



IUT DU LITTORAL CÔTE D'OPALE

UE216 COMPLÉMENTS INFORMATIQUES

---

## Discord MemeBot

---

*Enseignant :*  
Antoine Gamelin  
Département Informatique

*Groupe :*  
Isaac Ruchalski  
Guillaume Le Louarn  
Théo Helary

## Table des matières

<b>1</b>	<b>Introduction</b>	<b>2</b>
<b>2</b>	<b>Projet</b>	<b>3</b>
2.1	Bot Discord . . . . .	3
2.2	Outils . . . . .	3
<b>3</b>	<b>Contribution</b>	<b>4</b>
3.1	Commande d'aide . . . . .	4
3.2	Création de l'embed . . . . .	4
3.3	Rédaction du fichier README.md . . . . .	4
<b>4</b>	<b>Rétrospective</b>	<b>5</b>
4.1	Difficultés rencontrées . . . . .	5
4.2	Notions acquises . . . . .	5

## Table des figures

1	Logo Discord . . . . .	2
2	Exemple de Meme . . . . .	3
3	Embed aide . . . . .	4

# 1 Introduction



**Figure 1** – Logo Discord

Discord est un logiciel de VoIP et de messagerie instantanée, disponible pour tous les utilisateurs Windows, macOS, Android, iOS et Web.

Discord a gagné en popularité avec l'épidémie de COVID-19. Autrefois incontournable pour les joueurs, il est devenu un outil utile au quotidien dans l'enseignement, la vie en entreprise et autres structures contraintes au télétravail.

La communauté de Discord se montre assidûment active et rigoureuse quand il s'agit de relever des défis d'automatisation sur la plateforme. La programmation est un moyen de permettre à tous les utilisateurs de gagner en **efficacité** dans leur travail ou plus globalement améliorer leur expérience sur le logiciel au quotidien.

L'étude qui va suivre présentera le projet [Discord MemeBot](#), analysera ma contribution pour enfin conclure avec une rétrospective sur le travail réalisé

## 2 Projet

### 2.1 Bot Discord

Les bots Discord sont programmes **automatisés** qui ressemblent et agissent comme des utilisateurs. Ils répondent automatiquement aux événements et aux commandes sur Discord.

Le cœur de notre projet est un bot Discord. Il permet à l'utilisateur de récupérer des Memes (macro d'image accompagné d'un slogan humoristique), et de les afficher dans une conversation Discord avec un texte personnalisé.



Figure 2 – Exemple de Meme

### 2.2 Outils

Notre projet est développé avec [node.js](#) un environnement bas niveau qui nous permet d'exécuter du code JavaScript côté serveur.

Nous y avons ajouté le module [discord.js](#) pour pouvoir communiquer avec l'API Discord, représenté par le [Discord Developer Portal](#) qui donne la possibilité à n'importe quel utilisateur de concevoir des applications Discord.

Nous pouvons de même mentionner le package [node-rest-client](#) qui est essentiel au projet pour nous retourner des objets JSON en se connectant à une API. Le cœur de notre projet repose sur [Imgflip API](#), l'interface de programmation du site Imgflip, application populaire de diffusion et partage de Memes sur le Web.

## 3 Contribution

### 3.1 Commande d'aide

J'ai contribué au projet avec l'ajout d'une commande d'aide qui intervient tôt dans le code. Lorsque l'utilisateur va entrer la l'instruction " !help", un embed sera renvoyé dans le chat avec les commandes que nous avons ajouté à notre bot (" !getMemes" et " !makeMeme &id &text").

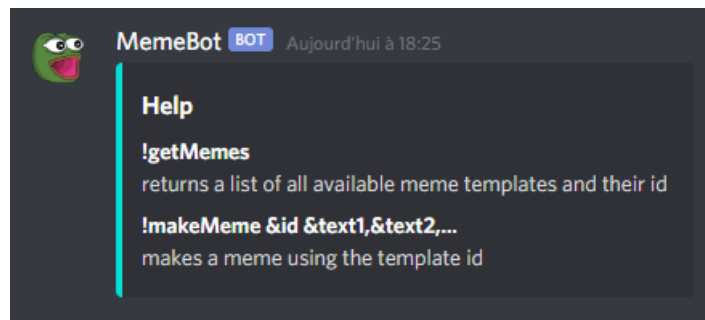


Figure 3 – Embed aide

La fonction makeEmbed que nous analyserons au point suivant, va nous permettre d'afficher les deux commandes à titre informatif grâce à un attribut name et value.

### 3.2 Création de l'embed

La création de l'embed passe par la fonction makeEmbedMemes précédemment énoncée. Une nouvelle instance de MessageEmbed() est requise, nous pouvons alors définir une couleur, un titre et les champs.

### 3.3 Rédaction du fichier README.md

Le fichier README.md est un élément incontournable du projet open source sur GitHub. Il s'agit d'un fichier pour introduire et expliquer de façon globale un projet.

Son contenu est variable selon l'application, dans notre cas nous introduisons rapidement le projet puis nous guidons l'utilisateur pour le premier lancement (clone, installation des modules node, ajout dans le serveur Discord). Il est important d'utiliser l'échappement de GitHub "sh pour signifier à l'utilisateur qu'il s'agit de lignes de commandes à interpréter.

Un aperçu de la mission de notre bot est de même apprécié, nous avons ajouté un exemple de meme avec les commandes à exécuter dans le chat pour obtenir le même résultat.

## 4 Rétrospective

### 4.1 Difficultés rencontrées

Je n'avais jamais contribué à un projet de bot Discord, il était important pour moi de me documenter un maximum sur le module discord.js pour comprendre comment cela fonctionne avant d'écrire mes premières lignes de codes.

Je n'ai pas l'habitude de rédiger des fichiers README.md aussi complets, je développe peu souvent des projets voués à l'open source, j'ai donc dû prendre le temps d'acquérir les meilleurs pratiques pour rédiger un fichier de présentation **synthétique**.

### 4.2 Notions acquises

L'utilisation des API est **incontournable** dans le métier de développeur web. J'ai essentiellement pratiqué les notions de l'API avec le framework Angular. Avec le projet Discord MemeBot, j'ai notamment appris à gérer les différents outils nécessaires pour me servir d'une API sans frameworks.

Le développement d'un bot Discord m'a apporté beaucoup de nouvelles connaissances pratiques notamment avec les embed et la détection de commandes rentrées par l'utilisateur.

De façon plus globale, il était important de rédiger du code efficace pour garder les marques d'un projet open source, un code qui serait exploitable pour l'évolution et l'ajout de fonctionnalités en interne ou par d'autres développeurs.

Dans la même idée, la rédaction de fichier README.md me sera utile dans chaque projet que je serais amené à développer afin de partager une présentation claire et facile d'accès aux utilisateurs.

Le travail fourni dans ce module m'a fait découvrir en parallèle le langage de programmation LaTeX avec son utilisation sur Overleaf.