
StegyJpgBot Documentation

Release 1.1

Gorbunov Pavel

Dec 03, 2017

Оглавление

1	StegoBot package	1
1.1	main module	1
1.2	algorithm module	5
1.3	datawork module	8
1.4	Strings module	9
2	Indices and tables	11
	Python Module Index	13

STEGOBOT PACKAGE

1.1 main module

Этот модуль описывает telegram-бота “SteganyBot”

`main.ChangeFlags(num_id, *args)`

Изменение состояния флага

Parameters

- `num_id` (int) – номер чата
- `*args` – Наименования флагов

`main.InitDict()`

Добавление флагов в `UsersFlags` для пользователей, чьи данные есть в базе данных

`main.SetArgs(num_id, path, *args)`

Инициализация флагов данными

Parameters

- `num_id` (int) – номер чата
- `path` (str) – путь до файла
- `*args` – Наименования флагов

`class main.SteganyBot(Token)`

Bases: object

Класс реализует telegram-бота “@StegyJpgBot”

`__token`

str – токен бота

`updater`

telegram.ext.Updater – объект представляет входящие обновления

`dispatcher`

telegram.ext.Dispatcher – объект представляет обработчик обновлений

`AddHandler()`

Добавление хэндлеров в Диспатчер

`AnswerLog(bot, update)`

Отправление сообщение с информацией о пользователе разработчику бота

Parameters

- `bot` (telegram.Bot) – хэндлер бота

- update (telegram.ext.Updater) – обновления

AskChooseUser(bot, update)

Попросить пользователя:

- Загрузить картинку
- Написать username адресата

Parameters

- bot (telegram.Bot) – хэндлер бота
- update (telegram.ext.Updater) – обновления

AskPicture(bot, update)

Запрос оправления фотографии для:

- Расшифрования
- Отправления другому пользователю

Parameters

- bot (telegram.Bot) – хэндлер бота
- update (telegram.ext.Updater) – обновления

AskText(bot, update)

Запрос сообщения для шифрования в png-файл

Parameters

- bot (telegram.Bot) – хэндлер бота
- update (telegram.ext.Updater) – обновления

CheckUser(bot, update, user)

Оправить пользователь png-файл

Parameters

- bot (telegram.Bot) – хэндлер бота
- update (telegram.ext.Updater) – обновления
- user (str) – username пользователя

Note: Если пользователя с переданным в базе данных нет, то сообщить об этом пользователю и прислать ему png-файл.

ClearDir(path, num=5)

Удаление файлов в указанной директории

Parameters

- path (str) – директория
- num (int, optional) – ограничение на количество файлов

Decrypt(bot, update)

Метод обработки команды бота /decrypt

Parameters

- bot (telegram.Bot) – хэндлер бота
- update (telegram.ext.Updater) – обновления

DecryptMessage(bot, update, number)

Расшифровка сообщения из png-файла

Parameters

- bot (telegram.Bot) – хэндлер бота
- update (telegram.ext.Updater) – обновления
- number (int) – будущий номер файла в папке

Echo(bot, update)

Ответ пользователю им же сообщением

Parameters

- bot (telegram.Bot) – хэндлер бота
- update (telegram.ext.Updater) – обновления

Encrypt(bot, update)

Метод обработки команды бота /encrypt

Parameters

- bot (telegram.Bot) – хэндлер бота
- update (telegram.ext.Updater) – обновления

EncryptMessage(bot, update, letter)

Шифрование сообщение, полученное от пользователя, в png-файл

Parameters

- bot (telegram.Bot) – хэндлер бота
- update (telegram.ext.Updater) – обновления
- letter (str) – полученное сообщение

Error(bot, update, error)

Обработка ошибок

Parameters

- bot (telegram.Bot) – хэндлер бота
- update (telegram.ext.Updater) – обновления

ExplicitPhoto(bot, update)

Запрещена обработка фотографий

Parameters

- bot (telegram.Bot) – хэндлер бота
- update (telegram.ext.Updater) – обновления

GetDoc(bot, update)

Обработка присланных документов пользователем

Parameters

- bot (telegram.Bot) – хэндлер бота
- update (telegram.ext.Updater) – обновления

GetId(bot, update)

Получение идентификатора чата

Parameters

- bot (telegram.Bot) – хэндлер бота
- update (telegram.ext.Updater) – обновления

Returns идентификатор чата с пользователем

Return type int

GetLength(bot, update)

Получение информации о возможной длины сообщения в переданную картинку

Parameters

- bot (telegram.Bot) – хэндлер бота
- update (telegram.ext.Updater) – обновления

GetText(bot, update)

Обработка текстовых сообщений от пользователя

Parameters

- bot (telegram.Bot) – хэндлер бота
- update (telegram.ext.Updater) – обновления

Help(bot, update)

Метод бота “/help”

Parameters

- bot (telegram.Bot) – хэндлер бота
- update (telegram.ext.Updater) – обновления

Info(bot, update)

Метод бота “/info”

Parameters

- bot (telegram.Bot) – хэндлер бота
- update (telegram.ext.Updater) – обновления

MakePath(folder)

Создание строки-пути

Parameters folder (str) – папка назначения

Returns Путь до папки назначения

Return type str

ParseUpdate(text)

Разделение строки на отдельные части

Parameters text (str) – строка обновлений от пользователя

Returns

данные о пользователе (username, имя, фамилия, номер чата)

Return type list

Picture(bot, update)

Inline-клавиатура в ответ на загрузку картинки со стороны пользователя

Parameters

- bot (telegram.Bot) – хэндлер бота
- update (telegram.ext.Updater) – обновления

Play(bot, update)

Метод бота “/play” :param bot: хэндлер бота :type bot: telegram.Bot :param update: обновления :type update: telegram.ext.Updater

ProcessButton(bot, update)

Обработка событий от inline-клавиатуры

Parameters

- bot (telegram.Bot) – хэндлер бота
- update (telegram.ext.Updater) – обновления

Refresh()

Обновление списка с флагами

Restore()

Обнуление флагов для encrypt, decrypt, upload, depart

SendPicToUser(bot, update)

Отправление png-файла пользователю

Parameters

- bot (telegram.Bot) – хэндлер бота
- update (telegram.ext.Updater) – обновления

SendPicture(bot, update, user=None)

Отправление png-файла:

- Пользователю, который зашифровал сообщение
- Пользователю, которую хотят отправить сообщение

Parameters

- bot (telegram.Bot) – хэндлер бота
- update (telegram.ext.Updater) – обновления
- user (str, optional) – username пользователя-адресата

Start(bot, update)

Метод бота “/start”

Args: bot (telegram.Bot): хэндлер бота update(telegram.ext.Updater): обновления

StartBot()

Запуск telegram-бота

Upload(bot, update)

Метод обработки команды бота /upload

Parameters

- bot (telegram.Bot) – хэндлер бота
- update (telegram.ext.Updater) – обновления

1.2 algorithm module

В этом модуле реализована работа с png-файлом и механизм внедрения информации в картинку с помощью алгоритма Куттера-Джордана-Боссена

class algorithm.Container

Bases: object

Класс предоставляет интерфейс для шифрования/зашифрования сообщение в png-файл.

Note: Класс недоступен пользователю

MyImage

Picture – Объект класса 'Picture'

Pathfile

str – Путь до png-файла

UserName

str – Username пользователя

CheckLength()

Получение возможной длины сообщения

Returns возможная длина сообщения

Return type int

Decrypt()

Расшифрование сообщения из png-файла

Returns

Статус-строка str: {

 "sorry, there are undecoded byte in decrypt message", "Ops, are you sure with
 your choice?))) I think, i haven't got this picture before"

}

Return type str

Encrypt(text)

шифрование переданного сообщения в png-файл

Parameters text (str) – сообщение пользователя

ParseStr(text)

Декодирование байтов в utf8

Parameters text (bitarray) – byte-строка

Returns str сообщение

Save()

Сохранение в png-файл

Returns Путь до файла с зашифрованным сообщением

Return type str

SetModule(path)

Установка значений для полей Pathfile и MyImage

Parameters path (str) – путь до файла

SetUser(name)

Установка значения для поля UserName

Parameters name (str) – username пользователя

class algorithm.Picture(path)

Bases: object

Это класс предоставляет работу с файлом изображения

Note: Этот класс недоступен пользователю.

image

Image – Переданный png-файл для контейнера

width

int – Ширина картинки

height

int – Высота картинки

draw

ImageDraw – Перо для работы с png-файлом

Parameters path (str) – Путь до png-файла

ChangePix(text)

Внедрение сообщения в png-файл с помощью алгоритма

:param (bitarray): исходное сообщение в битах

Returns Статус о завершении шифровки

Return type str

Decto256(num)

Перевод десятичного числа в 256-ую систему счисления

Parameters num (int) – исходное десятичное число

Returns число в 256-ой системе счисления

Return type list

GetBit()

Расшифровка сообщения из png-файла, внедренного алгоритмом
Боссена.

Куттера-Джордана-

Проходим по всему png-файлу и достаём биты сообщения.

Returns биты сообщения

Return type bitarray

GetLen()

Получение длины сообщения, зашифрованного в png-файле

Returns Длина сообщения, зашифрованного в файле

Return type int

LengthText()

Максимальная длина возможного сообщения

Returns: int Длина сообщения

Numtodec(num)

Перевод 256-ого числа в десятичную систему счисления

Parameters num (list) – Исходное число в 256-ой системе счисления

Returns Число в десятичной системе счисления

Return type int

SaveChange(path)

Сохранение результатов шифрования в png-файл

Parameters path (str) – путь до файла

Returns Путь до файла с зашифрованным сообщением

Return type str

SaveLen(lenT)

Сохранение длины сообщения в картинке

Parameters lenT (int) – длина сообщения, внедряемого в png-файл

WritePix(flag)

Запись всех пикселей в файл

Parameters flag (str) – составляющая названия файла

1.3 datawork module

Этот модуль соержит класс DataBase, реализующий работу с базой данных

class datawork.DataBase

Bases: object

Реализует интерфейс работы с базой данных

Note: У пользователя прямого доступа к классу нет

conn

connection – Дескриптор подключения к базе данных

cur

cursor – Перео для выполнения PostgreSQL команд.

Parameters

- (`cur`) – class: connection): Дескриптор подключения к базе данных. Создается подключение между программой и базой данных, созданная в приложении в сервисе Heroku.
- (– class: cursor): Перо для выполнения PostgreSQL команд. Создается и связывается с текущим соединением к базе данных.

AddUser(user)

Добавление пользователя в базу данных

Parameters user (list) – Данные о пользователе: + Username + Имя + Фамилия + Номер чата

Returns

Возвращает строку статуса:

```
{
    "add username success": успешное добавление username'а пользователя в базу данных
    "notice user": у пользователя нет username'а "already exist": данные пользователя в базе данных "successful addUser": успешное добавление данных пользователя в базу данных
}
```

Return type str

AddUserName(user)

Обновление поля username у пользователя

Parameters user (list) – данные о пользователе

AskUserName(username)

Проверка наличия пользователя с переданным username'ом

Parameters username (str) – Username пользователя

Returns True если есть, False иначе.

Return type bool

CheckFullUser(user)

Проверка наличия username'а пользователя

Parameters user (list) – данные о пользователе

Returns True если есть username, False в противном случае

Return type bool

CheckUser(id)

Проверка наличия данных о пользователе в базе данных

Agrs: id (int): номер чата

Returns True если есть, False в противном случае

Return type bool

GetId(user)

Возвращает номер чата переданного пользователя

Parameters user (list) – данные о пользователе

Returns Номер чата с данным пользователем

Return type int

1.4 Strings module

Этот модуль соержжит статус-строки

:param startGreet str:

- строка-приветствие

:param info str:

- Информация о боте

:param greeting2 str:

- строка при вызове /play

:param helpString str:

- строка при вызове /help

:param answerPhoto str:

- ответ на получение ботом
файла типа “картинка”

:param addUsername str:

- уведомление об отсутствии username’a

:param getPic str:

- получение документа без
использования /decrypt

:param lengthMessage str:

- уведомление о длине сообщения

:param askUser str:

- строка о запросе username’a

:param addFriendUser str:

- уведомление, что пользователь
получатеть еще не пользовался ботом

:param afterEncrypt str:

- строка после завершения encrypt’a

:param messageUser str:

- уведомление о получении сообщения
от пользователя

:param uploadpicture str:

- запрос о загрузки документа

:param uploadSuccess str:

- статус-строка об успешной загрузки

:param AnswerStatus dict:

- словарь статус-состояний после
добавления пользователя в базу данных

INDICES AND TABLES

- `genindex`
- `modindex`
- `search`

PYTHON MODULE INDEX

a

algorithm, 5

d

datawork, 8

m

main, 1

s

Strings, 9