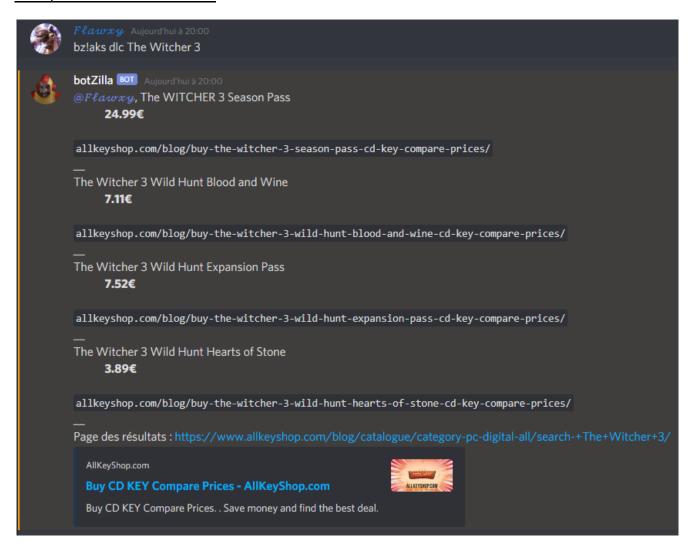
aks

Permet de faire une recherche sur le site allkeyshop.com (web scraping) pour un jeu ou bien un DLC.

Exemple de recherche de jeu:



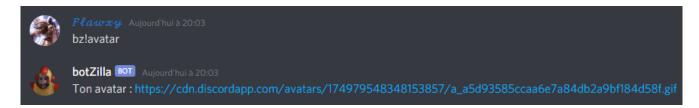
Exemple de recherche de DLC:



avatar

Permet d'afficher le lien vers l'avatar de l'utilisateur ou d'un/plusieurs utilisateur(s) précisés.

Exemple pour l'avatar de l'utilisateur :



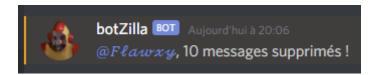
Exemple pour obtenir plusieurs avatars :



delete

Permet de supprimer un (par défaut) ou plusieurs messages. Nécessite des droits spécifiques.

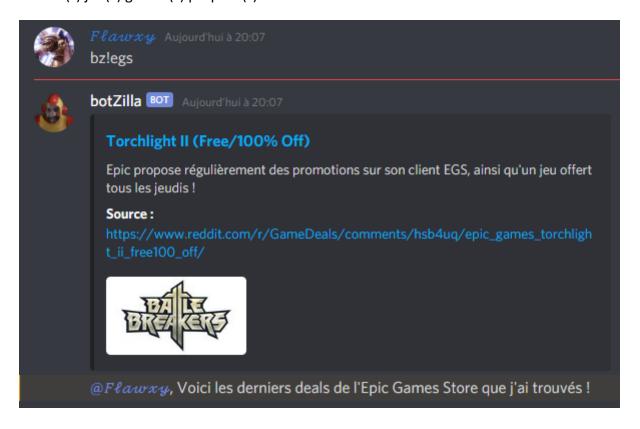
Exemple avec bz!delete 10:



(Le message de confirmation de suppression s'efface automatiquement au bout de quelques secondes.)

egs

Epic Game Store propose régulièrement des jeux gratuits sur son client. Cette commande permet d'afficher le(s) jeu(x) gratuit(s) proposé(s).

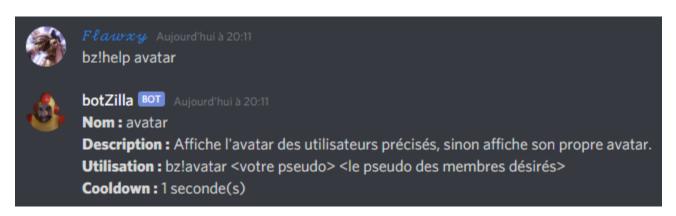


help

Permet d'afficher les commandes disponibles (les commandes d'administration restent cachées). Il est possible d'obtenir une aide personnalisée sur chaque commande en utilisant bz!help <nom de la commande>.



Exemple d'aide personnalisée :



kick

Permet de kick un membre du serveur. <u>Nécessite des droits spécifiques</u>.



listening

Permet de savoir sur quel serveur (local ou distant) le bot est connecté. <u>Utilisable uniquement par l'administrateur</u>.



quote

Permet d'afficher une citation du stream de Bobzilla choisie au hasard (utilise l'API de Twitch).





role

Permet aux utilisateurs de s'attribuer un rôle existant en réagissant avec un émoji. Afin d'améliorer l'expérience utilisateur, botZilla ajoute une réaction à son message avec l'émoji en question.

Nécessite des droits spécifiques.



roll

Permet de lancer un dé. Le dé lancer par défaut est 1d6. La commande accepte plusieurs syntaxes.

Si plusieurs dés sont lancés, seul le résultat est affiché. Le détail des lancers est envoyé en mp à l'utilisateur.

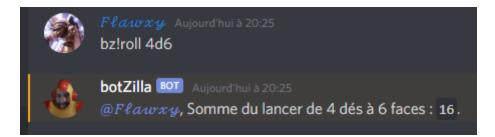
Syntaxe de base :



Syntaxe plus précise :



Exemple d'un jet de plusieurs dés :



Avec le détail dans les dm:



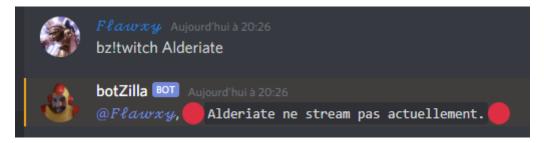
server

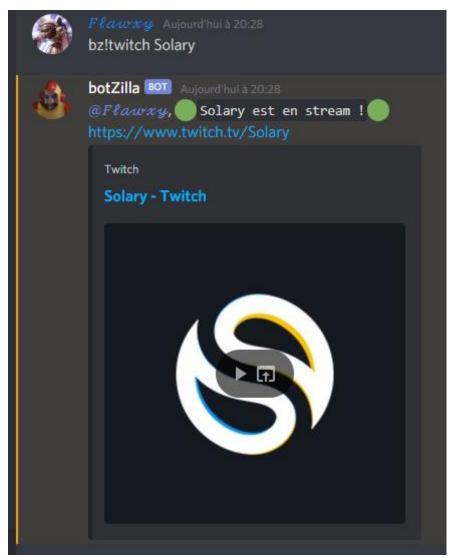
Permet d'obtenir des informations sur le serveur où la commande est utilisée.



twitch

Permet de savoir si un streamer Twitch est en live actuellement. Cette commande utilise l'API Twitch.





version

Permet de connaître la version actuelle de botZilla. <u>Utilisable uniquement par l'administrateur</u>.



Fonctions automatisées

Ces fonctions sont incorporées au bot et ne nécessitent aucune action de la part des utilisateurs.

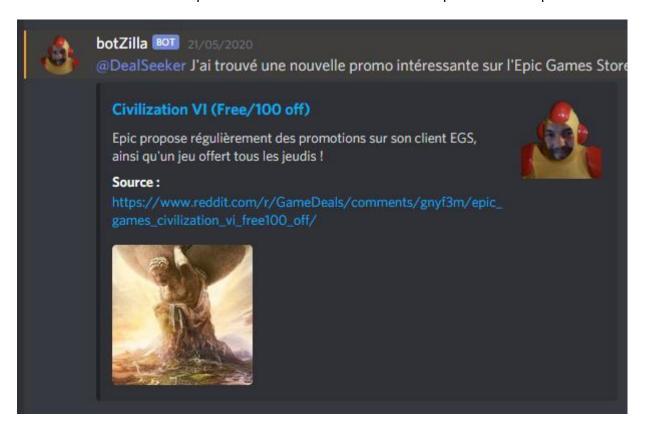
Accueil des nouveaux venus

BotZilla possède une fonction qui permet d'accueillir un nouveau venu sur le serveur Discord.



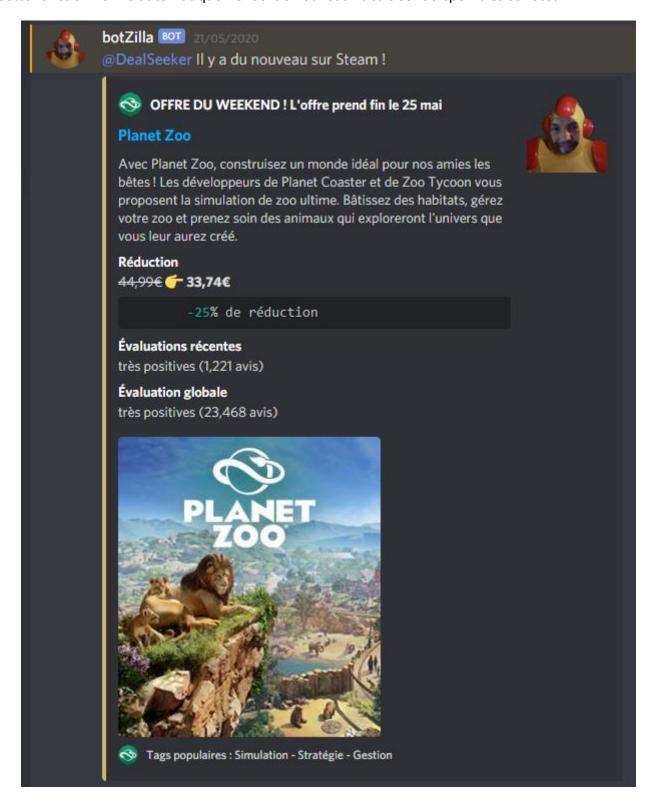
egsFetcher

Cette fonction vérifie automatiquement si de nouveaux deals sont disponibles sur l'Epic Games Store.



steamFetcher

Cette fonction vérifie automatiquement si de nouveaux deals sont disponibles sur Steam.



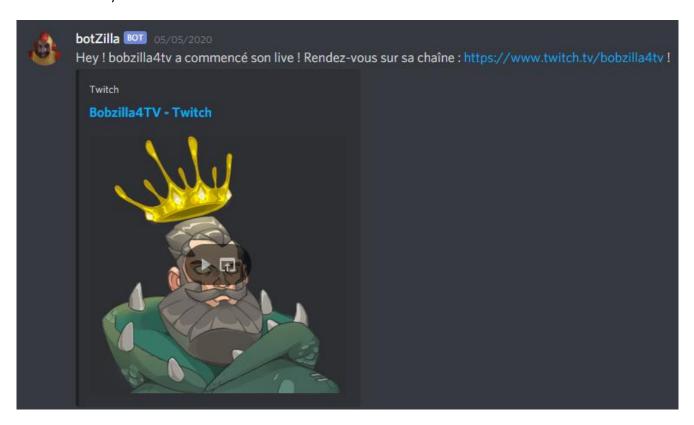
steamFeed

Cette fonction permet de surveiller un jeu Steam en particulier afin d'en obtenir toutes les mises à jour.



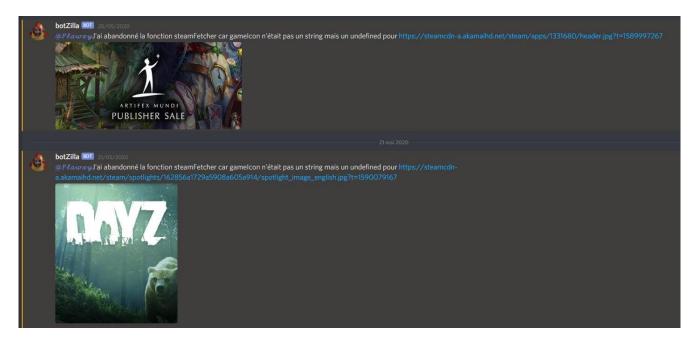
streamStarting

Cette fonction permet d'alerter le serveur Discord quand Bobzilla lance son stream sur Twitch (utilise l'API de Twitch).



Gestion des erreurs

Les éventuelles erreurs de BotZilla sont rapportées automatiquement à l'administrateur en messages privés.



Les commandes utilisées ainsi que leurs utilisateurs sont enregistrés dans un log serveur.

```
Commande "mutate" réussie : réclamée par Neöphix sur le serveur 麦わらの一味.
Commande "mymaterials" réussie : réclamée par Neöphix sur le serveur 麦わらの一味.
Commande "mymutations" réussie : réclamée par Neöphix sur le serveur 麦わらの一味.
Commande "attack" réussie : réclamée par Neöphix sur le serveur 麦わらの一味.
```

LootZilla

Contexte

BotZilla incorpore un jeu à part entière (LootZilla) qui dispose de commandes dédiées.

Le but du jeu est d'obtenir le plus de monnaie du jeu (**bobzi\$\$**) possible. Cette monnaie n'a pour l'instant aucune utilisé particulière, à part être dépensée dans le jeu. L'objectif au long terme est de proposer des avantages Discord/Twitch en échange de cette monnaie.

Le jeu est régit par un système de temps d'attente (cooldown) qui force le joueur à réfléchir à ses actions et à patienter entre chaque action effectuée.

Commandes

Jeu de base

Au départ, seul le système de collecte de sacs était disponible. Au fur et à mesure, j'ai incorporé de nombreuses mises à jour, accompagnées de patch notes, qui ont permis de développer de plus en plus LootZilla.

fiche

Cette commande permet à un joueur de s'inscrire au jeu (obligatoire) ou bien de consulter sa fiche.

Fiche d'un nouveau joueur :



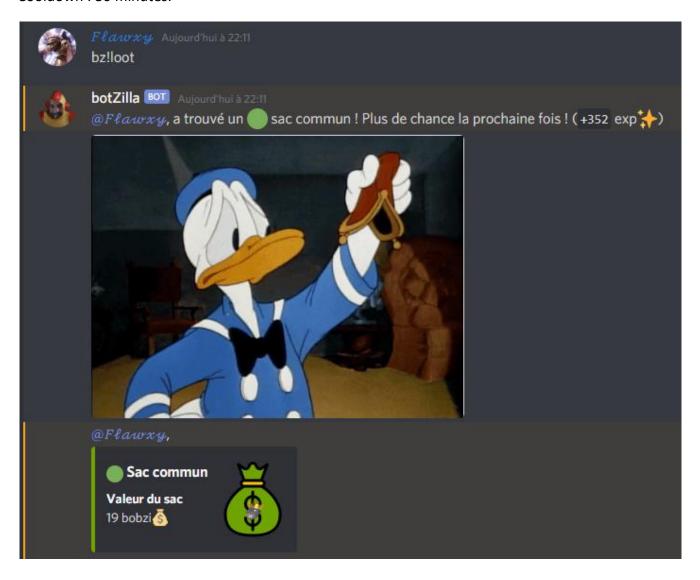
Fiche d'un joueur avancé :



loot

La commande principale. Elle permet d'obtenir un sac contenant des bobzi\$\$.

Cooldown: 30 minutes.



loutre

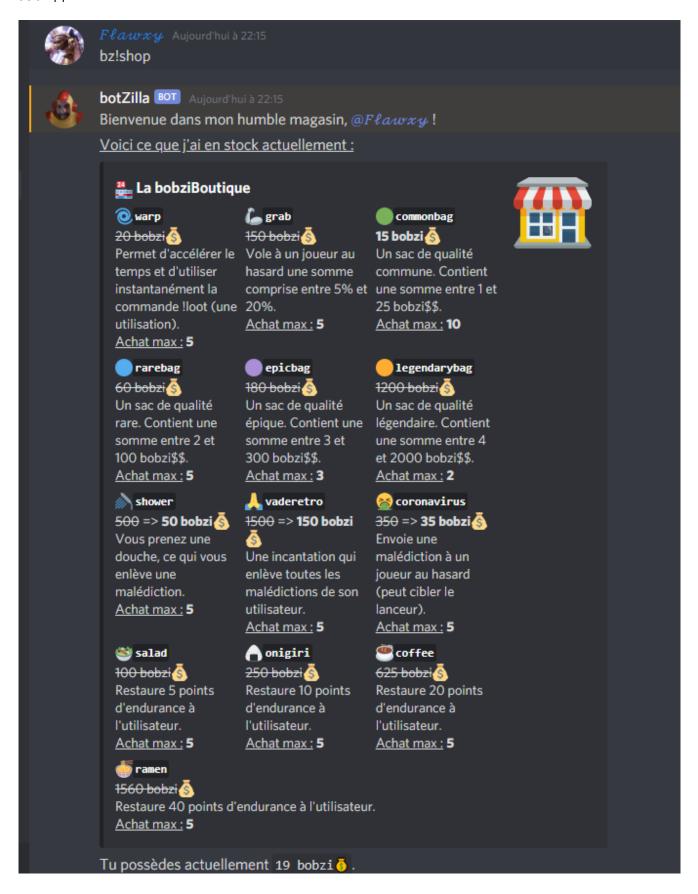
Cette « commande » a remplacé la commande loot pendant la journée du 1^{er} avril.

Elle affiche une photo aléatoire d'une loutre prise dans une base de données tout en générant une phrase du type « C'est un animal très... » suivie d'un adjectif aléatoire.



shop

Permet d'afficher le magasin, ses articles et ses promotions du moment. Tous les achats se font en bobzi\$\$.

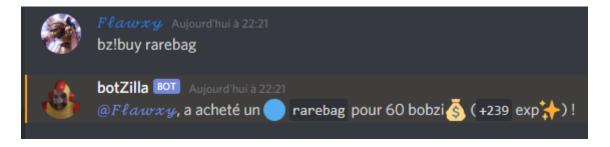


(Le prix est barré si le joueur n'a pas assez d'argent pour acheter le produit)

buy

Permet d'acheter un objet qui se trouve dans le magasin.

Cooldown: 10 minutes.



useItem

Permet d'utiliser un objet qui se trouve dans l'inventaire du joueur.

Cooldown : 60 minutes. Tous les objets partagent le même CD (un joueur ne peut utiliser qu'un objet par heure).



money

Permet à un joueur d'afficher les bobzi\$\$ qu'il possède sans avoir à afficher sa fiche de personnage complète.



moneyStats

Permet de connaître le classement des joueurs en fonction de leur richesse.



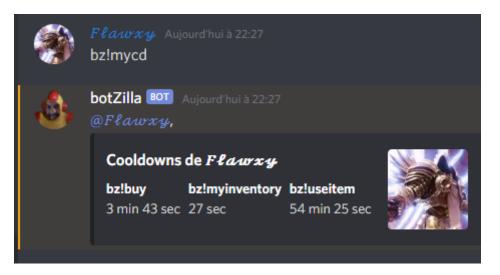
myInventory

Permet à un joueur d'afficher son inventaire sans avoir à afficher sa fiche de personnage complète.



mycd

Permet à un joueur de connaître ses cooldowns actuels.



admin

Cette commande permet d'ajouter ou d'enlever des ressources (objet, expérience, endurance, etc.) à un joueur. Elle permet aussi d'afficher la fiche d'un joueur. <u>Utilisable uniquement par l'administrateur</u>.



Challenges

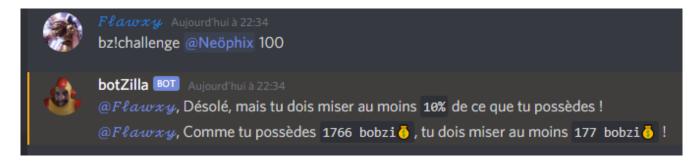
J'ai introduit les challenges lors d'une mise à jour de LootZilla.

Ce système permet aux joueurs de se défier entre eux. Une somme de bobzi\$\$ est mise en jeu et le vainqueur du challenge (un pile ou face) remporte la somme.

Les challenges peuvent être proposés n'importe quand (dans la limite du cooldown du joueur).

Les challenges peuvent être acceptés ou refusés.

Afin de ne pas abuser de cette commande et farmer l'expérience facilement, il faut miser au moins 10% de la somme que l'on possède.



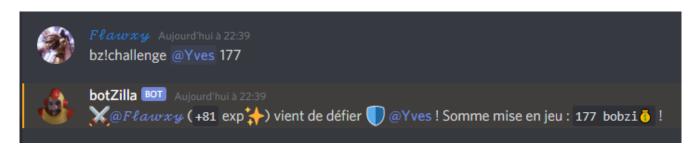
La fiche d'un joueur garde une trace des statistiques de ses challenges.



challenge

Permet de proposer un challenge à un joueur.

Cooldown: 60 minutes.



acceptChallenge

Permet d'accepter un challenge proposé.

Il n'y a aucune limite de temps pour accepter un challenge (les challenges sont stockés en base de données en attendant leur issue).

Une fois un challenge accepté, un « lancer de pièce » est effectué et le vainqueur remporte la mise.



refuseChallenge

Permet de refuser un challenge proposé. Aucun malus n'est appliqué.



myChallenges

Permet d'afficher la liste des challenges en attente d'un joueur.



Endurance, maps et aventures

Une nouvelle mise à jour a incorporé le système d'endurance.

Chaque joueur de niveau 1 possède 20 points d'endurance. Les points d'endurance sont restaurés à l'aide d'objets ou lors d'une montée de niveau.

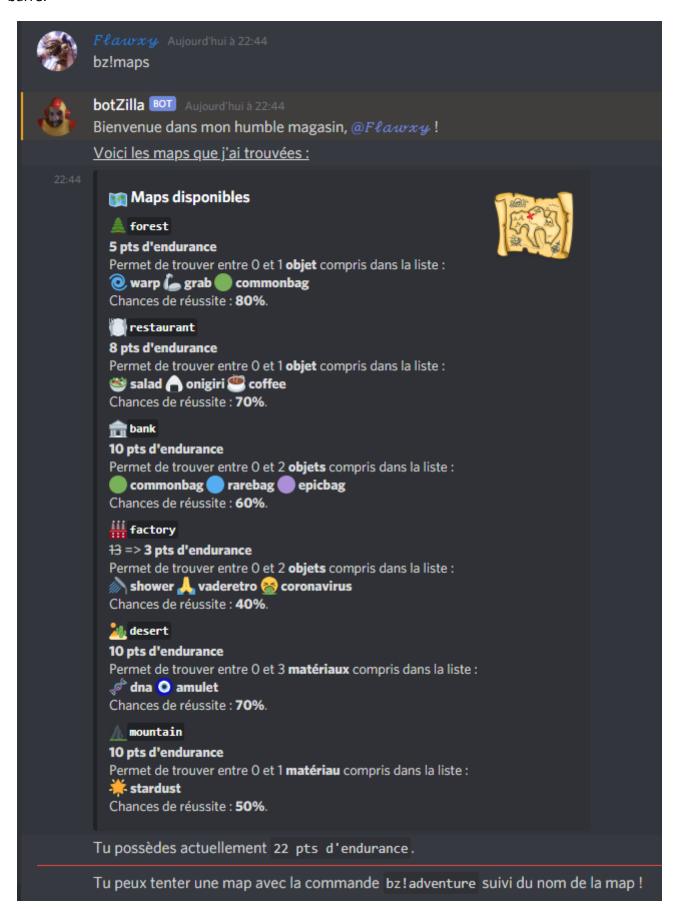
Chaque montée de niveau octroie au joueur +1 point d'endurance max.

L'endurance permet d'effectuer des aventures dans des maps*. Ce sont des expéditions qui permettent de récolter des objets.

*Le nom « map » a été choisi en l'honneur des « maps » de l'Atlas de Path of Exile.

maps

Permet d'afficher les maps disponibles. Si le joueur n'a pas assez d'endurance, le prix de la map est barré.

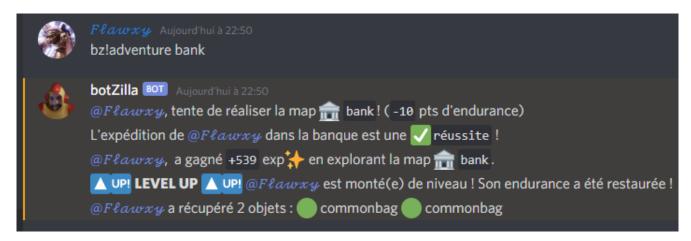


adventure

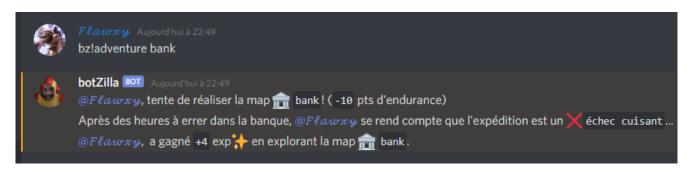
Permet de tenter une expédition sur une map. Une map réussie ne garantit pas forcément un gain d'objet. Une map ratée ne rapporte jamais rien.

Cooldown: 10 minutes.

Exemple de map réussie :



Exemple de map ratée :

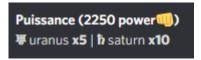


Craft, mutations et attaques

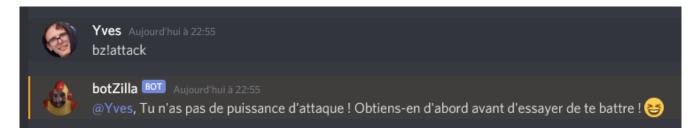
Lors de cette mise à jour, un nouveau type d'objet a été instauré : les matériaux.



Ils se récupèrent sur certaines maps et permettent de crafter des mutations qui donnent de la puissance d'attaque au joueur.



Chaque mutation possède une base connue mais la puissance finale est déterminée aléatoirement grâce à un multiplicateur.



Une fois qu'un joueur a acquis de la puissance, il peut attaquer les autres joueurs de manière aléatoire.

mutations

Permet d'afficher les mutations disponibles ainsi que leur coût de crafting.

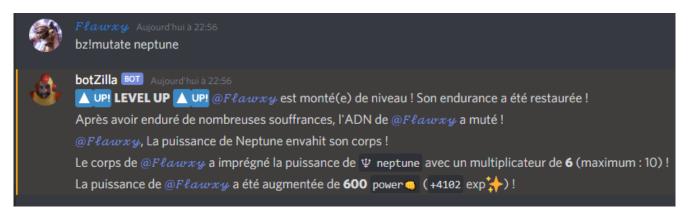


mutate

Permet d'obtenir une mutation et ainsi de la puissance d'attaque.

Cooldown: 30 minutes.

Exemple de mutation Neptune avec un multiplicateur obtenu de 6/10 :



myMaterials

Permet d'afficher les matériaux possédés par un joueur sans avoir à afficher sa fiche complète de personnage.



myMutations

Permet d'afficher les mutations possédées par un joueur ainsi que sa puissance totale sans avoir à afficher sa fiche complète de personnage.



power

Permet d'afficher la puissance totale d'un joueur sans avoir à afficher sa fiche complète de personnage.

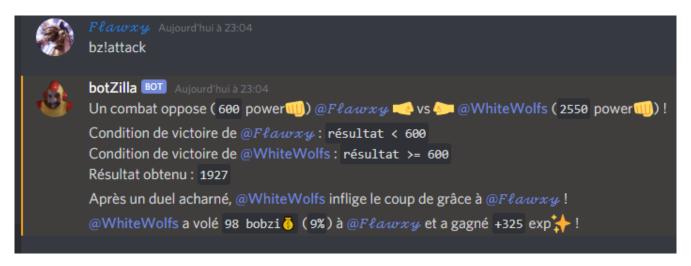


attack

Permet d'attaquer un joueur pris au hasard (les joueurs ne possédant pas encore de puissance d'attaque ne peuvent pas être choisis).

Les chances de réussite de l'attaque dépendent de la puissance de l'attaquant ainsi que de celle du défenseur.

Le joueur avec le moins de puissance a tout de même des chances de remporter le duel.



Améliorations

Les améliorations ont été intégrées dans la dernière mise à jour en date du jeu.

Elles permettent d'utiliser certains matériaux pour améliorer la qualité d'un sac.

Ainsi, après une amélioration réussie, un sac commun deviendra rare. Un sac rare deviendra épique. Un sac épique deviendra légendaire.

```
botZilla or Aujourd'hui à 23:08

@Flawxy, Voici tout ce que tu dois savoir sur l'amélioration d'objet!
Pour commencer, il te faut un ** stardust dans tes matériaux!

Tu dois ensuite choisir quel objet tu veux améliorer avec la commande bz!improve! Tu dois bien sûr posséder cet objet!
commonbag -> rarebag: 90% de chance de réussite

rarebag -> epicbag: 75% de chance de réussite

epicbag -> legendarybag: 50% de chance de réussite

En cas de réussite, ton objet verra sa qualité augmenter! Et le ** stardust sera consommé!

En cas d'échec, tu perdras l'objet de base, mais pas le ** stardust!
```

improve

Permet de tenter une amélioration d'objet. Si l'amélioration échoue, l'objet est perdu.

Cooldown: 10 minutes.

Exemple d'amélioration réussie :



improveHelp

Permet d'obtenir des informations sur le système d'amélioration.



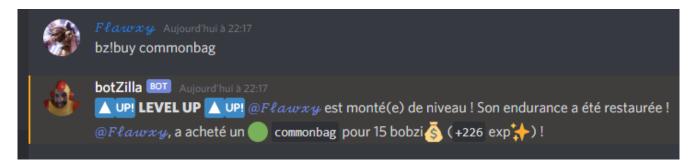
Annexes

Système de niveau et d'endurance

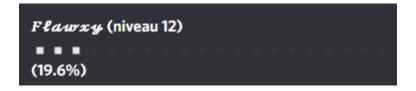
Chaque commande de jeu rapporte de l'expérience.

Quand un joueur monte de niveau, il voit son endurance restaurée et augmentée de 1 point.

Plus le joueur est haut niveau plus il a besoin d'expérience pour monter de niveau.

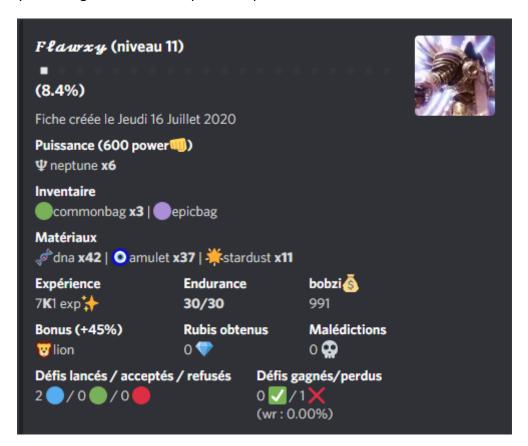


Une barre d'expérience dynamique est affichée sur la fiche d'un joueur.



Fiche de personnage, inventaires et limitations

La fiche d'un personnage est divisée en plusieurs parties.



Les joueurs ne peuvent posséder qu'un nombre limité du même objet dans leur inventaire. Cette limitation peut cependant être dépassée lors de gain d'objets au cours d'une map.



Les sacs de butin, les sacs maudits et les rubis

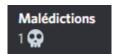


Il existe 4 qualités de sacs : commun, rare, épique, légendaire.



Les joueurs peuvent tomber sur un sac maudit. Ce dernier ne rapporte aucune pièce et maudit le joueur, lui infligeant un malus allant jusqu'à -25% de pièces en moins à chaque sac. Les malédictions sont cumulables et peuvent être enlevées à l'aide d'objets à acheter en boutique.

La fiche de personnage indique le nombre de malédiction qu'un joueur subit actuellement.

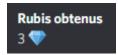




Les sacs réguliers ont une chance de contenir un rubis, qui rapporte énormément de bobzi\$\$.



La fiche de personnage garde une trace du nombre de rubis trouvés par un joueur.



Les items

Il existe de nombreux items à acheter au magasin avec des bobzi\$\$:



Il est indiqué combien d'exemplaires du même item un joueur peut avoir dans son inventaire.

Les animaux (buffs)

Les animaux sont des buffs permanents qui peuvent être trouvés dans des sacs non-maudits.



Il existe 4 raretés d'animaux : commun, peu commun, rare, très rare.



Cat, animal commun rapportant +5% de bobzi\$\$ trouvé dans les sacs.



Dog, animal commun rapportant +10% de bobzi\$\$ trouvé dans les sacs.



Rabbit, animal commun rapportant +15% de bobzi\$\$ trouvé dans les sacs.



Monkey, animal commun rapportant +15% de bobzi\$\$ trouvé dans les sacs.



Wolf, animal peu commun rapportant +20% de bobzi\$\$ trouvé dans les sacs.



Fox, animal peu commun rapportant +25% de bobzi\$\$ trouvé dans les sacs.



Tiger, animal rare rapportant +35% de bobzi\$\$ trouvé dans les sacs.



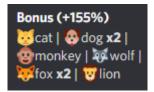
Lion, animal rare rapportant +45% de bobzi\$\$ trouvé dans les sacs.



Unicorn, animal très rare rapportant +50% de bobzi\$\$ trouvé dans les

sacs.

Les animaux sont stockés dans un inventaire à part et ne peuvent pas être perdus ou volés.



Merci d'avoir pris le temps de lire ce document!

Pour toute question ou suggestion, n'hésitez pas à me contacter :



vogel.yann@gmail.com



https://twitter.com/YannVogel



https://www.linkedin.com/in/yann-vogel-154920198/



https://github.com/Flawxy