

Routes :

Envoyer un message.

POST /Message

body.

idUser , *isVisible* , *message* , *color* , *effect* , *font* , *pseudo* , *idChatLog*

Récupérer tous les messages.

GET /Messages

body.

idChatLog

Récupérer message par son id.

GET /Message

body.

idMessage

Modifier message.

PATCH /Message

body.

idMessage , *isVisible* , *message* , *color* , *effect* , *font*

Modifier message.

DELETE /Message

body.

idMessage

La configuration globale du projet est dans ./env

```
.env.dev  X  .env.prod  JS Index.js  JS Routes.js
G: > DATA_2019_08 > GIT > ChatlogTxtAPI > env > .env.dev
1  NODE_ENV=development
2  APP_NAME=Chatlog Txt (Dev version)
3  API_URL=127.0.0.1
4  PORT=2299
5  VERS=2020-10-21_1
6  SOMEONE_WRITING_ON_TXTFILE=false
7  TXT_FILE=chatlog\chatlog.txt
8  DEFAULT_COLOR_MESSAGE=B00000
9  DEFAULT_FONT_MESSAGE=
10 DEFAULT_EFFECT_MESSAGE=
11 DEFAULT_PSEUDO=Anonymous
12 DEFAULT_VISIBLE_MESSAGE=true
13 DEFAULT_IDCHATLOG=home
14 TIMEOUT_ACTIONFILE=50
```

Il y a 2 fichiers de conf, la version local (dev) & la version prod

Structure D'un message dans fichier texte

Les structures s'incrémentes.

```
structMessage_Default = { Blocks: [
  {idBlock: "STARTBLOCK", length: 8, val: "<START_>"},
  {idBlock: "version", length: 32, val: ""},
  {idBlock: "idMessage", length: 32, val: ""},
  {idBlock: "idUser", length: 32, val: ""},
  {idBlock: "isVisible", length: 32, val: "1"}, // <- on front
  {idBlock: "idChatlog", length: 64, val: ""},
  {idBlock: "pseudo", length: 64, val: ""},
  {idBlock: "color", length: 7, val: ""},
  {idBlock: "effect", length: 64, val: ""},
  {idBlock: "font", length: 64, val: ""},
  {idBlock: "message", length: 3000, val: ""},
  {idBlock: "timestamp", length: 50, val: ""},
  {idBlock: "info1", length: 32, val: ""},
```

```
{idBlock: "info2", length: 32, val: ""},  
{idBlock: "info3", length: 32, val: ""},  
{idBlock: "info4", length: 64, val: ""},  
{idBlock: "info5", length: 64, val: ""},  
{idBlock: "info6", length: 64, val: ""},  
{idBlock: "info7", length: 128, val: ""},  
{idBlock: "info8", length: 128, val: ""},  
{idBlock: "ENDBLOCK", length: 8, val: "<END___>"},  
}};
```

Pour écrire dans le fichier texte, on utilise les variables d'environnements afin de « bloquer » l'accès de l'API durant l'écriture, peu importe où on se trouve.

todo: ajouter attribut "size" à message

POUR EXECUTER LE PROJET

npm run dev

(ou "npm run prod")